

BAB II

LANDASAN TEORI

2.1. Konsep Dasar *Web*

Menurut Alexander F.K Sibero (2013:1) “*Internet (Internasional Networking)* adalah kumpulan dari berbagai komputer diseluruh dunia yang terhubung satu sama lain, yang lazim disebut *worldwide network*”.Seperti diketahui bahwa program merupakan suatu rangkaian atau kumpulan instruksi-instruksi yang disusun secara logis dan sistematis. Untuk merancang sebuah program haruslah terlebih dahulu mendefinisikan masalah yang ada dan mempelajari program yang diberikan untuk mengetahui tujuan pengolahan, bentuk dan keadaannya serta output yang di hasilkan. Pada dasarnya program di rancang untuk memenuhi kebutuhan pemakai, dalam arti dapat meminimalkan kesalahan dan di harapkan mampu menyelesaikan permasalahan yang timbul akibat dari sistem berjalan yang telah ada.



Contoh teori konsep dasar program pembuatan aplikasi penjualan, antara lain:

A. *Website*

Menurut Ardhana (2012:3), “*Website* adalah sejumlah halaman web yang dimiliki topik saling terkait, terkadang disertai pula dengan berkas – berkas gambar, video, atau berkas lainnya”.

Dalam pembuatan aplikasi penjualan yang berbasis *web* ini dibutuhkan beberapa *tool web* , diantaranya sebagai berikut:

1. *Web Browser*

Menurut Alexander F.K Sibero (2013:12), “*Web Browser* adalah aplikasi perangkat lunak yang digunakan untuk mengambil dan menyajikan sumber informasi *web*. Sumberinformasi *web* diidentifikasi dengan *Uniform Resource Identifier* (URI) yang dapat terdiri dari halaman *web*, gambar, atau konten lainnya.”. Salah satu program *webbrowser*, seperti: *Mozilla Firefox*, *Internet Explorer (IE)* dan sebagainya. Fungsi *web browser* adalah menerjemahkan kode (*script*) HTML yang biasanya juga ditambahkan *JavaScript*, menjadi sebuah informasi yang sudah tertata dan menarik secara visual.



2. *Web Server*

Menurut Sadeli (2014:2), “*Web Server* adalah sebuah perangkat lunak server yang berfungsi menerima HTTP atau HTTPS dari klien yang dikenal dengan *web browser* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen HTML”.*Web server* berfungsi sebagai pusat kontrol dari pengolahan data *website* sehingga setiap instruksi yang diberikan oleh pemakai *internet* akan diolah dan selanjutnya dikembalikan lagi kepada pemakainya. Ada beberapa macam *web server* yang ada didunia, antara lain *web server* milik Windows, yaitu Apache, Tomcat, IIS (*Internet Information Service*), dan lain sebagainya. Saat ini *web server* yang banyak digunakan adalah Apache karena telah

banyak mendukung format *file* server tanpa perlu tambahan komponen aplikasi lagi. Hali ini berbeda dengan IIS milik Windows yang tidak dapat membaca *file* server dengan format PHP (IIS memerlukan komponen untuk menjalankan format *file* server PHP ini) dalam keadaan *deafult*.

3. WWW (*World Wide Web*)

MenurutArdhana (2012:31),“WWW adalah kependekan dari *World Wide Web*, merupakan suatu layanan *Internet* yang menggunakan konsep *hypertext* antar dokumen yang berkaitan”.

Pada WWW terdapat jurusan halaman *web*.Salah satu layanan aplikasi di internet adalah *World Wide Web* (WWW), pelayanan yang dikembangkan di *internet* menjadi layanan aplikasi yang paling yang populer digunakan pemakaian jaringan *internet*. WWW atau yang biasa disebut *web*, bekerja dengan teknologi yang biasa disebut *hypertext*, yang kemudian dikembangkan menjadi suatu protocol aplikasi yang disebut HTTP (*Hypertext Transfer Protocol*). WWW atau *web* merupakan fasilitas *hypertext* untuk menampilkan data berupa *text*, grafik, atau gambar, suara, animasi, dan sebagainya. Dengan kata lain WWW merupakan perpustakaan besar yang menyediakan berbagai buku dengan berbagai informasi.

B. Bahasa Pemograman

Bahasa pemograman adalah perangkat lunak atau *software* yang akan di gunakan dalam proses pembuatan program yang melalui beberapa tahapan-tahapan penyelesaian masalah.

Dalam pembuatan aplikasi penjualan berbasis *web* ini, penulis memakai bahasa pemrograman, antara lain :

1. PHP

Menurut Ardhana (2012:88), “*PHP* merupakan bahasa pemrograman berbasis *server-side* yang dapat melakukan *parsingscriptphp* yang menjadi *script web* sehingga dari sisi client menghasilkan tampilan yang menarik”. HTML merupakan *file* umum yang dijalankan oleh *browser*. Dengan demikian, *client* tidak dapat melihat kode PHP yang dibuat oleh *developerwebsite*, tetapi hanya bisa melihat kode HTML yang merupakan hasil olahan dari *Engine PHP*. Agar *webserver* dapat mengolah *file PHP* ini maka diperlukan aplikasi PHP yang kita sebut sebagai *Engine PHP* untuk ditanamkan bersama aplikasi *webserver* tersebut.

Kelebihan PHP dari bahasa pemrograman lain

1. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. *Web Server* yang mendukung PHP mudah didapat dari mulai Apache sampai XAMPP yang sudah berisi paket lengkap berikut dengan *database* Mysql, yang dapat dikonfigurasi dengan mudah. *Webserver* tersebut bersifat *open source* yang sapat digunakan diberbagai mesin (linux, unix, windows) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah sistem.



3. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena banyaknya referensi dari buku maupun tutorial yang terdapat di internet.

2. HTML

Menurut Utomo (2012:7), “*HTML* merupakan bahasa standar yang menjadi antarmuka dari *website* dan menjadi dasar ketika anda akan membangun sebuah *website*”.

HTML pada dasarnya merupakan himpunan bagian dari bahasa yang lebih luas jangkauannya yaitu *Standard Generalized Markup Language* (SGML), yang merupakan sistem pengkodean dan pemformatan untuk dokumen yang ditampilkan di layar komputer maupun di kertas.

World Wide Web pertama kali dikembangkan oleh fisikawan dan ilmuwan komputer inggris yang bernama Timothy Berners-Lee pada salah satu proyek yang diselenggarakan oleh Pusat Riset Nuklir Eropa (*European Organization for Nuclear Research* (CERN)) di Genave, Swiss. Berners-Lee menggabungkan beberapa ide yang ada saat itu ke suatu sistem tunggal yang memudahkan para fisikawan menggunakan data-data dan mengkomunikasikannya lewat *internet*. Yang juga penting, Barners-Lee menambahkan kemampuan multimedia pada dokumen-dokumen *web*. Berners-Lee mulai bekerja dengan *hypertext* pada awal tahun 1980-an dan prototipe-nya diimplementasikan oleh CERN pada tahun 1989. Selanjutnya ide-ide Berners-Lee secara cepat menyebar ke universitas-universitas di seluruh dunia



(dan akhirnya ke dunia komersial). Pada tahun 1989, Berners-Lee merumuskan suatu proposal tentang sebuah sistem *hypertext* yang memiliki 3 komponen sebagai berikut:

a. Antarmuka yang konsisten untuk semua jenis *platform*.

Antarmuka ini harus menyediakan akses yang dapat digunakan oleh berbagai jenis komputer.

b. Akses informasi yang bersifat universal. Setiap pengguna harus dapat mengakses setiap informasi yang berbeda.

c. Antarmuka yang menyediakan akses ke berbagai jenis dokumen dan protokol.

C. Basis Data

Menurut Utami (2012:43) "Basis Data adalah media untuk penyimpanan data agar dapat diakses dengan mudah dan cepat". Berdasarkan pendapat para ahli di atas, maka basis data adalah kumpulan data yang saling berhubungan secara logikal yang dapat digunakan untuk membantu dalam pengambilan keputusan pada sebuah organisasi atau perusahaan.



Berbagai bentuk aplikasi basis data atau *database* dan penjelasannya, antara lain:

1. Adobe Dreamweaver

Menurut Utami (2012:10) "Macromedia Dreamweaver adalah program *editor html* yang sangat handal dan banyak diminati oleh para *web desainer*". Adobe Dreamweaver merupakan salah satu *software* dari kelompok Adobe yang banyak digunakan untuk mendesain situs *web*. Adapun Adobe Dreamweaver itu sendiri adalah sebuah HTML *editor profesional* untuk mendesain secara visual dan mengelola situs atau halaman *web*. Adobe Dreamweaver memiliki performa yang

lebih baik dan memiliki tampilan yang memudahkan anda untuk membuat halaman *web*, maupun dalam jendela kode rumus. *Adobe Dreamweaver* didukung dengan cara pemakaian yang praktis dan standar, dan juga didukung untuk pengembangan penggunaan CSS, XML dan RSS, dan kemudahan-kemudahan lain yang diperlukan.

Dreamweaver merupakan *software* yang digunakan oleh *web deseiner* maupun *web programmer* dalam mengembangkan *web*. Hal ini disebabkan ruang kerja, fasilitas dan kemampuan *Dreamweaver* yang mampu meningkatkan produktivitas dan efektivitas dalam desain maupun membangun sebuah situs *web*.

2. XAMPP

Menurut Sadeli (2014:4), "Xampp adalah program yang berisi paket *Apache*, *MySQL* dan *phpMyAdmin*. Kita dapat memperoleh program tersebut dengan mendownload pada <http://www.apachefriends.org/en/xampp-windows.html> untuk mendapat versi terbaru". *MySQL* *Openbase* dan penterjemah bahasa yang ditulis dengan bahasa pemograman *PHP* dan *Perl*. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam GNU General Public Lisensi dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis.

Mengenal bagian XAMPP yang biasa digunakan pada umumnya:

- a. *htdocs* adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas *PHP*, *HTML* dan skrip lain.
- b. *phpMyAdmin* merupakan bagian untuk mengelola basis data *MySQL* yang ada di komputer. Untuk membukanya, buka *browser* lalu ketikkan alamat <http://localhost/phpMyAdmin> maka akan muncul halaman *phpMyAdmin*.



- c. Kontrol Panel yang berfungsi untuk mengelola layanan (*service*) XAMPP. Seperti menghentikan (*stop*) layanan, ataupun melalui (*start*).

3. MySQL

Menurut Sadeli (2014:10), "MySQL adalah database yang menghubungkan *script* PHP menggunakan perintah query dan *escapt character* yang sama dengan PHP. Mysql mempunyai tampilan *client* yang mempermudah anda dalam mengakses database". MySQL jangan disama-artikan dengan SQL (*Structure Query Language*) yang didefinisikan sebagai sintaks perintah-perintah tertentu dalam bahasa (program) yang digunakan untuk mengelola suatu *database*.

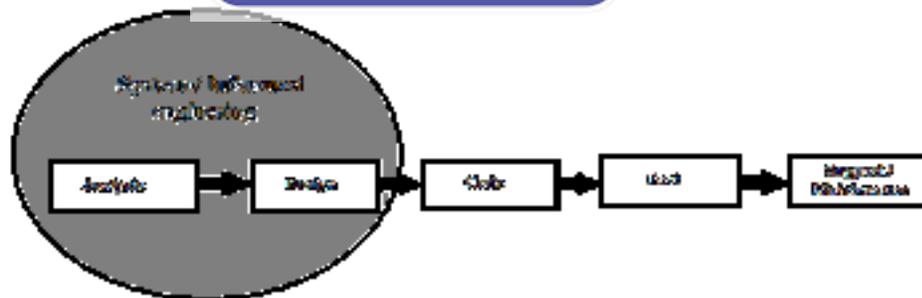
Kelebihan MySQL :

- a. Merupakan sebuah *database* yang mampu menyimpan data berkapasitas sangat besar hingga berukuran *gigabyte* sekalipun.
- b. Didukung oleh server *OOBC* yang artinya MySQL dapat diakses menggunakan aplikasi apa saja termasuk berupa visual seperti delphi maupun *Visual Basic*.
- c. *Database* yang menggunakan enkripsi *password*.
- d. Merupakan *server database multi user* artinya database ini dapat digunakan oleh banyak orang.
- e. Dapat menciptakan lebih dari 16 kunci per table dan satu kunci memungkinkan belasan *field*.



D. Model Pengembangan Perangkat Lunak

Menurut Pressman (2010:39), "Waterfall adalah model klasik yang bersifat sistematis, berurutan dalam membangun software". Pendekatan SDLC dengan berbagai pekerjaan di suatu tahap diselesaikan terlebih dulu sebelum pekerjaan dilanjutkan ke tahap berikutnya. Aliran SDLC sebagian besar tetap sama, berapa pun jumlah tahapannya. Siklus hidup pengembangan sistem (*System Development Life Cycle – SDLC*) adalah metode pengembangan sistem tradisional yang digunakan oleh berbagai perusahaan untuk proyek TI besar seperti infrastruktur TI. SDLC adalah kerangka kerja terstruktur yang terdiri atas berbagai proses berurutan untuk mengembangkan sistem informasi. Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian, dan tahap pendukung (*support*). Berikut adalah gambar model air terjun :



Sumber : Pressman (2010:39)

Gambar II.1 Waterfall Model

1. Analisis kebutuhan perangkat lunak (*Analysis*)

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

2. Desain (*Design*)

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini menranslasi kebutuhan perangkat lunak, dari tahap analisis kebutuhan representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak, dari tahap analisis kebutuhan representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.



3. Pembuatan kode program (*Code*)

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

4. Pengujian (*Test*)

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

5. Pendukung (*Support*) atau 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 analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

Dari kenyataan yang terjadi sangat jarang model air terjun dapat dilakukan sesuai alurnya karena sebab berikut.

- a. Perubahan spesifikasi perangkat lunak terjadi di tengah alur pengembangan.
- b. Sangat sulit bagi pelanggan untuk mendefinisikan semua spesifikasi di awal alur pengembangan. Pelanggan sering kali butuh contoh (*prototype*) untuk menjabarkan spesifikasi kebutuhan sistem lebih lanjut.
- c. Pelanggan tidak mungkin bersabar mengakomodasikan perubahan yang diperlukan di akhir alur pengembangan.

Model air terjun sangat cocok digunakan kebutuhan pelanggan sudah sangat dipahami dan kemungkinan terjadinya perubahan kebutuhan selama pengembangan perangkat lunak kecil. Hal positif dari model air terjun adalah struktur tahap pengembangan sistem jelas, dokumentasi dihasilkan di setiap tahap pengembangan, dan sebuah tahap dijalankan setelah tahap sebelumnya selesai dijalankan (tidak ada tumpang tindih pelaksanaan tahap).

2.2. Teori Pendukung

A. Struktur Navigasi

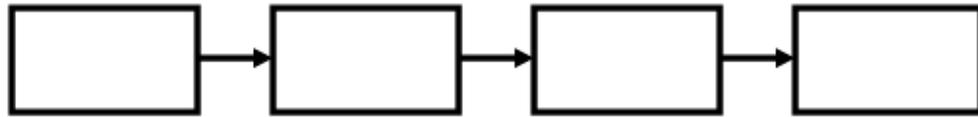
Menurut Suyanto (2006:62)“Struktur navigasi adalah gabungan dari struktur referensi informasi situs web dan mekanisme link yang mendukung pengunjung untuk melakukan penjelajahan situs”.Struktur navigasi suatu situs *web* sangat di pengaruhi oleh tujuan dari situs *web* yang akan dibuat.

Struktur navigasi adalah struktur atau alur dari sebuah program yang termasuk unsur penting didalam pembuatan aplikasi halaman *website*. Struktur navigasi ini merupakan perancangan hubungan dan rantai kerja dari beberapa lokasi yang berbeda dari halaman-halaman *website*. Selain itu struktur navigasi memberikan kemudahan dalam menganalisa keinteraktifan seluruh objek didalam halaman *website*. Struktur pembuatan halaman *website* terdiri dari *linier*, *non linier*, hirarki dan campuran.

1. *Linier* (Satu Alur)

Linear (satu alur) merupakan struktur yang hanya mempunyai satu rangkaian cerita yang berurut. Dengan kata lain struktur ini hanya dapat menampilkan satu demi satu tampilan layar secara berurut menurut urutannya. Tampilan yang dapat di tampilkan pada struktur jenis ini adalah satu halaman sebelumnya atau satu halaman sesudahnya dan tidak dapat menampilkan dua halaman sebelumnya atau dua halaman sesudahnya. Salah satu yang terpenting dari struktur ini adalah tidak diperkenankan terjadinya percabangan.



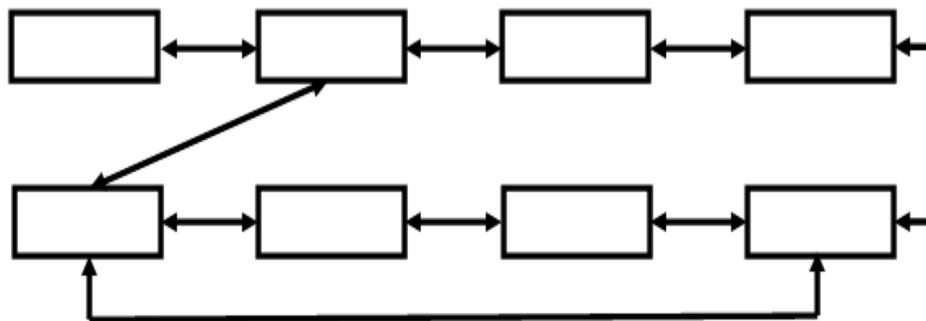


Sumber: Suyanto (2006:62)

Gambar II.2. Struktur Navigasi *Linier*

2. *Non Linier* (Tidak Berurut)

Struktur penjejakan *Non Linear* (tidak berurut) merupakan pengembangan dari struktur penjejakan *Linear*. Pada struktur ini diperkenankan membuat penjejakan bercabang. Pemakai bebas menelusuri *website* tanpa dibatasi oleh suatu rute dimana kontrol navigasi dapat mengakses ke semua halaman manapun. Percabangan yang dibuat pada struktur *Non Linear* ini berbeda dengan percabangan pada struktur *Hierarchi*, karena pada percabangan *Non Linear* ini walaupun terdapat percabangan, tetapi tiap – tiap tampilan mempunyai kedudukan yang sama tidak ada *Master Page* dan *Slave Page*.



Sumber: Suyanto (2006:65)

Gambar II.3. Struktur Navigasi *Non – Linier*

3. *Hierarchical* (Hirarki)

Struktur *Hierarchi* (bercabang) ini percabangan untuk menampilkan data berdasarkan kriteria tertentu. Tampilan pada menu pertama akan disebut sebagai *Master Page* (halaman utama kesatu), halaman utama ini akan mempunyai halaman percabangan yang dikatakan *Slave Page* (halaman pendukung). Jika salah satu halaman pendukung dipilih atau diaktifkan, maka tampilan tersebut akan bernama *Master Page* (halaman utama kedua), dan seterusnya. Yang terpenting dari struktur penjejakan ini tidak diperkenankan adanya tampilan secara *linear*.



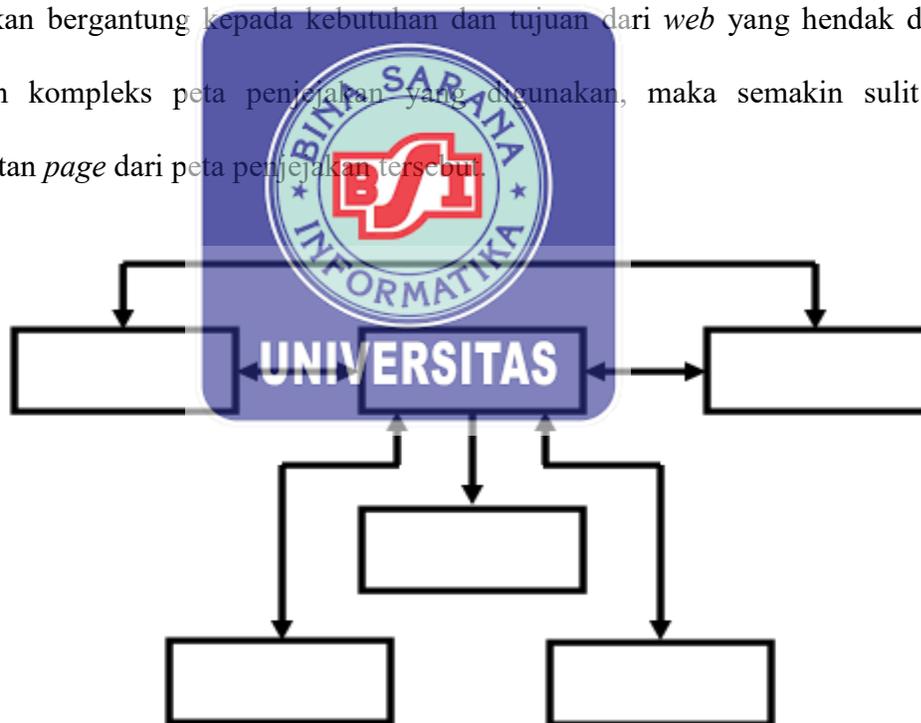
Sumber: Suyanto (2006:70)

Gambar II.4. Struktur Navigasi *Hierarchical*

4. Composite (Campuran)

Composite (campuran) atau disebut juga struktur penjejakan bebas merupakan gabungan dari ketiga struktur sebelumnya yaitu *Linear*, *Non Linear* dan *Hierarchi*. Jika suatu tampilan membutuhkan percabangan, maka dapat dibuat percabangan, dan bila dalam percabangan tersebut terdapat suatu tampilan yang sama kedudukannya maka dapat dibuat struktur *Linear* dalam percabangan tersebut.

Setiap struktur peta penjejakan seperti yang baru di bahas mempunyai fungsi dan tujuan tersendiri, tidak ada yang lebih baik atau lebih buruk. Penggunaan peta penjejakan bergantung kepada kebutuhan dan tujuan dari *web* yang hendak dibuat. Semakin kompleks peta penjejakan yang digunakan, maka semakin sulit pula pembuatan *page* dari peta penjejakan tersebut.



Sumber: Suyanto (2006:73)

Gambar II.5. Struktur Navigasi *Composite*

B. ERD (Enterprise Relationship Diagram)

1. Pengertian ERD (Enterprise Relationship Diagram)

Menurut Yasin (2012:276), "ERD (*Entity Relationship Diagram*) adalah suatu rancangan atau bentuk hubungan sesuatu kegiatan yang berkaitan langsung dan mempunyai fungsi didalam proses tersebut". ERD menggunakan simbol-simbol tertentu dalam menggambarkan elemen-elemen data.

2. Komponen – komponen ERD (Enterprise Relationship Diagram)

1. Entitas (*Entity*)

Entitas adalah Objek yang dapat dibedakan dengan objek lainnya.

2. Relasi (*relationship*)

Relasi adalah hubungan antara satu entitas dengan entitas yang lain.

3. Atribut (*attribute*)

Atribut adalah elemen yang dimiliki entitas yang akan disimpan datanya berfungsi mendeskripsikan karakteristik dari entitas tersebut.



3. Kardinalitas (*cardinality*)

Kardinalitas adalah banyaknya jumlah entitas yang saling berelasi. Kardinalita ada tiga macam, yaitu ;

a. *One to one*

Entitas pada himpunan A berhubungan paling banyak satu entitas pada himpunan entitas B. Begitu pula sebaliknya

b. *One to many*

Setiap entitas pada himpunan A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi setiap entitas pada entitas B dapat berhubungan dengan satu entitas pada himpunan entitas A.

c. *Many to many*

Setiap entitas pada himpunan A dapat berhubungan dengan banyak entitas.

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

Menurut Priyadi (2014:15), “*Logical Record Structure (LRS)* adalah representasi dari struktur *record-record* pada tabel – tabel yang terbentuk dari hasil relasi antar himpunan entitas”. Beberapa tipe *record* digambarkan oleh kotak empat persegi panjang dan dengan nama yang unik. Beda LRS dengan diagram ERD nama tipe tipe *record* berada diluar kotak *field tipe record* ditempatkan. LRS terdiri dari *link-link* diantaranya *tipe record*. Penggambaran LRS mulai dengan menggunakan model yang dimengerti. Metode yang lain dimulai dengan ERD dan langsung dikonversikan ke LRS.



C. **Pengujian Web**

Untuk mengetahui apakah sistem berjalan sesuai dengan kebutuhan *user*, maka diperlukan uji coba dengan metode *black box testing*.

Menurut Pressman (2010:495), “*Black Box Testing* berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineers* untuk memperoleh

kondisi input yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program”.Metode uji coba *blackbox* memfokuskan pada keperluan fungsional, karena itu uji coba *blackbox* memungkinkan pengembangan program untuk membuat himpunan kondisi *input* yang akan melatih syarat–syarat fungsional suatu program.

