

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Web

Pembuatan Tugas Akhir ini tidak terlepas dari teori-teori yang mendukung dalam mempelajari serta merancang *website e-commerce* yang diharapkan berfungsi secara maksimal. Kemudahan dalam melihat suatu *website* bagi setiap pengunjung akan sangat membantu dalam menyelesaikan setiap pencarian katalog serta transaksi penjualan secara langsung.

A. Website

Menurut Hidayat (2006:6) “*Website* adalah suatu sistem yang berkaitan dengan dokumen digunakan sebagai media untuk menampilkan teks, gambar, multimedia dan lainnya pada jaringan *internet*”. Seperti halnya jaringan komputer lokal maupun jaringan komputer area, *internet* juga menggunakan protokol komunikasi yang sama yaitu TCP/IP (Transmission Control Protocol / Internet Protocol)”.

Adapun istilah yang ada di *internet* meliputi sebagai berikut :

1. Web Browser

Menurut Sibero (2011:19) “Web Browser merupakan program yang berfungsi untuk menampilkan dokumen-dokumen web dalam format HTML”.

Menurut Sibero (2011:12) “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 terdiri dari halaman web, video, gambar ataupun konten lainnya”.

Dari pendapat diatas, dapat disimpulkan bahwa web browser adalah program aplikasi yang digunakan untuk menampilkan informasi berupa video, gambar, ataupun konten lainnya yang dibaca dari format HTML.

2. WWW (*World Wide Web*)

Menurut Sidik (2009:6) “*World Wide Web* (WWW) lebih dikenal dengan *web* awalnya merupakan suatu layanan penyajian informasi di *internet*, karna kepopulerannya saat ini, *web* sudah menjadi *interface* aplikasi untuk melakukan informasi yang lengkap deri seluruh dunia.

3. *E-Commerce*

Menurut Triton (2006:16) “*E-Commerce* dapat didefinisikan sebagai perdagangan elektronik dimana bentuk transaksi perdagangan baik membeli maupun menjual dilakukan melalui elektronik pada jaringan *internet*”. Istilah *E-Commerce* juga dapat digunakan untuk menunjuk pada penggunaan jaringan-jaringan komputer mencari informasi yang menunjang proses pembuatan keputusan oleh individu dan perusahaan. Adapun ruang lingkup dalam *e-commerce*, diantaranya:

a. *Business To Business* (B2B)

Merupakan sitem komunikasi bisnis antar pelaku bisnis atau transaksi secara elektronik antar perusahaan yang dilakukan secara rutin dan dalam kapasitas produk yang besar.

b. *Business To Customer* (B2C)

Merupakan sistem komunikasi bisnis antar pelaku bisnis dengan konsumen untuk memenuhi kebutuhan tertentu kebutuhan pada saat tertentu.

c. *Customer To Customer (C2C)*

Merupakan sistem komunikasi dan transaksi bisnis antar konsumen untuk memenuhi kebutuhan tertentu pada saat tertentu.

d. *Customer To Business (C2B)*

Merupakan individu yang menjual produk atau jasa kepada organisasi dan individu yang mencari penjual dan melakukan transaksi.

4. *Internet (Interconnection Networking)*

Menurut Sibero (2011:10) “*Interconnection Networking (Internet)* adalah jaringan computer yang menghubungkan antar jaringan secara *global*, *internet* dapat juga disebut jaringan dalam suatu jaringan yang luas”. *Internet* juga menggunakan *protocol* komunikasi *Transmission Control/Internet Protocol Suite (TCP/IP)*. *Protocol* adalah spesifikasi sederhana mengenai bagaimana komputer saling bertukar informasi.

Transmission Control Protocol (TCP) bertugas untuk memastikan bahwa semua hubungan bekerja dengan benar, sedangkan *Internet Protocol Suite (IP)* yang mentransmisikan data dari suatu komputer ke komputer yang lain.

B. Bahasa Pemrograman

1. *Personal Home Page (PHP)*

Menurut Prasetyo (2014:122) “*PHP (Hypertext Preprocessor)* adalah anak *script* yang ditanam disisi *server*.” Seperti sebagian besar bahasa *script* lainnya, *PHP* dapat ditanamkan langsung kedalam *HTML*. Kode *PHP* dipisahkan dari *HTML* dengan menggunakan tanda *start* `<?php` dan *end* `?>`. Ketika sebuah dokumen dibaca,

prosesor *PHP* hanya menterjemahkan area yang ditandai saja dan menampilkan hasilnya apda tempat yang sama.

2. *JQuery*

Menurut Wahana Komputer (2012:8) “*JQuery* merupakan salah satu dari sekian banyak *framework* yang digunakan dan memiliki pengguna yang paling banyak. *JQuery* merupakan *JavaScript* yang dibangun untuk mempercepat dan memperingkas serta menyederhanakan manipulasi dokumen *HTML*, penanganan *event*, animasi, dan interaksi *Ajax* untuk mempercepat pengembangan *web*. Dengan *JQuery*, *developer* akan dimanjakan dengan suatu pemrograman *JavaScript* yang sangat sederhana jika dibandingkan dengan *native Java Script*. Sebelum anda memulai mempelajari *JQuery*, anda harus mempunyai pengetahuan dasar mengenai *HTML*, *CSS*, dan *JavaScript*.”

3. *HTML*

Menurut Saputra (2012:1) “*HTML* merupakan singkatan dari *Hyper Teks Markup Language*. *HTML* bisa disebut bahasa paling dasar dan penting yang digunakan untuk menampilkan dan mengelola tampilan pada halaman *website*”.

HTML digunakan untuk digunakan menampilkan berbagai informasi di dalam sebuah penjelajah *wen internet* dan *formatting hypertext* sederhana yang ditulis kedalam berkas format *ASCII* agar dapat menghasilkan tampilan wujud yang terintegrasi.

Dengan kata lain, berkas yang dibuat di dalam perangkat lunak pengolah kata dan disimpan kedalam format *ASCII* normal akan menjadi *homepage* dengan perintah-perintah *html*.

Html mempunyai dua macam ekstensi (agar kita dapat membangun dokumen html). Extensi tersebut adalah htm dan html. Format ekstensi htm awalnya hanya untuk mengakomodasi [pengguna html dalam operasi DOS. Html saat ini telah mencapai versi 5, dimana beberapa fitur unggulan telah disuntikkan kedalam pemrograman tersebut misalnya *audio*, *video*, dan lain-lain.

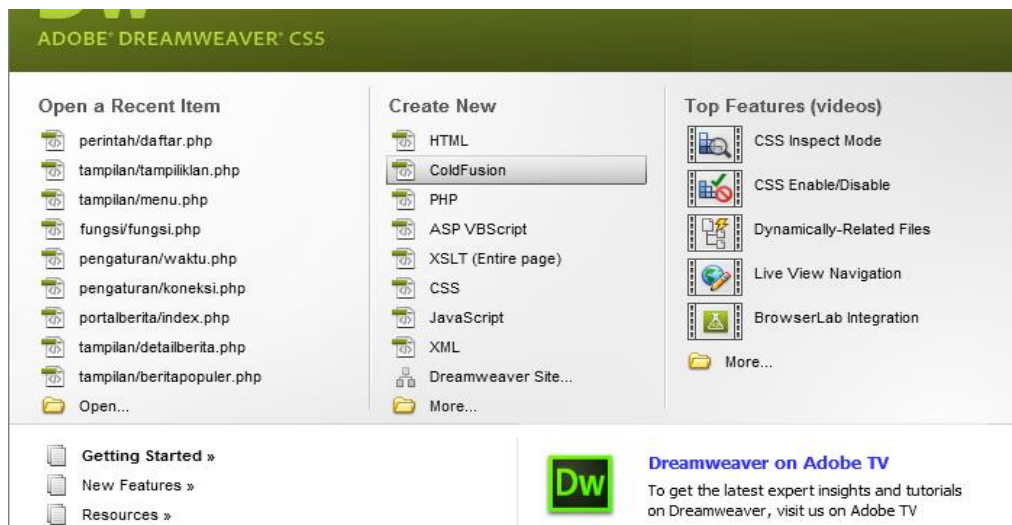
4. *Javascript*

Javascript menurut Sunyoto (2007:17) adalah “Bahasa *scripting* yang populer di *internet* dan dapat bekerja di sebagian besar *browser* populer seperti *Internet Explorer (IE)*, *Mozilla Firefox*, *Google*, *Netscape* dan *Opera*”. Kode *Javascript* dapat disisipkan dalam halaman *web* menggunakan *tag Script*.

5. *Adobe Dreamweaver CS5*

Menurut Sadeli (2011:2) “*Dreamweaver CS5* merupakan perangkat lunak *web editor* keluaran *Adobe System* yang digunakan untuk membangun dan mendesain suatu *website* dengan fitur-fitur yang menarik dan mudah untuk penggunaannya.”

Berikut tampilan Halaman Awal *Adobe Dreamweaver CS6*..



Sumber : Sadeli (2011:2)

Gambar II.2

Tampilan Adobe Dreamweaver CS6

C. Basis Data

1. Pengenalan Database

Database atau basis data adalah sekumpulan data yang memiliki hubungan secara logika dan diatur dengan susunan tertentu serta adalah representasi dari semua fakta yang ada pada dunia nyata. Definisi lebih rinci diberikan oleh *Fathansyah* (2012:1) “Basis data (*database*) dapat dibayangkan sebagai sebuah lemari jika kita memiliki sebuah lemari arsip dan bertugas untuk mengelolanya”. Adapun struktur *database* adalah *database, file* atau *table, records*, elemen data atau *field*..

2. MySQL

Menurut Anhar (2010:6) “MySQL adalah suatu perangkat lunak sistem manajemen basis data (*Database Management System*) atau DBMS yang menggunakan perintah standar SQL (*Structured Query Language*)”. Dimana MySQL mampu untuk melakukan banyak eksekusi perintah *query* dalam satu permintaan (*multithread*), baik itu menerima dan mengirimkan data. Menu MySQL digunakan untuk manajemen *database MySQL* yang ada di *server hosting/web server*. MySQL adalah pengolah *database* yang memiliki sekumpulan prosedur dan struktur sedemikian rupa sehingga mempermudah dalam menyimpan, mengatur dan menampilkan data.

3. XAMPP

Menurut Sikumbang (2006:2) “XAMPP adalah sebuah paket instalasi mudah *Apache Server* yang termasuk di dalamnya *MySQL*, *PHP*, dan perlu menjalankan *CMS* secara *live*”. Yang dimaksud secara *live* dengan semua fungsi yang dapat berjalan atau disimulasikan dengan sempurna, setidaknya membutuhkan sebuah komputer dengan sistem operasi yang di dalamnya telah terinstal aplikasi *Apache Server*, *PHP* dan *MySQL*. Aplikasi inilah yang akan mengolah bahasa-bahasa pemrograman dari *CMS* yang telah dibuat kemudian menerjemahkannya ke dalam sebuah *output* yang nantinya akan dikenal oleh *browser* untuk kemudian ditampilkan ke dalam sebuah tampilan atau *layout* di dalam *browser*. Perintah-perintah SQL yang sering digunakan untuk kebutuhan *web database* diantaranya :

a. *SELECT*

Digunakan untuk melihat dari data satu atau beberapa tabel :

SELECT kolom-kolom;

FROM Nama tabel;

Untuk melihat seluruh isi kolom dari suatu tabel digunakan *query SELECT** :

*SELECT * FROM* nama tabel;

b. *WHERE*

Digunakan untuk menyaring hasil *query* sehingga *record* yang dikeluarkan hanyalah *record* yang sesuai dengan yang diinginkan.

SELECT kolom 1, kolom 2

FROM kolom 1 *WHERE* kolom 2 < criteria;

c. *INSERT*

Digunakan untuk mengisi data atau menambahkan *record* pada suatu tabel.

INSERT INTO nama –tabel (kolom 1, kolom 2..) *VALUES* (nilai 1, nilai 2..);

d. *BETWEEN*

Digunakan untuk membatasi suatu kolom berada pada suatu batas nilai tertentu.

SELECT kolom 1, kolom 2, kolom 3

FROM kolom 1 *WHERE* kolom 2 *BETWEEN..AND..*;

e. *DELETE*

Dapat digunakan untuk menghilangkan *record-record* yang sama.

DELETE FROM nama – tabel *WHERE* criteria;

SELECT DISTINCT kolom 2 *FROM* kolom 1;

f. *DELETE*

Digunakan untuk menghapus suatu *record* dengan criteria tertentu.

DELETE FROM nama-tabel *WHERE* kriteria;

Untuk menghapus *record* pada suatu tabel, digunakan perintah *DELETE* tanpa menentukan krtierianya.

DELETE FROM nama-tabel;

g. *UPDATE*

Digunakan untuk memodifikasi nilai kolom dari suatu *record*.

UPDATE nama-tabel

SET nama – kolom= nilai – baru 1, nilai – kolom 2 = nilai – baru 2,...

WHERE kriteria;

h. *LIKE*

Digunakan untuk pencarian data yang memiliki pola terntentu.

SELECT kolom 1, kolom 2

FROM kolom 1, kolom 2

WHERE kolom 1 *LIKE* 'A%';

i. *ORDER BY*

Digunakan untuk mensortir data hasil *query* sesuai dengan kebutuhan.*SELECT*

kolom 1, kolom 2

2. Fungsi *PHP* dan *MySQL*

Berikut ini fungsi *PHP* untuk mengakses *MySQL* yang biasa digunakan diantaranya adalah :

a. *mysql_connect()*

Fungsi *mysql_connect* adalah untuk menghubungkan *PHP* dengan *database MySQL*. Format fungsinya adalah :

mysql_connect(string hostname, string username, string password);

b. *mysql_select_db()*

Setelah terhubung ke *database MySQL* dengan menggunakan *mysql_connect*, langkah selanjutnya adalah memilih *database* yang akan digunakan. Fungsi *mysql_select_db* digunakan untuk memilih *database*. Format fungsinya adalah :

mysql_select_db (string database, koneksi);

c. *mysql_query*

Kegunaan dari fungsi ini adalah untuk menghitung jumlah baris yang dikenai oleh proses *SQL*. Format fungsinya adalah :

int mysql_num_rows (string query,int [link_identifier]);

d. *mysql_num_rows*

Kegunaan dari fungsi ini adalah untuk menghitung jumlah baris yang dikenai oleh proses *SQL*. Format penulisannya adalah :

int mysql_num_rows (int result);

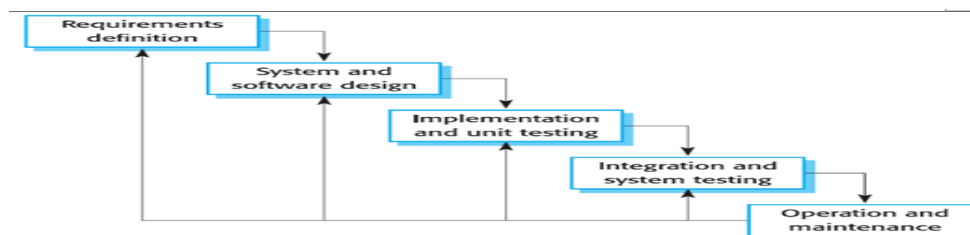
e. *mysql_fetch_array*

Fungsi ini berkaitan dengan menampilkan data, digunakan fungsi *mysql_fetch_array*. Dengan fungsi ini, hasil *query* ditampung dalam bentuk *array*. Format fungsinya adalah :

array_mysql_array (int [result_type];

D. Model Pengembangan Sistem Perangkat Lunak

Menurut (sommerville, 2007), model waterfall adalah model klasik yang bersifat sistematis, berurutan dalam membangun software. Nama model ini sebenarnya adalah “Linear Sequential Model”. Model ini sering disebut dengan “classic life cycle” atau model waterfall. Model ini termasuk kedalam model generic pada rekayasa perangkat lunak dan pertama kali diperkenalkan oleh Winston Royce sekitar tahun 1970 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai didalam Software Engineering (SE). Model ini melakukan pendekatan secara sistematis dan berurutan. Disebut dengan waterfall karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan.



Sumber: Metode Waterfall sommerville (2007),

Gambar II.3

Waterfall Model

Tahap utama model ini dibagi kedalam 4 bagian berdasarkan pengembangan kegiatannya :

1. *Requirements Definition* : Layanan, batasan dan tujuan dari sistem ditetapkan melalui konsultasi dengan penggunaan sistem. Semua itu didefinisikan secara *detail* dan bertindak sebagai spesifikasi sistem.
2. *System and Software Design* : Proses desain sistem membagi kebutuhan menjadi *hardware* dan *software*. Ini menetapkan hampir seluruh perancangan sistem. Desain *software* melibatkan pengidentifikasian dan penggambaran mengenai pemisahan dasar sistem *software* dan hubungannya.
3. *Implementation and Unit Testing* : Dalam tahap ini, desain *software* adalah menyadari sebagai kumpulan program atau suatu program. *Unit testing* melibatkan verifikasi bahwa setiap *unit* telah mencapai spesifikasi.
4. *Integration and Sistem Testing* : Suatu program atau kumpulan program diintegrasikan dan dites sebagai sistem yang telah selesai, untuk menjamin bahwa kebutuhan *software* telah terpenuhi. Setelah pengujian, sistem *software* dikirimkan kepada pelanggan.

Menurut Sommerville (2007:67-68) “kelebihan *waterfall* model adalah dokumentasi dihasilkan dalam setiap tahap, dan ini cocok dengan model proses *engineering*”. Komitmen harus dibuat dalam tahap awal dari proses ini, sehingga sulit menanggapi perubahan permintaan pelanggan. Oleh karena itu, *waterfall* model seharusnya hanya digunakan saat kebutuhan sangat dimengerti dan tidak mungkin berubah sama sekali dalam pengembangan sistem.

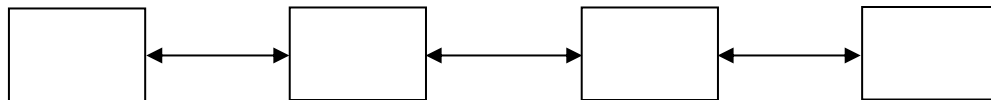
2.2. Teori Pendukung

A. Struktur Navigasi

Menurut Prihatna (2008:51) “Struktur Navigasi adalah susunan menu atau hirarki dari suatu situs yang menggambarkan isi dari setiap halaman dan *link* atau navigasi tiap halaman pada suatu situs *web*”. Bentuk dasar dari struktur navigasi yang biasa digunakan dalam proses pembuatan aplikasi multimedia ada empat macam, yaitu struktur navigasi linier, hirarki, non linier dan campuran.

1. *Linear* (Satu alur)

Linear atau 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. Salah satu yang terpenting dari struktur ini adalah tidak diperkenankan terjadinya percabangan.

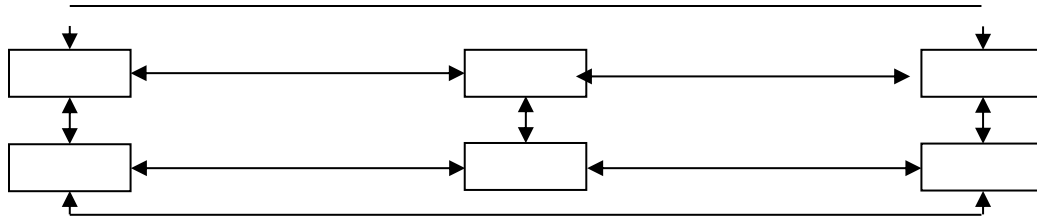


Sumber : Prihatna (2008:51)

Gambar II.4
Struktur Navigasi *linear*

2. *Hierarchical* (Hirarki)

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



isi.Sumber : : Prihatna (2008:51)

Gambar II.5

Struktur Navigasi *Hierarchical* (Hirarki)

3. *Non Linear* (Tidak Berurut)

Struktur navigasi *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.

B. *Enterprise Relationship Diagram (ERD)*

Menurut Brady dan Loonam (2010:67), “*Entity Relationship diagram (ERD)* merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh Sistem Analis dalam tahap analisis persyaratan proyek pengembangan sistem”. Sementara seolah-olah teknik diagram atau alat peraga memberikan dasar untuk desain *database* relasional yang mendasari sistem informasi yang dikembangkan. *ERD* bersama-sama dengan *detail* pendukung merupakan model data yang pada gilirannya digunakan sebagai spesifikasi untuk *database*.

Terdapat 3 komponen yang akan dibentuk yaitu :

1. Entitas

Pada *post* sebelumnya mengenai basis data telah dijelaskan sedikit tentang pengertian *entity* (entitas) yaitu suatu objek yang dapat dibedakan dari lainnya yang dapat diwujudkan dalam basis data. Pengertian lainnya menurut Brady dan Loonam (2010:67),” Entitas adalah objek yang menari di bidang organisasi yang dimodelkan”.

Contoh : Mahasiswa, Kartu Anggota Perpustakaan (KAP), dan Buku.

2. Relasi

Relasi adalah hubungan antara dua jenis entitas dan dipresentasikan sebagai garis lurus yang menghubungkan dua entitas.

Contoh : Mahasiswa mendaftar sebagai anggota perpustakaan (KAP), relasinya adalah mendaftar.

3. Atribut

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

Derajat Relasi atau kardinalitas rasio

Menjelaskan jumlah maksimum hubungan antara satu entitas dengan entitas lainnya. Beberapa relasi dalam *ERD* adalah :

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

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

b. *One to Many* (1:M)

Setiap anggota entitas A dapat berhubungan 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.

C. ***Logical Relational Strucure (LRS)***

Menurut Saputra (2012:60) mengatakan bahwa “*Logical Record Structure* (LRS) merupakan suatu hubungan yang menghubungkan antara tabel satu dengan tabel lain. dibentuk dengan nomor tipe *record*”. *Logical Record Structure* (LRS) representasi dari struktur dari *record-record* pada tabel-tabel yang terbentuk dari hasil relasi antara himpunan entitas. Beberapa tipe *record* digambarkan oleh kotak empat persegi panjang dan dengan nama yang unik.

Perbedaan *LRS* dengan *ERD* dan tipe *record* berada diluar *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 type record*. Penggambaran *LRS* mulai dengan menggunakan model yang dimengerti. Dua metode yang dapat digunakan, dimulai dengan hubungan kedua model yang dapat dikonversikan ke *LRS*. Metode yang lain dimulai dengan *Entity Relationship Diagram* dan langsung dikonversikan ke *LRS*.

Berikut tahapan transformasi *ERD* ke *LRS* menurut Wulandari (2013:15-16)

1. Konversi *ERD* ke *LRS*, *Entity Relationship Diagram* harus diubah ke bentuk *LRS* (struktur *record* secara *logic*). Dari bentuk *LRS* inilah yang nantinya dapat ditransformasikan ke bentuk relasi tabel.
2. Konversi *ERD* ke *LRS* sebuah model sistem yang digambarkan dengan sebuah model sistem yang digambarkan dengan sebuah *ERD* akan mengikuti pola pemodelan tertentu. Dalam kaitannya dengan konversi ke *LRS*, untuk perubahan yang terjadi adalah mengikuti aturan-aturan berikut:
 - a. Setiap entitas diubah ke bentuk kotak dengan nama entitas, berada diluar kotak dan atribut berada didalam kotak.
 - b. Sebuah *relationship* kadang disatukan, dalam sebuah kotak bersama entitas, kadang sebuah kotak bersama-sama dengan entitas, kadang disatukan dalam sebuah kotak tersendiri.
 - c. Konversi *LRS* ke relasi tabel atau tabel adalah bentuk pernyataan data

secara grafis dimensi, yang terdiri dari kolom dan baris. Relasi adalah bentuk visual bentuk lingkaran *Entity Relationship Diagram* dikenal dengan sebutan atribut. Konversi dari *logical structure*, dilakukan dengan cara:

- 1) Nama *logical record structure* menjadi nama relasi.
- 2) Tiap atribut menjadi sebuah kolom didalam relasi.

D. Pengujian Web (*Black Box Testing*)

Menurut *Pressman* (2010:14) “*Black Box Testing* (pengujian kotak hitam) atau disebut juga *Behavioral Testing*, berfokus pada persyaratan fungsional dari perangkat lunak”. Artinya teknik *Black Box Testing* memungkinkan untuk mendapat set kondisi masukan yang sepenuhnya akan melaksanakan semua persyaratan fungsional untuk suatu program. *Black Box Testing* bukan merupakan *alternatif* dari pengujian *White Box Testing*. Sebaliknya, *Black Box Testing* adalah pendekatan komplementer yang mungkin untuk mengungkap kelas yang berbeda dari kesalahan metode *White Box Testing*.

Black Box Testing mencoba untuk menemukan kesalahan dalam kategori berikut :

1. Fungsi tidak benar atau hilang.
2. Kesalahan *interface* atau antarmuka.
3. Kesalahan dalam struktur data atau akses *database eksternal*.
4. Kesalahan kinerja atau perilaku.
5. Kesalahan inisialisasi dan terminasi.

Tidak seperti *White Box Testing* yang dilakukan pada awal proses pengujian, *Black Box Testing* cenderung diterapkan pada tahap selanjutnya dari pengujian.

Karena *Black Box Testing* sengaja mengabaikan struktur control perhatian difokuskan pada domain informasi.