

BAB II

LANDASAN TEORI

Dasar dari sebuah penelitian adalah landasan teori. Selain untuk mendukung pelaksanaan pada penelitian, landasan teori akan membuktikan bahwa penelitian tersebut berkualitas, dan memiliki pengetahuan yang cukup untuk melanjutkan penelitian sebelumnya, memperbaiki teori-teori pada penelitian sebelumnya, atau bahkan mematahkan teori sebelumnya.

2.1. Pengertian Klinik

Klinik merupakan fasilitas pelayanan kesehatan yang menyelenggarakan pelayanan kesehatan perorangan yang menyediakan pelayanan medis dasar dan/ atau spesialisik, diselenggarakan oleh lebih dari satu jenis tenaga kesehatan dan dipimpin oleh seorang tenaga medis.

(e-journal.uajy.ac.id/833/3/2TA12616.pdf)

2.2. Rancang Bangun

Rancang bangun merupakan serangkaian prosedur untuk menerjemahkan hasil analisa dari sebuah sistem ke dalam bahasa pemograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem diimplementasikan (Pressman, 2005).

2.3 Sistem

Sistem dapat didefinisikan dengan pendekatan prosedur dan pendekatan komponen. Dengan pendekatan prosedur sistem dapat didefinisikan sebagai kumpulan dari prosedur – prosedur yang mempunyai tujuan tertentu. Contoh sistem yang didefinisikan dengan pendekatan prosedur ini adalah sistem akuntansi. Sistem ini didefinisikan sebagai kumpulan dari prosedur-prosedur penerimaan kas, pengeluaran kas, penjualan, pembelian dan buku besar.

Dengan pendekatan komponen, sistem dapat didefinisikan sebagai kumpulan dari komponen yang saling berhubungan satu dengan yang lainnya membentuk satu kesatuan untuk mencapai tujuan tertentu. Contoh sistem yang didefinisikan dengan pendekatan ini misalnya adalah sistem komputer yang didefinisikan sebagai kumpulan dari perangkat keras dan perangkat lunak (Jogiyanto HM, 2005).

2.4. Sistem informasi

Menurut (Jogiyanto HM, 2005), sistem informasi mempunyai beberapa komponen seperti dibawah ini.

a. Model

Terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

b. *Output*

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

c. *Input*

Input mewakili data yang masuk ke dalam sistem informasi, *input* termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

d. *Database*

Merupakan kumpulan dari data yang saling berhubungan satu dengan lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

e. Teknologi

Sebagai kotak alat (*tool box*) untuk menerima *input*, menjalankan model, menyimpan dan membantu pengendalian sistem secara keseluruhan.

f. Desain Kontrol

Sistem informasi dapat berjalan sesuai yang diinginkan pengendali didalamnya.

Setelah mengetahui pengertian-pengertian dan komponen yang mendukung suatu sistem informasi, tahap selanjutnya yaitu menganalisa suatu sistem informasi yang ada. Menurut (Jogiyanto HM, 2005) analisis sistem merupakan penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-

kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya. Dapat ditarik kesimpulan bahwa analisis sistem yaitu tahapan yang sangat penting untuk memperbaiki atau membuat suatu sistem baru, dengan memahami suatu sistem maka tahapan untuk memperbaiki sistem akan dapat dengan mudah untuk dilakukan.

2.4.1. Kualitas Informasi (*Quality Of Information*)

Sistem Informasi Manajemen (Sutabri, 2005) kualitas dari suatu informasi bergantung dari tiga hal yaitu :

1. Akurat (*Accurate*)

Berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya.

2. Tepat Waktu (*Timelines*)

Brarti informasi yang datang pada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai ligika karena informasi merupakan landasan dalam pengambilan keputusan.

3. Relevan (*Relevance*)

Berarti informasi tersebut bermanfaat bagi pemakainya


2.5 Alat dan Teknik Pengembangan Sistem

Berikut ini adalah beberapa alat dan teknik pengembangan sistem :

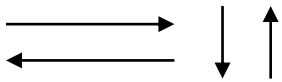
2.5.1 Data Flow Diagram (DFD)

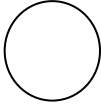
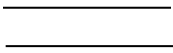
DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan proses tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.

Tabel 2.5.1 *Document flowchart* (Bagan Alir Dokumen)

Simbol	Keterangan
<i>(external entity)</i> 	Merupakan sumber atau tujuan dari aliran data dari atau ke sistem

Tabel 2.5.1 *Document flowchart* (Bagan Alir Dokumen Lanjutan)



Arus data (<i>data flow</i>) 	Menggambarkan arus data
---	-------------------------

Proses (<i>process</i>) 	Merupakan kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.
Simpanan data (<i>data store</i>) 	Merupakan komponen yang berfungsi untuk menyimpan data atau <i>file</i> .



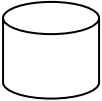
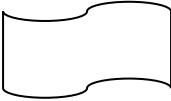

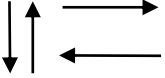

2.5.2 Document Flowchart (bagan alir dokumen)

bagan alir dokumen merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan – tembusan. Simbol – simbol yang digunakan adalah sebagai berikut :

Tabel 2.5.2 Document flowchart (Bagan Alir Dokumen)

Nama simbol	Keterangan
	Simbol dokument menunjukkan dokumen <i>input</i> dan <i>output</i> baik untuk proses <i>manual</i> , mekanik atau komputer.
	Proses dengan komputer

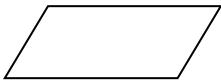


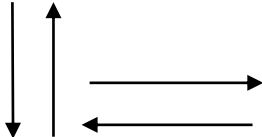
Tabel 2.5.2 Document flowchart (Bagan Alir Dokumen Lanjutan)

	Menunjukkan kegiatan manual
	Menunjukkan operasi yang dilakukan di luar proses operasi kompuetr
	Menunjukkan <i>input/output</i> menggunakan <i>hard disk</i>
	Menunjukkan <i>input/output</i> menggunakan pita kertas berlubang
	Menunjukkan <i>input</i> menggunakan <i>on-line keyboard</i>
	Menunjukkan arus dari proses
	Menunjukkan penghubung ke halaman yang masih sama atau di halaman lain

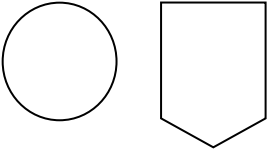

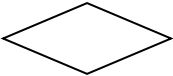
2.5.3 Program Flowchart (Bagan Alir Program)

Bagan alir program merupakan bagan yang menjelaskan secara rinci langkah – langkah dari proses program. Simbol – simbol yang digunakan adalah sebagai berikut

Tabel 2.5.3 Program Flowchart (bagan alir program)

Nama simbol	Keterangan
	Digunakan untuk mewakili data input atau output
	Digunakan untuk mewakili suatu proses
	Digunakan untuk menunjukkan suatu operasi yang rincinya ditunjukkan ke tempat lain
	Menunjukkan arus dari proses

Tabel 2.5.3 *Program Flowchart* (bagan alir program lanjutan)

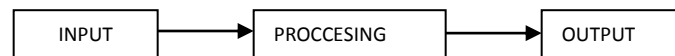
	Menunjukkan sambungan dari bagan alir yang terputus di halaman yang masih sama atau di halaman yang lainnya
	Menunjukkan awal atau akhir suatu proses
	Menunjukkan suatu penyeleksian kondisi di dalam program

2.6 Data

Menurut Jogianto (Jogianto HM, 2005) data merupakan bentuk jamak dari bentuk tunggal datum atau data item. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata

2.7 Pengolahan Data

Pengolahan data menurut (Jogianto HM, 2005) Pengolahan data (*data processing*) adalah manipulasi dari data ke dalam bentuk yang lebih berguna dan lebih berarti, berupa suatu informasi. Suatu proses pengolahan data (*data processing cycle*), yaitu input, processing dan output.



Gambar 2.1 Diagram pengolahan data

Tiga tahap dasar dari siklus pengolahan data tersebut dapat dikembangkan lebih lanjut. Siklus pengolahan data yang dikembangkan (*expanded data processing cycle*) dapat ditambahkan tiga tahap atau lebih tahapan lagi yaitu *original*, *storage* dan *distribution*.

2.8 Sistem Kode

Menurut jogiyanto HM (2005), Pengkodean adalah suatu kode yang digunakan untuk tujuan mengklasifikasi data, memasukan data ke dalam komputer dan untuk mengambil bermacam-macam informasi yang berhubungan dengannya. Ada beberapa macam tipe dari kode yang digunakan didalam sistem informasi diantaranya adalah :

- a. Kode (*Memonic Code*)

Kode Memonik digunakan untuk tujuan supaya mudah diingat kode memonik dibuat dengan dasar singkatan atau mengambil sebagian karakter dari item yang diwakili dengan kode ini. Umumnya kode memonik menggunakan huruf, tetapi juga dapat dengan gabungan huruf dan angka.

b. Kode Urut (*Sequential Code*)

Kode urut disebut juga dengan kode seri (*serial code*) merupakan kode kode nilainya urut antara satu kode dengan kode berikutnya.

c. Kode Blok (*Block Code*)

Kode blok mengklasifikasikan item ke dalam kelompok blok tertentu yang mencerminkan satu klasifikasi tertentu atas dasar pemakaian maksimum yang diharapkan.

d. Kode Grup (*Grup Code*)

Kode grup merupakan kode yang berdasarkan *field-flied* dan tiap-tiap *flied* kode mempunyai arti.

e. Kode Desimal (*Desimal Code*)

Kode desimal kode atas dasar 10 menit unit angka desimal dimulai angka 0 sampai dengan angka 9 atau 00 sampai dengan 99 tergantung dari banyak kelompok.

2.9 Basis Data (*Database*)

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan disimpanan luar komputer dan digunakan perangkat lunak tertentu untuk memanipulasinya (Jogiyanto HM, 2013). Penyusunan basis data

digunakan untuk mengatasi masalah-masalah pada penyusunan data antara lain sebagai berikut :

a) Redudansi dan Inkonsistensi data

Yaitu jika file program aplikasi diciptakan oleh programmer yang berbeda pada waktu berselang cukup panjang, maka ada beberapa bagian data yang mengalami penggandaan (*redudancy*) pada file yang berbeda. Penyimpanan data yang sama berulang-ulang di beberapa file dapat mengakibatkan juga inkonsistensi atau tidak konsisten, hal ini terjadi satu file record diubah tanpa mengubah file yang sama pada record yang lain.

b) Kesulitan Pengaksesan Data

Kesulitan pengaksesan data timbul bila suatu saat terjadi pengolahan data yang kompleks dan dalam jumlah yang besar, sementara belum tersedia program untuk menunjang hal itu.

c) Isolasi Data Standarisasi

Jika data terbesar beberapa file dalam format yang tidak sama, maka ini akan menyulitkan dalam menulis program aplikasi untuk mengambil dan menyimpan data, maka data harus dalam satu *database* yang dibuat satu format sehingga mudah dibuat program aplikasi.

d) Banyak Pemakaian (*Multiple User*)

Dalam rangka mempercepat daya guna sistem dan mendapat respon waktu cepat, beberapa sistem mengizinkan banyak pemakaian untuk meng-update data nantinya, akan dipakai dalam waktu yang berbeda.

e) Masalah keamanan (*Security*)

Setiap merancang suatu *database*, masalah keamanan atau *security* harus sangat diperhatikan agar pemakai (*user*) tidak dapat mengakses semua data.

2.9.1 Relasi (*relationship*) Basis Data

Relasi antara dua file atau dua tabel dapat dikategorikan menjadi tiga macam. Demikian pula untuk membantu gambaran relasi secara lengkap terdapat juga tiga macam relasi dalam hubungan atribut dalam satu file.

a) One to one relationship



Gambar 2.2 Model relasi *one to one relationship*

Hubungan antara file pertama dengan file kedua adalah satu berbanding satu. Hubungan tersebut dapat digambarkan dengan tanda kotak untuk menunjukkan table dan relasi antara keduanya diwakilkan dengan tanda panah tunggal.

b) One to many relationship



Gambar 2.3 Model relasi *one to many relationship*

Hubungan antara file pertama dengan file kedua adalah satu berbanding banyak atau dapat pula dibalik banyak lawan satu. Hubungan tersebut dapat digambarkan dengan tanda kotak untuk menunjukkan tabel dan relasi antara

keduanya diwakilkan dengan tanda panah ganda pada salah satu hubungan.

c) *Many to many relationship*



Gambar 2.4 Model relasi *many to many relationship*

Hubungan antara file pertama dengan file kedua adalah banyak berbanding banyak. Hubungan tersebut dapat digambarkan dengan tanda kotak untuk menunjukkan tabel dan relasi antara keduanya diwakilkan dengan tanda panah ganda pada kedua hubungan tersebut.

2.10 Kamus Data

Kamus Data (*Data Dictionary*) Kamus data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari sistem informasi. Notasi standar yang digunakan pada kamus data yaitu :

SIMBOL	ARTI
=	Terdiri dari
+	<i>And</i> (Dan)
()	Pilihan (boleh Ya atau Tidak)
{ }	Iterasi / pengulangan proses
[]	Pilih salah satu pilihan
I	Pemisah pilihan didalam tanda []
*	Keterangan atau catatan
@	Petunjuk (<i>key field</i>)
Alias	Nama lain untuk data

Dibawah ini merupakan gambar dari contoh kamus data

Tabel : Mahasiswa

NIM	Nama	Jenis Kelamin	Tgl Lahir	Gol Darah	Alamat
3001	James	L	01/01/1980	A	Jl. Perummas No. 1
3002	Rudolfo	L	02/01/1982	B	Jl. Perummas No. 2
3003	Thomas	L	03/01/1981	A	Jl. Perummas No. 3
3004	Clara	W	04/01/1983	O	Jl. Perummas No. 4
3005	Habsi	L	05/01/1979	AB	Jl. Perummas No. 5
3006	Natalia	W	06/01/1980	A	Jl. Perummas No. 6
3007	Ahmed	L	07/01/1993	B	Jl. Perummas No. 7
3008	Syalton	L	08/01/1978	O	Jl. Perummas No. 8
3009	Avril	W	09/01/1984	AB	Jl. Perummas No. 9

Tabel : Matakuliah

Kode matakuliah	Nama Matakuliah	Semester
INFO01	Pengantar Teknologi Informasi	1
INFO02	Penrograman C	2
INFO03	Struktur Data	3
INFO04	Rekayasa Perangkat Lunak	5
INFO05	Analisa Perancangan Sistem Informasi	4

Tabel : Nilai

ID	NIM	Kode matakuliah	Nilai
1	3001	INFO01	A
2	3002	INFO02	B
3	3003	INFO03	B
4	3004	INFO04	B
5	3005	INFO05	B
6	3006	INFO05	B
7	3007	INFO05	B
8	3008	INFO05	A
9	3009	INFO05	A
10	3006	INFO05	A
11	3007	INFO05	C
12	3008	INFO05	C
13	3009	INFO05	C
14	3001	INFO04	C
15	3002	INFO04	B
16	3003	INFO04	B
17	3004	INFO01	B
18	3005	INFO01	B
19	3001	INFO02	A
20	3002	INFO01	A
21	3003	INFO02	A

Gambar 2.5 Contoh Kamus Data

2.11 Tahapan Perancangan Sistem

Menurut Presman (Presman, 2013) pendapatnya mengusulkan sebuah pendekatan kepada perkembangan perangkat lunak yang sistematis dan sequensial yang mulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode, pengujian, dan pemeliharaan. Model sequensial linier melingkupi aktivitas-aktivitas sebagai berikut:

1. Rekayasa dan Pemodelan Sistem Informasi

Karena perangkat lunak selalu merupakan bagian dari sebuah sistem (bisnis) yang lebih besar, kerja dimulai dengan membangun syarat dari semua elemen sistem dan mengalokasikan beberapa subset dari kebutuhan ke perangkat lunak tersebut.

2. Analisis kebutuhan Perangkat Lunak (*Analysis*)

Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khususnya pada perangkat lunak.

3. Desain (*Design*)

Desain perangkat lunak sebenarnya adalah proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda; struktur data, arsitektur perangkat lunak, *representasi interface*, dan *detail (algoritma) procedural*.

4. Generasi Kode (*Coding*)

Desain harus diterjemahkan kedalam bentuk mesin yang bisa dibaca.

5. Pengujian (*Testing*)

Sekali kode dibuat, pengujian program dimulai.

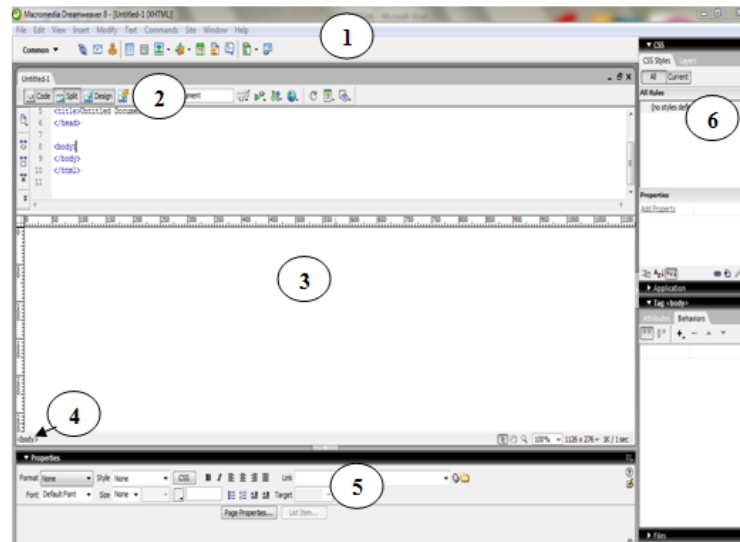
6. Pemeliharaan (*Maintenance*)

Perangkat lunak akan memahami perubahan setelah disampaikan kepada pelanggan.

2.12 Macromedia Dreamweaver

Menurut Dedi Efendi (2007) *Macromedia Dreamweaver* adalah sebagai *tool* untuk membuat halaman web atau menulis kode bahasa pemrograman (*coding*). Alasannya karena selain mudah digunakan dan tampilannya yang *user friendly* juga memiliki banyak kemampuan yang tidak dimiliki *software* desain web lainnya. *Dreamweaver* menyertakan banyak perangkat yang berkaitan dengan pengkodean dan fitur seperti *HTML*, *CSS*, *Javascript*, *PHP*, *ASP*, *Coldfusion* dan *XML*.

Macromedia Dreamweaver menawarkan cara mendesain situs dengan dua langkah sekaligus dalam satu waktu, yaitu mendesain dan memprogram. Setiap kali Kita mendesain halaman *web* seperti menulis kata-kata, meletakkan gambar, membuat tabe dan proses lainnya, maka kode *HTML* dan hasilnya dapat langsung dilihat dilayar.



Gambar. 2.6 Gambar Macromedia Dreamweaver

2.13 PHP

Menurut . (Nugroho, 2006), *PHP* adalah sebuah bahasa pemrograman yang berbentuk *scripting*. Sistem kerja ini adalah *interpreter* bukan sebagai kompiler. Bahasa *interpreter* adalah bahasa yang *script* program tidak harus diubah kedalam bentuk *source code*, sedangkan bahasa kompiler adalah bahasa yang mengubah *script*2 program kedalam *source code* , selanjutnya dari bentuk *source code* akan diubah menjadi *object code*, bentuk dari *object code* akan menghasilkan *file* yang lebih kecil dari file mentah sebelumnya

2.14 *PHPMysqlAdmin*

PHPMysqlAdmin merupakan sebuah program yang berbasis *web* yang dibuat menggunakan aplikasi *PHP*. Program ini adalah untuk mengakses database *MySQL*, intinya adalah digunakan untuk menjadi administrator dari *server MySQL*. Dengan adanya program ini akan mempermudah dan mempersingkat kinerja, dengan kelebihan² yang ada mengakibatkan pengguna awam tidak harus mampu mengetahui *script SQL* dalam pembuatan database dan tabel . (Nugroho, 2006)

2.15 *MySQL*

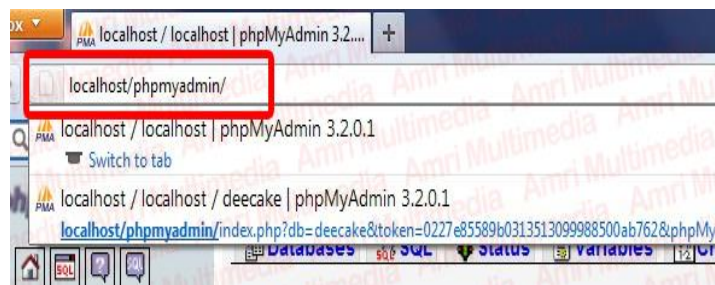
Menurut . (Nugroho, 2013), *MySQL* merupakan sebuah bentuk *database* yang berjalan sebagai *server*, tidak meletakkan *database* tersebut dalam satu mesin dengan aplikasi yang digunakan, sehingga dapat meletakkan sebuah *database* pada sebuah mesin khusus dan dapat diletakkan ditempat yang jauh komputer pengaksesannya. *MySQL* merupakan *database* yang sangat kuat dan cukup stabil digunakan sebagai media penyimpanan data. Sebagai *database server* yang mampu memanejem *database* dengan baik, *MySQL* terhitung merupakan *database* yang paling banyak digemari dan paling banyak digunakan dibanding *database* yang lain. *MySQL* memiliki kelebihan – kelebihan diantaranya seperti dibawah ini:

1. *MySQL* sebagai *database management sistem* (DBMS).
2. *MySQL* sebagai *relation database management sistem* (RDBMS).

3. MySQL sebuah *software database* yang *open source*, artinya program ini bersifat *free* atau bebas digunakan oleh siapapun tanpa harus membeli dan membayar pada pembuatnya.
4. MySQL merupakan *database server*, jadi dengan menggunakan *database* ini dapat menghubungkan ke media *internet* sehingga dapat diakses dari jauh.
5. MySQL mampu menerima *query* yang menumpuk dalam satu permintaan atau yang sering disebut *multi-threading*.
6. MySQL sebuah database yang mampu menyimpan data berkapasitas sangat besar sehingga berukuran *gigabyte* sekalipun.

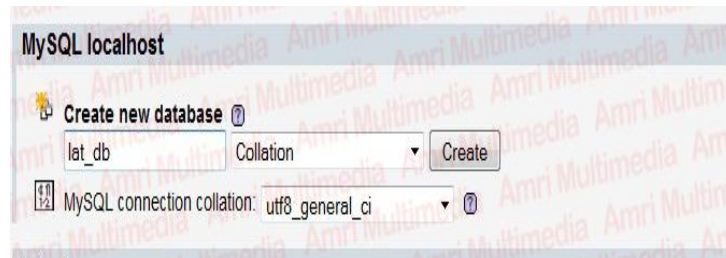
Untuk mengelola *database* di MySQL ada beberapa cara, yaitu menggunakan *prompt DOS (tool command line)* atau dapat juga menggunakan program *utility*. Salah satu program *utility* yang digunakan yaitu PHP *MyAdmin*. PHP *MyAdmin* yaitu perangkat lunak yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi MySQL dengan menggunakan *protocol www* secara langsung dari *browser*. Berikut ini akan dijelaskan langkah-langkah dalam PHP *MyAdmin* :

1. Ketikkan alamat <http://localhost/phpmyadmin>



Gambar 2.7 Tampilan Awal *Localhost*.

2. Mulai membuat *database*, ketikkan nama *database* baru kita di *form* berikut.



Gambar 2.8 Tampilan membuat database

3. Jika berhasil maka *PHPMYAdmin* akan memberi laporan seperti ini, dan akan langsung masuk ke dalam *database* tersebut.



Gambar 2.9 Tampilan Bentuk Database.

4. Selanjutnya mulai membuat tabel baru, ketik nama tabel baru di *form create new table*, dan tentukan jumlah *field*, dan klik *go*.
5. Tentukan nama *field*, *tipe data*, *length*, *validasi*, *primary key*, *auto increment* dan sebagainya, jika selesai klik *save*. Jika berhasil maka akan langsung diarahkan ke *table structure* seperti dibawah ini.



Gambar 2.10 Tampilan Isi Field Database.

6. Masukkan *record* ke dalam tabel yang baru dibuat, klik *insert* pada *tabulasi_bar*. Masukkan *record* yang diinginkan, dan jika berhasil maka akan seperti berikut. (klik *browse*, untuk melihat isi *record* dari tabel)



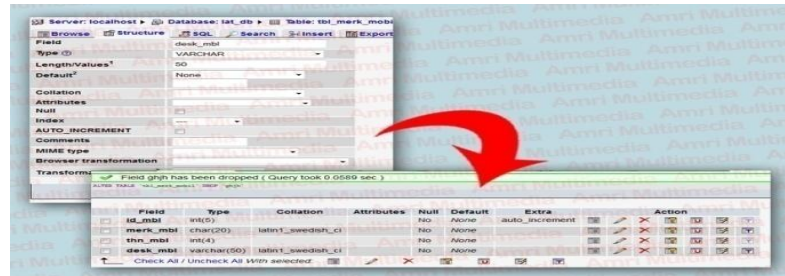
Gambar 2.11 Tampilan *Record Database*

7. Berikut kita akan mencoba menghapus *record* dan mengubah isi *record*, klik *delete*, maka akan muncul *pop-up window* untuk konfirmasi menghapus, dan apabila ingin meng-*update*, seperti gambar berikut ini.



Gambar 2.12 Tampilan *Update Isi Record*.

8. Menyisipkan *field*, pilih *tab structure*, setelah selesai mengisi tabel selanjutnya klik *go*, dan kalian akan diarahkan ke halaman seperti berikut, dan jika sudah klik *save*.



Gambar 2.13 Tampilan Isi *Database*.

Tipe data adalah suatu bentuk pemodelan data yang dideklarkan pada saat melakukan pembuatan tabel (Andi, 2004). Jenis tipe data pada MySQL dapat dibagi kedalam beberapa bentuk yaitu:

1. *Data numeric*.

Numeric adalah salah satu bentuk data yang angka. Berikut adalah beberapa bentuk dari data *numeric*.

- (a) *Tinyint* : tipe ini merupakan bentuk *numeric* yang paling kecil dalam menangani data didalamnya.
- (b) *Smallint* : memiliki kemampuan menangani data lebih besar.
- (c) *Mediumint* : tipe ini adalah tipe *numeric* yang paling sederhana.
- (d) *Int* : tipe ini sangat sering digunakan dalam pembuatan *database* karena data ini dirasakan cukup untuk menangani data yang lumayan besar.

(e) *Bigint* : bentuk terbesar dalam menangani data *numeric* adalah data *Bigint*.

2. Data untuk penanggalan dan waktu

MySQL juga memiliki data tersendiri dalam hal menangani masalah penanggalan dan waktu. Waktu dan tanggal adalah salah satu bentuk yang sangat penting dalam sebuah *database* agar kita dapat melakukan penyimpanan data menyangkut waktu dan tanggal seperti tanggal lahir, tanggal mulai, waktu mulai dan lain sebagainya.

Berikut adalah tipe data yang menangani waktu dan tanggalan.

(a) *Date time* : bentuk ini merupakan tipe data yang dapat menyimpan dua bentuk tipe data sekaligus, yaitu penanggalan dan waktu.

(b) *Date* : bentuk ini digunakan untuk menyimpan data penanggalan dan memiliki bentuk penulisan.

(c) *Time stamp* : bentuk penanggalan *time stamp* dapat diciptakan berjajar tanpa ada pembatasnya.

(d) *Time* : bentuk *time* digunakan untuk menyimpan data berbentuk penanggalan dan dimulai dari tahun dan diikuti bulan dan tanggal.

(e) *Year* : bentuk yang paling sederhana adalah *year*, yang hanya dapat menyimpan data berupa tahun yang hanya ditulis secara lengkap 4 digit.

3. Data *string*

Data *string* adalah tipe data yang mampu menyimpan semua data yang berupa *string*. Bentuk *string* adalah satu bentuk yang harus dimiliki dan didukung oleh setiap *database*. Data *string* dapat menyimpan semua data, baik data berupa numerik maupun tanggal dan waktu. Dalam tipe data *String* MySQL memiliki dua bentuk dukungan yaitu : kelompok data yang berbentuk teks besar dan karakter atau teks kecil.