

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Web

2.1.1. Internet

Menurut Sibero (2013:10) Internet (*Interconnected Network*) adalah “jaringan komputer yang menghubungkan antar jaringan secara global, internet juga dapat disebut jaringan dalam suatu jaringan yang luas”.

Menurut Simarmata (2010:47) “*internet* adalah kelompok atau kumpulan dari jutaan komputer”.

Supardi (2009:1) “*internet* yang merupakan akronim dari *Interconnection Networking* adalah jaringan komputer dengan skala dunia”.

Berdasarkan kutipan paragraf ahli diatas, internet merupakan sebuah teknologi sistem komunikasi atau media penghubung para pengguna untuk mengakses apa saja yang terdapat dalam jaringan di seluruh dunia.

2.1.2. World Wide Web (WWW)

Menurut Dipraja (2013:9) web adalah “fasilitas hypertext yang mampu menampilkan data berupa text, gambar, suara, animasi, dan multimedia”. Dan menurut Simarmata (2010:47) web adalah “sebuah sistem informasi dengan informasi yang disajikan dalam bentuk teks, gambar, suara, dan lain-lain yang tersimpan dalam sebuah server web internet yang disajikan dalam bentuk hiperteks”.

Sedangkan menurut Sibero (2013:11) “*world wide web* adalah suatu sistem yang berkaitan dengan dokumen yang digunakan sebagai media untuk menampilkan teks, gambar, multimedia dan lainnya pada jaringan *internet*”.

Berdasarkan kumpulan pendapat dari para ahli diatas, bahwa *world wide web* (WWW) merupakan kumpulan dari *website* mengizinkan para pengguna untuk mengakses halaman *web* yang terdapat dalam *domain* yang mengandung informasi dalam skala internasional

2.1.3. Website

Website merupakan kumpulan halaman *web* disimpan dalam situs *web* (*website*) yang merupakan server komputer yang mendukung HTTP (*Hypertext Transfer Protocol*). Halaman-halaman tersebut diakses dan dibaca melalui *browser* (Hall, 2007:185).

Website adalah kumpulan halaman *web* yang disajikan dalam bentuk teks, gambar, suara, dan lain-lain yang tersimpan dalam sebuah server *web internet* yang disajikan dalam bentuk hiperteks (Simarmata, 2010:47).

Menurut Menurut Sibero (2013:11) “*website* adalah suatu sistem yang berkaitan dengan dokumen digunakan sebagai media untuk menampilkan teks, gambar, multimedia dan lainnya pada jaringan *internet*”.

Menurut Hidayat (2010:2) mengemukakan bahwa:

Website atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, atau gabungan semuanya, baik yang bersifat dinamis atau statis yang membentuk suatu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

Berdasarkan kumpulan pendapat dari para ahli diatas, *website* adalah keseluruhan halaman-halaman *web* yang terdapat dalam sebuah domain yang

mengandung informasi atau kumpulan halaman yang digunakan untuk menampilkan informasi teks, gambar, animasi, suara dan gabungan dari semua baik yang bersifat statis maupun dinamis

2.1.4. Web Server

Menurut Saputra (2010:2) “*Webserver* adalah server *internet* yang digunakan sebagai *koneksi* dan transfer data (HTML,asp, aspx, php, js, dan lain sebagainya)”.

Menurut Sunyoto (2007:1) “*web server* merupakan tempat melayani *request*”.

Supono dan Putratama (2016:6) menyatakan bahwa “Paket *web server* adalah sebuah perangkat lunak *server* yang berfungsi untuk menerima permintaan dalam bentuk situs *web* melalui HTTP atau HTTPS dari klien itu, yang dikenal sebagai *browser web* dan mengirimkan kembali (reaksi) hasil dalam bentuk situs yang biasanya merupakan dokumen HTML”.

Berdasarkan pengertian diatas, bahwa *web server* adalah bagian perangkat lunak yang mengandung bermacam *protocol web* untuk dapat menjalankan perintah dari *client*, *server internet* yang digunakan sebagai *koneksi* dan transfer data (HTML,asp, aspx, php, js, dan lain) selain itu, *web server* adalah aplikasi terpending dari *world wide web (www)*.

2.1.5. Web Browser

Web browser digunakan untuk menampilkan hasil kepada *user* atas *request* yang telah dilakukan pada *web server* (Sunyoto, 2007:1).

Web dapat diakses oleh perangkat lunak *client web* yang disebut *browser*. *Browser* membaca halaman-halaman *web* yang tersimpan dalam server *web* melalui protokol yang disebut HTTP (Simarmata, 2010:50).

Menurut Solichin (2016:9) mengemukakan bahwa “peramban *web* atau lebih dikenal dengan *web browser* merupakan perangkat lunak yang berfungsi untuk menerima dan menyajikan sumber informasi di *internet*”.

Berdasarkan pengertian diatas, bahwa *web browser* adalah aplikasi yang digunakan untuk menampilkan halaman halaman *web* untuk proses mengelola informasi, pengambilan dan penyajian informasi pada *website*.

2.1.6. Bahasa Pengembangan Web (*Web Development Languages*)

Dalam penulisan tugas akhir ini penulis menggunakan bahasa pemrograman yang dikembangkan untuk membangun sebuah aplikasi yang dinamis. Adapun bahasa pemrograman yang digunakan sebagai berikut :

1. *Hypertext Markup Language (HTML)*

Menurut Solichin (2016:10) mengemukakan bahwa “HTML merupakan bahasa pemrograman *web* yang memberitahukan peramban *web* (*web browser*) bagaimana menyusun dan menyajikan konten di halaman *web*”.

Simarmata (2010:52) “HTML adalah bahasa *markup* untuk menyebarkan informasi pada *web*”.

Menurut Prasetio (2010:4) mengemukakan bahwa HTML merupakan “bahasa pemrograman yang digunakan untuk mendesain sebuah halaman *web*”.

Berdasarkan teori diatas HTML (*Hypertext Markup Language*) adalah bahasa standar pemrograman *web* yang digunakan untuk menampilkan konten diatas dihalaman *web*.

Berdasarkan teori diatas HTML (*Hypertext Markup Language*) adalah bahasastandar pemrograman *web* yang digunakan untuk menampilkan konten diatas dihalaman *web*.

2. *Hypertext Preprocessor (PHP)*

Menurut Supono dan Putratama (2016:3) mengemukakan bahwa” PHP (*PHP: Hypertext Preprocessor*) adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang berbasis *server-side* yang dapat ditambahkan ke dalam HTML”.

Menurut Solichin (2016:11) mengemukakan bahwa “PHP merupakan salah satu bahasa pemrograman berbasis *web* yang ditulis oleh dan untuk pengembang *web*”.

Berdasarkan teori diatas PHP (*Hypertext Preprocessor*) adalah penerjemahan baris kode mesin yang di komputer secara langsung untuk menempelkan kode di dalam HTML.

3. *CodeIgniter*

Menurut Supono dan Putratama (2016:109) “*CodeIgniter* adalah aplikasi *open source* berupa *framework* dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP”.

CodeIgniter adalah aplikasi *open source* yang berupa *framework* dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP (Pratama, 2010:11).

Berdasarkan teori diatas, *CodeIgniter* (CI) adalah salah satu *framework* yang memiliki kerangka untuk membangun *website* dinamis dengan menggunakan PHP.

4. *Cascading Style Sheet* (CSS)

Menurut Prasetio (2014:252) menyatakan bahwa “CSS adalah suatu teknologi yang digunakan untuk memperindah tampilan halaman *website* (situs)”. CSS atau *cascading style sheet* bahasa pemrograman yang berfungsi untuk mempercantik tampilan *web* (Solichin, 2016:10).

Tohirudin (2011:3) “CSS adalah bahasa pemrograman *web* untuk mengendalikan komponen pada sebuah *web* sehingga lebih terstruktur”.

Berdasarkan teori diatas, CSS (*cascading style sheet*) adalah bahasa pemrograman yang digunakan untuk mendesain tampilan *web* agar terstruktur dan seragam.

5. *Twitter Bootstrap*

Menurut Setiawan (2013:2) “*twitter bootstrap* adalah sekumpulan perangkat gratis yang digunakan untuk membuat *website* dan aplikasi *web*”.

Menurut Masuara dkk, (2015:69) “*twitter bootstrap* adalah *framework* ataupun tool untuk membuat aplikasi web ataupun situs web *responsive* secara cepat, mudah dan gratis”.

Berdasarkan teori-teori diatas, *bootstrap* adalah sebuah *CSS framework* yang digunakan untuk membuat *website* atau aplikasi *web reponsive* secara cepat dan mudah.

6. *Javascript*

Menurut Sunyoto (2007:17) “JavaScript adalah bahasa *scripting* yang populer di internet dan dapat bekerja di sebagian besar *browser populer* seperti Internet Explorer (IE), Mozilla Firefox, Netscape, dan Opera”.

Wahyono (2009:97) menyatakan bahwa:
Javascript adalah bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML. Bahasa ini adalah bahasa pemrograman untuk memberikan kemampuan tambahan terhadap bahasa HTML dengan mengizinkan pengekseskuan perintah di sisi *user*, yang artinya di sisi *browser* bukan di sisi server *web*.

Solichin (2016:11) “fungsi utama dari javascript adalah untuk menambah fungsionalitas dan kenyamanan halaman *web*”.

Dari penjelasan teori diatas,*Javascript* merupakan salah satu bahasa pemograman komputer yang dinamis. *Javascript* merupakan tampilan pendukung yang ada pada *website*, sebuah bahasa *script* yang populer di *internet*, salah satunya untuk memberikan pesan menarik untuk *user*.

7. *JQuery*

Menurut Wicaksono (2011:1) mengemukakan bahwa “JQuery adalah *library* atau kumpulan kode JavaScript siap pakai”.

Kun (2010:1) “JQuery adalah salah satu *library* JavaScript. Dengan JQuery, kita dapat melakukan banyak hal yang tidak bisa dilakukan oleh HTML maupun CSS”.

Berdasarkan teori diatas, jquery adalah suatu pustaka yang di kembangkan untuk memudahkan penulisan kode JavaScript”.

8. *Framework*

Menurut Simarmata (2010:82) “*framework* adalah sistem perangkat lunak yang dapat digunakan kembali dengan fungsi umum yang telah diterapkan”.

Wahana Komputer (2010:133) “*framework* adalah komponen *software* yang bisa ditambahkan atau bisa juga sudah termasuk fitur dari *Microsoft Windows*”.

Dari definisi diatas bahwa *Framework* merupakan sebuah kerangka kerja. *Framework* juga dapat diartikan sebagai kumpulan *script* (terutama *class* dan *function*) yang dapat membantu *developer/programan* dalam menangani berbagai masalah masalah dalam pembrograman seperti koneksi ke *database*, pemanggilan variabel, *file* sehingga developer lebih fokus dan lebih cepat membangun aplikasi.

2.1.7. Normalisasi

Menurut Lubis (2016:55) “normalisasi merupakan tahapan pengelompokan komponen data yang menjadi tabel yang menunjukkan entitas pada suatu relasi”.

Menurut Fatta (2007:129) menyatakan bahwa “normalisasi adalah teknik yang digunakan untuk memvalidasi model data”. Serangkaian aturan yang diberlakukan pada data model logis untu meningkatkan pengaturannya”.

Berdasarkan kutipan dari para ahli di atas, bahwa normalisasi merupakan suatu teknik yang digunakan untuk menciptakan struktur-struktur entitas yang dapat mengurangi redundansi data dan meningkatkan stabilitas *database* dengan serangkaian aturan pada data model logis.

Normalisasi memiliki aturan-aturan tertentu dan memiliki bentuk formal.

Adapun bentuk formal (Fatta, 2007:130) yaitu:

1. Bentuk normal pertama (1NF)

Bentuk normal pertama dikenal dengan *unnormalized form* atau bentuk yang merupakan kumpulan data yang akan direkam, tidak ada keharusan dalam mengikuti format tertentu, dapat saja tidak lengkap dan terduplikasi.

2. Bentuk normal kedua (2NF)

Bentuk normal kedua ini adalah bentuk selanjutnya dari tahapan 1NF dimana sudah dilakukan penyesuaian kunci ketergantungan, dan memiliki ketergantungan, namun belum di *filter* lebih lanjut sehingga masih mengakibatkan redundansi data.

3. Bentuk normal ketiga (3NF)

Bentuk ketiga ini sebenarnya telah dapat menghasilkan suatu rancangan *database* yang baik yang sudah dilakukan normalisasi.

2.1.8. Administrasi

Administrasi merupakan suatu fungsi tertentu untuk mengendalikan, menggerakkan, mengembangkan, dan mengarahkan suatu organisasi, yang dijalankan oleh administrator dibantu oleh tim bawahannya (Hidayat, 2007:1).

Menurut Haryadi (2009:1) “administrasi dalam arti luas adalah kegiatan kerja sama yang dilakukan oleh sekelompok orang berdasarkan pembagian kerja sebagaimana ditentukan dalam struktur dengan mendayagunakan sumber daya untuk mencapai tujuan secara efektif dan efisien”.

Berdasarkan kutipan diatas, maka administrasi merupakan suatu fungsi tertentu untuk mengendalikan, menggerakkan, mengembangkan, dan mengarahkan suatu organisasi saling kerja sama berdasarkan pembagian kerja ditentukan dalam struktur dengan mendayagunakan sumber daya untuk mencapai tujuan secara efektif dan efisien.

2.1.9. Rekam Medis

Menurut Hanafiah dan Amir (2007:258) “rekam medis kesehatan adalah rekam dalam bentuk tulisan atau gambaran aktivitas pelayanan yang diberikan oleh pemberi layanan medis/kesehatan kepada seorang pasien.

Menurut Ismainar (2015:89) “berkas rekam medis adalah milik sarana pelayanan kesehatan, namun isi rekam medis adalah milik pasien”.

Berdasarkan kutipan diatas, maka rekam medis adalah berkas yang berisikan catatan, dan dokumen tentang identitas pasien, pemeriksaan, pengobatan, tindakan dan pelayanan lain pada pasien pada sarana pelayanan kesehatan.

2.1.10. Basis Data

Menurut Lubis (2016:2) basis data merupakan ”tempat berkumpulnya data yang saling berhubungan dalam suatu wadah (organisasi/perusahaan) bertujuan agar dapat mempermudah dan mempercepat untuk pemanggilan atau pemanfaatan kembali data tersebut”.

Zaki dan SmithDev Community (2008:94) “basis data merupakan sebuah tempat penyimpanan untuk menyimpan data yang jenisnya beraneka ragam”.

Berdasarkan teori diatas, basis data adalah media tempat penyimpanan data mudah dan cepat diakses.

Menurut Kadir dan Triwahyuni (2013:339) “Basis data adalah suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi”.

Menurut Rosa dan Shalahuddin (2015:43) “Basis data adalah media untuk penyimpanan data agar dapat diakses dengan mudah dan cepat”.

Menurut Rosa dan Shalahuddin (2014:46) “SQL (*structured query language*) adalah bahasa yang digunakan untuk mengelola data pada RDBMS”.

Berdasarkan teori diatas, basis data adalah media tempat penyimpanan data mudah dan cepat diakses. Dalam mengelola basis data menggunakan DBMS, dan juga beberapa bahasa perintah standar yang biasa digunakan untuk mengakses dan memanipulasi objek pada basis data yang bisa disebut dengan *Structure Query Language* (SQL). Didalam menggunakannya perintah *SQL* Menurut Priyadi (2010:82) dibagi menjadi tiga, yaitu:

1. *Data Definition Language* (DDL)

DDL merupakan perintah SQL yang digunakan untuk melakukan definisi awal suatu basis data dan tabel pada konsep RDBMS. Adapun perintah yang termasuk ke dalam DDL adalah:

- a. *Create* adalah perintah yang digunakan untuk membuat objek pada basis data.
- b. *Alter* adalah perintah yang digunakan untuk merubah objek pada basis data.

- c. *Drop* adalah perintah yang digunakan untuk menghapus objek pada basis data.

2. *Data Manipulation Language (DML)*

DML merupakan perintah SQL yang digunakan untuk melakukan pengolahan *record* atau manipulasi *record* pada tabel dalam suatu basis data. Berikut perintah yang termasuk ke dalam DML:

- a. *Insert* adalah perintah yang digunakan untuk menambah atau memasukkan data pada basis data.
- b. *Update* adalah perintah yang digunakan untuk merubah atau memperbaharui data pada basis data.
- c. *Select* adalah perintah yang digunakan untuk menampilkan data dari basis data.
- d. *Delete* adalah perintah yang digunakan untuk menghapus data dari basisdata.

3. *Data Control Language (DCL)*

DCL merupakan perintah SQL yang digunakan untuk melakukan pengaturan hak akses suatu objek data para pengguna dalam basis data. Adapun perintah dari DCL adalah sebagai berikut:

- a. *Grant* adalah perintah yang digunakan untuk memberikan hak akses kepada pengguna untuk mengakses basis data tertentu.
- b. *Revoke* Perintah yang digunakan untuk mencabut hak akses pengguna pada tabel dalam basis data tertentu.
- c. *Data Integrity* adalah perintah yang digunakan untuk menjaga konsistensi dan kebenaran data yang disimpan.

- d. *Auxiliary* adalah perintah yang melengkapi SQL, seperti unload, load dan *rename column*.

2.1.11. Model Pengembangan Perangkat Lunak

Model SDLC air terjun (waterfall) sering juga disebut model sekuensial linier (sequential linier) atau alur hidup klasik (classic life cycle). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisa, desain, pengkodean, pengujian, dan tahap pendukung (support). Berikut adalah model air terjun: (Rosa dan Shalahuddin, 2013:29):

1. Analisa Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara insentif untuk memesifikasi kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. (Rosa dan Shalahuddin, 2013:29).

2. Desain

Desain perangkat lunak adalah proses multi langkah yang focus pada pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur perangkat lunak. (Rosa dan Shalahuddin, 2013:29).

3. Pembuatan Kode Program

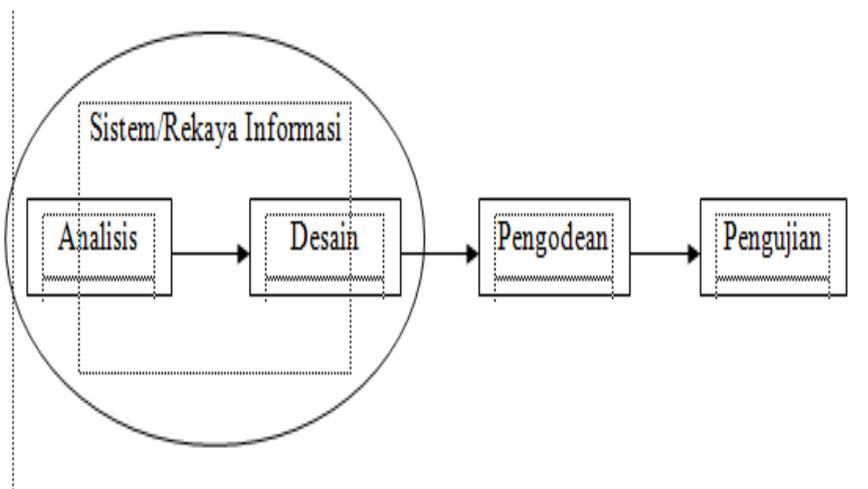
Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program computer harus sesuai dengan desain yang telah dibuat pada tahapan desain. (Rosa dan Shalahuddin, 2013:29).

4. Pengujian

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 dengan yang diinginkan. (Rosa dan Shalahuddin, 2013:30).

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.



Sumber: Rosa dan Shalahuddin (2013:30)

Gambar II.1 Ilustrasi Model *Waterfall*

2.2. Teori Pendukung

Dalam pembuatan tugas akhir ini penulis menggunakan beberapa teori pendukung. Adapun teori pendukung yang digunakan, yaitu :

2.2.1. Software Pendukung

Dalam pembuatan tugas akhir ini penulis menggunakan beberapa software pendukung guna menunjang keberhasilan tugas akhir ini. Adapun software pendukung yang di gunakan adalah sebagai berikut,yaitu :

1. *Sublime Text 3*

Menurut Faridi (2015:3) “*sublime text* adalah teks *editor* berbasis *python* sebuah teks *editor* yang elegan, kaya fitur, *cross platform*, mudah dan *simple* yang cukup terkenal di kalangan *developer* (pengembang), penulis dan *desainer*”.

Sedangkan menurut Putratama (2016:14) *sublime text* merupakan “perangkat lunak text editor yang digunakan untuk membuat atau meng-edit suatu aplikasi”.

Berdasarkan teori-teori diatas, *sublime text* adalah merupakan sebuah teks *editor* yang sangat elegan, kaya fitur, cangkih untuk *coding*, *markup* mudah dan *simple*.

2. *PhpMyAdmin*

Menurut Sibero (2013:376) “*PhpMyAdmin* adalah aplikasi *web* yang dibuat oleh *phpmyadmin net*”.

Sedangkan menurut Zaki dan Smitdev Community (2008:97) “*PHPMyadmin* adalah *MySQL* client yang berupa aplikasi *web* dan umumnya tersedia di *server PHP* seperti *XAMPP* ataupun server komersil lainnya”.

Berdasarkan teori diatas, *PhpMyAdmin* adalah sebuah program aplikasi yang memudahkan pengguna dalam mengakses *database*.

3. Xampp Server

Menurut Madcoms (2009:1) “*Xampp* adalah salah satu paket *software web server* yang terdiri *Apache*, *MySQL*, *PHP* dan *phpMyadmin*”.

Menurut Sibero (2013:370) “*Xampp* adalah yang berjalan pada *platform* yang berbeda”.

Berdasarkan teori diatas, *Xampp* adalah merupakan salah satu *software* yang membantu dalam mengakses *database*.

2.2.2. Struktur Navigasi

Ada beberapa desain struktur navigasi yang biasa digunakan dalam merancang sebuah program, menurut Vaughan (2006:367-368) yaitu sebagai berikut :

1. Linear

Pengguna akan melakukan melkukan navigasi secara berurutan, dari frame atau byte informasi yang satu ke yang lainnya.

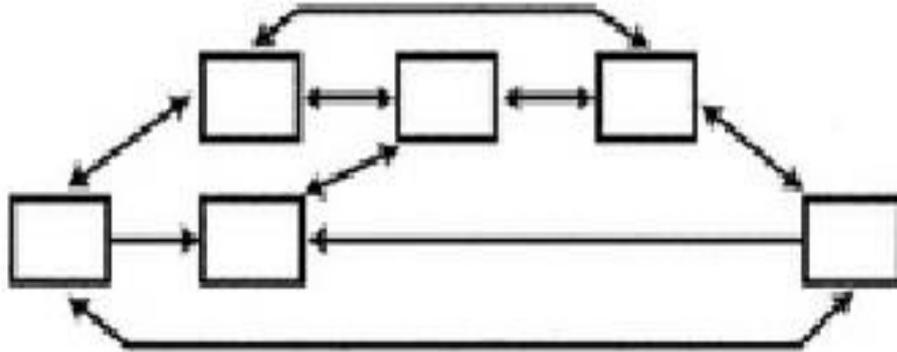


Sumber: Vaughan (2006:367-368)

Gambar II.2 Navigasi Linier

2. Nonlinear

Pengguna akan melakukan navigasi dengan bebas melalui isi proyek dengan tidak terikat dengan jalur yang sudah ditentukan sebelumnya.

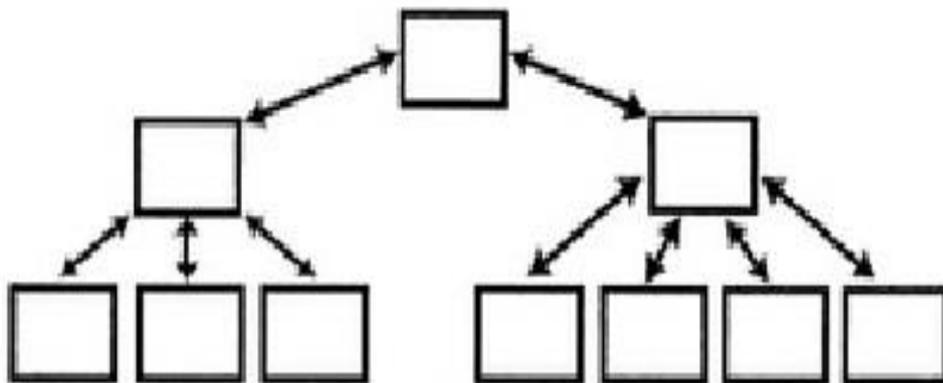


Sumber: Vaughan (2006:367-368)

Gambar II.3 Navigasi Nonlinear

3. Hierarkis

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

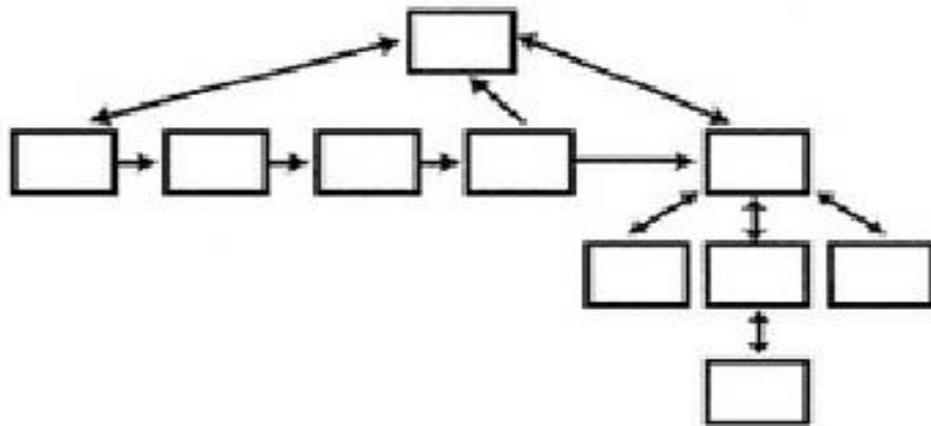


Sumber: Vaughan (2006:367-368)

Gambar II.4 Navigasi Hierarki

4. Komposit

Pengguna akan melakukan navigasi dengan bebas (secara non-linear), tetapi terkadang dibatasi presentasi linear film atau informasi penting dan/atau pada data yang paling terorganisasi secara logis pada suatu hierarki.



Sumber: Vaughan (2006:367-368)

Gambar II.5 Navigasi Komposit

2.2.3. ERD (Entity Relationship Diagram)

1. Pengertian ERD

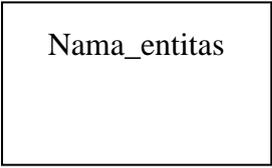
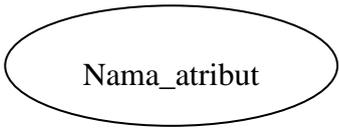
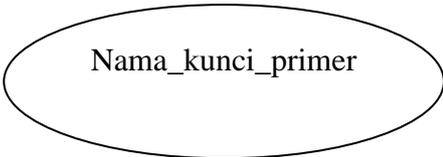
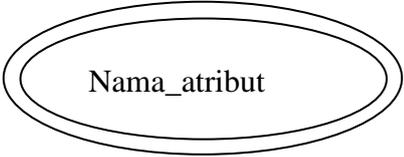
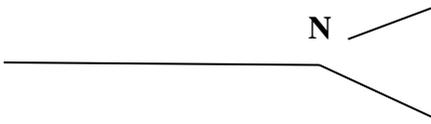
Menurut Ladjamudin (2013:142) ERD adalah “suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak”.

Menurut Rosa dan Shalahuddin (2015:50) Model E-R atau ERD adalah “pemodelan awal basis data yang paling banyak digunakan adalah *Entity Relationship Diagram* (ERD)”.

Berdasarkan teori di atas ERD adalah suatu model jaringan yang menggunakan susunan data yang paling banyak digunakan.

Adapun beberapa dasar membangun pemodelaan ERD menurut Rossa dan Shalahudin (2015:50) sebagai berikut:

Tabel II.1. Komponen ERD

<p>Entitas / <i>entity</i></p> 	<p>Entitas merupakan data inti yang akan disimpan; bakal tabel basis data.</p>
<p>Atribut</p> 	<p><i>Field</i> atau kolom data yang harus disimpan dalam suatu entitas.</p>
<p>Atribut kunci primer</p> 	<p><i>Field</i> atau kolom data yang harus disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i>. Yang diinginkan, biasanya berupa id.</p>
<p>Atribut multivalui / <i>Multivalued</i></p> 	<p><i>Field</i> atau kolom data yang harus disimpan dalam satu entitas yang dapat memiliki nilai lebih dari satu.</p>
<p>Relasi</p> 	<p>Relasi yang menghubungkan antara entitas; biasanya awali dengan kata kerja.</p>
<p>Asosiasi / <i>association</i></p> 	<p>Penghubung antara relasi dan entitas di mana kedua ujungnya memiliki <i>multiplicity</i>.</p>

Sumber : Rosa dan Shalahuddin (2015:50)

2. Kardinalitas

Menurut Yanto (2016:40) derajat kardinalitas merupakan ”penjabaran dari hubungan antara entitas ”.

Terdapat 3 macam kardinalitas menurut Yanto(2016:40) yaitu:

a. Derajat kardinalitas One to One

Derajat kardinalitas one to one jika satu entitas x hanya berelasi dengan satu entitas y, ataupun sebaliknya

b. Derajat kardinalitas one to many

Derajat kardinalitas one to many terjadi jika satu entitas x berelasi dengan banyak entitas y, ataupun sebaliknya.

c. Derajat kardinalitas many to many

Derajat kardinalitas many to many terjadi jika banyak entitas x berelasi dengan banyak entitas y ataupun sebaliknya.

Menurut Ladjamudin (2013:144) derajat relationship adalah “Jumlah entitas yang berpartisipasi dalam satu relationship”. Terdapat 3 macam kardinalitas

Menurut Ladjamudin (2013:144) yaitu:

1. One To One tingkat hubungan satu ke satu, dinyatakan dengan satu kejadian pada entitas pertama, hanya mempunyai satu hubungan dengan satu kejadian pada entitas yang kedua dan sebaliknya.
2. One to many atau many to one tingkat hubungan satu ke banyak adalah nama dengan banyak kesatu. Tergantung dari arah mana hubungan tersebut dilihat. Untuk satu kejadian pada entitas yang pertama dapat mempunyai hubungan dengan kejadian pada entitas yang keduanya. Sebaliknya satu

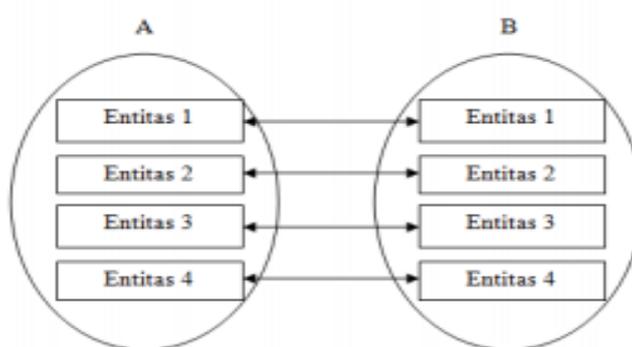
kejadian pada entitas yang kedua hanya dapat mempunyai satu hubungan dengan satu kejadian pada entitas pertama.

3. many to many tingkat hubungan banyak ke banyak terjadi jika tiap kejadian pada sebuah entitas akan mempunyai banyak hubungan dengan kejadian pada entitas lainnya.

Menurut Muhib (2013:10) kardinalitas atau derajat relasi adalah “Kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain”.

Kardinalitas yang terjadi diantara dua himpunan entitas (misal A dan B) dapat berupa :

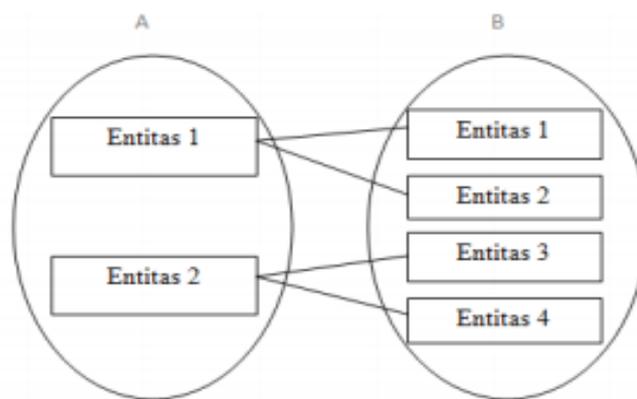
1. Satu ke Satu (One to One) yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu juga sebaliknya, setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.



Sumber : Muhib (2013:10)

Gambar II.6 Kardinalitas relasi satu ke satu

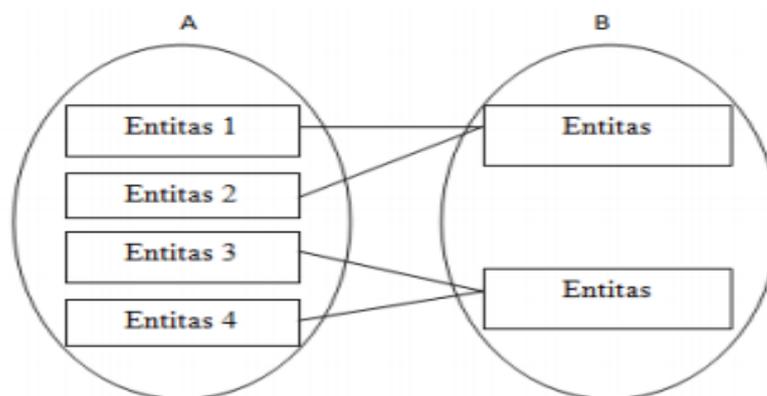
2. Satu ke Banyak (One to Many) yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.



Sumber : Muhib (2013:10)

Gambar II.7 Kardinalitas relasi satu ke banyak

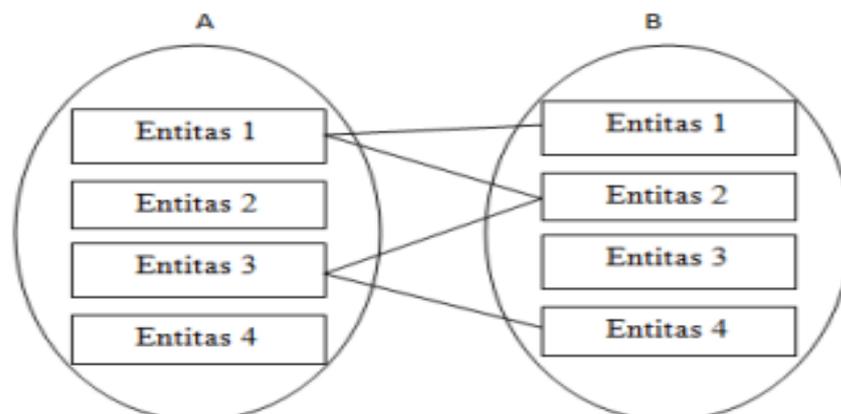
3. Banyak ke Satu (Many to One) Setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada satu entitas himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B.



Sumber : Muhib (2013:10)

Gambar II.8. Kardinalitas relasi banyak ke satu

4. Banyak ke Banyak (Many to Many) yang berarti setiap entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian juga sebaliknya, dimana setiap entitas pada himpunan B dapat berhubungan dengan banyak entitas pada himpunan entitas A.



Sumber : Muhib (2013:10)

Gambar II.12. Kardinalitas relasi banyak ke banyak

2.2.4. LRS (Logical Record Structur)

Menurut Hasugian dan Shidiq (2012:608) *logical record structure* (LRS) adalah “sebuah model sistem yang di gambarkan dengan sebuah diagram-ER akan mengikuti pola atau aturan pemodelan tertentu dalam kaitannya dengan konversi ke LRS, maka perubahan terjadi adalah mengikuti aturan-aturan ini”.

Menurut Ladjamudin (2013:163) “*logical record structure* menyatakan transformasi ERD/LRS sering disebut dengan mapping ERD ke *database relational*”.

Berdasarkan teori di atas, LRS (*logical record strukture*) adalah gambaran sebuah model dengan mapping ERD yang akan mengikuti pola/aturan pemodelan tertentu dalam kaitan dengan konversi ke LRS.

2.2.4. Pengujian Web

Menurut Rosa dan Shalahuddin (2015:275) “*black-box-testing* adalah pengujian perangkat lunak dari segi spesifikasi fungsional tanpa penguji desain dan kode program”.

Menurut Rizki (2011:264) “*black-bok-testing* adalah tipe *testing* yang memperlakukan perangkat lunak yang tidak diketahui kinerja internalnya”.

Berdasarkan teori di atas, *black-box-testing* adalah pengujian perangkat lunak dari segi spesifikasi fungsional tanpa penguji desain dan kode program yang tidak di ketahui kinerja internalnya.

Menurut Simarmata (2010:316-321) klasifikasi *black box* mencakup beberapa pengujian, yaitu:

1. Pengujian Fungsional (*Functional Testing*)

Pengujian dilakukan dalam bentuk tertulis untuk memeriksa apakah aplikasi berjalan seperti yang diharapkan. Pengujian fungsional meliputi seberapa baik system melaksanakan fungsinya, termasuk perintah-perintah pengguna, manipulasi data, pencarian dan proses bisnis, pengguna layar, dan integrasi. Pengujian fungsional juga meliputi permukaan yang jelas dari jenis fungsi-fungsi, serta operasi *back-end* (seperti, keamanan dan bagaimana meningkatkan sistem).

2. Pengujian Tegangan (*Stress Testing*)

Pengujian tegangan berkaitan dengan kualitas aplikasi didalam lingkungan. Pengujian ini adalah hal yang paling sulit, cukup kompleks dilakukan, dan memerlukan upayah bersama dari semua tim.

3. Pengujian Beban (*Load Testing*)

Pada pengujian beban, aplikasi akan diuji dengan beban berat atau masukan, seperti yang terjadi pada pengujian situs *web*/untuk mengetahui apakah aplikasi/situs gagal atau kinerjanya menurun.

4. Pengujian Khusus (*Ad-hoc Testing*)

Jenis pengujian ini dilakukan tanpa penciptaan rencana pengujian (*test plan*) atau kasus pengujian *test case*.

5. Pengujian Penyelidikan (*Exploratory Testing*)

Pengujian penyelidikan mirip dengan pengujian khusus dan dilakukan untuk memperlajari/mencari aplikasi.

6. Pengujian Usabilitas (*Usability Testing*)

Pengujian ini disebut juga sebagai pengujian untuk keakraban pengguna (*testing for user-friendliness*).

7. Pengujian Asap (*Smoke Testing*)

Jenis pengujian ini disebut juga pengujian kenormalan (*sanity testing*).

8. Pengujian Pemulihan (*Recovery Testing*) Pengujian pemulihan pada dasarnya dilakukan untuk memeriksa seberapa cepat dan baiknya aplikasi bisa pulih terhadap semua jenis *crash* atau kegagalan *hardware*, masalah bencana, dan lain-lain.

9. Pengujian Volume (*Volume Testing*)

Pengujian volume dilakukan terhadap efisiensi dari aplikasi.

10. Pengujian Domain (*Domain Testing*)

Pengujian domain merupakan penjelasan yang paling sering menjelaskan teknik pengujian.

11. Pengujian Skenario (*Scenario Testing*)

Pengujian skenario adalah pengujian yang *realistis*, *kredibel* dan memotivasi *stakeholder*, tantangan untuk program dan mempermudah penguji untuk melakukan evaluasi

12. Pengujian Regresi (*Regression Testing*)

Pengujian regresi adalah gaya pengujian yang berfokus pengujian ulang (*retesting*) setelah ada perubahan.

13. Pengujian Pengguna (*User Acceptance*)

Pada jenis pengujian ini, perangkat lunak akan diserahkan kepada pengguna untuk mengetahui apakah perangkat lunak memenuhi harapan pengguna dan bekerja seperti yang diharapkan.

14. Pengujian Alfa (*Alpha Testing*)

Pada jenis pengujian ini, pengguna akan diundang ke pusat pengembangan.

15. Pengujian Beta (*Beta Testing*)

Pada jenis pengujian ini, perangkat lunak didistribusikan sebagai sebuah versi beta dengan pengguna yang menguji aplikasi di situs mereka.