



---

## BAB II

### TINJAUAN PUSTAKA

#### 2.1. Teori Khusus

##### 2.1.1. Metode Pengembangan Sistem

Rosa 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.

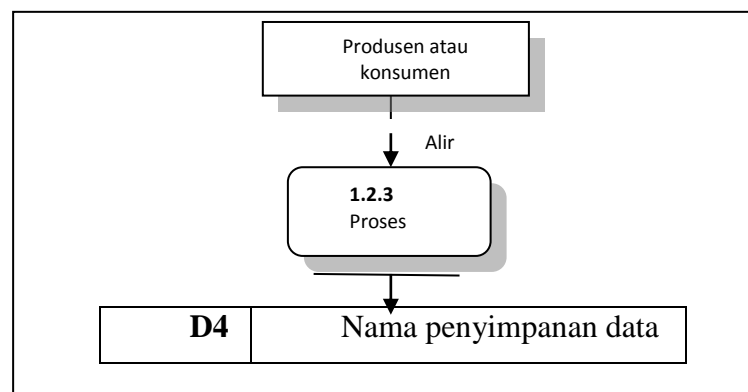
e. Pendukung (*support*) atau Pemeliharaan (*maintenance*)



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.

### 2.1.2. DFD (Data Flow Diagram)

Rosa 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:



**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



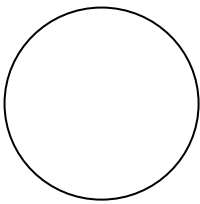
transformasi yang dibutuhkan. *Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).

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 mencampurkan pemrograman terstruktur dan pemrograman berorientasi objek.

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

**Table 2.1 Notasi-notasi pada DFD**

Notasi	Keterangan
	<p>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.</p> <p>Catatan:</p> <p>Nama yang diberikan pada sebuah proses biasanya berupa kata kerja.</p>

Lanjutan **Table 2.1** Notasi-notasi pada DFD

Notasi	Keterangan
	<p><i>File</i> atau basisdata 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, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data (<i>Entity Relationship Diagram</i> (ERD), <i>Conceptual Data Model</i> (CDM), <i>Physical Data Model</i> (PDM))</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 system lain yang terkait dengan aliran data dari system yang dimodelkan.</p> <p>Catatan: Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda</p>
	<p>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>)</p> <p>Catatan: Nama yang digunakan pada aliran data biasanya berupa kata benda,</p>



	dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”
--	--------------------------------------------------------------------------------------------

(Sumber: Rosa dan Shalahuddin, 2014:69)

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.



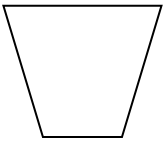

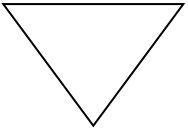
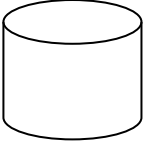


### 2.1.3. Pengertian Blockchart

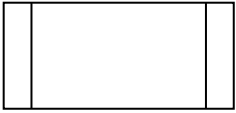
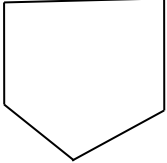
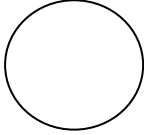

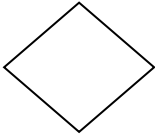
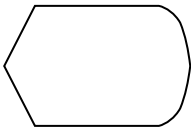

Kristanto (2011:68) menjelaskan, “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *Blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.”

Kristanto (2011:68) juga menjelaskan tentang 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
2.		Multi dokumen
3.		Proses Manual
4.		Proses dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan ( <i>Storage</i> )

Lanjutan Tabel 2.2 Simbol-simbol dalam *Block Chart*

No	Simbol	Keterangan
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.
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 ( <i>Decision</i> ).
12.		Layar peraga ( <i>monitor</i> ).
13.		Pemasukkan data secara manual.

(Sumber: Kristanto, 2011:68)

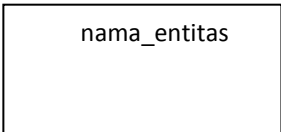
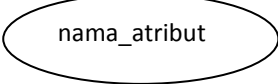
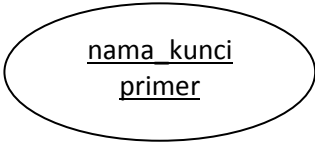


#### 2.1.4. ERD (Data Flow Diagram)

Rosa 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.

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

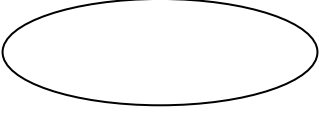
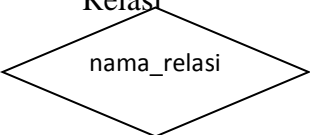
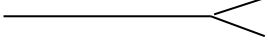
**Table 2.3 Simbol-Simbol pada ERD dengan notasi Chen**

Notasi	Keterangan
Entitas/ <i>entity</i> 	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 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
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)





Lanjutan Table 2.3 Simbol-Symbol pada ERD dengan notasi Chen

Atribut multivalai / <i>mvalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian  Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas

(sumber: Rosa dan Shalahuddin 2014:71)



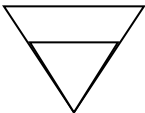
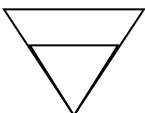
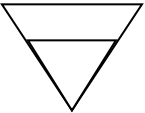


ERD biasanya memiliki hubungan *binary* (satu relasi menghubungkan dua buah entitas). Beberapa metode perancangan ERD menoleransi hubungan relasi *ternary* (satu relasi menghubungkan tiga buah relasi) atau *N-ary* (satu relasi menghubungkan banyak entitas), tapi banyak metode perancangan ERD yang tidak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



### 2.1.5. Flowchart

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




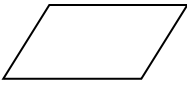
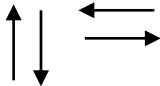
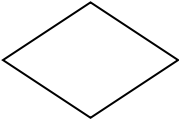
**Tabel 2.4. Simbol-simbol *Flowchart***

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 ( <i>numerical</i> )/
			File non-komputer yang diarsip urut huruf ( <i>alphabetical</i> ).
			File non-komputer yang diarsip urut tanggal ( <i>cronological</i> ).
4.	Simbol kartu plong		Menunjukkan <i>input/output</i> yang menggunakan kartu plong ( <i>punched card</i> ).
5.	Simbol proses		Menunjukkan kegiatan proses dari operasi program komputer.

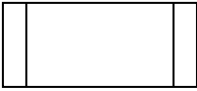

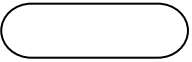
Lanjutan Tabel 2.4. Simbol-simbol *Flowchart*

No	Nama Simbol	Simbol	Fungsi
6.	Simbol operasi luar		Menunjukkan operasi yang dilakukan di luar proses operasi komputer.
7.	Simbol pita magnetic		Menunjukkan <i>input/output</i> menggunakan pita magnetik.
8.	Simbol <i>hard disk</i>		Menunjukkan <i>input/output</i> menggunakan <i>hard disk</i> .
9.	Simbol <i>diskette</i>		Menunjukkan <i>input/output</i> menggunakan <i>diskette</i> .
10.	Simbol drum magnetic		Menunjukkan <i>input/output</i> menggunakan drum magnetik.
11.	Simbol pita kertas berlubang		Menunjukkan <i>input/output</i> menggunakan pita kertas
12.	Simbol <i>keyboard</i>		Menunjukkan <i>input</i> menggunakan <i>on-line keyboard</i> .
13.	Simbol <i>display</i>		Menunjukkan <i>output</i> yang ditampilkan di monitor.

Lanjutan Tabel 2.4. Simbol-simbol *Flowchart*

No	Nama Simbol	Simbol	Fungsi
14.	Simbol pita kontrol		Menunjukkan penggunaan pita kontrol ( <i>control tape</i> ) dalam <i>batch control total</i> untuk pencocokan di proses <i>batch processing</i> .
15.	Simbol hubungan komunikasi		Menunjukkan proses transmisi data melalui channel komunikasi.
16.	Simbol penghubung		Menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain.
17.	Simbol <i>input/output</i>		Simbol <i>input/output</i> ( <i>input/output symbol</i> ) digunakan untuk mewakili data <i>input/output</i> .
18.	Simbol garis alir		Simbol garis alir ( <i>flow lines symbol</i> ) digunakan untuk menunjukkan arus dari proses.
19.	Simbol keputusan		Simbol keputusan ( <i>decision symbol</i> ) digunakan untuk suatu penyeleksian kondisi didalam program.

Lanjutan Tabel 2.4. Simbol-simbol *Flowchart*

No	Nama Simbol	Simbol	Fungsi
20.	Simbol proses terdefinisi		Simbol proses terdefinisi ( <i>predifined process symbol</i> ) digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain.
21.	Simbol persiapan		Simbol persiapan ( <i>preparation symbol</i> ) digunakan untuk memberi nilai awal suatu besaran.
22.	Simbol titik terminal		Simbol titik terminal ( <i>terminal point symbol</i> ) digunakan untuk awal dan akhir dari suatu proses.

(sumber: Supardi, 2013:53—59)



### 2.1.6. Pengertian Kamus Data

Rosa dan Shalahuddin (2013:73), "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)."

Fungsi kamus data adalah untuk mengetahui *field* dari tabel-tabel yang ada di dalam database secara lebih detail termasuk tipe *field* dan panjang karakter tulisan yang ada pada masing-masing *field*.

Rosa dan Shalahuddin (2013:74), menjelaskan simbol-simbol yang di gunakan dalam kamus data yaitu :

**Table 2.5 Simbol-simbol dalam Kamus Data**

No	Simbol	Arti
1	=	disusun atau terdiri atas
2	+	Dan
3	[]	baik ...atau...
4	{ } <sup>n</sup>	n kali diulang/ bernilai banyak
5	( )	data operasional
6	*...*	batas komentar

(Sumber : Rosa dan Shalahuddin (2013:74))

## 2.2. Pengertian Judul

### 2.2.1. Pengertian Aplikasi

Sujatmiko (2012:23), aplikasi adalah program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu.

Jogiyanto (2015), aplikasi merupakan penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output.



---

### **2.2.2. Pengertian Sarana Prasarana**

Anggraeni (2014), sarana prasarana adalah alat penunjang keberhasilan suatu proses upaya yang dilakukan di dalam pelayanan publik, karena apabila kedua hal ini tidak tersedia maka semua kegiatan yang dilakukan tidak akan mencapai hasil yang diharapkan sesuai dengan rencana.

Menurut Kamus Besar Bahasa Indonesia (dikutif oleh Maulidi, 2016), pengertian sarana adalah segala sesuatu yang dipakai sebagai alat untuk mencapai makna dan tujuan. Sedangkan prasarana adalah segala sesuatu yang merupakan penunjang utama terselenggaranya suatu proses.

Dari beberapa definisi diatas penulis menyimpulkan bahwa, sarana adalah segala sesuatu yang dapat dipakai sebagai alat dan bahan untuk mencapai maksud dan tujuan dari suatu proses produksi. Sedangkan prasarana adalah segala sesuatu yang merupakan penunjang utama terselenggaranya produksi.

### **2.2.3. Pengertian Olahraga**

Mutohir (dikutif oleh Rahayu, 2013:2), olahraga adalah proses sistematis yang berupa segala kegiatan atau usaha yang dapat mendorong mengembangkan, dan membina potensi-potensi jasmaniah dan rohaniah seseorang sebagai perorangan atau anggota masyarakat dalam bentuk permainan, perlombaan/pertandingan, dan kegiatan jasmani yang intensif untuk memperoleh rekreasi, kemenangan, dan prestasi puncak dalam rangka pembentukan manusia Indonesia seutuhnya yang berkualitas berdasarkan Pancasila.

Nixon dan Cozens (dikutif oleh Rahayu, 2013:2), mengemukakan bahwa pendidikan jasmani didefinisikan sebagai fase dari seluruh proses pendidikan yang berhubungan dengan aktivitas dan respons otot yang giat dan berkaitan dengan perubahan yang dihasilkan individu dari respons tersebut.

Dari beberapa definisi diatas penulis menyimpulkan bahwa, olahraga adalah aktivitas untuk melatih tubuh seseorang, tidak hanya secara jasmani namun juga secara rohani.



#### **2.2.4. Pengertian Atlet**

Wiyanto, dkk (2015:135), atlet adalah nama lain olahragawan, yaitu orang yang bisa berolahraga.

Sutardi (2010:175), Atlet (sering pula dieja sebagai atlit; dari bahasa Yunani: athlos yang berarti “kontes”) adalah orang yang ikut serta dalam suatu kompetisi olahraga kompetitif.

Dari beberapa definisi diatas penulis menyimpulkan bahwa, atlet adalah nama lain dari olahragawan, yaitu orang-orang yang biasa berolahraga yang memiliki kemampuan dalam bidang olahraga yang memiliki ketangkasan, kekuatan dan kelincahan.

#### **2.2.5. Pengertian Aplikasi sarana prasarana olahraga dan calon**

**atlet pada Dinas Pemuda dan Olahraga Provinsi Sumatera Selatan yaitu,** suatu aplikasi yang dibuat untuk mengetahui jumlah sarana prasarana olahraga yang ada pada Provinsi Sumatera Selatan, dan mengetahui jumlah pemuda pada Provinsi Sumatera Selatan serta calon atlet di bidang olahraga. Dengan adanya aplikasi ini dapat membantu Dinas Pemuda dan Olahraga Provinsi Sumatera Selatan dalam melakukan pengolahan data baik data Sarana Prasarana Olahraga, dan data calon atlet pada Provinsi Sumatera Selatan.

### **2.3. Teori Program**

#### **2.3.1. Sekilas Tentang PHP**

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

Oktavian (2012:31), PHP adalah akronim dari Hypertext Preprocessor, yaitu suatu bahasa pemrograman berbasis kode-kode (script) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke web browser menjadi kode HTML.





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.

### 2.3.2. Sintaks Dasar Php

Supono (2016:18), sama seperti bahasa pemrograman yang lain, bahasa pemrograman PHP memiliki aturan dasar cara penulisannya. Berikut ini beberapa cara penulisan sintak PHP yang bisa dipakai dalam pembuatan penulisan bahasa pemrograman PHP:

a. Cara I

<?php (menandai awal tag)

Isi script php

?> (menandai akhir tag)

b. Cara II

<? (menandai awal tag)

Isi script php

?> (menandai akhir tag)

c. Cara III

<% (menandai awal tag)

%> (menandai akhir tag)

Cara ini sama dengan tag pada ASP. Opsi bisa dilakukan jika nilai `asp_tags` pada php ini bernilai on.

d. Cara IV

<script language="php"> menandai awal script

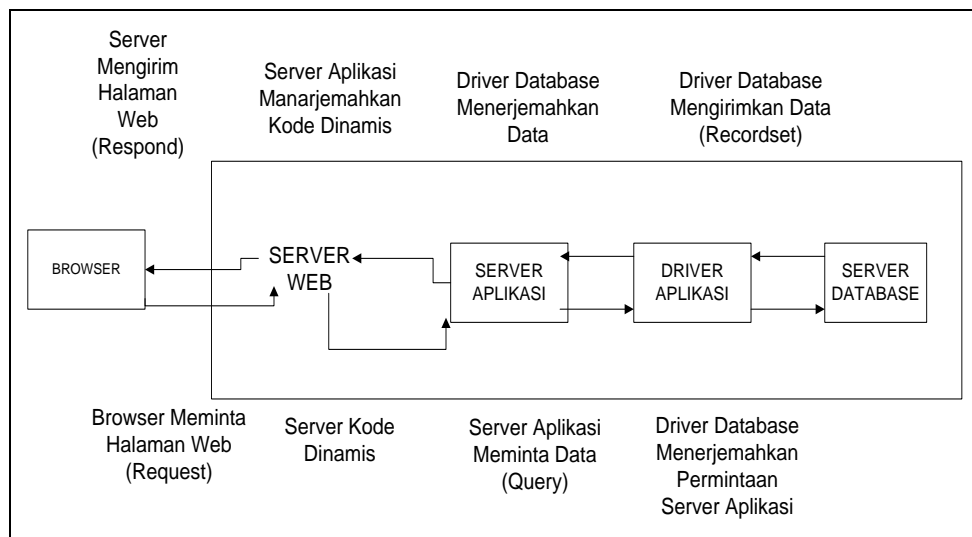
Isi script php

</script> menandai akhir script



### 2.3.3. Prinsip Kerja PHP

Supono (2016:4), bahasa pemrograman PHP merupakan bahasa pemrograman yang dikategorikan kepada server side programming, yang artinya bahasa pemrograman ini memerlukan penerjemah dalam hal ini web server untuk menjalankannya. Berikut ini diberikan gambaran tentang cara kerja bahasa pemrograman PHP:



**Gambar 2.2** Prinsip Kerja PHP

Penjelasan gambar:

1. Client.user mengirimkan file php (menggunakan browser) melalui web server (seperti internet explorer, mozilla firefox, google chrome, dll).
2. Web server mendapatkan request atau permintaan dari user lalu meneruskan ke server melalui jaringan internet.
3. Web server lalu meneruskan permintaan file php tersebut ke php processor. Php processor dapat berupa modul (bagian dari web-server) atau terpisah (sebagai CGI/Fast-CGI).
4. Permintaan diproses oleh php dan diteruskan ke database (Jika terdapat permintaan ke database), kemudian hasilnya dikirim kembali ke web-server.
5. Web Server memaket kembali hasil tersebut dengan menambahkan jaringan internet.
6. Browser memproses HTTP paket dan menampilkannya kembali kepada user sebagai file HTML.



### **2.3.4. Pengertian MySQL**

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.

Faizal, dkk (2015:4), MySQL adalah sebuah perangkat lunak system manajemen basis data SQL (Bahasa Inggris: database management system) atau DBMS yang multithread, multi-user, dengan sekitar 6 (enam) juta instansi di seluruh dunia.

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.

Dari beberapa definisi diatas penulis menyimpulkan bahwa, MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan menyimpan datanya dalam bentuk table –tabel yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

### **2.3.5. Tipe Data MySQL**

Hidayatullah dan Kawistara (2014:181), MySQL menggunakan tipe data numeric standar ANSI. Berikut ini adalah tipe data dalam *MySQL* yaitu :

#### **2.3.5.1. Tipe Data Numerik**

Hidayatullah dan Kawistara (2014:181) “*MySQL* menggunakan seluruh tipe data numeric standar ANSI”. Berikut ini adalah tipe data numeric yang biasanya digunakan beserta penjelasannya.




---

**Table 2.6 Tipe Data Numerik**

Tipe Data	Deskripsi
INT	Nilai Integer yang bias bertanda atau tidak. Jika Bertanda, maka rentang yang diperbolehkan adalah -2147483648 sampai 2147483647, sedangkan jika tidak bertanda maka rentangnya dari 0 sampai 4294967295
TINYINT	Nilai integer yang sangat kecil. Rentangnya 128-127 untuk yang bertanda dan 0-255 untuk yang tidak bertanda.
SMALLINT	Nilai integer yang sangat kecil dengan rentang 31768 sampai 32767 untuk yang bertanda sedangkan untuk yang tidak bertanda dari 0-65535.
MEDIUMINT	Integer dengan ukuran sedang dengan rentang -8388608 sampai 8388607a atau 0 sampai 16777215.
BIGINT	Integer dengan ukuran besar dengan rentang 9223372036854775808 sampai 92233722036854775807 atau 0 sampai 18446744073709551615.
FLOAT(M,D)	Bilangan pecahan dengan panjang (termasuk jumlah decimal) M dan jumlah decimal D. Presisi desimalnya bisa sampai 24 digit. Defaultnya float (10,2). Bilangan float selalu bisa bertanda
DOUBLE(M,D)	Adalah bilangan pecahan dengan presisi dua kali lipat. Panjang (termasuk jumlah decimal M dan jumlah decimal D. Presisi desimalnya bisa sampai 53 digit. Defaultnya Double(16,4). Bilangan float selalu bisa bertanda, Sinonim dari DOUBLE and REAL
DECIMAL(M,D)	Adalah bilangan pecahan dan harus didefinisikan M dan D-nya. Setiap decimal membutuhkan tempat 1 byte. Sinonim dari DECIMAL dan NUMERIC.

(Sumber: Hidayatullah dan Kawistara, 2014:181)



### 2.3.5.2. Tipe Data Tanggal dan Waktu

Berikut ini adalah tipe data tanggal dan waktu di dalam *MySQL* yakni :

**Table 2.7 Tipe Data Tanggal dan Waktu**

Tipe Data	Deskripsi
DATE	Adalah tipe data tanggal dengan format YYYY-MM-DD antara 1000-01-01 and 9999-12-31 Contoh: 17 Agustus 1945 akan di simpan sebagai 1945-08-17
DATETIME	Adalah kombinasi tanggal dan waktu dengan format YYYY-MM-DD HH:SS dan rentang data antara 1000-01-01 00-00-00 sampai dengan 9999-12-31 23-59-59. Contoh: Pukul 10:00 pagi pada tanggal 17 Agustus 1945 akan disimpan sebagai 1945-08-17 10:00:00
TIMESTAMP	Sebuah penanda waktu antara 1 Januari 1970 tengah malam sampai dengan tahun 2037. Formatnya mirip dengan DATETIME tetapi tanpa pembatas di antara angkanya. Contoh: pukul 10:00 pagi pada tanggal 17 Agustus 1945 akan disimpan sebagai 19450817000000
TIME	Menyimpan waktu dalam format HH:MM:SS contoh pukul 10:00 akan disimpan menjadi 10:00:00
YEAR(M)	Menyimpan data tahun dalam format 2 dan 4 digit. Jika M diisi dengan nilai 2, maka rentang tahunnya dari 1970 – 2069 sedangkan jika M diisi dengan nilai 4 maka YEAR bisa bernilai 1901 sampai dengan 2155. Default nilai M adalah 4.

(Sumber: Hidayatullah dan Kawistara, 2014:182)



### 2.3.5.3. Tipe Data String

Berikut ini adalah tipe data String yang paling umum di dalam MySQL yaitu :

**Table 2.8 Tipe Data String**

Tipe Data	Deskripsi
CHAR(M)	String dengan ukuran tetap. Ukurannya antara 1 sampai 255 karakter. Ukuran ditentukan dengan nilai M. Contoh: CHAR (6)
VARCHAR(M)	String dengan ukuran bervariasi antara 1 sampai dengan 255 karakter. Contoh: VARCHAR(25)
TEXT	String dengan ukuran maksimum 65535 karakter. String yang tersimpan di dalam TEXT dianggap tidak case sensitive. Untuk kapasitas yang lebih kecil bisa menggunakan TINYTEXT dengan kapasitas maksimal 255 karakter. Sedangkan untuk kapasitas yang lebih besar bisa menggunakan MEDIUMTEXT (maksimal 16777215 karakter) dan LONGTEXT (maksimal 4294967295 karakter).
BLOB	Binary Large Object (BLOB) adalah tipe data untuk menyimpan data binary dalam jumlah besar. Biasanya digunakan untuk menyimpan citra. Untuk menyimpan data yang lebih kecil bisa menggunakan TINYBLOB (maksimal 255 karakter) sedangkan untuk kapasitas yang lebih besar bisa menggunakan MEDIUMBLOB (maksimal 16777215 karakter) dan LONGBLOB (maksimal 4294967295 karakter).
ENUM	Enumerasi atau sebuah list (daftar). Jadi misalnya anda ingin bahwa sebuah nilai terbatas hanya boleh dengan nilai tertentu saja maka anda bisa membuat sebuah daftar. Misalnya saja nilai hanya bisa menjadi ENUM ('A', 'B', 'C', 'D', 'E').

(Sumber: Hidayatullah dan Kawistara, 2014:183)



---

### 2.3.6. Keistimewaan MySQL

Hendry (2015:79), MySQL memiliki beberapa keistimewaan, antara lain:

1. Portabilitas

MySQL dapat berjalan stabil pada berbagai sistem operasi, seperti Windows, Linux, FreeBSD, 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 socket (Unix), atau Named Pipes (NT).

#### 10. Lokalisasi

MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa 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 seperti PostgreSQL ataupun Oracle.

### 2.3.7. Pengertian HTML (Hypertext Markup Language)

Winarno (2014:1), Hypertext Markup Language (HTML), adalah sebuah bahasa untuk menampilkan konten di web. HTML sendiri adalah Bahasa pemrograman bebas, artinya tidak dimiliki oleh siapapun.

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 formatting 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.

### **2.3.8. Pengertian Adobe Dreamweaver CS6**

Madcoms (2013:1), adobe dreamweaver CS6 adalah perangkat lunak terkemuka untuk desain web yang menyediakan kemampuan visual yang intuitif termasuk pada tingkat kode, yang dapat digunakan untuk membuat dan mengedit website HTML serta aplikasi mobile seperti smartphone, tablet, dan perangkat lainnya.

Madcoms (2012:2), dreamweaver adalah sebuah HTML editor professional untuk mendesain web secara visual dan mengelola situs atau halaman web.

Dari beberapa definisi diatas penulis menyimpulkan bahwa, dreamweaver adalah program yang digunakan untuk membuat atau menyunting halaman web. Aplikasi ini banyak digunakan oleh para programmer, desainer dan developer web dikarenakan kemudahan dalam penggunaannya, kelengkapan fiturnya dan juga dukungannya terhadap teknologi terkini.

### **2.3.9. Pengertian CSS (cascading style sheet)**

Bekti (2015:46), CSS (cascading style sheet) merupakan salah satu Bahasa pemograman web yang digunakan untuk mempercantik halaman web dan mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam.

Bekti (2015:46), CSS (cascading style sheet) adalah suatu teknologi yang digunakan untuk memperindah tampilan halaman website (situs).



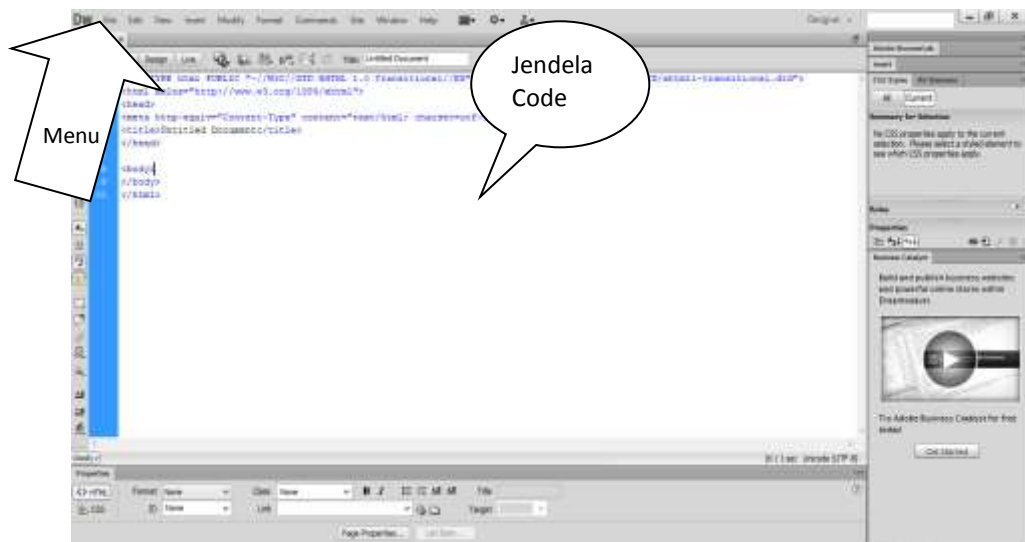
### 2.3.10. Menjalankan Dreamweaver CS6

Langkah untuk menjalankan Dreamweaver CS6 adalah, pilih **Start > All Program > Adobe Master Collection CS6 > Adobe Dreamweaver CS6**.



**Gambar 2.3** Tampilan halaman welcome screen dari Dreamweaver CS6

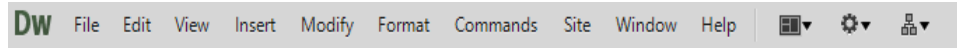
### 2.3.11. Mengenal Ruang Kerja Dreamweaver CS6



**Gambar 2.4** Tampilan Lembar Kerja Dreamweaver



1. Application Bar, berada bagian paling atas jendela aplikasi Dreamweaver CS6. Baris berisi tombol workspace (workspace switcher)



**Gambar 2.5** Tampilan Application Bar

2. Toolbar Document, berisi tombol-tombol yang digunakan untuk mengubah tampilan jendela dokumen, sebagai contoh tampilan Design atau tampilan Code.



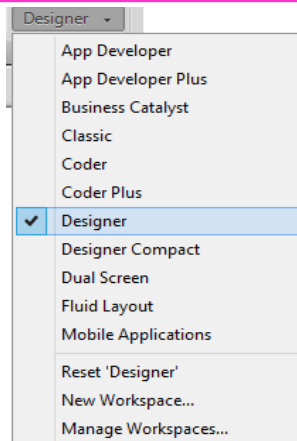
**Gambar 2.6** Tampilan Toolbar Document

3. Jendela Dokumen, adalah lembar kerja tempat anda membuat dan mengedit desain halaman web.



**Gambar 2.7** Jendela Dokumen

4. Workspace Switcher, digunakan untuk mengubah tampilan ruang kerja (workspace) Dreamweaver CS6.



**Gambar 2.8** Tombol perintah untuk memilih jenis tampilan ruang kerja

5. Panel Group, adalah kumpulan panel yang saling berkaitan, panel-panel ini dikelompokkan pada judul-judul tertentu berdasarkan fungsinya.
6. Tag Selector, diletakkan di bagian bawah jendela dokumen, satu baris dengan status bar. Bagian ini menampilkan hirarki pekerjaan yang sedang terpilih pada jendela pada jendela dokumen, dapat juga digunakan untuk memilih objek pada jendela desain berdasarkan jenis atau kategori objek tersebut.
7. Panel Property, digunakan untuk melihat dan mengubah berbagai properti objek atau teks pada jendela design.
8. Toolbar Coding, berisi tombol-tombol yang digunakan untuk melakukan operasi code-code standar. Toolbar ini hanya tampil pada jendela code.
9. Panel Insert, berisi tombol-tombol untuk menyisipkan berbagai jenis objek, seperti image, table, atau objek media ke dalam jendela dokumen.
10. Panel Groups adalah kumpulan panel yang saling berkaitan satu sama lainnya yang dikelompokkan dibawah satu judul, seperti: Insert, Files, CSS Style dan sebagainya.
11. Panel Files, digunakan untuk mengatur file-file dan folder-folder yang membentuk situs web anda.