BAB II

LANDASAN TEORI

2.1. Konsep Dasar *Web*

A. *Website*

Situs *web* (bahasa inggris *website*) atau sering disingkat dengan istilah situs. Menurut Widodo (2013:1), "sebuah halaman yang berisi tulisan-tulisan yang lazimnya dilengkapi dengan gambar-gambar maupun foto-foto. Tulisan-tulisan tersebut di tulis dengan berbagai macam tujuan. Ada yang bertujuan untuk bisnis, pendidikan, promosi barang, maupun informasi hal-hal tertentu".

Menurut Yuhefizar (2008:9), "*Web* adalah suatu metode untuk menampilkan informasi di internet, baik berupa teks, gambar, suara, maupun video yang interaktif dan mempunyai kelebihan untuk menghubungkan *(link)* satu dokumen dengan dokumen lainnya *(hypertext)* yang dapat diakses melalui sebuah *browser*".

BINA SARANA INFORMATIKA
UNIVERSITAS

B. *Internet*

Menurut Sibero (2011:10) "*Internet* (*Interconnected Network)* adalah jaringan komputer yang menghubungkan antar jaringan secara global, internet dapat juga dapat disebut jaringan alam suatu jaringan komputer area. Seperti halnya jaringan komputer lokal maupun jaringan komputer area, internet juga menggunakan protokol komunikasi yang sama yaitu *TCP/IP* (*Tranmission Control Protocol/Internet Protocol)".*

Menurut Sarwono dan K. Prihatono A.H (2012:17), "*Internet* merupakan sekumpulan jaringan yang berskala global. Tidak ada satu pun orang, kelompok atau organisasi yang bertanggung jawab untuk menjalankan *internet*".

Internet yang kita kenal saat ini pertama kali dikembangkan tahun 1969 dengan nama *ARPANET* (*US Defense Advanced Research Projecst Agency*) oleh Departemen Pertahanan Amerika Serikat. Kejadian ini berlangsung dua bulan setelah Neil Amstrong melangkah ke Bulan. Diawali dari suatu proyek yang dinamakan *Advance Research Project Agency Network* (*ARPANET*). Karena perkembangan sangat pesat, jaringan ini tidak dapat disebut sebagai *ARPANET*, konsep ini kemudian berkembang dan dikenal sebagai konsep *internetworking.*

Adapun layanan multimedia internet atau yang dikenal *Word Wide Web* (*WWW*) merupakan aplikasi internet yang paling diminati para pengakses. *WWW* disukai karena mencangkup sumber daya multimedia, antara lain suara, gambar, video, audio dan animasi. Oleh karena itu, aplikasi demikian menjadi semacam sarana untuk menyajikan informasi yang interaktif dan atraktif. Layanan *WWW* lebih dikenal dengan sebutan *web*.

C. *HTML (Hyper Text Markup Language)*

Menurut Sibero (2012:19), "*HTML* (*Hyper Text Markup Language)* adalah bahasa yang digunakan pada dokumen *web* sebagai bahasa untuk pertukaran dokumen *web*".

Menurut Kustiyahningsih dan Anamisa (2011:13), "*HTML* kependekan *Hyper Text Markup Language*. Dokumen *HTML* adalah *text file* murni yang dapat

dibuat dengan editor teks sembarang. Dokumen ini dikenal sebagai *web page*. *File-file HTML* ini berisi instruksi-instruksi yang kemudian diterjemahkan oleh *browser* yang ada di komputer *client* (*user*) sehingga isi informasinya dapat di tampilkan secara visual di komputer pengguna".

Berdasarkan pendapat para ahli yang dikemukakan di atas dapat ditarik kesimpulan bahwa *HTML* adalah bahasa yang sangat tepat dipakai untuk menampilkan informasi pada halaman *web*, karena *HTML* menampilkan informasi dalam bentuk *hypertext* dan juga mendukung sekumpulan perintah yang dapat digunakan untuk mengatur tampilnya informasi tersebut, sesuai dengan namanya, bahasa ini menggunakan tanda (*markup*) untuk menandai perintah-perintahnya.

D. *PHP (Personal Home Page)*

Menurut Anhar (2012:3), "*PHP* singkatan dari *Hypertext Preprocessor* yaitu bahasa pemrograman *web server-side* yang bersifat *open source*".

Menurut Sibero (2012:49), "*PHP* adalah pemrograman (interpreter) adalah proses penerjemahan baris sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan".

Dengan *PHP*, halaman *web* dapat dibuat lebih interaktif. Misalnya dengan memasukan fungsi untuk buku tamu, pengisian formulir, *hot counter*, dan masih banyak lagi. *PHP* juga akan dipermudah ketika melakukan *updating website* karena *PHP* dapat berhubungan langsung dengan *database.*

UNIVERSITAS INFORMATIKA BSI

Secara teknis, *database* menyimpan data yang diolah oleh bahasa pemrograman *PHP* yang ada di *web server* kemudian data tersebut ditampilkan dengan *web browser* dengan cara menuliskan alamat/*URL* sesuai nama *file* yang terdapat pada *web server*.

Salah satu kelebihan dari *PHP* adalah mampu berkomunikasi dengan berbagai *database* yang terkenal diantaranya *BASE, DBM, FilePro (Personix, Inc), Informix, Ingres, Interbase, Microsoft, MYSQL, MSQL, Oracle, PostgresSQL, Sysbase,* MariaDB. Dengan demikian, menampilkan data yang bersifat dinamis, yang diambil dari *database*, hal yang mudah untuk diimplementasikan. Itu sebabnya *PHP* sangat cocok untuk membangun halaman-halaman *web* dinamis.

E. *CSS (Cascading Style Sheet)*

Menurut Khafidli (2011:30), "*CSS* (*Cascading Style Sheet*) merupakan salah satu bahasa pemograman *web* yang berguna untuk mengendalikan beberapa elemen dalam sebuah *web* sehingga lebih terstruktur dan seragam". Pada umumnya, *CSS* digunakan untuk memformat tampilan halaman *web* yang dibuat dengan menggunakan *HTML*.

CSS bekerja sebagai pelengkap pada *HTML*. Penggunaan *CSS* dilakukan untuk memperluas kemampuan *HTML* dalam memformat dokumen *web* atau untuk mempercantik tampilan *web*. Penulisan kode *CSS* disisipkan pada *tag HTML*. Kode *CSS* ditulis dalam *tag* <*style*> dan </*style*> dengan membuat (mendefinisikan) suatu *style* baru yang kemudian dapat digunakan berulang kali. Penulis kode *CSS* dapat

langsung pada dokumen *HTML* atau disimpan dalam dokumen tersendiri kemudian dipanggil untuk digunakan.

F. *Jquery*

JQuery merupakan salah satu *java script frame work* terbaik saat ini *Jquery* dikembangkan oleh John Resig pada tahun 2006 di BarmCamp NYC. Pada awal perkembangannya Jquery pertama dibuat untuk meringkas pengguna *CSS selector* dalam suatu pustaka fungsi. *Jquery* memiliki ciri khas pada perintahnya, *prefix* untuk *Jquery* dengan tanda $ kemudian dilanjutkan dengan fungsi atau perintah.

Menurut Hakim (2013:3) "*Jquery* adalah *(Javascript Library)* kumpulan kode atau fungsi *javascript* siap pakai. Sehingga mempermudah dan mempercepat kita dalam membuat kode *javascript"*.

Maka diperlukan kode yang cukup panjang bahkan terkadang sulit dipahami. Disinilah peran *Jquery* sebagai *Javascript Library*, dimana kita langsung memanggil fungsi yang terdapat didalam *Library* tersebut *Jquery* juga bertujuan untuk menyederhanakan kode *javascript*. Hal ini sesuai dengan slogannya *"Write less do more"* cukup menulis sedikit, akan tetapi bisa melakukan banyak hal.

G. *MySQL* (*My structured Query Language*)

Menurut Arief (2011:152) "*MySQL* adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengolahan datanya".

MySQL dikembangkan oleh perusahaan Swedia bernama *MySQL AB* yang pada saat ini bernama *Tcx DataKonsult AB* sekitar tahun 1994-1995, namun cikal-bakal kodenya sudah ada sejak tahun 1979. Awalnya *Tcx* merupakan perusahaan

pengembang *software* dan konsultan *database*, dan saat ini *MySQL* sudah diambil alih oleh *Oracle Corp*.

Kepopuleran *MySQL* antara lain karena *MySQL* menggunakan *SQL* sebagai bahasa dasar untuk mengakses *database*-nya sehingga mudah untuk digunakan, kinerja *query* cepat, dan mencukupi untuk kebutuhan *database* perusahaan-perusahaan yang berskala kecil sampai menengah, *MySQL* juga bersifat *open source* (tidak berbayar).

MySQL merupakan *database* yang pertama kali didukung oleh bahasa pemrograman *script* untuk *internet* (*PHP* dan *Perl*). *MySQL* dan *PHP* dianggap sebagai pasangan *software* pembangun aplikasi *web* yang ideal. *MySQL* lebih sering digunakan untuk membangun aplikasi berbasis *web*, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman *script PHP*.

H. *Sublime Text 3*

UNIVERSITAS

Sublime Text merupakan teks editor berbagai bahasa pemrograman mulai dari *C* hingga *Java*, menurut penulis *Sublime Text* merupakan teks editor yang sangat baik dan nyaman untuk digunakan sebagai alat berkoding. Karena tampilannya yang minimalis namun elegan, bawaannya yang ringan namun diboboti dengan keakuratan dalam penulisan yang mengikuti algoritma dan aturan setiap bahasa pemrograman yang digunakan. Kini pada versi terbaru *Sublime Text* sudah dapat dijalankan pada *distro linux Debian* dan turunannya seperti *Ubuntu*.

Sublime Text yang saat ini ada mencegah *plugin* merusak *Sublime Text*, dan mempercepat pembukaan aplikasi di awal. Selain peningkatan kecepatan, sekarang

Sublime Text juga menggunakan *Python* 3.3 untuk setiap *plugin*, serta memperkenalkan *symbol indexing* yang membaca semua *file* dalam sebuah proyek untuk dibuat indeks simbol pada saat pencarian.

Berikut adalah fitur-fitur *Sublime Text 3*:

a. *Symbol Indexing*

Sublime Text sekarang memindai *file* dalam proyek dan membangun sebuah indeks yang berisi *file* yang simbol. Ini fitur baru *Goto* Definisi dan *Goto* Simbol di Proyek, yang keduanya tersedia dari menu *Goto*. *Goto* Definisi membawa definisi simbol di bawah tanda sisipan, sementara *Goto* Simbol di Proyek meminta untuk memilih simbol melalui pencocokan *fuzzy*, dan kemudian membawa Anda ke definisi simbol itu.

b. *Management Pane*

Bekerja dengan beberapa panel sekarang lebih efisien, dengan perintah untuk menciptakan dan menghancurkan panel, dan cepat memindahkan *file* antara panel. Sehingga dapat melihat pilihan baru di bawah *View* / *Group*, *View* / *Focus Group* dan *View* /*Move File Group*.

c. Kecepatan

Sublime Text selalu memiliki kecepatan sebagai fitur, namun versi 3 memiliki beberapa titik lemah. Waktu *startup* sekarang hampir cepat, dan *plugin* tidak lagi memiliki kesempatan untuk membawa ini ke bawah. Ganti semua kinerja juga secara signifikan lebih cepat.

Sublime Text sekarang menggunakan *Python* 3.3 untuk *plugin*, dan berjalan keluar dari proses, sehingga setiap *plugin* yang memuat kode asli ada resiko lagi dalam proses *Sublime Text* utama.

I. *XAMPP*

Menurut Peter (2007:12) *XAMPP* (*Exentensible Messaging and Presence Protokol*) merupakan "Sebuah *software* yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program, dan *XAMPP* adalah program yang sangat mudah digunakan, gratis dan sangat mudah digunakan dalam sistem operasi *linux* dan *windows*". *XAMPP* merupakan singkatan dari *X* (empat sistem operasi apapun), *Apache, Mysql, PHP, Perl*. *XAMPP* merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dalam paketnya sudah terdapat *Apache* (*web server*), *MySQL* (*database*), *PHP* (*server side scripting*), *Perl, FTP server, phpMyadmin* dan berbagai pustaka bantu lainnya. Dengan menginstall *XAMPP* maka tidak perlu melakukan instalasi dan konfigurasi *web server Apache*, *PHP* dan *MySQL*. Masing-masing huruf dari *XAMPP* memiliki kepanjangan, yaitu:

X : program ini dapat dijalankan pada *window, linux, mac OS*.

Apache : sebuah aplikasi *server web open source*.

Mysql : sebuah aplikasi *server database open source*.

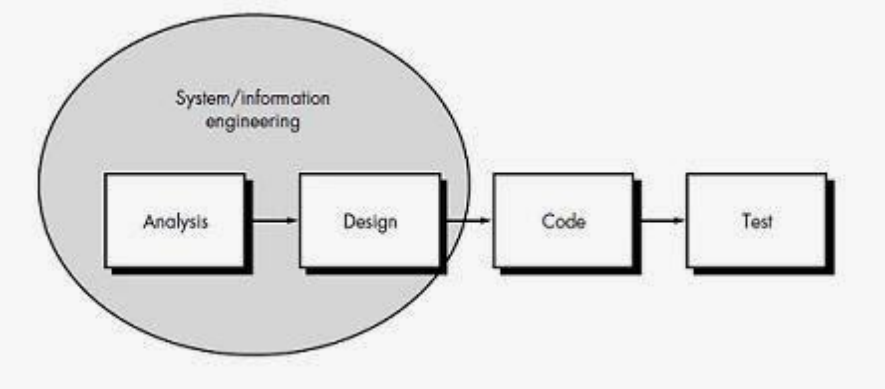
PHP : sebuah aplikasi pemrograman untuk *web*.

Perl : sebuah aplikasi pemrograman *system*.

UNIVERSITAS BINA SARANA INFORMATIKA

J. Model Pengembangan Perangkat Lunak *Waterfall*

Model *SDLC* air terjun (*waterfall*) Menurut A.S Rosa dan Shalahuddin (2014:28) yaitu "Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung (*support*)."



Sumber: A.S Rosa dan Shalahuddin (2014:28)

Gambar II.I Ilustrasi Model *Waterfall*

1. Analisis Kebutuhan

   Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean.

3. Pengkodean

Desain harus ditranslasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian Unit

Pengujian fokus pada perangkat lunak secara dari segi lojik dan fungsional dan memastikan semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dipastikan sesuai dengan yang diinginkan.

5. Pemeliharaan (*maintenance)*

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari tahap analisis spesifikasi untuk perubahan perangkat lunak baru.

Menurut A.S Rosa dan Shalahuddin (2014:30), “hal positif dari model air terjun adalah struktur tetap pengembangan sistem jelas, dokumentasi dihasilkan disetiap tahap pengembangan, dan sebuah tahap dijalankan setelah tahap sebelumnya selesai dijalankan (tidak ada tumpang tindih pelaksanaan tahap)”.

K. Sistem Informasi

Pengertian sistem informasi menurut Sutabri (2012:38) adalah “Suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian”. Mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang dibutuhkan oleh pihak luar tertentu. Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*Building Block*) menurut Sutabri (2012:38), yang terdiri dari:

a. Blok Masukan (*Input Block*)

   Merupakan input yang mewakili data yang masuk ke dalam sistem informasi, yang dapat berupa dokumen-dokumen dasar.

b. Blok Model (*Model Block*)

   Blok ini terdiri dari kombinasi prosedur, logika model matematika, yang akan memanipulasi data *input* dan data yang tersedia dalam basis data untuk memperoleh keluaran yang diinginkan.

c. Blok Keluaran (*Output Block*)

   Produk dalam sistem informasi adalah keluaran yang berupa informasi yang berkualitas dan dokumen yang berguna.

BINA SARANA INFORMATIKA UNIVERSITAS

d. Blok Teknologi (*Technology Block*)

Teknologi merupakan "kotak alat" dalam sistem informasi, teknologi terdiri dari 3 (tiga) bagian utama, yaitu:

1) Teknisi (*Brainware*)

2) Perangkat lunak (*Software*)

3) Perangkat keras (*Hardware*)

e. Blok Basis Data (*Database Block*)

Basis data merupakan kumpulan data yang saling berhubungan dan tersimpan di perangkat keras dan digunakan diperangkat lunak untuk memanipulasinya. Karena itu, data perlu disimpan dan diorganisasikan dalam basis data untuk menghasilkan informasi yang berkualitas.

f. Blok Kendali (*Control Block*)

Blok yang digunakan untuk mengatasi hal-hal yang dapat merubah sistem, baik yang berupa sabotase, ketidak efisienan, kecurangan, kegagalan dan sebagainya.

BINA SARANA INFORMATIKA UNIVERSITAS

2.2. Teori Pendukung Web

Untuk mendesain suatu model sistem informasi maka diperlukan peralatan pendukung (*tool system*) untuk menggambarkan bentuk sistem secara struktural dan aktual dengan suatu pendekatan analisa terstruktur, *tools sytem* juga berguna untuk mempermudah dalam pembuatan program aplikasi yang akan dibuat, contohnya adalah dalam pembuatan logika dan algoritma program mulai dari *input, process* dan *output* yang akan dihasilkan. Adapun *tools system* yang akan dijelaskan sebagai model sistem yang akan dirancang, yaitu:

1. Struktur Navigasi

Menurut Suyanto (2008:62) "Struktur navigasi dalam situs *web* melibatkan sistem navigasi situs *web* secara keseluruhan dan desain *interface* situs *web* tersebut, navigasi memudahkan jalan yang mudah ketika menjelajahi situs *web*". Struktur navigasi juga dapat diartikan sebagai struktur alur dari suatu program yang merupakan rancangan hubungan dan rantai kerja dari beberapa area yang berbeda dan dapat membantu mengorganisasikan seluruh elemen pembuatan *website*. Ada empat macam bentuk dasar yaitu:

a. Struktur Navigasi *Linear* (Satu Alur)

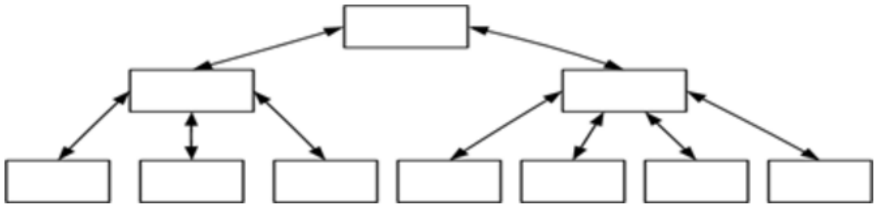
Struktur navigasi *linear* hanya mempunyai satu rangkaian cerita yang berurut yang menampilkan satu demi satu tampilan layar secara berurut menurut urutannya. Tampilan yang dapat ditampilkan pada struktur jenis ini adalah satu halaman sebelumnya atau satu halaman sesudahnya, tidak dapat dua halaman sebelumnya atau dua halaman sesudahnya, pengguna akan melakukan navigasi secara berurutan, dalam *frame* atau *byte* informasi satu ke yang lainnya.

Sumber : Binanto (2010:269)

Gambar II.2 Struktur Navigasi *Linear*

b. Struktur Navigasi *Hirarki* (Bercabang)

Struktur dasar ini disebut juga struktur *linear* dengan percabangan karena pengguna melakukan navigasi disepanjang cabang pohon struktur yang terbentuk oleh logika isi.

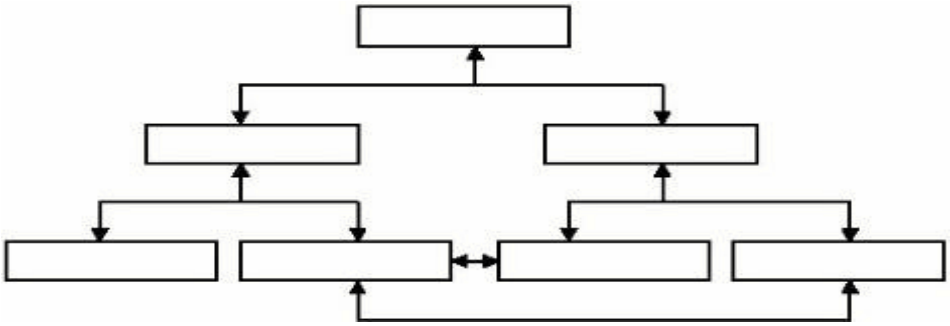


Sumber : Binanto (2010:269)

Gambar II.3 Struktur Navigasi *Hirarki*

c. Struktur Navigasi *Non Linear* (Tidak Berurut)

Struktur navigasi *non-linear* merupakan pengembangan dari struktur navigasi *linear*. Pada struktur ini diperkenankan membuat navigasi bercabang. Percabangan yang dibuat pada struktur *non-linear* ini berbeda dengan percabangan pada struktur *hirarki*, karena pada percabangan *non-linear* ini walaupun terdapat percabangan tetap tiap-tiap tampilan mempunyai kedudukan yang sama yaitu tidak ada *Master Page* dan *Slave Page*, pengguna akan melakukan navigasi dengan bebas melalui isi proyek dengan tidak terikat dengan jalur yang sudah ditentukan sebelumnya.

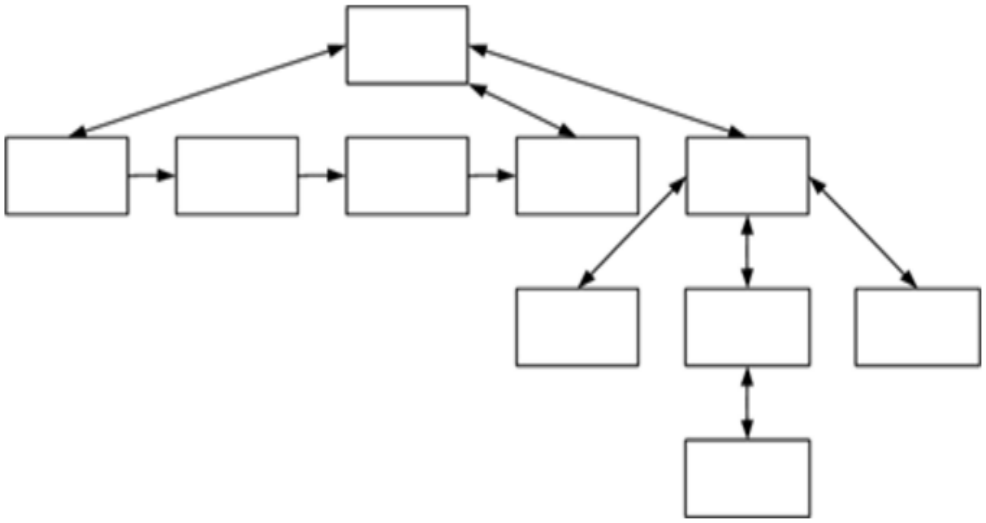


Sumber : Binanto (2010:270)

Gambar II.4 Struktur Navigasi *Non Linear*

d. Struktur Navigasi *Composite* (Campuran)

Struktur navigasi pengguna akan melakukan navigasi dengan bebas (secara *non-linier*), tetapi terkadang dibatasi presentasi *linear film* atau informasi penting dan pada data yang paling terorganisasi secara logis pada suatu *hirarki*.



Sumber : Binanto (2010:270)

Gambar II.5 Struktur Navigasi *Composite* (Campuran)

2. *ERD* (*Entity Relationship Diagram*)

Menurut Yasin (2012:276) "*Entity Relationship Diagram* (*ERD*) adalah suatu rancangan atau bentuk hubungan suatu kegiatan didalam sistem yang berkaitan langsung dan memiliki fungsi didalam proses tersebut."

ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang memiliki hubungan antar relasi. *ERD* untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan simbol. Pada dasarnya ada tiga simbol yang digunakan, yaitu:

1. *Entity*

Entity merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Simbol dari *entity* biasanya digambarkan dengan persegi panjang.

2. Atribut

Setiap entitas pasti mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dari atribut mempunyai sesuatu yang dapat mengidentifikasi isi elemen satu dengan yang lain. Gambar atribut diwakili oleh simbol *elips*.

Macam macam atribut :

a. *Simple attribute*

Simple attribute, yaitu atribut yang terdiri atas satu komponen tunggal dengan keberadaannya yang independen dan tidak dapat dibagi lagi ke komponen yang lebih kecil. *Simple attribute* dikenal juga dengan nama *atomic attribute*.

UNIVERSITAS BINA SARANA INFORMATIKA

b. *Composite attribute*

Composite attribute, yaitu atribut yang memiliki berbagai komponen dimana semua komponennya memiliki keberadaan yang independen.

c. *Single value attribute*

Single value attribute, yaitu sebuah atribut yang mempunyai nilai tunggal untuk setiap kejadian.

d. *Multi valued attribute*

Multi valued attribute, yaitu sebuah atribut yang mempunyai beberapa nilai untuk setiap kejaidan pada sebuah entitas.

e. *Derived attribute*

Derived attribute, yaitu atribut yang memiliki nilai yang dihasilkan dari satu atau beberapa atribut lainnya, dan tidak harus dari satu entitas.

3. Hubungan atau relasi

Relasi didefinisikan sebagai hubungan yang terjadi antar *entity*. Representasi diagram relasi adalah sebuah garis lurus yang menghubungkan dua buah *entity*. Jenis-jenis atau hubungan yang biasa terjadi antar satu *entity* dengen *entity* lain dalam sebuah basis data, meliputi:

a. *One to one* atau satu ke satu (1 : 1)

Hubungan relasi satu ke satu yaitu setiap entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan B. Contoh hubungan antara *entity* mahasiswa denga kartu mahasiswa. Seorang mahasiswa hanya boleh memiliki satu kartu mahasiswa , satu kartu mahasiswa hanya boleh dimiliki oleh satu mahasiswa.

b. *One to many* atau satu ke banyak (1 : *)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi ssetiap entitas pada entitas B dapat berhubungan dengan satu entitas pada himpunan entitas A. Contohnya hubungan yang terjadi antara *entity* konsumen dengan motor. Seorang konsumen dapat memiliki satu atau lebih motor. Sedangkan satu motor hanya dapat dimiliki oleh seorang konsumen.

c. *Many to many* atau banyak ke banyak (* : *)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak pada himpunan entitas B. Contohnya hubungan yang terjadi antar mahasiswa dengan mata kuliah. Satu mahasiswa dapat mengikuti lebih dari satu matakuliah dan satu matakuliah dapat diikuti lebih dari satu mahasiswa.

3. *LRS* (*Logical Record Structure*)

Membentuk SKEMA *DATABASE* atau *LRS* (*Logical Record Stucture*) berdasarkan diagram *ERD*.

a. Jika relasinya satu-ke-satu, maka *foreign key* diletakan pada salah satu dari 2 entitas yang ada atau menyatukan kedua entitas tersebut.

b. Jika relasinya satu-ke-banyak, maka *foreign key* diletakan pada entitas Many.

c. Jika relasinya diletakan banyak-ke-banyak, maka dibuat "*file* konektor" yang berisi 2 *foreign key* yang berasal dari dua entitas.

4. Pengujian *Web*

Menurut A.S Rosa dan Shalahuddin (2013:272) "Pengujian adalah salah satu set aktifitas yang direncanakan dan sistematis untuk menguji atau mengevaluasi

kebenaran yang diinginkan". Pengujian untuk validasi memiliki beberapa pendekatan sebagai berikut :

1. *Black-Box Testing* (pengujian kotak hitam)

   Yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

2. *White-Box Testing* (pengujian kotak putih)

   Yaitu pengujian perangkat lunak dari segi desain atau kode program apakah mampu menghasilkan fungsi-fungsi, masukan, dan keluaran yang sesuai dengan spesifikasi kebutuhan.

BINA SARANA INFORMATIKA
UNIVERSITAS