

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Web

Web adalah dokumen yang ditulis dalam format HTML (*Hyper Text Markup Language*), yang hampir selalu bisa diakses melalui HTTP, yaitu protokol yang menyampaikan informasi dari *server website* untuk ditampilkan kepada para pemakai melalui *web browser*. Semua publikasi dari *website-website* tersebut dapat membentuk sebuah jaringan informasi yang sangat besar.(Abdullah, 2015).

A. Website

Website dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang di seluruh dunia.(Abdulloh, 2018).

Adapun aplikasi yang membantu dalam merancang, menuliskan skrip program, maupun menampilkan hasil program menurut (Abdulloh, 2018), adalah sebagai berikut:

1. Text Editor

Text Editor memiliki fungsi untuk mengetikkan skrip program dan untuk menampilkan hasilnya memerlukan *web browser*. *Text Editor* merupakan aplikasi yang ringan dan *open source*. Yang termasuk kedalam aplikasi jenis ini yaitu Notepad, Notepad++, Sublime Text, Bracket dan sebagainya.

2. Web Browser

Web browser digunakan untuk menampilkan dan mengetes hasil program. Beberapa skrip CSS3 dan HTML5 ada yang hanya *support* pada *web*

browser tertentu, sehingga menggunakan lebih dari satu *web browser* akan lebih baik.

3. *Web Server*

Web server merupakan perangkat lunak yang berfungsi untuk menerima permintaan (*request*) melalui protokol *HTTP* atau *HTTPS* dari *client* kemudian mengirimkan kembali dalam bentuk halaman-halaman *web*. Contoh yang termasuk kedalam *web server* adalah Apache. Dalam penggunaannya, biasanya sudah menjadi satu paket dengan PHP dan MYSQL diantaranya Xampp dan Appserv.

Dalam membangun aplikasi berbasis *web* tersebut, penulis menggunakan Sublime Text sebagai *text editor*, Google Chrome sebagai *browser* dan aplikasi Xampp yang didalamnya terdapat Apache yang digunakan sebagai *web server*.

B. Bahasa Pemrograman

Bahasa pemrograman utama yang digunakan dalam membuat website agar aplikasi website menarik, responsif dan atraktif menurut (Abdulloh, 2018), ialah sebagai berikut:

1. *HTML*

HTML merupakan singkatan dari *Hypertext Markup Language* yaitu bahasa standar *web* yang dikelola penggunaannya oleh W3C (*World Wide Web Consortium*) berupa tag-tag yang menyusun setiap elemen dari *website*.

2. CSS

CSS adalah singkatan dari *Cascading Style Sheet* yaitu dokumen *web* yang berfungsi untuk mengatur elemen *HTML* dengan berbagai *property* yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan.

3. PHP

PHP adalah singkatan dari *PHP Hypertext Preprocessor* yaitu bahasa pemrograman *web* yang dapat disisipkan dalam skrip *HTML* dan bekerja disisi *server*. Pemroses data pada sisi *server* sesuai yang diminta oleh *client* menjadi informasi yang ditampilkan, juga sebagai penghubung aplikasi *web* dengan *database*.

4. JavaScript

JavaScript merupakan bahasa pemrograman *web* yang pemrosesannya dilakukan di sisi *client*. Karena berjalan di sisi *client*, *JavaScript* dapat dijalankan hanya dengan menggunakan *browser*. Berbeda dengan *PHP* yang bekerja di sisi *server*, untuk menjalankan skrip *JavaScript* tidak memerlukan *refresh* pada *browser*.

C. Basis Data

Menurut (Pamungkas, 2017) mengatakan bahwa :

Basis data merupakan suatu kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, dan dengan *software* untuk melakukan manipulasi untuk kegunaan tertentu. Basis data bisa diartikan juga sebagai sekumpulan data yang disusun dalam bentuk beberapa tabel yang saling memiliki relasi maupun berdiri sendiri.

Teori aplikasi basis data dan program:

1. XAMPP adalah salah satu paket instalasi apache, PHP, dan MySQL secara *instant* yang dapat digunakan untuk membantu proses instalasi ketiga produk tersebut. (Sitohang, 2018)
2. *MySQL*
MySQL (*My Structure Query Language*) adalah sebuah software database, yang merupakan tipe data relasional yang artinya MySQL penyimpanan datanya dalam bentuk tabel-tabel yang saling berhubungan. (Sitohang, 2018)
3. *PhpMyAdmin*
PhpMyAdmin adalah aplikasi web yang dibuat oleh phpmyadmin.net.phpMyAdmin digunakan untuk administrasi database MySQL. Untuk menggunakan phpmyadmin pada layar utama WAMP pilih phpMyAdmin). (Simangunsong, 2018)

D. Model Pengembangan Perangkat Lunak

Model pengembangan perangkat lunak air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisi, desain, pengodean, pengujian, dan tahap pendukung (*support*). (Sukamto & Shalahuddin, 2018).

2.2. Teori Pendukung (*Tool System*)

A. Struktur Navigasi

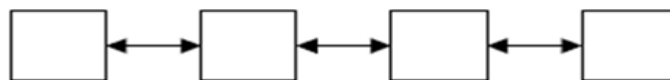
Struktur navigasi menggambarkan deretan menu yang terdapat dalam sebuah *website*, merupakan media yang menghubungkan manusia dengan komputer agar

dapat berinteraksi kedalam sistem. Dengan adanya struktur navigasi pengguna dapat dengan mudah mengoperasikan sistem.(Andriansyah, 2016).

Bentuk dari struktur navigasi sebagai berikut:

1. Struktur Navigasi *Linear*

Pengguna akan melakukan navigasi secara berurutan dan *frame* atau *byte* informasi yang satu ke yang lainnya.

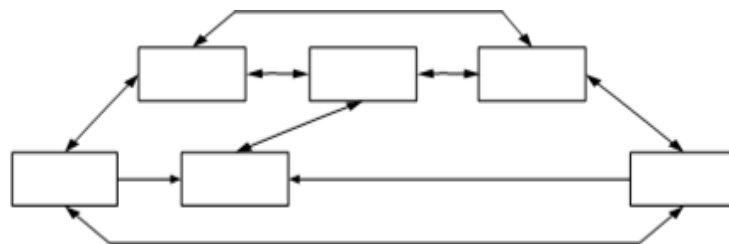


Sumber: Andriansyah

Gambar II.1 Struktur Navigasi Linier

2. Struktur Navigasi *Non-Linier*

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

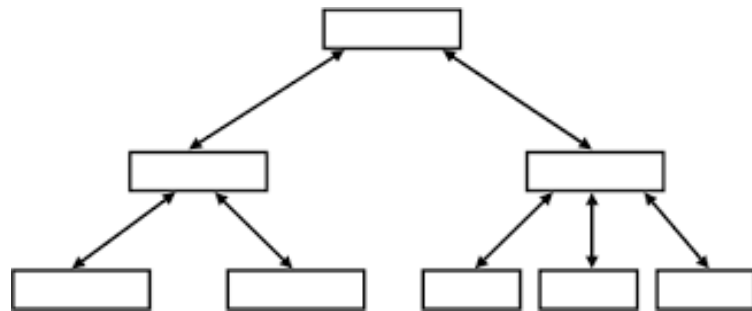


Sumber: Andriansyah

Gambar II.2 Struktur Navigasi Non-Linier

3. Struktur Navigasi *Hirarki*

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

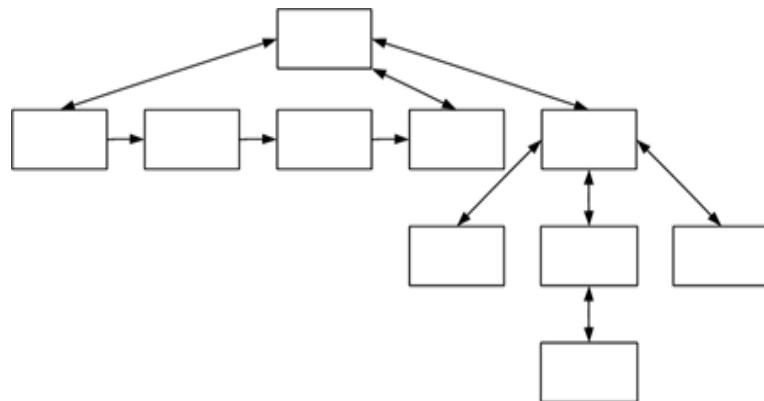


Sumber: Andriansyah

Gambar II.3 Struktur Navigasi Hirarki

4. Struktur Navigasi Campuran *(Composite)*

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



Sumber: Andriansyah

Gambar II.4 Struktur Navigasi Campuran

B. *Enterprise Relationship Diagram*

Menurut Muhammad dalam (Yanto, 2016). Mengemukakan bahwa “*Entity Relationship Diagram* adalah suatu diagram untuk menggambarkan desain konseptual dari model konseptual suatu basis data relasional. ERD juga merupakan gambaran yang merelasikan antara objek yang satu dengan objek yang lain dari objek di dunia nyata yang sering dikenal dengan hubungan antar entitas”. ERD terdiri dari 3 Komponen Utama, yaitu:

1. Entitas (*Entity*)

Entitas adalah suatu objek di dunia nyata yang dapat dibedakan dengan objek lainnya. Objek tersebut dapat berupa orang, benda ataupun hal lainnya. Entitas terbagi atas dua yaitu:

a. Entitas Kuat (*Strong Entity*)

Entitas kuat adalah entitas yang dapat berdiri sendiri tidak bergantung pada entitas lainnya, entitas kuat memiliki atribut *key* dan entitas kuat digambarkan sebagai kotak persegi panjang bergaris tunggal.

b. Entitas Lemah (*Weak Entity*)

Entitas lemah adalah entitas yang tidak dapat berdiri sendiri. Entitas lemah merupakan hasil dari pembentukan entitas kuat, entitas lemah tidak memiliki atribut *key* dan entitas lemah digambarkan kotak persegi panjang bergaris ganda.

2. Atribut (*attribute*)

Atribut merupakan semua informasi yang berkaitan dengan entitas. Atribut sering dikenal dengan *property* dari suatu entitas atau objek. Atribut digambarkan dalam bentuk lingkaran *elips*. Macam-macam atribut:

a. Atribut Sederhana (*Simple Attribute*)

Atribut sederhana adalah atribut yang nilainya tidak dapat dibagi lagi menjadi banyak atribut yang lebih kecil.

b. Atribut Komposit (*Composite attribute*)

Atribut komposit adalah atribut gabungan yang nilainya dapat dipecah menjadi bagian yang lebih kecil. Atau sering disebut atribut yang terdiri dari beberapa atribut kecil didalamnya.

c. Atribut Bernilai Tunggal (*Single values attribute*)

Atribut bernilai tunggal adalah jenis atribut yang nilainya hanya satu dari suatu entitas.

d. Atribut Bernilai Banyak (*Multivalues Attribute*)

Atribut bernilai banyak adalah jenis atribut yang nilainya lebih dari satu dalam suatu entitas tertentu.

e. Atribut Turunan (*Derived attribute*)

Atribut turunan adalah jenis atribut yang nilainya diperoleh dari atribut yang lain.

f. Atribut Identitas (*Key Attribute*)

Atribut identitas adalah atribut yang dijadikan sebagai kunci pada suatu table. Atribut identitas terdiri dari beberapa jenis yaitu:

1. *Super Key*

Super key adalah satu atribut atau kumpulan atribut yang secara unik mengidentifikasi sebuah baris didalam relasi atau himpunan dari satu atau lebih entitas yang dapat digunakan untuk mengidentifikasi secara unik sebuah entitas dalam set entitas.

2. *Candidate Key*

Candidate key adalah atribut yang menjadi determinan yang dapat dijadikan identitas dari sebuah relasi.

3. *Primary key*

Primary key adalah *kandidat key* yang dipilih untuk mengidentifikasi baris data secara unik dalam relasi.

4. *Alternative key*

Alternative key adalah *candidate key* yang tidak terpilih sebagai *primary key* atau atribut untuk menggantikan kunci utama.

5. *Foreign Key*

Foreign key adalah atribut dengan *domain* yang sama yang menjadi kunci utama sebuah relasi, tetapi pada relasi lain atribut tersebut sebagai atribut biasa.

6. *Composite Key*

Composite key adalah kunci yang terdiri dari dua atribut atau lebih. Atribut-atribut tersebut jika berdiri sendiri tidak menjadi identitas baris, tetapi bila dirangkai menjadi satu kesatuan akan dapat mengidentifikasi secara unik.

3. Tipe Relasi

Gambar belah ketupat merupakan perlambangan relasi antar entitas atau sering disebut kerelasian. Ada 2 macam penggambaran relasi yaitu relasi kuat dan relasi lemah. Relasi kuat adalah untuk menghubungkan antar entitas kuat sedangkan relasi lemah untuk menghubungkan antar entitas kuat dengan entitas lemah.

Ada tiga macam relasi menurut derajatnya, yaitu:

- a. *Unary* adalah relasi yang menghubungkan entitas yang sejenis.
- b. *Binary* adalah relasi yang menghubungkan entitas yang tidak sejenis
- c. *Ternary* adalah relasi yang menghubungkan lebih dari dua entitas yang tidak sejenis.

Menurut Kuryanti dalam (Tabrani 2014:33-41). Mengemukakan bahwa “*Logical Record Structure* dibentuk dengan nomor dari tipe *record*”. Beberapa tipe *record* digambarkan oleh kotak persegi panjang dan dengan nama yang unik. Perbedaan *LRS* dengan *E-R diagram* adalah nama 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 field-field* yang kelihatan pada kedua *link* tipe *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 *ERD* dan langsung dikonversikan ke *LRS*.

C. Implementasi dan Pengujian Web

Perangkat lunak sering mengandung kesalahan (*error*) pada proses-proses tertentu pada saat perangkat lunak sudah berada ditangan *user*. Kesalahan-kesalahan (*error*) pada perangkat lunak ini sering disebut dengan *bug*. Untuk menghindari *bug* maka diperlukan adanya pengujian perangkat lunak sebelum perangkat lunak diberikan ke pelanggan atau selama perangkat lunak masih terus dikembangkan. Pengujian yang dipakai adalah *Black-Box Testing*.

Black-Box Testing (pengujian kotak hitam)

Yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-

fungsi, masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. 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 dibuat adalah(Sukamto & Shalahuddin, 2018):

1. Jika user memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang benar
2. Jika user memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang salah, misalnya nama pemakai benar tapi kata sandi salah, atau sebaliknya, atau keduanya salah.