



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II

LANDASAN TEORI

2.1 Pengisian Kartu Rencana Studi

Kartu Rencana Studi (KRS) adalah lembar yang berisikan mata kuliah beban belajar mahasiswa serta jadwal, ruang kuliah, dan dosen, yang pengisiannya dilakukan setelah mahasiswa mendapat bimbingan dosen pembimbing akademik (STIM Lasharan Jaya, 2015). Pengisian Kartu Rencana Studi (KRS) adalah kegiatan yang dilakukan mahasiswa yang mata kuliahnya belum terpaket, sehingga mengharuskan mahasiswa untuk memilih mata kuliah apa saja yang akan diambil sesuai dengan jumlah SKS maksimal. Beberapa mata kuliah yang akan diambil dalam pengisian KRS memiliki persyaratan untuk dapat diambil, yaitu ada mata kuliah yang harus diambil sebelumnya.

2.2 Aplikasi Berbasis Web

2.2.1 Situs Web

Website pertama kali ditemukan oleh Sir Timothy John Berners Lee pada tahun 1991 (Rahmadian, 2011). Tujuan dari Sir Timothy ketika membuat *website* adalah untuk mempermudah tukar menukar data.

Situs *web* ditulis dan diakses melalui *web browser* atau yang disebut juga HTTP *client*. Halaman *web* diakses melalui jaringan internet, dengan perangkat seperti komputer atau *mobile device*. Situs web memiliki protokol jaringan yang bernama *Hypertext Transfer Protocol* (HTTP).

Ada dua jenis *website*, yaitu *web* statis dan *web* dinamis. *Web* statis adalah *web* yang tidak dimaksudkan untuk *update* secara berkala, sedangkan *web* dinamis adalah *website* yang secara berkala *update* dan interaksi sudah

berlangsung dua arah. Jika *user* memberikan *request* untuk berpindah halaman atau meminta data, *web server* akan merespon dengan mengambil data yang diperlukan dan menampilkannya di sisi *client*.

2.2.2 Aplikasi Berbasis Web

Aplikasi berbasis *web* merupakan aplikasi memanfaatkan mekanisme dan aplikasi yang sudah ada pada sistem *web* (Rahmadian, 2011). Aplikasi berbasis *web* bisa langsung dijalankan dengan koneksi internet, dan tidak terbatas pada sistem operasi dari perangkat yang digunakan. Sistem *web* merupakan aplikasi yang memiliki arsitektur *client-server*, menggunakan protokol HTTP, dan memiliki fungsi untuk menjalankan isi dokumen di *server* dan menampilkannya ke *client*.

Beberapa keunggulan aplikasi berbasis *web* adalah sebagai berikut.

1. Tidak bergantung pada platform, bisa dijalankan dari sistem operasi windows, linux, BSD, dan Mac. Program yang diperlukan hanyalah browser.
2. Tidak perlu *install* aplikasi dan bisa dijalankan di banyak komputer.
3. Bisa dijalankan dari jarak jauh dengan menggunakan internet.
4. Selalu mendapat versi terbaru, karena data berada di *server*, jika diperbarui maka *client* akan mendapat *update* otomatis jika ada versi baru.

Beberapa kekurangan aplikasi berbasis *web* adalah sebagai berikut.

1. Antarmuka yang dapat dibuat terbatas sesuai spesifikasi standar untuk membuat dokumen *web* dan keterbatasan *web browser* untuk menampilkan.
2. Jika kecepatan internet lambat akan membuat respon aplikasi menjadi lambat

2.3 Sistem Pendukung Keputusan

2.3.1 Keputusan

Istilah keputusan terurai menjadi dua yaitu keputusan terprogram dan keputusan tidak terprogram (Simon, 1977). Keputusan terprogram yaitu bersifat berulang-ulang dan rutin. Pada suatu tingkat tertentu dan prosedur telah ditetapkan untuk menanganinya sehingga ia dianggap suatu *denovo* (yang baru) setiap kali terjadi.

Keputusan tidak terprogram yaitu bersifat baru, tidak terstruktur, dan biasanya tidak urut. Simon juga menjelaskan bahwa dua jenis keputusan tersebut hanyalah kesatuan ujung yang terangkai secara hitam putih, sifatnya begitu kelabu atau tidak jelas, namun demikian konsep keputusan terprogram dan tidak terprogram sangatlah penting, karena masing-masing memiliki teknik berbeda.

2.3.2 Decision Support System

Istilah *Decision Support System* atau Sistem Pendukung Keputusan (SPK) diciptakan oleh dua professor MIT yaitu Anthony Gorry dan Michael S.Morton pada tahun 1971 (McLeod). Definisi Sistem Pendukung Keputusan (SPK) adalah sistem berbasis komputer interaktif, yang membantu para pengambil keputusan untuk menggunakan data dan berbagai model untuk memecahkan masalah-masalah tidak terstruktur, Morton (Turban, 2000).

Menurut Sparague dan Watson (1993) definisi Sistem Pendukung Keputusan (SPK) sebagai sistem yang memiliki lima karakteristik utama sebagai berikut.

1. Sistem yang berbasis komputer.
2. Dipergunakan untuk membantu para pengambil keputusan.

3. Untuk memecahkan masalah-masalah rumit yang mustahil dilakukan dengan kalkulasi manual.
4. Melalui cara simulasi yang interaktif.
5. Dimana data dan model analisis sebagai komponen utama.

Tujuan dari Sistem Pendukung Keputusan menurut Turban (2005) adalah sebagai berikut.

1. Membantu manajer dalam pengambilan keputusan atas masalah semi terstruktur.
2. Memberikan dukungan atas pertimbangan manajer
3. Meningkatkan efektivitas keputusan yang di ambil manajer lebih daripada perbaikan efisiensinya
4. Kecepatan komputasi.
5. Peningkatan produktivitas.
6. Dukungan kualitas
7. Berdaya Saing
8. Mengatasi keterbatasan kognitif dalam pemrosesan dan penyimpanan.

2.4 Technique for Order Preference by Similarity to Ideal Solution

Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) adalah metode yang ditemukan oleh Yoon dan Hwang pada tahun 1981 (Yoon & Hwang, 1981). Metode ini menggunakan prinsip bahwa alternatif yang dipilih harus memiliki jarak terdekat dari solusi ideal positif dan juga jarak terjauh dari solusi ideal negatif. Solusi ideal positif adalah nilai terbaik yang dapat dicapai dari setiap atribut, sedangkan solusi ideal negatif adalah nilai terburuk yang dapat dicapai dari setiap atribut.

Langkah - langkah metode TOPSIS:

a. Membuat matriks keputusan yang ternormalisasi (Bustami, 2012).

Proses ini berfungsi untuk mendapatkan nilai yang sebanding (*comparable*) dari satu data dengan data lainnya (Kurniawan, M, & Aknuranda, 2011).

$$r_{ij} = \frac{X_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}} \quad \dots \text{Rumus 2.1}$$

Dengan $i = 1, 2, \dots, m$; dan $j = 1, 2, \dots, n$

Dimana :

r_{ij} adalah matriks ternormalisasi $[i][j]$

x_{ij} adalah matriks keputusan $[i][j]$

m adalah jumlah baris matriks

n adalah jumlah kolom matriks

b. Membuat matriks keputusan yang ternormalisasi terbobot (Bustami, 2012).

$$y_{ij} = w_i \cdot r_{ij} \quad \dots \text{Rumus 2.2}$$

Dengan $i = 1, 2, \dots, m$; dan $j = 1, 2, \dots, n$

Dimana :

y_{ij} adalah matriks ternormalisasi terbobot $[i][j]$

w_i adalah vektor bobot $[i]$

c. Menentukan matriks solusi ideal positif dan solusi ideal negatif (Bustami, 2012).

$$A^+ = (y_1^+, y_2^+, \dots, y_n^+) \quad \dots \text{Rumus 2.3}$$

$$A^- = (y_1^-, y_2^-, \dots, y_n^-) \quad \dots \text{Rumus 2.4}$$

Dimana :

$Y_j^+ = \max y_{ij}$ jika j adalah atribut keuntungan

$Y_j^+ = \min y_{ij}$ jika j adalah atribut biaya

$Y_j^- = \max y_{ij}$ jika j adalah atribut biaya

$Y_j^- = \min y_{ij}$ jika j adalah atribut keuntungan

d. Menentukan jarak setiap alternatif dengan matriks solusi ideal positif dan solusi ideal negatif (Bustami, 2012).

Jarak alternatif A_i dengan solusi ideal positif dapat dihitung dengan Rumus 2.5.

$$D_i^+ = \sqrt{\sum_{i=1}^n (y_i^+ - y_{ij})^2} \quad \dots \text{Rumus 2.5}$$

Dengan $i = 1, 2, \dots, m$; dan $j = 1, 2, \dots, n$

Dimana :

D_i^+ = jarak alternatif A_i dengan solusi ideal positif

y_i^+ = solusi ideal positif [i]

Jarak alternatif A_i dengan solusi ideal negatif dapat dihitung dengan Rumus 2.6.

$$D_i^- = \sqrt{\sum_{j=1}^n (y_{ij} - y_i^-)^2} \quad \dots \text{Rumus 2.6}$$

Dengan $i = 1, 2, \dots, m$; dan $j = 1, 2, \dots, n$

Dimana :

D_i^- = jarak alternatif A_i dengan solusi ideal negatif

y_i^- = solusi ideal negatif [i]

e. Menentukan nilai preferensi untuk setiap alternatif (Bustami, 2012).

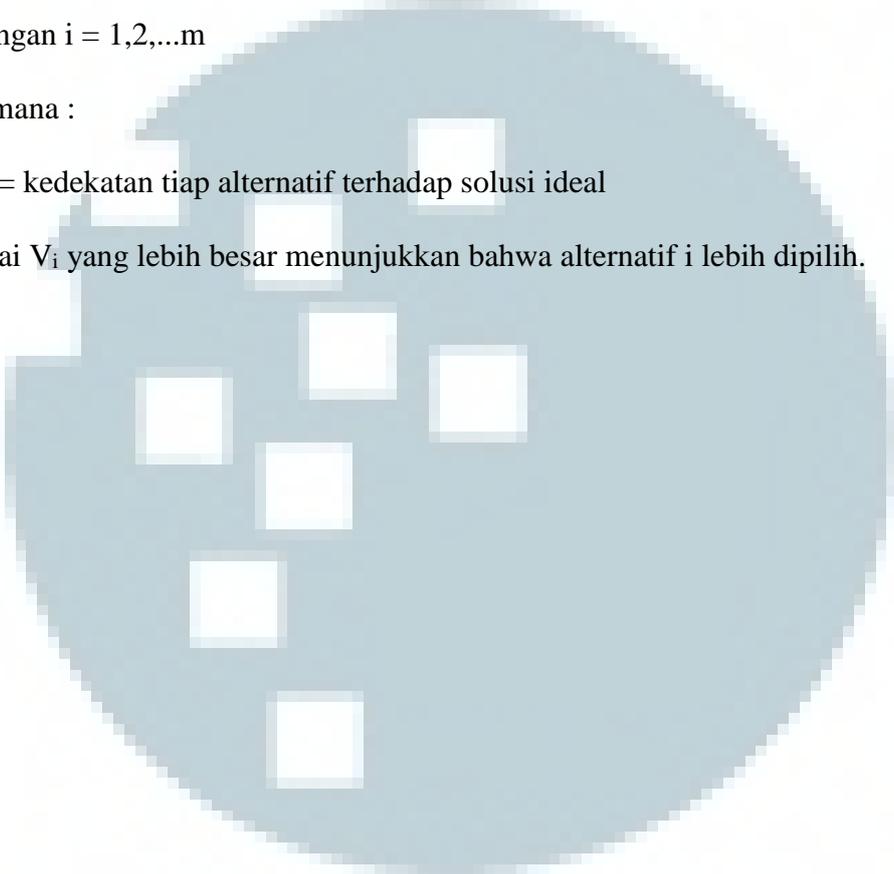
$$V_i = \frac{D_i^-}{D_i^- + D_i^+} \quad \dots \text{Rumus 2.7}$$

Dengan $i = 1, 2, \dots, m$

Dimana :

V_i = kedekatan tiap alternatif terhadap solusi ideal

Nilai V_i yang lebih besar menunjukkan bahwa alternatif i lebih dipilih.



UMMN