

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Teori-Teori Dasar**

Berikut ini adalah beberapa teori-teori dasar yang terkait dengan masalah yang dibahas pada penulisan ini.

##### **2.1.1 Sistem**

Pengertian sistem menurut Sutabri (2016), “Sistem adalah terdiri atas objek-objek atau unsur-unsur yang berkaitan atau berhubungan satu sama lain nya sedemikian rupa sehingga unsur-unsur tersebut merupakan suatu kesatuan pemrosesan atau pengolahan yang tertentu. Berdasarkan pengertian tersebut dapat disimpulkan bahwa sistem adalah kumpulan dari komponen-komponen yang saling berkaitan satu dengan yang lain untuk mencapai tujuan dalam melaksanakan suatu kegiatan pokok perusahaan.

##### **2.1.2 Informasi**

Menurut Sutabri (2016) dalam buku sistem informasi manajemen, “Informasi adalah data yang telah diklasifikasi atau diolah atau interpretasi untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi mengolah data menjadi informasi atau tepatnya pengolah data dari bentuk tak berguna menjadi berguna bagi penerimanya.”

##### **2.1.3 E-commerce**

Secara sederhana E-commerce dapat diartikan sebagai konsep penerapan E-business sebagai strategi jual-beli barang dan jasa melalui jaringan elektronik dan biasanya melibatkan transaksi data elektronik, sistem manajemen inventory otomatis dan sistem pengumpulan data otomatis. Hal ini disebabkan semakin pesatnya perkembangan teknologi informasi baik dalam segi efisiensi serta keamanannya, sehingga memunculkan ide-ide gagasan untuk menjadikan teknologi informasi itu sebagai media untuk melakukan pemasaran, promosi,

bahkan transaksi data yang dianggap bisa lebih efisien dan mempermudah transaksi jual-beli.

#### **2.1.4 Aplikasi**

Aplikasi menurut Dhanta dikutip dari Sanjaya (2015) adalah software yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya Microsoft Word, Microsoft Excel. Aplikasi berasal dari kata application yang artinya penerapan lamaran penggunaan.

Jadi aplikasi merupakan sebuah transformasi dari sebuah permasalahan atau pekerjaan berupa hal yang sulit difahami menjadi lebih sederhana, mudah dan dapat dimengerti oleh pengguna. Sehingga dengan adanya aplikasi, sebuah permasalahan akan terbantu lebih cepat dan tepat. Dari pengertian diatas dapat disimpulkan aplikasi web merupakan aplikasi yang diakses menggunakan web browser melalui jaringan internet atau intranet. Aplikasi web juga merupakan suatu perangkat lunak komputer yang dikodekan dalam bahasa pemrograman yang mendukung perangkat lunak berbasis web seperti HTML, JavaScript, CSS, Ruby, Python, PHP, Java dan bahasa pemrograman lainnya.

#### **2.1.5 Website**

Menurut Agus Hariyanto (2015), “ Website adalah kumpulan halaman yang menampilkan informasi data teks, data gambar, data animasi, suara, video, dan gabungan yang membentuk satu rangkaian bangunan yang saling terkait, dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*).”

#### **2.1.6 Basis Data**

Menurut Sutabri (2016), “*Database* adalah suatu kumpulan data terhubung (interrelated data) yang disimpan secara bersama-sama pada suatu media, tanpa

mengatap satu sama lain atau tidak perlu suatu kerangkapan data (controlled redundancy.”

### **2.1.7 MySQL**

Menurut MADCOMS (2016) “MySQL adalah sistem manajemen Database SQL yang bersifat Open Source dan paling populer saat ini. Sistem Database MySQL mendukung beberapa fitur seperti multithreaded, multiuser dan SQL Database manajemen system (DBMS)”.

### **2.1.8 XAMPP**

Menurut MADCOMS (2016) “Xampp adalah sebuah paket kumpulan software yang terdiri dari Apache, MySQL, PhpMyAdmin, PHP, Perl, Filezilla, dan lain.” Xampp berfungsi untuk memudahkan instalasi lingkungan PHP, di mana biasanya lingkungan pengembangan web memerlukan PHP, Apache, MySQL dan PhpMyAdmin.

### **2.1.9 HTML**

HTML merupakan singkatan dari *Hypertext Markup Language* adalah script untuk menyusun dokumen – dokumen web. Dokumen HTML disimpan dalam format teks regular dan mengandung tag – tag yang memerintahkan web browser untuk mengeksekusi perintah – perintah yang dispesifikasikan. HTML merupakan standard bahasa yang di gunakan untuk menampilkan dokumen web, yang bisa dilakukan dengan HTML yaitu :

- Mengontrol tampilan dari web page dan contentnya.
- Mempublikasikan dokumen secara online sehingga bisa di akses dari seluruh dunia.
- Membuat online form yang bisa di gunakan untuk menangani pendaftaran, transaksi secara online.

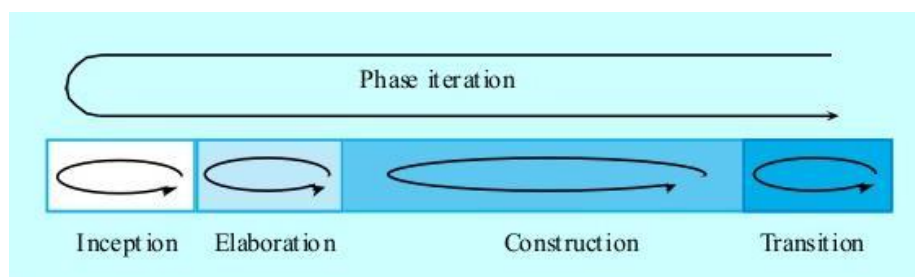
- Menambahkan objek – objek seperti gambar, audio, video dan juga dokumen HTML.

### 2.1.10 PHP

Menurut Rohi Abdullah (2015:2) “PHP singkatan dari *Hypertext Preprocessor* yang merupakan *server-side programming*, yaitu bahasa pemrograman yang diproses di sisi server.” Fungsi utama PHP dalam membangun website adalah untuk melakukan pengolahan data pada database. Data website akan dimasukkan ke database, diedit, dihapus, dan ditampilkan pada website yang diatur oleh PHP.

### 2.1.11 Rational Unified Process

Menurut Rosa A.S dan M.Shalahuddin (2015) “*Unified Process* atau dikenal juga proses iterative dan incremental merupakan sebuah proses pengembangan perangkat lunak yang dilakukan secara iteratif (berulang) dan incremental (bertahap dengan proses menaik).” Salah satu *unified process* yang dikenal adalah RUP (*Relational Unified Process*). Berikut *Phase* RUP yang digunakan :



**Gambar 2.1** Phase Relational Unified Process

*Relational Unified Process* (RUP) memiliki 4 tahap atau fase yang dapat dilakukan pula secara iterative. Berikut adalah penjelasan untuk setiap fase RUP :

### 1. *Inception* (Permulaan)

Tahap ini lebih pada pemodelan proses bisnis yang dibutuhkan (*business modeling*) dan mendefinisikan kebutuhan akan sistem yang akan dibuat (*requirements*).

### 2. *Elaboration* (Perluasan/Perencanaan)

Tahap ini lebih difokuskan pada perencanaan arsitektur sistem. Tahap ini juga dapat mendeteksi apakah arsitektur sistem yang diinginkan dapat dibuat atau tidak. Mendeteksi risiko yang mungkin terjadi dari arsitektur yang dibuat. Tahap ini lebih analisis dan desain sistem serta implementasi sistem yang fokus pada sistem (*prototype*).

### 3. *Construction* (Konstruksi)

Tahap ini fokus pada pengembangan komponen dan fitur – fitur sistem. Tahap ini lebih pada implementasi dan pengujian sistem yang fokus pada implementasi perangkat lunak pada kode program.

### 4. *Transition* (Transisi)

Tahap ini lebih pada *deployment* atau instalasi sistem agar dapat dimengerti oleh *user*. Aktivitas pada tahap ini termasuk pada pelatihan *user*, pemeliharaan dan pengujian sistem apakah sudah memenuhi harapan *user*.

## **2.2 Alat dan Teknik Pengembangan Sistem Diagram**

### **2.2.1 UML ( Unified Modelling Language)**


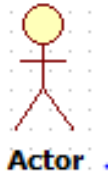



*Unified Modeling Language* (UML) adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO


(*Object-Oriented*). UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem software (Siti Fatima, 2015).

### 2.2.2 Usecase

Use Case Diagram merupakan pemodelan untuk melakukan (behavior) sistem informasi yang akan dibuat. Use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Berikut adalah simbol-simbol yang ada pada diagram use case (Rosa dan Shalahuddin, 2015):

Tabel 2.1 Simbol *Use Case* Diagram

No	Simbol	Keterangan
1.	<p><i>Use case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit - unit yang saling bertukar pesan antar unit atau aktor biasanya dinyatakan dengan menggunakan kata kerjadi awal frasenama <i>use case</i>
2.	<p>Aktor/ <i>Actor</i></p> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama <i>actor</i>
3.	<p>Association</p> 	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
4.	<p>Extend</p> 	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu
5.	<p>Include</p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya.



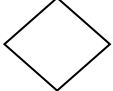

6.	<b>Generalization</b> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
----	--	--

### 2.2.3 Activity Diagram

*Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Perlu diperhatikan bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas (Rosa dan Shalahuddin, 2015):

Tabel 2.2 Simbol - simbol diagram *Activity diagram*








No	Simbol	Keterangan
1.	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.	Decision 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
4.	Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
5.	Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

### 2.2.4 Class Diagram

*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

Berikut adalah simbol-simbol yang ada pada diagram Kelas (Rosa dan Shalahuddin, 2016:146) :

Tabel 2.3 Simbol *Class Diagram*

No.	Simbol	Keterangan
1.	 Kelas	Kelas pada struktur system
2.	Antarmuka/interface 	Sama dengan konsep interface dalam pemrograman berorientasi objek
3.	Asosiasi 	Relasi antara kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity
4.	Asosiasi berarah 	Relasi antara kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity
5.	Generalisasi 	Relasi antar kelas dengan makna generalisasi – spesialisasi (umum khusus)
6.	Kebergantungan 	Relasi antar kelas dengan makna kebergantungan antar kelas
7.	Agregrasi 	Relasi antar kelas dengan makna semua bagian


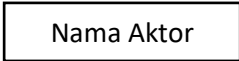
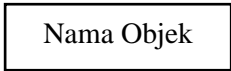








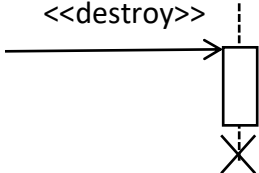
### 2.2.5 Sequence Diagram

Diagram *sequence* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Untuk menggambarkan diagram sekuen maka harus diketahui objek- objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

Berikut adalah simbol-simbol yang ada pada diagram sekuen (Rosa dan Shalahuddin,2015:165):

Tabel 2.4 Simbol *Sequence*

No.	Simbol	Deskripsi
1	Aktor  Atau  Nama Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
2	Objek  Nama Objek	Menyatakan objek yang berinteraksi pesan.
3	Garis hidup / <i>Lifeline</i> 	Menyatakan kehidupan suatu objek.
4	Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
5	Pesan tipe <i>create</i> <<create>> 	Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat.
6	Pesan tipe <i>call</i> 1: nama_metode() 	Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri.

7	Pesan tipe <i>send</i> 1: masukan 	Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
8	Pesan tipe <i>return</i> 1: keluaran 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
9	Pesan tipe <i>destroy</i> <<destroy>> 	Menyatakan suatu objek mengakhiri hidup objek lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i>

### 2.2.6 Skenario Use Case

Setiap use case diagram dilengkapi dengan skenario, skenario use case / use case skenario adalah alur jalannya proses use case dari sisi aktor dan system. Berikut adalah format tabel skenario use case. Skenario use case dibuat per use case terkecil, misalkan untuk generalisasi maka scenario yang dibuat adalah use case yang lebih khusus. Skenario normal adalah scenario bila system berjalan normal tanpa terjadi kesalahan atau error. Sedangkan skenario alternatif adalah scenario bila system tidak berjalan normal atau mengalami error. Skenario normal dan skenario alternatif dapat berjumlah lebih dari satu. Alur skenario inilah yang nantinya menjadi landasan pembuatan sequence diagram / diagram sekuen.

## 2.3 Teori Dasar Pemrograman Perangkat Lunak

Berikut ini adalah teori-teori perangkat lunak yang digunakan dalam penelitian ini.

### 2.3.1 Database

Database merupakan pangkalan data (Tempat Data) yang disusun sedemikian rupa dengan tujuan penyimpanan data yang lebih efektif dan efisien serta mengikuti

aturan-aturan alat desain database terlepas dari perangkat lunak pemrograman yang digunakan. Bahasa pemrograman yang digunakan dalam database pada umumnya menggunakan bahasa pemrograman *database Structured Query Language (SQL)*.

Pengertian dasar dalam *Database* antara lain:

1. *Entity*

Orang, tempat, kejadian konsep yang informasinya menarik untuk disimpan.

2. *Atribut*

Karakteristik yang menjadi ciri suatu entitas.

3. *Field*

Suatu unit informasi mengenai suatu entitas yang mempunyai arti.

4. *Data Value*

Isi atau nilai dari suatu data elemen.

5. *Record*

Kumpulan dari suatu *Field* informasi mengenai entitas tertentu atau kumpulan dari item data yang saling berhubungan.

6. *File*

Kumpulan dari elemen tetapi data *Value* berbeda.

7. *Tabel*

Untuk memberikan atau membuat struktur *Database*.

Beberapa bentuk *key* yang ada dalam *Database*:

1. *Superkey*

Satu atribut atau lebih yang secara unik mengidentifikasi sebuah tuple didalam relasi.

2. *Candidatkey*

Atribut didalam relasi yang biasanya mempunyai nilai unik.

3. *Primarykey*

*Candidate key* yang dipilih untuk mengidentifikasi tuple secara unik didalam relasi.

4. *Alternatekey*

*Candidate key* yang tidak dipilih sebagai *Primarykey*.

## 5. Foreignkey

Atribut dengan domain yang sama dan menjadi kunci utama pada sebuah relasi tetapi pada relasi lain atribut tersebut hanya sebagai atribut biasa.

*Structured Query Language (SQL)* adalah salah satu dari sekian banyak bahasa pemrograman *database* yang paling populer dan menjadi standar perintah *database*. Dengan bahasa SQL memungkinkan kita untuk membuat *database*, menambah, menghapus, mengubah, dan mencari data.

Secara umum *SQL* hanya memiliki 3 komponen penting, yang digunakan untuk mendefinisikan, memanipulasi serta sebagai pengontrol data yang ada pada sebuah *database*. ketiga komponen tersebut antara lain:

### 1. Data Definisi Language (DDL)

DDL merupakan bahasa atau perintah pada SQL yang digunakan untuk mendefinisikan data seperti menulis, menghapus dan mengatur atau mengubah data. Jenis-jenis Query DDL yaitu sebagai berikut:

#### 1) CREATE

- *CREATE DATABASE* digunakan untuk membuat *database* baru
- *CREATE FUNCTION* digunakan untuk membuat fungsi di *database*
- *CREATE INDEX* digunakan untuk membuat index
- *CREATE TABEL* digunakan untuk membuat tabel baru
- *CREATE VIEW* digunakan untuk membuat laporan
- *CREATE PROCEDURE* digunakan untuk membuat prosedur dalam *database*
- *CREATE TRIGGER* digunakan untuk membuat trigger (perintah sql yang dikerjakan otomatis pada event tertentu)

#### 2) ALTER, RENAME

- *ALTER DATABASE* digunakan untuk mengubah pengaturan *database*
- *ALTER FUNCTION* digunakan untuk mengubah function
- *ALTER PROCEDURE* digunakan untuk mengubah prosedur
- *ALTER TABLE* digunakan untuk mengubah susunan pengaturan tabel

- *ALTER VIEW* digunakan untuk mengubah pengaturan laporan
- *RENAME TABLE* digunakan untuk mengubah nama tabel

### 3) *DROP*

- *DROP DATABASE* digunakan untuk menghapus database
- *DROP FUNCTION* digunakan untuk menghapus function
- *DROP INDEX* digunakan untuk menghapus index
- *DROP TABLE* digunakan untuk menghapus tabel
- *DROP PROCEDURE* digunakan untuk menghapus prosedur
- *DROP TRIGGER* digunakan untuk menghapus trigger
- *DROP VIEW* digunakan untuk menghapus format laporan

## 2. *Data Manipulation Language*

DML adalah kependekan dari *Data Manipulation Language*. DML adalah kumpulan perintah *SQL* yang berhubungan dengan proses mengolah dan memanipulasi data dalam *table*.

DML tidak terkait dengan perubahan struktur dan definisi tipe data dari objek database. Jenis *Query* DML akan yaitu sebagai berikut:

- 1) *SELECT* digunakan untuk menampilkan data
- 2) *INSERT* digunakan untuk menambahkan data baru
- 3) *UPDATE* digunakan untuk mengubah data yang sudah ada
- 4) *DELETE* digunakan untuk menghapus data

## 3. *Data Control Language (DCL)*

**Data Control Language (DCL)** adalah jenis instruksi *SQL* yang berkaitan dengan manajemen hak akses dan pengguna (*user*) yang dapat mengakses database maupun tabel. Jenis *Query* DCL adalah sebagai berikut:

- 1) *GRANT*: Perintah ini digunakan untuk memberikan hak/ijin akses oleh administrator (pemilik utama) server kepada user (pengguna biasa). Hak akses tersebut berupa hak membuat (*CREATE*), mengambil (*SELECT*), menghapus (*DELETE*), mengubah (*UPDATE*), dan hak khusus berkenaan dengan sistem databasenya.

2) *REVOKE* : Perintah ini memiliki kegunaan terbalik dengan *GRANT*, yaitu untuk menghilangkan atau mencabut hak akses yang telah diberikan kepada user oleh administrator.

Secara umum tipe data terbagi menjadi 3 jenis yaitu string (tipe string) dan numeric (angka).

#### 1. Tipe Data *String* (huruf)

Tipe data sting memiliki kelebihan dapat menyimpan data dalam berbagai jenis karakter baik huruf, angka, karakter special, waktu dan tanggal, dan kombinasi dari jenis karakter.

Contoh dari tipe data string ini antara lain *blob*, *char*, *char for bit data*, *clob*, *long varchar*, *long varchar for bit data*, *varchar*, dan *xml*.

#### 2. Tipe Data *Numerik*

Tipe data numeric (angka) hanya digunakan untuk menyimpan data angka dan saat mencoba menyimpan data selain angka database akan mengalami kesalahan. Contoh tipe data numeric antara lain *smallint*, *integer*, *bigint*, *real*, *double*, *float*, *decimal*, dan *numeric*.

#### 3. Tipe Data Waktu

Selain tipe string dan numeric database juga mempunyai tipe data untuk menyimpan tanggal dan waktu. Contoh tipe data waktu ini seperti *date*, *time*, dan *timesamp*

Database mempunyai beberapa statemen yang sering digunakan untuk mengelola pemrograman database. Stetemen tersebut adalah sebagai berikut:

##### 1. *Statemen Insert*

Statemen insert digunakan untuk perintah memasukkan data/nilai kedalam suatu tabel.

##### 2. *Statemen Select*

Statemen Select digunakan untuk mencari dan mendapatkan data kemudian ditampilkan dalam sebuah tabel.

##### 3. *Statemen Update*

Statemen update digunakan untuk mengubah atau mengedit data yang sebelumnya telah dimasukkan ke dalam tabel.

#### 4. *Statemen Delete*

Statemen delete digunakan untuk menghapus sebuah baris atau record yang terdapat didalam tabel.

### 2.3.2 Sekilas Tentang MySQL

Menurut Kadir (2015) Mengemukakan bahwa "MySQL adalah nama database server, database server adalah server yang berfungsi untuk menangani database, MySQL adalah database server yang sangat populer dan banyak digunakan untuk menangani data yang disajikan dihalaman Web". Raharjo (2015:16), menjelaskan *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*).

#### a) Cara menjalankan *MySQL*

Salah satu cara penanganan *MySQL* dengan menggunakan *PHP MyAdmin*. Dapat ditempuh melalui beberapa cara, antara lain sebagai berikut.

- 1) Ketikkan alamat *http://localhost/phpmyadmin*
- 2) Mulai membuat *database*, ketikkan nama *database* baru kita di *form* berikut.
- 3) Jika berhasil maka *PHPMyAdmin* akan memberi laporan seperti ini, dan akan langsung masuk ke dalam *database* tersebut.
- 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*, *autoincrement* dan sebagainya. Lalu jika sudah selesai klik *save*. Jika berhasil maka kalian akan langsung diarahkan ke *table structure* seperti dibawah ini.
- 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).

- 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.
- 8) Yang terakhir adalah menyisipkan *field*, pilih *tab structure*, setelah selesai mengisi tabel selanjutnya klik *go*, dan kalian akan diarahkan ke laman seperti berikut, dan jika sudah klik *save*.

b) Perintah dalam *MySQL*

Berikut adalah beberapa intruksi bantuan yang ada pada *MySQL* dan semua perintah ini dapat langsung di ketik pada *prompt mysql*. (;) merupakan tanda yang menyatakan bahwa semua *query* harus diakhiri dengan tanda titik koma (;). Tanda ini menunjukkan bahwa *query* telah berakhir dan siap dieksekusi.

- 1) *Help* (\h) tanda ini digunakan untuk menampilkan *file* bantuan.
- 2) ? (\?) perintah ini sama memiliki fungsi sama dengan *help*.
- 3) *Clear* (\c) tanda ini berguna untuk membersihkan semua perintah yang telah berjalan dalam satu *prompt*, baik perintah salah maupun benar.
- 4) *Connect* (\r) tanda ini untuk melakukan penyegaran koneksi kedalam *database* yang ada pada *server host*.
- 5) *Ego* (\G) tanda ini berguna untuk menampilkan data secara *horizontal* (kekanan).
- 6) *Go* (\g) tanda ini memberi perintah *server* untuk mengeksekusi.
- 7) *Note* (\t) tanda ini berguna untuk mendokumentasikan semua *query* dan hasilnya kedalam sebuah *file* yang akan disimpan kedalam direktori *server*.
- 8) *Print* (\p) tanda ini berguna untuk mencetak semua *query* yang telah diperintahkan kelayar.
- 9) *Prompt*(\R) tanda ini memiliki perintah mengubah *prompt* standar *mysql* menjadi *prompt* yang dikehendaki.
- 10) *Quit* (\q) tanda ini memiliki perintah untuk keluar dari *server mysql*.
- 11) *Source* (\.) tanda ini untuk mengeksekusi *query* yang berasal dari *file* luar.  
 Status (\s) tanda ini untuk melihat status *server* yang sedang digunakan.



## 2.4 Referensi Jurnal Yang Digunakan

No	Nama	Judul	Keterangan
1	Halimah Yunus, S.Kom., M.T.I, Susanti Saleh, S.Kom., M.T.I	Pengembangan Dan Pelatihan E-Commerce Hasil Kerajinan Napi Perempuan Lapas Way Hui Bandar Lampung	
2	TM. Zaini, M.TI, Driya Wiryawan	Pengelolaan Data Produksi Berbasis Web Pada Cv. Sumber Proteina Bandar Lampung	
3	Agus Rahardi, S.Kom., M.T.I, Neni Purwati, S.Kom., M.T.I, Halimah Yunus, S.Kom., M.T.I	Perancangan Website Program Studi Sistem Informasi Institut Informatika Dan Bisnis Darmajaya Bandar Lampung	
4	Arman Suryadi Karim, S.Kom., M.T.I, Melda Agarina, S.Kom., M.T.I	Rancang Bangun Sistem Informasi Kegiatan Seminar Nasional Berbasis Web Pada Institut Informatics Dan Bisnis Darmajaya	
5	Hendra Kurniawan, S.Kom., M.T.I, Wicakso Bandung Bondowoso	Sistem Informasi Terintegrasi Tugas Akhir/Skripsi Berbasis Web (Studi Kasus: Jurusan Sistem Informasi Institut Informatika Dan Bisnis Darmajaya)	