



BAB II TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Teknologi

Rusdiana dan Irfan (2014: 205), teknologi adalah suatu benda atau objek yang diciptakan oleh manusia yang dapat bermanfaat bagi kelangsungan hidup manusia.

Anglin (dikutip Rusdiana dan Irfan, 2014: 206), teknologi adalah penerapan ilmu-ilmu perilaku dan alam serta pengetahuan lain secara sistematis dan menyistem untuk memecahkan berbagai masalah yang dihadapi manusia.

Kamus Besar Bahasa Indonesia (dikutip Rusdiana dan Irfan, 2014: 204-205) menyatakan bahwa definisi teknologi adalah:

- (1) metode ilmiah untuk mencapai tujuan praktis ilmu pengetahuan terapan,
- (2) keseluruhan sarana untuk menyediakan barang-barang yang diperlukan bagi kelangsungan dan kenyamanan hidup manusia.

Berdasarkan pengertian di atas penulis menyimpulkan bahwa teknologi adalah suatu objek yang diciptakan oleh manusia sebagai sarana untuk memudahkan atau bermanfaat bagi kelangsungan dan kenyamanan hidup manusia itu sendiri.

2.1.2. Pengertian Perangkat Lunak

Ladjamudin (2013: 20), *software* merupakan kumpulan dari perintah/ fungsi yang ditulis dengan aturan tertentu untuk memerintahkan komputer melaksanakan tugas tertentu.

Rosa dan Shalahuddin (2015: 2), perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).

Rusdiana dan Irfan (2014: 212), *software* (perangkat lunak) adalah rangkaian prosedur dan dokumentasi program yang berfungsi untuk menyelesaikan berbagai masalah yang dikehendaki.



Berdasarkan pengertian di atas penulis menyimpulkan bahwa perangkat lunak adalah suatu program yang dibuat untuk memerintahkan komputer melaksanakan tugas tertentu.

2.1.3. Pengertian Komputer

Blissmer (dikutip Sutabri, 2016: 95), komputer adalah suatu alat elektronik yang mampu melakukan beberapa tugas sebagai berikut: menerima *input*, memproses input sesuai dengan programnya, menyimpan perintah dan hasil pengolahan, serta menyediakan *output* dalam bentuk informasi.

Fuori (dikutip Sutabri, 2016: 96), komputer adalah pemrosesan data (*data processor*) yang dapat melakukan perhitungan yang besar dan cepat, termasuk perhitungan aritmatika atau operasi logika, tanpa campur tangan manusia yang mengoperasikan selama pemrosesan.

Davis (dikutip Sutabri, 2016: 96), komputer adalah tipe khusus alat penghitung yang mempunyai sifat tertentu yang pasti.

Berdasarkan pengertian di atas penulis menyimpulkan bahwa komputer adalah perangkat elektronik yang mampu melakukan pemrosesan dan perhitungan dengan cepat, tanpa campur tangan manusia yang mengoperasikan selama pemrosesan

2.1.4. Pengertian Data

Susanto (dikutip Rusdiana dan Irfan, 2014: 68), data adalah fakta yang dapat digunakan sebagai input dalam menghasilkan informasi.

Ladjamudin (2013: 9), data adalah kenyataan yang menggambarkan kejadian-kejadian dan kesatuan yang nyata.

McLeod (dikutip Rusdiana dan Irfan, 2014: 301), data adalah sekumpulan baris fakta yang mewakili peristiwa yang terjadi pada organisasi atau pada lingkungan fisik sebelum diolah ke dalam format yang bisa dimengerti dan digunakan manusia.



Berdasarkan pengertian di atas penulis menyimpulkan bahwa data adalah fakta atau kenyataan yang dapat diolah menjadi informasi yang dapat dimengerti dan digunakan manusia.

2.1.5. Pengertian Database

Rusdiana dan Irfan (2014: 302), *database* adalah susunan *record* data operasional lengkap dari suatu organisasi atau perusahaan, yang diorganisasi dan disimpan secara terintegrasi dengan menggunakan metode tertentu sehingga mampu memenuhi informasi yang optimal yang dibutuhkan oleh para pengguna.

Rosa dan Shalahuddin (2015: 43), basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.

Kristanto (dikutip Rusdiana dan Irfan, 2014: 301), basis data adalah kumpulan *file* yang saling berelasi.

Berdasarkan pengertian di atas penulis menyimpulkan bahwa *database*/basis data adalah media yang digunakan untuk menyimpan susunan record data yang saling berelasi agar dapat diakses dengan mudah dan cepat.

2.1.6. Model Pengembangan Perangkat Lunak

Rosa dan Shalahuddin (2015: 28-30), menjelaskan bahwa model *Waterfall* sering disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

a. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk memesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu didokumentasikan.

b. 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.

c. 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 tahap desain.

d. 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.

e. 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.

2.2. Teori Judul

2.2.1. Pengertian Aplikasi

Sujatmiko (2012: 23), *application* adalah program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu, misalnya *Ms-Word*, *Ms-Excel*.

Asropudin (2013: 6), *application* adalah *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Ms-Word*, *Ms-Excel*.

Berdasarkan pengertian di atas penulis menyimpulkan bahwa aplikasi adalah program komputer atau *software* yang dibuat oleh suatu perusahaan untuk membantu manusia dalam mengerjakan tugas-tugas tertentu.

2.2.2. Pengertian Pengolahan Data

Ladjamudin (2013: 9), pengolahan data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan.



Sutabri (2016: 98), pengolahan data adalah suatu proses menerima data sebagai masukan (*input*), memproses (*processing*) menggunakan program tertentu, dan mengeluarkan hasil proses data tersebut dalam bentuk informasi (*output*).

Berdasarkan pengertian di atas penulis menyimpulkan bahwa pengolahan data adalah suatu proses yang mendeskripsikan perubahan bentuk data menjadi informasi yang berguna.

2.2.3. Pengertian Kenaikan Pangkat

Naabil (2017), kenaikan pangkat adalah tindakan resmi dalam administrasi kepegawaian yang mengakibatkan perubahan pangkat pegawai negeri sipil, yakni pangkatnya menjadi lebih tinggi daripada pangkat semula.

Nur (2014), kenaikan pangkat adalah penghargaan yang diberikan atas prestasi kerja dan pengabdian Pegawai Negeri Sipil terhadap Negara, serta sebagai dorongan kepada Pegawai Negeri Sipil untuk lebih meningkatkan prestasi kerja dan pengabdianya.

Berdasarkan pengertian di atas penulis menyimpulkan bahwa kenaikan pangkat adalah perubahan pangkat yang diberikan kepada PNS atas prestasi dan pengabdianya.

2.2.4. Pengertian Pegawai

Setiawan (2016), pegawai adalah orang yang bekerja pada pemerintah (perusahaan, dan sebagainya).

Subekti (2016), pegawai adalah orang pribadi yang bekerja pada pemberi kerja, berdasarkan perjanjian atau kesepakatan kerja baik secara tertulis maupun tidak tertulis, untuk melaksanakan suatu pekerjaan dalam jabatan atau kegiatan tertentu dengan memperoleh imbalan yang dibayarkan berdasarkan periode tertentu, penyelesaian pekerjaan, atau ketentuan lain yang ditetapkan pemberi kerja, termasuk orang pribadi yang melakukan pekerjaan dalam jabatan negeri.

Berdasarkan pengertian di atas penulis menyimpulkan bahwa pegawai adalah orang yang berkerja pada pemberi kerja berdasarkan perjanjian tertentu.



2.2.5. Pengertian Web

Sujatmiko (2012:317), *web* adalah salah satu aplikasi internet yang terdiri dari perangkat lunak, kumpulan protokol, dan seperangkat aturan yang memungkinkan kita untuk mengakses informasi di internet.

Sora (2014), *web* diartikan suatu kumpulan-kumpulan halaman yang menampilkan berbagai macam informasi teks, data, gambar diam ataupun bergerak, data animasi, suara, video maupun gabungan dari semuanya, baik itu yang bersifat statis maupun yang dinamis, yang dimana membentuk satu rangkaian bangunan yang saling berkaitan dimana masing-masing dihubungkan dengan jaringan halaman atau *hyperlink*.

Berdasarkan pengertian di atas penulis menyimpulkan bahwa *web* adalah suatu aplikasi yang memungkinkan kita untuk mengakses informasi di internet.

2.2.6. Pengertian Aplikasi Pengolahan Data Kenaikan Pangkat Pegawai pada Kantor Camat Ilir Barat I Palembang Berbasis Web

Aplikasi Pengolahan Data Kenaikan Pangkat Pegawai pada Kantor Camat Ilir Barat I Palembang Berbasis *Web* adalah sebuah aplikasi khusus berbasis *web* yang dibangun untuk mempermudah Sub Bagian Kepegawaian dalam mengolah data kenaikan pangkat pegawai serta pembuatan lembar Sasaran Kerja Pegawai (SKP) yang berguna sebagai persyaratan berkas dalam proses kenaikan pangkat pegawai di Kantor Camat Ilir Barat I Palembang.

2.3. Teori Khusus

2.3.1. Kamus Data

2.3.1.1. Pengertian Kamus Data

Rosa dan Shalahuddin (2015: 73), kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan).

Ladjamudin (2013: 70), kamus data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi.



Berdasarkan pengertian di atas penulis menyimpulkan bahwa kamus data adalah kumpulan daftar elemen data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi yang mengalir pada sistem perangkat lunak

2.3.1.2. Simbol Kamus Data

Rosa dan Shalahuddin (2015: 74), simbol-simbol kamus data, sebagai berikut:

Tabel 2.1. Simbol Kamus Data

| Simbol | Keterangan |
|------------------|----------------------------------|
| = | Disusun atau terdiri dari |
| + | Dan |
| [] | Baik...atau... |
| { } ⁿ | N kali diulang / bernilai banyak |
| () | Data opsional |
| * _* | Batas komentar |

2.3.2. Data Flow Diagram (DFD)

2.3.2.1. Pengertian Data Flow Diagram (DFD)

Rosa dan Shalahuddin (2015: 70), *data flow diagram* (DFD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).

Ladjamudin (2013: 64), DFD merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil.

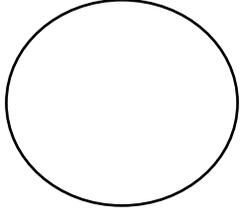
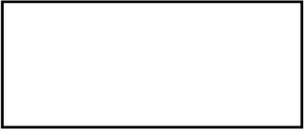
Berdasarkan pengertian di atas penulis menyimpulkan bahwa DFD adalah model dari sistem yang menggambarkan aliran informasi dan transformasi data.

2.3.2.2. Notasi Data Flow Diagram (DFD)

Supardi (2013: 8-9), notasi-notasi pada DFD (Edward Yourdon & Tom De Marco) sebagai berikut.



Tabel 2.2. Notasi DFD

| Notasi | Keterangan |
|---|---|
|  | <p>Proses (<i>Process</i>) atau fungsi (<i>function</i>) atau prosedur (<i>procedure</i>). Pada pemrograman terstruktur, notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode pemrograman. Catatan: nama proses biasanya berupa kata kerja.</p> |
|  | <p><i>File</i> atau basis data atau penyimpanan (<i>storage</i>) pada pemrograman terstruktur notasi inilah yang harusnya menjadi tabel-tabel basis data yang dibutuhkan. Tabel-tabel ini harus sesuai dengan ERD (<i>Entity Relationship Diagram</i>), CDM (<i>Conceptual Data Model</i>), PDM (<i>Physical Data Model</i>). Catatan: Nama penyimpanan biasanya berupa kata benda.</p> |
|  | <p>Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.</p> |
|  | <p>Aliran data merupakan data yang dikirim antar-proses penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>). Catatan: Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data. Misalnya “data siswa” atau tanpa kata data, misalnya “Siswa”.</p> |

2.3.2.3. Larangan dalam DFD

Ladjamudin (2013: 75), dalam menggambar/mendesain DFD ada beberapa hal yang harus dihindari, sehingga DFD tersebut menggambarkan secara keseluruhan *system* yang akan dirancang, hal-hal tersebut adalah:

- a. Arus data tidak boleh dari entitas luar langsung menuju entitas luar lainnya, tanpa melalui suatu proses.



- b. Arus data tidak boleh dari simpanan data langsung menuju ke entitas luar, tanpa melalui suatu proses.
- c. Arus data tidak boleh dari simpanan data langsung menuju ke simpanan data lainnya, tanpa melalui proses.
- d. Arus data dari suatu proses langsung menuju proses lainnya, tanpa melalui suatu simpanan data, sebaiknya/sebisa mungkin dihindari.

2.3.3. Blockchart

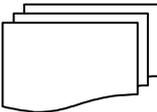
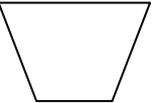
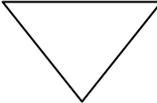
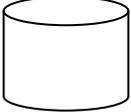
2.3.3.1. Pengertian Blockchart

Kristanto (2008: 75), *blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu.

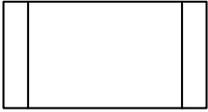
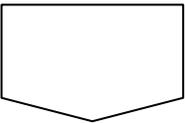
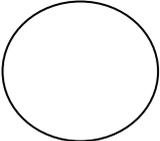
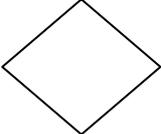
2.3.3.2. Simbol-Simbol Blockchart

Kristanto (2008: 75-77), adapun simbol-simbol yang sering digunakan dalam *blockchart* adalah sebagai berikut:

Tabel 2.3. Simbol *Blockchart*

| Simbol | Arti |
|---|--|
|  | Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan |
|  | Multi dokumen |
|  | Proses manual |
|  | Proses yang dilakukan oleh komputer |
|  | Menandakan dokumen yang diarsipkan (arsip manual) |
|  | Data penyimpanan (<i>data storage</i>) |

Lanjutan Tabel 2.3 Simbol *Blockchart*

| Simbol | Arti |
|---|---|
|  | Proses apa saja yang tidak terdefiniskan termasuk aktivitas fisik |
|  | Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain |
|  | Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama |
|  | Terminasi yang menandakan awal dan akhir dari suatu aliran |
|  | Pengambilan keputusan (<i>decision</i>) |
|  | Layar peraga (<i>monitor</i>) |
|  | Pemasukan data secara manual |

2.3.4. *Flowchart* (Diagram Alur)

2.3.4.1. Pengertian *Flowchart* (Diagram Alur)

Supardi (2013: 51), bagan alir (*flowchart*) merupakan bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika.



2.3.4.2. Simbol *Flowchart*

Supardi (2013: 58-59), simbol-simbol *flowchart*, sebagai berikut berikut:

Tabel 2.4. Simbol *Flowchart*

| Simbol | Keterangan |
|--------|--|
| | Simbol input/output (<i>input / output symbol</i>) digunakan untuk mewakili data <i>input/output</i>) |
| | Simbol proses digunakan untuk mewakili suatu proses |
| | Simbol garis alir (<i>flow lines symbol</i>) digunakan untuk menunjukkan arus dari proses |
| | Simbol penghubung (<i>connector symbol</i>) digunakan untuk menunjukkan sambungan dari bagan alir yang terputus di halaman yang masih sama atau di halaman lainnya |
| | Simbol keputusan (<i>decision symbol</i>) digunakan untuk suatu penyeleksian kondisi di dalam program |
| | Simbol proses terdefinisi (<i>predigined process symbol</i>) digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain |
| | Simbol persiapan (<i>preparation symbol</i>) digunakan untuk memberi nilai awal suatu besaran |
| | Simbol titik terminal (<i>terminal point symbol</i>) digunakan untuk menunjukkan awal dan akhir dari suatu proses |



2.3.5. Entity Relationship Diagram (ERD)

2.3.5.1. Pengertian Entity Relationship Diagram (ERD)

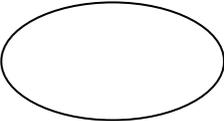
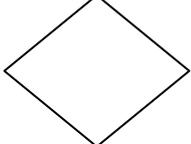
Rosa dan Shalahuddin (2015: 53), ERD adalah bentuk paling awal dalam melakukan perancangan basis data relasional.

Supardi (2013:13), ERD merupakan model konseptual yang mendeskripsikan hubungan antar-penyimpanan (dalam DFD).

2.3.5.2. Notasi ERD

Supardi (2013:14), pada dasarnya ada tiga macam simbol ERD yang digunakan, yaitu:

Tabel 2.5. Notasi ERD

| Notasi | Keterangan |
|---|--|
|  | Entitas (<i>Entity</i>) dalam ERD disimbolkan persegi panjang, entitas merupakan data inti. Pengertian lain <i>entity</i> adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai. |
|  | Atribut (<i>Attribute</i>) disimbolkan oval dalam ERD. Atribut merupakan elemen dari <i>entity</i> , dan berfungsi mendeskripsikan karakter ini. Atribut juga dapat diartikan <i>field</i> atau kolom. |
|  | Hubungan (<i>Relation</i>) disimbolkan belah ketupat (<i>diamond</i>). Relasi yang menghubungkan antar-entitas, biasanya diawali kata kerja. Dalam ERD, hubungan ini dapat terjadi atas sejumlah <i>entity</i> yang disebut sebagai derajat hubungan, tetapi pada umumnya hampir semua model hanya menggunakan hubungan dengan derajat dua (<i>binary-relationship</i>). |

2.3.5.3. Jenis Relasi ERD

Supardi (2013: 16-17), dalam hubungan antar-*Entity* dibagi menjadi tiga sebagai berikut:

a. Satu ke Satu (*One to One*)

ERD untuk jenis hubungan satu ke satu antar-*entity* adalah *entity* yang berhubungan tepat maksimal satu masing-masing *entity*, dan sebaliknya.

b. Satu ke Banyak atau Banyak ke Satu (*One to Many/Many to One*)



ERD untuk jenis hubungan satu ke banyak atau banyak ke satu antar-*entity* adalah *entity* yang berhubungan maksimal satu ke maksimal banyak *entity*, dan sebaliknya.

c. Banyak ke Banyak (*Many to Many*)

ERD untuk jenis hubungan banyak ke banyak adalah *entity* yang berhubungan maksimal satu ke maksimal banyak *entity*, dan maksimal satu ke maksimal banyak.

2.4. Teori Program

2.4.1. XAMPP

Riyanto (2015:1), XAMPP merupakan paket PHP dan MySQL berbasis *open source*, yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis PHP.

Wahana Komputer (2014: 72), XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP, dan Perl. XAMPP adalah *tool* yang menyediakan paket perangkat lunak dalam satu buah paket.

Sidik (2014: 72), XAMPP merupakan paket *server web* PHP dan database MySQL yang paling populer di kalangan pengembang *web* dengan menggunakan PHP dan MySQL sebagai *databasenya*.

2.4.2. PHPMyAdmin

2.4.2.1. Pengertian PHPMyAdmin

Wahana Komputer (2014: 46), *PhpMyAdmin* adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi MySQL melalui *world wide web*.

2.4.2.2. Cara Mengaktifkan PHPMyAdmin

Wahana Komputer (2014: 74), untuk mengaktifkan *PhpMyAdmin*, langkah-langkahnya sebagai berikut:

1. Setelah XAMPP Anda terinstal, Anda harus mengaktifkan *web server* Apache dan MySQL dari *control panel* XAMPP.
2. Jalankan *browser* kesayangan Anda (IE, Mozilla Firefox, atau Opera), lalu ketikkan alamat *web* <http://localhost/phpmyadmin/> pada *address bar*, kemudian tekan **Enter**.
3. Apabila telah tampil *interface* (tampilan antarmuka) *PhpMyAdmin*, Anda bisa memulainya dengan mengetikkan nama *database*, nama tabel, dan seterusnya.



2.4.3. MySQL (*My Structured Query Language*)

Rosa dan Shalahuddin (2015: 46), SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada RDBMS.

Wahana Komputer (2014: 73), SQL merupakan bahasa terstruktur yang khusus digunakan untuk mengolah *database*.

Hendry (2015: 7), SQL adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah dan secara otomatis.

2.4.4. *Hypertext Preprocessor* (PHP)

2.4.4.1. Pengertian PHP

Wahana Komputer (2014: 73), bahasa pemrograman PHP merupakan bahasa pemrograman untuk membuat *website* yang bersifat *server-side scripting*.

2.4.4.2. Struktur PHP

Wahana Komputer (2014: 33-34), dalam penulisannya, instruksi-instruksi PHP diawali dengan *tag* `<?php` dan diakhiri dengan *tag* `?>`. Dengan adanya *tag* ini, *web server* akan menerjemahkan semua instruksi yang ditulis di dalamnya sebagai kode instruksi PHP. Contoh dari struktur PHP adalah sebagai berikut:

```
<html>
  <head>
    <title>contoh script php dalam dokumen html </title>
  </head>
  <body>
    <?php
      echo "<center>halo, kalimat ini dibuat dengan script
      php</center>";
    ?>
  </body>
</html>
```

Karena mengandung instruksi PHP, dokumen ini harus disimpan dengan format *file* PHP, yaitu dengan menggunakan ekstensi `.php`.



2.4.5. *Hyper Text Markup Language* (HTML)

2.4.5.1. Pengertian HTML

Wahana Komputer (2014: 2), HTML merupakan bahasa (kode) yang digunakan untuk membuat halaman *web*.

2.4.5.2. Struktur Dasar HTML

Wahana Komputer (2014: 3-4), struktur dasar sebuah dokumen HTML dapat dijelaskan dengan contoh sebagai berikut:

```
<!doctype html>
<html>
  <head>
    <title> struktur dasar html </title>
  </head>
  <body>
    isi dokumen html di sini
  </body>
</html>
```

Penjelasan dari struktur kode HTML di atas adalah sebagai berikut:

- a. Setiap dokumen HTML harus diawali *tag* `<html>` dan diakhiri *tag* `</html>`. *Tag* `<html>` adalah *tag* pembuka dokumen HTML. *Tag* `</html>` adalah *tag* penutup dokumen HTML.
- b. Bagian `<head>` merupakan kepala dokumen HTML. Bagian ini menyimpan metadata pada dokumen HTML. Format penulisannya adalah diawali *tag* `<head>` dan diakhiri `</head>`. *Tag* `<head>` adalah *tag* pembuka untuk kepala dokumen HTML. *Tag* `</head>` adalah *tag* penutup untuk kepala dokumen HTML.
- c. Judul dokumen akan ditampilkan pada bagian kiri atas jendela *web browser*. Cara penulisan judul dokumen adalah diawali *tag* `<title>` dan diakhiri `</title>`. Bagian ini termasuk golongan metadata sebuah dokumen HTML, sehingga harus diletakkan di antara *tag* `<head>` dan `</head>`.
- d. Bagian `<body>` dokumen HTML adalah tempat di mana seluruh isi dokumen HTML yang hendak ditampilkan pada *web browser* dituliskan. Format penulisannya adalah diawali *tag* `<body>` dan diakhiri `</body>`.

2.4.6. *Notepad++*

2.4.6.1. Pengertian *Notepad++*

Wahana Komputer (2014: 69), *notepad++* adalah sebuah penyunting teks dan penyunting kode sumber yang berjalan di sistem operasi *Windows*.



2.4.6.2. Fungsi Notepad++

Wahana Komputer (2014: 70-71), fungsi dari *Notepad++* yaitu:

1. *Notepad* sebagai *digital diary*. Anda dapat menggunakan *notepad diary digital* dan secara otomatis memasukkan tanggal dan informasi waktu untuk setiap baris yang Anda ketik di *notepad*. Sebagai contoh, buka *file notepad* baru ketik *.LOG* di bagian atas. Kemudian ketikkan beberapa baris seperti contoh di atas dan simpan. Buka *file* tersebut dan Anda akan melihat tanggal dan waktu di setiap baris teks *notepad*. Anda dapat menambahkan waktu secara manual dengan menekan tombol **F5**. Hasilnya, secara otomatis akan menambah waktu dengan sendirinya.
2. *Notepad* sebagai *HTML stripper*. Idealnya, teks pada halaman sebuah *web* diformat khusus untuk jenis *font*, warna ukuran, elemen desain, dan gambar. Dengan *notepad*, Anda dapat mengedit komponen teks saja. Anda tinggal menyalin dari kode HTML yang ada di halaman *web* dan Anda *paste* di *notepad* yang kemudian dapat Anda simpan untuk digunakan lagi di lain waktu.
3. *Notepad* sebagai pembuat *script* baik maupun *script* jahat. Seperti yang sudah dibahas bahwa *notepad* merupakan editor teks yang sederhana yang dapat digunakan untuk membuat *script* yang kompleks. Anda dapat membuat *script* yang jahat seperti *script* memformat *harddisk* maupun *script* untuk men-shutdown, membuat virus, dan masih banyak lagi.
4. *Notepad* sebagai jalan pintas *Windows Explorer*. Terkadang ada beberapa *file* yang tidak dapat dihapus. Anda dapat mencoba menggunakan *notepad* sebagai jalan pintas pengganti *Windows Explorer* untuk menghapus *file* tersebut. Anda bisa mencoba **File > Open >** pada **Type file** pilih **All**. Anda tinggal mencari *file* yang ingin dihapus.
5. Bisa digunakan membuat/mengedit *file* berekstensi *.bat* atau *.cmd* untuk *Windows 2000* ke atas. *Bat* atau *cmd* adalah aplikasi under *DOS* yang bisa digunakan bermacam-macam fungsi tergantung *command/perintah* yang diketik di *notepad*. Misalnya, buat perintah **shutdown**.
6. Membuat *file* berekstensi *.inf*. *Inf* adalah *file* yang digunakan untuk install driver *hardware*, yang sekarang menjadi multifungsi. Ekstensi ini bisa jadi virus atau antivirus, tergantung isinya. Misalnya *autorun.inf*, yang sebenarnya dulu digunakan untuk *CD*. Bila dimasukkan ke komputer akan langsung menjalankan aplikasi tertentu. Tetapi, sekarang digunakan untuk menjalankan virus di *flashdisk*.
7. Membuat/mengedit *file* berekstensi *.reg*. *Reg* adalah *file* yang digunakan untuk memasukkan pengaturan *Windows* melalui *registry Windows (regedit.exe)*.