

## BAB II LANDASAN TEORI

### 2.1 Sistem Informasi

Sistem Informasi (SI) adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi itu untuk mendukung operasi dan manajemen. Dalam arti yang sangat luas, istilah sistem informasi yang sering digunakan merujuk kepada interaksi antara orang, proses algoritmik, data, dan teknologi. Dalam pengertian ini, istilah ini digunakan untuk merujuk tidak hanya pada penggunaan organisasi teknologi informasi dan komunikasi (TIK), tetapi juga untuk cara di mana orang berinteraksi dengan teknologi ini dalam mendukung proses bisnis.

Ada yang membuat perbedaan yang jelas antara sistem informasi, dan komputer sistem TIK, dan proses bisnis. Sistem informasi yang berbeda dari teknologi informasi dalam sistem informasi biasanya terlihat seperti memiliki komponen TIK. Hal ini terutama berkaitan dengan tujuan pemanfaatan teknologi informasi. Sistem informasi juga berbeda dari proses bisnis. Sistem informasi membantu untuk mengontrol kinerja proses bisnis.

Alter berpendapat untuk sistem informasi sebagai tipe khusus dari sistem kerja. Sistem kerja adalah suatu sistem di mana manusia dan mesin melakukan pekerjaan dengan menggunakan sumber daya untuk memproduksi produk tertentu dan jasa bagi pelanggan. Sistem informasi adalah suatu sistem kerja yang kegiatannya ditujukan untuk pengolahan (menangkap, transmisi, menyimpan, mengambil, memanipulasi dan menampilkan) informasi.

Dengan demikian, sistem informasi antar-berhubungan dengan sistem data di satu sisi dan sistem aktivitas di sisi lain. Sistem informasi adalah suatu bentuk komunikasi sistem di mana data yang mewakili dan diproses sebagai bentuk dari memori sosial. Sistem informasi juga dapat dianggap sebagai bahasa semi formal yang mendukung manusia dalam pengambilan keputusan dan tindakan.

### 2.2 PHP

PHP (PHP: *Hypertext Preprocessor*) adalah bahasa *server-side* scripting yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Karena PHP merupakan *server-side* scripting maka sintaks dan perintah-perintah PHP akan dieksekusi di server kemudian hasilnya dikirimkan ke *browser* dalam format HTML. Dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh user sehingga keamanan halaman Web lebih terjamin. PHP dirancang untuk membentuk halaman Web yang dinamis, yaitu halaman Web yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman Web.

PHP termasuk dalam *Open Source Product*, sehingga *source code* PHP dapat diubah dan didistribusikan secara bebas. PHP juga mampu lintas *platform*. Artinya PHP dapat berjalan dibanyak sistem operasi yang beredar saat ini, diantaranya: Sistem Operasi Microsoft Windows (semua versi), Linux, Mac OS, Solaris. PHP dapat dibangun sebagai modul pada *web server* Apache dan sebagai binary yang dapat berjalan sebagai CGI (Common Gateway Interface). PHP dapat mengirim HTTP *header*, dapat mengatur *cookies*, mengatur *authentication* dan *redirect user*.

Salah satu keunggulan yang dimiliki PHP adalah kemampuannya untuk melakukan koneksi ke berbagai macam software sistem manajemen basis data/*Database Management System* (DBMS), sehingga dapat menciptakan suatu halaman web yang dinamis. PHP mempunyai konektivitas yang baik dengan beberapa DBMS antara lain Oracle, Sybase, mSQL, MySQL, Microsoft SQL Server, Solid, PostgreSQL, Adabas, FilePro, Velocis, dBase, Unix dbm, dan tak terkecuali semua database ber-interface ODBC.

#### 2.2.1 Skrip PHP

PHP yang merupakan sebuah bahasa scripting yang terpasang pada HTML (*Hyper Text Markup Language*) adalah bahasa standar untuk membuat halaman web.

#### 2.2.2 Variabel PHP

Sebuah bahasa pemrograman menyediakan variabel, variabel biasanya untuk menyimpan suatu nilai, dan nilai tersebut bisa diubah setiap saat. Nama variabel dalam PHP diawali dengan karakter "\$" contoh \$nama, \$jumlah.

### 2.3 Yii

Yii adalah *framework* (kerangka kerja) PHP berbasis komponen, berkinerja tinggi untuk pengembangan aplikasi Web berskala besar. Yii menyediakan *reusability* maksimum dalam pemrograman Web dan mampu meningkatkan kecepatan pengembangan secara signifikan. Nama Yii (dieja sebagai /i:/) singkatan dari “Yes It Is!”.

Yii adalah *framework* pemrograman umum berbasis Web yang bisa dipakai untuk mengembangkan semua jenis aplikasi Web. Dikarenakan sangat ringan dan dilengkapi dengan mekanisme *caching* tinggi yang canggih, Yii sangat cocok untuk pengembangan aplikasi dengan lalu lintas tinggi, seperti portal, forum, *Content Manajemen System* (CMS), sistem *e-commerce*, dan lain-lain.

Aplikasi yang dibangun menggunakan *framework* Yii membutuhkan beberapa *file* pustaka (lib) yang disediakan dalam *framework* Yii, dan menggunakan metode pemrograman berorientasi objek, atau dalam bahasa Inggris disebut (*Object Oriented Programming*).

Membangun sebuah aplikasi dengan *framework* Yii, pengembang cukup menggunakan kelas yang tersedia pada PHP sendiri. Semua yang dibangun menggunakan Yii menggunakan arsitektur *Model-View-Controller* (MVC). Syarat untuk menjalankan aplikasi web berbasis Yii, memerlukan server Web yang mendukung PHP 5.1.0.

*Framework* Yii seperti pada kebanyakan *framework* PHP lainnya, *Framework* Yii adalah MVC *framework*. Yii mempunyai kekayaan fitur dan kejelasan dokumentasi, sehingga sangat efisiensi dalam pembuatan aplikasi, Yii bukan berasal dari produk beberapa proyek maupun konglomerasi pekerjaan pihak ketiga.

## 2.4 Framework

*Framework* bila diterjemahkan secara harfiah berarti “kerangka kerja”. *Framework* juga dapat diartikan sebagai kumpulan *script* (terutama *class* dan fungsi) yang dapat membantu *developer/programmer* dalam menangani berbagai masalah dalam pemrograman seperti koneksi ke *database*, pemanggilan variabel, berkas dan lain-lain sehingga *developer* lebih fokus dan lebih cepat membangun aplikasi.

Secara sederhana bisa dijelaskan bahwa *framework* adalah kumpulan fungsi dari kumpulan library. Sehingga seorang *programmer* tinggal memanggil fungsi-fungsi yang sudah ada di dalam *framework*. Tentunya cara menggunakan fungsi-fungsi tersebut sudah ditentukan oleh *framework*.

## 2.5 MVC (Model, View, Controller)

*Framework* menggunakan konsep yang sama dalam membangun suatu aplikasi, yaitu konsep MVC (*Model View Controller*). Konsep MVC adalah konsep pemisahan antara logika dengan tampilan dan database. Manfaat konsep ini adalah membuat *coding logic* lebih sederhana dan dapat membuat *programmer* dapat bekerja secara terpisah dengan *designer* berkuat dengan rancangan penampilan.

Dalam MVC, model menggambarkan informasi (data) dan proses bisnis, *view*(tampilan) berisi elemen antarmuka user seperti *teks*, formulir masukan. Sementara *controller* mengatur komunikasi antar model dan *view*.

### 2.5.1 Model

Model merupakan kelas yang mendasari logika proses dalam aplikasi perangkat lunak dan kelas yang terkait dengannya. Model adalah suatu objek yang tidak mengandung informasi tentang *user interface*. Model juga suatu kelas yang berisi metode/fungsi yang merupakan kumpulan dari proses-proses.

### 2.5.2 View

*View*, merupakan kumpulan dari kelas yang mewakili unsur-unsur dalam antarmuka *user* (semua hal user dapat melihat dan merespon pada layar, seperti tombol, tampilan kotak dan sebagainya).

Dalam *view* terdapat nama yang dipakai untuk mengidentifikasi *file* skrip tampilan saat dipanggil lewat fungsi *render*. Nama *view* sama seperti nama *file* skrip *view*-nya. Sebagai contoh, *file view* edit merujuk pada *file* skrip bernama *edit.php*. Untuk *me-render view*, *controller* menjalankan metod `Ccontroller::render()` dengan nama *view*. Method tersebut akan mencari file *view* terkait didalam direktori `protected/views/controllerID`.

### 1. Layout

*Layout* adalah tata letak *view* khusus yang dipakai untuk mengatur bentuk tampilan. Biasanya berisi bagian antarmuka dengan user yang umum diantara beberapa *view*. *Layout* secara implisit diterapkan saat memanggil menjalankan fungsi `render()`. Secara *default*, skrip *view* `protected/views/layouts/main.php` dipakai sebagai *layout*.

### 2. Widget

Widget adalah *instance* dari `CWidget` atau anak kelas dari `CWidget`. Widget biasanya digunakan dalam skrip *view* untuk menghasilkan beberapa antarmuka user yang kompleks dan berdiri sendiri. Sebagai contoh, widget kalender bisa dipakai untuk menyiapkan antarmuka kalender user yang kompleks.

### 3. View Sistem

*View* (tampilan) sistem merujuk pada tampilan yang dipakai oleh Yii untuk menampilkan kesalahan dan pencatatan (*logging*) informasi. Sebagai contoh, ketika user meminta ke *controller* ketika *controller* tidak ada fungsi *action* maka *view system* akan menampilkan kesalahan dengan *exception*. Yii menampilkan *exception* menggunakan tampilan sistem tertentu. Penampilan *view system* mengikuti beberapa aturan. Nama seperti `error_xx` merujuk pada tampilan untuk menampilkan `CHttpException` dimunculkan dengan kode kesalahan `_xx`. Sebagai contoh, `CHttpException` dimunculkan dengan kode kesalahan 404, *view error 404* akan ditampilkan.

## 2.5.3 Controller

*Controller*, yang merupakan kelas yang menghubungkan model dan *view*, dan digunakan untuk berkomunikasi antara kelas dalam model dan *view*. *Controller* memiliki *action* standar. Ketika permintaan user tidak menetapkan *action* mana yang dijalankan adalah *action* standar. Biasanya *action* disalutnya bernama `actionIndex`. *Action default* bisa diubah dengan mengeset fariabel *instan public*, `Ccontroller::defaultAction`.

- **Route**

*Controller* dan *action* diidentifikasi oleh ID. ID *Controller* dalam format `path/ke/xyz` sesuai dengan letak file kelas *controller* di `protected/controllers/path/ke/XyzController/.php.xyz` harus diganti dengan nama sebenarnya. ID *Action* adalah nama metode *action* tanpa berawalan *action*. Permintaan user ke *controller* dan *action* tertentu sesuai dengan aturan *route*. *Route* dibentuk dengan menggabungkan ID *controller* dan ID *action* yang dipisahkan dengan garis miring. Sebagai contoh, route `tblMhs/khs` merujuk ke `TblMhsController` dan *action* `khs`. Dan secara *default*, URL `http://nama_host/index.php?r=TblMhs/khs` akan meminta *controller* dan *action* `khs`.

- **Instansi Controller**

Instansi *controller* dibuat ketika `CWebApplication` menangani permintaan yang masuk. Berdasarkan ID *controller* yang diberikan, aplikasi akan menentukan kelas *controller* apa dan dimana file kelas ditempatkan, dengan menggunakan aturan berikut :

- Jika `CWebApplication::catchAllRequest` ditetapkan, controller akan dibuat berdasarkan ID *controller* yang diberikan. Ini dipakai terutama untuk penyimpanan aplikasi dalam pemeliharaan mode menampilkan halaman statis.
- Jika ID ditemukan dalam `CWebApplication::controllerMap`, konfigurasi controller terkait akan dipakai dalam membuat turunan controller.
- Jika ID ada dalam format `'path/ke/abc'`, nama kelas controller diasumsikan adalah `AbcController` dan file kelas terkait adalah `protected/controllers/path/ke/AbcController.php`.

- **Action**

Seperti yang dijelaskan diatas, action dapat didefinisikan sebagai metode yang namanya dimulai dengan kata action. Cara lebih lanjut adalah dengan mendefinisikan kelas action dan meminta controller untuk menurunkannya apabila diperlukan. Dengan demikian memungkinkan sebuah action untuk dipakai ulang. Dengan menulis action berbasis-kelas, kita dapat mengatur aplikasi dalam gaya modular.

- **Action Parameter Binding**

Semenjak versi 1.1.4, Yii menambah dukungan binding (pengikatan) parameter action otomatis, yakni sebuah action controller dapat mendefinisikan parameter yang nilainya akan secara otomatis diisi oleh Yii dari nilai `$_GET`.

#### 2.5.4 Alur Kerja MVC

- **User** melakukan permintaan dengan memanggil URL `http://localhost/test/index.php?r=site/login.php`.
- **Skrip** bootstrap membuat sebuah instan aplikasi dan menjalankannya.
- Aplikasi mendapatkan rincian informasi permintaan user dari komponen aplikasi bernama `request`.
- Aplikasi menentukan controller dan aksi terdapat pada fungsi action yang diminta dengan bantuan komponen aplikasi bernama `urlManager`. Dalam contoh ini, controller adalah `Site` yang merujuk pada kelas `SiteController`; dan aksi adalah `login.php` yang arti sebenarnya ditentukan oleh controller.
- Aplikasi memuat instance controller yang diminta untuk selanjutnya menangani permintaan user. Controller menentukan aksi `login.php` merujuk pada sebuah metode bernama `actionLogin` dalam kelas controller. Kemudian membuat dan menjalankan filter (contoh kontrol akses, pengukuran) terkait dengan aksi ini. Aksi dijalankan jika diijinkan filter.
- Aksi membaca model `LoginForm` dengan parameter `$model` yang terdapat pada controller dalam `actionLogin` diambil dari pernyataan :  

```
$model -> attributes=$_POST['LoginForm']; \
```
- Aksi menyiapkan view (tampilan) bernama `login.php` dengan model `LoginForm`.
- View membaca dan menampilkan atribut model `login.php`
- View menjalankan beberapa widget dan view menyiapkan hasil yang dipasangkan dalam layout (tata letak)

## 2.6 OOP PHP

**Object Oriented Programming (OOP)** merupakan metode pemrograman yang berorientasi pada objek. Semua data dan fungsi metode ini dibungkus dalam kelas-kelas atau objek-objek. Jika dibandingkan dengan logika pemrograman terstruktur, setiap objek dapat menerima pesan, memproses data, dan mengirim pesan ke objek lainnya. OOP diciptakan untuk mengatasi keterbatasan pada bahasa pemrograman tradisional. Konsep dari OOP sendiri adalah semua pemecahan masalah dibagi satu kesatuan yang dapat disebut sebagai objek.

### 1. Objek

Sederhananya, sebuah objek adalah kumpulan dari variabel dan fungsi yang dibungkus menjadi satu entitas. Entitas tersebut dapat berupa variabel biasa. Sebuah objek diciptakan melalui sebuah kelas atau dengan istilah instan dari kelas. Objek memiliki 2 elemen utama :

- **Attributes atau Properties** : yaitu nilai-nilai yang tersimpan dalam objek tersebut dan secara langsung maupun tidak langsung menentukan karakteristik dari objek tersebut.
- **Method** : yaitu suatu aksi yang akan dijalankan atau dikerjakan oleh objek tersebut.

### 2. Kelas

Kelas atau class dapat didefinisikan sebagai struktur data dari suatu objek. Lebih jelasnya adalah sebuah bentuk dasar yang mendefinisikan variabel, method umum pada semua objek. Objek sendiri adalah kumpulan variabel dan fungsi yang dihasilkan dari template khusus atau disebut class.

### 3. Menciptakan Objek

Setelah kelas didefinisikan, objek atau disebut dengan istilah instan kelas dapat dibentuk.

### 4. Kelas Extends

Sebuah kelas dapat merupakan perluasan dari kelas yang sebelumnya. Hal ini yang merupakan kelebihan pemrograman berorientasi objek adalah sifat pewarisan (*inheritence*).

## 2.7 Database

Database bisa dikatakan sebagai suatu kumpulan dari kata yang tersimpan dalam tabel dan diatur atau diorganisasikan sehingga data tersebut bisa diambil atau dicari dengan mudah dan efisien.

Database secara umum berarti suatu koneksi data yang saling terkait. Secara Praktis, Basisdata dapat dianggap sebagai suatu penyusunan data yang terstruktur yang disampaikan dalam media pengingat (hard disk) yang tujuannya adalah agar data tersebut dapat diakses dengan mudah dan cepat. Database memiliki beberapa macam yaitu database hierarkis, database jaringan, dan database rasional. Database rasional merupakan database yang saat ini banyak digunakan dan diterapkan pada berbagai platform. Dari PC hingga minikomputer.

### 2.7.1 Pengertian DBMS (Database Managemen System)

Data merupakan fakta mengenai suatu object seperti manusia, benda, peristiwa, konsep, keadaan dan lain sebagainya yang dapat dicatat dan mempunyai arti secara implisit. Data dapat dinyatakan dalam bentuk angka, karakter atau simbol, sehingga bila data dikumpulkan dan saling berhubungan maka dikenal dengan istilah basis data(database).

Basisdata merupakan kumpulan data yang saling berhubungan satu dengan yang lainnya yang diorganisasikan sesuai struktur tertentu dan disimpan dengan baik. Untuk mendapatkan informasi yang berguna. Database Managemen System (DMS) merupakan software yang berbasis komputerisasi. DMS membantu dalam pemeliharaan dalam jumlah besar. Sehingga dengan menggunakan DBMS tidak menimbulkan kekacauan dan dapat digunakan oleh pengguna sesuai dengan kebutuhan.

### 2.7.2 Komponen Utama DBMS

Komponen utama DTMS terdiri dari :

- Hardware yang melakukan pemrosesan menyimpan database.
- Data
- User, Dapat Diklasifikasikan menjadi:

- a. End User
  - Pengguna Aplikasi yang dapat mengoperasikan program aplikasi
  - Pengguna interaktif yang memberikan perintah berbasis tinggi (sintak-sintak query).
- b. Programmer Aplikasi yang membuat program aplikasi
- c. Database administrator yang bertanggung jawab terhadap pengelolaan database.

- Software, Sebagai interface antara user dan database

### 2.7.3 Perintah untuk Mengelola dan Mengoperasikan Data

Perintah untuk mengelola dan mengoperasikan data terdiri dari :

- Data Definition Language:

Perintah yang biasa digunakan oleh DBA, untuk mendefinisikan skema ke DBMS. Skema merupakan deskripsi lengkap tentang struktur fields, record dan hubungan data pada database. Hal yang perlu didefinisikan dalam DBMS:

- Nama Database
- Nama file pada database
- Nama field dan record
- Deskripsi file, record dan field

DDL juga digunakan untuk menciptakan, mengubah dan menghapus database. Yang termasuk dalam kelompok DDL sebagai berikut:

- CREATE : Membuat tabel
- ALTER : Mengubah Struktur tabel
- DROP : Menghapus Tabel.

- Data Manipulation Language:

Perintah yang digunakan untuk mengubah, memanipulasi dan mengambil database. DML dibagi menjadi 2 bagian, yaitu :

- Prosedural yakni menuntut user menentukan data apa saja yang diperlukan dan bagaimana cara mendapatkan data tersebut.
- Non prosedural yakni menuntut user menentukan apa saja yang diperlukan tetapi tidak perlu menyebutkan cara mendapatkannya.

Adapun perintah-perintah yang termasuk dalam DML:

- SELECT yakni digunakan untuk memilih data
- INSERT yakni digunakan untuk menambahkan data
- DELETE digunakan untuk Menghapus data.
- UPDATE digunakan untuk mengubah data.

## 2.8 ERD (Entity Relationship Diagram)

ERD adalah suatu pemodelan dari basisdata relasional yang didasarkan atas persepsi di dalam dunia nyata, dunia yang terdiri dari sekumpulan object yang saling berhubungan antara satu sama lain. Suatu objek disebut entity dan hubungan

yang dimilikinya disebut relationship. Menjadi mentalis suatu entity bersifat unik dan memiliki atribut sebagai pembeda dengan entity lainnya.

E-R Diagram merupakan suatu bahasa pemodelan yang mana posisinya dapat dianalogikan dengan story board dalam industri film, blue print, arsitektur suatu bangunan, miniatur, dan lain-lain. Dalam praktiknya membangun suatu sistem terlebih dahulu dilakukan suatu perencanaan. Pemodelan merupakan suatu sub bagian dari perencanaan secara keseluruhan sebagai salah satu pemodelan setidaknya memiliki beberapa karakteristik dan manfaat sebagai berikut:

- Memudahkan untuk dilakukannya analisis dan perubahan sistem sejak dini yang bersifat murah dan cepat.
- Memberikan gambaran umum akan sistem yang akan dibuat sehingga memudahkan developer.
- Menghasilkan Dokumentasi yang baik untuk client sebagai bahan diskusi dengan bentuk ERD itu sendiri.
- Kamus data bagi para pengembang handy database.

Struktur dari ERD secara umum ialah terdiri dari:

- Entitas merupakan object utama yang informasi akan disimpan, biasanya berupa data ex; Mahasiswa, Dosen, Nasabah, Mata kuliah, Ruangan, dan lain lain. Object dapat berwujud nyata maupun abstrak.
- Atribut merupakan deskripsi dari object yang bersangkutan.
- Relationship merupakan suatu hubungan yang terjalin antara dua entitas yang ada.

## 2.9 Flowchart

Flowchart adalah suatu gambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. Flowchart menolong analis dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian.

Flowchart biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut.

### 2.9.1 Simbol simbol Flowchart

Simbol-simbol flowchart yang biasa digunakan adalah simbol flowchart standart yang dikeluarkan oleh ANSI dan ISO.

**Tabel 2.1 Simbol Standart Flowchart**

Simbol	NAMA	Fungsi
	Transminator	Permulaan / Akhir program
	Garis Alir (Flow Line)	Arah aliran program
	Preparation	Proses inisialisasi/ pemberian harga awal
	Proses	Proses penghitungan / proses pengelolaan data
	Input/ output data	Proses input/ output data, parameter, informasi

	Predefinised proses (sub Program)	Permulaan sub program/ proses menjalankan sub program
	Decision	Pebandingan pernyataan, penyeleksi dala yang memberikan pilihan untun langkah utnuk selanjutnya.
	On page	Penhubung bagian bagian flowcart yang berada pada suatu halman
	Off page connector	Penghubungan bagian flowcart yang berada pada halamanberbeda



## 2.10 MySQL

MySQL (baca: mai-se-kyu-el) merupakan *software* yang tergolong sebagai DBMS (*Database Management System*) yang bersifat *Open Source*. Open Source menyatakan bahwa software ini dilengkapi dengan *source code* (kode yang dipakai untuk membuat MySQL), selain tentu saja bentuk *executable*-nya atau kode yang dapat dijalankan secara langsung dalam sistem informasi, dan bisa diperoleh dengan cara *men-download* (mengunduh) secara gratis.

## 2.11 Database MySQL

Database MySQL merupakan sebuah database server SQL multiuser dan multi threaded. MySQL dibuat oleh TcX dan telah dipercaya mengelola sistem dengan 40 buah database berisi 10.000 tabel dan 500 diantaranya memiliki 7 juta baris (kira-kira 100 gigabyte data). Database ini dibuat untuk keperluan sistem database yang cepat, andal dan mudah digunakan. Walaupun memiliki kemampuan yang cukup baik, MySQL untuk sistem operasi Unix bersifat freeware, dan terdapat versi shareware untuk sistem operasi windows.

### 2.11.1 Hierarki dalam Database MySQL

Dalam database atau skema merupakan kumpulan dari :

- Tables
- Views
- Stored Procedure
- Function
- Trigger
- Events

#### 1. Tabel

Sebagaimana basis data sistem yang lain, dalam MySQL juga dikenal hierarki server dengan database. Tiap-tiap database memiliki field-field. Umumnya informasi tersimpan dalam tabel yang secara logik merupakan struktur 2 dimensi terdiri dari baris dan kolom. Field-field (atribut, kolom) tersebut dapat berupa data seperti integer, real, date, time, dan lainnya.

#### 2. View

View adalah tabel yang isi barisnya secara tidak eksplisit tersimpan dalam database tetapi jika diperlukan akan diperhitungkan dari definisi view.

#### 3. Function

Dalam database server ada 2 kategori fungsi : (1) fungsi aggregate adalah fungsi bawaan seperti SUM(), MAX(), MIN(), COUNT() dan lain-lain, yang terdapat pada sistem database tersebut, (2) fungsi yang diciptakan oleh user sendiri. Fungsi dibuat menggunakan rangkaian perintah ditulis dengan bahasa PL/SQL. Ketika diciptakan akan tersimpan secara permanen dalam database server.

#### 4. Stored Procedure

Stored Procedure hampir sama dengan fungsi, tetapi cara memanggilnya menggunakan perintah CALL nama\_prosedure(parameter1,.....)

#### 5. Trigger

Pemicu proses untuk perubahan isi tabel terjadi pada saat pengguna melakukan transaksi. Trigger adalah suatu prosedur berkaitan dengan perintah DML (INSERT,UPDATE,DELETE) pada tabel.

#### 6. Event

MySQL Even Scheduler mengola penjadwalan dan menjalankan perintah tertentu secara terjadwal secara otomatis. Perintah Event hanya terdapat pada MySQL versi 5.1. cara kerja seperti “cron job” pada sistem UNIX/LINUX dan Windows Task Scheduler untuk sistem Windows.