

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Penelitian Sebelumnya**

Penelitian terdahulu ini menjadi salah satu acuan penulis dalam melakukan penelitian sehingga dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Dari penelitian terdahulu, penulis tidak menemukan penelitian dengan judul yang sama seperti judul penelitian penulis. Namun penulis mengangkat beberapa penelitian sebagai referensi dalam memperkaya bahan kajian pada penelitian penulis. Berikut merupakan penelitian terdahulu berupa beberapa jurnal terkait dengan penelitian yang dilakukan penulis.

Faisal rahman, (2013) dengan judul Sistem informasi inventori dengan menggunakan metode average dimana penelitian yang dikembangkan hanya dapat menghasilkan kartu persediaan.

Reza abadi pratama, (2012) dengan judul Aplikasi persediaan barang berbasis web (studi kasus pada koperasi pegawai republik indonesia) dimana penelitian yang dilakukan dapat menghasilkan penjualan, pembelian dan kartu persediaan.

Penelitian Ketiga dikutip dari jurnal ilmiah mahasiswa universitas Surabaya (2013) dimana penelitian ini dilakukan oleh Edo Yosia dengan judul sistem informasi manajemen sediaan restoran top ten group Surabaya. Sistem informasi yang dihasilkan adalah sistem pergudangan sediaan yang terkomputerisasi dan memudahkan dalam sistem operasional badan usaha.

Penelitian Keempat dikutip dari judul Perancangan Sistem Informasi Inventory Spare Part Elektronik Berbasis Web PHP. “Perancangan Sistem Informasi *Inventory Spare Part* Elektronik Berbasis Web PHP” yang dilakukan oleh Mhd Bustanur Rahmat dan Tedy Setiady bertujuan untuk menghasilkan perancangan sistem informasi *inventory spare part* elektronik yang dapat dijadikan sebagai alat bantu informasi dalam peningkatan melakukan pengolahan data barang serta stok gudang dan mencatat penggunaan biaya *inventory*. Melalui

sistem informasi inventory ini, pihak manajemen dapat mengambil keputusan berdasarkan rekapitulasi transaksi, sisa stok dan informasi lain

Penelitian Kelima dikutip dari judul “Analisis dan Perancangan Sistem Informasi Manajemen Gudang pada PT Mitra Pinastika Mulia Surabaya” yang dilakukan oleh Arel Riedsa Adiguna, Mochamad Chandra Saputra dan Fajar Pradana yang bertujuan menjadikan pemodelan sistem manajemen gudang yang baru terhadap spesifikasi kebutuhan sehingga terbentuk perancangan sistem manajemen gudang baru. Perancangan sistem ini mencakup pemodelan dari analisis usecase, analisis aktifitas sistem dengan pengguna dalam *activity diagram*, analisis alur sistem dalam *sequence diagram*. Analisis kelas pada *class diagram*, analisis database dalam CDM serta PDM dan *wireframe* sistem manajemen gudang.

## 2.2 Aplikasi

Menurut Hasan Abdurahman dan Asep Ririh Riswaya (2014), aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut.

Sedangkan menurut *Kamus Besar Bahasa Indonesia*, Aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna.

Jadi, dapat disimpulkan bahwa aplikasi adalah penerapan dari rancangan sistem berupa program siap pakai yang dapat digunakan untuk menjalankan perintah ataupun mengolah data yang bertujuan untuk menjalankan dan melaksanakan tugas khusus dari pengguna (*user*).

### **2.3 Data**

Definisi Data menurut (Sutabri, 2012:2) Data adalah bahan mentah yang diproses untuk menyajikan informasi.

Sedangkan Data menurut (Sutanta, 2011:13) Data dapat didefinisikan sebagai bahan keterangan tentang kejadian-kejadian nyata atau fakta-fakta yang dirumuskan dalam sekelompok lambang tertentu yang tidak acak, yang menunjukkan jumlah, tindakan, atau hal.

Dari berbagai penjelasan di atas dapat disimpulkan bahwa data adalah bahan mentah seperti kejadian atau fakta, nilai, keadaan atau sifat yang belum mempunyai makna atau belum berarti bagi penerima, sehingga perlu diolah atau diproses untuk menyajikan suatu informasi.

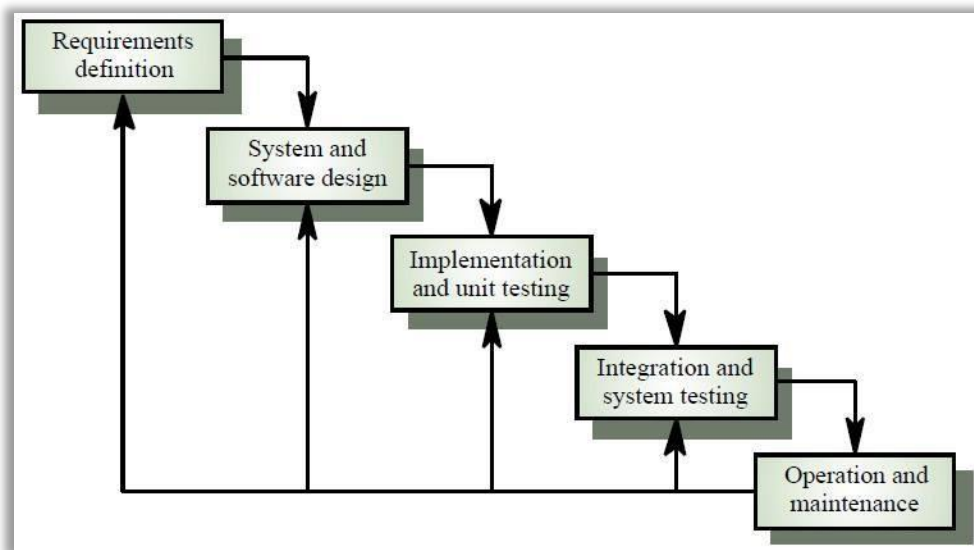
### **2.4 Persediaan(Stok) Barang**

Barang adalah benda dalam berbagai bentuk dan uraian, yang meliputi bahan baku, barang setengah jadi, barang jadi atau peralatan, yang spesifikasinya ditetapkan oleh pengguna barang dan jasa.

Persediaan barang adalah istilah yang digunakan untuk menunjukkan barang yang dimiliki oleh suatu perusahaan atau instansi yang tergantung pada jenis usahanya. Secara umum istilah persediaan barang dipakai untuk menunjukkan barang yang telah dimiliki oleh perusahaan atau toko untuk dijual kembali. Istilah ini digunakan pada perusahaan dagang, jika perusahaan itu bergerak dalam pengelolaan barang atau memproses barang untuk meningkatkan nilai barang jual. Persediaan (Stok) barang berkaitan erat dengan penjualan. Penjualan adalah suatu proses perorangan atau kelompok yang membantu dan meyakinkan calon pembeli untuk membeli barang atau jasa agar mencapai tujuan perdagangan yang penting bagi pihak penjual.

## 2.5 System Development Life Cycle (SDLC)

Menurut Pressman (2001), model *System Development Life Cycle (SDLC)* ini biasa disebut juga dengan model *waterfall* atau disebut juga *classic life cycle*. Adapun pengertian dari SDLC ini adalah suatu pendekatan yang sistematis dan berurutan. Tahapan-tahapannya adalah *Requirements* (analisis sistem), *Analysis* (analisis kebutuhan sistem), *Design* (perancangan), *Coding* (implementasi), *Testing* (pengujian) dan *Maintenance* (perawatan). Adapun tahapan SDLC adalah sebagai berikut :



**Gambar 2.4 System Development Life Cycle (SDLC)**

### 2.5.1 Requirement (Analisis Kebutuhan Sistem)

Pada tahap awal ini dilakukan analisa guna menggali kebutuhan yang dibutuhkan, yaitu kebutuhan data dan kebutuhan *user*. Kebutuhan sebenarnya dibedakan menjadi tiga jenis kebutuhan. Pertama tentang kebutuhan teknologi. Dari hal ini dilakukan analisa mengenai kebutuhan teknologi yang diperlukan dalam pengembangan suatu sistem seperti data penyimpanan informasi atau *database*. Kedua kebutuhan informasi, contohnya seperti informasi mengenai visi dan misi perusahaan, sejarah perusahaan, latar belakang perusahaan. Ketiga yaitu kebutuhan *user*. Dalam hal ini dilakukan analisa terkait kebutuhan *user* dan kategori *user*. Dari analisa yang telah disebutkan di atas, terdapat satu hal yang

tidak kalah penting dalam tahap analisa pada metode SDLC, yaitu analisa biaya dan resiko. Dalam tahap ini diperhitungkan biaya yang akan dikeluarkan seperti biaya implementasi, *testing* dan *Maintenance*.

Analisis sistem pada tingkat teknik pertama, disebut sebagai model analisis yang menggambarkan serangkaian model representasi dari sistem yang akan dibuat (Pressman : 2002).

Model analisis harus mencapai tiga sasaran utama : (1) untuk menggambarkan apa yang dibutuhkan, (2) untuk membangun dasar bagi pembuatan desain perangkat lunak, (3) untuk membatasi serangkaian persyaratan yang dapat divalidasi begitu perangkat lunak dibangun.

Untuk mencapai sasaran tersebut, alat-alat bantu yang dapat digunakan untuk mempermudah dalam merancang dan membangun suatu perangkat lunak, yaitu :

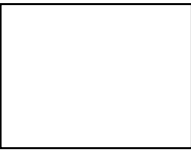

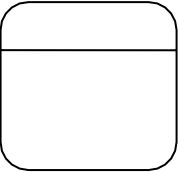

**a.** *Contex Diagram* (CD)

*Contex Diagram* atau diagram konteks merupakan sebuah diagram aliran data yang memfokuskan pada aliran data dari dan ke dalam sistem, serta memproses data-data tersebut. Komponen – komponen dasar dari setiap program komputer yang digambarkan secara mendetail, dapat digunakan untuk menganalisis keakuratan dan kompetensi sistem (Kendall : 2003).

**b.** *Data Flow Diagram* (DFD)

*Data Flow Diagram* (DFD) merupakan gambaran alur data atau informasi tanpa mengaitkan bentuk fisik media penyimpanan data atau *hardware* (Kendall : 2003). Beberapa simbol digunakan dalam DFD dapat dilihat Tabel 1.



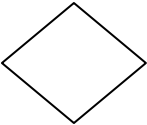
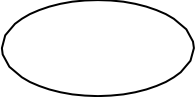
**Tabel 2.1 Simbol Data Flow Diagram (Kendall, 2002:40)**

Simbol	Arti	Keterangan
	Terminator	Menunjukkan kesatuan luar atau <i>eksternal</i> (dapat berupa kelompok orang atau departemen atau sistem) yang bisa menerima informasi
	Aliran Data	Menunjukkan aliran atau arus data dimana informasi sedang melintas dan atau menuju ke suatu proses.
	Proses	Menunjukkan proses dimana beberapa tindakan atau sekelompok tindakan dijalankan
	Data Store	Menunjukkan penyimpanan data atau <i>Database</i>

**c. Entity Relationship Diagram (ERD)**

*Entity Relationship Diagram* merupakan diagram yang berisi komponen- komponen himpunan entitas dan himpunan relasi yang masing- masing dilengkapi dengan atribut- atribut yang merepresentasikan seluruh fakta yang ditinjau (Fatansyah, 1999 :70). Tabel 2 merupakan simbol- simbol yang digunakan dalam pembuatan ERD.

**Tabel 2.2 Simbol dalam ERD**

Simbol	Arti	Keterangan
	Entitas	Menunjukkan himpunan entitas yang merupakan suatu obyek yang dapat diidentifikasi dalam lingkungan pemakai
	Aliran Data	Menunjukkan aliran atau arus data dimana informasi sedang melintas dan atau menuju ke suatu proses.
	Proses	Menunjukkan proses dimana beberapa tindakan atau sekelompok tindakan dijalankan
	Atribut	Menunjukkan atribut yang berfungsi mendeskripsikan karakter entitas.

Kerelasiaan menyatakan hubungan antar relasi dalam basis data. Kerelasiaan antar relasi dituliskan oleh *foreign key* atau relasi-relasi bertipe transaksi yang digunakan dalam basis data. Jenis-jenis kerelasiaan antar relasi, meliputi :

1. Kerelasiaan Satu ke Satu (*One to One Relationship*)

Kerelasiaan satu ke satu terjadi jika setiap nilai pada suatu relasi hanya mengimplikasikan sebuah nilai pada relasi yang direlasikan secara logik.

2. Kerelasiaan Satu ke Banyak (*One to Many Relationship*)

Kerelasian satu ke banyak terjadi jika setiap nilai pada suatu relasi mengimplikasikan banyak nilai pada relasi lain yang direlasikan secara logik.

3. Kerelasian Banyak ke Banyak (Many to Many Relationship)

Kerelasian banyak ke banyak terjadi jika banyak nilai pada suatu relasi mengimplikasikan banyak nilai pada relasi lain yang direlasikan secara logik.

**d. Skema Diagram /Skema Diagram**

Skema diagram merupakan diagram yang menggambarkan beberapa aspek dari suatu skema basis data (Murni :2006).

**2.5.2 Perancangan (Design)**

Fase perancangan merupakan inti teknis dari rekayasa perangkat lunak. Selama desain, penyaringan yang *progresif* dari struktur data, arsitektur data, arsitektur program, *interface*, dan detail prosedural dikembangkan, dikaji, dan didokumentasikan. Desain menghasilkan representasi perangkat lunak yang dapat ditaksir kualitasnya(Presman : 2002).

**a. Process Specification (PSPEC)**

*Process Specification* digunakan untuk menggambarkan semua proses model aliran yang nampak pada tingkat akhir penyaringan. Kandungan dari PSPEC dapat termasuk teks naratif, *Program Design Language* (PDL), tabel, diagram, atau bagan (Pressman, 2002 : 386).

**b. Deskripsi Data**

Deskripsi data merupakan deskripsi isi dari kamus data, yang merepresentasikan data komposit. Deskripsi ini memerlukan penyaringan lebih jauh dalam kamus data, sampai semua item data direpresentasikan sebagai item *elementary* atau sampai semua objek data direpresentasikan dalam bentuk yang tidak ambigu bagi pembaca (Pressman, 2002 : 389-390). Sebagai contoh diberikan study kasus mengenai deskripsi data dari tabel request barang pada tabel 2.4.3 dibawah ini:



**Tabel 2.5.2 Tabel Request Barang**

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Ukuran Field</b>	<b>Deskripsi</b>	<b>Null</b>	<b>Primary Key</b>
idrequestbarang	<i>int</i>	11	identitas request barang	Not null	Key
tanggalpermintaan	<i>Date</i>	-	Tanggal Permintaan barang	Not null	
namabidang	<i>varchar</i>	200	Nama Bidang	Not null	
jenisbarang	<i>varchar</i>	100	Jenis barang yang diminta	Not null	
namabarang	<i>Varchar</i>	200	nama barang yang diminta	Not null	
jumlahbarang	<i>int</i>	11	Jumlah barang	Not null	
foto	<i>varchar</i>	200	Foto barang	Not null	

### 2.5.3 Implementation (Coding)

Rancangan yang telah dibuat dalam tahap sebelumnya akan diterjemahkan dalam bentuk bahasa yang dapat dibaca oleh komputer untuk diolah. Tahap ini juga dapat disebut dengan tahap implementasi, yaitu tahap yang mengkonversi hasil perancangan sebelumnya ke dalam sebuah bahasa pemrograman yang dimengerti oleh komputer. Kemudian komputer akan menjalankan fungsi- fungsi yang telah didefinisikan sehingga mampu memberikan layanan-layanan kepada penggunanya.

#### **2.5.4 Testing (Pengujian)**

Pengujian program dilakukan untuk mengetahui kesesuaian sistem berjalan sesuai prosedur atau tidak dan memastikan sistem terhindar dari *error* yang terjadi. *Testing* juga dapat digunakan untuk memastikan ke-valid-an dalam proses *input*, sehingga dapat menghasilkan *output* yang sesuai. Pada tahap ini terdapat dua metode pengujian, yaitu metode *black-box* dan *white-box*. Pengujian dengan metode *black-box* menekankan pada fungsionalitas dari sebuah perangkat lunak tanpa harus mengetahui bagaimana struktur di dalam perangkat lunak tersebut.

#### **2.5.5 Maintenance (Perawatan)**

Tahap terakhir dari metode SDLC ini adalah *Maintenance*. Pada tahap ini jika sistem sesuai dengan tujuan yang ditentukan dan dapat menyelesaikan masalah pada perusahaan, maka akan diberikan kepada pengguna. Setelah digunakan dalam periode tertentu pasti terdapat penyesuaian atau perubahan sesuai dengan keadaan yang diinginkan, sehingga membutuhkan perubahan terhadap sistem tersebut. Tahap ini dapat pula diartikan sebagai tahap penggunaan perangkat lunak yang disertai dengan perawatan dan perbaikan. Perawatan dan perbaikan suatu perangkat lunak diperlukan, termasuk di dalamnya adalah pengembangan, karena dalam prakteknya ketika perangkat lunak digunakan terkadang masih terdapat kekurangan atau penambahan fitur- fitur baru yang dirasa perlu.



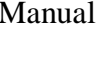




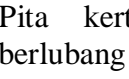

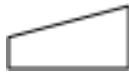


### **2.6 Flowchart**

Menurut I Gusti Ngurah Suryantara (2009), badan alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi.

Tujuan utama penggunaan *flowchart* adalah untuk menggambarkan suatu tahap penyelesaian masalah secara sederhana, terurai, rapi, dan jelas dengan

menggunakan simbol-simbol yang standar. Dalam penulisan *flowchart* dikenal dua model yaitu *flowchart* sistem dan *flowchart* program. *Flowchart* sistem merupakan diagram alir yang menggambarkan suatu sistem peralatan komputer yang digunakan dalam proses pengolahan data serta hubungan antara peralatan tersebut. *Flowchart* program merupakan diagram alir yang menggambarkan suatu logika dari suatu prosedur pemecahan masalah. Berikut merupakan beberapa simbol pada *flowchart*.

**Tabel 2.6 Simbol Flowchart**

No	Simbol	Keterangan	No	Simbol	Keterangan
1.	Dokumen 	Menunjukkan dokumen <i>input</i> dan <i>output</i> baik proses manual, mekanil atau <i>computer</i>	11.	Hard disk 	Menunjukkan <i>input/output</i> menggunakan hard disk
2.	Kegiatan Manual 	Menunjukkan pekerjaan manual	12.	Diskette 	Menunjukkan <i>input/output</i> menggunakan diskette
3.	Simpanan Offline 	File non-komputer yang diarsip urut angka ( <i>numerical</i> )	13.	Drum magnetik 	Menunjukkan <i>input/output</i> menggunakan Drum magnetik
4.	Simpanan Offline 	File non-komputer yang diarsip urut huruf ( <i>alphabetical</i> )	14.	Pita kertas berlubang 	Menunjukkan <i>input/output</i> menggunakan Pita kertas berlubang
5.	Simpanan Offline 	File non-komputer yang diarsip urut tanggal ( <i>cronological</i> )	15.	Keyboard 	Menunjukkan <i>input/output</i> menggunakan on-line keyboard
6.	Kartu plong 	Menunjukkan <i>input/output</i> yang menggunakan kartu plong	16.	Display 	Menunjukkan output yang tampil di komputer

## 2.7 Basis Data (Database)

Menurut Bhojaraju G. dan M.M.Koganurmath (2003:3), Basis data adalah kumpulan informasi terorganisir yang disimpan dengan redundan yang minimum. dengan tujuan untuk memudahkan akses bagi semua aplikasi. Basis data juga bisa diartikan sebagai sekumpulan data atau informasi yang memiliki relasi, artinya data tersebut bisa saling berhubungan satu sama lain. Setiap basis data juga memiliki sejumlah objek pokok basis data seperti tabel, indeks dan lain sebagainya. Adapun contoh dari basis data yang sering dipakai oleh pengguna ialah *Mysql*, *Oracle*, dan lain sebagainya.

### **DBMS (*Database Management System*)**

Menurut Fatansyah (1999) DBMS (*Database Management System*) adalah suatu perangkat khusus yang menangani basis data, seperti menentukan bagaimana data diorganisasi, disimpan, diubah, diambil kembali, DBMS juga menerapkan mekanisme pengamanan data, pemakaian secara bersama, pemaksaan keakuratan atau konsistensi data dan sebagainya.

Perangkat lunak termasuk DBMS seperti *dbase III +*, *dbase IV*, *FoxBase*, *Rbase*, *MS.Access*, *Borland-Paradox*, *MS-SQLServer*, *MySQL*, *Oracle*, *Informix*, *Sybase*, dan sebagainya. Pada penelitian ini penulis menggunakan dua perangkat lunak khusus yaitu untuk pembuatan aplikasi stok barang digunakan bahasa pemrograman *PHP*, sedangkan untuk pengelolaan basis data digunakan *MySQL*.

### ***Structured Query Language (SQL)***

*SQL (Strutured Query Language)* merupakan bahasa ANSI (American National Standart Input) yang digunakan untuk melakukan *query* data pada database. Semua pengoperasian data dapat dikerjakan secara mudah dengan menggunakan bahasa ini, terutama dalam pemasukan dan seleksi data.

Hampir semua *software* database mengimplementasikan bahasa ini sebagai komponen utama dari produknya. Contohnya adalah *MySQL*, *PostrgresSQL*, *Oracle*, *Infomix*, *Sybase* dan beberapa *database* lain yang memiliki konsep *database* modern.

Bahkan Microsoft Access yang merupakan *database stand alone* juga mendukung bahasa SQL (Prasetyo :2003).

SQL berisi *syntax* atau pernyataan yang dapat digunakan untuk memasukan, merubah dan menghapus, memilih dan melindungi data. Berikut daftar sejumlah pernyataan SQL yang mengacu pada manual MySQL dalam situs resminya [www.mysql.com](http://www.mysql.com) (MySQL AB:1997-2000).

a. CREATE DATABASE [IF NOT EXISTS] db\_name

Keterangan : *syntax* diatas adalah perintah untuk menciptakan database dan memberikan nama database. Pesan *error* akan muncul apabila database yang diciptakan telah ada atau terdapat database dengan nama yang sama.

b. DROP DATABASE [IF EXISTS] db\_name

Keterangan : *syntax* diatas adalah perintah untuk menghapus database dan seluruh tabel yang ada didatabase tersebut. Sebaiknya berhati-hatilah dengan *syntax* ini.

c. CREATE [TEMPORARY] TABLE [IF NOT EXISTS] tbl\_name [(create\_definition,...)][table\_options][select\_statement]

Keterangan : *syntac* diatas adalah perintah untuk menciptakan tabel dengan memberi nama tabel pada database yang telah ditentukan.

d. ALTER[IGNORE] TABLE tbl\_name alter\_spec [, alter\_spec...]

Keterangan : *syntac* diatas adalah perintah untuk megubah struktur tabel yang ada. Sebagai contoh kita dapat menambah atau menghapus kolom, *indeks*, mengubah nama kolom, dan lain sebagainya.

e. DELETE [LOW\_PRIORITY] FROM tbl\_name [WHERE where\_definition][LIMIT rows]

Keterangan : *syntax* diatas adalah perintah untuk menghapus baris dari *tbl\_name* yang memenuhi keadaan dengan pemberian *where\_definition*, dan *returns the number of record deleted*.

f. SELECT [STRAIGHT\_JOIN][SQL\_SMALL\_RESULT]  
[SQL\_BIG\_RESULT][SQL\_BUFFER\_RESULT]  
[HIGHT\_PRIORITY][DISTINCT | DISTINCTROW | ALL]

Keterangan : *syntax* diatas adalah perintah untuk menampilkan data melalui pemilihan baris dan kolom dari satu atau banyak tabel.

g. INSERT [LOW\_PRIORITY | DELAYED][IGNORE][INTO]tbl\_name  
[(col\_name,...)] VALUES (expression,...)

Keterangan : *syntax* diatas adalah perintah untuk memasukkan atau menyisipkan baris baru ke dalam tabel.

h. GRANT priv\_type [(column\_list)][,priv\_type[(column\_list)]...] ON  
{tbl\_name | \*.\*|db\_name.\*} TO user\_name [IDENTIFIED BY  
'password']...][WITH GRANT OPTION]

Keterangan : *syntax* diatas adalah perintah untuk memberikan hak ijin akses bagi *user* agar dapat mengakses *database*, tabel dan kolom. Selain itu, kita juga dapat menambah *user* baru dengan perintah GRANT ini. Perintah GRANT tersedia pada MySQL sejak versi

3.22.11 dan memiliki empat tingkat pilihan yang dapat dipergunakan, yaitu :

- *Global level / Ijin Akses Penuh*

Global level digunakan untuk mengakses seluruh *database* yang berada di *server*, konfigurasi akses ini terletak pada tabel *user*.

- *Database level / Ijin Akses Database*

Database level digunakan untuk mengakses tabel-tabel yang berada pada database yang telah ditentukan, konfigurasi akses terletak tabel *db* dan tabel *host*.

- *Tabel Level / Ijin Akses Tabel*

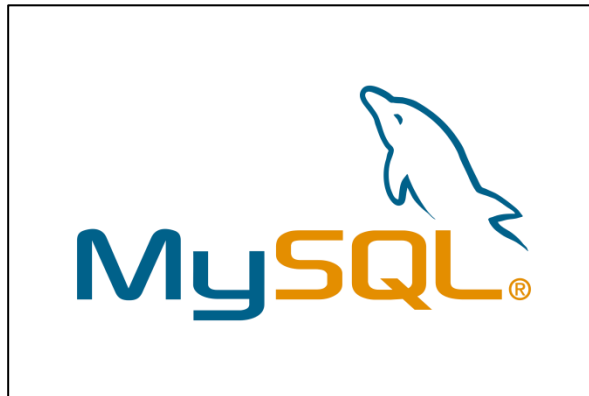
Tabel level digunakan untuk mengatur ijin akses terhadap semua kolom yang terdapat pada tabel yang ditentukan. Konfigurasi akses ini terletak pada *tables\_priv*.

- *Column Level* / Ijin Akses Kolom

Column level ini digunakan untuk mengatur ijin akses pada kolom yang ditentukan saja, konfigurasi akses ini terletak pada *columns\_priv*.

## 2.8 MySql

Menurut Achmad Solichin (2010:8), MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang bersifat *multithread* dan *multi-user* dengan sekitar 6 juta instalasi di seluruh dunia. Tidak seperti PHP atau Apache yang merupakan software yang dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing-masing, MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia yaitu MySQL AB. MySQL AB memegang penuh hak cipta hampir atas semua kode sumbernya.



Gambar 2.8 Logo MySql

## 2.9 Bahasa Pemrograman

### 2.9.1 PHP

PHP atau *Hypertext Preprocessor* adalah bahasa pemrograman *script server-side* yang didesain untuk pengembangan web dan aplikasi. Disebut *server-side* karena PHP diproses pada komputer *server*. Hal ini berbeda dibandingkan dengan bahasa pemrograman *client-side* seperti Javascript yang diproses pada web browser (*client*).

Bahasa pemrograman PHP dapat digunakan secara gratis dan bersifat *open source* atau dapat digunakan siapa saja. PHP dirilis dalam lisensi *PHP License*, sedikit berbeda dengan lisensi *General Public License* (GPL) yang biasa digunakan untuk proyek *open source*. Kemudahan dan kepopuleran PHP sudah menjadi standar bagi para programmer di seluruh dunia.

Adapun kelebihan dari bahasa pemrograman PHP ini adalah sebagai berikut :

1. Tidak memerlukan kompilasi (*compile*) dalam penggunaannya.
2. Program yang dibuat dengan PHP bisa dijalankan oleh semua Sistem Operasi karena PHP berjalan secara Web Base yang artinya semua Sistem Operasi bahkan perangkat *mobile* yang mempunyai web browser dapat menggunakan program PHP.
3. Mendukung banyak paket *Database* seperti MySQL, Oracle, PostgreSQL dan masih banyak lagi.
4. Banyak *Web Server* yang mendukung PHP seperti Apache, Lighttpd dan lainnya.
5. Pengembangan Aplikasi PHP terbilang mudah karena banyak dokumentasi, referensi, dan *Developer* yang membantu dalam pengembangannya.





**Gambar 2.9.1 Logo PHP**

### **2.9.2 HTML**

HTML (HyperText Markup Language) merupakan sebuah bahasa markup (tanda) yang digunakan untuk membuat sebuah "halaman web" dan menampilkan berbagai informasi didalam sebuah browser internet yang merupakan standart internet yang saat ini dikendalikan oleh World Wide Web Consortium (W3C). Bermula dari sebuah bahasa yang sebelumnya banyak digunakan didunia penerbitan dan pencetakan yang disebut dengan SGML. HTML berupa kode-kode tag yang mengintruksikan browser untuk menghasilkan tampilan sesuai dengan yang diinginkan. Sebuah file yang merupakan file HTML dapat dibuka dengan menggunakan browser webseperti Mozilla Firefox atau Microsoft Internet Explorer. HTML juga dapat dikenali oleh aplikasi pembuka email ataupun dari PDA dan program lain yang memiliki kemampuan browser (Rachdin: 2006).

### **2.9.3 Visual Studio**

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi console, aplikasi Windows, ataupun aplikasi Web. Visual Studio mencakup kompiler, SDK, *Integrated Development Environment* (IDE), dan dokumentasi (umumnya berupa *MSDN Library*). Kompiler yang dimasukkan ke dalam paket Visual Studio

antara lain *Visual C++*, *Visual C#*, *Visual Basic*, *Visual Basic .NET*, *Visual InterDev*, *Visual J++*, *Visual J#*, *Visual FoxPro*, dan *Visual SourceSafe*.

Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam native code (dalam bentuk bahasa mesin yang berjalan di atas Windows) ataupun managed code (dalam bentuk *Microsoft Intermediate Language* di atas *.NET Framework*). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi Silverlight, aplikasi Windows Mobile (yang berjalan di atas *.NET Compact Framework*).



**Gambar 2.9.3 Logo Visual Studio**