

## BAB II

### LANDASAN TEORI

#### 2.1 Konsep Dasar *Web*

Pembuatan *website* tidak terlepas dari berbagai macam teori-teori yang mendukung dalam mempelajari serta merancang sebuah *website*.beberapa konsep dasar *website* akan dibahas sebagai berikut:

##### A. *Website*

Menurut (Hastanti,2015), “*Website* adalah lokasi di internet yang menyajikan kumpulan informasi sehubungan dengan profil pemilik situs”. *Website* adalah suatu halaman yang memuat situs-situs *web page* yang berada di internet yang berfungsi sebagai media penyampaian informasi, komunikasi, atau transaksi

Di dalam sebuah *website*, ada beberapa komponen pendukung agar sebuah *website* dapat diakses oleh pemakai, diantaranya :

##### 1. *Internet*

Menurut (Hastanti,2015), “menjelaskan bahwa *Internet* adalah sebagai jaringan komputer yang sangat luas dan besar dan mendunia, menghubungkan pemakai komputer dari satu negara ke negara lain di seluruh dunia”, dimana di dalamnya terdapat berbagai sumber informasi dan fasilitas–fasilitas layanan internet.

##### 2. *Web Browser*

Menurut (Hastanti,2015), “menjelaskan bahwa *web browser* adalah sebuah perangkat lunak atau software yang berfungsi untuk menampilkan dan melakukan interaksi dengan dokumen-dokumen yang disediakan oleh *server web*”. Dengan *web browser* kita dapat memperoleh informasi yang disediakan oleh *server web*. *Web*

*browser* dikenal juga dengan istilah *browser*, atau peselancar, atau *Internet browser* adalah suatu program computer yang menyediakan fasilitas untuk membaca halaman *web* di suatu computer.

Dua program *web browser* yang cukup populer saat ini adalah *Microsoft Internet Explorer*, *Mozilla Firefox*, *Opera*, *Safari* dan *Netscape Navigator*. Program *browser* pertama adalah *mosaic*, yang merupakan suatu *text browser*, yang sekarang *web browser* telah berkembang ke dalam bentuk multimedia.

### 3. *Web Server*

Menurut (Hastanti,2015) mengemukakan bahwa:

*Web server* adalah *system computer* dan *software* yang menyimpan serta mendistribusikan data ke computer lain lewat *internet* yang meminta informasi tersebut. *Web Server* memberikan layanan data yang berfungsi menerima permintaan *HTTP* atau *HTTPS* dari klien yang dikenal dengan *browser web* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen *HTML*.

### 4. *XAMPP*

Menurut Abdulloh Pada (Tasiati & Corie Mei Hellyana,2017)“*XAMPP* adalah salah satu paket installer yang berisi *apache* yang merupakan *web server* tempat menyimpan file-file yang diperlukan *website*, dan *phpmyadmin* sebagai aplikasi yang digunakan untuk perancangan database *MySQL*”.

## **B. Bahasa Pemrograman**

Bahasa Pemrograman yang digunakan dalam perancangan *website* antara lain :

### 1. *Hypertext Preprocessor (PHP)*

Menurut (Hastanti,2015), “menjelaskan bahwa *PHP* adalah bahasa pemrograman skrip sederhana yang digunakan untuk pemrosesan *HTML Form* di dalam halaman *web*”. Strukturnya sangat sederhana sehingga *PHP* dapat dengan mudah dipelajari programmer pemula bahkan orang tanpa latar belakang Teknologi

Informasi. Hal inilah yang menyebabkan PHP sangat cepat populer di kalangan pengembang aplikasi *web*. Membuat program menggunakan PHP itu mudah, cukup sediakan saja sebuah program editor teks sederhana untuk menuliskan programnya, seperti Notepad (Windows) dan vi editor (Linux), atau program editor yang lebih advance, seperti EditPlus, Notepad++, atau Dreamweaver. Ekstensi file PHP yang umum digunakan adalah .php (selain itu .php3 dan .phtml).

## 2. *JavaScript* (JS)

Menurut Suryatiningsih dan Wardani pada (Kusniawan & Sardiarinto, 2016)“Javascript adalah bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML, sepanjang sejarah internet bahasa ini adalah bahasa skrip pertama untuk web.”

## 3. *Hyper Text Markup Language* (HTML)

Menurut (Prayitno & Safitri, 2015) “HTML singkatan dari Hypertext Markup Language dan berguna untuk menampilkan halaman web”.

## 4. *Cascading Style Sheet* (CSS)

Menurut (Prayitno & Safitri,2015), menerangkan bahwa “CSS merupakan bahasa pemrograman *web* yang digunakan untuk mengatur *style-style* yang ada di tag- tag HTML”.

## 5. *JQuery*

Menurut (Roki Aditama,2016:38)“Jquery adalah javascript *library* yang berisikan kode-kode javascript bertujuan untuk mempersingkat dalam penulisan kode javascript dan pastinya cepat dan ringan dalam membuat sebuah program”.

## 6. Ajax

Menurut (Rohi Abdulloh, 2017:2) “AJAX adalah singkatan dari Asynchronous JavaScript and XML merupakan solusi untuk permasalahan pada *web*

tradisional dimana ketika klien mengirim permintaan ke server harus menunggu respon dari *server*, baru bisa melanjutkan aktivitas berikutnya”.

### **C. Basis Data**

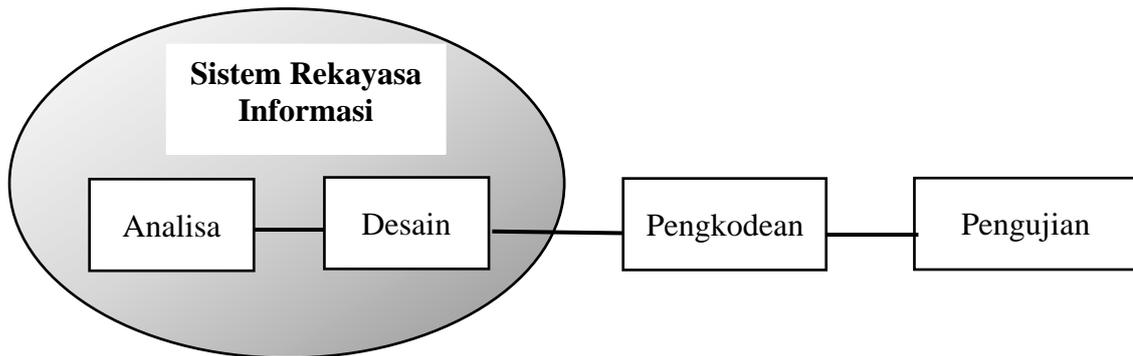
Menurut (Hidayat & Marlina, 2017), “Basis data adalah sekumpulan data yang saling berhubungan secara logis dan terorganisir dengan baik”. Basis data merupakan salah satu komponen utama pendukung program aplikasi. Hampir semua program aplikasi yang melibatkan pengolahan data dapat dipastikan menggunakan basisdata sebagai tempat penyimpanan datanya. Basis data (*database*) yang digunakan dalam pembuatan program adalah MySQL. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya.

### **D. Model Pengembangan Perangkat Lunak**

Menurut (Soekamto & Shalahuddin, 2016:28) “Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linier*) atau alur hidup klasik (*classic lifecycle*)”.

Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian dan tahap pendukung (*support*).

Berikut adalah gambar model air terjun :



Sumber : (Soekamto & Shalahuddin,2016:28)

**Gambar II.1**

**Ilustrasi model *waterfall***

1. Analisis kebutuhan perangkat lunak  
Proses pengumpulan kebutuhan dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.
2. Desain  
Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean.
3. Pembuatan kode program  
Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahanan desain.
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.

5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*.

## 2.2 *Tools Program*

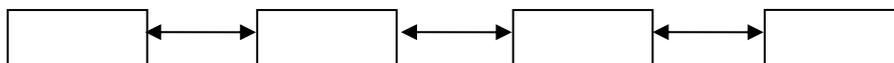
Adapun tools program yang mendukung kemudahan dalam mempelajari dan merancang program aplikasi ini sebagai berikut :

### A. Struktur Navigasi

Menurut (Andriansyah,2016) “Struktur navigasi adalah Dalam sebuah pembuatan *website*, diperlukan struktur navigasi sebagai bentuk penggambaran dari halaman-halaman apa saja yang tersedia didalam *website*”. Layaknya petunjuk arah, navigasi memudahkan pengguna untuk berselancar didalam sebuah *website*.

1. Struktur Navigasi *Linier*

Pengguna akan melakukan navigasi secara berurutan, dari *frame* atau *byte* informasi ke informasi lainnya.

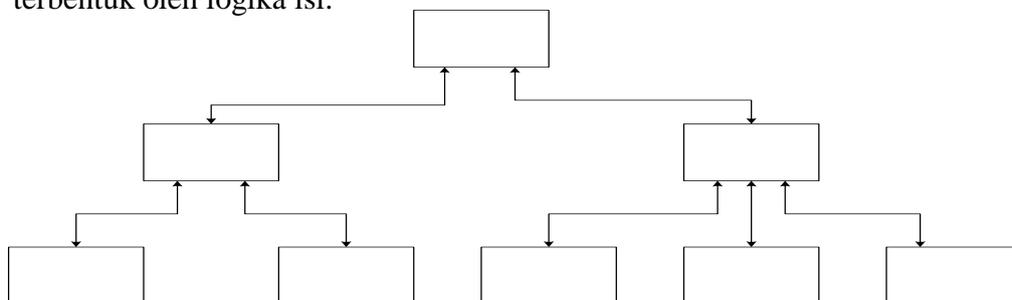


Sumber: (Andriansyah,2016)

**Gambar II.1.**  
**Struktur Navigasi *Linier***

## 2. Struktur Navigasi *Hirarki*

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

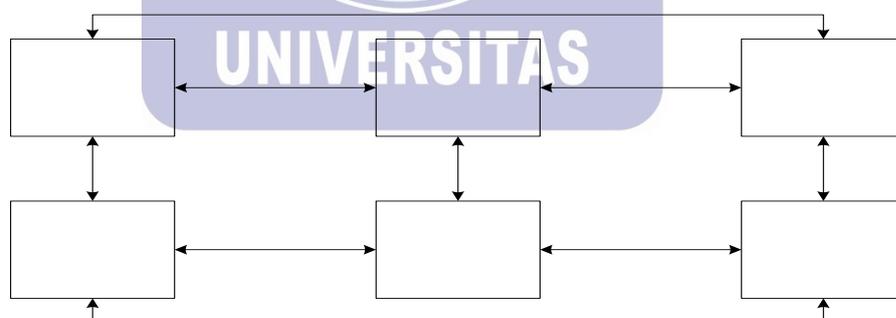


Sumber: (Andriansyah,2016)

**Gambar II.2.**  
**Struktur Navigasi *Hirarki***

## 3. Struktur Navigasi *Non-Linier*

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

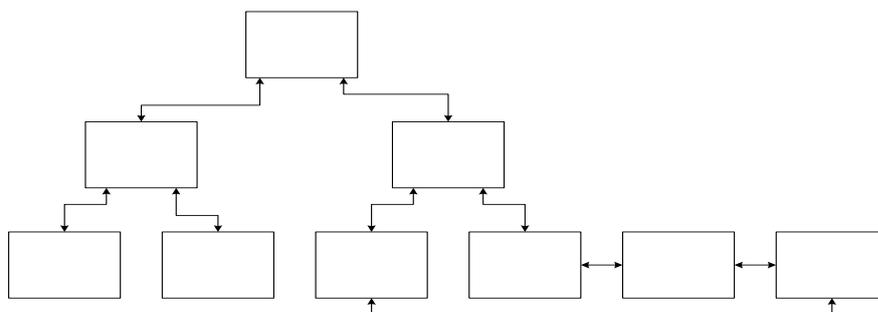


Sumber: (Andriansyah, 2016)

**Gambar II.3.**  
**Stuktur Navigasi *Non Linier***

#### 4. Struktur Navigasi Campuran (*Composite*)

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



Sumber: (Andriansyah,2016)

**Gambar II.4.**  
**Struktur Navigasi *Composite***

#### B. *Entity Relational Diagram (ERD)*

Menurut (Sulianta,2017), *Entity Relationship Diagram (ERD)* adalah pemodelan awal basis data yang akan dikembangkan berdasarkan teori himpunan dalam bidang matematika untuk pemodelan basis data relasional”. ERD ini dibentuk berdasarkan tiga komponen (Sulianta, 2017) yaitu:

##### 1. Entitas

Entitas adalah objek dalam bentuk fisik atau konse. Entitas ini akan dibuat unik atau berbeda dengan entitas lainnya, contohnya entitas mahasiswa, dosen, mata kuliah dan lain sebagainya. Entitas dapat digolongkan menjadi, entitas kuat dan entitas lemah. Entitas kuat merupakan entitas yang keberadaannya dapat berdiri sendiri tanpa tergantung dengan entitas lain. Sebagai contoh entitas kuat, misalnya entitas “orang tua”, sedangkan entitas lemah adalah anak. Dalam hal ini

entitas “orang tua” dapat berdiri sendiri, sedangkan entitas anak tergantung dari entitas lainnya, dalam hal itu entitas “orang tua”.

## 2. Atribut

Atribut adalah karakteristik atau properti dari entitas, contohnya entitas mahasiswa memiliki atribut Nomor Induk Mahasiswa (NIM), nama, alamat, jenis kelamin, hobi dan lain-lain. Atribut dibagi menjadi beberapa tipe, ada yang dinamakan dengan atribut kunci, atribut bernilai banyak, atribut komposit dan atribut derivative. Atribut kunci adalah atribut yang bernilai unik dan berbeda dengan lainnya, misalnya entitas mahasiswa memiliki atribut NIM, atribut ini akan berbeda nilainya antara satu mahasiswa dengan mahasiswa lainnya. Untuk menandakan atribut kunci biasanya menggunakan garis bawah (*underline*) pada penamaan atributnya.

Atribut bernilai tunggal adalah atribut yang hanya memiliki satu buah nilai, misalnya nama, alamat, dan jenis kelamin. Masing-masing atribut tersebut nilainya hanya boleh satu. Atribut bernilai banyak adalah atribut yang memiliki nilai lebih dari satu. Atribut komposit adalah atribut yang terdiri beberapa atribut yang lebih kecil, contohnya nama jalan, nomor rumah, dll. Sedangkan atribut derivative adalah atribut yang nilainya berasal dari atribut lain misalnya umur, yaitu tanggal lahir dikurangi dengan tanggal saat ini.

## 3. Relasi

Relasi adalah hubungan antara entitas yang satu dengan yang lainnya, misalnya terdapat hubungan antara entitas mahasiswa dengan mata kuliah karena mahasiswa nantinya akan mengambil mata kuliah.

#### 4. Kardinalitas

Derajat relationship atau yang sering disebut kardinalitas antara lain sebagai berikut:

- 1) Satu-ke-satu (*one-to-one*), setiap satu entitas hanya dapat berhubungan dengan satu entitas yang lain.
- 2) Satu-ke-banyak (*one-to-many*), setiap satu entitas dapat berhubungan dengan lebih dari satu entitas tapi tidak sebaliknya.
- 3) Banyak-ke-banyak (*many-to-many*), setiap entitas dapat berhubungan dengan banyak entitas lain dan demikian pula sebaliknya.

#### C. *Logical Record Structure (LRS)*

*Logical Record Structure (LRS)* merupakan sebuah model sistem yang digambarkan dengan menggunakan sebuah ERD yang akan mengikuti pola/aturan pemodelan tertentu dalam kaitannya dan dikonversi ke LRS, maka perubahan yang terjadi adalah mengikuti aturan-aturan sebagai berikut: setiap entitas akan diubah ke dalam bentuk simbol kotak, sebuah atribut relasi disatukan dalam sebuah kotak bersama entitas jika hubungan yang terjadi pada ERD 1:M (*one-to-many*) atau tingkat hubungan 1:1 (*one-to-one*), sebuah relasi dipisah dalam sebuah kotak tersendiri (menjadi entitas baru) jika tingkat hubungannya M:M (*many to many*) dan memiliki *foreign key* sebagai *primary key* yang diambil dari kedua entitas yang sebelumnya saling berhubungan menurut Hasugian dan Shidiq dalam (Tasiati & Corie Mei Hellyana, 2017: 68).

#### D. *Pengujian Web (Black Box Testing)*

*Black Box Testing* berfokus pada spesifikasi fungsional dari perangkat lunak. Tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada

spesifikasi fungsional program. *Black Box Testing* bukanlah solusi alternatif dari *White Box Testing* tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh *White Box Testing*. *Black Box Testing* cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (*performance errors*).

