BAB II
TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Asropudin (2013:19) menjelaskan bahwa, komputer adalah alat bantu pemrosesan data secara elektronik dan cara pemrosesan datanya berdasarkan urutan instruksi atau program yang tersimpan dalam memori masing-masing komputer.

Sujatmiko (2012:156) mengemukakan bahwa, Komputer adalah mesin yang dapat mengelola digital dengan mengikuti serangkaian perintah atau program.

Dari pernyataan di atas dapat disimpulkan bahwa merupakan suatu alat elektronik yang mampu melakukan keempat tugas berikut: menerima inputan, memproses inputan, menyimpan perintah dan hasil pemrosesan, serta menyediakan *output* berupa informasi.

2.1.2 Pengertian Informasi

Rusdiana dan irfan (2014:74) menjelaskan, Informasi merupakan sesuatu yangdihasilkan dari pengolahan data yang sudah ada dikemas dan diolah sedemikian rupa sehingga menjadi sebuah informasi yang berguna.

Rusdiana dan irfan (2014:74) menjelaskan, Informasi adalah suatu data yang sudah diolah atau diproses sehingga menjadi suatu bentuk yang memiliki arti bagi penerima informasi yang memiliki nilai bermanfaat.

Rusdiana dan irfan (2014:74) menjelaskan, Informasi adalah kumpulan dari data yang diolah sehingga menjadi sesuatu yang berarti dan bermanfaat.

Dari pernyataan di atas dapat disimpulkan bahwa informasi adalah suatu data atau objek yang diproses terlebih dahulu sedmikian rupa sehingga dapat tersusun dan terklasifikasi dengan baik sehingga memiliki arti bagi penerimanya, yang selanjutnya menjadi pengetahuan untuk pengambilan keputusan.

2.1.3 Klasifikasi Sistem

Yakub (2012:4-5) menjelaskan, sistem dapat diklasifikasikan menjadi beberapa bagian yaitu:

a. Sistem tak tentu *(probabilistic system)*

   Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem deterministik. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program – program komputer yang dijalankan.Sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi, karena mengandung unsur probabilitas.

b. Sistem abstrak *(abstrac system)*

   Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu sistem yang berupa pemikiran hubungan antara manusia dengan tuhan; sedangkan sistem fisik merupakan sistem yang ada secara fisik, seperti sistem komputer, sistem produksi, sistem penjualan, sistem administrasi personalia dan lain sebagainya.

c. Sistem alamiah dan sistem buatan manusia

   Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam, pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan interaksi manusia dengan mesin, yang disebut *human machine system*.Sistem informasi berbasis komputer merupakan contohnya, karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

d. Sistem terbuka dan sistem tertutup

   Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa ada campur tangan dari pihak luar. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

2.1.4 Metode Pengembangan Sistem

Sukamto dan Shalahuddin (2013:28) menjelaskan tentang metode pengembangan sistem yaitu *waterfall*. Metode air terjun (*waterfall*) sering juga 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 mulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

a. Analisis

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

b. Desain

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

c. Pengodean

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

d. Pengujian

   Tahap pengujian fokus pada sistem dari segi logika 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 sistem mengalami perubahan ketika sudah dikirimkan 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 baru.

2.2 Teori Judul

2.2.1 Pengertian Aplikasi

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

Sujatmiko (2012:23-24) , Application merupakan program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu misalnya Ms-Word, Ms-Excel. Application berbeda dengan sistem operasi (yang menjalankan komputer), utility (yang melaksanakan perawatan atau tugas-tugas umum) dan bahasa (yang digunakan untuk, membuat program-program komputer).

Sutabri (2012:148), aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya.

Dari beberapa definisi di atas dapat disimpulkan bahwa aplikasi adalah sebuah perangkat lunak komputer yang digunakan untuk membantu mengerjakan masalah masalah tertentu.

2.2.2 Pengertian *Web*

Ardrana (2011:3), *Web (world wide web/www)* adalah suatu layanan di dalam jaringan internet yang berupa informasi.

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

Asropudin (2013:109) menjelaskan bahwa, Website adalah sebuah kumpulan halaman (webpages) yang diawali dengan halaman muka (homepage) yang berisikan informasi, iklan, serta program interaksi.

2.2.3 Pengertian Data

Yakub (2012:51-52) menjelaskan, data merupakan representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagaimana yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya.

Rusdiana dan irfan (2014:71), Data adalah fakta atau bagian dari fakta yang mengandung arti sehubungan dengan kenyataan, simbol-simbol, gambar-gambar, angka-angka, huruf atau symbol yang menunjukkan suatu ide, objek, kondisi, atau situasi lainnya, yang didapatkan melalui suatu observasi atau secara data diartikan keterangan tentang sesuatu.

Sujatmiko (2012:76), Data merupakan kumpulan dari angka-angka maupun karakter-karakter yang tidak memiliki arti. Data dapat diolah sehingga menghasilkan informasi.

Dari beberapa definisi di atas dapat disimpulkan bahwa data adalah fakta yang dapat digunakan sebagai keterangan atau *input*kemudian diolah dan menghasilkan sebuah informasi.

2.2.4 Pengertian Pelatihan

Menurut Kamus Besar Bahasa Indonesia Online, Pelatihan adalah proses, cara, perbuatan melatih kegiatan atau pekerjaan melatih yang diberikan belum cukup di bidang industri, perusahaan itu sudah mulai melakukan sendiri.
(http://kbbi.web.id/pelatihan)

2.2.5 Pengertian Manajemen

Rusdiana dan irfan (2014:113), Manajemen berarti proses proses perencanaan, pengorganisasian, pengarahan, dan pengawasan aktivitas-aktivitas pelayanan bimbingan dan konseling serta penggunaan sumber daya lainnya untuk mencapai tujuan yang telah ditetapkan.

2.2.6 Pengertian Kepegawaian

Menurut Kamus Besar Bahasa Indonesia Online, Pegawai adalah orang yang bekerja pada pemerintah, perusahaan, dan sebagainya sekalian negeri bersumpah akan setia. (http://kbbi.web.id/pegawai)

2.2.7 Pengertian Rumah Sakit

Menurut Undang Undang Republik Indonesia Nomor 44 Tahun 2009 Pasal 1 menjelaskan, Rumah Sakit merupakan institusi pelayanan kesehatan perorangan secara paripurna yang menyediakan pelayanan rawat inap, rawat jalan, dan gawat darurat.

2.2.8 Pengertian Rumah Sakit Ernaldi Bahar

Rumah Sakit Ernaldi Bahar Palembang pada mulanya bernama Rumah Sakit Jiwa yang didirikan pada tahun 1920. Rumah sakit ini sebagai salah satuan kerja pernagkat daerah (SKPD) di lingkungan pemerintahan Provinsi Sumatera Selatan berkewajiban untuk membuat Rencana Kerja (Renja). Rumah sakit ini melayani berbagai pelayanan kesehatan yang berhubungan dengan penyakit jiwa atau gangguan *psikologi* seperti rawat inap, rawat jalan, graha spesialis, penunjang medis, dan poliklinik.

2.2.9 Aplikasi Pendataan Pelatihan Manajemen Kepegawaian Pada Rumah Sakit Ernaldi Bahar Palembang

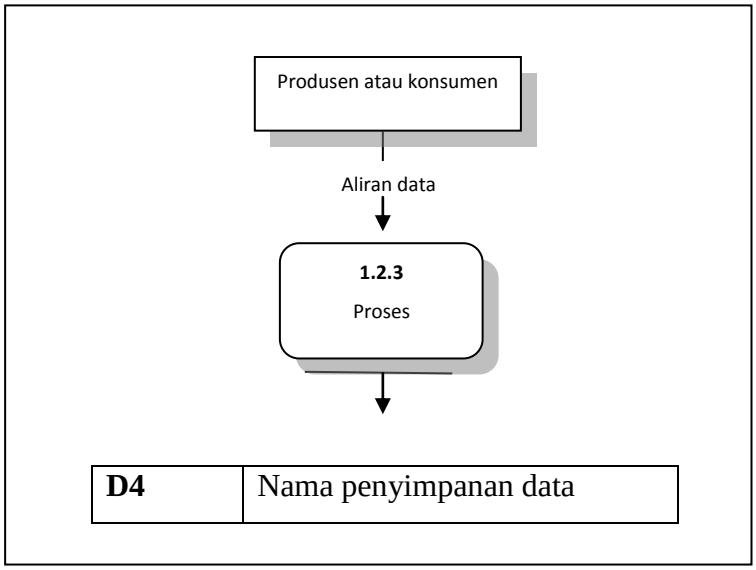
Aplikasi Pendataan Pelatihan Manajemen Kepegawaian ini sebagai Alat bantu untuk memudahkan bagian administrasi kepegawaian pada Rumah Sakit Ernaldi Bahar Palembang dalammengelola data dan mengetahui data pegawai yang telah mengikuti pelatihan dengan pegawai yang belum mengikuti pelatihan.

2.3 Teori Khusus

2.3.1 Pengertian DFD (*Data Flow Diagram)*

Sukamto dan Shalahuddin (2014:69) menjelaskan, "*Data Flow Diagram* (DFD) awalnya dikembangkan oleh Chris Gane dan Trish Sarson pada tahun 1979 yang termasuk dalam *Structured Systems Analysis and Design Methodology* (SSADM) yang ditulis oleh Chris Gane dan Trish Sarson. System yang

dikembangkan ini berbasis pada dekomposisi fungsional dari sebuah system." Berikut adalah contoh DFD yang dikembangkan oleh Chris Gane dan Trish Sarson:



(*sumber: Sukamto dan Shalahuddin 2014:6)*

Gambar 2.1 Contoh DFD yang dikembangkan Chris Gane dan Trish Sarson

Edward Yourdon dan Tom DeMarco memperkenalkan metode yang lain pada tahun 1980-an di mana mengubah persegi dengan sudut lengkung (pada DFD Chris Gane dan Trish Sarson) dengan lingkaran untuk menotasikan. DFD Edward Yourdon dan Tom DeMarco populer digunakan sebagai model analisis system perangkat lunak untuk system perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur.

Informasi yang ada di dalam perangkat lunak dimodifikasi dengan beberapa transformasi yang dibutuhkan. *Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan *(input)* dan keluaran *(output)*.

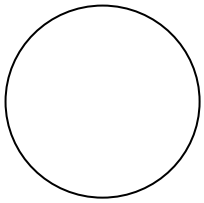
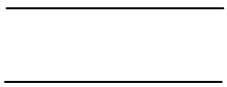
DFD dapat digunakan untuk merepresentasikan sebuah system atau perangkat lunak pada beberapa level abstraksi. DFD dapat dibagi menjadi beberapa level yang lebih detail untuk merepresentasikan aliran informasi atau

fungsi yang lebih detail. DFD menyediakan mekanisme untuk pemodelan fungsional ataupun pemodelan aliran informasi. Oleh karena itu, DFD lebih sesuai digunakan untuk memodelkan fungsi-fungsi perangkat lunak yang akan diimplementasikan menggunakan pemrograman terstruktur karena pemrograman terstruktur membagi-bagi bagiannya dengan fungsi-fungsi dan prosedur-prosedur.

DFD tidak sesuai untuk memodelkan system perangkat lunak yang akan dibangun menggunakan pemrograman berorientasi objek. Paradigma pemrograman terstruktur dan pemrograman berorientasi objek merupakan hal yang berbeda. Jangan mencampuradukkan pemrograman terstruktur dan pemrograman berorientasi objek.

Sukamto dan Shalahuddin (2014:71) menjelaskan, "notasi- notasi pada DFD (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut:

Tabel 2.1 Notasi-notasi pada DFD (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 harusnya menjadi fungsi atau prosedur di dalam kode program. Catatan: Nama yang diberikan pada sebuah proses biasanya berupa kata kerja.
	*File* atau basisdata atau penyimpanan *(storage)*; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data *(Entity Relationship Diagram* (ERD), *Conceptual Data Model* (CDM), *Physical Data Model* (PDM))

Lanjutan Tabel 2.1. Notasi-notasi pada DFD (Edward Yourdon dan Tom DeMarco)

Notasi	Keterangan
	Entitas luar *(external entity)* atau masukan *(input)* atau keluaran *(output)* atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau system lain yang terkait dengan aliran data dari system yang dimodelkan. Catatan: Nama yang digunakan pada masukan *(input)* atau keluaran *(output)* biasanya berupa kata benda
	Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan *(input)* atau keluaran *(output)* 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"

(*sumber: Sukamto dan Shalahuddin 2014:71)*

Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD Level 0 atau sering disebut juga *Context Diagram*

DFD Level 0 menggambarkan system yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun system lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara system yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam system yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2

Modul-modul pada DFD Level 1 dapat di-*breakdown* menjadi DFD Level 2. Modul mana saja yang harus di-*breakdown* lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di-*breakdown* lagi. Untuk sebuah system, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di- *breakdown*.

4. Membuat DFD Level 3 dan seterusnya

DFD Level 3, 4, 5 dan seterusnya merupakan *breakdown* dari modul pada DFD Level di-atasnya. *Breakdown* pada level 3, 4, 5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

Pada satu diagram DFD sebaiknya jumlah modul tidak boleh lebih dari 20 buah. Jika lebih dari 20 buah modul, diagram akan terlihat rumit dan susah untuk dibaca sehingga menyebabkan system yang dikembangkan juga menjadi rumit.

2.3.2 Pengertian Blockchart

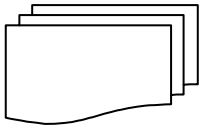
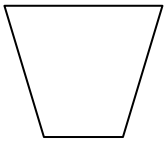
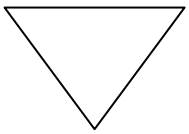
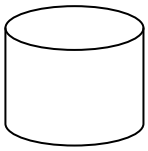
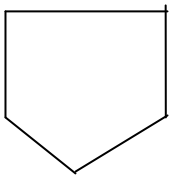
Kristanto (2008:68) menjelaskan, "*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *Blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi."

Kristanto (2008:68) menjelaskan, "Simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:

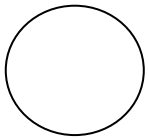
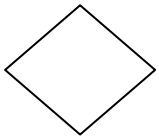
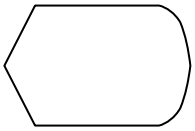
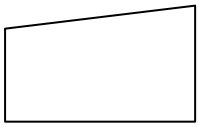
Tabel 2.2 Simbol-simbol dalam *Block Chart*

No	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan

Lanjutan Tabel 2.2 Simbol-simbol dalam *Block Chart*

No	Simbol	Keterangan
2.		Multi dokumen
3.		Proses Manual
4.		Proses dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan (*Storage)*
7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik.
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.

Lanjutan Tabel 2.2 Simbol-simbol dalam *Block Chart*

No	Simbol	Keterangan
9.		Terminal yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
10.		Terminal yang menandakan awal dan akhir dari suatu aliran.
11.		Pengambilan keputusan (*Decision).*
12.		Layar peraga (*monitor).*
13.		Pemasukkan data secara manual.

(*sumber: Kristano 2008:68)*

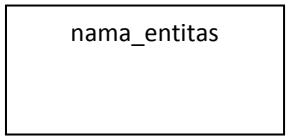
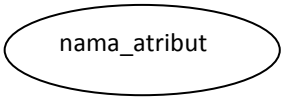
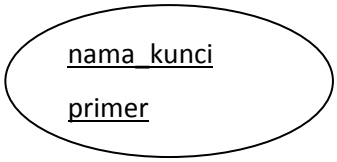
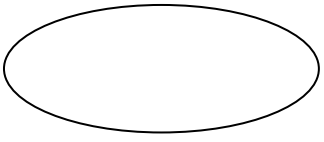
2.3.3 Pengertian ERD (*Entity Relationship Diagram*)

Sukamto dan Shalahuddin (2014 : 50) menjelaskan, tentang pemodelan awal basis data yang paling banyak digunakan adalah: *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan OODBMS maka perancangan basis data tidak perlu menggunakan ERD. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow's Foot, dan beberapa notasi

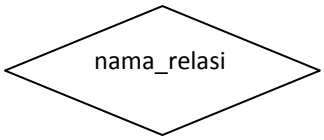
lain. Namun yang banyak digunakan adalah notasi dari Chen.

Sukamto dan Shalahuddin(2014:71) menjelaskan, "Berikut adalah symbol-simbol yang digunakan pada ERD dengan notasi Chen:

Tabel 2.3 Simbol-simbol ERD dengan notasi Chen

Simbol	**Deskripsi**
Entitas / *entity* 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi computer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
Atribut 	*Field* atau kolom data yang butuh disimpan dalam suatu entitas
Atribut kunci primer 	*Field* atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses *record* 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)
Atribut multinilai / *multivalue* 	*Field* atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu

Lanjutan Tabel 2.3 Simbol-simbol ERD dengan notasi Chen

Simbol	Deskripsi
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
Asosiasi / *association*	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki *multiplicity* kemungkinan jumlah pemakaian Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas

(*sumber: Sukamto dan Shalahuddin 2014:71)*

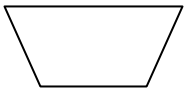
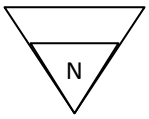
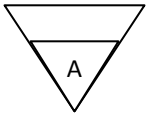
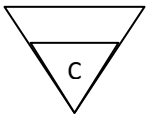
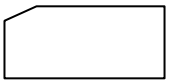
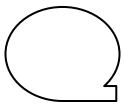
ERD biasanya memiliki hubungan *binary* (satu relasi menghubungkan dua buah entitas). Beberapa metode perancangan ERD menoleransi hubungan relasi *ternary* (satu relasi menghubungkan tiga buah relasi) atau N-*ary* (satu relasi menghubungkan banyak entitas), tapi banyak metode perancangan ERD yang tidak mengizinkan hubungan *ternary* atau N-*ary*.

2.3.4 Pengertian *Flowchart*

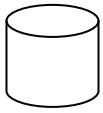
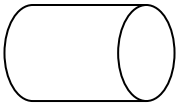
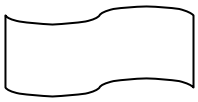
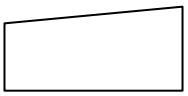
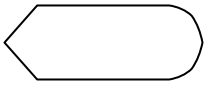
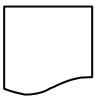
Supardi (2013:51) mengemukakan bahwa, bagan alir (*flowchart*) merupakan bagan (*chart*) yang menunjukkan alir (*flow*) didalam program atau prosedur sistem secara logika.

Ladjamudin (2013:211) mengemukakan bahwa, *flowchart* adalah bagan – bagan yang mempunyai arus yang menggambarkan langkah – langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma.

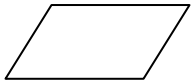
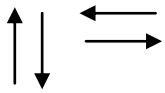
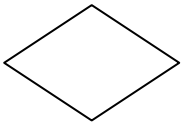
Tabel 2.4 Simbol-simbol *Flowchart*

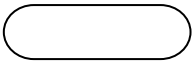
No	Nama Simbol	Simbol	Fungsi
1.	Simbol dokumen		Menunjukkan dokumen input untuk proses manual, mekanik atau komputer.
2.	Simbol kegiatan manual		Menunjukkan pekerjaan manual.
3.	Simbol simpanan offline		File non-komputer yang diarsip urut angka (*numerical*)/
			File non-komputer yang diarsip urut huruf (*alphabetical*).
			File non-komputer yang diarsip urut tanggal (*cronological*).
4.	Simbol kartu plong		Menunjukkan *input/output* yang menggunakan kartu plong (*punched card*).
5.	Simbol proses		Menunjukkan kegiatan proses dari operasi program komputer.
6.	Simbol operasi luar		Menunjukkan operasi yang dilakukan di luar proses operasi komputer.
8.	Simbol pita magnetic		Menunjukkan *input*/*output* menggunakan pita magnetik.

Lanjutan Tabel 2.4 Simbol-simbol *Flowchart*

No	Nama Simbol	Simbol	Fungsi
9.	Simbol *hard disk*		Menunjukkan *input*/*output* menggunakan *hard disk*.
10.	Simbol *diskette*		Menunjukkan *input*/*output* menggunakan *diskette*.
11.	Simbol drum magnetic		Menunjukkan *input*/*output* menggunakan drum magnetik.
12.	Simbol pita kertas berlubang		Menunjukkan *input*/*output* menggunakan pita kertas berlubang.
13.	Simbol *keyboard*		Menunjukkan *input* menggunakan *on-line keyboard*.
14.	Simbol *display*		Menunjukkan *output* yang ditampilkan di monitor.
15.	Simbol pita kontrol		Menunjukkan penggunaan pita kontrol (*control tape*) dalam *batch control total* untuk pencocokan di proses *batch processing*.

Lanjutan Tabel 2.4 Simbol-simbol *Flowchart*

No	Nama Simbol	Simbol	Fungsi
16.	Simbol hubungan komunikasi		Menunjukkan proses transmisi data melalui channel komunikasi.
18.	Simbol penghubung		Menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain.
19.	Simbol *input/output*		Simbol input/output (*input/output symbol*) digunakan untuk mewakili data input/output.
20.	Simbol garis alir		Simbol garis alir (*flow lines symbol*) digunakan untuk menunjukkan arus dari proses.
21.	Simbol keputusan		Simbol keputusan (*decision symbol*) digunakan untuk suatu penyeleksian kondisi didalam program.
22.	Simbol proses terdefinisi		Simbol proses terdefinisi (*predifined prosessmsymbol*) digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain.
23.	Simbol persiapan		Simbol persiapan (*preparation symbol*) digunakan untuk memberi nilai awal suatu besaran.

24.	Simbol titik terminal		Simbol titik terminal (*terminal point symbol*) digunakan untuk awal dan akhir dari suatu proses.

(*sumber:Supardi, 2013:53—59)*

2.3.5 Pengertian Kamus Data

Sukamto dan Shalahuddin (2014:73) menjelaskan, "Kamus data *(data dictionary)* dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD. Kamus data adalah kumpulan daftar elemen data yang mengalir pada system perangkat lunak sehingga masukan *(input)* dan keluaran *(output)* dapat dipahami secara umum (memiliki standar cara penulisan). Kamus data dalam implementasi program dapat menjadi parameter masukan atau keluaran dari sebuah fungsi atau prosedur. Kamus data biasanya berisi:

a. Nama – nama dari data

b. Digunakan pada – merupakan proses-proses yang terkait data

c. Deskripsi – merupakan deskripsi data

d. Informasi tambahan – seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data

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

Tabel 2.5 Simbol-simbol dalam Kamus Data

Simbol	**Keterangan**
=	disusun atau terdiri dari
+	Dan
[|]	baik ...atau...
$\{\}^n$	n kali diulang/ bernilai banyak

Lanjutan Tabel 2.5 Simbol-simbol Kamus Data

Simbol	Keterangan
( )	data opsional
...	batas komentar

(*sumber: Sukamto dan Shalahuddin 2014:73)*

Kamus data pada DFD nanti harus dapat dipetakan dengan hasil perancangan basis data yang dilakukan sebelumnya. Jika ada kamus data yang tidak dapat dipetakan pada tabel hasil perancangan basis data berarti hasil perancangan basis data dengan perancangan dengan DFD masih belum sesuai, sehingga harus ada yang diperbaiki baik perancangan basis datanya, perancangan DFD-nya, atau keduanya.

2.4 Teori Program

2.4.1 Sekilas Tentang Basis Data (*Database*)

Sukamto dan Shalahuddin (2014:43) menjelaskan, "System basis data adalah system terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.

Yakub (2012:51-52), Basis data (database) merupakan kumpulan data yang saling berhubungan (punya relasi).

Rusdiana dan irfan (2014:302) menjelaskan, Database adalah kumpulan informasi yang disimpan dalam komputer secara sistematik untuk memperoleh informasi dari basis data.

Berikut ini Karakteristik database dalam *Database Management System* (DBMS) Menurut Rusdiana dan irfan (2014:303) memilik tiga karakteristik utama, yaitu :

a. Data yang sama dapat diakses secara serempak oleh berapa pengguna untuk berbagai kegunaan yang berbeda.

b. Data tidak bergantung pada struktur penyimpanan atau cara membaca data dari program aplikasi, atau data bersifat transparan terhadap program aplikasi.
c. Dapat Memiliki integritas (akurasi dan validasi) yang terkendali.

2.4.2 Sekilas Tentang *HTML*

Pratama (2014:218) menjelaskan, "HTML atau *Hyper Text Markup Language* merupakan salah satu bahasa pemrograman yang digunakan di dalam pembuatan aplikasi dan layanan berbasis web."

Asropudin (2013:44) menjelaskan bahwa, *HTML* (*Hypertext Markup Language*) adalah bahasa komputer yang digunakan untuk membuat suatu halaman *webpage*

Winarno dkk (2014:1) *HyperText Markup Language (HTML*) adalah sebuah bahasa untuk Menampilkan Kontent di Web.

Madcoms (2016:15) mengemukakan bahwa, *HyperText Markup Language (HTML*) adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi di dalam sebuah penjelajah web internet dan formating hypertext sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi.

Sujatmiko (2012:128) menjelaskan bahwa, *HTML* (*Hypertext Markup Language*) adalah Bahasa komputer yang digunakan untuk membuat sebuah halaman web page atau program yang digunakan untuk menulis (membuat) halaman web di internet, fasilitas / bahasa ini biasanya mempunyai ekstensi **htm, html, atau shtml.**

2.4.3 Pengertian *MySQL*

Raharjo (2015:16), "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*)*"*.

Winarno dkk (2014:102) juga berpendapat bahwa MySQL adalah software database. MySQL merupakan tipe data relasional yang artinya MySQL menyimpan datanya dalam bentuk table -tabel yang saling berhubungan.

Hendry (2015:7), MySQL merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukkan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Berdasarkan teori para ahli di atas dapat disimpulkan bahwa, MySQL adalah sebuah software database yang diolah dengan sangat cepat dan dapat menampung data dalam jumlah yang sangat besar serta diakses oleh user.

Berikut tipe-tipe data pada *MySQL*:

Tabel 3.8 Tipe Data pada *MySQL*

Tipe Data	**Ukuran (Bytes)**
TINYINT	1
SMALLINT	2
MEDIUM	3
INT	4
BIGINT	8
FLOAT(p)	4
FLOAT	4
DOUBLE	8
REAL	8
DECIMAL (*L*,*D*)	L + 1
NUMERIC (*L*,*D*)	L + 1
BIT (*L*)	8
DATE	3
TIME	3
DATETIME	8
TIMESTAMP	4

Lanjutan Tabel 3.8 Tipe Data pada *MySQL*

BINARY(*L*)	L + 1
VARCHAR (*L*)	L + 1
VARBINARY (*L*)	L + 1
TINYBLOB	L + 1
TINYTEXT	L + 1
BLOB	L + 2
TEXT	L + 2
MEDIUMBLOB	L + 3
MEDIUMTEXT	L + 3
LONGBLOB	L + 4
LONGTEXT	L + 4
ENUM (‘nilai1’,..)	1 – 2
SET (‘nilai1’,..)	1 – 8
YEAR	1
CHAR (*L*)	L

(*sumber: Hendry 2015:7)*

Keistimewaan MySQL

Hendry (2015:7—9), MySQL memiliki beberapa keistimewaan, antara lain:

1. Portabilitas

   MySQL dapat berjalan stabil pada berbagai sistem operasi, seperti Windows, Linux, FreeBDS, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.

2. Perangkat lunak sumber terbuka

   MySQL didistribusikan sebagai perangkat lunak sumber terbuka, di bawah lisensi GPL sehingga dapat digunakan secara gratis.

3. *Multi-user*

   MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

4. *Performance tuning*

   MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana. Dengan kata lain, dapat memproses lebih banyak SQL per satuan waktu.

5. Ragam tipe data

   MySQL memiliki ragam tipe data yang sangat kaya, seperti signed/unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.

6. Perintah dan Fungsi

   MySQL server memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (*query*).

7. Keamanan

   MySQL memiliki beberapa lapisan keamanan seperti level subnetmask, nama host, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.

8. Skalabilitas dan Pembatasan

   MySQL mampu menangani basis data dalam skala cukup besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu table, serta 5 miliar baris. Selain itu, batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

9. Konektivitas

   MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix soket (Unix), atau Named Pipes (NT).

10. Lokalisasi

    MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahas indonesia belum termasuk di dalamnya.

11. Antarmuka

    MySQL memiliki antarmuka (*interface*) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

12. Klien dan Peralatan

    MySQL dilengkapi dengan berbagai peralatan (*tool*) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada, disertakan petunjuk online.

13. Struktur Tabel

    MySQL memiliki struktur tabel lebih fleksibel dalam menangani ALTER TABLE, dibandingkan basis data lainnya seacam PostgreSQL ataupun Oracle.

2.4.4 Sekilas Tentang *PHP*

Prasetio (2015:130) mengemukakan bahwa, PHP (PHP:Hypertext Preprocessor) adalah bahasa script yang ditanam di sisi sever. Proses PHP dijalankan di server sat sebuah halaman dibuka dan mengandung kode PHP, prosesor itu akan menerjemahkan dan mengeksekusikan semua perintah dalam halaman tersebut, dan kemudian menampilkan hasilnya ke browser sebagai halaman html biasa.

Winarno dkk (2014:49) PHP atau *PHP Hypertext Preprocessor*, adalah sebuah bahasa pemrograman web berbasis server (server-side) yang mampu memparsing kode PHP dari kode web dengan ekstensi .php, sehingga menghasilkan tampilan website yang dinamis di sisi client (browser).

Sujatmiko (2012:213), PHP adalah Bahasa pemrograman yang bisa kita gunakan untuk membuat aplikasi web. PHP banyak digunakan karena bisa kita ubah dan kita gunakan secara bebas.

Sidik (2017:4), PHP merupakan secara umum dikenal sebagai bahasa pemrograman script – script yang membuat dokumen HTML secara *on the fly* yang dieksekusi di server web, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML.dikenal juga sebagai bahasa pemrograman server side.

2.4.4.1 Tipe Data Dalam PHP

Menurut Winarno dkk (2014:65) menjelaskan bahwa tipe data PHP berbeda dengan bahasa pemograman lain, variable di PHP lebih fleksibel, anda tidak perlu mendefinisikan jenisnya ketika mendefinisikan pertam kali. Ada 6 variabel dasar yang dapat diakomodasikan di PHP, seperti terlihat ditabel:

Tabel 3.6 Tipe Data pada PHP

Tipe	**Contoh**	**Penjelasan**
Integer	134	Semua angka bukan pecahan
Double	5.123	Nilai pecahan
String	“asep”	Kumpulan karakter
Boolean	False	Salah satu nilai true atau false
Object		Sebuah instansi dari class
Array		Larik

(Sumber : Winarno dkk (2014:65*)*

2.4.5 Sekilas tentang *Dreamweaver*

2.4.5.1 Pengertian *Dreamweaver*

Sadeli (2014:12), Dreamweaver merupakan suatu perangkat lunak web editor keluaran *adobe system* yang digunakan untuk mengembangkan dan mendesain suatu website dengan fitur-fitur yang menarik dan kemudahan dalam penggunaannya.

Madcoms (2012:51) menjelaskan, Dreamweaver adalah *HTML* editor profesional untuk mendesain web secara visual dan mengelola situs atau halaman web.

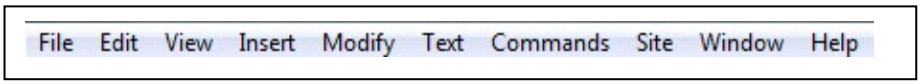
2.4.5.1 Tampilan Ruang Kerja *Adobe Dreamweaver CS 6*

Elcom (2013:2) menjelaskan Ruang kerja Dreamweaver CS6 memiliki komponen-komponen yang memberikan fasilitas dan ruang untuk menuangkan kreasi pada saat bekerja. Komponen-komponen yang disediakan oleh ruang kerja

Dreamweaver CS6 antara lain Insert Bar, Document Toolbar, Jendela Dokumen, Panel Group, Tag Selector, Property Inspector, dan Site Panel.

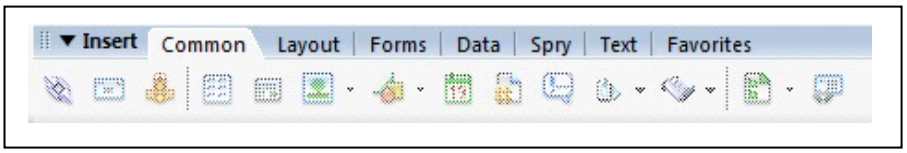
Berikut Ruang Kerja *Adobe Dreamweaver CS 6 :*

1. **Menu,** adalah media yang digunakan untuk melakukan atau menjalankan perintah pada *Dreamweaver*. Menu berisi perintah untuk pembuatan, pengaturan, manipulasi dan konfigurasi pada *Dreamweaver.*



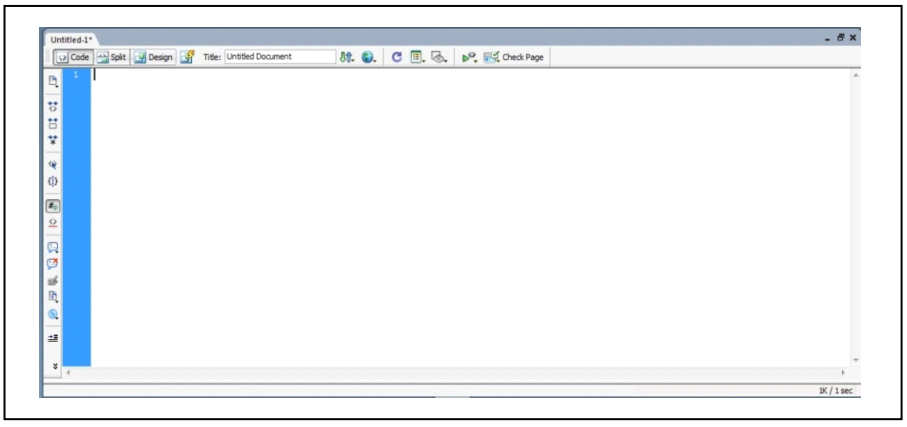
Gambar 2.2 Tampilan Menu *Adobe Dreamweaver CS 6*

2. ***InsertBar,*** adalah suatu bentuk *toolbar* yang berisi perntah untuk menambahkan suatu elemen pada dokumen.



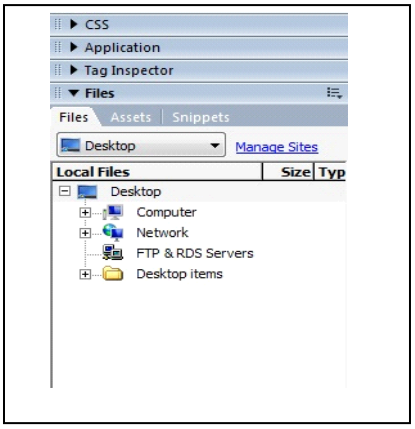
Gambar 2.3 Tampilan Insert Bar *Adobe Dreamweaver CS 6*

3. ***DocumentWindow,*** adalah editor *text* atau visual pada ruang kerja. *DocumentWindow* menyajiakn bentuk dokumen secara kode maupun visual.



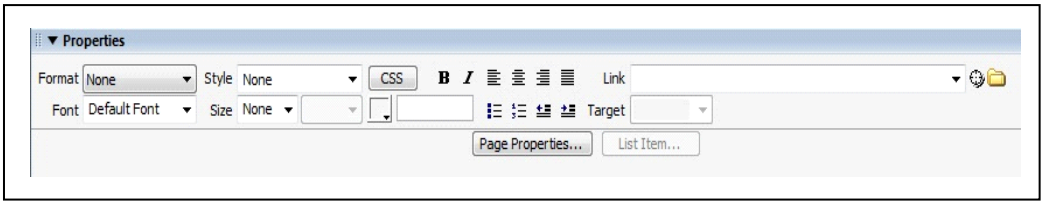
Gambar 2.4 Tampilan *DocumentWindowAdobe Dreamweaver CS 6*

4. ***PanelGroup,*** adalah bagian ruang kerja *Dreamweaver* yang berisikan kumpulan *panel*. *PanelGroup* terdiri dari kumpulan beberapa *panelcontrol*



Gambar 2.5 Tampilan *Panel* Group *Adobe Dreamweaver CS 6*

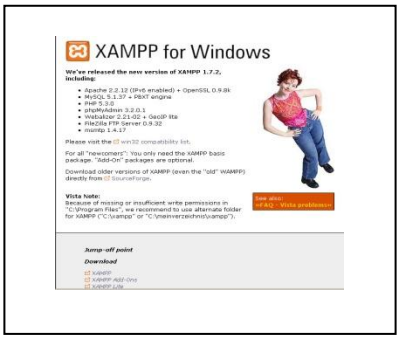
5. ***PropertyInspector,*** digunakan untuk menampilkan informasi *property* elemen *HTML* tertentu yang sedang dipilih.



Gambar 2.6 Tampilan *PropertyInspectorAdobe Dreamweaver CS 6*

2.4.6 Pengertian XAMPP

Sadeli (2014:4) mengemukakan bahwa, Xampp adalah program yang berisi paket apache, MySQL, dan phpMyAdmin.



Gambar 2.7 Halaman download xampp

Adapun langkah-langkah untuk menginstall program xampp pada komputer sebagai berikut .

1. Klik 2x (Double Klik) file insalasinya, selanjutnya anda akan diminta untuk menentukan bahasa yang digunakan saat instalasi.

Gambar 2.8 Menjalankan file instalasi

2.4.7 Pengertian phpMyAdmin

Sadeli (2014:10) mengemukakan bahwa, phpMyAdmin bahwa, sebuah software yang berbentuk seperti halaman situs yang terdapat pada web server.

Gambar 2.9 Logo phpMyAdmin

Fungsi dalam halaman ini adalah sebagai pengendali database MySQL sehingga pengguna MySQL tidak perlu report untuk menggunakan perintah-perintah SQL. Karena dengan adanya halaman ini semua yang ada pada halaman phpMyAdmin.

2.4.8 Sekilas Tentang *CSS*

Madcoms (2012:51) menjelaskan bahwa, *Cascanding Style Sheet (CSS)* adalah kumpulan kode-kode yang digunakan untuk mengendalikan tampilan isi suatu halaman web.

Suryana dan Koesheryatin (2014:101) menjelaskan bahwa, CSS (*Cascanding Style Sheet)* adalah suatu bahasa stylesheet yang digunakan untuk tampilan suatu website baik tata letaknya, jenis huruf, warna dan semua yang berhubungan dengan tampilan.

2.4.9 Sekilas Tentang *JavaScript*

Asropudin (2013:53) menjelaskan bahwa, *Javascript* adalah bahasa *script* interpreter yang digunakan untuk menciptakan isi yang aktif dalam halaman web.

Winarno dkk (2014:105) menjelaskan bahwa, *Javascript* adalah bahasa scripting *client side* yang sangat populer karena *Javascript* bisa dipakai di *HTML*, web, untuk server, pc, tablet, ponsel dan lainnya.