

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Web

World Wide Web (WWW), lebih dikenal dengan *web*, merupakan salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke *Internet* (Sidik, 2007: 1).

Menurut Arief (2011:7) “*Web* adalah salah satu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, suara, animasi, video) di dalamnya yang menggunakan protokol HTTP (*Hypertext Transfer Protocol*) dan untuk mengaksesnya menggunakan perangkat lunak yang disebut *browser*”.

Dapat disimpulkan bahwa *web* adalah sebuah layanan yang berisi dokumen multimedia yang dapat diakses dengan *browser* lewat komputer yang terhubung ke *internet*.

2.1.1 Website

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

1. Internet

Menurut Kotler dan Armstrong (2010:24) “*Internet* merupakan jaringan global dari jaringan-jaringan komputer yang luas dan berkembang tanpa adanya manajemen atau kepemilikan terpusat”. Saat ini, internet menghubungkan individu dan perusahaan satu sama lain dengan informasi di seluruh dunia. Internet menyediakan koneksi informasi, hiburan, dan komunikasi kapanpun, dimanapun. Perusahaan menggunakan internet untuk membangun hubungan yang lebih baik dengan pelanggan dan rekan bisnis, serta untuk mendistribusikan dan menjual produk-produk mereka dengan lebih efektif dan efisien.

2. *Web Browser*

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

Menurut Limantara (2009:1) “*Web Browser* adalah aplikasi perangkat lunak yang memungkinkan penggunaanya untuk berinteraksi dengan teks, *image*, video, *games*, dan informasi lainnya yang berlokasi pada halaman *web* pada *World Wide Web* (WWW) atau *Local Area Network* (LAN)”.

Teks dan *image* pada halaman *web* dapat berisi *hyperlink* ke halaman *web* lain pada *website* yang sama maupun berbeda. Dengan *web browser*, seorang pengguna bisa mengakses informasi yang di sediakan pada banyak *website* secara cepat dan mudah. *Web Browser* memformat informasi *HTML* untuk ditayangkan, oleh karena itu penampakan halaman *web* akan agak berbeda dan satu *browser* ke *browser* lain. *Web Browser* yang paling populer adalah *google chrome* buatan *google*.

3. *Web Server*

Menurut Supardi (2010:2) “*Web Server* merupakan perangkat lunak yang mengelola (mengatur) permintaan *user* dari *browser* dan hasilnya dikembalikan kembali ke *browser*”. Contoh *web server* adalah IIS (*Internet Information Services*) produk *Microsoft Corp*.

Web Server bekerja sebagai penyedia layanan yang dapat diakses oleh banyak pengguna. Dukungan perangkat lunak sangat dibutuhkan agar *web server* dapat berjalan secara optimal. Setiap perangkat lunak *web server* memiliki karakteristik dan teknologi yang dibutuhkan untuk mengatur kerja sistemnya.

2.1.2 Bahasa Pemrograman

Bahasa pemrograman yang digunakan dalam perancangan *website*, antara lain:

1. HTML

Hypertext Markup Language (HTML) adalah sebuah bahasa dasar bahasa markup untuk memformat konten halaman *web* yang digunakan untuk merancang halaman *website* statis (Wahana Komputer, 2012:3).

Menurut Sibero (2013:19) “*Hypertext Markup Language* atau HTML adalah bahasa yang digunakan pada dokumen *web* sebagai bahasa untuk pertukaran dokumen *web*. Struktur dokumen HTML terdiri dari tag pembuka dan tag penutup”.

Berdasarkan definisi di atas, dapat disimpulkan bahwa HTML merupakan bahasa pemrograman *web* yang digunakan untuk mengatur format dan tampilan konten/isi dari halaman *website*.

2. PHP

Menurut Anhar (2010:49) “*PHP* merupakan bahasa pemrograman *web server-side* yang bersifat *open source*”. *PHP* merupakan *script* yang terintegrasi dengan *HTML* dan berada pada *server* (*server side HTML embedded scripting*).

Dengan kata lain, PHP merupakan bahasa pemrograman *web* yang bersifat sebagai *server*, memudahkan dalam pengelolaan *database*, serta dapat digunakan dalam *file* HTML.

3. CSS

Menurut Aditama (2013:367) “*CSS (Cascading Style Sheet)* merupakan salah satu bahasa pemrograman *web* yang bertujuan untuk membuat *website* agar lebih menarik dan terstruktur.”

Dari beberapa pendapat di atas, dapat disimpulkan bahwa CSS adalah suatu bahasa pemrograman *web* yang berfungsi mengatur tampilan teks dan gambar dari suatu *website* agar terlihat lebih menarik dan terstruktur.

4. JQuery

Menurut Wahana Komputer (2012:2) “*JQuery* merupakan pustaka *JavaScript* yang dibangun untuk mempercepat dan memperingkas serta menyederhanakan manipulasi dokumen *HTML*, penanganan *event*, animasi, dan interaksi Ajax untuk mempercepat pengembangan *web*”.

Jquery pertama kali dirilis tahun 2006 oleh *John Resig JQuery* menjadi sangat populer hingga digunakan oleh banyak *website* kelas dunia seperti *Google*, *Amazon*, *Twitter*, *ESPN*, dan lain-lain.

5. JavaScript

Menurut Sibero (2013:150) “*JavaScript* adalah suatu bahasa pemrograman yang dikembangkan untuk dapat berjalan pada *web browser*. Pada awalnya *JavaScript* dikembangkan pada *web browser Netscape* oleh *Brenden Eich* dengan nama *Mocha*, kemudian berubah menjadi *Live-Script* dan yang akhirnya menjadi *JavaScript*”.

Menurut Arifin dan *Community* (2008:28) “*JavaScript* adalah *script* program berbasis *client* yang di eksekusi oleh *browser* sehingga membuat halaman *web* melakukan tugas-tugas tambahan yang tidak bisa dilakukan oleh *script HTML* biasa”.

2.1.3 Basis Data

Menurut Aditama (2012:7) “*Database* adalah tempat media penyimpanan data kita dalam membuat sebuah program yang berisikan tabel, *field* dan record yang diselimuti namanya DBMS (*Database Management System*)”.

Di dalam DBMS (*Database Management System*), terdapat struktur *query* yang digunakan untuk mengolah *database* yang dikenal dengan SQL. Dalam proses pembuatan *database* SQL diperlukan sebuah perangkat lunak

yang dikenal dengan nama *MySQL*. *MySQL* merupakan salah satu perangkat lunak untuk sistem manajemen *database* SQL. *MySQL* merupakan tipe data relasional yang artinya *MySQL* menyimpan datanya dalam bentuk tabel-tabel yang saling berhubungan. (Wahana Komputer, 2011:15)

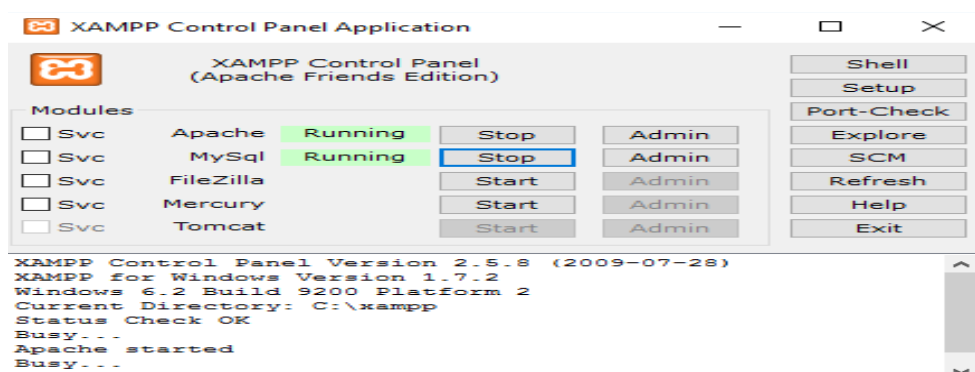
2.1.4 Aplikasi Perancangan Web

Dalam perancangan *website* ini, ada beberapa aplikasi yang digunakan, diantaranya:

1. Xampp Server

Dalam pembangunan sebuah *website* pastinya setiap *programmer* memerlukan bantuan *web server* untuk mengkoneksikan file-file *website* ke basis data. Beberapa *web server* yang sering digunakan diantaranya: *Apache Web Server*, *Sun Java System Web Server*, *Xampp Server*, *Wamp server*, *Xitami Web Server*, dan sebagainya. Dalam hal ini, penulis menggunakan *Xampp Server* dalam membangun *web* tersebut.

Menurut Winpec Solution (2010:1) “*XAMPP* merupakan suatu paket instalasi *Apache*, *PHP*, dan *MySQL*”. Dengan aplikasi ini, anda dapat langsung melakukan instalasi *Apache*, *PHP*, dan *MySQL* sekaligus Aplikasi *XAMPP* ini dapat diperoleh cukup dengan melakukan *download*.

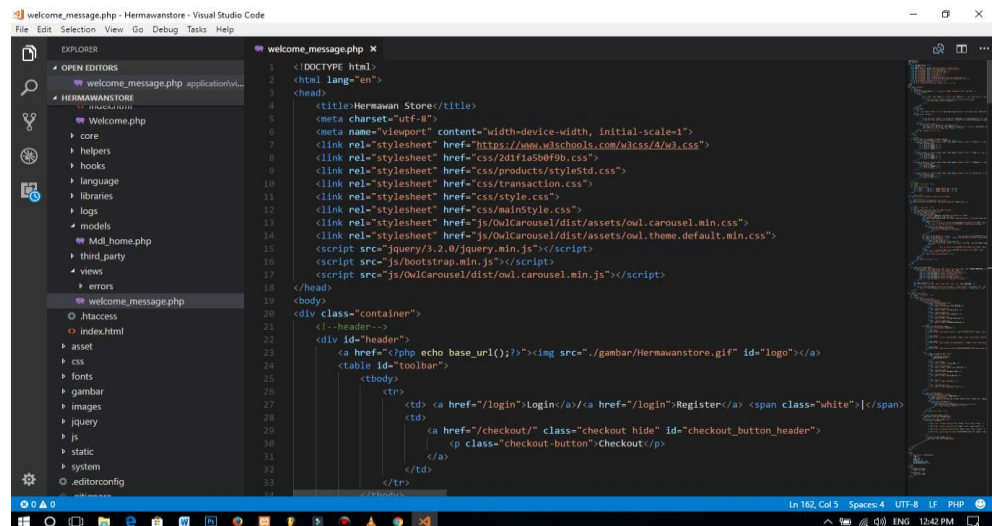


Gambar II.1 XAMPP

2. Visual Studio Code

Visual Studio Code adalah Software yang sangat ringan, namun kuat *editor* kode sumbernya yang berjalan dari desktop. Muncul dengan *built-in* dukungan untuk *JavaScript*, naskah dan *Node.js* dan memiliki *array* beragam ekstensi yang tersedia untuk bahasa lain, termasuk C ++, C #, *Python*, dan *PHP*. Hal ini didasarkan sekitar *Github* ini *Elektron*, yang merupakan versi *cross-platform* dari Atom komponen *code-editing*, berdasarkan *JavaScript* dan *HTML5*. Editor ini adalah fitur lengkap lingkungan pengembangan terpadu (IDE) dirancang untuk pengembang yang bekerja dengan teknologi *cloud* yang terbuka *Microsoft*. *Visual Studio Code* menggunakan *open source NET* perkakas untuk memberikan dukungan untuk *ASP.NET C #* kode, membangun alat pengembang *Omnisharp NET* dan *compiler Roslyn*. Antarmuka yang mudah untuk bekerja dengan, karena didasarkan pada gaya *explorer* umum, dengan panel di sebelah kiri, yang menunjukkan semua file dan folder Anda memiliki akses ke panel editor di sebelah kanan, yang menunjukkan isi dari file yang telah dibuka. Dalam hal ini, editor telah dikembangkan dengan baik, dan menyenangkan pada mata. Ia juga memiliki fungsi yang baik, dengan *intellisense* dan *autocomplete* bekerja dengan baik untuk *JSON*, *CSS*, *HTML*, {kurang}, dan *Node.js*.

Visual Studio Code telah dirancang untuk bekerja dengan alat-alat yang ada, dan *Microsoft* menyediakan dokumentasi untuk membantu pengembang bersama, dengan bantuan untuk bekerja dengan *ASP.NET 5*, *Node.js*, dan *Microsoft* naskah, serta alat-alat yang dapat digunakan untuk membantu membangun dan mengelola aplikasi *Node.js*. *Visual Studio Code* benar-benar sedang ditargetkan pada pengembang *JavaScript* yang ingin alat pengembangannya lengkap untuk *scripting server-side* mereka dan yang mungkin ingin usaha dari *Node.js* untuk kerangka berbasis *NET*. *Visual Studio Code*, adalah belum solid, lintas platform kode Editor ringan, yang dapat digunakan oleh siapa saja untuk membangun aplikasi untuk Web.



Gambar II.2 Ruang Kerja Visual Studio Code

3. Framework

Framework atau bahasa indonesianya kerangka kerja adalah sebuah *software* untuk memudahkan para *programmer* membuat aplikasi atau *web* yang isinya adalah berbagai fungsi, *plugin*, dan konsep sehingga membentuk suatu sistem tertentu. Dengan menggunakan *framework*, sebuah aplikasi akan tersusun dan terstruktur dengan rapi.

2.1.5 Model Pengembangan Perangkat Lunak

Metode perancangan software berdasarkan teori model *waterfall* menurut Sommerville (2007:66) "Tahapan utama yang langsung mencerminkan dasar pembangunan kegiatan", berikut ini tahapannya:

1. *Requirements Definition*

Layanan, batasan, dan tujuan dari sistem ditetapkan melalui konsultasi dengan pengguna sistem. Semua itu didefinisikan secara *detail* dan bertindak sebagai spesifikasi sistem.

2. *System and software Design*

Proses disain sistem membagi kebutuhan menjadi *hardware* atau *software*. Ini menetapkan hampir seluruh perancangan sistem. Disain *software* melibatkan pengidentifikasian dan penggambaran mengenai pemisahan dasar sistem *software* dan hubungannya.

3. *Implementation and unit testing*

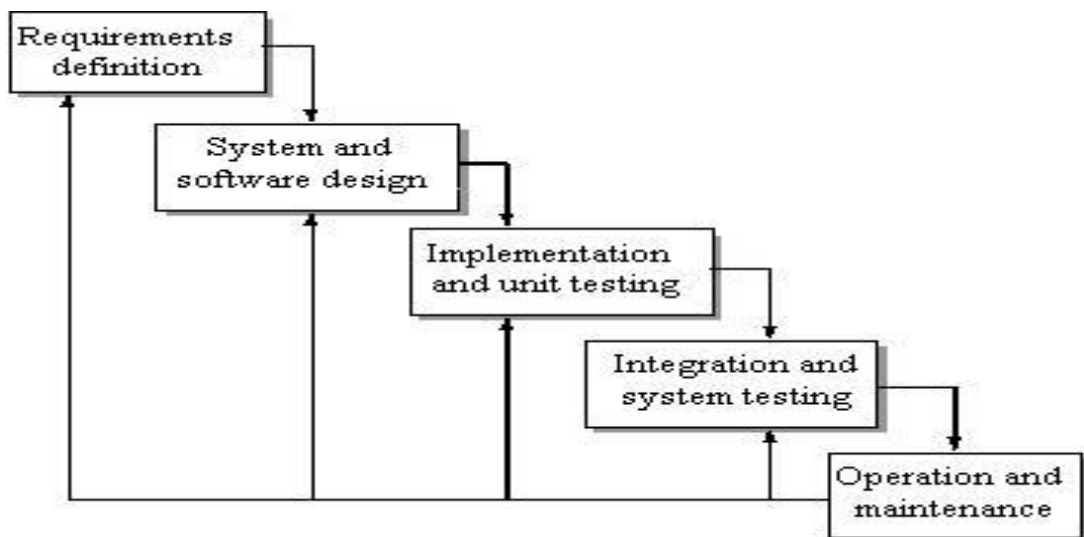
Dalam tahap ini, *design software* adalah menyadari sebagai kumpulan program atau satuan program. *Unit testing* melibatkan verifikasi bahwa setiap *unit* telah mencapai spesifikasinya.

4. *Integration and system testing*

Satuan program atau kumpulan program diintegrasikan dan di tes sebagai sistem yang telah selesai, untuk menjamin bahwa kebutuhan *software* telah terpenuhi. Setelah pengetesan, sistem *software* dikirimkan kepada pelanggan.

5. *Operation and maintenance*

Biasanya, ini adalah bagian siklus hidup *software* yang paling lama. Sistem di-*install* dan dimasukkan kedalam penggunaan. Pemeliharaan melibatkan membenaran kesalahan yang tidak ditemui dalam tahap awal siklus, meningkatkan implementasi satuan sistem, dan meningkatkan layanan sistem sehubungan ditemukannya kebutuhan baru.



Sumber : Sommerville (2007:66)

Gambar II.3 Waterfall Model

2.2 Teori Pendukung

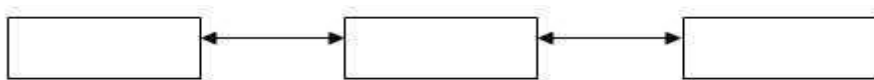
Dalam proses pembuatan *website* ini penulis menggunakan beberapa peralatan pendukung. Adapun beberapa peralatan pendukung yang digunakan adalah:

2.2.1 Struktur Navigasi

Menurut Suyanto (2008:62) “Struktur navigasi dalam situs *web* melibatkan sistem navigasi situs *web* secara keseluruhan dan desain *interface* situs *web* tersebut, navigasi memudahkan jalan ketika menjelajahi situs *web*”. Ada empat macam bentuk dasar yaitu :

1. Struktur Navigasi Linier

Struktur ini hanya mempunyai satu rangkaian cerita yang berurut, yang menampilkan satu demi satu tampilan layar secara berurut menurut urutannya. Tampilan yang dapat ditampilkan pada struktur jenis ini satu halaman sebelumnya atau sesudahnya, pengguna akan melakukan navigasi secara berurutan dalam *frame* atau *byte* informasi satu ke yang lainnya.

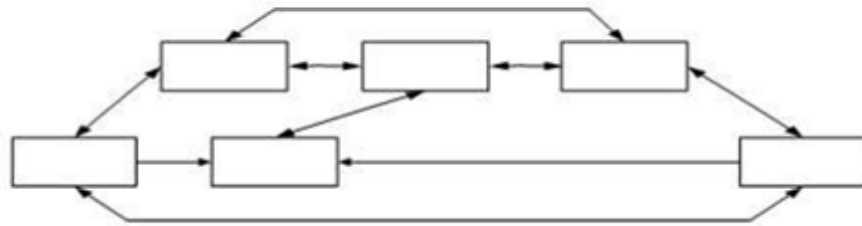


Sumber: Binanto (2010:269)

Gambar II.4 Struktur Navigasi Linier

2. Struktur Navigasi Non-Linier

Struktur navigasi *non-linier* merupakan pengembangan dari struktur navigasi *linier*. Pada struktur ini diperkenankan membuat navigasi bercabang. Percabangan pada struktur *non linier* berbeda dengan percabangan pada struktur hirarki. Karena pada percabangan ini walaupun terdapat percabangan, tetapi tiap-tiap tampilan mempunyai kedudukan yang sama yaitu tidak ada *Master Page* dan *Slave Page*.

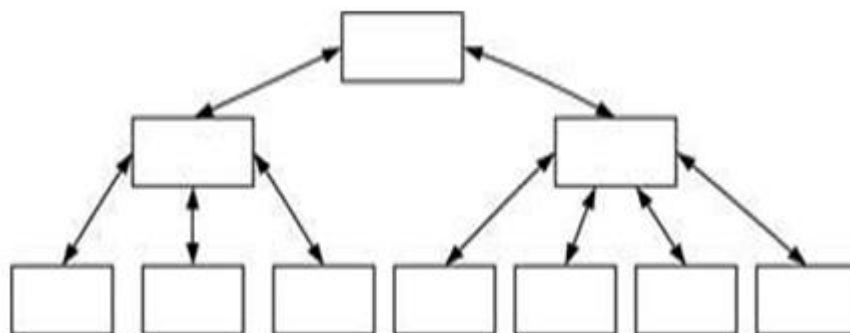


Sumber: Binanto (2010:270)

Gambar II.5 Struktur Navigasi Non-Linier

3. Struktur Navigasi Hirarki

Struktur hirarki biasa disebut stuktur bercabang, merupakan suatu struktur yang mangandalkan percabangan untuk menampilkan data berdasarkan kriteria tertentu.

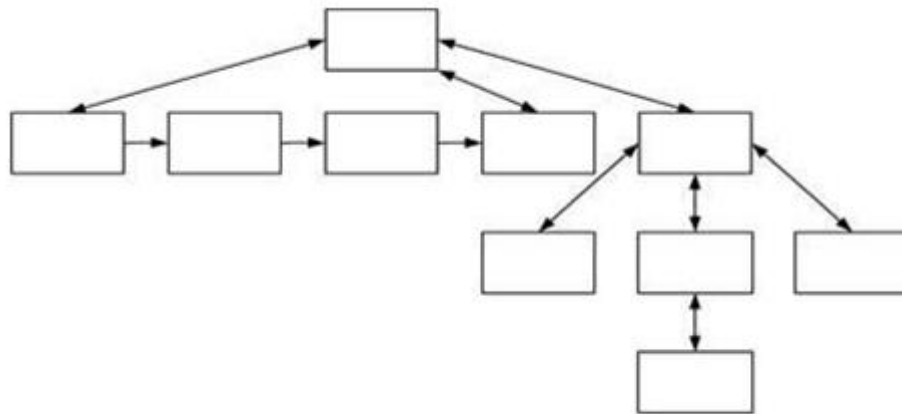


Sumber: Binanto (2010:269)

Gambar II.6. Struktur Navigasi Hirarki

4. Struktur Navigasi Campuran

Struktur navigasi ini disebut juga struktur navigasi bebas yang merupakan gabungan dari ketiga struktur yang ada. Struktur navigasi ini biasa digunakan dalam pembuatan *multimedia*.



Sumber: Binanto (2010:270)

Gambar II.7. Struktur Navigasi Campuran

2.2.2 Entity Relationship Diagram

Al Fatta (2007:121) mendefinisikan bahwa “ERD (*Entity Relationship Diagram*) adalah gambar atau diagram yang menunjukkan informasi dibuat, disimpan, dan digunakan dalam sistem bisnis”.

ERD merupakan representasi data sebagai *entitas*, *atribut* dan *relasi*. *Entitas* menggambarkan kumpulan dari segala data, lalu *Atribut* atau elemen data merupakan unit terkecil dari data yang dapat menjelaskan apa yang dimiliki oleh suatu *entitas* (karakteristik dari *entitas*) sedangkan *relasi* menjelaskan keterkaitan di antara dua *entitas* yang berbeda.

Entity Relationship Diagram digunakan untuk mempermudah dalam perancangan *database* yang mengutamakan untuk permodelan desain konseptual. Karena *Entity Relationship Diagram* menggambarkan struktur dan keterkaitan tabel-tabel data yang menyusun *database* secara detail.

Sifat hubungan atau *relasi* antar *entitas* dapat dibedakan menjadi tiga jenis (Marimin, 2006:112) sebagai berikut:

1. Hubungan Satu-ke-Satu (*One-to-One Relationship*)

Akan terjadi jika setiap *entitas* dalam suatu himpunan *entitas* hanya berhubungan dengan satu *entitas* pada himpunan *entitas* lain, dan sebaliknya.

2. Hubungan Satu-ke-Banyak (*One-to-Many-Relationship*)

Terjadi jika setiap *entitas* dalam suatu himpunan *entitas* dapat berhubungan dengan beberapa *entitas* pada himpunan *entitas* lain, tetapi tidak sebaliknya

3. Hubungan Banyak-ke-Banyak (*Many-to-Many-Relationship*)

Terjadi jika setiap *entitas* dalam suatu himpunan *entitas* dapat berhubungan dengan beberapa *entitas* pada himpunan *entitas* lain, demikian juga sebaliknya.

Menurut Riyanto (2005:22) “*Logical Record Structur* (LRS) merupakan representasi dari *struktur record-record* pada tabel-tabel yang terbentuk dari hasil *relasi* antar himpunan *entitas*”. Menentukan kardinalitas, jumlah tabel, dan *foreign key* (FK).

2.2.3 Pengujian Web

Al Fatta (2007:169) berpendapat bahwa “Pengujian sistem merupakan proses mengeksekusi sistem perangkat lunak untuk menentukan apakah sistem perangkat lunak tersebut cocok dengan spesifikasi sistem dan berjalan sesuai dengan lingkungan yang diinginkan”.

Pengujian program atau sistem harus dijalankan secara rutin, hal ini dilakukan untuk menemukan kesalahan-kesalahan dalam penulisan suatu pernyataan dalam program, sehingga akan diperoleh program yang sesuai dengan yang diharapkan.

Beberapa metode pengujian sistem menurut Al Fatta (2007:170) diantaranya:

6. *Stub Testing*

Pengujian yang difokuskan pada pengujian struktur kendali sebelum semua modul dituliskan. Pengujian ini penting untuk mengecek apakah struktur kendali sudah memetakan kinerja keseluruhan modul secara tepat.

7. *Unit Testing*

Pengujian unit digunakan untuk menguji setiap modul untuk menjamin setiap modul menjalankan fungsinya dengan baik. Ada 2 Metode untuk melakukan *unit testing*, yaitu:

- a. *Black Box Testing*, Terfokus pada apakah unit program memenuhi kebutuhan (*requirement*) yang disebutkan dalam spesifikasi. pada *black box testing*, cara pengujian hanya dilakukan dengan menjalankan atau mengensekusi unit atau modul, kemudian diamati apakah hasil dari unit itu sesuai dengan proses bisnis yang diinginkan. Jika ada yang unit yang tidak sesuai outputnya maka untuk menyelesaikanya, diteruskan pada pengujian yang kedua, yaitu *white box testing*.
 - b. *White Box Testing*, cara pengujian dengan melihat kedalam modul untuk meneliti kode-kode program yang ada, dan menganalisis apakah ada kesalahan atau tidak. Jika ada modul yang menghasilkan *output* yang tidak sesuai dengan proses bisnis yang dilakukan, maka baris-baris program, *variabel*, dan *parameter* yang terlibat pada unit tersebut akan dicek satu persatu dan diperbaiki, kemudian di-*compile* ulang.
- 8. *Integration Testing*
 Pengujian interaksi dari modul-modul yang menyusun sistem informasi untuk menjamin bahwa mereka bekerja dengan baik. *Integration test* terdiri dari serangkaian tes sebagai berikut:
 - a. Ujicoba antarmuka
 - b. Ujicoba skenario pengguna
 - c. Ujicoba aliran data
 - d. Ujicoba sistem antarmuka
- 9. *Pengujian Sistem*
 Pengujian sistem yang terdiri dari sistem transmisi data perangkat keras (komputer, *magnetic reader tes*).