

BAB II. KAJIAN PUSTAKA

A. Tanaman Pangan

Dalam Purwono dan Purnamawati (2007), pangan diartikan sebagai segala sesuatu yang bersumber dari sumber hayati dan air, baik yang diolah maupun tidak diolah. Pangan diperuntukan bagi konsumsi manusia sebagai makanan atau minuman, termasuk bahan tambahan pangan, bahan baku pangan, dan bahan-bahan lain yang digunakan dalam proses penyiapan, pengolahan, dan atau pembuatan makanan dan minuman.

Komoditas pangan harus mengandung zat gizi yang terdiri atas karbohidrat, protein, lemak, vitamin dan mineral yang bermanfaat bagi pertumbuhan dan kesehatan manusia. Kelompok tanaman budidaya yang tergolong komoditas ini meliputi kelompok tanaman pangan, tanaman hortikultura non tanaman hias, dan kelompok tanaman lain penghasil bahan baku produk yang memenuhi batasan pangan.

Batasan untuk tanaman pangan adalah kelompok tanaman sumber karbohidrat dan protein. Namun, secara sempit, tanaman pangan biasanya dibatasi pada kelompok tanaman yang berumur semusim. Batasan ini di masa mendatang harus diperbaiki karena akan menyebabkan sumber karbohidrat menjadi terbatas. Tanaman pangan sebaiknya memasukan jenis tanaman lain yang dapat menjadi sumber karbohidrat tanpa dibatasi pada kelompok tanaman semusim. Dengan perbaikan batasan ini, tanaman umbian selain ubi kayu, ubi jalar, dan talas dapat masuk kedalam kelompok tanaman pangan, misalnya garut, ganyong, dan kimpul. Demikian juga dengan buah yang merupakan sumber karbohidrat dapat masuk kedalam tanaman pangan, misalnya sukun.

Tanaman pangan menyebar hampir secara merata di seluruh wilayah Indonesia meskipun sentra beberapa jenis tanaman pangan terdapat di daerah tertentu. Hal ini disebabkan oleh kesesuaian lahan dan kultur masyarakat dalam mengembangkan jenis tanaman pangan tertentu.

B. Sistem Pendukung Keputusan

Wibisono (2003) menyatakan bahwa sistem pendukung keputusan adalah sistem berbasis komputer yang membantu para pengambil keputusan mengatasi berbagai masalah melalui interaksi langsung dengan sejumlah *database* dan perangkat lunak analitik. Tujuan dari sistem adalah untuk menyimpan data dan mengubahnya ke informasi yang terorganisir yang dapat diakses dengan mudah, sehingga keputusan-keputusan yang diambil dapat dilakukan dengan cepat, akurat, dan murah.

Sistem pendukung keputusan ini beroperasi dalam konteks sistem informasi global yang melayani unit bisnis yang spesifik dalam suatu perusahaan. Sistem pendukung keputusan tidak terlepas dari sistem informasi global yang lebih komprehensif. Sistem pendukung keputusan yang berhasil harus mempercepat aliran informasi ke pengambil keputusan. Data yang disimpan harus berkesinambungan secara terjadwal dan dapat diakses dengan mudah.

Sukoco (2007) menyatakan bahwa sistem pendukung keputusan adalah sistem informasi pada tingkatan manajemen yang mengkombinasikan data dengan sistem analisis data untuk mendukung pengambilan keputusan yang terstruktur maupun tidak.

Menurut Little (1970) dalam Turban, dkk. (2005) mendefinisikan DSS sebagai “sekumpulan prosedur berbasis model untuk data pemrosesan dan penilaian guna membantu para manajer mengambil keputusan”. Dia menyatakan bahwa untuk sukses, sistem tersebut haruslah sederhana, cepat, mudah dikontrol, adaptif, lengkap dengan isu-isu penting, dan mudah berkomunikasi.

Menurut Turban (2005) dalam Azis (2012) mendefinisikan proses pengambilan keputusan terdiri dari tiga fase yaitu sebagai berikut :

1. *Intelligence*

Tahap ini meliputi *scanning* (pemindaian) lingkungan entah *intermittent* ataupun terus menerus. *Intelligence* mencakup aktifitas yang menekankan identifikasi masalah.

2. *Design*

Tahap ini meliputi penemuan atau mengembangkan dan menganalisis tindakan yang mungkin untuk dilakukan. Hal ini meliputi pemahaman terhadap masalah dan menguji dengan solusi yang layak.

3. *Choice*

Pada tahap ini merupakan tindakan pengambilan keputusan yang nyata dan diambil suatu komitmen untuk mengikuti suatu tindakan tertentu.

C. *Simple Additive Weighting (SAW)*

Kusumadewi, dkk. (2006) menyatakan bahwa metode *Simple Additive Weighting (SAW)* sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut. Metode SAW membutuhkan proses

normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternative yang ada. Formula untuk melakukan normalisasi tersebut tampak seperti pada Gambar 1 berikut ini.

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max}_i(x_{ij})} & \text{Jika } j \text{ adalah kriteria keuntungan (benefit)} \\ \frac{\text{Min}_i(x_{ij})}{x_{ij}} & \text{Jika } j \text{ adalah kriteria biaya (cost)} \end{cases}$$

Gambar 1. Formula Normalisasi

Dimana r_{ij} adalah rating kinerja ternormalisasi dari alternatif A_i pada atribut C_j ; $i=1,2,\dots,m$ dan $j=1,2,\dots,n$. Gambar 2 merupakan formula nilai preferensi untuk setiap alternatif (V_i) diberikan.

$$V_i = \sum_{j=1}^n w_j r_{ij}$$

Gambar 2. Formula Nilai Preferensi

Nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih.

Contoh Kasus:

Suatu perusahaan di Daerah Istimewa Yogyakarta (DIY) ingin membangun sebuah gedung yang akan digunakan sebagai tempat untuk menyimpan sementara hasil produksinya. Ada 3 lokasi yang akan menjadi alternatif, yaitu: A_1 = Ngemplak, A_2 = Kalasan, A_3 = Kota Gedhe. Ada 5 kriteria yang dijadikan acuan dalam pengambilan keputusan, yaitu:

- C_1 = jarak dengan pasar terdekat (km),
- C_2 = kepadatan penduduk di sekitar lokasi (orang/km²),

- C_3 = jarak dari pabrik (km),
- C_4 = jarak dengan gudang yang sudah ada (km),
- C_5 = harga tanah untuk lokasi ($\times 1000$ Rp/m²).

Rating kecocokan setiap alternatif pada setiap kriteria, dinilai dengan 1 sampai 5, yaitu:

- 1 = Sangat buruk,
- 2 = Buruk,
- 3 = Cukup,
- 4 = Baik,
- 5 = Sangat Baik.

Tabel 1 menunjukkan rating kecocokan dari setiap alternatif pada setiap kriteria.

Sedangkan tingkat kepentingan setiap kriteria, juga dinilai 1 sampai dengan 5, yaitu:

- 1 = Sangat rendah,
- 2 = Rendah,
- 3 = Cukup,
- 4 = Tinggi,
- 5 = Sangat Tinggi.

Tabel 1. rating kecocokan dari setiap alternatif pada setiap kriteria

Alternatif	Kriteria				
	C_1	C_2	C_3	C_4	C_5
A ₁	4	4	5	3	3
A ₂	3	3	4	2	3
A ₃	5	4	2	2	2

Karena setiap nilai yang diberikan ada setiap alternatif di setiap kriteria merupakan nilai kecocokan (nilai terbesar adalah terbaik), maka semua kriteria yang diberikan diasumsikan sebagai kriteria keuntungan.

Pengambil keputusan memberikan bobot preferensi sebagai:

$$W = (5, 3, 4, 4, 2)$$

Matriks keputusan dibentuk dari tabel kecocokan sebagai berikut:

$$X = \begin{pmatrix} 4 & 4 & 5 & 3 & 3 \\ 3 & 3 & 4 & 2 & 3 \\ 5 & 4 & 2 & 2 & 2 \end{pmatrix}$$

Pertama-tama, dilakukan normalisasi matriks X berdasarkan persamaan pada Gambar 1 sebagai berikut:

$$r_{11} = \frac{4}{\max\{4; 3; 5\}} = \frac{4}{5} = 0,8$$

$$r_{21} = \frac{3}{\max\{4; 3; 5\}} = \frac{3}{5} = 0,6$$

$$r_{31} = \frac{5}{\max\{4; 3; 5\}} = \frac{5}{5} = 1$$

$$r_{22} = \frac{4}{\max\{4; 3; 4\}} = \frac{4}{4} = 1$$

$$r_{22} = \frac{3}{\max\{4; 3; 4\}} = \frac{3}{4} = 0,75$$

$$r_{32} = \frac{4}{\max\{4; 3; 4\}} = \frac{4}{4} = 1$$

dan seterusnya, sehingga diperoleh matriks ternormalisasi R sebagai berikut:

$$R = \begin{pmatrix} 0,8000 & 1,0000 & 1,0000 & 1,0000 & 1,0000 \\ 0,6000 & 0,7500 & 0,8000 & 0,6667 & 1,0000 \\ 1,0000 & 1,0000 & 0,4000 & 0,6667 & 0,6667 \end{pmatrix}$$

Proses perankingan diperoleh berdasarkan persamaan pada Gambar 2 sebagai berikut:

$$V_1 = (5)(0,8) + (3)(1) + (4)(1) + (4)(1) + (2)(1) = 17$$

$$V_2 = (5)(0,6) + (3)(0,75) + (4)(0,8) + (4)(0,6667) + (2)(1) = 13,1176$$

$$V_3 = (5)(1) + (3)(1) + (4)(0,4) + (4)(0,6667) + (2)(0,6667) = 13,6$$

Nilai terbesar ada pada V_1 sehingga alternatif A_1 adalah alternatif yang terpilih sebagai alternatif terbaik. Dengan kata lain, Ngemplak akan terpilih sebagai lokasi untuk mendirikan gudang baru.

Menurut Kusumadewi (2006) dalam Nugraha (2011) mendefinisikan *simple additive weighting* (SAW) merupakan metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua kriteria. Metode SAW membutuhkan proses normalisasi matrik keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada.

Metode SAW mengenal adanya 2 (dua) atribut yaitu kriteria keuntungan (*benefit*) dan kriteria biaya (*cost*). Perbedaan mendasar dari kedua kriteria ini adalah dalam pemilihan kriteria ketika mengambil keputusan.

Adapun langkah penyelesaian dalam menggunakannya adalah:

1. Menentukan alternatif, yaitu A_i .
2. Menentukan kriteria yang akan dijadikan acuan dalam pengambilan keputusan, yaitu C_j .
3. Memberikan nilai rating kecocokan setiap alternatif pada setiap kriteria.

- Menentukan bobot preferensi atau tingkat kepentingan (W) setiap kriteria seperti Gambar 3.

$$W = [W_1 \quad W_2 \quad W_3 \quad \dots \quad W_j]$$

Gambar 3. Bobot Preferensi

- Membuat tabel rating kecocokan dari setiap alternatif pada setiap kriteria.
- Membuat matriks keputusan yang dibentuk dari tabel rating kecocokan dari setiap alternatif pada setiap kriteria. Nilai setiap alternatif (A_i) pada setiap kriteria (C_j) yang sudah ditentukan, dimana, $i=1,2,\dots,m$ dan $j=1,2,\dots,n$ seperti Gambar 4.

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1j} \\ \vdots & \vdots & \dots & \vdots \\ x_{i1} & x_{i2} & \dots & x_{ij} \end{bmatrix}$$

Gambar 4. Matriks Keputusan

- Melakukan normalisasi matriks keputusan dengan cara menghitung nilai rating kinerja ternormalisasi (r_{ij}) dari alternatif A_i pada kriteria C_j seperti Gambar 5.

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max}_i(x_{ij})} & \text{Jika } j \text{ adalah kriteria keuntungan (benefit)} \\ \frac{\text{Min}_i(x_{ij})}{x_{ij}} & \text{Jika } j \text{ adalah kriteria biaya (cost)} \end{cases}$$

Gambar 5. Normalisasi Matriks

Keterangan :

- a. Dikatakan kriteria keuntungan apabila nilai X_{ij} memberikan keuntungan bagi pengambil keputusan, sebaliknya kriteria biaya apabila X_{ij} menimbulkan biaya bagi pengambil keputusan.
 - b. Apabila berupa kriteria keuntungan maka nilai X_{ij} dibagi dengan nilai $Max_i X_{ij}$ dari setiap kolom, sedangkan untuk kriteria biaya, nilai $Min_i X_{ij}$ dari setiap kolom dibagi dengan nilai X_{ij} .
8. Hasil dari nilai rating kinerja ternormalisasi (r_{ij}) membentuk matriks ternormalisasi (R) seperti Gambar 6.

$$R = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1j} \\ \vdots & \vdots & \vdots & \vdots \\ r_{i1} & r_{i2} & \cdots & r_{ij} \end{bmatrix}$$

Gambar 6. Matriks Ternormalisasi

9. Hasil akhir nilai preferensi (V_i) diperoleh dari penjumlahan dari perkalian elemen baris matrik ternormalisasi (R) dengan bobot preferensi (W) yang bersesuaian elemen kolom matrik (W) seperti Gambar 7.

$$V_i = \sum_{j=1}^n w_j r_{ij}$$

Gambar 7. Nilai Preferensi

Hasil perhitungan nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i merupakan alternatif terbaik.

D. PHP

Menurut Anhar (2010) PHP singkatan dari PHP: *Hypertext Preprocessor* yaitu bahasa pemrograman *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 baru atau *up to date*. Semua *script* PHP dieksekusi pada server dimana *script* tersebut dijalankan.

PHP adalah bahasa *open source* yang dapat digunakan di berbagai sistem operasi seperti: Linux, Unix, Macintosh dan Windows. PHP dapat dijalankan secara *runtime* melalui *console* serta dapat menjalankan perintah-perintah *system*. *Open source* artinya kode-kode PHP terbuka untuk umum dan kita tidak harus membayar biaya pembelian atas keaslian *license* yang cukup mahal. Karena *source code* PHP tersedia secara gratis, maka hal tersebut memungkinkan komunitas milis-milis dan *developer* untuk melakukan perbaikan, pengembangan dan menemukan *bug* dalam bahasa PHP.

Menurut Sakur (2010) *Hypertext PreProcessor* atau dikenal dengan nama PHP merupakan salah satu aplikasi *server* yang sangat banyak digunakan sampai sekarang karena kemudahan dan keandalannya. PHP pada mulanya dikembangkan hanya untuk mengantisipasi penggunaan *database*, namun kemudian dikembangkan menjadi aplikasi yang telah memiliki banyak fitur dalam mengembangkan aplikasi *non-database*.

PHP merupakan bahasa pemrograman untuk *server* yang dapat digabungkan ke dalam halaman HTML dengan beberapa cara penulisan, baik secara langsung maupun menggunakan fungsi referensi.

E. *Database*

Wibisono (2003) menyatakan bahwa *database* adalah sekumpulan data mentah yang disusun menurut logika tertentu dan terorganisasi dalam bentuk yang dapat disimpan dan diproses oleh komputer. Contoh *database* dapat berisi data pegawai, data penjualan, pembayaran, dan lain-lain. Data internal dari akunting, keuangan, penjualan dan bidang-bidang bisnis lainnya yang disimpan dalam suatu sistem komputer dan disusun menurut logika tertentu disebut sebagai *internal database*.

Database sering kali disimpan dalam suatu perangkat tertentu pada komputer, seperti *harddisk*, *compact disk*, dan sebagainya. Hubungan antar sistem *database* dan sistem *software* sangat kuat karena sistem *database* yang dipakai sangat menentukan kemudahan aksesnya data sementara *software* sendiri memungkinkan peneliti memanipulasi data untuk dianalisis.

Menurut Anhar (2010) *database* adalah sekumpulan tabel-tabel yang berisi data dan merupakan kumpulan dari *field* atau kolom. Struktur *file* yang menyusun sebuah *database* adalah data *record* dan *field*.

1. Data adalah satu satuan informasi yang akan diolah. Sebelum diolah, data dikumpulkan di dalam suatu file *database*.
2. *Record* adalah data yang isinya merupakan satu kesatuan seperti NamaUser dan Password. Setiap keterangan yang mencakup NamaUser dan *Password*

dinamakan satu *record*. Setiap *record* diberi nomor urut yang disebut nomor *record* (*record number*).

3. *Field* adalah sub bagian dari *record*. Dari contoh isi *record* diatas, maka terdiri dari 2 *field*, yaitu *field* NamaUser dan *Password*.

Kristanto (1993) menyatakan bahwa *database* adalah kumpulan *file-file* yang saling berelasi, relasi tersebut biasa ditunjukkan dengan kunci dari setiap *file* yang ada. Satu *database* menunjukkan satu kumpulan data yang dipakai dalam satu lingkup perusahaan, instansi. Dalam satu *file* terdapat *record-record* yang sejenis, sama besar, sama bentuk, merupakan satu kumpulan *entity* yang seragam. satu *record* terdiri dari *field-field* yang saling berhubungan untuk menunjukkan bahwa *field* tersebut dalam satu pengertian yang lengkap dan direkam dalam satu *record*.

Untuk menyebut isi dari *field* maka digunakan *atribute* atau merupakan judul dari satu kelompok *entity* tertentu, misalnya *atribute* Alamat menunjukkan *entity* alamat dari siswa. *Entity* adalah suatu objek yang nyata dan akan direkam.

F. MySQL

Anonim (2010) menyatakan bahwa MySQL adalah salah satu *software* sistem manajemen *database* (DBMS) *structured query language* (SQL) yang bersifat *open source*. SQL adalah bahasa standar untuk mengakses *database* dan didefinisikan dengan standar ANSI/ISO SQL. MySQL dikembangkan, disebarluaskan, dan didukung oleh MySQL AB. MySQL AB adalah perusahaan komersial yang didirikan oleh pengembang MySQL. MySQL merupakan aplikasi *relational database management system* (RDBMS) yang dapat digunakan sebagai aplikasi *client-server* atau sistem *embedded*.

MySQL mempunyai beberapa sifat yang menjadikannya sebagai salah satu *software database* yang banyak digunakan oleh pemakai diseluruh dunia. Sifat-sifat yang dimiliki oleh MySQL antara lain:

1. MySQL merupakan DBMS (*Database Management System*).
2. *Database* adalah kumpulan data yang terstruktur. Data dapat berupa daftar belanja, kumpulan gambar, atau yang lebih luas yaitu informasi jaringan perusahaan. Agar dapat menambah, mengakses, dan memproses data yang tersimpan pada sebuah komputer database, dibutuhkan sistem manajemen *database* (DBMS) seperti MySQL *Server*. Sejak komputer sangat baik dalam menangani sejumlah besar data, sistem manajemen *database* (DBMS) memainkan peran utama dalam perhitungan baik sebagai peralatan yang berdiri sendiri maupun bagian sebuah aplikasi.
3. MySQL merupakan RDBMS (*Relational Database Management System*).
4. *Relational Database* menyimpan data pada tabel-tabel yang terpisah, bukan menyimpan data dalam ruang penyimpanan yang besar, hal ini menambah kecepatan dan fleksibilitas.
5. MySQL merupakan *software open source*.
6. *Open source* berarti setiap orang dapat menggunakan dan mengubah *software* yang bersangkutan. Setiap orang dapat mengunduh *software* MySQL dari internet dan menggunakannya tanpa membayar. *Software* MySQL menggunakan GNU/GPL (*General Public License*).
7. MySQL mempunyai performa yang sangat cepat, dapat dipercaya, dan mudah digunakan.

8. MySQL Server sebenarnya dikembangkan untuk menangani *database* yang lebih cepat daripada solusi yang ada dan telah berhasil digunakan pada lingkungan produksi dengan permintaan tinggi untuk beberapa tahun terakhir.
9. MySQL Server bekerja pada *client-server* atau pada sistem *embedded*.

G. HTML

Menurut Oktavian (2010), HTML (*Hypertext markup language*) adalah suatu bahasa yang dikenali oleh *web browser* untuk menampilkan informasi dengan lebih menarik dibandingkan dengan tulisan teks biasa (*plain text*). Sedangkan *web browser* adalah program komputer yang digunakan untuk membaca HTML, kemudian menerjemahkan dan menampilkan hasilnya secara visual ke layar komputer. Karena sebuah bahasa, maka HTML mempunyai aturan dan struktur tertentu untuk menuliskan perintah-perintahnya yang biasa dinamakan TAG HTML. Aturan tersebut diawali dengan lambang `<tag>` dan biasanya diakhiri dengan lambang `</tag>`. Berikut gambaran struktur HTML.

```
1 <html>
2   <head>
3     <title>Judul web ditulis disini</title>
4   </head>
5   <body>
6     ... kode-kode HTML lainnya ditulis di bagian ini...
7   </body>
8 </html>
```

Gambar 8. Contoh Source Code HTML

Dari struktur diatas, terlihat bahwa penulisan kode-kode HTML lainnya untuk keperluan isi situs web akan diletakkan di bagian tag `<body>`. Dengan demikian dapat dikatakan bahwa informasi yang berupa kode-kode akan diapit oleh tag awal

dan tag akhir, dan sebuah apitan tag bisa juga diapit oleh tag lainnya. Dalam HTML ada beberapa tag yang tidak perlu diakhiri atau ditutup, misal tag
.

H. Penelitian Sejenis

1. Sistem Pendukung Keputusan Menggunakan Metode *Simpel Additive Weighting* (SAW) Untuk Menentukan Jurusan Pada SMK Bakti Purwokerto (Hermanto, 2012)

Dari hasil perancangan dan pembuatan aplikasi sistem pendukung keputusan menggunakan metode *Simple Additive Weighting* (SAW), dapat disimpulkan bahwa:

- a. Sistem pendukung keputusan yang telah dibuat dapat mempermudah dan mempercepat proses penjurusan oleh panitia penerimaan siswa baru karena menggunakan proses perhitungan yang cepat dan tepat
- b. Sistem pendukung keputusan yang telah dibuat dapat diakses dari mana saja selama tersedia jaringan internet karena dibuat berbasis *web*

2. Sistem Pendukung Keputusan Pemilihan Karyawan Terbaik Dengan Metode SAW (*Simple Additive Weighting*) (Ariyanto, 2012)

Berdasarkan kegiatan yang telah dilakukan oleh penulis selama perancangan sampai implementasi sistem pendukung keputusan pemilihan karyawan terbaik ini, maka dapat diambil beberapa kesimpulan berikut:

- a. Telah berhasil dibuat sistem informasi pendukung keputusan pemilihan karyawan terbaik dengan metode *Simple Additive Weighting* (SAW) di Pamella Swalayan

b. Sistem memberikan solusi rekomendasi karyawan terbaik kepada pengguna (*user*) sesuai dengan kriteria dan bobot yang ditentukan diawal sebelum perhitungan

3. Perancangan Sistem Pendukung Keputusan Pengambilan Keputusan Untuk Penerimaan Beasiswa Dengan Metode SAW (*Simple Additive Weighting*) (Eniyati, 2011)

Dari hasil Perancangan Sistem Pendukung Keputusan Pengambilan Keputusan Untuk Penerimaan Beasiswa Dengan Metode SAW (*Simple Additive Weighting*), dapat disimpulkan bahwa perancangan yang telah disusun sebagian besar merupakan kriteria untuk penerimaan beasiswa dalam sekolah. Bobot perhitungan adalah merupakan salah satu indikator penting dalam perhitungan untuk penerimaan beasiswa.

