

BAB II

LANDASAN TEORI

2.1 Tinjauan Terhadap Penelitian Terkait

Berikut beberapa penelitian terkait yang menjadi referensi pada penelitian ini:

1. Irfan Ardhiyanto, Veronica Lusiana, dan Novita Mariana dalam skripsinya yang berjudul: “Implementasi Metode Weighted Product Pada Sistem Pendukung Keputusan penilaian Karyawan Terbaik pada Pandanaran Hotel Semarang”.

Sistem pendukung keputusan untuk pemilihan karyawan terbaik adalah sebuah alternatif untuk membantu pengambilan keputusan dalam proses pemilihan karyawan terbaik tiap periodenya. Sistem yang dirancang untuk menghasilkan informasi yang dapat membantu manajer personalia untuk menilai kinerja karyawan dengan menggunakan data dan model untuk memecahkan masalah dengan menggunakan metode Weighted Product (WP).

Metode WP lebih spesifik langsung kepada bobot-bobot nilai disetiap kriterianya dan mudah untuk dilakukan perangkian. Metode WP hampir sama dengan metode Simple Additive Weighting (SAW) tetapi berbeda dalam masalah pemberian nilai bobot tiap kriteria dan proses perhitungan akhir (V). Berdasarkan hasil penelitian terdahulu menunjukkan pengujian yang dilakukan menghasilkan SAW sebesar 12,64% dan WP sebesar 35.75%, dimana nilai tersebut merupakan standar deviasi dari perangkian preferensi alternatif V, sehingga diperoleh kesimpulan bahwa metode WP

lebih baik dari perbandingan antara metode WP dengan metode SAW pada penelitian tersebut.

2. Dahri Yani Hakim Tanjung , Robiatul Adawiyah dalam skripsinya yang berjudul: “Perancangan Sistem Pendukung Keputusan Dengan Metode Simple Multi Attribute Rating Technique (SMART) Dalam Penilaian Kinerja Dosen”.

Adapun metode problem solving yang digunakan dalam penelitian ini adalah Sistem Pendukung Keputusan. Sistem pendukung keputusan yang ditawarkan menggunakan metode Simple Multi Attribute Rating Technique (SMART) dalam menyelesaikan persoalan. Metode ini meliputi proses penilaian kinerja yang dimulai dari pembobotan kriteria untuk mengetahui bobot kepentingan masing-masing indikator kemudian penjabaran tujuan strategis ke dalam indikator kinerja. Dari pembobotan indikator tersebut dapat menghasilkan bobot alternatif untuk mengetahui nilai tertinggi dari alternatif yang ada. Dari hasil perangkingan tersebut diharapkan dapat memberikan alternatif optimal yaitu dosen terbaik yang akan dipertimbangkan oleh pengambil keputusan.

2.2 Sistem Pendukung Keputusan

DSS merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data. Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi yang semi terstruktur dan situasi tidak terstruktur, di mana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat.

Menurut Dadan (2001), konsep Sistem Pendukung Keputusan (SPK) pertama kali diungkapkan pada awal tahun 1970-an oleh Michael S. Scott Morton yang menjelaskan bahwa “Sistem Pendukung Keputusan adalah suatu sistem yang berbasis komputer yang ditujukan untuk membantu pengambil keputusan dalam memanfaatkan data dan model tertentu untuk memecahkan berbagai persoalan yang tidak terstruktur”. Selain itu Efraim Turban mengemukakan bahwa “Sistem Pendukung Keputusan merupakan sebuah sistem yang dimaksudkan untuk mendukung para pengambil keputusan manajerial dalam situasi keputusan semi terstruktur”.

Menurut Mcleod, Jr & Schell (2008) dalam bukunya yang berjudul *Information Management* menyatakan bahwa “*Decision Support System* adalah suatu sistem yang membantu seorang manager atau sekelompok manager dalam memecahkan suatu masalah”. Sedangkan menurut Krishnamuti (2008) dalam bukunya yang berjudul *Developing a Work in Process Tracking System Via a Decision Support System*, *Decision Support System* adalah sebuah sistem informasi yang bersifat dan bersifat sebagai suatu sistem penyokong dalam mengambil keputusan untuk memecahkan suatu masalah dan mengambil keputusan yang didasarkan dari informasi.

2.2.1 Tahapan Sistem Pendukung Keputusan

1. Definisi masalah
2. Pengumpulan data atau elemen informasi yang relevan
3. Pengolahan data menjadi informasi baik dalam bentuk laporan grafik maupun tulisan
4. Menentukan alternatif-alternatif solusi (bisa dalam persentase)

2.2.2 Tujuan dari Sitem Pendukung Keputusan

1. Membantu menyelesaikan masalah semi-terstruktur
2. Mendukung manajer dalam mengambil keputusan suatu masalah
3. Meningkatkan efektifitas bukan efisiensi pengambilan keputusan

Dalam pengolahan, DSS bisa menggunakan bantuan sistem lain seperti Artificial Intelligence, Expert Systems, Fuzzy Logic, dll

2.2.3. Manfaat Sistem Pendukung Keputusan

SPK dapat memberikan berbagai manfaat dan keuntungan. Manfaat yang dapat diambil dari SPK adalah :

1. SPK memperluas kemampuan pengambil keputusan dalam memproses data / informasi bagi pemakainya.
2. SPK membantu pengambil keputusan untuk memecahkan masalah, terutama dalam berbagai isu yang sangat kompleks dan tidak terstruktur.
3. SPK dapat menghasilkan solusi yang lebih cepat dan hasil yang lebih dapat diandalkan.
4. Walaupun suatu SPK mungkin tidak dapat memecahkan masalah yang dihadapi oleh pengambil keputusan, tapi dia bisa menjadi stimulan bagi para pengambil keputusan dalam memahami masalah, karena mampu menghadirkan berbagai solusi alternatif.

2.2.4 Komponen Sistem Pendukung Keputusan

Secara umum, Sistem Pendukung Keputusan yang dikembangkan oleh tiga komponen utama, yaitu manajemen database, Basis Model dan Sistem Software / User Interface. Komponen SPK dapat digambarkan sebagai berikut.

1. Database Management

Adalah subsistem dari data yang terorganisir dalam database. Data adalah suatu sistem pendukung keputusan dapat berasal dari luar dan dalam lingkungan.

2. Model Base

Adalah model yang mewakili masalah dalam format kuantitatif (model matematika sebagai contoh) sebagai dasar simulasi atau pengambilan keputusan, termasuk tujuan permasalahan (tujuan), komponen terkait, keterbatasan yang ada (kendala), dan hal-hal terkait lainnya .

3. User Interface / Pengelolaan Dialog

Kadang-kadang disebut sebagai subsistem dialog, penggabungan antara dua komponen sebelumnya, yaitu Manajemen Database dan Model Basis tergabung dalam tiga komponen (*user interface*), setelah sebelumnya menjabat dalam bentuk model komputer untuk memahami.

2.2.5 Tahap – tahap Pengambilan Keputusan

Menurut Herbert A. Simon (Kadarsah, 2002:15-16), tahap – tahap yang harus dilalui dalam proses pengambilan keputusan sebagai berikut :

1. Tahap Pemahaman

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

2. Tahap Perancangan

Tahap ini merupakan proses pengembangan dan pencarian alternatif tindakan / solusi yang dapat diambil. Tersebut merupakan representasi kejadian nyata yang disederhanakan, sehingga diperlukan proses validasi dan verifikasi untuk mengetahui keakuratan model dalam meneliti masalah yang ada.

3. Tahap Pemilihan

Tahap ini dilakukan pemilihan terhadap diantara berbagai alternatif solusi yang dimunculkan pada tahap perencanaan agar ditentukan / dengan memperhatikan kriteria kriteria berdasarkan tujuan yang akan dicapai

4. Tahap Impelementasi

Tahap ini dilakukan penerapan terhadap rancangan sistem yang telah dibuat pada tahap perancangan serta pelaksanaan alternatif tindakan yang telah dipilih pada tahap pemilihan.

2.3 Metode Weighted Product (WP)

Metode Weighted Product (WP) adalah salah satu metode penyelesaian pada sistem pendukung keputusan. Metode ini mengevaluasi beberapa alternatif terhadap sekumpulan atribut atau kriteria, dimana setiap atribut saling tidak bergantung satu dengan yang lainnya.pengembangan dari algoritma ID3 yang di ciptakan oleh J. Rose Quinlan. Menurut Yoon (dalam

buku Kusumadewi, 2006), metode weighted product menggunakan teknik perkalian untuk menghubungkan rating atribut, dimana rating tiap atribut harus dipangkatkan terlebih dahulu dengan bobot atribut yang bersangkutan.

Langkah-langkah penyelesaian WP sebagai berikut:

1. Menentukan kriteria - kriteria
2. Menentukan rating kecocokan
3. Melakukan normalisasi bobot
4. Menentukan nilai vector S
5. Menentukan nilai vector V
6. Merangking nilai vector V

Berikut Adalah Langkah Langkah Perhitungan Weighted Product sebagai berikut:

1. Untuk melakukan normalisasi bobot, bobot setiap kriteria penjumlahan semua bobot kriteria. Nilai dari total bobot harus memenuhi persamaan:

$$W_j = \frac{w_j}{\sum w_j}$$

2. Dengan cara mengalikan seluruh kriteria bagi sebuah alternatif dengan bobot sebagai pangkat positif untuk kriteria benefit dan bobot berfungsi sebagai pangkat negatif pada kriteria cost.

Rumus untuk menghitung nilai preferensi untuk alternatif A_i , diberikan sebagai berikut:

$$S_i = \pi_{j=1}^n X_{ij}^{w_j}$$

S : menyatakan preferensi alternatif yang dianalogikan sebagai vektor S

x : menyatakan nilai kriteria

w : menyatakan bobot kriteria

i : menyatakan alternatif

j : menyatakan kriteria

n : menyatakan banyaknya kriteria

3. Menentukan N Vektor

Yaitu nilai yang akan digunakan untuk perangkingan.

Nilai preferensi relatif dari setiap alternatif dapat dihitung dengan rumus:

$$V_i = \frac{\sum_{j=1}^n X_{ij} w_j}{\sum_{j=1}^n X_{ij} * w_j} \text{ atau } V_i = \frac{S_i}{\varepsilon S_i}$$

2.4 Metode Simple Multi Attribute Rating Technique (SMART)

1. Menentukan Kriteria

Menentukan kriteria yang digunakan dalam menyelesaikan masalah pengambilan keputusan. Untuk menentukan kriteria-kriteria apa saja yang digunakan dalam sistem pengambilan keputusan ini diperlukan data-data dari pengambil keputusan atau pihak yang berwenang/kompeten terhadap masalah yang akan diselesaikan.

2. Menentukan Bobot Kriteria

Memberikan bobot kriteria pada masing-masing kriteria dengan menggunakan interval 1-100 untuk masing-masing kriteria dengan prioritas terpenting

3. Normalisasi Bobot Kriteria

Menghitung normalisasi bobot dari setiap kriteria dengan membandingkan nilai bobot kriteria dengan jumlah bobot kriteria, menggunakan persamaan :

$$w_i = \frac{w'_i}{\sum_{j=1}^m w_j}$$

Keterangan

- w_i : bobot kriteria ternormalisasi untuk kriteria ke-i
- w'_i : bobot kriteria ke-i
- w_j : bobot kriteria ke-j
- j : 1,2,3, ... , m jumlah kriteria

4. Memberikan Nilai Parameter untuk Tiap Kriteria

Memberikan nilai kriteria untuk setiap alternatif, nilai kriteria untuk setiap alternatif ini dapat berbentuk data kuantitatif (angka) ataupun berbentuk data kualitatif, misalkan nilai untuk kriteria harga sudah dapat dipastikan berbentuk kuantitatif sedangkan nilai untuk kriteria fasilitas bisa jadi berbentuk kualitatif (sangat lengkap, lengkap, kurang lengkap).

Apabila nilai kriteria berbentuk kualitatif maka kita perlu mengubah ke data kuantitatif dengan membuat parameter nilai kriteria, misalkan sangat lengkap artinya 3, lengkap artinya 2 dan tidak lengkap artinya 1.

5. Menentukan Nilai Utility

Menentukan nilai utility dengan mengkonversikan nilai kriteria pada masing-masing kriteria menjadi nilai kriteria data baku. Nilai utility ini tergantung pada sifat kriteria itu sendiri.

Kriteria Biaya (Cost Criteria)

Kriteria yang bersifat "lebih diinginkan nilai yang lebih kecil" kriteria seperti ini biasanya dalam bentuk biaya yang harus dikeluarkan (misalkan kriteria harga, kriteria penggunaan bahan bakar per kilometer untuk pembelian mobil, periode pengembalian modal dalam suatu usaha,

kriteria waktu pengiriman) dapat dihitung dengan menggunakan persamaan :

$$u_i(a_i) = (c_{\max} - c_{out})(c_{\max} - c_{\min})$$

Keterangan

- $u_j(a_i)$: nilai utility kriteria ke-j untuk alternatif ke-i
- $c_{\max}$: nilai kriteria maksimal
- $c_{\min}$: nilai kriteria minimal
- c_{out} : nilai kriteria alternatif ke-i

6. Menentukan Nilai Akhir

Menentukan nilai akhir dari masing-masing dengan mengalikan nilai yang didapat dari normalisasi nilai kriteria data baku dengan nilai normalisasi bobot kriteria. Kemudian jumlahkan nilai dari perkalian tersebut

$$u(a_i) = \sum_{j=1}^m w_j * u_j(a_i)$$

Keterangan

- $u(a_i)$: nilai total untuk alternatif ke-i
- w_j : nilai bobot kriteria ke-j yang sudah ternormalisasi
- $u_j(a_i)$: nilai utility kriteria ke-j untuk alternatif ke-i

7. Perangkingan

Hasil dari perhitungan Nilai akhir kemudian diurutkan dari nilai yang terbesar hingga yang terkecil, alternatif dengan nilai akhir yang terbesar menunjukkan alternatif yang terbaik

2.5 Unified Modeling Language

UML (Unified Modeling Language) merupakan metodologi kolaborasi antara metoda – metoda *booch*, *OMT (Object Modeling Technique)* serta *OOSE (Object Oriented Engineering)* dan beberapa metoda lainnya

merupakan metodologi yang paling sering digunakan saat ini untuk mengadaptasi maraknya penggunaan bahasa “pemograman berorientasi objek” (OOP). (Adi Nogroho; 2017: 4).

2.5.1 Usecase Diagram

Diagram yang menggambarkan *actor*, *use case* dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor. Sebuah *use case* digambarkan sebagai *elips horizontal* dalam suatu diagram UML

2.5.2 Activity Diagram

Activity diagram atau Diagram Aktivitas adalah teknik untuk mendeskripsikan logika prosedural, proses bisnis dan aliran kerja dalam banyak kasus. Diagram aktivitas mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah diagram aktivitas bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa. Sebuah *activity diagram* memiliki :

2.5.3 Sequence Diagram

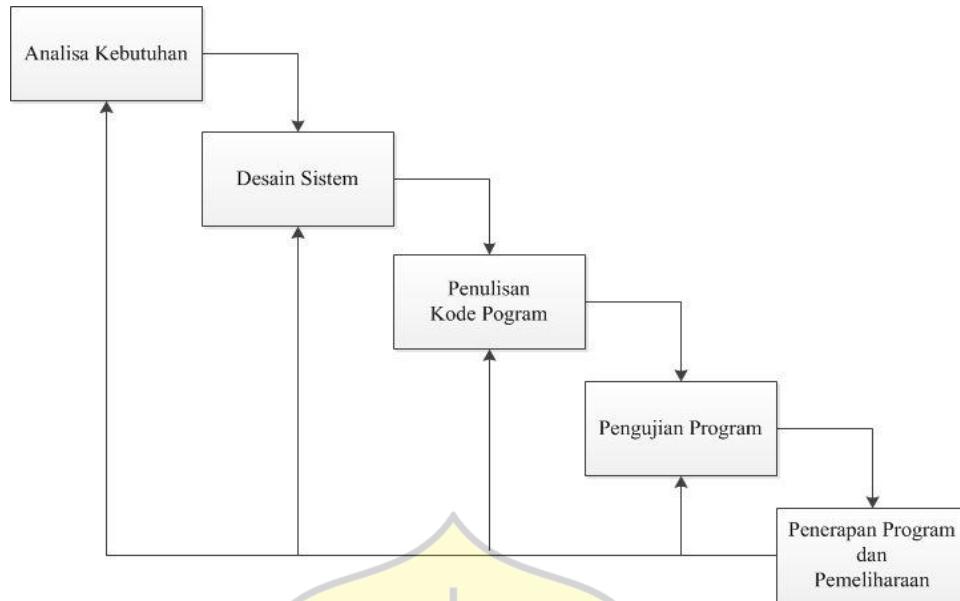
Merupakan diagram yang menunjukkan aliran fungsionalitas dalam *usecase*. *Sequence* diagram adalah satu dari dua interaksi diagram yang mengilustrasikan objek-objek yang berhubungan dengan *usecase* dan *message* atau pesan-pesannya. Komponen utama *sequence* diagram terdiri atas objek yang dituliskan dengan kotak segiempat

bernama. *Message* diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan *progress vertical*.

2.6 Metodologi Pengembangan Sistem Waterfall

Metode penelitian yang diterapkan pada penelitian ini adalah dengan pengembangan metode waterfall. Metode Waterfall adalah suatu proses pengembangan perangkat lunak berurutan, di mana kemajuan dipandang sebagai terus mengalir ke bawah (seperti air terjun) melewati fase-fase perencanaan, pemodelan, implementasi (konstruksi), dan pengujian. Menurut Wahyu Wijaya Widiyanto (2018:36), Model waterfall adalah model klasik yang bersifat sistematis, berurutan dalam membangun software.

Nama model ini sebenarnya adalah “Linear Sequential Model”. Model ini sering disebut juga dengan “*classic life cycle*” atau metode waterfall. Model ini termasuk ke dalam model generic pada rekayasa perangkat lunak dan pertama kali diperkenalkan oleh Winston Royce sekitar tahun 1970 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai dalam *Software Engineering (SE)*. Model ini melakukan pendekatan secara sistematis dan berurutan. Disebut dengan waterfall karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan



Gambar 2.1 Metodologi Waterfall (Wahyu Wijaya Widiyanto , 2018).

2.7 Pemrograman Aplikasi

2.7.1 Internet

Internet adalah jaringan global yang menghubungkan komputer-komputer diseluruh dunia. Dengan internet, sebuah komputer bisa mengakses data yang terdapat pada komputer lain di benua yang berbeda Internet (portmanteau dari interconnected network) adalah sistem jaringan komputer yang saling terhubung secara global dengan menggunakan paket protokol internet (TCP/IP) untuk menghubungkan perangkat di seluruh dunia. Ini adalah jaringan dari jaringan yang terdiri dari jaringan privat, publik, akademik, bisnis, dan pemerintah lokal ke lingkup global, dihubungkan oleh beragam teknologi elektronik, nirkabel, dan jaringan optik. Internet membawa beragam sumber daya dan layanan informasi, seperti dokumen hiperteks yang saling terkait dan

aplikasi World Wide Web (WWW), surat elektronik, telepon, dan berbagi berkas.

2.7.2 Web

Web merupakan suatu ruang informasi di mana sumber-sumber daya yang berguna diidentifikasi oleh pengenalan global yang disebut *Uniform User Identifier* (URL). URL dapat diibaratkan suatu alamat, dimana alamat tersebut terdiri atas:

1. Protokol yang digunakan oleh suatu *browser* untuk mengambil informasi.
2. Nama computer (*server*) dimana informasi tersebut berada.
3. Jalur atau *path* serta nama *file* dari suatu informasi.

Format umum dari URL adalah sebagai berikut :

Protokol_transfer://nama_host/path/nama_file.

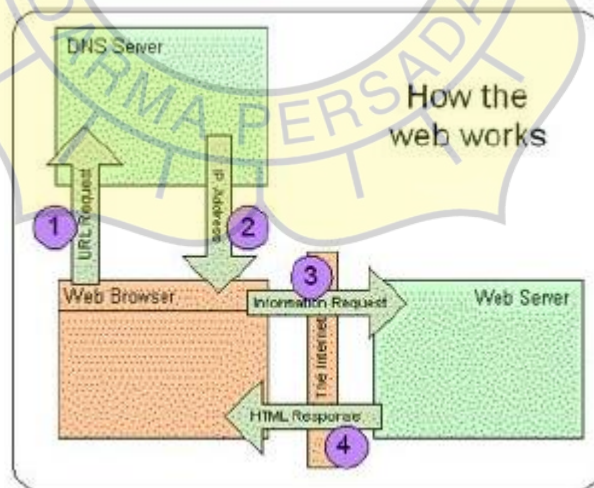
Contoh: <http://www.mine.com/e-journal/index.html>

Dari contoh tersebut dapat disimpulkan bahwa:

4. http adalah protocol yang digunakan.
5. www.mine.com adalah nama *host* atau *server* komputer dimana informasi yang dicari berada.
6. *e-journal* adalah jalur atau *path* dari informasi yang dicari.
7. Index.html adalah nama *file* dimana informasi tersebut berada.

Sebuah halaman *web* diakses dengan menggunakan *web browser* dengan menuliskan URL nya atau mengikuti *link* yang menuju kepadanya. *Uniform Resource Locator* (URL) akan menunjukkan lokasi dokumen yang dikelola oleh sebuah *web server*. URL diubah menjadi

alamat IP server web yang bersangkutan. Browser kemudian mengirimkan request *Hypertext Transfer Protocol* (HTTP) ke web server dan web server akan menjawab dokumen yang diminta dalam format *Hypertext Markup Language* (HTML). HTTP adalah suatu protocol yang menentukan aturan yang perlu diikuti oleh web browser dalam meminta atau mengambil suatu dokumen dan oleh web server dalam menyediakan dokumen yang diminta web browser. Protokol ini merupakan protocol standar yang digunakan untuk mengakses dokumen HTML. Dalam komunikasi jaringan computer diatur dengan protocol yang memungkinkan beragam jaringan computer untuk berkomunikasi. Protokol ini secara resmi dikenal dengan *Transmission Control Protocol* (TCP/IP) yang merupakan cara untuk mempacketkan sinyal elektronik sehingga data tersebut dapat dikirim ke computer lain. Gambar di bawah adalah menggambarkan cara kerja web yang diakses.



Gambar 2.2 Cara Kerja Web Yang di Akses

Web dapat dikategorikan menjadi dua yaitu web statis dan web dinamis atau interaktif. Web statis adalah web yang menampilkan

informasi-informasi yang sifatnya statis atau tetap, sedangkan *web* dinamis adalah *web* yang menampilkan informasi serta dapat berinteraksi dengan user yang sifatnya dinamis. Untuk membuat *web* dinamis dibutuhkan pemrograman *web* yang mempunyai dua kategori, yaitu :

1. Server-side Programming

Perintah-perintah program atau script dijalankan di *web server*, kemudian hasilnya dikirimkan ke *browser* dalam bentuk HTML.

2. Client-side Programming

Perintah program dijalankan di *web browser* sehingga ketika client meminta dokumen yang mengandung script, maka script tersebut akan didownload dari *server*nya kemudian dijalankan di *browser* yang bersangkutan.

2.7.3 HTML

Hypertext Markup Language (HTML) adalah bahasa yang digunakan untuk menulis halaman *web*. HTML merupakan pengembangan dari standar pemformatan dokumen teks yaitu Standard Generalized Markup Language (SGML) (Kadir, 2015). HTML sebenarnya adalah dokumen *ASCII* atau teks biasa, yang dirancang untuk tidak tergantung pada suatu sistem operasi tertentu.

Mendesain HTML berarti melakukan suatu tindakan pemrograman. Namun HTML bukanlah sebuah bahasa pemrograman. Namun HTML hanyalah berisi perintah-perintah yang telah terstruktur berupa tag-tag penyusun. Menuliskan tag-tag HTML tidaklah sebatas hanya memasukkan perintah-perintah tertentu agar HTML kita dapat

di akses oleh browser. Mendesain HTML adalah sebuah seni tersendiri. Homepage yang merupakan implementasi dari HTML adalah refleksi dari orang yang membuatnya. Untuk itu kita perlu mendesainnya dengan baik agar para pengunjung homepage yang kita buat merasa senang dan bermanfaat. Mendesain HTML dapat dilakukan dengan cara menggunakan *HTML Editor*, seperti notepad++, adobe dreamweaver dan lain-lain.

2.7.4 PHP

Menurut Kadir (2015) “*PHP* adalah pemrograman *interpreter* yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan”. *PHP* disebut juga pemrograman *Server Side Programming*, hal ini dikarenakan seluruh prosesnya dijalankan pada *server*. *PHP* adalah suatu bahasa dengan hak cipta terbuka atau yang juga dikenal dengan *open source* yaitu pengguna data mengembangkan kode-kode fungsi sesuai kebutuhannya.

Menurut Shalahuddin (2015:22) “*PHP (Perl Hypertext Preprocessor)* adalah bahasa *server-side-scripting* yang menyatu dengan *HTML* untuk membuat halaman web yang dinamis”. Dengan menggunakan program *PHP*, sebuah *website* akan lebih interaktif dan dinamis. Kelebihan-kelebihan dari *PHP* yaitu:

1. *PHP* merupakan sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya. Tidak seperti halnya bahasa pemrograman aplikasi yang lainnya.

2. *PHP* dapat berjalan pada *web server* yang dirilis oleh Microsoft, seperti *IIS* atau *PWS* juga pada *apache* yang bersifat *open source*.
3. Karena sifatnya yang *open source*, maka perubahan dan perkembangan interpreter pada *PHP* lebih cepat dan mudah, karena banyak milis-milis dan *developer* yang siap membantu pengembangannya.
4. Jika dilihat dari segi pemahaman, *PHP* memiliki referensi yang begitu banyak sehingga sangat mudah untuk dipahami.

PHP dapat berjalan pada 3 operating sistem, yaitu: *Linux*, *unix*, dan *windows*, dan juga dapat dijalankan secara *runtime* pada suatu *console*.

2.7.5 Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa markup (Kadir, 2015). Penggunaan yang paling umum dari CSS adalah untuk memformat halaman web yang ditulis dengan *HTML* dan *XHTML*. Walaupun demikian, bahasanya sendiri dapat dipergunakan untuk semua jenis dokumen *XML* termasuk *SVG* dan *XUL*. Spesifikasi CSS diatur oleh *World Wide Web Consortium (W3C)*. CSS digunakan oleh penulis maupun pembaca halaman web untuk menentukan warna, jenis huruf, tata letak, dan berbagai aspek tampilan dokumen. CSS digunakan terutama untuk memisahkan antara isi dokumen (yang ditulis dengan *HTML* atau *bahasa markup* lainnya) dengan presentasi dokumen (yang ditulis dengan *CSS*). Pemisahan ini dapat meningkatkan aksesibilitas isi, memberikan lebih banyak keleluasaan dan kontrol

terhadap tampilan, dan mengurangi kompleksitas serta pengulangan pada struktur isi.

2.7.6 JavaScript

JavaScript merupakan bahasa *scripting* yang pada awalnya dikembangkan oleh Netscape. Dengan menggunakan javascript, dapat dibuat tampilan web yang lebih interaktif lagi. Agar user dapat menjalankan javascript, alat yang dibutuhkan hanyalah *browser* yang kemampuan javascript-nya telah diaktifkan. Penulisan javascript dapat dipadukan pada halaman HTML, dan dari sini dapat dilihat bahwa javascript merupakan bahasa pemrograman yang berjalan di sisi *client*. Karena proses dari javascript tergantung proses dari *browser* yang digunakan oleh *user*.

2.7.7 Database

Menurut Mukhamad Masrur (2016), Database adalah sekumpulan file data yang satu sama lainnya saling berhubungan yang diorganisasi sedemikian rupa sehingga memudahkan untuk mendapatkan dan memproses data tersebut. Lingkungan sistem database menekankan pada data yang tidak tergantung (*independent*) pada aplikasi yang akan menggunakan data tersebut.

Penggunaan database pada komputer dilakukan dengan menggunakan table-table. Pada tabel-tabel tersebut masih dikelompokkan lagi menjadi beberapa bagian untuk membedakan data yang satu dengan data yang lain. Pada sebuah tabel database harus memiliki setau kategori data yang digunakan sebagai kunci untuk

membedakan data-data yang ada didalam satu tabel. Data kunci tersebut tidak boleh sama antara satu data dengan data lainnya. Data kunci sering disebut dengan *Primary Key*.

2.7.8 MySQL

MySql adalah suatu sistem manajemen database (*Relasional Database Management System*) bersifat terbuka (*Open Source*). Maksudnya terbuka adalah mysql boleh di-download oleh siapa saja baik versi kode program aslinya maupun versi binernya dan bisa digunakan secara gratis. Sejarah Mysql yang merupakan hasil buah pikiran dari Michael "Monty" Widenius, David Axmark dan Allan Larson dimulai tahun 1995. mereka bertiga mendirikan perusahaan bernama Mysql di Swedia. (Lukman,2015:148). Dengan berbagai keunggulan yang dimiliki, membuat software database ini banyak digunakan oleh praktisi untuk membangun suatu project. Adanya fasilitas API (*Application Programming Interface*) yang dimiliki oleh Mysql, memungkinkan bermacam-macam aplikasi Komputer yang ditulis dengan berbagai bahasa pemrograman dapat mengakses basis data MySQL.