

BAB II

LANDASAN TEORI

2.1. Konsep Dasar *Web*

Sebelum masuk kedalam pembahasan mengenai spesifikasi rancangan program, penulis juga akan mengemukakan teori perancangan programnya. Dalam perancangan suatu *website* kita harus melakukan beberapa tahap persiapan untuk mencapai hasil yang baik.

Langkah-langkah persiapan dalam membuat rancangan *website* menurut Sadeli (2011:21) adalah:

1. Penguasaan dan pemahaman ruang lingkup tentang *website* yang akan dibuat.
2. Setelah mengetahui dan memahami *website* yang akan dibuat. Tahap selanjutnya adalah membuat rancangan desain atau *draft website*. Tahap ini diperlukan untuk menyeimbangkan antara desain dengan *website* yang akan dibuat.
3. Selanjutnya adalah mempersiapkan gambar. Gambar disini maksudnya adalah gambar yang akan dipergunakan dalam membuat tampilan *website*, misalkan *header*, *background*, ataupun *icon-icon* gambar.
4. Kemudian membuat *interface website* dengan format HTML menggunakan *web editor*.

5. Kemudian merancang dan membuat *database* serta tabel yang akan digunakan.
6. Selanjutnya memberi *script* PHP pada *layout* yang telah dirancang sebelumnya. Supaya situs yang akan dibuat lebih interaktif.
7. Apabila proses pembuatan situs *website* telah selesai, tahap terakhir adalah mencari tempat *hosting*. Akan tetapi pada tahap terakhir ini penulis akan membatasi hanya sampai tingkat *localhost* saja.

A. Website

a. Pengertian Internet

Menurut Shalahuddin dan Rossa (2008:3),” *internet* atau *internetwork* adalah sekumpulan jaringan yang berbeda yang saling berhubungan bersama sebagai salah satu kesatuan dengan menggunakan berbagai macam *protocol* salah satunya adalah *protocol* TCP/IP (*Transmission Control Protocol/Internet Protocol*)”.

TCP/IP adalah *protocol* yang paling banyak di gunakan di *internet*. *Protocol* TCP/IP merupakan cara standar untuk memaketkan dan mengalamatkan data komputer sehingga data tersebut dapat di kirim ke komputer terdekat atau keliling dunia dan tiba dalam waktu yang cepat tanpa rusak dan hilang.

b. Pengertian Website

Menurut Hidayat (2010:2) *website* merupakan Situs dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, *videodan* atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing–masing dihubungkan dengan jaringan–jaringan halaman (*hyperlink*).

Menurut Simarmata,(2010:51), “*Web* dapat diartikan sebagai alat untuk menciptakan sistem informasi global yang mudah berdasarkan pada hiperteks”. Dapat disimpulkan *Website* adalah sebuah tempat di *Internet*, yang menyajikan informasi dengan berbagai macam format data seperti *text*, image, bahkan video dan dapat diakses menggunakan berbagai aplikasi klien sehingga memungkinkan penyajian informasi yang lebih menarik dan dinamis dengan pengelolaan yang terorganisasi.

c. Aplikasi Berbasis Web

Menurut Oetomo (2007:95), Aplikasi berbasis *web* adalah aplikasi yang dapat diakses melalui *internet* atau *intranet*, dan pada sekarang ini ternyata lebih banyak dan lebih luas dalam pemakaiannya dan lebih mudah digunakan karena tidak harus di *install* dulu di komputer atau aplikasi yang bisa langsung dijalankan dengan menggunakan *Web Browser*. *Web browser* dapat diartikan sebagai pelaku penjelajahan *internet* atau alat yang digunakan untuk menjelajahi *internet* bagi pengguna *internet*.

Suatu perangkat lunak yang digunakan untuk menampilkan halaman-halaman *website* yang ada di *internet*. contoh :*Mozila Firefox, Internet Explore, Netscape,*

Safari, *Google Chrome* dan lain-lain. Banyak dari perusahaan-perusahaan berkembang yang menggunakan Aplikasi Berbasis *Web* dalam merencanakan sumber daya mereka dan untuk mengelola perusahaan mereka.

Salah satu jenis aplikasi *web* menurut Simarmata (2010:61) adalah: aplikasi yang berhadapan dengan pelanggan-penyalur dikenal sebagai situs B2B (BisniskeBisnis) dan menggunakan ekstranet, perluasan dari *internet* yang memungkinkan perusahaan disisi luar untuk bekerja diruangan yang terlindungi kata sandi. Situs B2B menyediakan suatu jaminan untuk berbagi informasi yang terpilih. Salah satu contohnya adalah perangkat lunak rantai persediaan yang memungkinkan semua penyalur untuk melihat permintaan dan menginventarisasikan rantai persediaan, contoh lain adalah perangkat lunak pengadaan yang memungkinkan seorang pelanggan untuk mengirimkan *request for quotation (RFQ)* dan menerima kuota atas *web*.

B. Bahasa Pemrograman

a. PHP (*Personal Home Page*)

Menurut MADCOMS (2008:1) “Bahasa pemrograman PHP adalah bahasa pemrograman yang bekerja dalam sebuah *web server*”. *Script-script* PHP yang dibuat harus tersimpan dalam sebuah *server* dan di eksekusi atau diproses dalam *server* tersebut. Penggunaan program PHP memungkinkan sebuah *website* menjadi lebih interaktif dan dinamis. Data yang dikirim oleh pengunjung *website* akan diolah dan disimpan dalam *database server* dan bisa ditampilkan kembali apabila diakses.

Berikut beberapa keunggulan yang dimiliki program PHP :

1. PHP bersifat *Open Source* sehingga diterbitkan secara gratis
2. Beberapa *server* seperti *Apache*, *MicrosoftIIS*, *AOL server* dan *Xitami* mampu menjalankan PHP.
3. Tingkat akses PHP lebih cepat serta memiliki tingkat keamanan yang tinggi.
4. Termasuk bahasa yang *embedded* (bisa ditempel atau diletakan dalam tag HTML).
5. Dapat berjalan dalam *server web* yang berbeda dan dalam system operasi yang berbeda. PHP dapat berjalan pada system operasi *Linux/Unix*, *Windows* dan *Macintosh*.

b. HTML (*Hypertext Markup Language*)

Menurut Shalahudin (2008:19), HTML atau *Hypertext Markup language* “suatu format data yang digunakan untuk membuat dokumen *hypertext* (*text* pada komputer yang memungkinkan *user* saling mengirimkan informasi)”. Dokumen HTML harus disimpan dengan *ektensi htm* atau *html*. HTML

c. JQuery

Menurut Afiana (2012:1), [JQuery adalah library javascript](#), **JQuery** berisikumpulan kode-kode/fungsi-fungsi yang siap pakai sehingga mempermudah dan mempercepat waktu kita dalam menuliskan kode javascript, karena kita hanya memanggil fungsi-fungsi yang kita butuhkan yang tentu saja sudah ada dalam *jQuery* tersebut, serta tidak perlu lagi menyusun kode *javascript* dari nol. Tentunya ini sesuai dengan slogan *jQuery* itu sendiri, yakni *Write less, do more*.

C. Basis Data

a. Mysql

Menurut Madcoms (2010:8), *MySQL* adalah sebuah basis data yang mengandung satu atau jumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel. *MySQL* termasuk jenis RDBMS (*Relational Database Management System*). Sedangkan RDMS sendiri akan lebih banyak mengenal istilah seperti tabel, baris dan kolom digunakan dalam perintah-perintah di *MYSQL*. Tipe data *MySQL* adalah data yang terdapat dalam sebuah *table* berupa *field-fild* yang berisi nilai dari data tersebut. Nilai data dalam *field* memiliki tipe sendiri-sendiri. *MySQL* pun merupakan salah satu jenis *database server* yang banyak digunakan dan sangat terkenal. Kepopulerannya disebabkan *MySQL* menggunakan *SQL (Structured Query Language)* sebagai bahasa dasar untuk mengakses *database* yang terbagi menjadi tiga bagian, yaitu DDL, DML dan DCL. *MySQL* bersifat *open source* sehingga kita bias menggunakannya secara gratis.

2.2. Teori Pendukung

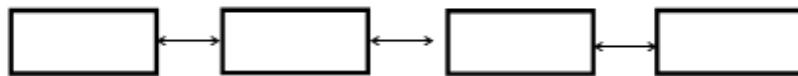
2.2.1. Struktur Navigasi

Menurut Binanto (2010:268) Menyatakan bahwa “ Struktur navigasi adalah gabungan dari struktur refrensi informasi situs *Web* dan mekanismelink yang mendukung pengunjung untuk melakukan penjelajahan situs ”.

Ada empat macam bentuk dasar dari struktur navigasi yang biasa digunakan yaitu :

1. Struktur navigasi linear

Menurut Binanto (2010:269) Pengguna akan melakukan navigasi secara beruntun, dari frame atau by terinformasi ke informasi lainnya.

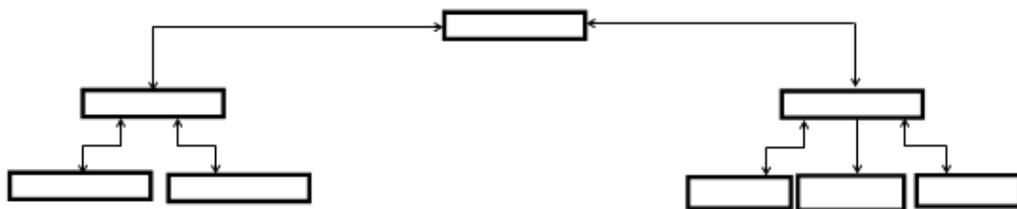


Sumber : Binanto (2010:269)

Gambar II.4. Struktur Navigasi Linear

2. Struktur Navigasi Hirarki

Menurut Binanto (2010:269) Struktur dasar ini disebut juga struktur linear dengan percabangan karena pengguna melakukan navigasi sepanjang cabang pohon struktur yang terbentuk oleh logika isi.

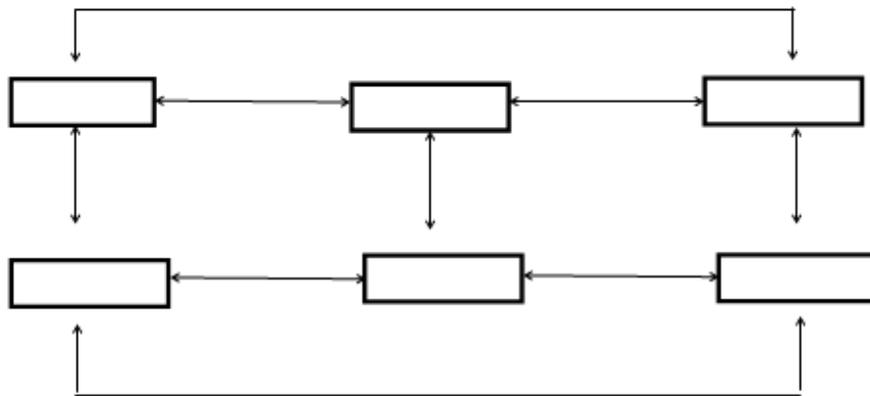


Sumber : Binanto (2010:269)

Gambar II.5. Struktur Navigasi Hirarki

3. Struktur Navigasi Non-Linear

Menurut Binanto (2010:269) Pengguna akan melakukan navigasi dengan bebas melalui isi proyek dengan tidak terkait dengan jalur yang sudah ditentukan sebelumnya.

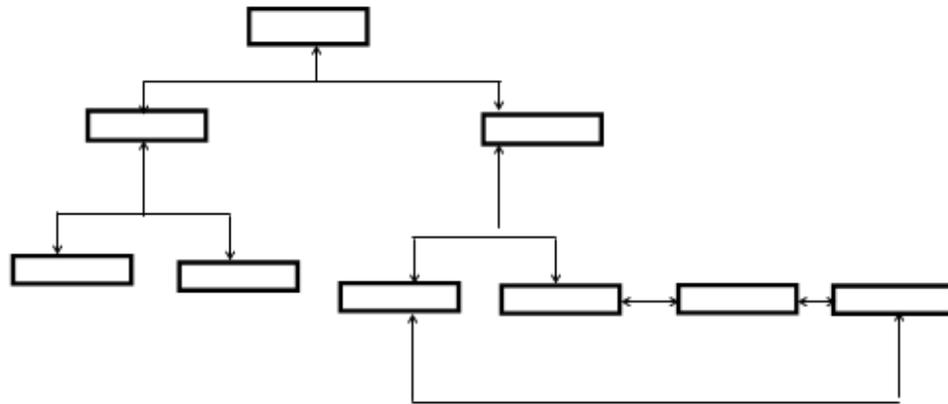


Sumber : Binanto (2010:269)

Gambar II.6. Struktur Navigasi Non-Linear

4. Struktur Navigasi Campuran (*Composite*)

Menurut Binanto (2010:269) Pengguna akan melakukan navigasi dengan bebas (secaa non linear), tetapi terkadang dibatasi presentasi linear film atau informasi penting dan atau data yang paling terorganisasi secara logis pada suatu hirarki.



Sumber : Binanto (2010:269)

Gambar II.7. Struktur Navigasi Campuran (*Composite*)

2.2.2. ERD (*Enterprise Relationship Diagram*)

1. Definisi ERD (*Enterprise Relationship Diagram*)

Menurut Winarko (2007:13) “ ERD adalah alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antar entitas ”.

2. Komponen – Komponen ERD (*Enterprise Relationship Diagram*)

Pada model Data Relation hubungan antara file direlasikan dengan *relation key* yang meruoakan kunci utama dari masing-masing *file*, adapun komponen utama dari ERD (*Enterprise Relationship Diagram*) adalah :

a. Entitas (*Entity*)

Entitas adalah sesuatu yang nyata atau abstrak di mana kita akan menyimpan data. Ada 4 kelas entitas, yaitu pegawai, pembayaran, kampus, dan buku.

Sebagai contoh suatu entitas disebut instansi, misalnya pegawai Adi, pembayaran Joko, dan lain sebagainya.

b. Atribut (*Attribute*)

Atribut adalah ciri umum semua atau sebagian besar instansi pada entitas tertentu. Sebutan lain atribut adalah properti, elemen data, dan fiel. Misalnya nama, alamat, nomor pegawai, dan gaji adalah atribut entitas pegawai.

c. Relasi (*Relationship*)

Relasi adalah hubungan alamiah yang terjadi anantara satu atau lebih entitas, misalnya proses pembayaran pegawai. Kardinalitas menentukan kejadian suatu entitas untuk satu kejadian pada entitas yang berhubungan. Misalnya, mahasiswa bisa mengambil banyak mata kuliah.

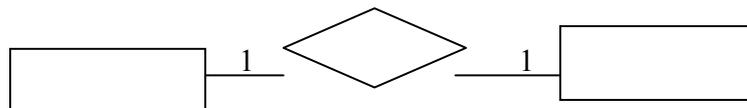
d. Asosiasi (*Association*)

Asosiasi adalah penghubung antara relasi dan entitas dimana dikedua ujungnya memiliki *Multiplicity* kemungkinan jumlah pemakaian

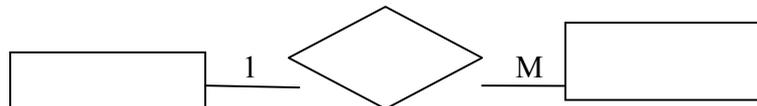
3. LRS (*Logical Record Structure*)

Menurut Husgiani dan Shidiq (2016:608) “ LRS adalah sebuah model sistem yang digambarkan dengan sebuah diagram-ER akan mengikuti pola atau aturan pemodelan tertentu dalam kaitannya dengan konversi ke LRS”.

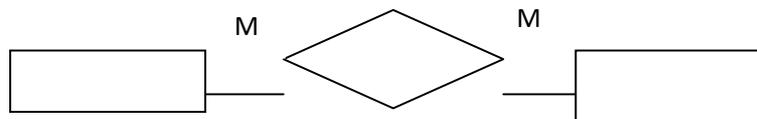
One to one



One to Many



Many to Many



Sumber: Husgiani dan Shidiq (2016:608)

Gambar II.8 Relasi ERD (*Enterprise Relationship Diagram*)

Hubungan relasi dalam ERD (*Enterprise Relationship Diagram*) Menurut Husgiani dan Shidiq (2016:608), yaitu :

- a. ***One to one***: Hubungan antara *file* yang satu dengan *file* yang kedua adalah satu berbanding satu.
- b. ***One to many*** : Hubungan antara *file* yang satu dengan *file* yang kedua adalah satu berbanding banyak atau banyak berbanding satu.

- c. *Many to many* : Hubungan antara *file* yang satu dengan *file* yang kedua adalah banyak berbanding banyak

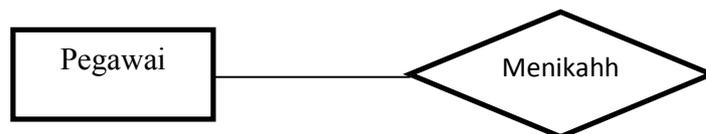
4. *Relationship Degree*

Menurut Ladjamudin (2007:191) menyatakan bahwa “ *Relationship degree* atau derajat *relationship* adalah entitas yang berpartisipasi dalam satu *relationship* ”.

Derajat *relationship* yang sering dipakai didalam ERD sebagai berikut :

a. *Unray relationship*

Menurut Ladjamudin (2007:192) *Unray Relationship* adalah model *relationship* yang terjadi diantara *entity* yang berasal dari *entity set* yang sama. Sering disebut juga sebagai *Recursive Relationship* atau *Reflective Relationship*.



Sumber: Ladjamudin (2007:192)

Gambar II.9 Diagram *Unary Relationship*

b. Binary Relationship

Menurut Ladjamudin (2007:192) *Binary Relationship* adalah model *r* *relationship* antara *instance-instance* dari suatu tipe entitas (dua *entity* yang

berasal dari *entity* yang sama). *Relationship* ini paling umum digunakan dalam pembuatan model data. Gambar dibawah ini menunjukkan bahwa *relationship* bekerja untuk merupakan *relationship* banyak ke satu.

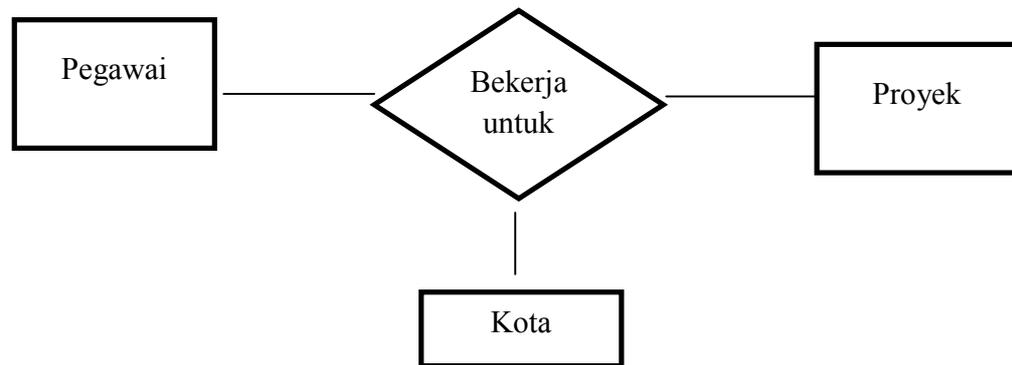


Sumber: Ladjamudin(2007:192)

Gambar II.10 Diagram *Binary Relationship*

c. Ternary Relationship

Menurut Ladjamudin (2007:193) *Ternary Relationship* merupakan *relationship* antara *instance-instance* dari tiga tipe entitas secara serentak. Perlu dicatat bahwa *relationship ternary* tidak sama dengan tiga *relationship binary*.



Sumber: Ladjamudin(2007:193)

Gambar II.11 Diagram Ternary Relationship

2.2.3. Pengujian Web

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

Menurut Rosa A.S (2013:275) Menyatakan bahwa *Black Box Testing* 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. Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah, misalkan untuk kasus proses login maka kasus uji yang dibuat adalah :

- a. Jika *user* memasukan nama pemakai (*username*) dan kata sandi (*password*) yang benar
- b. Jika *user* memasukan nama pemakai (*username*) dan kata sandi (*password*) yang salah, misalnya nama pemakai benar tapi kata sandi salah, atau sebaliknya, atau keduanya salah.

Menurut Al Fatta (2007:172) menjelaskan bahwa *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 di inginkan.

Metode ujicoba *black box testing* memfokuskan pada keperluan fungsional. Karena itu ujicoba *black box testing* memungkinkan pengembangan program untuk membuat himpunan kondisi input yang akan melatih syarat-syarat fungsional atau program.