

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Siswa

Menurut Prayitno dalam falentini, dik (2013:48-49). “siswa SMA adalah mencapai kematangan dalam pilihan karier yang akan dikembangkan lebih lanjut”. Menurut Prasetyo, dik dalam UU Sisdiknas No 20 Pasal 1 ayat 4 (2013:7). “siswa atau peserta didik adalah anggota masyarakat yang berusaha mengembangkan potensi diri melalui proses pembelajaran yang tersedia pada jalur, jenjang, dan jenis pendidikan tertentu, mereka yang secara khusus diserahkan oleh kedua orang tuannya untuk mengikuti pembelajaran yang diselenggarakan di sekolah, dengan tujuan untuk menjadi manusia yang berilmu pengetahuan, berketerampilan, berpengalaman, berkepribadian, berakhlak mulia, dan mandiri

Berdasarkan pengertian-pengertian di atas, dapat disimpulkan bahwa siswa adalah sekelompok orang dengan usia tertentu yang secara khusus di serahkan oleh kedua orang tuannya untuk mengikuti proses pembelajaran yang tersedia pada jalur jenjang, dan jenis pendidikan tertentu.

2.2 Sistem

Dalam merancang sebuah aplikasi atau sistem informasi selalu berhubungan dengan aspek sistem. Pembahasan mengenai sistem sangat penting karena di dalamnya membahas tentang alur proses. Pengertian sistem menurut Mulyadi (2008) adalah “ sekelompok unsur yang erat berhubungan satu dengan lainnya, yang berfungsi bersama-sama untuk mencapai tujuan tertentu “.

Pengertian sistem menurut Tata Sutabri (2012:6) pada buku Analisis Sistem Informasi, pada dasarnya sistem adalah sekelompok unsur yang erat berhubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu.

Selanjutnya menurut McLeod (2004) dikutip oleh Yakub dalam buku Pengantar Sistem Informasi (2012:1) mendefinisikan sistem adalah sekelompok elemen-elemen yang terintegrasi dengan tujuan yang sama untuk mencapai tujuan. Sistem juga merupakan satu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan satu kegiatan atau untuk tujuan tertentu.

Pengertian Menurut Jogianto (2005:2) pada buku Analisis dan Desain Sistem Informasi mendefinisikan sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi.

Pengertian sistem menurut Widjajanto (2008) adalah sesuatu yang memiliki bagian-bagian yang saling berinteraksi untuk mencapai tujuan tertentu melalui tiga tahapan yaitu *input*, proses dan *output*.

Dari wacana yang ada dapat diambil kesimpulan bahwa sistem merupakan kumpulan suatu komponen yang berbeda fungsi dan jenis dan saling berkaitan atau berhubungan untuk melakukan tujuan yang sama.

2.2.1 Elemen Sistem

Menurut McLeod yang dikutip oleh Yakub (2012:3) tidak semua sistem memiliki kombinasi elemen-elemen yang sama, tetapi susunan dasarnya sama.

Elemen-elemen yang terpadat dalam sistem ditandai dengan adanya :

1. Tujuan

Tujuan ini menjadi motivasi yang mengarahkan pada sistem, karena tanpa tujuan yang jelas sistem menjadi tidak terarah dan tak terkendali.

2. Masukan

Masukan (*input*) sistem adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk diproses. Masukan dapat berupa hal-hal berwujud maupun yang tidak berwujud. Masukan berwujud adalah bahan mentah, sedangkan yang tidak berwujud adalah informasi.

Proses merupakan elemen yang bertugas melakukan perubahan atau transformasi dari masukan / data menjadi keluaran / informasi yang berguna dan lebih bernilai.

3. Keluaran

Keluaran (*output*) merupakan hasil dari *input* yang sudah dilakukan pemrosesan sistem dan keluaran dapat menjadi masukan untuk subsistem lain.

4. Batasan

Batasan (*boundary*) sistem adalah pemisah antara sistem dan daerah di luar sistem. Selain itu juga sebagai batasan-batasan dari tujuan yang akan dicapai oleh sistem. Batasan sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem.

5. Umpan Balik

Umpan balik ini digunakan untuk mengendalikan masukan maupun proses.

Umpan balik juga bertugas mengevaluasi bagian dari yang dikeluarkan.

Tujuannya untuk mengatur agar sistem berjalan sesuai dengan tujuan

6. Lingkungan

Lingkungan adalah segala sesuatu yang berada di luar sistem.

2.2.2 Klasifikasi Sistem

Menurut Yakub (2012 : 4) pada buku Pengantar Sistem Informasi, Sistem dapat diklasifikasikan dari beberapa sudut pandang di antaranya :

1. Sistem abstrak (*Abstract System*)

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Sistem teologi yang berisi gagasan tentang hubungan manusia dengan Tuhan merupakan contoh *abstract system*.

2. Sistem fisik (*physical system*)

Sistem fisik adalah sistem yang ada secara fisik, Sistem komputer, sistem akuntansi, sistem produksi, sistem sekolah, dan sistem transportasi merupakan contoh *physical system*.

3. Sistem tertentu (*deterministic system*)

Sistem tertentu adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi, interaksi antara bagian dapat dideteksi dengan pasti sehingga keluarannya dapat diramalkan. Sistem komputer sudah diprogramkan, merupakan contoh *deterministic system* karena program komputer dapat diprediksi dengan pasti.

4. Sistem tertentu (*probabilistic system*)

Sistem tak tentu adalah suatu sistem yang kondisi masa depannya tidak dapat diprediksikan karena mengandung unsur probabilitas. Sistem arisan merupakan contoh *probabilistic system* karena sistem arisan tidak dapat diprediksikan dengan pasti.

5. Sistem tertutup (*Close System*)

Sistem tertutup merupakan sistem yang tidak bertukar materi, informasi, atau energi dengan lingkungan. Sistem ini tidak berinteraksi dan tidak dipengaruhi oleh lingkungan, misalnya reaksi kimia dalam tabung terisolasi

6. Sistem terbuka (*Open System*)

Sistem ini adalah sistem yang berhubungan dengan lingkungan dan dipengaruhi oleh lingkungan. Sistem perdagangan merupakan contoh *open system*, karena dapat dipengaruhi oleh lingkungan

2.2.3 Sistem Informasi

Pemahaman mengenai sistem informasi harus terlebih dahulu memahami dua komponen utama penyusun dari sistem informasi, yaitu sistem dan informasi. Pengertian sistem seperti halnya pada penjelasan sebelumnya tinggal memahami pengertian dari informasi. Informasi merupakan data yang diolah atau diproses sedemikian rupa sehingga menghasilkan suatu yang bermanfaat. Seperti halnya menurut Abdul Kadir (2003) bahwa “Informasi adalah data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakan data tersebut”.

Menurut McLeod dikutip oleh Yakub (2012:8) pada buku Pengertian Sistem Informasi, informasi adalah data yang diolah menjadi bentuk lebih berguna dan lebih berarti bagi yang menerimanya. Sedangkan menurut Tata Sutabari (2012:22) pada buku Analisis Sistem Informasi, Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan.

Sistem informasi memiliki komponen-komponen penyusun yang saling berinteraksi satu sama lain. Menurut Abdul Kadir (2003) komponen-komponen sistem informasi antara lain :

1. Perangkat Keras (*Hardware*)

Mencakup peranti fisik seperti *computer* dan *printer*.

2. Perangkat lunak (*Software*)

Sekumpulan instruksi yang memungkinkan perangkat keras untuk dapat memperoleh data.

3. Prosedur

Sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data dan pembangkitan keluaran yang dikehendaki.

4. Orang

Semua pihak yang bertanggung jawab dalam pengembangan sistem informasi, pemrosesan dan penggunaan keluaran sistem informasi.

5. Basis Data (*Database*)

Sekumpulan tabel, hubungan dan lain-lain yang berkaitan dengan penyimpanan data.

6. Jaringan komputer dan komunikasi data

Sistem penghubung yang memungkinkan sumber (*Resources*) dipakai secara bersamaan atau diakses oleh sejumlah pemakai.

Pembahasan mengenai sistem informasi merupakan sesuatu hal yang luas karena sistem informasi sendiri memiliki klasifikasi sesuai jenis dan fungsinya. Menurut Abdul Kadir (2003) dalam bukunya. Sistem informasi digolongkan ke dalam tiga area, yaitu :

1. Sistem informasi area fungsional

Sistem informasi dalam area fungsional diperuntukkan untuk fungsi operasional di dalam organisasi. Fungsi dari sistem informasi tingkat fungsional ini untuk memberikan informasi bagi kelompok orang yang berada ada bagian tertentu dalam perusahaan. Beberapa contoh sistem informasi fungsional yang umum seperti, sistem informasi akuntansi, sistem informasi pemasaran atau sistem informasi SDM.

2. Sistem informasi berdasarkan dukungan

Sistem informasi berdasarkan dukungan ini merupakan suatu benatu evolusi dari sistem informasi yang ada sebelumnya. Evolusi sistem tersebut juga menghasilkan suatu perubahan dari fungsi dan pemakai. Pada sistem informasi berdasarkan dukungan digunakan oleh kalangan manajer dan profesional di perusahaan. Secara sistem ini dapat membantu pemakai dalam mengambil suatu keputusan yang terstruktur dan semi-terstruktur, karena mengacu pada model atau perangkat untuk menganalisa informasi. Contoh dari sistem informasi berdasarkan dukungan antara lain, Sistem Informasi Manajemen (SIM), Sistem Pendukung Keputusan (SPK), Sistem Otomatisasi Perkantoran.

3. Sistem informasi pengetahuan

Sistem informasi pengetahuan (*Knowledge Information System*) merupakan sistem informasi yang mendukung aktivitas pekerja berpengetahuan. Sistem ini mengadopsi pola pikir para ahli atau pakar untuk diimplementasikan ke dalam sistem berbasis komputerisasi. Sistem ini difungsikan untuk pemakai (*user*) yang hendak memecahkan masalah yang membutuhkan ilmu pakar. Contoh dari sistem informasi ini antara lain, sistem pakar diagnosis penyakit tanaman

kedelai berbasis komputerisasi atau sistem pakar berbasis komputer untuk mengetahui kerusakan komputer.

2.3 Indeks Massa Tubuh (IMT)

Indeks massa tubuh (IMT) adalah metode yang murah, mudah dan sederhana untuk menilai status gizi pada seorang individu, namun tidak dapat mengukur lemak tubuh secara langsung. Pengukuran dan penilaian menggunakan IMT berhubungan dengan kekurangan dan kelebihan status gizi. Gizi kurang dapat meningkatkan resiko terhadap penyakit infeksi dan gizi lebih dengan akumulasi lemak tubuh berlebihan meningkatkan risiko menderita penyakit regeneratif.

IMT merupakan rumus matematis yang dinyatakan sebagai berat badan dalam kilogram (kg) dibagi dengan tinggi badan di kuadratkan (m^2) dan tidak terka pada jenis kelamin. IMT secara signifikan berhubungan dengan kadar lemak tubuh total sehingga dapat dengan mudah mewakili kadar lemak tubuh. Saat ini, IMT secara internasional diterima sebagai alat untuk mengidentifikasi kelebihan berat badan dan obesitas. (Hill, 2005).

Sejak pertengahan tahun 1980-an, prevalensi obesitas telah meningkat secara tetap dan terjadi baik di negara-negara barat dan negara-negara non-barat, dan tidak ada indikasi bahwa angka ini akan berkurang. Orang-orang dengan IMT lebih yaitu kelebihan berat badan dan obesitas pada hakikatnya meningkatkan morbiditas dan mortalitas akibat hipertensi, stroke, penyakit jantung koroner, dyslipedemia dan diabetes mellitus tipe 2.

Penggunaan IMT hanya berlaku untuk orang dewasa yang berusia 18 tahun ke atas. IMT tidak diterapkan pada bayi, anak, remaja, dan ibu hamil dan olahragawan. Disamping itu pula IMT tidak dapat diterapkan dalam keadaan

khusus (penyakit) lainnya seperti edeme, asites dan hepatomegali (Supariasa, 2001).

Rumus untuk mengetahui nilai IMT dapat dihitung dengan rumus metrik berikut :

$$\text{IMT} = \frac{\text{Berat badan (Kg)}}{[\text{Tinggi badan (m)}]^2}$$

Tabel 2.1 Klasifikasi IMT berdasarkan WHO

IMT (kg/m ²)	Klasifikasi
<16	Kurang energi protein III
16-16.9	Kurang energi protein II
17.0-18.5	Kurang energi protein I (<i>underweight</i>)
18.6-24.9	Normal
25.0-29.9	Kelebihan berat badan (<i>overweight</i>)
30.0-34.9	Obesitas I
35.0-39.9	Obesitas II
>40.0	Obesitas III

Tabel 2.2 Klasifikasi IMT berdasarkan Depkes RI (1994)

IMT (kg/m ²)	Kategori
<17.0	Kekurangan berat badan tingkat berat
17.0-18.4	Kekurangan berat badan tingkat ringan
18.5-25.0	Normal
25.1-27.0	Kelebihan berat badan tingkat ringan
>27.0	Kelebihan berat badan tingkat berat

Jika seseorang termasuk kategori :

1. IMT < 17,0: keadaan orang tersebut disebut kurus dengan kekurangan berat badan tingkat berat atau Kurang Energi Kronis (KEK) berat.
2. IMT 17,0 – 18,4: keadaan orang tersebut disebut kurus dengan kekurangan berat badan tingkat ringan atau KEK ringan.

Contoh cara menghitung IMT:

Eko dengan tinggi badan 148 cm, mempunyai berat badan 38 kg.

$$\frac{38}{(1,48 \times 1,48) \text{ m}} = 17,3$$

Status gizi Eko adalah kurus tingkat ringan. Eko dianjurkan menaikkan berat badan sampai menjadi normal antara 41- 54 kg dengan IMT 18,5 – 25,0.

2.4 Dasar-dasar *Analytical Hierarchy Process* (AHP)

Menurut Turban (2005), *Analytical Hierarchy Process* adalah suatu metode analisis dan sintesis yang dapat membantu proses pengambilan keputusan. AHP merupakan alat pengambilan keputusan yang *powerful* dan fleksibel, yang dapat membantu keputusan di mana aspek-aspek kualitatif dan kuantitatif terlibat dan keduanya harus dipertimbangkan. Dengan mereduksi faktor-faktor kompleks menjadi rangkaian *one on one comparisons* dan kemudian mensintesa hasil-hasilnya, maka AHP tidak hanya membantu orang dalam memilih keputusan yang tepat, tetapi juga dapat memberikan pemikiran/alasan yang jelas dan tepat.

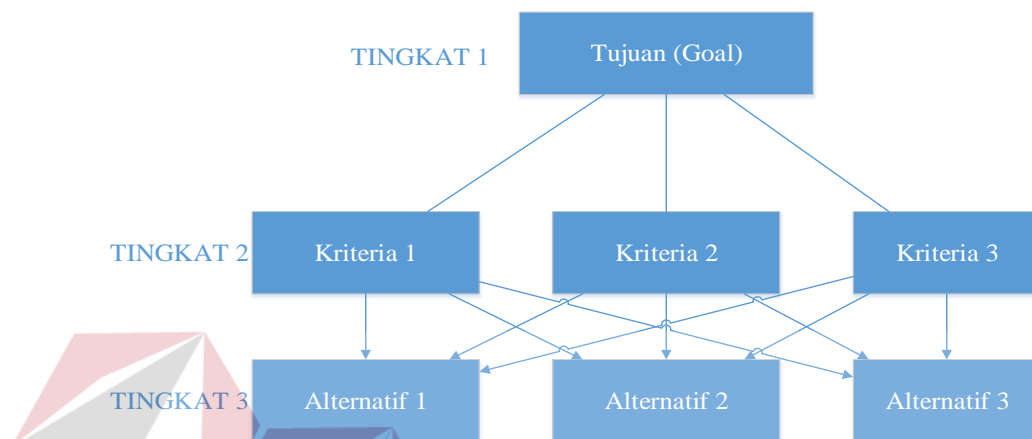
AHP juga dimungkinkan untuk pengambilan keputusan yang mengandalkan intuisi sebagai input utama. Untuk hal ini, perlu diperhatikan bahwa intuisi harus datang dari pengambilan keputusan yang cukup informasi, pengetahuan dan pengalaman, serta memahami masalah keputusan yang dihadapi. Metode ini dapat digunakan untuk menentukan skala rasio dari suatu perbandingan pasangan yang diskrit maupun kontinu. Nilai perbandingan dapat diambil dari ukuran actual ataupun dari skala dasar yang mencerminkan kekuatan perasaan dan preferensi relative. (Martiniatyas, 2007)

Menurut Bernard (1996) terdapat beberapa prinsip yang digunakan untuk menyelesaikan suatu masalah dengan menggunakan metode AHP adalah sebagai berikut :

1. *Decomposition*

Setelah didefinisikan, maka dilakukan tahap *decomposition* yaitu memecahkan persoalan yang utuh menjadi unsur-unsur di bawahnya. Hal ini yang menjadi alasan proses tersebut dinamakan hierarki.

Contoh :



Gambar 2.1 Struktur bagan AHP (Bernard W, 2005)

2. Comparative Judgement

Prinsip *comparative judgement* dilakukan dengan membuat penilaian tentang kepentingan relative dua elemen pada suatu tingkat tertentu dalam kaitannya dengan tingkat di atasnya. Penilaian ini sangat penting karena akan berpengaruh terhadap prioritas dari elemen-elemen yang ada. Hasil penilaian ini dituliskan dalam matriks yang disebut dengan matriks *pairwise comparison*.

Pertanyaan yang biasa diajukan dalam penyusunan skala kepentingan adalah :

1. Pertanyaan mana yang lebih (penting / disukai / mungkin / dan sebagainya) ?
2. Berapa kali lebih (penting / disukai / mungkin / dan sebagainya) ?

Patokan (skala dasar) yang dapat digunakan dalam penyusunan skala kepentingan ini dapat dilihat pada Tabel 2.3.

Tabel 2.3 Skala kepentingan (Bernard W, 2005)

Tingkat Kepentingan	Definisi
1	Sama pentingnya sebanding yang lain
3	Moderat pentingnya dibanding yang lain
5	Kuat pentingnya dibanding yang lain
7	Sangat kuat pentingnya dibanding yang lain
9	Ekstrim pentingnya dibanding yang lain
2, 4, 6, 8	Nilai di antara dua penilaian yang berdekatan
Reciprocal	Jika elemen I memiliki salah satu angka di atas ketika dibandingkan dengan j, maka j memiliki nilai kebalikannya ketika dibandingkan dengan elemen i.

Catatan :

- a. Perbandingan dua elemen yang sama akan menghasilkan angka 1 artinya sama pentingnya.
- b. Dua elemen yang berlainan dapat saja dinilai sama penting.

Contoh :

Matriks *Pairwise Comparison* untuk tujuan (*Goal*)

Tabel 2.4 Matriks *Pairwise Comparison* (Bernard W, 2005)

Tujuan/Goal	Kriteria 1	Kriteria 2	Kriteria 3	Kriteria 4
Kriteria 1	1	5	2	4
Kriteria 2	1/5	1	1/2	1/2
Kriteria 3	1/2	2	1	2
Kriteria 4	1/4	2	1/2	1

3. *Synthesis Of Priority*

Dari setiap matriks *pairwise comparison* kemudian dicari *local priority*.

Matriks-matriks *pairwise comparison* terdapat pada setiap tingkat, sehingga untuk mendapatkan *global priority* harus dilakukan sintesis di antara *local priority*.

Contoh :

Matriks *pairwise comparison* untuk tujuan (*goal*)

Tabel 2.5 Penjumlahan matriks *pairwise comparison* (Bernard W, 2005)

Tujuan/Goal	Kriteria 1	Kriteria 2	Kriteria 3	Kriteria 4
Kriteria 1	1	5	2	4
Kriteria 2	1/5	1	1/2	1/2
Kriteria 3	1/2	2	1	2
Kriteria 4	1/4	2	1/2	1
Jumlah	1.95	10	4	7.5

Tabel 2.6 Normalisasi matriks (Bernard W, 2005)

Tujuan/Goal	Kriteria 1	Kriteria 2	Kriteria 3	Kriteria 4
Kriteria 1	0.51282	0.5	0.5	0.53333
Kriteria 2	0.10256	0.1	0.125	0.06667
Kriteria 3	0.25641	0.2	0.25	0.26667
Kriteria 4	0.12821	0.2	0.125	0.13333

Tabel 2.7 Perhitungan *local priority* (Bernard W, 2005)

Tujuan/Goal	Kriteria 1	Kriteria 2	Kriteria 3	Kriteria 4	Local Priority
Kriteria 1	0.51282	0.5	0.5	0.53333	0.5115
Kriteria 2	0.10256	0.1	0.125	0.06667	0.0986
Kriteria 3	0.25641	0.2	0.25	0.26667	0.2433
Kriteria 4	0.12821	0.2	0.125	0.13333	0.1466

4. Logical Consistency

Konsistensi memiliki s makna, yaitu :

- a. Obyek-obyek yang serupa dapat dikelompokkan sesuai dengan keseragaman dan relevansi.
- b. Menyangkut tingkat hubungan antara obyek-obyek yang didasarkan pada kriteria tertentu.

Bila diketahui A adalah matriks *pairwise comparison* di mana penilaian kita sempurna pada setiap perbandingan, maka berlaku $a_{ij} \cdot a_{jk} = a_{ik}$ untuk semua i, j, k dan selanjutnya matriks A dikatakan konsisten. AHP mengukur seluruh

konsistensi penilaian dengan menggunakan Consistency Ratio (CR), yang dirumuskan :

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

di mana Consistency Index (CI) adalah

$$CI = \frac{(z_{maks} - n)}{n - 1}$$

Z_{maks} adalah nilai eigen maksimum dari matriks *pairwise comparison* yang didapatkan dari hasil penjumlahan perkalian matriks dan vektor kemudian dibagi dengan jumlah item yang diperbandingkan. Berikut adalah rumus untuk menghitung nilai Z_{maks} . $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \dots, \alpha_n$.

$$\alpha_{max} = \frac{\sum \alpha}{n}$$

Nilai Random Consistency Index (RI) dapat digunakan patokan tabel berikut (Mulyono, 2003) :

Tabel 2.8 Patokan nilai *random consistency index* (Mulyono, 2003)

n	1	2	3	4	5	6	7	8	9	10
RI	0	0	0.58	0.9	1.12	1.24	1.32	1.41	1.45	1.49

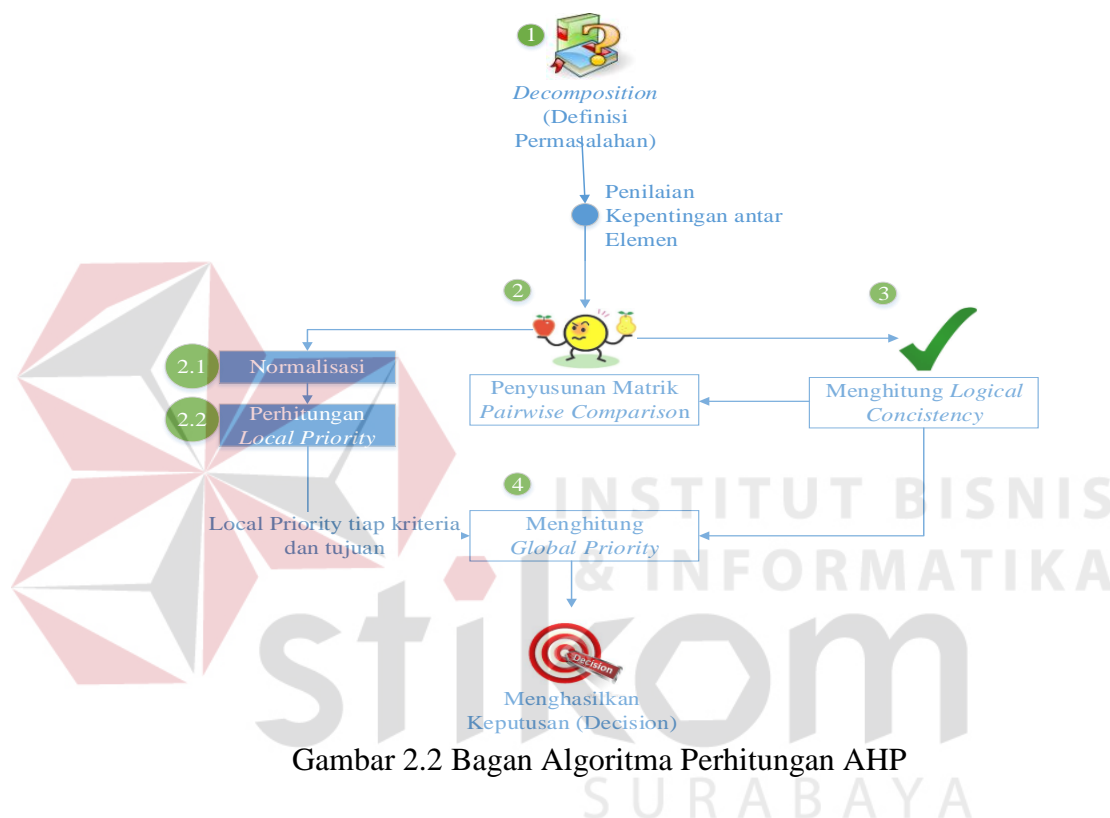
Perlu diingat bahwa nilai CR (*Consistency Ratio*) semestinya tidak boleh lebih dari 10%. Jika tidak, maka penilaian yang telah dibuat mungkin dilakukan secara random dan perlu direvisi.

5. *Global Priority*

Tahapan ini merupakan tahap akhir dari perhitungan AHP untuk memberikan kesimpulan mengenai keputusan yang akan diambil. Perhitungan global priority dilakukan dengan mengalikan nilai dari *local priority* dari masing-masing kriteria dengan *local priority* tujuan (*Goal*). Hasil perkalian akan

menghasilkan suatu keputusan pemilihan suatu alternative berdasarkan pada hasil perkalian tertinggi.

Penjabaran mengenai langkah-langkah perhitungan AHP di atas Akan dijelaskan juga dalam bagan alur perhitungan AHP pada gambar 2.3. Hal tersebut untuk mempermudah dalam memahami algoritma perhitungan AHP.



Gambar 2.2 Bagan Algoritma Perhitungan AHP

2.5 Databases

Menurut Raharjo (2011) *database* adalah kumpulan data yang terintegrasi dan diatur sedemikian rupa hingga data tersebut dapat dimanipulasi, diambil, dan dicari secara cepat. Isi dari *database* adalah *metadata*, yaitu data yang menjelaskan tentang struktur dari data itu sendiri. Data dalam *database* sendiri diklarifikasikan jenisnya dan disimpan di dalam wadah tersendiri, yang disebut tabel. Tabel adalah suatu entitas yang tersusun atas kolom (*field*) dan baris (*record*). Di dalam *database* juga mengenal *constraint*, yaitu aturan atau batasan yang didefinisikan nilai atau

data yang dapat disimpan di dalam *database*, antara lain melalui operasi *INSERT*, *UPDATE* dan *DELETE*. *Constraint* dibedakan menjadi empat, antara lain:

1. *Primary key*

Primary key adalah aturan untuk memastikan bahwa setiap baris data di dalam suatu tabel bersifat unik (berbeda antara baris yang satu dengan yang lainnya), *primary key* diterapkan pada kolom tabel untuk dijadikan sebagai pembeda antar baris tabel.

2. *Foreign key*

Foreign key berguna untuk mendefinisikan kolom-kolom pada suatu tabel yang nilainya mengacu ke tabel lain. Di mana nilai yang diisikan ke dalam kolom *foreign key* pada suatu tabel harus diacu/diambil dari kolom *primary key* pada tabel lain. Karena jika memasukkan nilai ke dalam kolom *foreign key* di mana nilai tersebut tidak terdapat pada tabel yang diacu, maka *server database* akan menolak cara menampilkan pesan kesalahan. Aturan dalam pendefinisian *foreign key*:

- Satu tabel dapat memiliki lebih dari satu *foreign key*
- Kolom yang diacu harus didefinisikan sebagai *primary key* atau *unique*
- *Foreign key* tidak bersifat unik.

3. *Unique*

Unique sama seperti *primary key*, yaitu untuk memastikan bahwa setiap baris data yang terdapat dalam suatu tabel bersifat unik (tidak sama).

4. *Check*

Check berfungsi untuk membatasi nilai-nilai yang dapat dimasukkan ke dalam suatu kolom di dalam tabel. Contohnya, kolom 'JENIS_KELAMIN' nilainya

dapat kita batasi, misalnya hanya berupa 'PRIA' dan 'WANITA' saja. Selain nilai tersebut, *server database* akan menolak.

2.6 MySQL

Menurut Nugroho (2014), MySQL adalah *software* atau program aplikasi *database*, yaitu *software* yang dipakai untuk menyimpan data berupa informasi teks dan juga angka. Dengan menggunakan *database* MySQL, maka data yang tersimpan di dalam *database* dapat diakses secara bersamaan menggunakan beberapa komputer yang berbeda, konsep ini sering disebut dengan sistem *multi user*. Sedangkan untuk konsep sistem *client/server*. *Database* MySQL diinstal pada sebuah komputer pusat (*server*) yang tersimpan di ruang *server*, kemudian datanya diakses melalui komputer yang diinstal dengan program *client* yang letaknya berbeda dengan tempat *server*. *Database* MySQL adalah *software* yang tidak memiliki *interface* (tampilan antarmuka) visual. Perintah dasar MySQL adalah SQL (*Structure Query Language*) di mana dapat diakses menggunakan DOS (*Disk Operation System*) *Prompt* atau *Command prompt* perintah dasar MySQL disebut dengan perintah *Data Definition Statement* (DDS), di dalamnya terdapat 3 perintah utama, yaitu:

- *Create* Perintah untuk membuat *database* baru, membuat tabel dan struktur datanya.
- *Alter* Perintah untuk memperbaiki nama *database*, nama tabel dan struktur datanya.
- *Drop* Perintah untuk menghapus nama *database* dan nama tabel.

Sedangkan untuk penampilan data, memperbaiki data dan menghapus data. Perintah SQL yang digunakan adalah perintah *Data Manipulation Statement* (DML), di dalamnya ada 4 perintah utama, yaitu:

- *Insert* Perintah untuk menyimpan data atau memasukkan data ke dalam tabel.
- *Select* Perintah untuk menampilkan data atau mengambil data yang ada di dalam tabel.
- *Update* Perintah untuk memperbaiki data atau mengubah data yang sudah ada di dalam tabel.
- *Delete* Perintah untuk menghapus data yang ada di dalam tabel.

Menurut Raharjo (2011) MySQL adalah *software* RDBMS (atau *server database*) yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak *user* (*multi-user*) dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi-threaded*). Ada beberapa kelebihan pada MySQL:

- Fleksibel: MySQL dapat digunakan untuk mengembangkan aplikasi *desktop* maupun aplikasi web, baik menggunakan PHP, JSP, Java, Delphi, C++ maupun yang lainnya.
- Performa Tinggi: MySQL mempunyai mesin *query* dengan performa tinggi, yang bisa membuat proses transaksional menjadi lebih cepat.
- Lintas Platform: MySQL bisa digunakan pada platform (dalam hal ini Sistem Operasi) yang beragam, bisa menggunakan Microsoft Windows, Linux, atau UNIX. Dari hal tersebut, proses migrasi data (bila dibutuhkan) antar sistem operasi bisa dilakukan lebih mudah.

- Gratis: MySQL dapat digunakan atau bisa di unduh secara gratis tanpa biaya, tapi ada juga *software* MySQL yang bersifat komersial, yang tentunya juga memiliki tambahan kemampuan spesifik dan mendapat pelayanan dari *technical support*.
- Proteksi Data yang Handal: MySQL menyediakan mekanisme yang canggih untuk menangani perlindungan terhadap keamanan data, dengan menyediakan fasilitas manajemen *user*, enkripsi data, dan lainnya.
- Komunitas Luas: MySQL memiliki komunitas yang luas dikarenakan penggunaanya banyak. Komunitas ini berguna sebagai pemberi solusi apabila mendapatkan permasalahan dalam penggunaan MySQL dengan cara melalui atau mengikuti forum.

2.7 PHP *Hypertext Preprocessor*

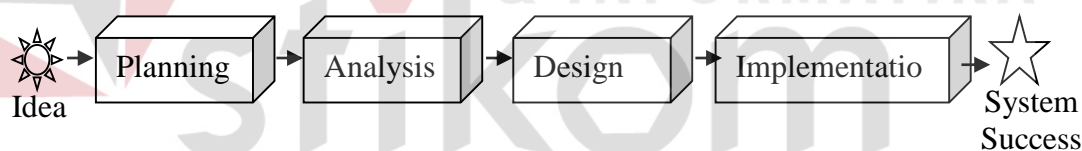
Menurut Nugroho (2013) *PHP Hypertext Preprocessor* (PHP) merupakan bahasa pemrograman berbasis web atau bahasa program yang digunakan untuk membuat aplikasi berbasis web (*Website, blog* atau aplikasi web). PHP merupakan bahasa program yang bisa dijalankan di sisi *server* atau sering disebut dengan “*side server language*”. Jadi program yang dibuat dengan kode PHP tidak bisa berjalan kecuali dia dijalankan pada server web, tanpa adanya server web yang terus berjalan dia tidak akan bisa dijalankan.

Karena berbasis web, maka aplikasi yang dibuat dengan PHP adalah berbasis web. Hasil dari PHP adalah berbentuk web, baik itu *website* (web pribadi, web portal berita, *web profile* perusahaan, dll). Hasilnya bisa dilihat dengan menjalankan *web browser*. PHP tidak ada aplikasi/*software* nyata yang bisa kita jalankan untuk menuliskan kodenya, butuh aplikasi editor untuk membuat kode

program PHP, contohnya seperti editor Notepad++ atau editor lainnya. PHP adalah bahasa programnya, dalam membuat *website* ataupun aplikasi berbasis web, bukan hanya kode PHP saja yang kita butuhkan tetapi juga menggunakan kode HTML (*Hyper Text Markup Language*) untuk desain tampilan, yaitu mengatur teks, tabel dan juga membuat *form*. Selain itu butuh CSS (*Cascading Style Sheets*) sebagai kode pemanis web, yang bisa jadi pengganti HTML. Jadi, dalam membuat website pasti menggunakan kode HTML, CSS dan PHP.

2.8 Metode Pengembangan Sistem

Menurut Dennis, dkk (2013), dalam membangun sistem dengan menggunakan SDLC memiliki empat fase dasar yaitu perencanaan, analisis, desain, dan implementasi. Setiap fase itu sendiri terdiri atas serangkaian langkah dengan mengandalkan teknik sehingga menghasilkan produk.



Gambar 2.3 Fase System Development Life Cycle

1. Perencanaan

Fase perencanaan ini adalah proses dasar dalam memahami mengapa sistem informasi harus dibuat dan menjelaskan bagaimana tim proyek akan melakukannya.

2. Analisis

Fase analisis ini menjelaskan pertanyaan tentang siapa yang akan menggunakan sistem, apa yang akan dilakukan sistem, dimana dan kapan sistem tersebut digunakan. Di dalam fase ini tim proyek melakukan investigasi

sistem saat ini, mengidentifikasi adanya perbaikan, dan mengembangkan konsep untuk sistem yang baru.

3. Desain

Fase desain ini menentukan bagaimana sistem akan beroperasi dengan perangkat keras, perangkat lunak, dan infrastruktur jaringan yang ada. Fase ini juga termasuk menentukan tampilan antarmuka, formulir, laporan yang akan digunakan, spesifikasi program, basis data, dan bahan-bahan yang dibutuhkan.

4. Implementasi

Fase akhir di dalam SDLC adalah fase implementasi, di mana sistem ini sudah benar-benar dibangun. Ini adalah fase yang biasanya paling diperhatikan, karena ini adalah bagian yang terpanjang dan termahal di dalam proses pengembangan.

2.9 Pengertian Bagan Alir Dokumen

Menurut Krismiaji (2010), Bagan alir merupakan teknik analitis yang digunakan untuk menjelaskan aspek-aspek sistem informasi secara jelas, tepat dan logis. Bagan alir menggunakan serangkaian simbol standar untuk menguraikan prosedur pengolahan transaksi yang digunakan oleh sebuah perusahaan, sekaligus menguraikan aliran data dalam sebuah sistem.

Menurut Jogiyanto (2005 : 20) Bagan alir dokumen (*document flowchart*) atau di sebut juga bagan alir formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan (*charts*) yang menunjukkan alir (*flow*) didalam program atau prosedur sistem secara logika dapat didefinisikan sebagai bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem dengan menggunakan simbol seperti pada tabel berikut:

Tabel 2.9 Simbol-simbol *flowchart* (Jogiyanto, 2005)

No.	Simbol	Nama Simbol <i>Flowchart</i>	Fungsi
1.		Dokumen	Untuk menunjukkan dokumen <i>input</i> dan <i>output</i> baik untuk proses manual, mekanik atau komputer.
2.		Proses Komputerisasi	Menunjukkan kegiatan dari operasi program komputer.
3.		Database	Untuk menyimpan data.
4.		Penghubung	Menunjukkan hubungan di halaman yang sama.
5.		Penghubung Halaman Lain	Menunjukkan hubungan di halaman lain.
6.		<i>Terminator</i>	Menandakan awal/akhir dari suatu sistem.
7.		<i>Decision</i>	Menggambarkan logika keputusan dengan nilai <i>true</i> atau <i>false</i> .
8.		Kegiatan Manual	Untuk menunjukkan pekerjaan yang dilakukan secara manual.
9.		Simpanan <i>Offline</i>	Untuk menunjukkan file non-komputer yang diarsip urut angka.

Dan terdapat beberapa jenis bagan alir yang biasa digunakan, yaitu sebagai berikut:

a. Bagan Alir Sistem (*System Flowchart*)

Bagan alir sistem (*system flowchart*) merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urutan dari

prosedur-prosedur yang ada di dalam sistem. Bagan alir sistem menunjukkan apa yang dikerjakan dalam sistem. Bagan alir sistem digambarkan dengan menggunakan simbol-simbol yang telah ditentukan.

b. Bagan Alir Dokumen (*Document Flowchart*)

Bagan alir dokumen (*document flowchart*) atau disebut dengan bagan alir formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya. Bagan alir dokumen ini menggunakan simbol-simbol yang sama dengan yang digunakan didalam bagan alir sistem.

c. Bagan Alir Skematik (*Schematic Flowchart*)

Bagan alir skematik (*schematic flowchart*) merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu menggambarkan prosedur dalam sistem. Perbedaannya adalah bagan alir skematik selain menggunakan simbol-simbol bagan alir sistem, juga menggunakan gambar-gambar komputer dan peralatan lainnya yang digunakan. Maksud penggunaan gambar-gambar ini adalah untuk memudahkan dalam menjelaskan simbol-simbol bagan alir kepada orang yang masih awam.

d. Bagan Alir Program (*Program Flowchart*)

Bagan alir program (*program flowchart*) terdiri dari dua macam, yaitu bagan alir logika program (*program logic flowchart*) dan bagan alir program komputer terinci (*detailed computer program flowchart*). Bagan alir logika program digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika. Bagan alir logika program ini dipersiapkan oleh analis sistem.

e. Bagan Alir Proses (*Process Flowchart*)


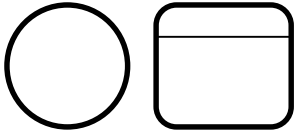

Bagan alir proses (*process flowchart*) merupakan bagan alir yang banyak digunakan di teknik industri. Berguna bagi analisis sistem untuk menggambarkan proses dalam suatu prosedur.

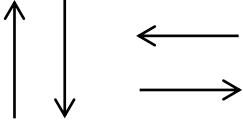
Berdasarkan penjelasan di atas, dapat ditarik kesimpulan bahwa bagan alir (*flowchart*) adalah suatu gambaran umum tentang sistem yang berjalan dan berfungsi sebagai alat bantu komunikasi serta untuk mendokumentasikan dan menyajikan kegiatan mulai dari manual, semi manual maupun komputerisasi.

2.10 Pengertian *Data Flow Diagram*

Menurut Hartono (2003), *Data Flow Diagram* (DFD) adalah diagram yang menggunakan notasi simbol untuk menggambarkan arus data sistem. Kita dapat menggunakan DFD untuk dua hal utama, yaitu untuk membuat dokumentasi dari sistem informasi yang ada, atau untuk menyusun dokumentasi untuk sistem informasi yang baru. Empat simbol yang digunakan yaitu :

Tabel 2.10 Simbol *Data Flow Diagram*

Simbol	Keterangan
	<i>External Entity</i> , merupakan kesatuan di lingkungan luar sistem yang bisa berupa orang, organisasi atau sistem lain.
	<i>Process</i> , merupakan proses seperti perhitungan aritmatika penulisan suatu formula atau pembuatan laporan
	<i>Data Store</i> (Simpan Data), dapat berupa suatu <i>file</i> atau <i>database</i> pada sistem komputer atau catatan manual

Simbol	Keterangan
	<i>Data Flow</i> (Arus Data), arus data ini mengalir di antara proses, simpan data dan kesatuan luar

Menurut Hartono (2009), ada beberapa simbol digunakan pada DFD untuk mewakili:

1. Kesatuan Luar (*External Entity*)

Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi, atau sistem lain yang berada pada lingkungan luarnya yang memberikan *input* atau menerima *output* dari sistem.

2. Proses (*Process*)

Proses (*process*) menunjukkan pada bagian yang mengubah *input* menjadi *output*, yaitu menunjukkan bagaimana satu atau lebih *input* diubah menjadi beberapa *output*. Setiap proses mempunyai nama, nama dari proses ini menunjukkan apa yang dikerjakan proses.

3. Simpanan Data (*Data Store*)

Data Store merupakan simpanan dari data yang dapat berupa suatu *file* atau *database* pada sistem komputer.

4. Arus Data (*Data Flow*)

Arus Data (*Data Flow*) di DFD diberi simbol suatu panah. Arus data ini mengalir di antara proses, simpan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

2.11 Pengertian Testing

Pengertian *testing* menurut Nidhra & Dondeti (2012), pengujian *software* merupakan suatu prosedur atau teknik yang digunakan untuk verifikasi dan validasi kualitas suatu *software*, di mana *software* akan dieksekusi untuk menemukan kesalahan yang ada di dalamnya. Sedangkan menurut Quadri & Farooq (2010), pengujian *software* adalah proses verifikasi dan validasi apakah sebuah aplikasi *software* atau program memenuhi persyaratan bisnis dan persyaratan teknis yang mengarahkan desain dan pengembangan serta cara kerjanya seperti yang diharapkan, juga untuk mengidentifikasi kesalahan yang penting yang digolongkan berdasarkan tingkat *severity* pada aplikasi yang harus diperbaiki. Dan menurut Simarmata (2010) *Testing* atau pengujian adalah proses terhadap program untuk menemukan segala kesalahan dan segala kemungkinan yang akan menimbulkan kesalahan sesuai dengan spesifikasi perangkat lunak yang telah ditentukan sebelum program tersebut diserahkan kepada pelanggan. Prinsip yang harus diperhatikan pada testing/pengujian adalah.

1. Semua pengujian harus dapat ditelusuri sesuai permintaan pengguna.
2. Pengujian harus direncanakan lama sebelum pengujian dimulai.
3. Pengujian harus dimulai dari “hal kecil” dan berkembang ke “hal besar”.
4. Pengujian yang mendalam tidak mungkin dilakukan.
5. Agar efektif, pengujian dilakukan pihak ketiga yang independen.

Terdapat 2 metode pengujian perangkat lunak yang dapat digunakan, yaitu: metode *white-box* dan *black-box*. Pengujian metode *white-box* yaitu menguji struktur internal perangkat lunak dengan melakukan pengujian pada algoritma yang digunakan oleh perangkat lunak sedangkan pengujian metode *black-box* berfokus

pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black-box* memungkinkan perancang perangkat lunak mendapatkan serangkaian kondisi *input* yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program.

Dari pendapat-pendapat tersebut dapat disimpulkan bahwa pengujian *software* merupakan proses verifikasi dan validasi sebuah *software* apakah sudah memenuhi kebutuhan dan untuk mengidentifikasi kesalahan-kesalahan yang ditemukan saat eksekusi program.

