



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Sujatmiko (2012:156), komputer adalah mesin yang data mengolah data digital dengan mengikuti serangkaian perintah atau program yang memegang peran penting dalam teknologi komunikasi.

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

Dari kedua pengertian diatas dapat disimpulkan bahwa komputer merupakan alat bantu mesin yang mengolah data berdasarkan perintah masing-masing.

2.1.2 Pengertian Perangkat Lunak (*Software*)

Sujatmiko (2012:256), perangkat lunak merupakan kumpulan beberapa perintah yang dieksekusi oleh mesin komputer dalam menjalankan pekerjaannya. Perangkat lunak merupakan catatan bagi mesin komputer untuk menyimpan perintah ataupun dokumen.

Sukamto dan Shalahuddin (2013:2), perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan model desain, dan cara penggunaan (*user manual*).

Jadi, berdasarkan pengertian dari para ahli diatas penulis menyimpulkan bahwa perangkat lunak merupakan suatu program yang berupa intruksi-intruksi untuk melakukan tugas tertentu.



2.1.3 Pengertian Basis Data (*Database*)

Kadir (2014:218), basis data adalah suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi.

Sujatmiko (2012:76), basis data adalah representasi kumpulan fakta yang saling berhubungan disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.

Dari pengertian diatas dapat disimpulkan penulis bahwa basis data merupakan suatu pengorganisasian sekumpulan data yang fakta dan berhubungan disimpan sedemikian rupa dan tanpa pengulangan yang tidak perlu.

2.2 Teori Judul

2.2.1 Pengertian Aplikasi

Asropudin (2013:7), aplikasi adalah *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Microsoft Word, Microsoft Excel*.

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

Jadi, berdasarkan pengertian diatas dapat disimpulkan bahwa aplikasi adalah *software* yang dibuat oleh suatu perusahaan dalam mengerjakan tugas tertentu.

2.2.2 Pengertian Penjemputan

Menurut Kamus Besar Bahasa Indonesia pada tahun 2012, Penjemputan adalah proses,cara, perbuatan menjemput.



2.2.3 Pengertian Barang

Menurut Kamus Besar Bahasa Indonesia pada tahun 2012 , Barang adalah benda umum (segala sesuatu yang berwujud atau berjasad).

Sholihin (2016:20), “Barang adalah setiap benda bergerak atau tidak bergerak, berwujud atau tidak berwujud, dapat dihabiskan, yang dipakai, digunakan, dan dapat dimanfaatkan.”

2.2.4 Pengertian Android

Kadir (2013:2), Android adalah sistem operasi yang bersifat open source untuk digunakan pada ponsel, netbook, e-reader, dan tablet.

2.2.5 Pengertian PT Pos Indonesia (Persero) Palembang

Pengertian PT Pos Indonesia (Persero) Palembang adalah sebuah badan usaha milik negara (BUMN) Indonesia yang bergerak di bidang layanan pos. Saat ini, merupakan perseroan terbatas dan sering disebut dengan PT Pos Indonesia.

2.2.6 Pengertian Aplikasi Penjemputan Barang pada PT Pos Indonesia (Persero) Palembang

Pengertian Aplikasi Penjemputan barang pada PT Pos indonesia (Persero) Palembang adalah aplikasi untuk memudahkan pelanggan dalam melakukan pengantaran barang.

2.3 Teori Khusus

2.3.1 *Unified Modeling Language (UML)*

Sukamto dan Shalahuddin (2013:133), *UML (Undefined Modelling Language)* adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.



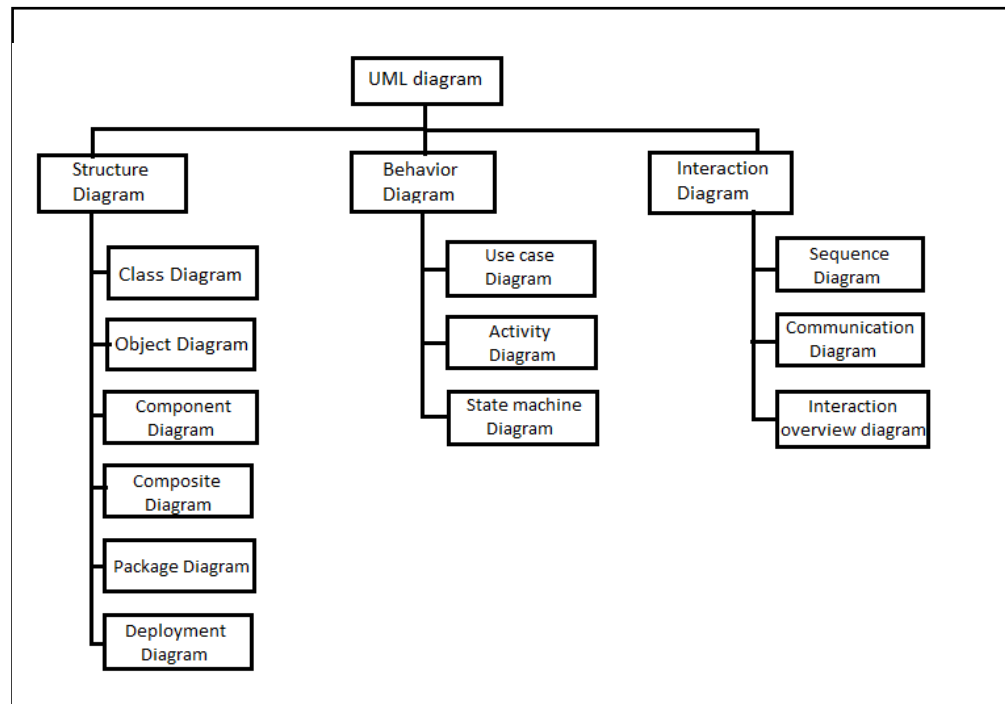
UML menyediakan bahasa pemodelan visual yaitu proses penggambaran informasi-informasi secara grafis dengan notasi-notasi baku yang telah disepakati sebelumnya. Dengan menggunakan pemodelan *UML* ini, pengembang dapat melakukan:

- a. Tinjauan umum bagaimana arsitektur sistem secara keseluruhan.
- b. Penelaahan bagaimana objek-objek dalam sistem saling mengirimkan pesan (*message*) dan saling bekerjasama satu sama lain.
- c. Menguji apakah sistem/perangkat lunak sudah berfungsi seperti yang seharusnya.
- d. Dokumentasi sistem/perangkat lunak untuk keperluan-keperluan tertentu di masa yang akan datang.



2.3.1.1 Macam-macam Diagram UML

Pada *UML* terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar dibawah ini



Gambar 2.1 Macam-macam Diagram UML

Berikut ini penjelasan singkat dari pembagian kategori tersebut.

a. *Structure Diagram*

Yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.

b. *Behavior Diagram*

Yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.

c. *Interaction Diagram*




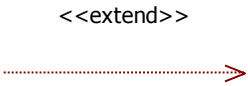
Yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.





2.3.1.2 Use Case Diagram

Sukamto dan Shalahuddin (2013:155), *use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat.

Tabel 2.1 Simbol-simbol dalam *Use case*

No	Simbol	Deskripsi
1	Use Case 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
2	Aktor / actor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
3	Asosiasi / association 	Komunitas antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.
4	Ekstensi / extend 	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tamhanan itu; biasanya use case yang menjadi extend-nya merupakan jenis yang sama dengan use case yang menjadi induknya.




5	Generalisasi / <i>Generalization</i> 	Hubungan generalisasi dan spesifikasi (umum-khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6	<i>Include</i> <<include>> 	Relasi use case tambahan ke sebuah use case di mana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini; <i>include</i> berarti use case yang ditambahkan akan selalu dipanggil saat use case tambahan dijalankan.

(Sumber: Sukamto dan Shalahuddin, 2013:155)

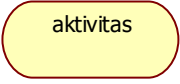
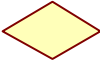


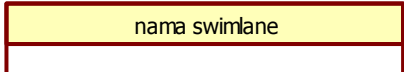
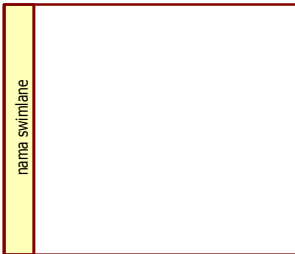
2.3.1.3 Activity Diagram

Pengertian *Activity Diagram* menurut Sukamto dan Shalahuddin (2013:161), diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.

Tabel 2.2 Simbol-simbol dalam Activity Diagram

No	Simbol	Deskripsi
1	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.



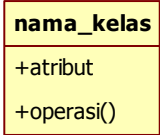





2	aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3	Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4	Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5	Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6	Swimlane  atau 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

(Sumber: Sukamto dan Shalahuddin, 2013:162)


2.3.1.4 Class Diagram

Sukamto dan Shalahuddin (2013:141), diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

Tabel 2.3 Simbol-Simbol *Class Diagram*

No	Simbol	Deskripsi
1	Kelas 	Kelas pada struktur sistem.
2	Antar muka/ <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3	Asosiasi / <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4	Asosiasi berarah / <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5	Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
6	Kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antarkelas.



7	Agregasi / <i>aggregation</i> 	Relasi antarkelas dengan makna semua-bagian (whole-part).
---	--	---

(Sumber: Sukamto dan Shalahuddin, 2013:146)

2.4 Teori Program

2.4.1 Pengetian Eclipse

Pengertian *Eclipse* menurut Juhara (2016:44), Eclipse IDE adalah tempat di mana Anda akan menulis kode program. Aplikasi ini sifatnya *open source* gratis yang awalnya dikembangkan oleh IBM, lalu dikelola dan dikembangkan lebih lanjut oleh Eclipse Foundation. Eclipse IDE tersedia untuk beragam platform sistem operasi, mulai dari Windows, Mac OS, hingga Linux. Eclipse IDE sesungguhnya adalah IDE yang sifatnya umum dan modular.

Komputer (2013:10), Eclipse adalah software development environment multibahasa yang berfitur extensible plugin. Extensible plugin berarti pengembang dapat mengembangkan beberapa jenis aplikasi, misalkan Java, Ada, C, C++, atau Python dalam satu aplikasi IDE dengan cara menambahkan plugin.

Jadi, dari pengertian diatas dapat disimpulkan bahwa *Eclipse* adalah suatu software tempat anda mengkoding program.

2.4.2 Pengertian Javascript

Pengertian *Javascript* menurut para ahli Suryana dan Koesheryatin (2014:101), menjelaskan *Javascript* adalah bahasa *script* berdasar pada objek



yang memperbolehkan pemakai untuk mengendalikan banyak aspek interaktif pemakai pada suatu dokumen HTML.

Sidik (2011:1), *Javascript* adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokumen HTML yang ditampilkan dalam browser menjadi lebih interaktif, tidak sekadar indah saja.

Dari pengertian para ahli diatas dapat disimpulkan bahwa *Javascript* adalah bahasa yang digunakan untuk membuat program.

2.4.3 Pengertian *jQuery*

Pengertian *jQuery* menurut Rohingun (2015:1), *jQuery* merupakan salah satu dari sekian banyak JavaScript library yaitu kumpulan fungsi JavaScript yang siap pakai, sehingga mempermudah dan mempercepat kita dalam membuat kode JavaScript. Dengan menggunakan *jQuery*, skrip JavaScript yang panjang dapat disingkat menjadi beberapa baris kode saja.

2.4.4 Pengertian *PHP*

Pengertian *PHP* menurut Winarno, dkk. (2014:49) menjelaskan, *PHP* atau *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).

Badiyanto (2013:32) “*PHP : Hypertext Preprocessor* adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML/*PHP* banyak dipakai untuk embuat situs web dinamis”.

Jadi menurut beberapa pendapat di atas, *PHP (Hypertext Preprocessor)* adalah sebuah bahasa pemrograman berbasis *web* yang digunakan untuk membuat halaman HTML.



2.4.4.1 Script PHP

PHP yang merupakan sebuah bahasa scripting yang terpasang pada HTML, dan skrip HTML (*Hyper Text Markup Language*) adalah bahasa standar untuk membuat halaman *web*. Biasa *file* dituliskan dengan ekstensi *.htm* atau *.html*.

```
<html>

  <head>

    <title>Latihan PHP</title>

  </head>

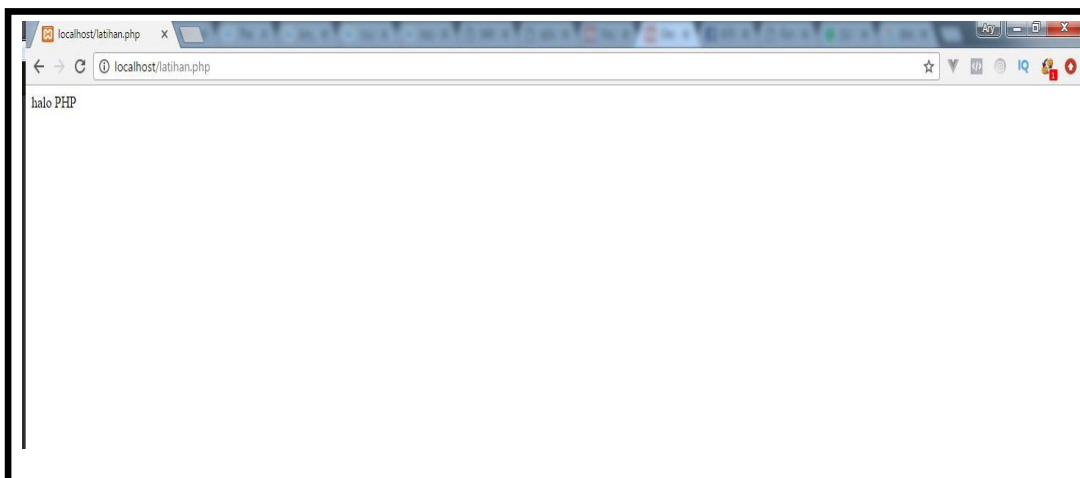
  <body>

    halo PHP

  </body>

</html>
```

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



Gambar 2.2 Tampilan program HTML pada *browser*



2.4.4.2 Teknik Penulisan Script *PHP*

Zaki et al. (2014:24), block kode *PHP* dimulai dengan sintak tag `<?PHP` dan di akhiri dengan `?>`, berikut adalah contoh blok kode *PHP*:

```
<?PHP

//kode tertentu diisikan di sini

?>
```

Selain itu ada beberapa tag lain yang bisa di pakai, seperti:

Tabel 2.5 Script Dasar *PHP*

Jenis Tag	Tag Pembuka	Tag Penutup
Tag standar	<code><?PHP</code>	<code>?></code>
Tag pendek	<code><?</code>	<code>?></code>
Tag ASP	<code><%</code>	<code>%></code>
Tag script	<code><script language="PHP"></code>	<code></script></code>

(Sumber: Zaki et al. 2014:24-25)

Echo sering digunakan di dalam *PHP*. Fungsinya untuk meng-ouput data. Berikut ini contoh penggunaan echo:

```
<?PHP

Echo '<p> selamat datang </p>';

?>
```

Di dalam *script PHP* juga dapat membuat komentar dan komentar tersebut tidak akan dieksekusi sebagai sebuah *script*. Beberapa cara untuk memberi keterangan



di dalam *script PHP* adalah sebagai berikut:

- 1) Gunakan *tag /** dan akhiri *tag */* apabila jumlah keterangan lebih dari 1 baris.
- 2) Gunakan *tag //*, *tag* ini digunakan untuk keterangan yang hanya terdiri dari 1 baris saja.
- 3) Gunakan *tag #*, *tag* ini juga digunakan untuk 1 baris komentar saja.

Dalam penulisannya, baris perintah *script PHP* selalu diakhiri dengan menuliskan tanda titik koma (;). Satu baris *script PHP* tidak harus berada dalam satu baris, melainkan dapat menuliskan perintah *script PHP* lebih dari satu baris.

2.4.5 Pengertian MySQL

Sukanto et al. (2014:46), “SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada RDBMS. SQL awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus”.

Winarno (2014:102), “MySQL adalah sebuah software database. MySQL merupakan tipe data relasional yang artinya MySQL menyimpan datanya dalam bentuk tabel-tabel yang saling berhubungan”.

Berdasarkan beberapa pendapat diatas, MySQL merupakan sebuah bahasa yang digunakan untuk mengelolah data dalam bentuk tabel-tabel yang saling berhubungan.

2.4.5.1 Fungsi-Fungsi MySQL

Sejumlah fungsi yang berawalan **mysql_** yang digunakan untuk mengakses *database server MySQL* adalah sebagai berikut:

Tabel 2.6 Fungsi-fungsi MySQL

No.	Fungsi MySQL	Penggunaan
1	<i>Mysql_connect()</i>	Membuat hubungan ke <i>database MySQL</i> yang terdapat pada suatu host



2	<i>Mysql_close()</i>	Menutup hubungan ke <i>database MySQL</i>
3	<i>Mysql_select_db()</i>	Memilih <i>database</i>
4	<i>Mysql_query()</i>	Mengeksekusi permintaan terhadap sebuah tabel atau sejumlah tabel
5	<i>Mysql_db_query()</i>	Menjalankan suatu permintaan terhadap suatu <i>database</i>
6	<i>Mysql_num_rows()</i>	Memperoleh jumlah baris dari suatu hasil permintaan yang menggunakan <i>SELECT</i>
7	<i>Mysql_affected_rows()</i>	Memperoleh jumlah baris yang dikenai operasi <i>INSERT, DELETE, UPDATE</i>
8	<i>Mysql_num_fields()</i>	Memperoleh jumlah kolom pada suatu hasil permintaan
9	<i>Mysql_fetch_row()</i>	Menghasilkan <i>array</i> /baris yang berisi seluruh kolom dari sebuah baris pada suatu himpunan hasil
10	<i>Mysql_fetch_array()</i>	Menghasilkan <i>array</i> /baris yang berisi seluruh kolom dari sebuah baris pada suatu himpunan hasilnya akan disimpan dua kali pada <i>array</i> hasil
11	<i>Mysql_fetch_field()</i>	Menghasilkan informasi suatu kolom
12	<i>Mysql_data_seek()</i>	Memindahkan pointer pada suatu himpunan hasil supaya menunjuk ke baris tertentu
13	<i>Mysql_field_seek()</i>	Memindahkan pointer pada suatu himpunan hasil supaya menunjuk ke kolom tertentu
14	<i>Mysql_create_db()</i>	Membuat <i>database MySQL</i>
15	<i>Mysql_drop_db()</i>	Menghapus <i>database MySQL</i>
16	<i>Mysql_list_dbs()</i>	Menghasilkan daftar <i>database MySQL</i>
17	<i>Mysql_list_tables()</i>	Memperoleh daftar nama tabel dalam suatu <i>database</i>



18	<i>mysql_list_fields()</i>	Memperoleh daftar nama kolom dalam suatu database
----	----------------------------	---

(Sumber: Abdul Kadir (2008:360-379))

2.4.5.2 Alasan Menggunakan MySQL

Berikut ini beberapa alasan mengapa penulis memilih *MySQL* sebagai *database server*

1. **Fleksibel.** *MySQL* dapat digunakan untuk mengembangkan aplikasi *desktop* maupun aplikasi *web* dengan menggunakan teknologi yang bervariasi. Ini berarti bahwa *MySQL* memiliki fleksibilitas terhadap teknologi yang akan digunakan sebagai pengembang aplikasi, apakah itu *PHP*, *JSP*, *Java*, *Delphi*, *C++*, maupun yang lainnya dengan cara menyediakan *plug-in* dan *driver* yang spesifik untuk masing-masing teknologi tersebut.
2. **Performa Tinggi.** *MySQL* memiliki *query* performansi tinggi dengan demikian proses transaksional dapat dilakukan dengan sangat cepat.
3. **Lintas Platform.** *MySQL* dapat digunakan pada *platform* atau lingkungan sistem operasi yang beragam.
4. **Gratis.** *MySQL* dapat digunakan secara gratis.
5. **Proteksi data yang handal.** Perlindungan terhadap keamanan data merupakan hal nomor satu yang dilakukan oleh para profesional di bidang *database*. *MySQL* menyediakan mekanisme yang powerful untuk menangani hal tersebut, yaitu dengan menyediakan fasilitas manajemen *user*, enkripsi data, dan lain sebagainya.