

BAB II

LANDASAN TEORI

2.1 Sistem Pendukung Keputusan

2.1.1 Pengertian Sistem Pendukung Keputusan

Sistem Pendukung Keputusan merupakan salah satu produk perangkat lunak yang dikembangkan secara khusus untuk membantu dalam proses pengambilan keputusan. Sesuai dengan namanya, tujuan dari dipergunakannya sistem ini adalah sebagai “*second opinion*” atau “*information sources*” yang dapat dipakai sebagai bahan pertimbangan sebelum memutuskan kebijakan tertentu.

2.1.2 Proses Pengambilan Keputusan

Pengambilan keputusan adalah pemilihan beberapa tindakan alternatif yang ada untuk mencapai satu atau beberapa tujuan yang telah ditetapkan (Turban, 2005).

Pengambilan keputusan meliputi beberapa tahap dan melalui beberapa proses. Menurut Supranto (2005), pengambilan keputusan meliputi empat tahap yang saling berhubungan dan berurutan. Empat proses tersebut adalah :

a. *Intelligence*

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

b. *Design*

Tahap ini merupakan proses menemukan dan mengembangkan alternatif. Tahap ini meliputi proses untuk mengerti masalah, menurunkan solusi dan menguji kelayakan solusi.

c. *Choice*

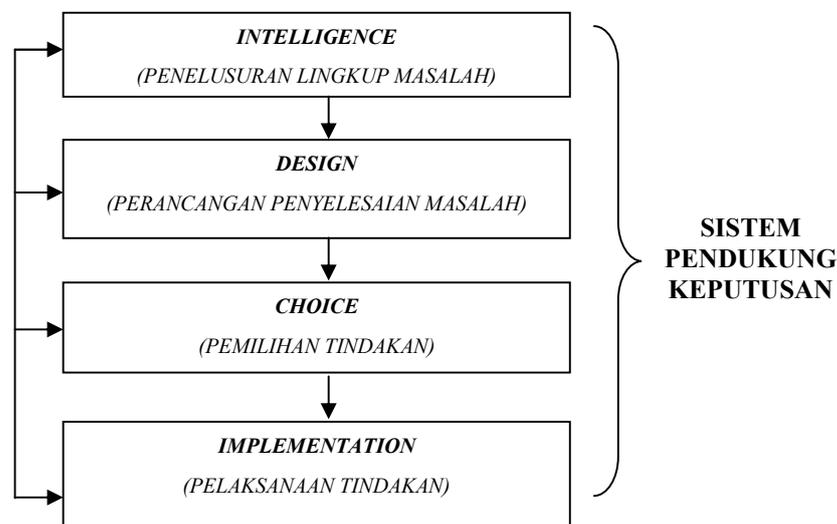
Pada tahap ini dilakukan proses pemilihan di antara berbagai alternatif tindakan yang mungkin dijalankan. Tahap ini meliputi pencarian, evaluasi, dan rekomendasi solusi yang sesuai untuk model yang telah

dibuat. Solusi dari model merupakan nilai spesifik untuk variabel hasil pada alternatif yang dipilih.

d. *Implementation*

Tahap implementasi adalah tahap pelaksanaan dari keputusan yang telah diambil. Pada tahap ini perlu disusun serangkaian tindakan yang terencana, sehingga hasil keputusan dapat dipantau dan disesuaikan apabila diperlukan perbaikan.

proses pengambilan keputusan, seperti terlihat pada gambar berikut.



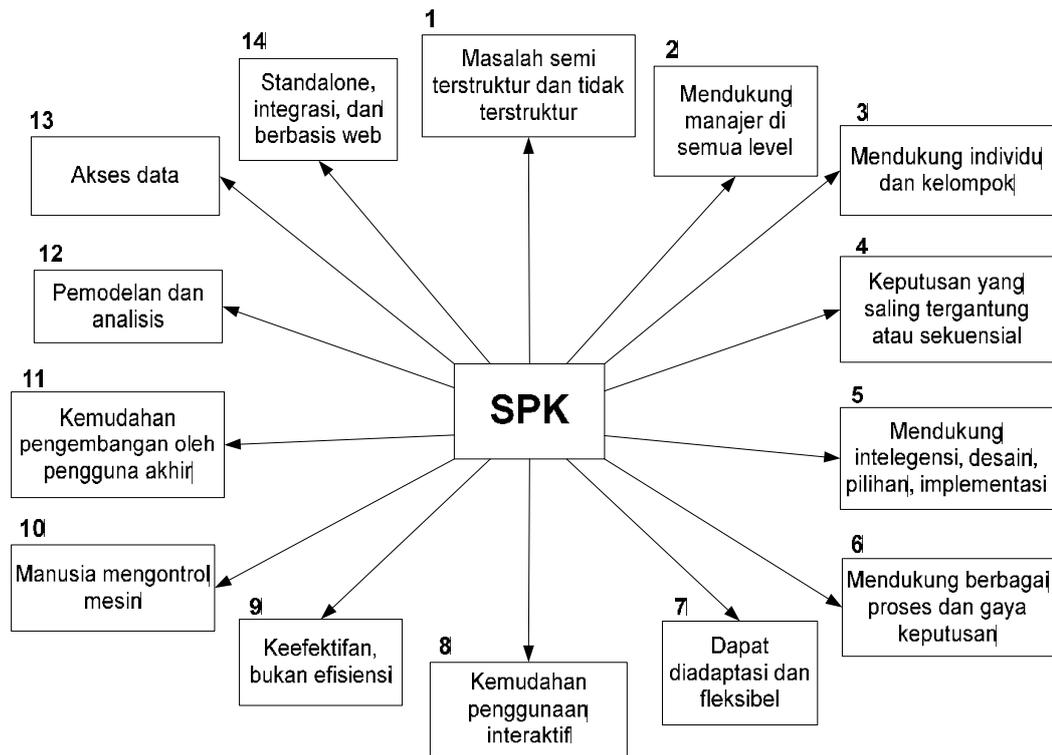
Gambar 2.1 Fase Proses Pengambilan Keputusan

2.1.3 Karakteristik Pengambilan Keputusan

Turban (2005) mengemukakan karakteristik dan kapabilitas kunci dari Sistem Pendukung Keputusan adalah sebagai berikut :

1. Dukungan untuk pengambil keputusan, terutama pada situasi semi terstruktur dan tak terstruktur.
2. Dukungan untuk semua level manajerial, dari eksekutif puncak sampai manajer lini.
3. Dukungan untuk individu dan kelompok.
4. Dukungan untuk semua keputusan independen dan atau sekuensial.
5. Dukungan di semua fase proses pengambilan keputusan: inteligensi, desain, pilihan, dan implementasi.

6. Dukungan pada berbagai proses dan gaya pengambilan keputusan.
7. Kemampuan sistem beradaptasi dengan cepat dimana pengambil keputusan dapat menghadapi masalah-masalah baru dan pada saat yang sama dapat menanganinya dengan cara mengadaptasikan sistem terhadap kondisi-kondisi perubahan yang terjadi.
8. Pengguna merasa seperti di rumah. *User-friendly*, kapabilitas grafis yang kuat, dan sebuah bahasa interaktif yang alami.
9. Peningkatan terhadap keefektifan pengambilan keputusan (akurasi, *timelines*, kualitas) dari pada efisiensi (biaya).
10. Pengambil keputusan mengontrol penuh semua langkah proses pengambilan keputusan dalam memecahkan masalah.
11. Pengguna akhir dapat mengembangkan dan memodifikasi sistem sederhana.
12. Menggunakan model-model dalam penganalisisan situasi pengambilan keputusan.
13. Disediaknya akses untuk berbagai sumber data, format, dan tipe, mulai dari sistem informasi geografi (GIS) sampai sistem berorientasi objek.
14. Dapat dilakukan sebagai alat *standalone* yang digunakan oleh seorang pengambil keputusan pada satu lokasi atau didistribusikan di satu organisasi keseluruhan dan di beberapa organisasi sepanjang rantai persediaan.



Gambar 2.2 Karakteristik dan Kapabilitas SPK

2.1.4 Keuntungan Sistem Pendukung Keputusan

Keuntungan dari sistem pendukung keputusan sebagai berikut :

1. Dapat memperluas kemampuan seseorang untuk mengambil keputusan dalam memproses data atau informasi pemakainya.
2. Membantu mengambil keputusan dalam hal penghematan waktu yang dibutuhkan untuk memecahkan masalah, terutama berbagai masalah yang sangat kompleks dan tidak terstruktur.
3. Dapat menghasilkan solusi dengan lebih cepat serta hasilnya dapat diandalkan.
4. Dapat menjadi stimulan bagi pengambil keputusan dalam memahami permasalahannya, karena sistem pendukung keputusan mampu menyajikan berbagai alternatif.
5. Mampu menyediakan bukti tambahan untuk memberikan pembenaran, sehingga dapat memperluas posisi pengambilan keputusan.

Sistem Pendukung Keputusan terdiri atas 3 komponen atau subsistem, sebagai berikut:

1. Subsistem Data (*Data Subsystem*)

Merupakan komponen sistem penunjang keputusan sebagai penyedia data bagi sistem yang mana data disimpan dalam *Database Manajement System* (DBMS), sehingga dapat diambil dan diekstraksi dengan cepat.

2. Subsistem Model (*Model Subsystem*)

Keunikan dari sistem ini adalah kemampuannya dalam mengintegrasikan data dengan model-model keputusan.

3. Subsistem Dialog (*User Sistem Interface*)

Melalui sistem dialog ini, sistem dapat diartikulasikan dan diimplementasikan sehingga pengguna atau pemakai dapat berkomunikasi dengan sistem yang dirancang.

2.2 Multi Criteria Decision Making

Menurut Bawono(1999) dalam buku berjudul “Analisis Pengambilan Keputusan Dengan Banyak Persyaratan (MCDM)”, MCDM melibatkan banyak tanda, banyak tujuan atau keduanya. Alternatif keputusan memiliki tanda atau atribut. Atribut adalah karakteristik atau kualitas dari beberapa alternatif. Pengambilan keputusan dengan multiatribut melibatkan pemilahan alternatif terbaik dari beberapa macam alternatif. Tujuannya adalah menghadirkan penerapan dari atribut. Tujuan akhir yang betul-betul diinginkan adalah sebagai tingkat sasaran atribut. Sementara sebuah ciri khas dari sebuah pilihan keputusan adalah sebuah atribut. Maksimasi atau minimasi yang merupakan ciri khas dari sebuah tujuan dan tujuan sasaran akhir dari untuk ciri khas sebuah tujuan akhir.

2.3 SAW (*Simple Additive Weighting*)

Metode SAW sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif dari semua atribut. Metode SAW membutuhkan proses

normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada. (Kusumadewi, 2005).

Rumus 2.1

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\max_i x_{ij}} & \text{jika } j \text{ atribut keuntungan (benefit)} \\ \frac{\min_i x_{ij}}{x_{ij}} & \text{jika } j \text{ atribut biaya (cost)} \end{cases}$$

dimana r_{ij} adalah rating kinerja ternormalisasi dari alternatif A_i pada atribut C_j ; $i=1,2,\dots,m$ dan $j=1,2,\dots,n$. Nilai preferensi untuk setiap alternatif (V_i) diberikan sebagai:

Rumus 2.2

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

Nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih. (Kusumadewi, 2005).

2.4 Basis Data

Database adalah kumpulan file-file yang mempunyai kaitan antara satu file dengan file yang lain sehingga membentuk satu bangunan data untuk menginformasikan satu perusahaan, instansi dalam batasan tertentu. (Kristanto, 2003)

Istilah-istilah yang digunakan dalam basis data:

- 1) *File* : merupakan kumpulan dari atribut *record-record* sejenis yang mempunyai panjang elemen yang sama, atribut yang sama namun berbeda-beda dalam data *value*-nya.

- 2) *Record* : merupakan kumpulan dari elemen-elemen yang saling berhubungan atau berkaitan menginformasikan tentang *entry* secara lengkap.
- 3) *Field* : merupakan sekumpulan tanda-tanda yang berbentuk kesatuan tersendiri, merupakan bagian terkecil dari *record* dan bentuknya unik dijadikan *field* kunci yang dapat mewakili *record*-nya.
- 4) *Entity* : merupakan tempat kejadian atau konsep yang informasikan direkam.

2.5 MySQL

MySQL merupakan software sistem manajemen database (*Database Management System - DBMS*) yang sangat populer di kalangan pemrograman *web*, terutama di lingkungan linux dengan menggunakan *script* PHP dan Perl. *Software* database ini kini telah tersedia juga pada *platform* sistem operasi *windows* (98/ME atupun NT/2000/XP) (Sidik, 2005).

MySQL merupakan *database* yang paling populer digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelolaan datanya. Kepopuleran MySQL dimungkinkan karena kemudahannya untuk digunakan, cepat secara kinerja *query*, dan mencukupi untuk kebutuhan *database* perusahaan-perusahaan skala menengah kecil. MySQL merupakan *database* yang digunakan oleh situs-situs terkemuka di *internet* untuk menyimpan datanya (Sidik, 2005).

Software database MySQL kini dilepas sebagai *software* manajemen *database* yang *open source*, sebelumnya merupakan *software database* yang *shareware*. *Shareware* adalah suatu *software* yang dapat didistribusikan secara bebas untuk keperluan penggunaan secara pribadi, tetapi jika digunakan secara komersial maka pemakai harus mempunyai lisensi dari pembuatnya. *Software open source* menjadikan *software* dapat didistribusi secara bebas dan dapat dipergunakan untuk keperluan pribadi maupun komersial, termasuk di dalamnya *source code* dari *software* tersebut (Sidik, 2005).

Database MySQL tersedia secara bebas cuma-cuma dan boleh dipergunakan oleh setiap orang, dengan lisensi *open source* GNU *General Public License* (GPL) ataupun lisensi komersial non GPL. saat ini diperkirakan lebih dari 3 juta pemakai di seluruh dunia, dengan lebih dari setengah juta *server* yang memasangnya, termasuk di dalamnya Yahoo!, MP3.com, Motorola, NASA, Silicon Graphics, HP, Xerox, Cisco dan Texas Instruments (Sidik, 2005).

Database MySQL, merupakan *database* yang menjanjikan sebagai alternatif pilihan *database* yang dapat digunakan untuk sistem *database* personal atau organisasi. Oracle sebagai *database* besar telah membuat *kit* (modul) untuk memudahkan proses migrasi dari MySQL ke dalam Oracle, hal ini dapat menunjukkan bahwa oracle telah memperhitungkan *database* MySQL sebagai *database* alternatif masa depan. Demikian juga dengan pengguna dari *database* MySQL, menunjukkan makin banyaknya perusahaan besar menggunakannya (Sidik, 2005).

Keistimewaan yang dimiliki MySQL adalah sebagai berikut.

1. *Portability*

MySQL dapat berjalan stabil pada berbagai sistem operasi diantaranya seperti Windows, Linux, FreeBSD dan masih banyak lagi.

2. *Open Source*

MySQL didistribusikan secara *open source* (gratis).

3. *Multiuser*

MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

4. *Performance Tuning*

MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan memproses lebih banyak SQL persatuan waktu.

5. *Column Types*

MySQL memiliki tipe kolom yang sangat kompleks.

6. *Command dan Functions*

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *SELECT* dan *WHERE* dalam *query*.

7. *Security*

MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta password terenkripsi.

8. *Scalability dan Limits*

MySQL mampu menangani *database* dalam skala besar, dengan jumlah record lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. Selain itu, batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

9. *Connectivity*

MySQL dapat melakukan koneksi dengan *client* menggunakan protocol TCP/IP, Unix socket (Unix), atau Named Pipes (NT).

10. *Localization*

MySQL dapat mendeteksi pesan kesalahan (*error code*) pada *client* dengan menggunakan lebih dari dua puluh bahasa.

11. *Interface*

MySQL memiliki *Interface* (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

12. *Client dan Tools*

MySQL dilengkapi dengan berbagai tool yang dapat digunakan untuk administrasi *database*, dan pada setiap tool yang ada pada petunjuk *online*.

13. Struktur Tabel

MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE*, dibandingkan *database* lainnya semacam PostgreSQL ataupun Oracle.

2.6 HTML (Hyper Text Markup Language)

HTML merupakan kepanjangan dari *Hyper Text Markup Language* adalah suatu bahasa yang digunakan untuk membuat halaman-halaman hypertext (hypertext page) pada internet. Dengan konsep hypertext ini, untuk membaca suatu dokumen anda tidak harus melakukannya secara urut, baris demi baris, atau halaman demi halaman. Tetapi anda tidak dapat dengan mudah melompat dari satu topik ke topik lainnya yang anda sukai, seperti halnya jika anda melakukan pada online Help

dari suatu aplikasi Windows. HTML dirancang untuk digunakan tanpa tergantung pada suatu platform tertentu (platform independent).

2.7 PHP (Hypertext Preprocessor)

PHP diciptakan oleh Rasmus Lerdorf, seorang pemrogram C yang handal. Semula PHP hanya digunakan untuk mencatat jumlah pengunjung pada homepagenya. Rasmus adalah seorang pendukung open source. Karena itulah ia mengeluarkan Personal Home Page Tools versi 1.0 secara gratis. Setelah mempelajari YACC dan GNU Bison, Rasmus menambah kemampuan PHP 1.0 dan menerbitkan PHP 2.0. PHP mudah dibuat dan cepat dijalankan, PHP dapat berjalan dalam web server yang berbeda dan dalam sistem operasi yang berbeda pula. PHP dapat berjalan di sistem operasi UNIX, Windows 98, Windows XP, Windows NT, dan Macintosh.

2.7.1 Sejarah PHP

PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP bernama FI (Form Interpreted). Pada saat tersebut PHP adalah sekumpulan script yang digunakan untuk mengolah data form dari web. Perkembangan selanjutnya adalah Rasmus melepaskan kode sumber tersebut dan menamakannya PHP/FI, pada saat tersebut kepanjangan dari PHP/FI adalah Personal Home Page/Form Interpreter. Dengan pelepasan kode sumber ini menjadi open source, maka banyak programmer yang tertarik untuk ikut mengembangkan PHP. Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini interpreter sudah diimplementasikan dalam C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan. Pada tahun 1997, sebuah perusahaan bernama Zend, menulis ulang interpreter PHP menjadi lebih bersih, lebih baik dan lebih cepat. Kemudian pada Juni 1998 perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan nama rilis tersebut menjadi PHP 3.0. Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai. Versi ini banyak dipakai sebab versi ini mampu dipakai untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan proses dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Versi ini adalah versi mutakhir dari PHP. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Dalam versi ini juga dikenalkan model pemrograman berorientasi objek baru untuk menjawab perkembangan bahasa pemrograman ke arah pemrograman berorientasi objek.

2.7.2 Pengertian PHP

Menurut Luke Welling dan Laura Thomson(2001,p1), PHP adalah *server-side scripting language* yang didesain secara spesifik untuk web. Dalam *page HTML*, dapat dimasukkan *code* PHP yang akan dieksekusi setiap kali halaman dikunjungi. PHP *code* diterjemahkan di *web-server* dan dirubah menjadi HTML atau output lain yang akan dilihat oleh pengunjung halaman. PHP adalah bahasa pemrograman script yang paling banyak dipakai saat ini. PHP banyak dipakai untuk memprogram situs web dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain. Contoh terkenal dari aplikasi PHP adalah phpBB dan MediaWiki (software di belakang Wikipedia). PHP juga dapat dilihat sebagai pilihan lain dari ASP.NET/C#/VB.NET Microsoft, ColdFusion Macromedia, JSP/Java Sun Microsystems, dan CGI/Perl. Contoh aplikasi lain yang lebih kompleks berupa CMS yang dibangun menggunakan PHP adalah Mambo, Joomla!, Postnuke, Xaraya, dan lain-lain.

2.7.3 Kelebihan PHP

1. PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. Web Server yang mendukung PHP dapat ditemukan dimana - mana dari mulai IIS sampai dengan apache, dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena referensi yang banyak. PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (linux, unix, windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah system.

2.7.4 Kekurangan PHP

Selain berbagai kemudahan, PHP memiliki beberapa masalah atau kekurangan. Pertama, dari segi bahasa ia bukanlah bahasa yang ideal untuk pengembangan berskala besar. Kekurangan utama adalah tidak adanya namespace. Namespace merupakan sebuah cara untuk mengelompokkan nama variabel atau fungsi dalam susunan hirarkis

2.8 World Wide Web

Web adalah suatu sistem informasi yang berbasis grafis yang menampilkan data berupa teks, gambar, animasi dan bunyi, yang diantara data tersebut saling berhubungan dengan yang lainnya di internet (Wahana, 2001).

Jenis – jenis *web* menurut kamus istilah internet penerbit Wahana Komputer Semarang adalah :

1. *Web Browser*

Suatu program dimana kita dapat mengambil dokumen HTML dari *Web Server* dengan menggunakan *protocol* dan format http (*Hypertext Transfer Protocol*) yang kemudian bisa ditampilkan.

2. *Web Edit*

Web edit adalah editor teks HTML berbasis *windows*, seperti : Ms Frontpage dan Note Pad

3. *Web Page*

Kemasan yang bersisi suara, grafis, film dan teks yang berasal dari *file-file* data HTML.

4. *Web Server*

Merupakan sistem komputer disuatu organisasi yang berfungsi sebagai server untuk fasilitas *Word Wide Web* dan dapat diakses oleh seluruh pemakai internet, seperti PWS, IIS, dan Apache.

5. *Web Site*

Web Site sering disebut *Web* atau *Home Page* adalah tempat atau alamat *internet* yang merupakan sampul halaman yang berisikan daftar isi atau menu dari sebuah situs *web*. *Website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar

diam atau gerak, animasi, suara, dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*).

2.9 Browser Web

Browser web adalah *software* yang digunakan untuk menampilkan informasi dari *server web*. *Software* ini kini telah dikembangkan dengan menggunakan *user interface grafis*, sehingga pemakai dapat dengan mudah melakukan ‘*point*’ dan ‘*click*’ untuk pindah antar dokumen.

Lynx adalah *browser web* yang masih menggunakan mode *teks*, yang akibatnya adalah tidak ada gambar yang dapat ditampilkan. *Lynx* ini ada pada lingkungan DOS dan Unix (keluarga sistem operasi Unix). Namun perkembangan dari *browser mode teks* ini tidaklah secepat *browser web* dengan GUI.

Dapat dikatakan saat ini hanya ada 2 *browser web* GUI yang populer yaitu: Internet Explorer dan Netscape Navigator. Kedua *browser* ini bersaing untuk merebut pemakainya, dengan berusaha mendekati standar spesifikasi dokumen HTML yang direkomendasikan oleh W3C.

2.10 Web Server

Suatu aplikasi yang jalan pada suatu situs web dan bertanggung jawab untuk merespon permintaan file dari *web browser*. Dalam melakukan permintaan suatu halaman pada suatu situs *web*, *browser* melakukan koneksi ke suatu *server* dengan protokol HTTP. *Server* akan menanggapi koneksi tersebut dengan mengirimkan isi file yang diminta dan memutuskan koneksi tersebut. Web Browser kemudian memformat informasi yang didapat dari *server*. Pada bagian *server*, *browser* yang berbeda dapat melakukan koneksi pada *server* yang sama untuk memperoleh informasi yang sama. Dalam memberikan halaman yang diminta *web server* dapat melakukan kerja sama dengan server lain seperti Aplikasi *Server*.

2.11 Rekayasa Perangkat Lunak

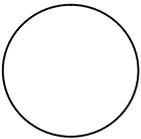
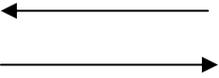
2.11.1 Diagram Konteks

Diagram Konteks merupakan tingkatan tertinggi dalam diagram aliran data dan hanya memuat satu proses, menunjukkan sistem secara keseluruhan. Proses tersebut diberi nomor nol. Semua entitas eksternal yang ditunjukkan pada diagram konteks berikut aliran data-aliran data utama menuju dan dari sistem. Diagram tersebut tidak memuat penyimpanan data dan tampak sederhana untuk diciptakan, begitu entitas-entitas eksternal serta aliran data-aliran daa menuju dan dari sistem diketahui penganalisis dari wawancara dengan *user* dan sebagai hasil analisis dokumen (Pressman, 2002).

Diagram Konteks menggarisbawahi sejumlah karakteristik penting dari suatu sistem (Pressman, 2002):

1. Kelompok pemakai, pihak yang akan memberikan data ke sistem
2. Data, apa saja yang diterima/dihasilkan sistem dari/ke dunia luar
3. Penyimpanan data, tempat sistem harus memberi informasi atau laporan
4. Batasan, yang membedakan antara sistem dan lingkungan

Tabel 2.1. Simbol-simbol Diagram Konteks (DFD)

Simbol	Keterangan
	<i>Terminator</i> Lingkungan luar sistem yang terlibat dalam sistem yang mempengaruhi kerja sistem
	Sistem Simbol ini menunjukkan sistem yang dibuat
	<i>Data Flow</i> Simbol ini digunakan untuk menggambarkan aliran data yang berjalan

2.12.2 DFD (*Data Flow Diagram*)

Data Flow Diagram (DFD) adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses

fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. DFD ini sering disebut juga dengan nama *Bubble chart*, *Bubble diagram*, model proses, diagram alur kerja, atau model fungsi (Pressman, 2002).

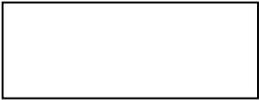
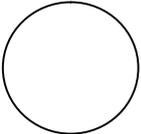
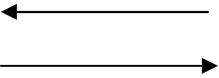
Pada saat informasi mengalir melalui perangkat lunak, dimodifikasi oleh suatu deretan transformasi. Diagram alir data/ DFD (*data flow diagram*) adalah sebuah teknis grafis yang menggambarkan aliran informasi dan transformasi yang diaplikasikan pada saat data bergerak dari *input* menjadi *output* (Pressman, 2002).

Berikut ini adalah aturan-aturan pembuatan (DFD) :

1. Didalam *Data Flow Diagram* (DFD) tidak boleh menghubungkan antara *external entity* dengan *entity* lainnya secara langsung.
2. Didalam *Data Flow Diagram* (DFD) tidak boleh menghubungkan *data store* yang satu dengan *data store* yang lainnya secara langsung.
3. Didalam *Data Flow Diagram* (DFD) tidak boleh menghubungkan *data store* dengan *external entity* secara langsung.
4. Setiap proses harus memiliki *data flow* yang masuk dan juga *data flow* yang keluar.

Berikut adalah simbol-simbol DFD dalam Pressman (2002) :

Tabel 2.2. Simbol-simbol *Data Flow Diagram* (DFD)

Simbol	Keterangan
	<i>External Entity</i> Simbol ini digunakan untuk menggambarkan asal atau tujuan data
	Proses Simbol ini menunjukan proses pengolahan atau tranformasi data
	<i>Data Flow</i> Simbol ini digunakan untuk menggambarkan aliran data yang berjalan

	<p><i>Data Store</i></p> <p>Simbol ini menggambarkan <i>data Flow</i> yang sudah disimpan atau diarsipkan</p>
---	---

Jenis-jenis DFD (*Data Flow Diagram*) dalam Pressman (2002) :

1. *Context Diagram* (CD)

Jenis pertama *Context Diagram*, adalah *data flow diagram* tingkat atas (DFD *Top Level*), yaitu diagram yang paling tidak detail, dari sebuah sistem informasi yang menggambarkan aliran-aliran data ke dalam dan ke luar sistem dan ke dalam dan ke luar entitas-entitas eksternal. (CD menggambarkan sistem dalam satu lingkaran dan hubungan dengan entitas luar. Lingkaran tersebut menggambarkan keseluruhan proses dalam sistem).

2. *Diagram Level n / Data Flow Diagram Levelled*

Dalam diagram n DFD dapat digunakan untuk menggambarkan diagram fisik maupun diagram diagram logis. Dimana *Diagram Level n* merupakan hasil pengembangan dari *Context Diagram* ke dalam komponen yang lebih detail tersebut disebut dengan *top-down partitioning*. Jika kita melakukan pengembangan dengan benar, kita akan mendapatkan DFD-DFD yang seimbang.

a. DFD Fisik

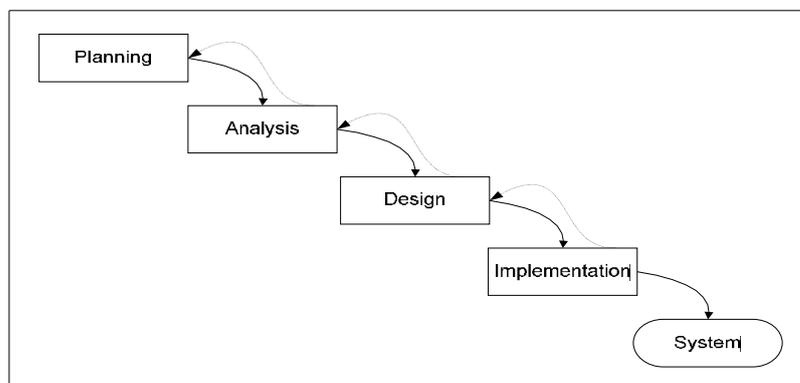
Adalah representasi grafik dari sebuah sistem yang menunjukkan entitas-entitas internal dan eksternal dari sistem tersebut, dan aliran-aliran data ke dalam dan keluar dari entitas-entitas tersebut.

b. DFD Logis

Adalah representasi grafik dari sebuah sistem yang menunjukkan proses-proses dalam sistem tersebut dan aliran-aliran data ke dalam dan ke luar dari proses-proses tersebut. Kita menggunakan DFD logis untuk membuat dokumentasi sebuah sistem informasi karena DFD logis dapat mewakili logika tersebut, yaitu apa yang dilakukan oleh sistem tersebut, tanpa perlu menspesifikasi dimana, bagaimana, dan oleh siapa proses-proses dalam sistem tersebut dilakukan.

2.12.3 Metode Pengembangan Perangkat Lunak

Pada metode penelitian ini dilakukan rekayasa perangkat lunak yang digunakan adalah *System Development Life Cycle* model *Waterfall* seperti pada gambar berikut ini (Wixom, 2003):



Gambar 2.3 Metode Pengembangan Model *Waterfall*

Keterangan:

1. *Planning* (Perencanaan)

Tahap perencanaan merupakan proses penting untuk mengetahui mengapa sistem harus dibuat dan menentukan bagaimana cara membangun sistem tersebut. Langkah pertama dari proses tersebut adalah dengan mengidentifikasi peluang apakah dapat memberikan kemungkinan biaya rendah tetapi menghasilkan keuntungan.

2. *Analysis* (Analisis)

Analisis sistem dilakukan untuk memberikan jawaban pertanyaan siapa yang akan menggunakan sistem. Apa yang akan dilakukan oleh sistem, dimana dan kapan sistem tersebut digunakan. Pada tahap ini pembuat sistem akan melakukan observasi dan pengamatan terhadap sistem yang lama, kemudian mengidentifikasi, memanfaatkan dan mengembangkan peluang, dan membangun konsep untuk sebuah sistem baru.

3. *Design* (perancangan)

Tahap perancangan dilakukan untuk menetapkan bagaimana sistem akan dioperasikan. Hal ini berkaitan dengan menentukan perangkat keras, perangkat lunak, jaringan, tampilan program, *form* dan laporan yang akan dipakai. Selain itu perlu juga menspesifikasi program, database dan file yang dibutuhkan.

4. *Implementation* (Implementasi)

Merupakan tahap berikutnya untuk menerjemahkan data atau pemecahan masalah yang telah dirancang ke dalam bahasa pemrograman komputer yang telah ditentukan. Semua tahap ini desain perangkat lunak sebagai sebuah program lengkap atau unit program.

5. *System* (Sistem)

Tahapan ini, merupakan hasil sistem yang telah dibuat dalam bentuk perangkat lunak yang telah dipasang dan digunakan, termasuk didalamnya proses pemeliharaan dan perbaikan kesalahan. Perangkat lunak yang telah selesai dibuat dapat mengalami perubahan-perubahan atau penambahan sesuai dengan permintaan *user* atau perubahan sistem.