

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Web

Sebuah situs atau *web* biasanya ditempatkan setidaknya pada sebuah *server web* yang dapat diakses melalui jaringan *internet*, ataupun jaringan wilayah lokal melalui alamat *internet* yang dikenali sebagai *url*. Gabungan atas semua situs yang dapat diakses publik di *internet* disebut juga sebagai *world wide web (www)*. Pada bab ini diuraikan beberapa konsep dasar *web*, diantara sebagai berikut :

A. Website

1. 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".

Sebuah *website* biasanya dibangun atas banyak halaman *web* yang saling berhubungan. Hubungan antara satu halaman *web* dengan halaman *web* lainnya disebut dengan *hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *hypertext*.

Suatu situs *web* akan lebih dikenal dengan cepat apabila informasi yang disajikan selalu *up-to-date* dan lengkap. Perlu disadari juga bahwa untuk melakukan



up-to-date merupakan suatu permasalahan yang tidak mudah untuk dilakukan, selain membutuhkan waktu juga sangat merepotkan dalam melakukan penambahan atau perubahan data yang akan disajikan.

Jika sebuah situs tidak dirancang secara sistematis, maka akan terjadi kesulitan dan kerepotan dalam melakukan pemeliharaan, meng-*update* dan melengkapi data. PHP dan *database* merupakan solusi yang tepat digunakan dalam mengelola situs. Hanya dengan melakukan penambahan atau perubahan data kedalam database tanpa harus melakukan penambahan atau editing HTML (*HyperText Markup Language*).

Website adalah halaman informasi yang disediakan melalui jalur internet sehingga bisa diakses di seluruh dunia selama terkoneksi dengan jaringan *internet*.

2. *Web Browser*

Menurut Sibero (2013:12) "*Web Browser* adalah aplikasi perangkat lunak yang digunakan untuk mengambil dan menyajikan sumber informasi *web*. Sumber informasi *web* diidentifikasi dengan *Uniform Resource Identifier* (URI) yang dapat terdiri dari halaman *web*, gambar, atau konten lainnya."



Salah satu program *web browser*, seperti: *Mozilla Firefox*, *Internet Explorer (IE)*, *Opera*, *Safari*, *Google Chrome* 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.

Dalam Bahasa Indonesia *browser* diterjemahkan sebagai perambah web. Perambah web adalah jenis perantara pengguna yang paling sering digunakan

3. *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”

Server *web* yang di usung oleh XAMPP adalah *apache* yang diakui kemampuan dan keandalannya. XAMPP itu sendiri adalah paket perangkat lunak yang mempunyai arti dari singkatannya, X yang berarti sifat (*cross-plaftform*), A (*webserver Apache*), M (*database Mysql*), P (bahasa pemrograman PHP) dan P (bahasa pemrograman *Perl*).



4. *Internet*

Menurut Saputra (2013:5), “*Internet* merupakan hubungan antar berbagai jenis komputer dan jaringan didunia yang berbeda sistem operasi maupun aplikasinya dimana hubungan tersebut kemajuan media komunikasi yang menggunakan protokol dalam berkomunikasi yaitu protokol TCP/IP”.

5. *E-Commerce*

Menurut Wong (2010:33), “*E-commerce* adalah pembelian, penjualan dan pemasaran barang serta jasa melalui sistem elektronik. Seperti televisi, radio dan jaringan komputer atau *internet*”.

E-commerce juga merupakan konsep baru yang bisa digambarkan sebagai proses jual beli barang atau jasa pada *world wide web internet* atau proses jual beli atau pertukaran produk, jasa dan informasi melalui jaringan informasi termasuk *internet*.

E-commerce pertama kali diperkenalkan pada tahun 1994 pada saat pertama kali banner-elektronik dipakai untuk tujuan promosi dan periklanan di suatu halaman-web (*website*). Menurut Riset Forrester, perdagangan elektronik menghasilkan penjualan seharga AS\$12,2 miliar pada 2003. Menurut laporan yang lain pada bulan oktober 2006 yang lalu, pendapatan ritel online yang bersifat non-travel di Amerika Serikat diramalkan akan mencapai seperempat trilyun dolar US pada tahun 2011.

B. Bahasa Pemrograman

1. HTML (*Hypertext Markup Language*)

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

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.

2. CSS (*Cascading Style Sheet*)

Menurut Ardhana (2012:108), “*Cascading Style Sheet (CSS)* merupakan salah satu bahasa pemrograman *web* untuk mengedalikan beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dan seragam”.

Pada umumnya CSS dapat mengedalikan ukuran gambar, ukuran border, spasi antar paragraf warna pada teks, warna tabel, atur margin, padding dan mengedalikan lainnya. CSS sama hal dengan *Microsoft Office Word* yang dapat mengatur beberapa *style*, misalnya *heading*, *subbab*, *bodytext*, *footer*, *header*, *images* dan *style* lainnya.

3. PHP (*Personal Home Page*)

Menurut Ardhana (2012:88), “*PHP* merupakan bahasa pemrograman berbasis *server-side* yang dapat melakukan *parsing script php* yang menjadi *script web* sehingga dari sisi client menghasilkan tampilan yang menarik”.

PHP pertama kali ditemukan oleh Rasmus Lerdoft pada tahun 1995. Pada waktu itu PHP masih bernama FI (*Form Interpreted*), yang wujudnya berupa skrip yang digunakan untuk mengolah data formulir dari *web*.

Sesuai dengan namanya, PHP digunakan untuk membuat website pribadi. Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang *powerful* dan tidak hanya digunakan untuk membuat halaman web sederhana, tetapi juga *website populer* yang digunakan oleh jutaan orang seperti *wikipedia*, *wordpress*, *joomla*, dan lain sebagainya.

4. Javascript

Menurut Sibero (2013:150), “*JavaScript* adalah suatu bahasa pemrograman yang dikembangkan untuk dapat berjalan pada *web browser* atau bahasa skrip (*Scripting Language*) dengan kumpulan intruksi perintah yang digunakan untuk mengendalikan beberapa bagian dari sistem operasi”.

JavaScript dikembangkan oleh *Netscape*, sebagai bahasa pemrograman “sederhana” karena tidak dapat digunakan untuk membuat suatu aplikasi ataupun *applet*. Namun, dengan *JavaScript*, kita dapat membuat sebuah halaman *web* yang interaktif dengan mudah.”

Javascript biasa disisipkan pada halaman web menggunakan tag `<script>`. Terkadang ada juga yang ingin menjalankan Javascript dipisahkan, kemudian berkas tersebut dipanggil dari dokumen HTML. Berkas Javascript tersebut disimpan dengan ekstensi `.js`. JavaScript bisa digunakan dalam membuat efek baik digambar maupun teks.



5. JQuery

Menurut Utomo (2012:62), “*JQuery* adalah *library JavaScript* yang banyak digunakan saat ini. *Jquery* dibuat oleh John Resig pada tahun 2006. Banyak website yang memanfaatkan *library* ini untuk menyederhanakan fungsi-fungsi yang ada pada *JavaScript* atau *Ajax*”.

6. 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".

7. Sublime Text

Menurut Guillermo (2015:4), "*Sublime Text* adalah teks editor serbaguna dan menyenangkan untuk kode dan prosa yang mengotomatisasi tugas yang berulang sehingga Anda dapat fokus hal-hal penting". *Sublime Text* bekerja pada sistem operasi *Windows* dan *Linux*". *Sublime Text* juga merupakan salah satu teks editor yang *professional* mendesain, menulis kode program dan mengembangkan *website*, halaman *web*, dan aplikasi *web*.

Sublime Text mendukung berbagai bahasa pemrograman dan mampu menyajikan fitur *syntax highlight* hampir semua bahasa pemrograman yang didukung ataupun dikembangkan oleh komunitas seperti *C*, *C++*, *C#*, *CSS*, *D*, *Dylan*, *Erlang*, *HTML*, *groovy*, *Haskell*, *Java*, *JavaScript*, *LaTeX*, *Lisp*, *Lua*, *PHP*, *Python*, *SQL*, *Textile* and *XML*. Biasanya bagi bahasa pemrograman yang didukung ataupun belum terdukung secara *default* dapat lebih dimaksimalkan atau didukung dengan menggunakan kebutuhan user.



Fungsi yang sangat populer di *Sublime Text* adalah sebagai berikut :

a) *Go to Anything*

Cara menggunakannya dengan menekan tombol CTRL + P / CMD + P untuk loncat ke suatu *Symbol*, sedangkan simbol # untuk pencarian didalam file tersebut dan titik dua (:) untuk loncat ke nomor baris tertentu.

b) *Multiple Selection*

Cara menggunakannya dengan menekan tombol CTRL + Shift + L / CMD + Shift + L. Tombol tersebut berfungsi untuk menyeleksi 1 baris kode untuk dipindahkan ke baris lain. Sedangkan CTRL + D / CMD + D untuk menambahkan penyeleksian lainnya.

c) *Command Pallete*

Cara menggunakannya dengan menekan tombol CTRL + Shift + P / CMD + Shift + P. Ini digunakan untuk penyortiran, mengubah sintaks dan mengubah pengaturan indentasi. Dengan hanya beberapa *keystrokes*, anda dapat mencari apa yang Anda inginkan, tanpa harus menavigasi melalui menu.



d) *Split Editing*

Dengan *Split editing*, anda dapat membagi layar editor untuk mengerjakan banyak *code* sekaligus. Caranya dengan Klik menu *View -> Layout*. Pilih *Column Split* yang diinginkan.

e) *Dukungan Plugin*

Ini yang tidak kalah penting, kita dapat menambahkan beberapa plugin yang kita inginkan seperti *snippet*, *auto complete* dalam bahasa pemrograman tertentu.

C. Basis Data

Menurut Utami (2013:43), “Basis Data adalah media untuk penyimpanan data agar dapat diakses dengan mudah dan cepat”. Sistem informasi tidak dapat dipisahkan dengan kebutuhan akan basis data apapun bentuknya, entah berupa *file teks* ataupun *Database Management System (DBMS)*.

Basis Data atau *Database* merupakan kumpulan dari item yang saling berhubungan satu dengan yang lainnya yang diorganisasikan berdasarkan sebuah skema atau struktur.

Aplikasi Basis Data yang digunakan yang digunakan antara lain :

1. MySQL

Menurut Sadeli (2014:10), “MySQL adalah database yang menghubungkan *script PHP* menggunakan perintah *query* dan *escape character* yang sama dengan PHP. Mysql mempunyai tampilan *Open* yang mempermudah anda dalam mengakses database”.



2. phpMyAdmin

Menurut Sadeli (2014:10), ”phpMyAdmin adalah sebuah perangkat lunak (*software*) yang berbentuk seperti halaman situs yang terdapat pada *web server*”.

Cara untuk mengakses PhpMyAdmin di web server lokal, anda dapat mengakses PhpMyAdmin menggunakan URL <http://localhost/PhpMyAdmin>.

Beberapa fitur yang dimiliki phpMyAdmin :

- a. *Impor* data dari CSV dan SQL
- b. *Ekspor* data ke berbagai format: CSV, SQL, XML, PDF, ISO / IEC 26300 - *OpenDocument Text* dan *Spreadsheet, Word, Excel, LATEKS* dan lain-lain

D. Model Pengembangan Perangkat Lunak

Menurut Pressman (2010:39), "Waterfall adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software*".

Fase-fase dalam model waterfall menurut referensi Pressman (2010:39):

1. *Communication*

Langkah ini merupakan analisis terhadap kebutuhan *software*, dan tahap untuk mengadakan pengumpulan data dengan melakukan pertemuan dengan *customer*, maupun mengumpulkan data-data tambahan baik yang ada di jurnal, artikel, maupun dari internet.

2. *Planning*

Proses *planning* merupakan lanjutan dari proses *communication* (*analysis requirement*). Tahapan ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan *software*, termasuk rencana yang akan dilakukan.



3. *Modeling*

Proses *modeling* ini akan menerjemahkan syarat kebutuhan ke sebuah perancangan *software* yang dapat diperkirakan sebelum dibuat *coding*.

4. *Construction*

Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu *software*, artinya penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat tadi. Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut untuk kemudian bisa diperbaiki.

5. *Deployment*

Tahapan ini bisa dikatakan *final* dalam pembuatan sebuah software atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan oleh *user*. Kemudian *software* yang telah dibuat harus dilakukan pemeliharaan secara berkala.

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).

Kelebihan dari model ini adalah selain karena pengaplikasian menggunakan model ini mudah juga dapat mendefinisikan kebutuhan sistem secara utuh, eksplisit, dan benar di awal proyek, maka *Software Engineering* (SE) dapat berjalan dengan

baik dan tanpa masalah. Meskipun seringkali kebutuhan sistem tidak dapat di definisikan se-eksplisit yang diinginkan, tetapi paling tidak, *problem* pada kebutuhan sistem di awal proyek lebih ekonomis dalam hal uang (lebih murah), usaha, dan waktu yang terbuang lebih sedikit jika dibandingkan *problem* yang muncul pada tahap-tahap selanjutnya.

Kekurangan yang utama dari model ini adalah kesulitan dalam mengakomodasi perubahan setelah proses dijalani. *Fase* sebelumnya harus lengkap dan selesai sebelum mengerjakan *fase* berikutnya.

2.2. Teori Pendukung

A. Struktur Navigasi

Menurut Binanto (2010:268), "Struktur navigasi adalah gabungan dari struktur referensi informasi situs web dan mekanisme link yang mendukung pengunjung untuk melakukan penjelajahan situs".

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.

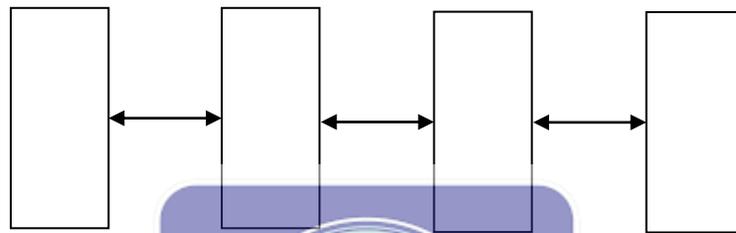
Struktur navigasi juga dapat diartikan sebagai struktur alur dari suatu program yang merupakan rancangan hubungan dan rantai kerja dari beberapa area yang berbeda dan dapat membantu mengorganisasikan seluruh elemen pembuatan *website*.



Ada 4 (empat) macam bentuk struktur navigasi yang biasa digunakan dalam pembuatan web, yaitu:

1. Struktur Navigasi *Linier*

Struktur navigasi *linier* adalah struktur yang hanya mempunyai satu rangkaian cerita yang berurut, yang menampilkan satu demi satu tampilan layar secara berurut menurut urutannya.



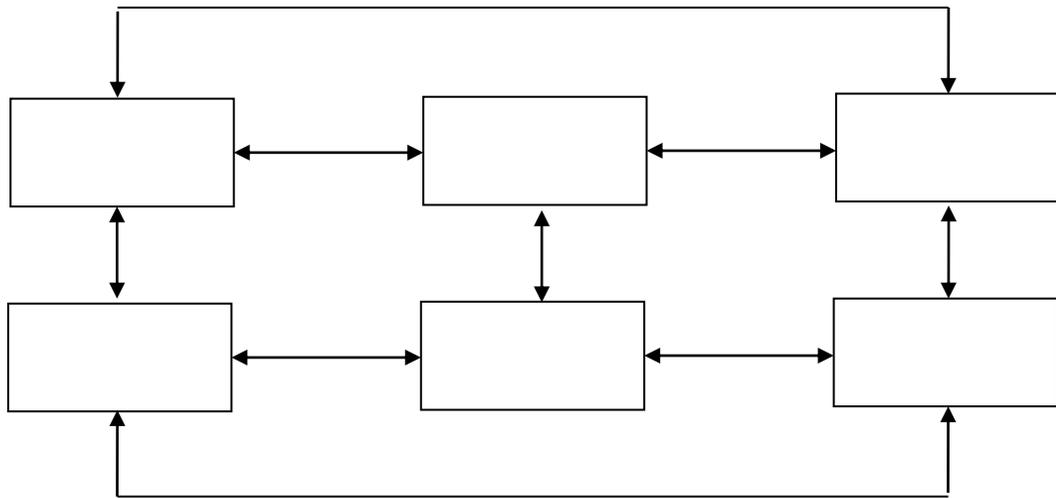
Sumber : Binanto (2010:269)

Gambar II.1

Struktur Navigasi *Linier*

2. Struktur Navigasi *Non Linier*

Struktur navigasi *non linier* merupakan pengembangan dari struktur navigasi *linier*. Pada struktur ini diperkenankan membuat navigasi bercabang. Percabangan pada struktur navigasi *non linier* ini berbeda dengan percabangan pada percabangan *non linier* ini walaupun terdapat percabangan, tetapi tiap-tiap tampilan mempunyai kedudukan yang sama yaitu tidak ada *Master Page* dan *Slave*.



Sumber: Binanto (2010:269)

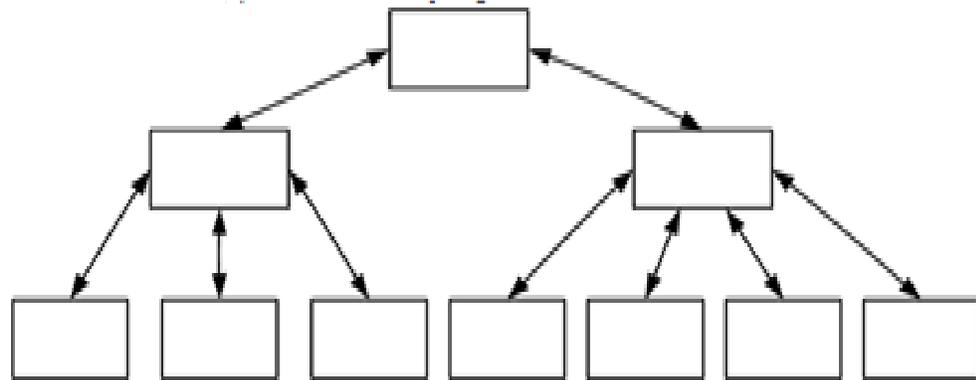


Gambar II.2

Struktur Navigasi Non Linier

3. Struktur Navigasi Hirarki

Struktur navigasi hirarki biasa disebut struktur bercabang, merupakan suatu struktur yang mengandalkan percabangan untuk menampilkan data berdasarkan kriteria tertentu. Dimana tampilan pada menu pertama akan disebut *Master Page* (halaman utama pertama) yang memiliki halaman percabangan yaitu *Slave Page* (halaman pendukung). Jika salah satu halaman pendukung akan diaktifkan, maka tampilannya disebut *Master Page* (halaman utama kedua) dan seterusnya.



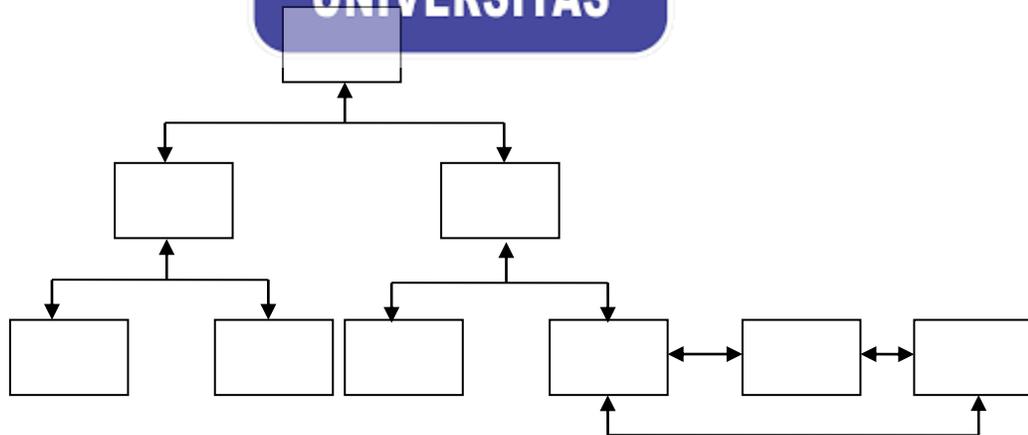
Sumber : Binanto (2010:269)

Gambar II.3

Struktur Navigasi Hirarki

4. Struktur Navigasi Campuran

Struktur navigasi campuran merupakan gabungan dari ketiga struktur navigasi *linier*, *non-linier* dan hirarki. Struktur navigasi campuran disebut juga sebagai struktur navigasi bebas. Struktur navigasi ini biasa digunakan dalam pembuatan multimedia karena dapat memberikan keinteraksian yang lebih tinggi.



Sumber: Binanto (2010:269)

Gambar II.4

Struktur Navigasi Campuran

B. ERD (*Entity Relationship Diagram*)

1. ERD

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".

Dengan ERD kita berusaha untuk mentransformasikan keadaan dari dunia nyata ke dalam bentuk basis data. Dalam pembentukan ERD terdapat 3 Komponen yang akan dibentuk yaitu :

a) Entitas (*Entity*)

Entitas merupakan mengenai basis data yaitu suatu obyek yang dapat dibedakan dari lainnya yang dapat diwujudkan dalam basis data. Pengertian lainnya.

Contoh : Buku, Perpustakaan

b) Hubungan / Relasi (*Relationship*)

Suatu hubungan adalah hubungan antara dua jenis entitas dan direpresentasikan sebagai garis lurus yang menghubungkan dua entitas.

Contoh : Pelanggan mendaftar sebagai member sewa pada rental mobil, relasinya adalah mendaftar.

c) Atribut

Atribut adalah suatu yang memberikan informasi lebih rinci tentang jenis entitas. Atribut memiliki struktur internal berupa tipe data.



Jenis-jenis atribut :

- 1) *Atribut Key* adalah satu atau gabungan dari beberapa atribut yang dapat membedakan semua baris data (*Row/Record*) dalam tabel secara unik.
- 2) *Atribut simple* adalah Atribut yang bernilai *atomic*, tidak dapat dipecah/dipilah lagi

Contoh : Alamat, penerbit, tahun terbit, judul buku.

- 3) *Atribut Multivalue* adalah Nilai dari suatu atribut yang mempunyai lebih dari satu (multivalue) nilai dari atribut yang bersangkutan.

Contoh : dari sebuah buku, yaitu terdapat beberapa pengarang.

- 4) *Atribut composite* adalah suatu atribut yang terdiri dari beberapa atribut yang lebih kecil yang mempunyai arti tertentu yang masih bisah dipecah lagi atau mempunyai sub atribut.

Contoh : dari entitas nama yaitu nama depan, nama tengah, dan nama belakang



- 5) *Atribut Derivatif* adalah atribut yang tidak harus disimpan dalam database. Total. atau atribut yang dihasilkan dari atribut lain atau dari suatu *relationship*. Atribut ini dilambangkan dengan bentuk *oval* yang bergaris putus-putus

2. Derajat *Relationship* (Kardinilitas ERD)

Menjelaskan jumlah maksimum hubungan antara satu entitas dengan entitas lainnya.

a) *One to One* (1:1)

Setiap anggota entitas A hanya boleh berhubungan dengan satu anggota entitas B, begitu pula sebaliknya.

b) *One to many* (1:M / Many)

Setiap anggota entitas A dapat berhubungan dengan lebih dari satu anggota entitas B tetapi tidak sebaliknya.

c) *Many to Many* (M:M)

Setiap entitas A dapat berhubungan dengan banyak entitas himpunan entitas B dan demikian pula sebaliknya.

3. LRS (*Logical Record Structure*)

Menurut Priyadi (2014:15), "*Logical Record Structure* (LRS) adalah resperentasi 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 *entity relationship* diagram nama tipe record berada diluar kotak field tipe *record* ditempatkan.

Perbedaan LRS dan ERD adalah nama dan tipe *record* berada diluar kotak *field* tipe *record* ditempatkan. LRS terdiri dari *link-link* diantara tipe *record*. *Link* ini menunjukkan arah dari satu tipe *record* lainnya. Banyak *link* dari LRS yang diberi



tanda *field-field* yang kelihatan pada kedua *link* tipe *record*. Pengambaran LRS mulai dengan menggunakan model yang dimengerti.

C. Pengujian Web

1. *Black Box Testing*

Black Box Testing terfokus apakah unit program memenuhi kebutuhan (*requirement*) yang disebutkan dalam spesifikasi. Pada *black box testing*, cara pengujian hanya dilakukan dengan menjalankan atau mengeksekusi unit atau modul. Kemudian diamati apakah hasil dari unit itu sesuai dengan proses bisnis yang diinginkan.

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".



Black Box Testing berusaha untuk menemukan kesalahan dalam kategori berikut:

- a) Fungsi yang tidak benar atau fungsi yang hilang
- b) Kesalahan antarmuka
- c) Kesalahan dalam struktur data atau akses database eksternal
- d) Kesalahan perilaku (*behavior*) atau kesalahan kinerja
- e) Inisialisasi dan pemutusan kesalahan