

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Program

Pada pembuatan tugas akhir ini tidak lepas dari teori yang sangat mendukung kemudahan dalam mempelajari serta merancang suatu program berbasis website yang harapannya semua konten berfungsi dengan baik. Jika program berbasis website mudah digunakan akan memperkecil kemungkinan terjadinya kesalahan yang dilakukan oleh user saat menjalankan program berbasis website tersebut.

Berikut ini merupakan teori pendukung yang memperkuat penulisan Tugas Akhir ini.

Konsep dasar program memegang peranan penting dalam perancangan, penyusunan, pembangunan, pemeliharaan hingga pengembangan suatu program, khususnya untuk program aplikasi yang kompleks dan besar. Komputer adalah mesin yang dapat melaksanakan seperangkat perintah dasar (instruction set). Agar komputer dapat melakukan sesuai kebutuhan maka komputer perlu diberikan instruksi. Instruksi tersebut dibentuk menjadi sebuah program. Pemrograman komputer bukan hanya pengetikan instruksi, akan tetapi instruksi tersebut haruslah memiliki fungsi didalam program tersebut sehingga tercapailah suatu program yang dapat memecahkan suatu masalah dan memberikan kemudahan bagi penggunaanya. (Rosidin & Lubis, 2017)

A. Bahasa Pemrograman

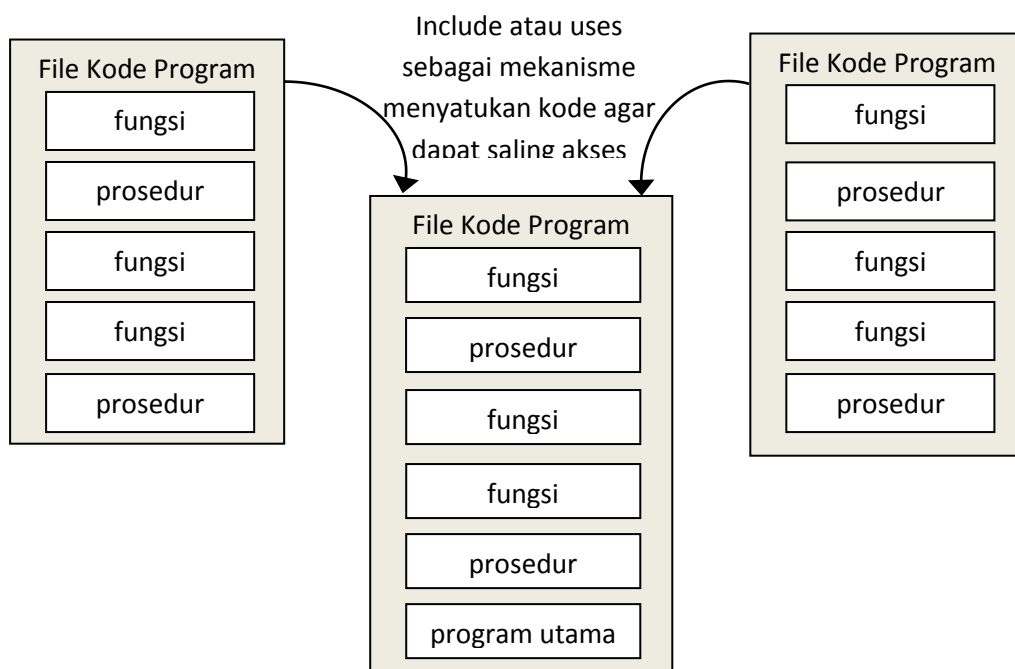
Bahasa pemrograman, atau sering diistilahkan juga dengan bahasa komputer atau bahasa pemrograman komputer, adalah instruksi standar untuk memerintah komputer. Bahasa pemrograman ini merupakan suatu himpunan dari aturan sintaks dan semantik yang dipakai untuk mendefinisikan program komputer. Bahasa ini memungkinkan seorang programmer dapat menentukan secara persis data mana yang akan diolah oleh komputer, bagaimana data ini akan disimpan/diteruskan, dan jenis langkah apa secara persis yang akan diambil dalam berbagai situasi.

Bahasa Pemrograman (programming language) adalah sebuah instruksi standar untuk memerintah komputer agar menjalankan fungsi tertentu. Bahasa pemrograman ini merupakan suatu himpunan dari aturan sintaks dan

semantik yang dipakai untuk mendefinisikan program komputer. Bahasa ini memungkinkan seorang programmer dapat menentukan secara persis data mana yang akan diolah oleh komputer, bagaimana data ini akan disimpan/diteruskan, dan jenis langkah apa secara persis yang akan diambil dalam berbagai situasi.

B. Pemrograman Terstruktur

Menurut (Sukamto & Shalahuddin, 2018) “Pemrograman Terstruktur adalah konsep atau paradigma atau sudut pandang pemrograman yang membagi-bagi program berdasarkan fungsi-fungsi atau prosedur-prosedur yang dibutuhkan program komputer. Modul-modul dibuat dengan mengelompokkan sebuah prosedur atau fungsi dari proses tertentu”. Fungsi-fungsi dan prosedur-prosedur yang ditulis secara sekuensial atau terurut dari atas kebawah sesuai dengan kebergantangan antar fungsi atau prosedur (fungsi atau prosedur yang dapat dipakai oleh fungsi atau prosedur dibawahnya harus yang sudah ditulis atau dideklarasikan diatasnya). Berikut adalah contoh ilustrasi untuk pemrograman terstruktur:



Sumber: (Sukamto & Shalahuddin, 2018)

Gambar II.1.

Ilustrasi Pemrograman Terstruktur

C. PHP

Menurut Kadir dalam (Susanti, 2016) “Php merupakan singkatan dari *Personal Home Page Hypertext Processor*. PHP merupakan bahasa *script* yang ditempatkan dalam server dan diproses di server hasilnya dikirimkan ke klien, tempat pemakainya menggunakan browser. Seperti bahasa pemrograman yang lain, PHP memiliki kelebihan dan juga kelemahan”.

Adapun kelebihan dari PHP antara lain :

- a. PHP merupakan suatu bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- b. PHP dapat berjalan pada web server yang dirilis oleh Microsoft, juga pada Apache yang bersifat *open source*.
- c. Karena sifatnya yang *open source*, maka perubahan dan perkembangan *interpreted* pada PHP lebih cepat dan mudah, karena banyak *milis-milis* dan *developer* yang siap membantu pengembangannya.
- d. PHP memiliki referensi yang begitu banyak sehingga sangat mudah untuk dipahami.

PHP dapat berjalan pada 3 *operating system*, yaitu *linux*, *Unix* dan *Windows*, dan juga dapat dijalankan secara *runtime* pada suatu *console*.

D. *Hypertext Markup Language (HTML)*

HTML singkatan dari *Hypertext Mark-up Language*, yaitu sebuah bahasa *standard* yang digunakan untuk menampilkan mengatur penampilan materi-materi informasi di internet. HTML merupakan bahasa penandaan (*markup*), yaitu suatu bahasa untuk menerangkan cara pemberian format bagi sebuah dokumen. Istilah *markup* berasal dari *copy editor* dahulu yang memberikan tanda dokumen-dokumen

untuk dikirimkan ke printer, yang biasanya berisi tentang huruf dipakai, dan lain-lain. Bahasa markup berisi perintah pemformatan secara eksplisit. Dokumen HTML adalah file teks murni yang dibuat dengan editor teks sembarang. Dokumen ini dikenal sebagai web page. Dokumen HTML merupakan dokumen yang disajikan dalam browser web surfer. Dokumen ini umumnya berisi informasi atau interface aplikasi di dalam internet. (Weriza, 2016)

Dokumen html merupakan teks sederhana yang sering dikenal sebagai ascii, file yang dibuat dengan menggunakan berbagai pengolahan kata (misalnya Notepad/Wordpad di Windows di Windows, Emacs atau vi di mesin UNIX serta BBedit di Macintosh). Bisa menampilkan secara utuh atau WYSIWYG (What You See Is What You Get). Setiap dokumen HTML harus berisi beberapa kode HTML standar. Setiap dokumen terdiri dari bagian kepala dan bagian badan aktual teks yang terdiri dari paragraph, list, dan elemen lainnya. Berikut contoh dokumen HTML sederhana :

```
<html>

<head><TITLE>Berikut          contoh          dokumen          HTML
sederhana</TITLE></head>

<body><<H1>Sistem informasi ini menggunakan HTML</H1>

<P>Di paragraph pertama ini menampilkan judul situs </P>

<P>Di paragraph selanjutnya keterangan lebih lengkap </P>

</body></html>
```

E. My Structure Query Language (MySQL)

Menurut Kadir dalam (Susanti, 2016) “MySQL adalah salah satu jenis database server yang menggunakan SQL sebagai bahasa dasar untuk mengakses

databasenya. Dengan menggunakan script PHP dan PERL Software database ini dapat berfungsi atau berjalan pada semua platform sistem operasi yang biasa digunakan (Windows, Linux, OS/2, berbagai varian Unix).

Apa yang menyebabkan MySQL sangat populer dikalangan *programmer* web? Terdapat beberapa alasan pokok menurut (Mukhtar, 2019), yaitu:

- a. MySQL tersedia di berbagai *platform*, baik itu linux atau windows serta juga dalam berbagai varian unik, misalnya MySQL dirilis oleh *Microsoft*.
- b. Fitur-fitur yang dimiliki oleh MySQL sangat banyak di butuhkan dalam aplikasi web, Contohnya: Kalusa Limit yang berfungsi untuk pengaturan halaman.
- c. MySQL memiliki *overhand* koneksi yang rendah. Karakteristik inilah yang menjadikan MySQL cocok bekerja dengan aplikasi CGI, dimana di setiap *request script* akan melakukan koneksi, mengirimkan satu atau lebih perintah SQL, lalu memutuskan koneksi lagi.

Adapun fungsi koneksi ke *server database* menggunakan pola yang sama yaitu *server, port, user, password*. Fungsi-fungsi tersebut adalah:

- a. *Mysql_connect (host,user, password)*

Fungsi ini digunakan untuk membuka koneksi dengan *server MySQL*.

- b. *Mysql_select_db (nama_database, nama_koneksi)*

Fungsi ini digunakan untuk memilih database yang akan digunakan.

- c. *Mysql_query (perintah_query, nama_koneksi)*

Perintah ini digunakan untuk mengirimkan *query* ke *server database* melalui link nama koneksi.

- d. *Mysql_fetch_array (hasil_query)*

Fungsi ini digunakan untuk melakukan pembacaan *record* hasil *query* yang dilakukan.

e. *Mysql_fetch_row (hasil_query)*

Fungsi ini mempunyai hampir sama dengan *mysql_fetch_array* tetapi array hasil pembacaan data hanya menggunakan indeks numeric saja yang dimulai dari 0 untuk kolom pertama sampai n-1 untuk kolom terakhir hasil *query*.

f. *Mysql_result (hasil_query, no_record, nama_field)*

Fungsi ini berguna untuk mengambil langsung nilai hasil *query* pada suatu baris dan kolom tertentu (satu sel) dengan menyebutkan parameter variable hasil proses *query*, *no_record*, untuk nomor baris (dimulai dari 0) dan nama field.

g. *Mysql_num_field (hasil_query)*

Fungsi ini berguna untuk mendapatkan jumlah *field* dari hasil *query*.

h. *Mysql_num_rows (hasil_query)*

Fungsi ini berguna untuk mendapatkan jumlah baris dari hasil *query*.

F. XAMPP

Menurut (Surniandari & Gustaman, 2012) “*Xampp* adalah perangkat lunak bebas yang merupakan kompilasi dari beberapa program yang mendukung banyak sistem operasi”.

Fungsi dari *web server* sebagai penerima berupa halaman *client* dan mengirimkan kembali hasil yang diminta dalam bentuk halaman *web*. *XAMPP* dapat di *download* melalui <http://www.apachefreinds.org/en/xampp-windows.html>

Cara mengaktifkan *Xampp*:

1. Instal *XAMP*, lalu aktifkan *web server Apache* dan *MySQL* dari kontrol panel *XAMPP*.
2. Jalankan *browser* (*IE*, *Mozilla Firefox* atau *Opera*) lalu ketikkan alamat *web* berikut : <http://localhost/phpmyadmin/> pada address bar lalu tekan Enter.

3. Apabila telah nampak *interface* (tampilan antar muka) phpMyAdmin bisa memulainya dengan mengetikkan nama database, nama tabel dan seterusnya.

G. Atom For Windows

Atom menggambarkan dirinya sebagai editor teks yang dapat diretas, dan artinya adalah memungkinkan programmer baru dan menengah memiliki kesempatan untuk membuat editor teks mereka sendiri tanpa pengalaman kerja atau pemrograman bertahun-tahun. Ini adalah editor teks sumber terbuka yang telah mengubah kodenya bahkan lebih mudah dari sebelumnya. (Guntoro, 2020)

H. Konsep Dasar Web

Teknologi *internet* bukan hal asing bagi masyarakat saat ini, bahkan bagi orang awam pun kata-kata *world wide web* (www) tidak terpisahkan dari kehidupan manusia modern saat ini. Dengan *website* seseorang bisa mendapatkan informasi yang dibutuhkan dengan mudah, cepat dan murah. *Website* juga menjadi sarana komunikasi yang disukai karena sifatnya yang mendunia dan menjadi ajang tukar pendapat bagi penggunanya.

Sistem informasi di *website* pada umumnya dituliskan dalam format HTML. Teknologi *internet* bekerja atas dasar *web browser* di komputer klien yang menampilkan data, gambar dan suara. *Web browser* dikenal sebagai perangkat lunak untuk menjelajah *website*.

I. Internet

Menurut Krisianto dalam (Puspitasari, 2015). “Internet adalah salah satu bentuk media komunikasi dan informasi interaktif. Wujud internet adalah jaringan

komputer yang terhubung di seluruh dunia. Internet digunakan untuk mengirim informasi antar komputer di seluruh dunia. Internet diartikan sebagai “The global public Transmission Control Protocol/Internet Protocol (TCP/IP) internetwork”. Jadi Internet adalah gabungan dari seluruh komputer di dunia yang di satukan oleh sebuah “bahasa” yang sama, adapun bahasa yang dimaksud adalah Transmission Control Protokol/Internet Protokol (TCP/IP).”

Banyak keuntungan yang didapat dari jaringan komputer, diantaranya produktivitas dan efisien. Jaringan komputer menurut area atau lokasi dapat dibagi menjadi empat yaitu:

- a. *Local Area Network (LAN)*, yaitu jaringan komputer dimana komputer-komputer yang terhubung masih dalam satu area atau lokasi.
- b. *Wide Area Network (WAN)*, yaitu koneksi antara LAN-LAN yang berbeda lokasi/area. Ciri utamanya adalah memiliki *bandwidth* yang terbatas karena disesuaikan dengan fungsi harga, adanya *problem delay antarstasiun*.
- c. *Metropolitan Area Network (MAN)*, yaitu sama seperti LAN hanya saja lebih luas areanya semisal dalam satu kota/daerah dengan range mencapai 50 km.
- d. *Internet*, yaitu kepanjangan dari *interconnection networking* atau juga yang telah menjadi *international networking* merupakan suatu jaringan yang menghubungkan komputer di seluruh dunia tanpa dibatasi oleh jumlah unit menjadi satu jaringan yang bisa saling mengakses.

J. Website

Menurut Rahdian dalam (Surniandari & Gustaman, 2012) “Website adalah suatu ruang informasi dimana sumber- sumber daya yang berguna diidentifikasi pengenalan global yang disebut Uniform Resource Locator (URL)“.

Secara termologi, pengertian website adalah kumpulan dari halaman-halaman situs atau link yang biasanya terangkum dalam sebuah domain atau subdomain yang tepatnya berada didalam World Wide Web (WWW) di Internet. Sebuah halaman web adalah dokumen yang ditulis dalam format Hyper Text Markup Language (HTML) yang hampir selalu bisa di akses melalui HTTP, yaitu protokol yang menyampaikan informasi dari server website untuk ditampilkan kepada para pemakai melalui web browser. Halaman-halaman dari sebuah website yang diidentifikasi oleh *Uniform Resource Locator* (URL) biasa Disebut *homepage* atau *domain name*.

Untuk membangun situs diperlukan beberapa unsur yang harus ada agar situs dapat berjalan dengan baik dan sesuai dengan yang diharapkan. Unsur-unsur yang harus ada dalam situs antara lain :

a. *Domain Name*

Domain name atau biasa disebut nama *domain* adalah alamat permanen situs di dunia *internet* yang digunakan untuk mengidentifikasi sebuah situs atau dengan kata lain *domain name* adalah alamat yang digunakan untuk menemukan situs kita pada dunia *internet*.

b. *Hosting*

Hosting dapat diartikan sebagai ruangan yang terdapat dalam *harddisk* tempat menyimpan berbagai data, *file-file*, gambar dan lain sebagainya, yang akan di tampilkan di situs. Besarnya *hosting* di tentukan ruangan *harddisk* dengan ukuran MB (*Mega Byte*) atau GB (*Giga Byte*).

c. *Script/Bahasa Program*

Adalah bahasa yang digunakan untuk menerjemahkan setiap perintah dalam situs pada saat di akses. Jenis *script* sangat menentukan statis, dinamis atau

interaktifnya sebuah situs. Jenis-jenis *script* yang banyak di gunakan para *designer* antara lain HTML, ASP, PHP, *Java Script*, *Java Aplets* dan sebagainya.

d. *Design Web*

Setelah melakukan penyewaan *domain* dan *hosting* serta penguasaan *scripts*, unsure situs yang paling penting dan utama adalah *design*. *Design web* sangat menentukan kualitas dan keindahan situs.

e. Publikasi

Keberadaan situs tidak ada gunanya di bangun tanpa di kunjungi atau dikenal masyarakat. Untuk mengenalkan situs kepada masyarakat memerlukan apa yang disebut publikasi atau promosi. Cara yang biasanya dilakukan dan paling efektif adalah dengan publikasi langsung di internet melalui *search-engine* (mesin pencari, seperti : *Yahoo*, *Google*, *Search Indonesia*).

K. *E-Commerce*

Menurut Fuady dalam (Surniandari & Gustaman, 2012), “E-Commerce dapat diartikan sebagai suatu proses berbisnis dengan memakai teknologi elektronik yang menghubungkan antara perusahaan, konsumen dan masyarakat dalam bentuk transaksi elektronik dan pertukaran/penjualan barang”.

E-commerce telah memberikan pengaruh yang besar terhadap pertumbuhan tata sosial ekonomi masyarakat. *E-commerce* telah menjadi bagian penting dari sektor bisnis baik yang bersifat *private* atau umum. Secara umum, kita dapat mengklasifikasikan *e-commerce* menjadi beberapa jenis, yaitu :

a. *Bussines to Bussines (B2B)*

Trading Partners yang sudah diketahui dan umumnya memiliki hubungan yang cukup lama. Informasi hanya dipertukarkan dengan *partner* tersebut sehingga

jenis informasi yang dikirimkan dapat disusun sesuai kebutuhan dan kepercayaan.

b. *Bussines to Cunsumer (B2C)*

Bussines to Consumer (B2C) merupakan transaksi ritel dengan pembeli individual. Selain itu *Bussines to Cunsumer (B2C)* juga dapat berarti mekanisme toko online (*electronic shopping mall*) yaitu transaksi antara *e-merchant* dengan *e-customer*.

c. *Consumer to Consumer (C2C)*

Consumer to Consumer (C2C) merupakan transaksi di mana konsumen menjual produk secara langsung kepada konsumen lainnya. Selain itu, seorang individu juga bisa mengiklankan produknya berupa barang atau jasa, pegetahuan, atau keahliannya di salah satu situs lelang.

d. *Consumer to Bussines (C2B)*

Consumer to Bussines (C2B) merupakan individu yang menjual produk atau jasa kepada organisasi dan individu yang mencari penjual.

e. *Non-Bussines Electronic Commerce*

Non-Bussines Electronic Commerce meliputi kegiatan non bisnis seperti kegiatan lembaga pendidikan, organisasi nirlaba, keagamaan dan lain-lain.

f. *Intrabussines (Organizational) Electronic Commerce*

Kegiatan ini meliputi semua aktivitas *internal* organisasi melalui *internet* untuk melakukan pertukaran barang, jasa, dan informasi serta menjual produk perusahaan kepada karyawan, dan lain-lain.

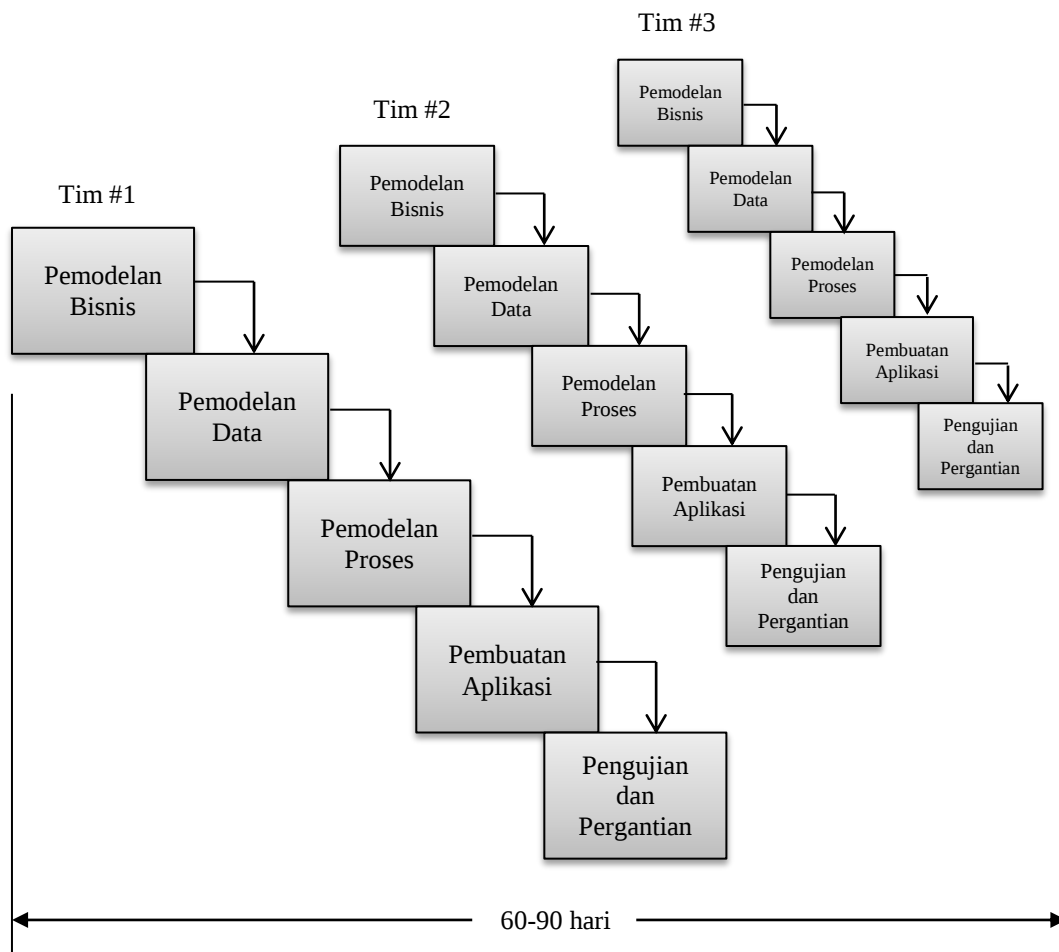
L. Web Browser

Menurut Kasiman dalam (Susanti, 2016) “*Web browser* adalah program untuk menampilkan halaman yang berbentuk kode HTML. Semua halaman web ditulis dengan bahasa *Hypertext Mark Up Language* (HTML). Walaupun beberapa file mempunyai ekstensi yang berbeda (contoh: .html, .php, .php3), output file-file tersebut tetap HTML. HTML adalah medium yang selalu dikirimkan ke web browser baik halaman itu berupa halaman statis, sebuah script (seperti PHP), ataupun yang dibuat oleh program *Common Gateway Interface* (CGI)”.

M. Model Rapid Application Development (RAD)

Menurut (Sukamto & Shalahuddin, 2018) “Rapid Application Development (RAD) adalah model proses pengembangan perangkat lunak yang bersifat inkremental terutama untuk waktu pengerjaan yang pendek. Model RAD adalah adaptasi dari model air terjun versi kecepatan tinggi dengan menggunakan model air terjun untuk pengembangan setiap komponen perangkat lunak.

Jika kebutuhan perangkat lunak dipahami dengan baik dan lingkungan perangkat lunak dibatasi dengan baik sehingga tim dapat menyelesaikan pembuatan perangkat lunak dengan waktu yang pendek. Model RAD membagi tim pengembang menjadi beberapa tim untuk mengerjakan beberapa komponen masing-masing tim pengerjaan dapat dilakukan secara paralel. Berikut adalah gambar dari model RAD:



Sumber: (Sukamto & Shalahuddin, 2018)

Gambar II.2.

Ilustrasi Model RAD

1. Pemodelan Bisnis

Pemodelan yang dilakukan untuk memodelkan fungsi bisnis untuk mengetahui informasi apa yang terkait pada proses bisnis, informasi apa saja yang harus dibuat, siapa yang harus membuat informasi itu, bagaimana alur informasi itu, proses apa saja yang terkait informasi itu.

2. Pemodelan Data

Memodelkan data apa saja yang dibutuhkan berdasarkan pemodelan bisnis dan mendefinisikan atribut-atributnya beserta relasinya dengan data-data yang lain.

3. Pemodelan Proses

Mengimplementasikan fungsi bisnis yang sudah didefinisikan terkait dengan pendefinisian data.

4. Pembuatan Aplikasi

Mengimplementasikan pemodelan proses dan data menjadi program. Model RAD sangat menganjurkan pemakaian komponen yang sudah ada jika dimungkinkan.

5. Pengujian dan Pergantian

Menguji komponen-komponen yang dibuat. Jika sudah teruji maka tim pengembang komponen dapat beranjak untuk mengembangkan komponen berikutnya.

2.2. Tools Program

A. Basis Data

1. *Entity Relationship Diagram (ERD)*

Menurut Munawar dalam (Bakhri, 2015) *Entity Relationship Diagram* merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan suatu persepsi bahwa *real world* terdiri dari *object-object* dasar yang mempunyai hubungan atau relasi antar *object-object* tersebut. Relasi antar *object* dilukiskan dengan menggunakan symbol-symbol grafis tertentu.

Komponen yang terdapat dalam *Entity Relationship Diagram* adalah sebagai berikut:

1. *Entity* (entitas)

Entitas adalah suatu data yang dapat disimpan dan berguna bagi badan atau perusahaan, dengan kata lain suatu obyek yang dapat dibedakan dengan objek lainnya. Entitas digambarkan dengan kotak persegi panjang.

Terdapat juga Entitas Lemah (*Weak Entity*), yaitu suatu entitas sangat bergantung dengan entitas biasa, dengan kata lain, entitas lemah tidak akan ada apabila tidak ada entitas biasa. Entitas lemah digambarkan dengan kotak persegi panjang dengan garis ganda.

2. Atribut

Atribut menunjukkan karakteristik dari tiap-tiap entitas. Atribut digambarkan dengan bentuk oval.

3. Relasi

Relasi menunjukkan hubungan yang terjadi diantara entitas. Relasi digambarkan dengan bentuk belah ketupat atau diamond.

4. *Line Connector*

Line Connector digambarkan dengan bentuk garis tunggal.

5. Atribut Utama

Atribut utama digambarkan dengan bentuk oval, dengan keterangan diberi garis bawah absolute.

6. Atribut Pilihan

Atribut Pilihan digambarkan dengan bentuk oval dengan keterangan diberi garis bawah putus-putus.

7. Kardinalitas

Kardinalitas merupakan tingkat hubungan yang terjadi diantara entitas di dalam sebuah sistem. Terdapat tiga tingkat hubungan yang terjadi, yaitu:

a. Hubungan Satu pada Satu (*One to One* atau 1:1)

Tingkat hubungan dinyatakan satu pada satu jika satu kejadian pada entitas pertama hanya mempunyai satu hubungan dengan suatu kejadian pada entitas

kedua. Demikian juga sebaliknya satu kejadian pada entitas kedua hanya bisa mempunyai satu hubungan dengan satu kejadian pada entitas yang pertama.

b. Hubungan Satu pada Banyak (*One to Many* atau 1:M)

Tingkat hubungan satu pada banyak (1:M) adalah sama dengan banyak pada satu (M:1), tergantung dari arah mana hubungan tersebut dilihat. Untuk satu kejadian pada entitas yang pertama dapat mempunyai banyak hubungan dengan kejadian pada entitas yang kedua. Sebaliknya satu kejadian pada entitas yang kedua, hanya bisa mempunyai satu hubungan dengan satu kejadian pada entitas yang pertama.

c. Hubungan Banyak pada Banyak (*Many to Many* atau M:N)

Tingkat hubungan banyak pada banyak (M:N) terjadi jika tiap kejadian pada sebuah entitas akan mempunyai banyak hubungan dengan kejadian pada entitas lainnya, baik dilihat dari sisi entitas yang pertama maupun dilihat dari sisi entitas yang kedua.

2. *Logical Relationship Structure (LRS)*

Menurut Pratama dalam (Rosidin & Lubis, 2017) “LRS merupakan transformasi dari penggambaran ERD dalam bentuk yang lebih jelas dan mudah untuk dipahami”. Penggambaran LRS hampir mirip dengan penggambaran normalisasi file, hanya saja tidak digambarkan simbol *asterix* (*) sebagai simbol *primary key* (kunci utama) dan *foreign key* (kunci tamu).

LRS merupakan hasil pemodelan *Entity Relationship* (ER) beserta atributnya sehingga bisa terlihat hubungan-hubungan antar entitas.

Dalam pembuatan LRS terdapat tiga hal yang dapat mempengaruhi, yaitu:

1. Jika tingkat hubungan (*cardinality*) satu pada satu (*one to one*), maka digabungkan dengan entitas yang lebih kuat (*strong entity*), atau digabungkan dengan entitas yang memiliki atribut yang lebih sedikit.
2. Jika tingkat hubungan (*cardinality*) satu pada banyak (*one to many*), maka hubungan relasi atau digabungkan dengan entitas yang tingkat hubungannya banyak.
3. Jika tingkat hubungan (*cardinality*) banyak pada banyak (*many to many*), maka hubungan relasi tidak akan digabungkan dengan entitas manapun, melainkan menjadi sebuah LRS.

3. **Constraint Basis Data**

Constraint basis data merupakan struktur yang dibuat oleh pengguna atau perancang basis data yang mencerminkan tingkah laku dari suatu tabel dan kolom. *Constraint* dirancang pada saat mendefinisikan basis data dengan tujuan utama memproteksi validitas data. *Constraint* pertama dan kedua yang digunakan untuk menjaga integritas entitas dan referensial sudah umum digunakan dalam perancangan basis data yakni dengan penggunaan kunci primer (*primary key*) dan kunci tamu (*foreign key*). Sedangkan tipe ke tiga yakni *constraint* domain masih hanya digunakan sampai dengan pemilihan tipe data yang tepat seperti contoh tabel di atas. Hampir semua penelitian yang membahas pembangunan sistem informasi dengan basis data relasional, perancangan basis datanya masih hanya menggunakan kunci primer, kunci tamu dan pemilihan tipe data. (Andria & Lenawati, 2015)

B. Pengkodean

Dari pemakaiannya, kita bisa membedakan adanya pengodean eksternal (*user-defined coding*) dan pengodean internal (*system coding*) menurut (Fathansyah, 2012) Pengkodean eksternal mewakili pengodean yang telah digunakan secara terbuka dan dikenal dengan baik oleh para pemakai awam (*end-user*). Atribut *nim* dan *kode_kul* merupakan dua contoh atribut yang termasuk dalam pengodean eksternal, karena data *nim* dan *kode_kul* ini telah digunakan secara terbuka dan dikenal dengan baik oleh para pemakai awam. Sementara, atribut *kode_dos*

merupakan contoh pengkodean internal. Namun, yang perlu diingat, pengodean internal tidak hanya dapat diterapkan pada pembuatan *key* alternatif seperti *kode_dos* tersebut, tapi juga dapat diterapkan pada atribut data lain (*non-key*) yang memang kita kelola.

Menurut (Fathansyah, 2012) ada 3 (tiga) bentuk pengkodean yang dapat kita pilih, yaitu:

1. *Sekuensial*

Dimana pengodean dilakukan dengan mengasosiasikan data dengan kode terurut (biasanya berupa bilangan asli atau abjad), misalnya data nilai mutu kuliah ('Sempurna', 'Baik', 'Cukup', 'Kurang', 'Buruk') dikodekan dengan 'A', 'B', 'C', 'D' dan 'E'.

2. *Mnemonic*

Dimana pengodean dilakukan dengan membentuk suatu singkatan dari data yang ingin dikodekan, misalnya data jenis-kelamin ('Laki-laki' dan 'Perempuan') dikodekan dengan 'L' dan 'P'.

3. *Blok*

Dimana pengodean dinyatakan dalam format tertentu, misalnya data no.induk mahasiswa dengan format XXYYYY yang terbentuk atas XX = dua digit terakhir angka tahun masuk dan YYYY = no.urut mahasiswa.

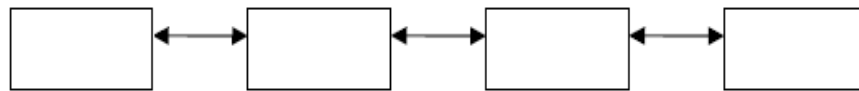
C. Struktur Navigasi

Menurut (Simarmata, 2010) Struktur navigasi merupakan “navigasi yang ada pada situs Web atau aplikasi Web menunjukkan sesuatu yang penting dan menjadi kata kunci usability aplikasi”. Struktur navigasi juga dapat diartikan sebagai struktur alur dari suatu program yang merupakan rancangan hubungan dan rantai kerja dari beberapa area yang berbeda dan dapat membantu mengorganisasikan seluruh elemen pembuatan *website*.

Ada empat macam bentuk dasar, yaitu:

1. Struktur Navigasi *Linear*

Struktur navigasi ini hanya memiliki suatu rangkaian cerita yang berurut, yang menampilkan satu demi satu tampilan layar secara berurut menurut urutannya. Tampilan yang dapat ditampilkan pada struktur jenis ini adalah satu halaman sebelum atau satu halaman sesudahnya, tidak dapat dua halaman sebelumnya atau halaman sesudahnya. (Binanto, 2010)



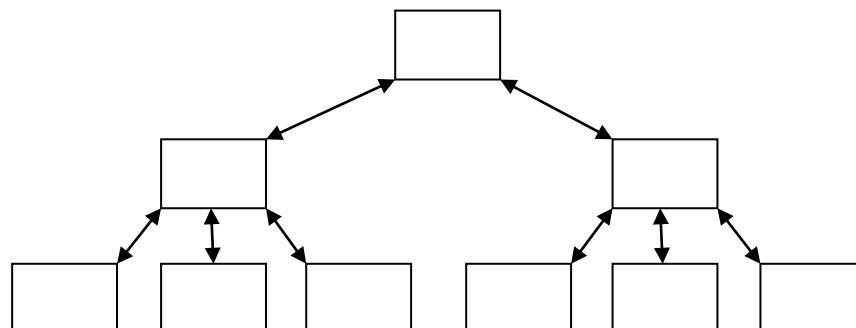
Sumber: (Binanto, 2010)

Gambar II.3.

Struktur Navigasi *Linear*

2. Struktur Navigasi Hierarki (*Hierarchical*)

Struktur navigasi ini disebut juga struktur linear dengan percabangan karena pengguna melakukan navigasi di sepanjang cabang pohon struktur yang terbentuk oleh logika isi, seperti yang terlihat pada gambar dibawah ini: (Binanto, 2010)



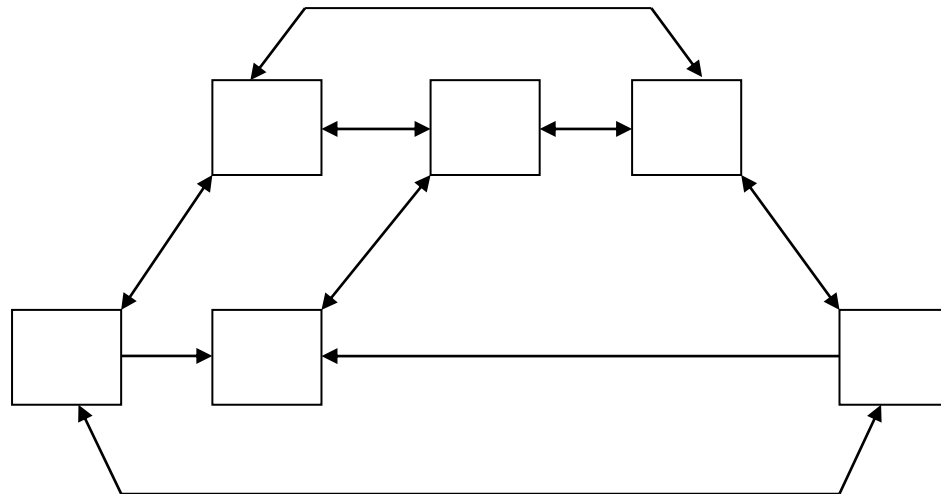
Sumber: (Binanto, 2010)

Gambar II.4.

Struktur Navigasi *Hierarchical*

3. Struktur Navigasi Tidak Berurut (*Non Linear*)

Struktur ini disebut juga struktur tidak berurut merupakan pengembangan dari struktur navigasi linear. Pada stuktur ini diperkenankan membuat navigasi bercabang. Walaupun terdapat percabangan, tetapi setiap tampilan punya kedudukan yang sama. (Binanto, 2010)



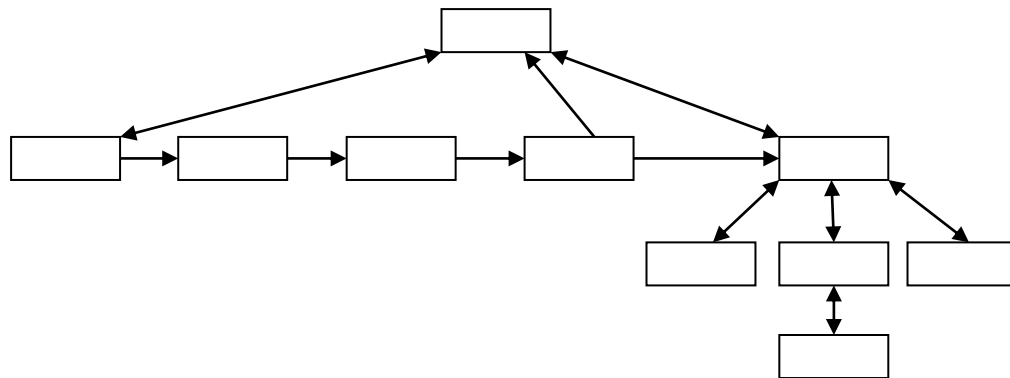
Sumber: (Binanto, 2010)

Gambar II.5.

Struktur Navigasi *Non Linear*

4. Struktur Navigasi Campuran (*Composite*)

Struktur ini merupakan gabungan dari ketiga struktur navigasi sebelumnya. Struktur navigasi ini juga biasa disebut navigasi bebas. Struktur ini banyak digunakan dalam pembuatan aplikasi multimedia sehingga dapat memberikan keinteraksian yang lebih tinggi. (Binanto, 2010)



Sumber: (Binanto, 2010)

Gambar II.6.

Struktur Navigasi Campuran (*Composite*)

D. *Black Box Testing*

Menurut (Rosa & Shalahuddin, 2013), “*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 yang dibuat adalah:

- a. Jika *user* memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang benar.
- b. 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.