

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Konsep Dasar Web**

Dunia internet berkembang pesat khususnya pada masyarakat modern saat ini. Internet menyediakan berbagai informasi, berita dan ilmu pengetahuan. Dalam mencari informasi dari internet, pengguna akan menuju ke sebuah alamat unik internet yang disebut nama domain dan menemukan gambar diam, gambar bergerak, suara, informasi berbentuk teks serta video dalam sebuah media yang biasa disebut *website* atau situs. *Website* ini biasa dibuka melalui sebuah program penjelajah (*Browser*) yang berada di komputer. Program penjelajah yang digunakan dalam komputer diantaranya : IE (*Internet Explorer*), Mozilla Firefox dan Opera.

Pada bab ini akan di jelaskan tentang *web* yang penulis kutip menurut beberapa para ahli sebagai berikut:

##### **2.1.1. Website**

Khusus di dunia pendidikan, Istilah ini akan sering kita dengar pada saat menulis sebuah makalah, Jurnal ataupun karya ilmiah. Penulis akan mencoba menguraikan berbagai pendapat menurut para ahli tentang pengertian Web.

Menurut Hidayat (2010:2) mengemukakan bahwa:

*Website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

Suatu sistem untuk menampilkan gambar, teks, multimedia dan lainnya pada jaringan internet yang berkaitan dengan dokumen sebagai media nya adalah *Website* (Sibero, 2013:11).

Kesimpulan dari pengertian di atas dapat dijelaskan bahwa *Website* adalah kumpulan halaman informasi yang dapat diakses lewat jaringan internet dengan melalui program penjelajah *browser*.

Untuk melihat halaman *website* diperlukan beberapa *software* pendukung, diantaranya:

1. Internet

Melihat sejarah internet dan jaringan komputer adalah hasil dari ARPANET, sebuah proyek riset tingkat tinggi yang dimiliki oleh Departemen Pertahanan Amerika Serikat. Berikut pengertian Internet menurut para ahli.

Menurut Hidayatullah dan Kawistara dalam bukunya tentang Pemrograman *Web* (2017:1), “Internet adalah jaringan global yang menghubungkan komputer-komputer di seluruh dunia”. Dengan internet sebuah komputer bisa mengakses data yang terdapat pada komputer lain di benua yang berbeda.

Sedangkan menurut Sibero (2013:10) mengatakan bahwa “Internet (*Interconnected Network*) adalah jaringan komputer yang menghubungkan antar jaringan secara global, internet dapat disebut jaringan dalam suatu jaringan yang luas”.

Pengertian di atas disimpulkan bahwa Internet adalah saling terhubungnya antara Jaringan komputer satu dengan yang lainnya untuk keperluan komunikasi dan informasi.

## 2. *Web Server*

Secara bentuk fisik dan cara kerjanya, perangkat keras *web server* tidak berbeda dengan komputer rumah atau PC, yang membedakan adalah kapasitas dan kapabilitasnya. *Web server* bekerja sebagai penyedia layanan yang dapat diakses oleh banyak pengguna, itulah yang menjadikan perbedaan karena membutuhkan kapasitas dan kapabilitas yang besar dibandingkan PC. Berikut penjelasan menurut para ahli tentang *Web server* :

Menurut Sibero (2013:11) dalam bukunya tentang *Web Programming Power Pack* menjelaskan bahwa “*Web server* adalah sebuah komputer yang terdiri dari perangkat keras dan perangkat lunak”. Setiap perangkat lunak *web server* memiliki karakteristik dan teknologi yang digunakan untuk mengatur kerja sistemnya.

Perangkat lunak yang hasilnya nanti akan dikembalikan ke *browser* sebelumnya harus mengelola permintaan *user* terlebih dahulu dari *brower* adalah pengertian dari *Web server* (Supardi, 2010:2).

Pengertian di atas disimpulkan bahwa untuk menguji apakah aplikasi *web* anda berjalan dengan baik atau tidak maka diperlukan *web server*.

## 3. *Web Browser*

Menggunakan *web browser* maka web akan dapat digunakan sebagai dokumen yang dapat dipertukarkan. Berikut pengertian *web browser* menurut para ahli.

*Web browser* adalah aplikasi perangkat lunak yang digunakan untuk mengambil dan menyajikan sumber informasi web serta mengakses halaman halaman web (Sibero, 2013:12).

Menurut Anhar (2010:6) mengatakan bahwa “*Browser* merupakan *software* yang diinstal di mesin *client*, berfungsi untuk menerjemahkan tag HTML menjadi halaman *web*”.

Pengertian di atas dapat disimpulkan bahwa, untuk dapat mengakses atau menjelajah internet sangat diperlukan perangkat lunak *web browser*.

### **2.1.2. Bahasa Pemrograman**

Instruksi standar untuk memerintah komputer yaitu bahasa pemrograman. Bahasa ini dapat memungkinkan seorang yang ahli program untuk menentukan secara persis data mana yang akan diolah oleh komputer. Berikut bahasa pemrograman yang digunakan, antara lain:

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

Bahasa pemrograman berbasis *web* yang memiliki *file* dengan *extensi file.php* dan tidak dapat diakses tanpa adanya *web server* adalah PHP. Menurut Hidayatullah dan Kawistara dalam bukunya yang berjudul *Pemrograman Web* (2017:223) mengemukakan bahwa “PHP *Hypertext Preprocessor* adalah suatu bahasa *scripting* khususnya digunakan untuk *web development*”. PHP memiliki sifat *server side scripting* sehingga untuk menjalankan PHP harus menggunakan *web server*.

Menurut Sibero (2013:49) mengatakan bahwa “PHP adalah pemrograman *Interpreter* yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan”.

Penjelasan di atas dapat disimpulkan bahwa bahasa pemrograman PHP adalah bahasa pemrograman yang dapat digunakan oleh pengguna untuk mengembangkan kode-kode fungsi PHP sesuai dengan kebutuhannya.

## 2. HTML (*Hypertext Markup Language*)

Sebuah bahasa markah untuk membuat halaman *web* dan bahasa yang digunakannya masih sangat standar seperti salah satu fungsinya untuk membuat tabel, menambahkan objek suara, video dan animasi adalah pengertian dari HTML (Hidayatullah dan kawistara, 2017:15).

Menurut Sibero (2013:19), “*Hypertext Markup Language* atau HTML adalah bahasa yang digunakan pada dokumen sebagai bahasa untuk pertukaran dokumen *web*”.

Pengertian di atas dapat disimpulkan bahwa HTML adalah sebuah dokumen yang berisikan tag, beberapa elemen dan atribut untuk menampilkan halaman pada *web browser*.

## 3. *Javascript*

Bahasa pemrograman yang berbentuk kumpulan skrip yang memiliki fungsi untuk memberikan tampilan agar tampak lebih interaktif pada dokumen *web* adalah pengertian dari *Javascript* (Wahana Komputer, 2010:1).

Sedangkan menurut Sibero (2013:150) dalam bukunya yang berjudul *Web Programming Power Pack* mengatakan bahwa “*Javascript* adalah suatu bahasa pemrograman yang dikembangkan untuk dapat berjalan pada *web browser*”.

Pengertian di atas dapat disimpulkan bahwa *Javascript* adalah bahasa pemrograman untuk memberikan kemampuan tambahan ke dalam bahasa pemrograman HTML (*Hypertext Markup Language*).

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

Sebuah *website* bisa terdiri dari berpuluh-puluh bahkan bisa beratus-ratus halaman. Jika setiap kita ingin merubah halaman *web* tersebut kita harus

mengubah formatnya satu per satu maka akan membuat kita repot. *Cascading Style Sheet* atau yang biasa kita sebut dengan CSS sangat tepat untuk menyimpan format dan menggunakannya kapanpun kita inginkan.

Menurut Saputra dan Agustin (2013:6), berpendapat bahwa: CSS atau yang memiliki kepanjangan *Cascading Style Sheet*, merupakan suatu bahasa pemrograman web yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam web sehingga tampilan web akan lebih rapih, terstruktur dan seragam.

Suatu teknologi yang digunakan untuk mempermudah pengguna mengubah secara keseluruhan warna, tampilan dan dikembangkan untuk menata gaya pengaturan halaman *web* biasa kita kenal dengan sebutan CSS (Sibero, 2013:112).

Pengertian di atas dapat disimpulkan bahwa CSS (*Cascading Style Sheet*) adalah bahasa pemrograman yang dikembangkan untuk mempercantik halaman *web*.

### **2.1.3. Basis Data**

Himpunan kelompok data yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah merupakan definisi dari Basis Data. Secara lebih lengkap pemanfaatan basis data dilakukan untuk memenuhi tujuan Kecepatan, kemudahan, efisiensi ruang penyimpanan, keakuratan, ketersediaan, kelengkapan, keamanan dan pemakaian bersama (Hidayatullah dan Kawistara, 2017:141).

Sedangkan menurut Rosa dan Shalahuddin (2015:43), “Basis Data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat”. Jadi, Basis Data adalah kumpulan data yang saling berhubungan agar dapat diakses oleh *software* tertentu.

### A. MySQL (*My Structure Query Language*)

Menurut Hidayatullah, Kawistara (2017:175), “MySQL adalah salah satu aplikasi DBMS yang sudah sangat banyak digunakan oleh para pemrogram aplikasi *web*”.

Sedangkan aplikasi perangkat lunak yang bertugas untuk menjalankan fungsi pengolahan data disebut MySQL. Pertama MySQL dikembangkan oleh MySQL AB yang kemudian diakuisisi *Sun Microsystem* dan terakhir MySQL dikelola oleh *Oracle Corporation* (Sibero, 2013:97). Dapat disimpulkan bahwa MySQL adalah Perangkat lunak atau *software* yang mengelola SQL.

### B. SQL (*Structure Query Language*)

Menurut Rosa, Shalahuddin (2015:46) berpendapat bahwa “SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada RDBMS”. SQL awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus.

SQL merupakan singkatan dari *Structure Query Language* yang digunakan untuk berkomunikasi dengan suatu *database*. Menurut ANSI (*American National Standards Institute*), SQL adalah bahasa standar untuk *relational database management system* yang digunakan untuk melakukan fungsi seperti *update* data pada *database* atau pengambilan data dari *database* (Sirenden dan Dachi, 2012:16). Dalam proses basis data yang berbasis SQL terdapat beberapa pernyataan SQL dan dikelompokkan menjadi 2 bagian menurut Sibero (2013:98), yaitu:

### 1. *Data Definition Language* (DDL)

DDL dapat digunakan untuk mendefinisikan, mengubah dan menghapus basis data serta objek-objek yang diperlukan dalam basis data, misalnya tabel atau *index*. DDL umumnya digunakan untuk mendefinisikan suatu wadah data atau *record*. DDL terdiri dari *CREATE* yang digunakan untuk membuat objek baru, *ALTER* digunakan untuk mengubah struktur tabel yang sudah ada, *DROP* digunakan untuk menghapus *database*, tabel atau *index* dan *RENAME* digunakan untuk mengganti nama objek (Sibero, 2013:98).

### 2. *Data Manipulation Language* (DML)

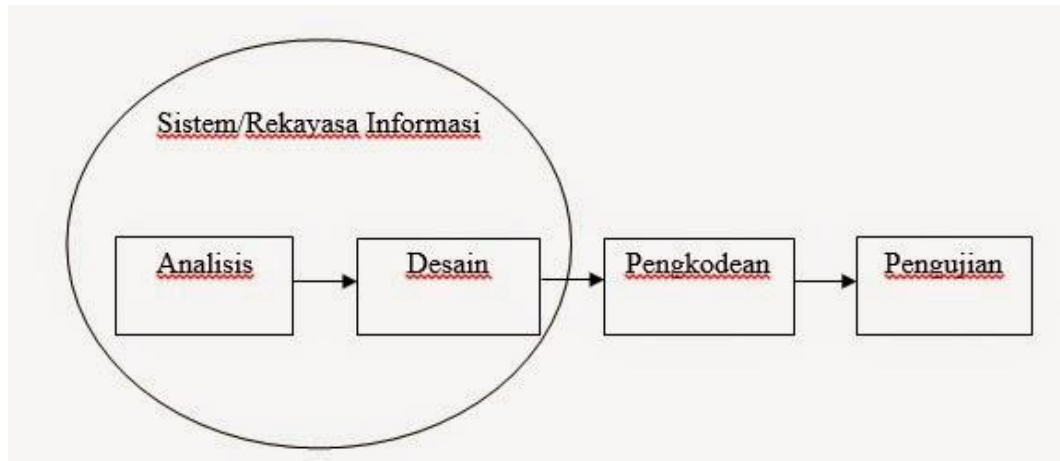
*Data Manipulation Language* (DML) adalah suatu tata bahasa manipulasi data pada MySQL, biasanya digunakan untuk memanipulasi data pada *TABLE Database*, terdiri dari *SELECT* yang digunakan untuk menampilkan data, *INSERT* digunakan untuk penambahan *record* baru ke dalam sebuah tabel, *UPDATE* digunakan untuk mengubah nilai atribut pada suatu *record* dari sebuah tabel, dan *DELETE* digunakan untuk menghapus *record* dari sebuah tabel (Sibero, 2013:104).

#### **2.1.4. Model Pengembangan Perangkat Lunak**

Model pengembangan perangkat lunak yang digunakan oleh penulis menggunakan model *Waterfall*. Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian dan tahap pendukung (*support*).

Berikut penjelasan model *waterfall* menurut Rosa dan Shalahuddin (2015:29), yaitu:





Sumber: Rosa dan Shalahuddin (2015:29)

**Gambar II.1.**

***Ilustrasi Model Waterfall***

**1. Analisis Kebutuhan Perangkat Lunak**

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan 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. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

### 3. Pembuatan Kode Program

Desain harus ditranslasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap 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 yang diinginkan (Rosa dan Shalahuddin, 2015).

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

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis, spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

#### **2.1.5. Aplikasi Pendukung**

Pembuatan sebuah *website* sangat dibutuhkan dengan adanya aplikasi pendukung, beberapa aplikasi pendukung yang digunakan oleh penulis adalah sebagai berikut:

## 1. *Framework*

Penulis akan membahas sedikit gambaran tentang *framework*, dengan tujuan agar anda bisa memahami kata-kata yang orang sering sebutkan dan memahami apa itu *Framework*.

Menurut Purbadian (2016:14) mengatakan bahwa “*Framework* adalah kerangka kerja atau sekumpulan *file-file* yang sudah ter-include, yang mana di dalam *file* tersebut terdapat perintah kode program dan fungsi dasar untuk melakukan tugas tertentu”.

Berbicara tentang *Framework*, yang harus dilakukan oleh *programmer* adalah mengerti, melakukan dan menerapkannya. *Programmer* tidak perlu membuat kode dari nol, karena *framework* telah menyediakan kerangka kerja dalam aplikasi *web* yang dimana di dalam nya memiliki potongan-potongan program yang telah disusun (Saputra, 2011:2).

Pengertian tentang *framework* di atas dapat disimpulkan bahwa *Framework* adalah kumpulan fungsi yang dapat membantu *programmer* dalam menangani masalah dalam pemrograman seperti salah satu contoh nya koneksi ke *database*.

## 2. *CodeIgniter*

Salah satu *framework* yang digunakan dalam pembuatan tugas akhir ini adalah *CodeIgniter*. Penulis menggunakan *Framework CodeIgniter* karena untuk melakukan pengembangan program tidak perlu membuat kode dari awal sehingga dalam proses kerja nya pun terasa lebih cepat.

Menurut Purbadian (2016:18) mengatakan bahwa “*CodeIgniter* adalah sebuah *framework* yang dibuat menggunakan bahasa pemrograman PHP yang

bertujuan untuk memudahkan para *programmer web* untuk membuat atau mengembangkan aplikasi berbasis *web*”.

*CodeIgniter* memiliki eksekusi tercepat dibandingkan dengan *framework* lainnya. *CodeIgniter* bersifat *open source* dan menggunakan model basis MVC (*Model View Controller*), yang merupakan model konsep modern saat ini. *CodeIgniter* juga menawarkan banyak *library* yang dapat digunakan (Saputra, 2011:2).

Metode MVC (*Model View Controller*) terdapat tiga komponen menurut Hidayatullah dan Kawistara (2017:285), yaitu :

- a. *Model*, mengelola basis data (RDBMS) seperti MySQL ataupun Oracle RDMS. Model berhubungan dengan *database* sehingga biasanya dalam model akan berisi *class* ataupun fungsi untuk membuat (*create*), melakukan pembaruan (*update*), menghapus data (*delete*), mencari data (*search*), dan mengambil data (*select*) pada *database*. Selain itu juga model akan berhubungan dengan perintah-perintah *query* sebagai tindak lanjut dari fungsi-fungsi (*create, update, delete, select*).
- b. *View*, bagian *User Interface* atau bagian yang nantinya merupakan tampilan untuk *end-user*. *View* bisa berupa halaman HTML, CSS, *Javascript*, *JQuery* dan *AJAX*, karena metode yang dipakai merupakan MVC sehingga *view* tidak boleh terdapat pemrosesan data ataupun pengaksesan yang berhubungan dengan *database*, sehingga *view* hanya menampilkan data-data hasil dari *Model* dan *Controller*.
- c. *Controller*, penghubung antara *view* dan model, maksudnya ialah karena model tidak dapat berhubungan langsung dengan *view* ataupun sebaliknya,

jadi *controller* inilah yang digunakan sebagai jembatan keduanya. Sehingga tugas *controller* ialah sebagai pemrosesan data atau *Alur Logic Program*, menyediakan *variable* yang akan ditampilkan di *view*, pemanggilan model sehingga model dapat mengakses *database*, *error handling*, validasi atau *check* terhadap suatu *input* data.

Kesimpulan dari pengertian di atas bahwa *CodeIgniter* adalah *Framework* PHP yang di dalamnya terdapat fitur lengkap aplikasi *web* yang sudah dikemas menjadi satu.

### 3. *Bootstrap*

Sebuah *Framework* CSS untuk membangun *website* yang menarik agar memudahkan pengembang disebut *Bootstrap*. Sulit untuk mengembangkan dan pemeliharaannya jika tidak ada konsistensinya terhadap aplikasi individual. *Bootstrap* memberikan solusi rapi dan seragam terhadap solusi yang umum, tugas interface yang setiap pengembang hadapi.

Menurut Alatas (2013:2) dalam bukunya yang berjudul *Responsive Web Design* dengan PHP dan *Bootstrap*, mengemukakan bahwa “*Bootstrap* merupakan *Framework* ataupun *Tools* untuk membuat aplikasi *web* ataupun situs *web responsive* secara cepat, mudah dan gratis”.

Kumpulan alat gratis untuk membuat *website* dan aplikasi *web* atau modular yang pada dasarnya terdiri dari serangkaian *stylesheet* LESS yang menerapkan berbagai komponen *toolkit* adalah *Bootstrap* (Riyanto, 2014:18).

Pengertian di atas dapat disimpulkan bahwa *bootstrap* adalah *Framework* CSS untuk membangun *website* yang menarik supaya memudahkan pengembang.

#### 4. *JQuery*

*JQuery* adalah *Javascript Library* atau kumpulan dari kode atau fungsi *Javascript* yang siap pakai, agar mempercepat dan memudahkan kita dalam membuat kode *Javascript*. Pada intinya *jQuery* menyederhanakan kode *javascript* (Saputra, 2011:44).

Menurut Hidayatullah, Kawistara (2017:395) dalam bukunya yang berjudul *Pemrograman Web* mengatakan bahwa “*JQuery* adalah *Library Javascript* yang akan menjadikan *web* anda lebih bagus dalam hal *user interface*, lebih stabil, dan dapat mempercepat waktu dan kinerja anda dalam membuat *web* karena anda hanya perlu memanggil fungsinya saja tanpa harus membuatnya dari awal”.

Pengertian di atas dapat disimpulkan bahwa *JQuery* adalah kumpulan fungsi-fungsi *Javascript* yang sudah dibentuk sebagai suatu objek.

#### 5. XAMPP

Salah satu *Software* yang digunakan dalam pembuatan tugas akhir ini adalah XAMPP. Penulis menggunakan *web server* XAMPP, karena semua kebutuhan telah disediakan oleh XAMPP. Berikut beberapa paket yang sudah disediakan (Nugroho, 2011:2), yaitu:

- a. *Apache HTTPD 2.0.54*
- b. *MySQL 4.1.12*
- c. *PHP 5.0.3*
- d. *FilZila FTP Server 0.9 Beta*
- e. *phpMyAdmin 2.6.1-pl3*

Menurut Purbadian (2016:1), mengatakan bahwa “XAMPP merupakan suatu *software* yang bersifat *open source* yang merupakan pengembangan dari LAMP (*Linux, Apache, MySQL, PHP and Perl*)”.

Kesimpulan dari pengertian di atas adalah XAMPP merupakan perangkat lunak yang digunakan untuk membangun *website* dan dapat mendukung banyak sistem seperti *Windows, Linux* dan *Mac* sehingga tidak masalah ketika berpindah-pindah sistem operasi.

#### 6. *Sublime Text*

*Text editor* yang cukup terkenal dikalangan pengembang atau *programmer* disebut *Sublime* ([www.linuxjournal.com](http://www.linuxjournal.com)). *Text editor* yang canggih untuk kode *markup* dan prosa agar pengguna menyukai antarmuka yang apik, tampilan luar biasa dan performa yang baik merupakan pengertian dari *Sublime text* ([www.sublimetext.com](http://www.sublimetext.com)).

Kesimpulan dari pengertian di atas adalah bahwa *Sublime* merupakan *software* yang digunakan *programmer* untuk membikin suatu tampilan *website*.

## 2.2 Teori Pendukung

Penulisan tugas akhir ini tidak lepas dari beberapa teori-teori yang ikut mendukung didalamnya. Adapun teori tersebut seperti pengertian Rekam Medis, Struktur Navigasi, ERD dan Pengujian *Web*. Berikut penjelasan dari beberapa teori pendukung yang digunakan oleh penulis.

### 2.2.1. Rekam Medis (*Medical Record*)

Rekam Medis adalah salah satu bagian yang ada di rumah sakit dan puskesmas. Segala bentuk pencatatan dan identitas pasien ada di dalam berkas rekam medis. Untuk lebih jelasnya mari kita cari tahu apa itu rekam medis:

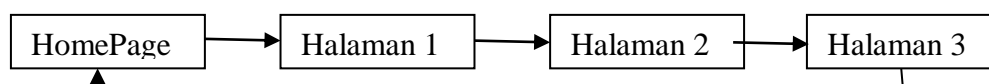
Menurut Peraturan Menteri Kesehatan Republik Indonesia (2008:2), menyatakan bahwa “Rekam Medis adalah berkas yang berisikan catatan dan dokumen tentang identitas pasien, pemeriksaan, pengobatan, tindakan dan pelayanan lain yang telah diberikan kepada pasien”. Kesimpulannya adalah bahwa Rekam medis merupakan Dokumen penting bagi setiap instansi Rumah Sakit yang harus terjaga kerahasiaan nya.

### 2.2.2. Struktur Navigasi

Menurut Sutopo (2007:245), berpendapat bahwa “Struktur Navigasi adalah struktur bagaimana halaman *web* dihubungkan dengan halaman lain”. Struktur navigasi lebih baik dilakukan sebelum kita membuat *website*. Berikut empat macam bentuk dasar struktur navigasi, yaitu:

#### 1. *Linear Navigation Model*

Digunakan oleh sebagian besar *website*. Informasi diberikan secara sekuensial dimulai dari satu halaman. Beberapa desainer *web* menggunakan satu halaman untuk masuk atau keluar dari *website*.



Sumber : Sutopo (2007:6)

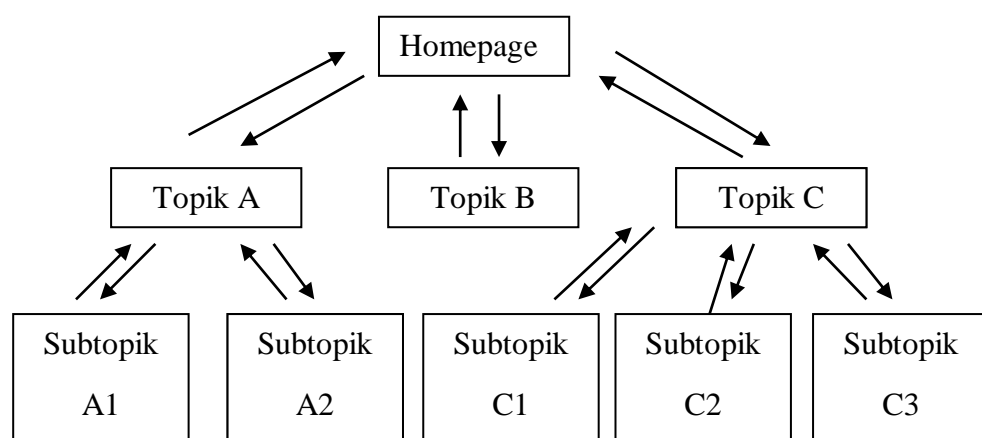
**Gambar II.2.**

**Struktur Navigasi *Linear Navigation Model***



## 2. *Hierarchical Model*

*Hierarchical Model* di adaptasi dari *top-down design*. Konsep navigasi ini dimulai dari satu node yang menjadi *homepage* yang dapat dibuat beberapa cabang ke halaman-halaman utama. Apabila diperlukan, dari tiap halaman utama dapat dikembangkan menjadi beberapa cabang lagi. Hal ini seperti struktur organisasi perusahaan.



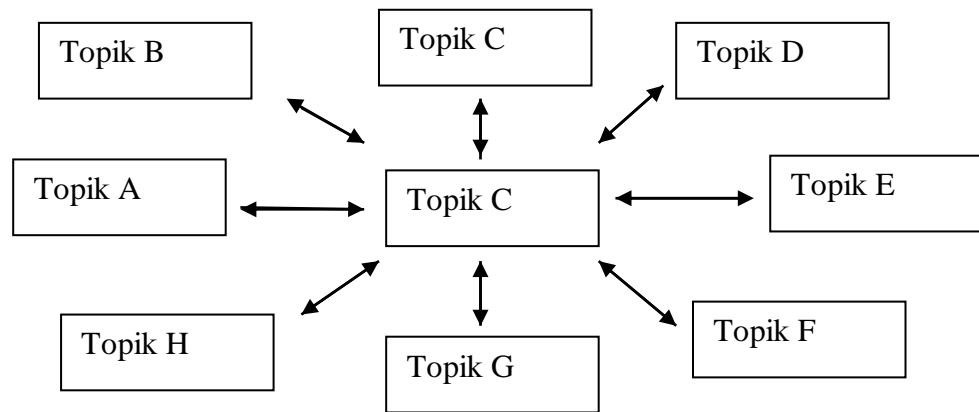
Sumber : Sutopo (2007:6)

**Gambar II.3.**

### **Struktur Navigasi *Hierarchical Model***

## 3. *Spoke-and-Hub Model*

Hanya menggunakan satu node untuk berhubungan dengan node lain. Pada model ini hanya terdapat dua macam link, yaitu *homepage* ke halaman tertentu dan dari halaman tersebut kembali ke *homepage*.



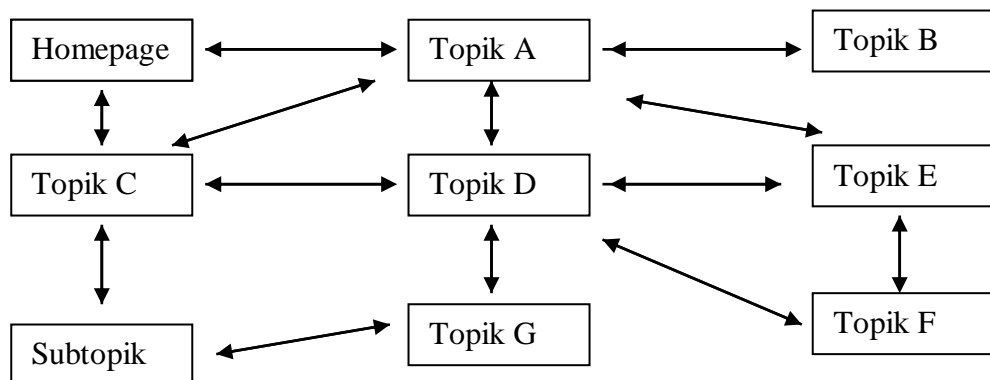
Sumber : Sutopo (2007:7)

**Gambar II.4.**

#### **Struktur Navigasi Spoke-and-Hub Model**

#### 4. *Full Web Model*

Memberikan kemampuan *hyperlink* yang banyak. *Full Web model* banyak digunakan karena *user* dapat mengakses semua topik dengan subtopik dengan cepat, namun kelemahan dari model ini yaitu dapat berakibat *user* kehilangan cara untuk kembali ke topik sebelumnya.



Sumber : Sutopo (2007:8)

**Gambar II.5.**

#### **Struktur Navigasi Full Web Model**

Struktur alur dari suatu program yang merupakan rantai kerja dan rancangan hubungan dari area yang berbeda dan dapat membantu mengorganisasikan semua elemen pembuatan *website* bisa juga disebut sebagai struktur navigasi. Navigasi membuat mudah saat menjelajahi situs web (Suyanto, 2008:62). Kesimpulan nya adalah Struktur Navigasi sangat berguna bagi sebuah *website* karena dengan adanya struktur navigasi ini kita dapat mengakses halaman yang ada pada Website.

### **2.2.3. Entity Relationship Diagram (ERD)**

Menurut Hasugian dan Shidiq (2012:608) mengemukakan bahwa, “ERD adalah sebuah model data yang menggunakan beberapa notasi untuk menggambarkan data dalam hal entitas dan relasi yang digambarkan oleh data tersebut”.

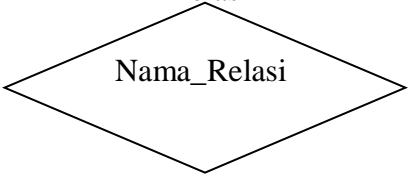
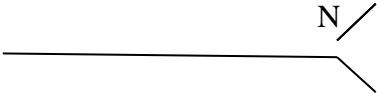
Pemodelan awal basis data yang banyak digunakan yaitu dengan menggunakan ERD (*Entity Relationship Diagram*). Bisa dikatakan juga bahwa ERD adalah bentuk paling awal dalam melakukan perancangan basis data relasional (Rosa dan Shalahuddin, 2015:53). Kesimpulan dari penjelasan di atas bahwa ERD (*Entity Relationship Diagram*) adalah suatu model basis data yang saling berhubungan atau berelasi antar objek

Berikut adalah simbol-simbol yang digunakan pada ERD dengan notasi chen menurut Rosa dan Shalahuddin (2015:50), yaitu:

## 1. Komponen ERD

**Tabel II.1.**  
**Komponen ERD**

Simbol	Deskripsi
<p style="text-align: center;">Entitas / <i>Entity</i></p> <div style="border: 1px solid black; width: 150px; height: 50px; margin: 10px auto; text-align: center; padding-top: 10px;">Nama_Entitas</div>	<p>Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.</p>
<p style="text-align: center;">Atribut</p> <div style="border: 1px solid black; border-radius: 15px; width: 150px; height: 30px; margin: 10px auto; text-align: center; padding-top: 5px;">Nama_atribut</div>	<p><i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.</p>
<p style="text-align: center;">Atribut Kunci Primer</p> <div style="border: 1px solid black; border-radius: 15px; width: 150px; height: 30px; margin: 10px auto; text-align: center; padding-top: 5px;">Nama_kunci_primer</div>	<p><i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).</p>
<p style="text-align: center;">Atribut multivalai / Multivalue</p> <div style="border: 1px solid black; border-radius: 15px; width: 150px; height: 30px; margin: 10px auto; text-align: center; padding-top: 5px;"> <div style="border: 1px solid black; border-radius: 15px; width: 100px; height: 20px; margin: 5px auto; text-align: center; padding-top: 2px;">Nama_Atribut</div> </div>	<p><i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu</p>

<p style="text-align: center;">Relasi</p>  <p style="text-align: center;">Nama_Relasi</p>	<p>Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.</p>
<p>Asosiasi / Association</p> 	<p>Penghubung antar relasi dan entitas dimana di kedua ujungnya memiliki multiplicity kemungkinan jumlah pemakaian</p> <p>Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut engan one to many menghubungkan entitas A dan entitas B.</p>

Sumber : Rosa AS dan M Shalahuddin (2015:50)

## 2. Derajat Relasi (*Relationship Degree*)

Derajat Relasi atau yang biasa disebut Kardinalitas dibedakan menjadi 3 bagian menurut Muslihudin dan Oktafianto (2016:85), yaitu :

### a. Satu ke satu (*one to one*)

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

### b. Satu ke banyak (*one to many*)

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

c. Banyak ke banyak (*many to many*)

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

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

Menurut Hasugian dan Shidiq (2012:608), mengatakan bahwa “LRS (*Logical Record Structure*) adalah sebuah model sistem yang digambarkan dengan sebuah diagram-ER akan mengikuti pola atau aturan pemodelan tertentu dalam kaitannya dengan konversi ke LRS, maka perubahan yang terjadi adalah mengikuti aturan-aturan berikut”.

Hasil dari pemodelan *Entity Relationship* (ER) dan atributnya agar dapat terlihat hubungan-hubungan antar entitas merupakan pengertian dari LRS (Friyadi dalam Amin, 2017:115).

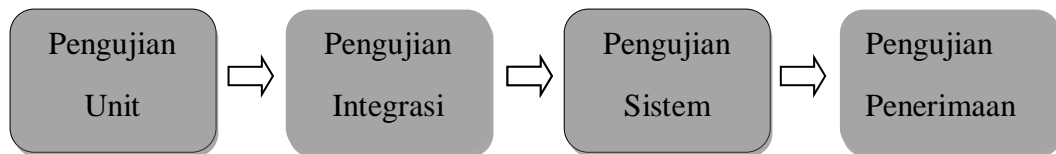
Kesimpulannya bahwa LRS adalah transformasi ERD setelah melalui representasi beberapa *record* menjadi bentuk LRS.

#### 2.2.4. Pengujian Web

Sebuah *software* harus dijaga kualitasnya agar pelanggan merasa puas. Perangkat lunak dijaga kualitasnya untuk dapat bersaing dengan perangkat lunak, yang lain dan dapat mempertahankan pelanggan agar tetap setia menggunakannya serta meningkatkan keuntungan. Pengujian diperlukan tidak hanya untuk meminimalisasi kesalahan teknis tetapi juga kesalahan non teknis.

Menurut Rosa dan Shalahuddin (2015:272), berpendapat bahwa “Pengujian adalah satu set aktifitas yang direncanakan dan sistematis untuk menguji atau mengevaluasi kebenaran yang diinginkan”. Aktifitas pengujian terdiri dari satu set atau sekumpulan langkah dimana dapat menempatkan desain

kasus uji yang spesifik dan metode pengujian. Tahapan pengujian yang secara keseluruhan (Rosa dan Shalahuddin, 2015:274), adalah sebagai berikut:



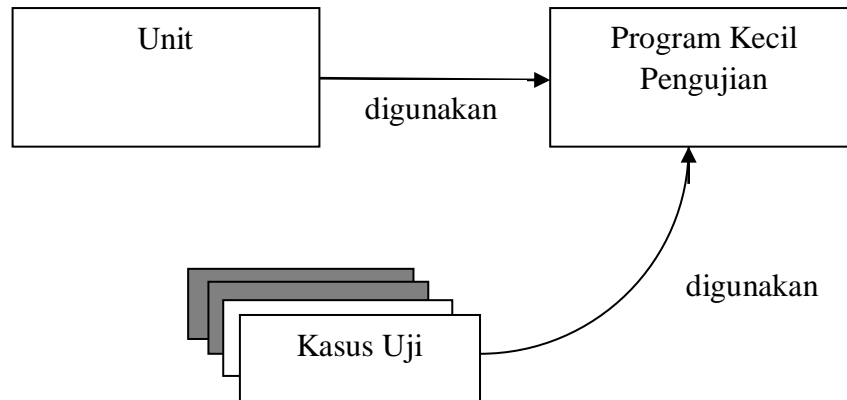
Sumber : Rosa dan Shalahuddin (2015:274)

**Gambar II.6.**  
**Pengujian Perangkat Lunak**

Pengujian sistem yang kualitas nya baik adalah pengujian yang dilakukan secara bertahap sejak awal pengembangannya, jika pengujian sistem nya hanya di akhir maka sudah dapat dipastikan bahwa kualitas sistemnya jelek. Menurut Rosa dan Shalahuddin (2015:275) ada beberapa langkah- langkah pengujian perangkat lunak sebagai berikut:

#### 1. Pengujian Unit

Pegujian unit fokus pada usaha verifikasi pada unit yang terkecil pada desain perangkat lunak (komponen atau modul perangkat lunak). Setiap unit perangkat lunak diuji agar dapat diperiksa apakah aliran masukan (*input*) dan keluaran (*output*) dari unit sudah sesuai dengan yang diinginkan. Pengujian unit biasanya dilakukan saat kode program dibuat, karena dalam sebuah perangkat lunak banyak memiliki unit-unit kecil maka untuk menguji unit-unit kecil ini biasanya dibuat program kecil (*main program*) untuk menguji unit-unit perangkat lunak.



Sumber : Rosa dan Shalahuddin (2015:277)

**Gambar II.7.**  
**Ilustrasi Pengujian Unit**

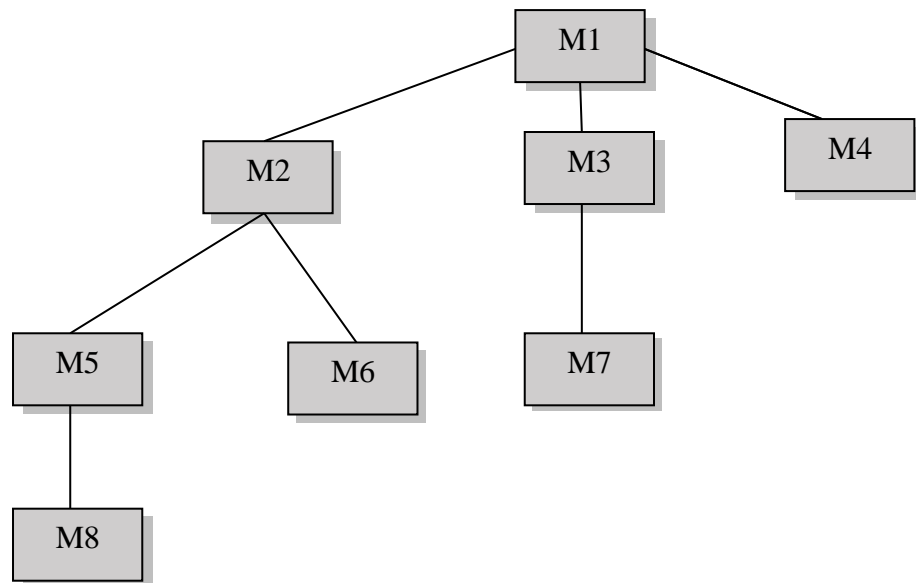
## 2. Pengujian Integrasi

Pengujian Integrasi adalah sebuah teknis yang sistematis untuk mengonstruksi struktur program seiring dengan menggabungkan fungsi program dengan antarmukanya. Pengujian ini bertujuan untuk mempergunakan komponen unit program yang sudah diuji dan membangun struktur seperti yang telah di desain sebelumnya.

## 3. Pengujian *top-Down Integration*

Pengujian integrasi dari atas kebawah ini merupakan pendekatan bertahap (*incremental*) untuk mengonstruksi struktur program. Modul diintegrasikan berdasarkan hirarki, dimulai dari modul yang lebih besar lalu di dekomposisi ke modul yang lebih kecil.





Sumber : Rosa dan Shalahuddin (2015:279)

**Gambar II.8.**

**Ilustrasi Hirarki *Top-Down Integration***

3. Pengujian *Bottom-Up Integration*

Pengujian ini dimulai dari pengujian modul yang paling kecil ke modul yang lebih besar. Pengujian ini sering dilakukan untuk pengembangan perangkat lunak yang tidak menggunakan alur *prototype* sehingga perangkat lunak dibangun dari modul-modul kecil ke yang besar sesuai dengan hirarki pemakaiannya.

4. Pengujian *Regression Integration*

Pengujian ini adalah eksekusi dari beberapa subset pengujian yang sudah terhubung atau saling terkait untuk menjamin bahwa modul yang baru masuk pengujian tidak mengubah fungsionalitas yang sudah diuji sebelumnya.

#### 5. Pengujian *Smoke Integration*

Pengujian asap adalah sebuah pendekatan pengujian integrasi yang biasa digunakan ketika waktu pengerjaan perangkat lunak cukup singkat dan biasanya untuk komponen atau modul yang ditambahkan pada perangkat lunak.