



BAB II

LANDASAN TEORI

2.1. Teori Umum

2.1.1. Pengertian Komputer

Sujatmiko (2012:156) menyatakan, “Komputer adalah mesin yang dapat mengolah data digital dengan mengikuti serangkaian perintah atau program”.

Hartono (2013:27) menyatakan, “Komputer adalah sebuah mesin yang dapat dikendalikan melalui perintah (*programmable machine*) yang dirancang untuk secara otomatis melakukan serangkaian urutan penghitungan (*arithmetic*) atau proses–proses yang diurutkan secara logis”.

Irwansyah dan Moniaga (2014:2) menyatakan, “Komputer merupakan perangkat elektronik, yang beroperasi di bawah kendali instruksi yang memiliki memori, dapat menerima dan mengolah data sesuai dengan aturan yang diterapkan, menghasilkan hasil atau informasi dan menyimpan hasil tersebut untuk digunakan di masa depan atau di kemudian hari”.

Dari pengertian diatas penulis menyimpulkan bahwa komputer adalah sebuah mesin yang dapat mengolah data, menyimpan data, dan dapat melakukan proses perhitungan yang rumit atau proses-proses lainnya.

2.1.2. Pengertian Sistem

Kristanto (2011:1) menyatakan “Sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu.

Sutabri (2012:10) menyatakan, “Sistem adalah suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain”.

Ladjamudin (2013:6) menyatakan, “Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lainnya”.



Dari pengertian diatas penulis menyimpulkan bahwa sistem merupakan suatu jaringan atau prosedur-prosedur yang saling berinteraksi, berhubungan, dan saling bergantung antar komponen satu dengan yang lain.

2.1.3. Karakteristik Sistem

Sutabri (2012:20-21) menjelaskan bahwa sistem memiliki karakteristik atau sifat-sifat tertentu yang yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud, yaitu:

1. **Komponen Sistem (*Components*)**
Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk suatu kesatuan.
2. **Batasan Sistem (*Boundary*)**
Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisah.
3. **Lingkungan Luar Sistem (*Environment*)**
Lingkungan luar sistem merupakan bentuk apapun yang ada di luar sistem yang akan mempengaruhi operasi sistem.
4. **Penghubung Sistem (*Interface*)**
Media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem atau interface.
5. **Masukan Sistem (*Input*)**
Energi yang dimasukkan ke dalam sistem disebut masukan sistem yang dapat berupa pemeliharaan (*maintanance input*) dan sinyal (*signal input*).
6. **Keluaran Sistem (*Output*)**
Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain.
7. **Pengolahan Sistem (*Process*)**
Suatu sistem mempunyai proses yang akan mengubah masukan menjadi keluaran.



8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki sasaran dan tujuan yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dapat dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.1.4. Metode Pengembangan Sistem

Sukanto dan Shalahudin (2013:28) menjelaskan tentang metode pengembangan sistem yaitu *waterfall*. Metode air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linier*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup yang terurut, mulai dari analisis, desain, pengkodean, pengujian, dan pemeliharaan.

1. Analisis

Tahap analisis dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami sistem seperti apa yang dibutuhkan.

2. Desain

Tahap desain adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur sistem, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

3. Pengkodean

Pada tahap pengkodean, desain harus ditranslasikan ke dalam program sistem. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

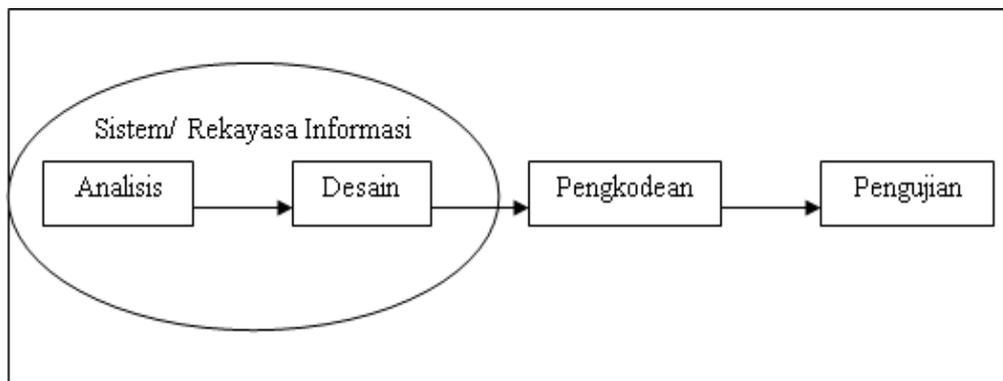
4. Pengujian

Tahap pengujian berfokus pada sistem dari segi logika dan fungsional serta memastikan bahwa semua bagian telah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan bahwa keluaran yang dihasilkan sesuai dengan yang diinginkan.



5. Pemeliharaan

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



(Sumber : Sukanto dan Shalahudin, 2013:28)

Gambar 2.1 Ilustrasi Model Waterfall

2.1.5. Pengertian Data

Kristanto (2011:7) menyatakan, “Data merupakan penggambaran dari suatu kejadian yang sedang terjadi, dimana data tersebut akan diolah dan diterapkan dalam sistem menjadi input yang berguna dalam suatu sistem”.

Ladjamudin (2013:8) menyatakan, “Data merupakan kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata.”

Sutabri (2012:3) menyatakan, “Data adalah hal, peristiwa, atau kenyataan lain apapun yang mengandung sesuatu pengetahuan untuk dijadikan dasar guna penyusunan keterangan, pembuatan kesimpulan, atau penetapan keputusan”.

Dari pengertian diatas penulis menyimpulkan bahwa data merupakan suatu keadaan yang menggambarkan suatu kejadian yang sedang terjadi, dan dapat diolah menjadi suatu informasi yang berguna.



2.1.6. Pengertian Basis Data (Database)

Sujatmiko (2012:40) menyatakan, “Basis data (*database*) merupakan kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut”.

Sutabri (2012:47) menyatakan, “Basis data (*database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu sama lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya”.

Sukanto dan Shalahuddin (2014:43) menyatakan, “Basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan”.

Dari pengertian diatas penulis menyimpulkan bahwa basis data (*database*) adalah kumpulan dari beberapa data yang saling berhubungan yang disimpan pada suatu media penyimpanan secara digital.

2.1.7. Pengertian Aplikasi

Sujatmiko (2012:23) menyatakan, “Aplikasi adalah program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas–tugas tertentu, misalnya MS-Word, MS-Excel”.

Sutabri (2012:147) menyatakan, “Aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya”.

Dari pengertian diatas penulis menyimpulkan bahwa aplikasi adalah suatu perangkat lunak komputer yang memanfaatkan kemampuan komputer untuk melakukan suatu tugas yang diinginkan.

2.1.8. Pengertian Pengolahan Data

Kristanto (2011:8) menyatakan, “Pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan.”



Sutabri (2012:22) menyatakan, “Pengolahan data adalah proses perubahan data menjadi sebuah informasi”.

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

Dari pengertian diatas penulis menyimpulkan bahwa pengolahan data adalah suatu proses yang menggambarkan perubahan data menjadi informasi yang lebih berguna.

2.1.9. Pengertian Akta Tanah

Salim (2016:8) menyatakan, “Akta tanah adalah surat tanda bukti berisi pernyataan (keterangan, pengakuan, keputusan) resmi yang dibuat menurut peraturan yang berlaku, disaksikan dan disahkan oleh notaris atau pejabat pemerintah yang berwenang”.

2.1.10. Pengertian PPAT (Pejabat Pembuat Akta Tanah)

Salim (2016:8) menyatakan, “PPAT merupakan pejabat yang diberi kewenangan untuk membuat akta yang berkaitan dengan peralihan, pemindahan, dan pembebanan terhadap hak atas tanah dan/atau hak milik atas satuan rumah susun”.

2.1.11. Pengertian Aplikasi Pengolahan Data Pembuatan Akta Tanah pada Kantor PPAT (Pejabat Pembuat Akta Tanah) Muhammad Zaini, S.H di Palembang

Aplikasi Pengolahan Data Pembuatan Akta Tanah pada Kantor PPAT (Pejabat Pembuat Akta Tanah) Muhammad Zaini, S.H di Palembang adalah suatu aplikasi yang dapat membantu mempermudah klien dalam hal pencarian informasi dalam proses pembuatan akta tanah, serta informasi mengenai perkembangan akta yang mereka buat dan mempermudah Staf Administrasi PPAT Muhammad Zaini, S.H dalam hal pengolahan data pembuatan akta tanah seperti penginputan, penyimpanan, dan pencarian data.



2.2. Teori Khusus

2.2.1. Pengertian *Data Flow Diagram* (DFD)

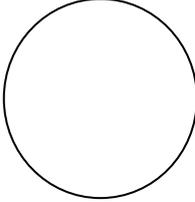
Kristanto (2011:55) menyatakan, “*Data flow diagram*/DFD adalah suatu model logika atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut”.

Sukanto dan Shalahuddin (2013:70) menyatakan, “*Data Flow Diagram* atau dalam bahasa Indonesia menjadi diagram alir data adalah representatif grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*)”.

Adapun simbol–simbol atau notasi–notasi yang menggambarkan *Data Flow Diagram* (DFD), sebagai berikut:

1. Teknik Edward Yourdon dan Tom DeMarco

Tabel 2.1 Simbol–simbol *Data Flow Diagram* (DFD) menurut Edward Yourdon dan Tom DeMarco

Notasi	Keterangan
	Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harus menjadi fungsi atau prosedur di dalam kode program.
	File atau basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel–tabel basis data yang dibutuhkan.

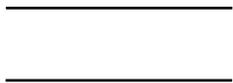


Notasi	Keterangan
	Entitas luar (<i>External Entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.
	Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>).

(Sumber: Sukamto dan Shalahuddin, 2014:71-72)

2. Teknik Gane dan Sarson

Tabel 2.2 Simbol- Simbol *Data Flow Diagram* (DFD) Gane & Sarson

Notasi	Keterangan
	<i>External Entity</i> , yaitu merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan <i>input</i> atau menerima <i>output</i> dari sistem.
	<i>Data Flow</i> , digunakan untuk menggambarkan aliran data dari satu proses ke proses lainnya.
	<i>Data Store</i> , digunakan untuk menyimpan data seperti: suatu <i>file</i> , suatu arsip, suatu kotak, suatu tabel dan suatu agenda.

(Sumber: Kristanto, 2011:56-58)

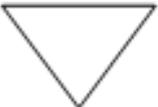


2.2.2. Pengertian *Blockchart*

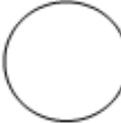
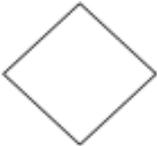
Kristanto (2011:68) menyatakan “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi”.

Simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:

Tabel 2.3 Simbol– simbol dalam *Blokchart*

No	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan.
2.		Multi dokumen.
3.		Proses Manual.
4.		Proses yang dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan(<i>data storage</i>)
7.		Proses apa saja yang tidak terdefinisi termasuk aktifitas fisik.

Lanjutan Tabel 2.3 Simbol– simbol dalam *Blokchart*

No	Simbol	Keterangan
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.
9.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
10.		Terminasi yang menandakan awal dan akhir dari suatu aliran
11.		Pengambilan keputusan (<i>decision</i>).
12.		Layar peraga (<i>monitor</i>)
13.		Pemasukan secara manual.

(Sumber: Kristanto, 2011:68-70)

2.2.3. Pengertian *Flowchart*

Sujatmiko (2012:106) menyatakan, “*Flowchart* merupakan suatu diagram operasi komputer, oran, atau sistem akuntansi yang melibatkan sistem atau aktivitas yang kompleks”.

Ladjamudin (2013:263) menyatakan, “*Flowchart* adalah bagan–bagan yang mempunyai arus yang menggambarkan langkah–langkah penyelesaian suatu masalah”.



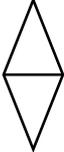
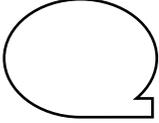
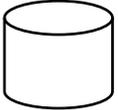
Supardi (2013:51), “*Flowchart* merupakan Diagram Alur yang sering digunakan sistem analis dalam membuat atau menggambarkan logika program.”

Simbol-simbol yang sering digunakan dalam *flowchart* dapat dilihat pada tabel berikut ini:

Tabel 2.4 Simbol-simbol dalam *Flowchart*

No	Simbol	Keterangan
1.		Simbol dokumen yang menunjukkan dokumen input dan output baik untuk proses manual, mekanik atau komputer.
2.		Simbol kegiatan manual yang menunjukkan pekerjaan manual.
3.		Simbol simpanan offline yang berupa file non komputer yang diarsip urut angka (<i>numerical</i>)
4.		Simbol simpanan offline yang berupa file non-komputer yang diarsip urut huruf (<i>alphabetical</i>).
6.		Simbol kartu plong yang menunjukkan <i>input/output</i> yang menggunakan kartu plong (<i>punched card</i>).
7.		Simbol proses yang menunjukkan kegiatan proses dari operasi program komputer.
8.		Simbol operasi luar luar yang menunjukkan operasi yang dilakukan di luar porses operasi komputer.

Lanjutan Tabel 2.4 Simbol-simbol dalam *Flowchart*

No	Simbol	Keterangan
9.		Simbol pengurutan offline yang menunjukkan proses pengurutan data dari luar proses komputer.
10.		Simbol pita magnetik yang menunjukkan bahwa <i>input/output</i> menggunakan pita magnetik.
11.		Simbol hard disk yang menunjukkan input/output menggunakan hard disk.
12.		Simbol diskette yang menunjukkan input/output menggunakan diskette.
13.		Simbol drum magnetik yang menunjukkan bahwa input/output menggunakan drum magnetik.
14.		Simbol pita kertas terhubung yang menunjukkan bahwa input/output menggunakan pita kertas terhubung.
15.		Simbol keyboard yang menunjukkan input/output menggunakan on-line keyboard
16.		Simbol display yang menunjukkan output yang ditampilkan di monitor.

Lanjutan Tabel 2.4 Simbol-simbol dalam *Flowchart*

No	Simbol	Keterangan
17.		Simbol pita kontrol yang menunjukkan penggunaan pita kontrol (control tape) dalam batch control total untuk pencocokan di proses batch processing.
18.		Simbol hubungan komunikasi yang menunjukkan proses transmisi data melalui channel komunikasi.
19.		Simbol penjelasan yang menunjukkan penjelasan dari suatu proses.
20.		Simbol penghubung yang menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain.
21.		Simbol input/output yang digunakan untuk mewakili data input/output.
22.		Simbol garis alir yang digunakan untuk menunjukkan arus dari proses
23.		Simbol keputusan (decision symbol) yang digunakan untuk suatu penyeleksian kondisi di dalam program

Lanjutan Tabel 2.4 Simbol-simbol dalam *Flowchart*

24.		Simbol proses terdefinisi yang digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain.
25.		Simbol persiapan yang digunakan untuk memberi nilai awal suatu besaran.
26.		Simbol titik terminal yang digunakan untuk menunjukkan awal dan akhir dari suatu proses.

(Sumber: Supardi, 2013:53)

2.2.4. Pengertian *Entity Relationship Diagram (ERD)*

Sukanto dan Shalahuddin (2013:50) menyatakan “*Entity Relationship Diagram (ERD)* merupakan bentuk paling awal dalam melakukan perancangan basis data relasional”.

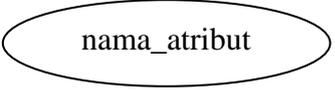
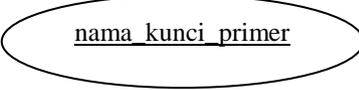
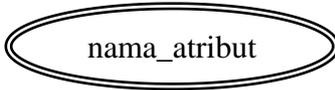
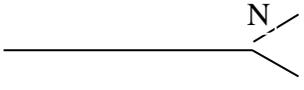
Ladjamudin (2013:142) menyatakan, “*Entity Relationship Diagram (ERD)* adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam system secara abstrak”.

Adapun simbol-simbol yang digunakan dalam *Entity Relationship Diagram (ERD)*, yaitu:

Tabel 2.5 Simbol–Simbol *Entity Relationship Diagram (ERD)*

No	Simbol	Deskripsi
1	Entitas/ <i>Entity</i> 	Entitas merupakan data inti yang akan tersimpan; 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.

Lanjutan Tabel 2.5 Simbol–Simbol *Entity Relationship Diagram* (ERD)

No	Simbol	Deskripsi
2	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
3	Atribut Kunci Primer 	<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).
4	AtributMultinilai/ <i>Multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
6	Asosiasi/ <i>Association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya punya <i>multiplicity</i> ke-mungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas yang lain disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B.

(Sumber: Sukamto dan Shalahuddin, 2013:50-51)



2.2.5. Pengertian Kamus Data (*Data Dictionary*)

Sukamto dan Shalahuddin (2013:73) menyatakan, “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 sering disebut juga dengan sistem data *dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi.”

Kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut:

Tabel 2.6 Simbol-simbol Kamus Data (*Data Dictionary*)

No	Simbol	Keterangan
1.	=	Disusun atau terdiri dari
2.	+	Dan
3.	[]	Baik ... atau...
4.	{ } ⁿ	n kali diulang/bernilai banyak
5.	()	Data opsional
6.	*...*	Batas komentar

(Sumber : Sukamto dan Shalahuddin, 2013:74)

2.3. Teori Program

2.3.1. Sekilas Tentang PHP

2.3.1.1. Pengertian PHP

Kadir (2008:2) menyatakan, menurut dokumen resmi PHP, PHP merupakan singkatan dari PHP Hypertext Preprocessor. Ia merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan diproses di server. Hasilnyalah yang dikirimkan ke klien tempat pemakai menggunakan browser.

Sidik (2012:5) menyatakan, PHP adalah kependekan dari PHP: Hypertext Preprocessor (rekursif, mengikut gaya penamaan di *nix), merupakan bahasa utama script server-side yang disisipkan pada HTML yang dijalankan diserver, dan juga bisa digunakan untuk membuat aplikasi dekstop”.



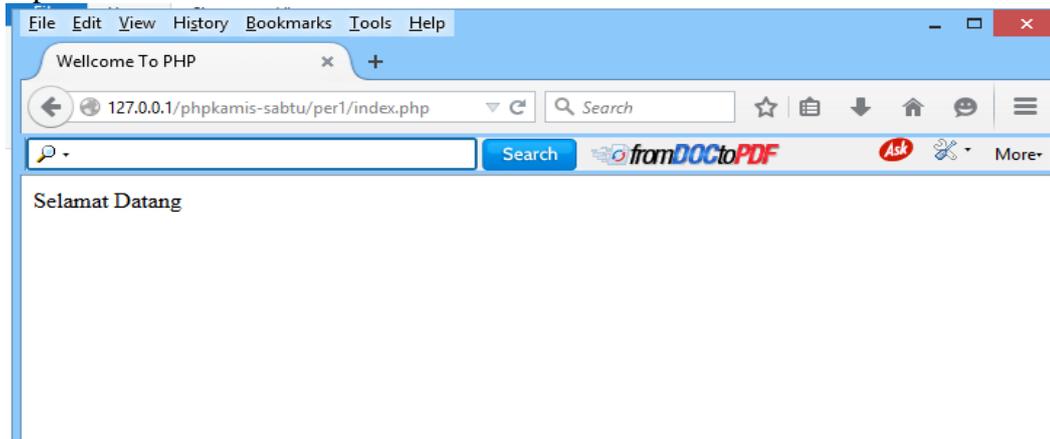
2.3.1.2. Script PHP

Script PHP berkedudukan sebagai tag dalam bahasa *HTML*. Sebagaimana diketahui, *HTML* (*Hypertext Markup Language*) adalah bahasa standar untuk membuat halaman-halaman *web*.

Sebagai contoh, berikut adalah kode *HTML* (disimpan dengan ekstensi *.htm* atau *.html*):

```
<html>
  <head>
    <title>Wellcome To PHP</title>
  </head>
  <body>
    Selamat Datang
  </body>
</html>
```

Bila dijalankan melalui *browser*, kode diatas akan menampilkan hasil seperti berikut:



Gambar 2.2 Tampilan Program HTML pada *browser*



2.3.2. Sekilas Tentang MySQL

2.3.2.1. Pengertian MySQL

Raharjo (2015:16) menyatakan, “MySQL merupakan software RDBMS (atau server database) yang dapat mengelola database dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak user (multi-user), dan dapat melakukan suatu proses secara sinkron atau berbarengan (multi-threaded)”.

Madcoms (2016:17) menyatakan, ”MySQL adalah sistem manajemen database SQL yang bersifat Open Source dan paling populer saat ini”.

2.3.2.2. Kelebihan MySQL

Menurut (Raharjo:16) menyebutkan bahwa *MySQL* memiliki beberapa kelebihan sebagai *server database* antara lain sebagai berikut;

1. Fleksibel

MySQL dapat digunakan untuk mengembangkan aplikasi dekstop maupun aplikasi web dengan menggunakan teknologi yang bervariasi.

2. Performa Tinggi

MySQL memiliki mesin query dengan performa tinggi, dengan demikian proses transaksional dapat dilakukan dengan sangat cepat.

3. Lintas Platform

MySQL dapat digunakan pada platform atau lingkungan (dalam hal ini Sistem Operasi) yang beragam seperti Windows, Linux, dan UNIX.

4. Gratis

MySQL dapat dengan mudah di dapatkan karena bersifat open source.

5. Proteksi Data yang Andal

MySQL menyediakan mekanisme yang powerful untuk menangani dalam hal keamanan data seperti menyediakan fasilitas manajemen user, enkripsi data, dan lain sebagainya.



2.3.3. Sekilas Tentang Notepad ++

2.3.3.1. Pengertian Notepad++

Madcoms (2016:3) menyatakan, “Notepad ++ adalah sebuah text editor yang sangat berguna bagi setiap orang dan khususnya bagi developer dalam membuat program”.

2.3.3.2. Kelebihan Notepad++

Madcoms (2016:3) menyebutkan beberapa kelebihan yang dimiliki notepad++ antara lain;

1. Notepad++ mampu menangani banyak bahasa pemrograman seperti bahasa C, C++, Java, C#, SML, HTML, PHP, Javascript, dan masih banyak lagi bahasa lainnya.
2. Notepad++ bersifat open source yaitu bisa didapatkan secara mudah dan gratis.
3. Notepad++ juga mempunyai beberapa fitur yang sangat berguna seperti fitur highlighting yang berguna untuk menandai sintaks dan variabel yang digunakan dalam source code. Selain itu terdapat fitur tab yang dapat membantu mengelola beberapa kode dalam waktu yang bersamaan.

2.3.4. Sekilas Tentang Xampp

Sidik (2012:72) menyatakan, “XAMPP (X(Windows/Linux) Apache MySQL PHP dan Perl) merupakan paket *server web* PHP dan database MySQL yang paling populer di kalangan pengembang *web* dengan menggunakan PHP dan MySQL sebagai databasenya”.

Sebelum menjalankan XAMPP (X(Windows/Linux) Apache MySQL PHP dan Perl) yang harus dilakukan adalah memastikan bahwa Apache dan MySQL terkoneksi, dengan cara mengubah tombol *stop* pada action menjadi *start*. Berikut adalah tampilan XAMPP (X(Windows/Linux) Apache MySQL PHP dan Perl) control panel untuk menjalankan Apache dan MySQL.