

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Web

2.1.1. Internet

Menurut Irawan (2011:2) “*Internet* merupakan kependekan dari kata “*internetwork*”, yang berarti rangkaian komputer yang terhubung menjadi beberapa rangkaian jaringan”. Menurut Simarmata (2010:47) “*Internet* adalah kelompok atau kumpulan dari jutaan komputer”. Berdasarkan dari kedua kutipan diatas penulis merangkum *internet* adalah kumpulan beberapa komputer yang tekoneksi ke jaringan *internet*.

2.1.2. World Wide Web (WWW)

Menurut Sibero (2013:11) dalam bukunya “ *Word Wide Web*” atau yang dikenal juga istilah *web* adalah suatu sistem yang berkaitan dengan dokumen yang digunakan sebagai media untuk menampilkan teks, gambar, multimedia dan lainnya pada jaringan *internet*”. Sedangkan menurut Yuhefizar (2008:47) “ *World Wide Web* sering disingkat dengan *www* atau *web* adalah suatu metode untuk menampilkan informasi di *internet*, baik berupa text, gambar, suara maupun video yang interaktif dan mempunyai kelebihan untuk menghubungkan (*link*) suatu dokumen dengan dokumen lainnya (*hypertext*) yang dapat diakses melalui sebuah *browser*.”.

Dari pengertian diatas penulis merangkum bahwa *World Wide Web* (WWW) adalah kumpulan situs *web* yang berisi dokumen yang digunakan sebagai

media untuk menampilkan teks, gambar dan multimedia lainnya dengan menggunakan jaringan *internet*.

2.1.3. Website

Menurut Hidayat(2010:2) Menyatakan bahwa:

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, baik yang bersifat statis maupun dinamis yang membentuk suatu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

Menurut Yuhefizar dkk (2009:2) “*website* adalah keseluruhan halaman-halaman *web* yang terdapat dalam sebuah domain yang mengandung informasi.”.

Dari kutipan di atas penulis merangkum bahwa *Website* adalah kumpulan halaman *web* yang terdapat dalam sebuah *domain* yang mengandung informasi atau halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya.

2.1.4. Web Server

Menurut Supardi (2010:2) “*Web server* merupakan perangkat lunak yang mengelolah (mengatur) permintaan *user* dari *browser* dan hasilnya dikembalikan kembali ke *browser*”. Menurut Oktavian (2010:11) “*Web server* adalah aplikasi yang berguna untuk menerima permintaan informasi dari pengguna melalui *web browser*, dan mengirimkan kembali informasi yang diminta melalui HTTP (*Hypertext Transfer Protocol*)”.

Berdasarkan dua kutipan di atas penulis merangkum *web server* merupakan sebuah perangkat lunak yang berfungsi untuk mengelolah permintaan *user* dan mengrimkan kembali informasi yang diminta melalui HTTP (*Hypertext Transfer Protocol*).

2.1.5. *Web Browser*

Menurut Irawan (2011:3) “*Web Browser* adalah aplikasi yang digunakan pada jaringan internet untuk mengakses informasi, berbagi pakai, berkomunikasi, dan sebagainya. Menurut Fauziah (2008:103) “*Web Browser* merupakan program aplikasi yang digunakan untuk mengakses segala sesuatu yang ada di *internet* dan menjalankan berbagai fungsi berkaitan dengan aktivitas pengaksesan *internet*”.

Dari penjelasan dua kutipan diatas penulis merangkum *web browser* merupakan aplikasi yang digunakan pada jaringan *internet* untuk mengakses informasi, mengakses segala sesuatu yang ada di *internet* dan menjalankan berbagai fungsi berkaitan dengan aktifitas pengaksesan *internet*.

2.1.6. *Bahasa Pengembangan Web*

1. *Hypertext Markup Language (HTML)*

Badiyanto (2013:23) Menyatakan bahwa:

TML (*Hypertext Markup Language*) adalah bahasa standar pemrograman untuk membuat halaman *web* yang terdiri dari kode-kode tag tertentu, kemudian kode-kode tersebut diterjemahkan oleh *web browser* untuk menampilkan halaman *web* yang terdiri dari berbagai macam format tampilan seperti teks, grafik, animasi link, maupun audio-video.

Menurut Sibero (2013:19) “*HyperText Markup Language* Atau HTML adalah bahasa yang digunakan pada dokumen *web* sebagai bahasa untuk pertukaran dokumen *web*”. Dari kutipan diatas penulis mengambil sebuah kesimpulan bahwa HTML (*Hypertext Markup Language*) adalah Bahasa pemograman yang digunakan pada dokumen *web* dimana bahasa tersebut terdiri dari kode-kode dan tag tertentu agar dapat diterjemahkan oleh *web browser* untuk menampilkan sebuah *web*.

2. *PHP Hypertext Preprocessor (PHP)*

Menurut Badiyanto (2013:32) “PHP : *Hypertext Preprocessor* adalah bahasa skrip yang dapat ditanamkan untuk disisipkan dalam HTML/PHP banyak dipakai untuk membuat situs *web* dinamis”. Menurut Anhar (2010:3) “PHP adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis”.

Berdasarkan dua kutipan diatas penulis merangkum PHP adalah skrip yang digunakan sebagai pendukung HTML untuk membuat halaman *website* dinamis.

3. *Javascript*

Menurut sunyoto (2007:17) “Javascript adalah scripting yang populer di internet dan dapat bekerja di sebagian besar *browser* populer seperti internet Explore (IE), Mozilla FireFox, Netscape dan Opera”. Menurut Wahyono, (2009:97) menyatakan bahwa “Javascript adalah bahasa yang berbentuk sekumpulan scrip yang pada fungsinya berjalan pada suatu dokumen HTML.”

Berdasarkan pernyataan diatas penulis merangkum bahwa javascript adalah Bahasa pemrograman dapat bekerja di sebagian besar *browser* populer seperti internet Explore (IE), Mozilla FireFox, Netscape dan Opera yang pada fungsinya berjalan pada suatu dokumen HTML.

4. *Cascading Style Sheet (CSS)*

Menurut Larry (2012:5) ”*Cascading Style Sheets (CSS)* adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa markup.

Menurut Sya’ban (2010:37) ” *CSS (Cascading Style Sheet)* adalah suatu bahasa yang dikhususkan untuk mengatur gaya atau *layout* sebuah halaman *web*”. Dari penjabaran diatas penulis merangkum *CSS* adalah bahasa pemrograman yang digunakan untuk mengatur tampilan *website* atau *layout* sebuah halaman *web*.

5. *Jquery*

Menurut Sigit W(2011:1) “*Jquery* adalah library atau kumpulan kode JavaScript siap pakai”. Menurut Bakti(2015:59) “*JQuery* merupakan suatu *library javascript* yang memungkinkan untuk membuat program *web* pada suatu halaman *web* tanpa harus menanbah *event* atau pun *property* pada halaman *web* tersebut. Dari kutipan para ahli diatas *JQuery* merupakan *library* atau kumpulan kode *javascript* yang memungkinkan untuk membuat program *web* pada suatu halaman *web* tanpa harus menanbah *event* atau pun *property* pada halaman *web* tersebut”.

2.1.7. **Basis Data (*Database*)**

Menurut Rosa dan Shalahuddin (2014:43)“Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan”. Menurut Puspitosari (2011:9)“*Database* atau sering juga disebut basis data adalah sekumpulan informasi yang disimpan dalam komputer secara sistematis dan

merupakan sumber informasi yang dapat diperiksa menggunakan suatu program komputer”.

Berdasarkan kutipan diatas penulis merangkum basis data adalah sekumpulan informasi yang disimpan dalam komputer yang berfungsi untuk menyimpan dan mengolah informasi saat dibutuhkan.

1. SQL

Menurut Malik (2009:157) “*Structured Query Language* (SQL) adalah bahasa pemrograman khusus untuk mengirim pencarian data (*query*) ke *database*.” Sedangkan menurut Rosa dan Shalahuddin (2011:46) “SQL (*Structured Query Language*) adalah bahasa yang dipergunakan untuk mengelola RDBMS awalnya dikembangkan berdasarkan teori rekational dan kalkulus.

Menurut Nugroho dalam penggunaann perintah SQL dikategorikan menjadi tiga sub perintah, yaitu:

- a. *Data Definition Language* (DDL) adalah merupakan sub bahasa SQL yang digunakan untuk membangun kerangka *database*. Ada tiga perintah yang termasuk dalam DDL yaitu:

- 1) *Creat*

Perintah ini digunakan untuk membuat, termasuk di antaranya membuat *database* baru, tabel baru, *view* baru dan kolom.

- 2) *Alter*

Perintah ini digunakan untuk mengubah struktur tabel yang telah dibuat.

3) *Drop*

Perintah ini digunakan untuk menghapus *database* dan tabel.

- b. *Data Manipulation Language* (DML) merupakan sub bahasa *SQL* yang digunakan untuk memanipulasi data dalam *database* yang terbuat.

Perintah yang digunakan, di antaranya:

1) *Insert*

Perintah ini digunakan untuk menyisipkan atau memasukkan data baru ke dalam tabel.

2) *Select*

Perintah ini digunakan untuk mengambil data atau menampilkan data dari satu tabel atau beberapa tabel dalam relasi.

3) *Update*

Perintah ini digunakan untuk memperbarui data lama menjadi data terkini.

4) *Delete*

Perintah ini digunakan untuk menghapus data dari tabel.

- c. *Data Control Language* (DCL) merupakan sub bahasa *SQL* yang digunakan untuk melakukan pengontrolan data dan *server* databasenya.

Perintah DCL, diantaranya:

1) *Grand*

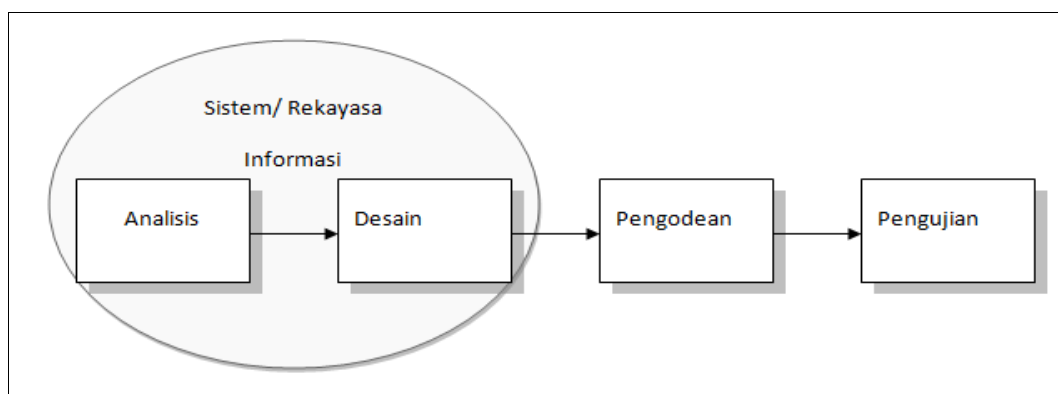
Perintah ini digunakan untuk memberikan hak/izin akses oleh *administrator* (pemilik utama) *server_* kepada *user* (pengguna biasa).

2) *Revoke*

Perintah ini memiliki kegunaan terbalik dengan *Grand* yaitu untuk menghilangkan atau mencabut hak akses yang telah diberikan kepada *user* oleh administrator.

2.1.8. Model Pengembangan Perangkat Lunak

Menurut Rosa A.S dan M.Shalahuddin (2015:28) 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, pengodean, pengujian, dan tahap pendukung (*support*). Berikut adalah gambar model air terjun.



Sumber : Rosa A.S dan Shalahuddin (2013:29)

Gambar II.1 Ilustrasi model *waterfall*.

1. Analisis Kebutuhan Perangkat Lunak

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 untuk didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang focus pada desain pembuatan program perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan 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

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

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

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

2.1.9. Aplikasi

Menurut Arifin Hasnul (2009:13)” Aplikasi merupakan perangkat lunak pendukung yang memiliki fungsi tertentu untuk membantu pengguna dalam menyelesaikan tugas tertentu.”.

Menurut S. Hakim Rahmad (2010:105)”Program aplikasi merupakan perangkat lunak (*software*) yang digunakan untuk tujuan tertentu, seperti mengelolah dokumen, mengatur windows 7, permainan (*game*), dan sebagainya”.

Dari pengertian diatas penulis merangkum aplikasi adalah kumpulan perintah program atau *software* yang digunakan untuk tujuan tertentu seperti mengelolah dokumen atau bermain *game*.

2.1.10. Data

Menurut Mulyadi (2007:301) “Data merupakan komponen fisik yang kedua dari bagian berwujud teknologi informasi”.

Sedangkan menurut Mills (1984) dalam Nuraida Nuraida (2008:25) “Data adalah fakta mentah, *observasi*, atau kejadian dalam bentuk angka atau *symbol* khusus.”

Dari pengertian diatas penulis merangkum data adalah komponen fisik yang kedua dari bagian berwujud teknologi informasi yang berupa fakta mentah, oservasi atau kejadian dalam bentuk angka atau *symbol* khusus.

2.1.11. Pengolahan Data

Menurut Ladjmanudin (2013:9) “Pengolahan data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan”. Sedangkan menurut Terry dalam buku Hutahaeen (2014:8) menyatakan “Pengolahan Data adalah serangkaian operasi atas informasi yang direncanakan guna mencapai tujuan atau hasil yang diinginkan”. Menurut pendapat diatas penulis merangkum pengolahan data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan menjadi serangkaian operasi atas informasi yang direncanakan guna mencapai tujuan atau hasil yang diinginkan.

2.2. Teori Pendukung

2.2.1. Enterprise Relationship Diagram (ERD)

1. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah pemodelan awal basis data yang paling banyak digunakan (Rosa. A.S dan Shalahuddin, 2014:50).

Berikut adalah simbol-simbol yang digunakan pada ERD dengan notasi Chen

Tabel II.1 Simbol simbol ERD

Simbol	Deskripsi
Entitas/<i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data yang harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
Nama atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).

Atribut multivalai/ <i>multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.
Asosiasi / <i>association</i> 	Penghubung antar relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimal keterhubungan antar entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan dari kardinalitas 1 ke N atau sering di sebut <i>one to many</i> menghubungkan entitas A dan entitas B.

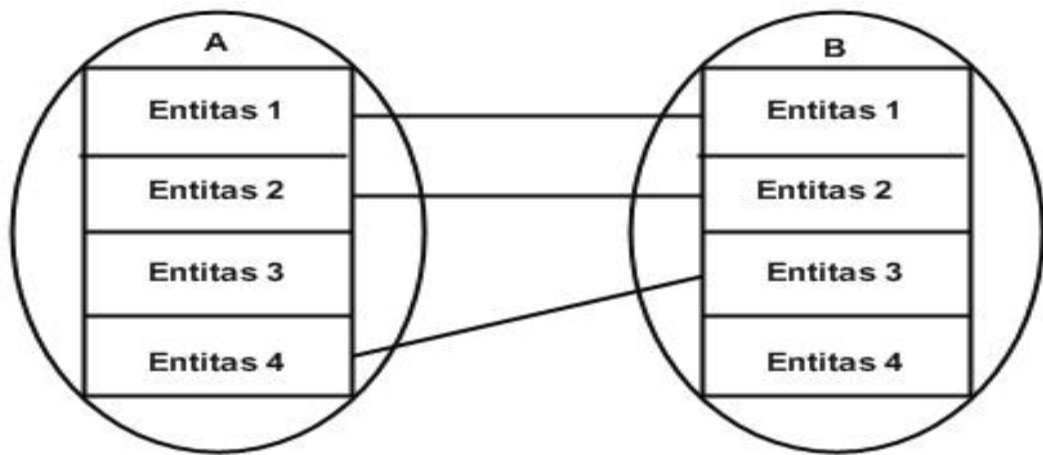
Sumber: Rosa. A. S dan Shalahuddin (2014:50)

2. Derajat *Relationship*

Menurut Fathansyah (2012:78), “Kardinalitas atau derajat *Relationship* menunjukan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain”. Adapun gambar Kardinalitas atau Derajat *Relationship* menurut Fathansyah (2012:79) adalah sebagai berikut:

a. Satu ke satu (*one to one*)

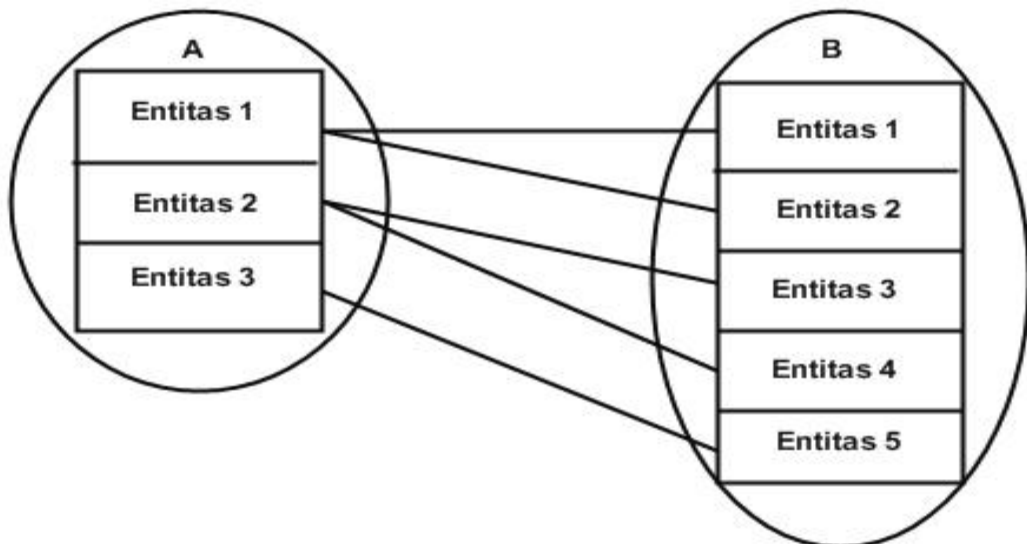
Yang berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu juga sebaliknya.



Sumber: Fathansyah (2012:79)

Gambar II.2 Derajat Relationship Satu ke Satu

b. Satu Ke Banyak (*one to many*)

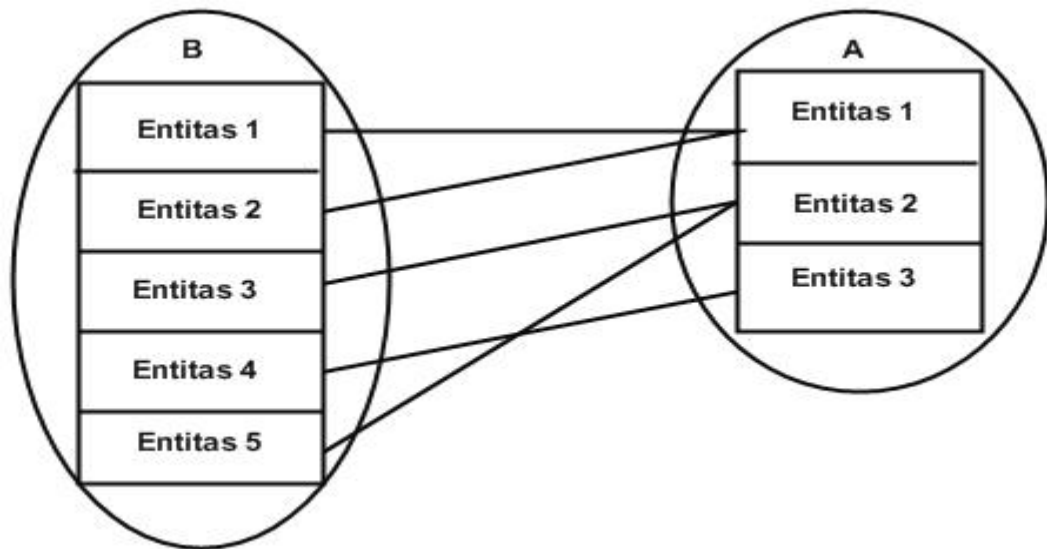


Sumber: Fathansyah (2012:80)

Gambar II.3 Derajat Relationship Satu ke Banyak

c. Banyak ke Satu (*many to one*)

Yang berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, tetapi tidak sebaliknya.

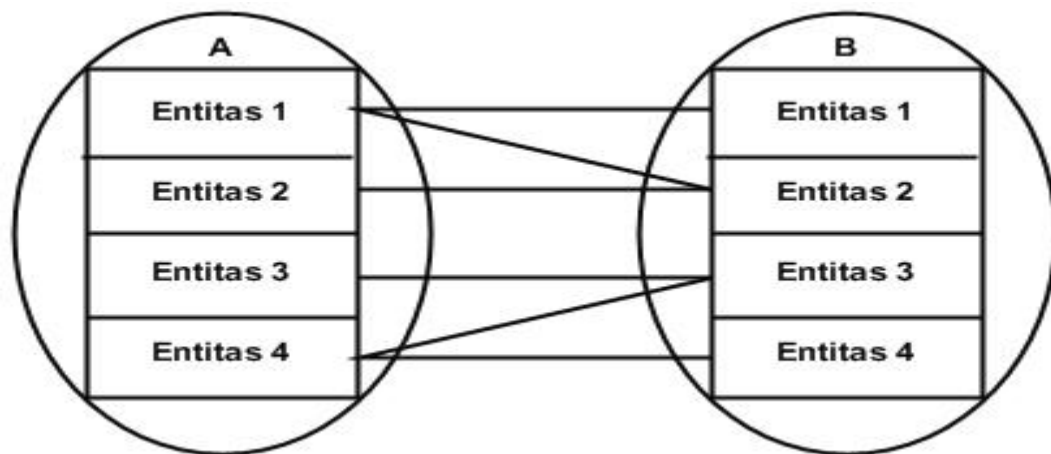


Sumber: Fathansyah (2012:80)

Gambar II.4 Derajat *Relationship* Banyak ke Satu

d. Banyak ke Banyak (*many to many*)

Yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas, dan demikian juga sebaliknya.



Sumber: Fathansyah (2012:80)

Gambar II.5 Derajat *Relationship* Banyak ke Banyak

Gambaran awal *website* akan ditampilkan secara keseluruhan mulai dari struktur navigasi, *Entity Relationship Diagram* dan *Logical Record Structure* serta pengujian dari *web*. Semua itu dibuat agar lebih memahami tentang *website* yang ada meliputi cara kerja berdasarkan struktur navigasi yang ada, gambaran mengenai *database*, serta pengujian yang sudah dilakukan agar *website* tersebut *valid* atau sesuai apa yang diinginkan. Berikut ini akan dijelaskan mengenai beberapa teori pendukung yang akan digunakan oleh penulis.

2.2.2. *Logical Record Structure (LRS)*

Menurut Ladjamudin (2013:159) menerangkan bahwa “*Logical Record Structure (LRS)* merupakan hasil transformasi *ERD* ke *LRS* yang memulai proses kardinalitas dan menghilangkan atribut-atribut yang saling berelasi”. Menurut Hasugian dan Shidiq (2012:608) memberikan batasan bahwa *LRS* adalah “sebuah model sistem yang digambarkan dengan sebuah diagram-ER akan mengikuti pola atau aturan permodelan tertentu dalam kaitanya dengan konvensi ke *LRS*”.

Menurut pendapat diatas dapat disimpulkan *Logical Record Structure (LRS)* merupakan hasil transformasi *ERD* ke *LRS* yang memulai proses

kardinalitas dan menghilangkan atribut-atribut yang saling berelasi dan memberikan batasan LRS adalah model system yang digambarkan dengan sebuah ER- yang mengikuti pola atau aturan permodelan tertentu dalam kaitanya dengan konvensi ke LRS.

2.2.3. Struktur Navigasi

Menurut Evi dan Malabay (2009:124) “Struktur *navigasi* merupakan rancangan hubungan dan rantai kerja dari beberapa *area* yang berbeda dan dapat membantu mengorganisasikan seluruh elemen *page*”. Menurut Binanto (2010:269) “Struktur navigasi adalah gabungan dari struktur referensi informasi situs *web* dan mekanisme *link* yang mendukung pengunjung untuk melakukan penjelajahan situs”.

Berdasarkan teori diatas dapat disimpulkan struktur navigasi merupakan alur dan suatu program yang merupakan rancangan hubungan (rantai kerja) dari beberapa area yang berbeda dan dapat membantu mengorganisasikan seluruh elemen *page* dari struktur referensi informasi situs *web* dan mekanisme *link* yang mendukung melakukan penjelajahan situs *web* tersebut.

Menurut Binanto (2010:269) terdapat 4 macam struktur navigasi antara lain:

1. Struktur Navigasi *Linier*

Struktur navigasi linier 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.

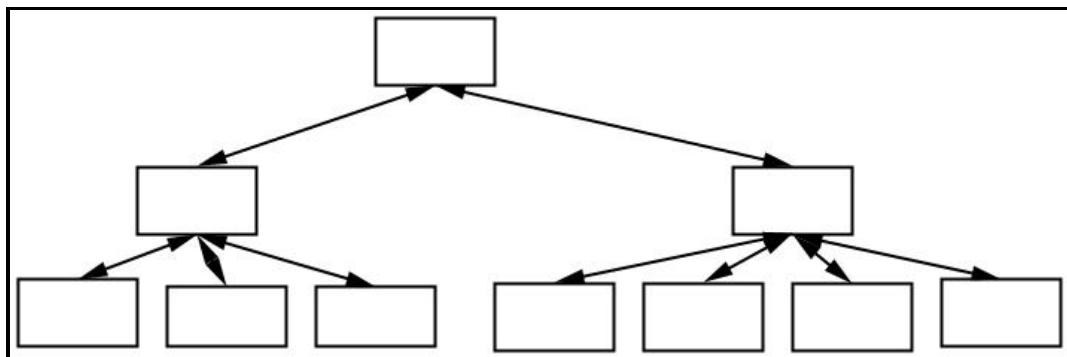


Sumber : Binanto (2010:269)

Gambar II.6 Contoh Struktur Navigasi *Linier*

2. Struktur Navigasi *Hierarkis*

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



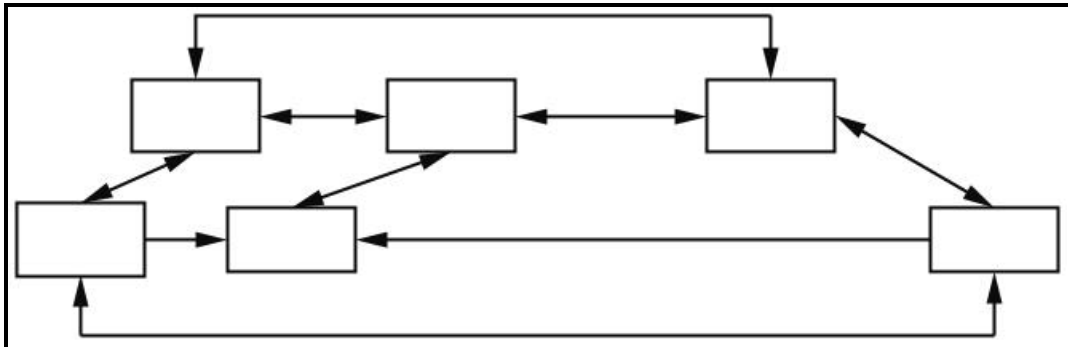
Sumber : Binanto (2010:269)

Gambar II.7 Contoh Struktur Navigasi *Hierarkis*

3. Struktur Navigasi *Non-Linier*

Non-Linier merupakan pengembangan dari struktur navigasi linier. Pada struktur ini diperkenankan membuat navigasi bercabang. Percabangan yang dibuat pada struktur non-linier 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.

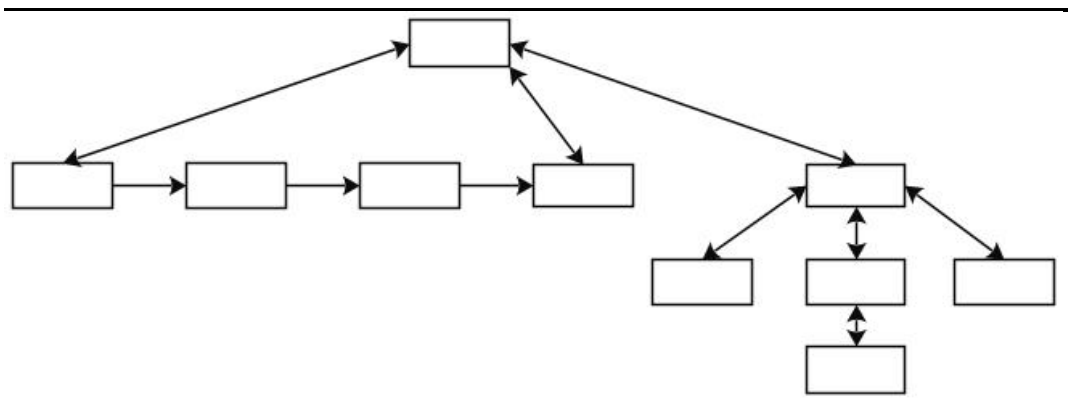


Sumber : Binanto (2010:270)

Gambar II.8 Contoh Struktur Navigasi *Non Linier*

4. Struktur Navigasi Campuran

Struktur navigasi campuran dapat melakukan navigasi dengan bebas (secara nonlinier),tetapi terkadang dibatasi presentasi linier film atau informasi penting dan pada data yang paling terorganisasi secara logis pada suatu hirarki.



Sumber : Binanto (2010:270)

Gambar II.9 Contoh Struktur Navigasi Campuran

2.2.4. Pengujian Web

Pengujian *web* dilakukan untuk memastikan bahwa *web* yang buat telah sesuai dengan rancangannya dan semua fungsi dapat digunakan sesuai dengan yang diharapkan tanpa ada kesalahan. Pengujian tersebut dimaksudkan untuk menguji sistem perangkat lunak secara keseluruhan dan diuji secara satu sistem tidak terpisah-pisah lagi (Rosa A.S dan M. Shalahuddin, 2013:275). Pengujian untuk *validasi* mempunyai beberapa pendekatan sebagai berikut:

Black-Box Testing (Penguji 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. Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan.

2.2.5. Aplikasi Pendukung

1. *Sublime text 3*

Menurut Supono dan Pratama (2013) “Sublime Text merupakan perangkat lunak text editor yang digunakan untuk membuat atau meng-edit suatu aplikasi.”. Menurut Firdaus dkk (2016:20) “Sublime Text adalah sebuah kode editor lintas form yang tersedia untuk windows, OS X, dan Linux. Sublime text bsa diunduh gratis untuk waktu yang tidak terbatas.”

Menurut Faridi (2015:3) menjelaskan bahwa “*Sublime Text 3* adalah editor berbasis *python*, sebuah teks editor yang elegan, kaya akan fitur, *cross*

platform, mudah dan simple yang cukup terkenal di kalangan *developer* (pengembang), penulis dan desainer”.

Dari hasil pengertian diatas, Sublime Text merupakan perangkat lunak text editor untuk meng-edit suatu aplikasi yang kaya akan fitur mudah dan simple cukup terkenal di kalangan developer dan desainer.

2. *Bootstrap*

Menurut Alatas (2013:2) “Bootstrap merupakan *framework* ataupun *tools* untuk membuat *aplikasi web* ataupun situs *web responsive* secara tepat, mudah dan gratis”. Menurut Riyanto (2014:18) bootstrap adalah kumpulan alat gratis untuk membuat *website* dan *aplikasi web*”.

Menurut pendapat diatas penulis merangkum Bootstrap merupakan kumpulan alat gratis yang sudah disediakan untuk membuat *website* dan merupakan *framework* atau *tools* untuk mempermudah pembuatan *website* menjadi lebih *responsive* dan dinamis.

3. *Framework*

Menurut Pratama (2010:10) “istilah *Framework* tentunya tidak asing. Terjemahan tugasnya adalah rangka, kerangka. Arti istilah tersebut dalam dunia pemograman kurang lebih adalah kumpulan kelas (*class*) dan fungsi (*function*, *method*) yang disusun secara sistematis berdasarkan kegunaan atau fungsionalitas tertentu untuk mempermudah pembuatan atau pengembangan suatu aplikasi.”. Sedangkan menurut Basuki (2015:13) “*Framework* adalah kumpulan potongan-potongan program (*kelas* dan *fungsi*) yang disusun dan diorganisasikan sedemikian rupa, sehingga dapat digunakan kembali untuk

membantu membuat aplikasi utuh tanpa harus membuat semua kodenya dari awal”.

Dari pengertian diatas penulis merangkum bahwa Dengan *framework* , anda tidak perlu menulis baris kode yang panjang untuk fungsi tertentu. Anda hanya perlu menggunakan fungsi yang sudah di tuliskan implementasinya dalam suatu *framework*.

4. *Codeigniter*

Menuru Sopono dan putratama (2016:109)”*Codeigniter* adalah aplikasi *open source* berupa *framework* dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP”.

Menurut Basuki (2016:12) “*Codeigniter* adalah *framework PHP* yang memakai sistem *Modal View Controller (MVC)*”.

Dari pernyataan dua kutipan diatas penulis merangkum *codeigniter* merupakan sebuah *framework PHP* dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP.

5. *PhpMyAdmin*

Menurut Puspitosari (2011:10) “*PhpMyAdmin* adalah program yang membantu pembuatan basis data (*database*) MySQL. *PhpMyadmin* sudah terinstal bersamaan saat kamu menginstal *AppServ*. dengan *PhpMyAdmin* kamu akan mudah dalam membuat dan mengelola *database*”. Menurut Rahman (2013:21) “*PHPMyAdmin* adalah sebuah *software* berbasis pemograman PHP yang dipergunakan untuk managemen *database*.”.

Dari kutipan diatas, dapat dirangkum bahwa phpMyAdmin adalah perangkat lunak bebas yang ditulis dalam bahasa pemograman php atau *software* berbasis pemograman PHP yang dipergunakan untuk managemen *database*.

6. XAMPP

Menurut Wicaksono (2008:7) dalam bukunya mengatakan bahwa “*Xampp* adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis PHP dan menggunakan pengolahan data *MySQL* dikomputer lokal. Menurut Aryanto (2016:4) “*Xampp* Merupakan sebuah aplikasi perangkat lunak pemrograman dan *database* seperti *Apache*, *HTTP Server*, *MySQL database*, bahasa pemograman PHP dan Perl.”

Dari pendapat diatas dapat disimpulkan bahwa *Xampp* merupakan *software* atau perangkat lunak yang berfungsi untuk menjalan *website* berbasis PHP dan menggunakan pengolahan data *MySQL* dikomputer local yang menghubungkan antara *Apache*, *HTTP Server*, *MySQL database*, bahasa pemograman PHP dan Perl.

7. MySQL

Menurut Zaki (2008:94) “*MySQL* adalah sebuah *software database*“. Menurut Sibero (2010:97) “*MySQL* salah satu *database server* open source yang populer.” Berdasarkan dua kutipan diatas penulis merangkum *MySQL* adalah salah satu *database server* open source yang populer atau sebuah *software database*.