

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Teori – Teori Dasar**

Berikut ini merupakan teori – teori dasar yang berkaitan dengan penelitian antara lain :

##### **2.1.1 Teori Dasar Analisis dan Perancangan Sistem**

Penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikannya. (Jogiyanto, 2005)

##### **2.1.2 Konsep Dasar Manajemen Persediaan**

Freddy Rangkuti (2006) berpendapat bahwa “persediaan merupakan sejumlah bahan-bahan, bagian-bagian yang disediakan dan bahan-bahan dalam proses yang terdapat dalam perusahaan untuk proses produksi, serta barang-barang jadi/produk yang disediakan untuk memenuhi permintaan dari konsumen atau langganan setiap waktu”. Sedangkan pengelolaan stok barang atau Manajemen Inventory (inventory control) adalah kegiatan yang berhubungan dengan perencanaan, pelaksanaan, dan pengawasan penentuan kebutuhan material sedemikian rupa sehingga di satu pihak kebutuhan operasi dapat dipenuhi pada waktunya dan di lain pihak investasi persediaan material dapat ditekan secara optimal. Menurut Freddy Rangkuti (2006) persediaan yang diadakan, antara lain berguna untuk :

- a. menghilangkan risiko keterlambatan datangnya barang,
- b. menghilangkan risiko barang yang rusak,
- c. mempertahankan stabilitas operasi perusahaan,
- d. mencapai penggunaan mesin yang optimal,
- e. memberi pelayanan yang sebaik-baiknya bagi konsumen.

Salah satu fungsi manajerial yang sangat penting adalah pengendalian persediaan. Apabila perusahaan menanamkan terlalu banyak dananya dalam persediaan, hal ini akan menyebabkan biaya penyimpanan yang berlebihan, dan mungkin mempunyai opportunity cost . Demikian pula apabila perusahaan tidak mempunyai persediaan yang mencukupi, dapat mengakibatkan biaya-biaya dari terjadinya kekurangan bahan (stockout cost). Selain itu Indrajit Djokopranoto (2005) menyebutkan bahwa dalam melakukan manajemen persediaan ada sejenis prinsip yang harus dianut yakni “penentuan jumlah dan jenis barang yang disimpan dalam persediaan haruslah sedemikian rupa sehingga produksi dan operasi perusahaan tidak terganggu, tetapi di lain pihak sekaligus harus dijaga agar biaya investasi yang timbul dari penyediaan barang tersebut seminimal mungkin”. Prinsip ini menandakan bahwa Manajemen Inventory haruslah berdaya guna (efisien) dan berhasil guna (efektif). Menjamin kelangsungan jalannya operasi perusahaan adalah soal efektivitas, sedangkan menekan persediaan sampai ke tingkat minimum adalah soal efisiensi.

Selain itu, sistem persediaan yang baik juga dilakukan dalam Manajemen Inventory. Freddy Rangkuti (2006) berpendapat bahwa sistem persediaan diartikan sebagai serangkaian kebijakan dan pengendalian yang memonitor tingkat persediaan dan menentukan tingkat persediaan yang harus dijaga, kapan persediaan harus disediakan dan berapa besar pesanan yang harus dilakukan. Sistem ini bertujuan menetapkan dan menjamin tersedianya sumber daya yang tepat, dalam kuantitas yang tepat dan pada waktu yang tepat. Pada dasarnya Manajemen Inventory akan mempermudah jalannya operasi perusahaan yang harus dilakukan secara berturut-turut untuk memproduksi barang-barang, selanjutnya menyampaikan kepada langganan atau konsumen.

### **2.1.3 Teori Dasar Sistem Informasi**

Sistem adalah suatu jaringan kinerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu. (Jogiyanto HM, 2005) Pendekatan sistem yang lebih menekankan pada elemen atau komponen.

Mendefinisikan sistem sebagai kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. (Jogiyanto HM, 2005)

Sistem mempunyai karakteristik atau sifat-sifat tertentu, diantaranya adalah sebagai berikut :

a. Komponen Sistem (Components)

Sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem.

b. Batas Sistem (Boundary)

Batas sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas sistem menunjukkan ruang lingkup (scope) dari sistem tersebut.

c. Lingkungan Luar Sistem (Environments)

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga merugikan suatu sistem.

d. Penghubung Sistem (Interface)

Penghubung merupakan media penghubung antara suatu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari suatu sub sistem ke subsistem lainnya.

e. Masukan Sistem (Input)

Masukan adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (maintenance input) dan masukan sinyal (signal input).

f. Keluaran sistem (Output)

Keluaran adalah hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat berupa masukan untuk subsistem yang lain atau kepada supra sistem.

g. Pengolah Sistem (Process)

Sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolah. Pengolah yang akan merubah masukan menjadi keluaran.

#### h. Sasaran Sistem (Objectives)

Sistem mempunyai tujuan atau sasaran yang akan menentukan masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem.

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. (Jogiyanto HM, 2005).

Informasi merupakan data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau mendatang. (Hanif Al Fattah, 2007).

Kualitas dari suatu informasi tergantung pada tiga hal, yaitu :

##### a. Akurat

Informasi harus bebas dari kesalahan-kesalahan dan harus jelas mencerminkan maksudnya.

##### b. Tepat Waktu

Informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan di dalam mengambil keputusan. Bila pengambilan keputusan terlambat, maka berakibat fatal bagi organisasi.

##### c. Relevan

Informasi tersebut mempunyai manfaat bagi pemakainya. Relevansi informasi untuk tiap-tiap orang antara yang satu dengan yang lainnya berbeda-beda.

Sistem informasi dapat didefinisikan sebagai “suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan- laporan yang di perlukan.” (Jogiyanto, 2005)

Secara garis besar sistem informasi dapat didefinisikan sebagai suatu sistem di dalam suatu organisasi yang merupakan kombinasi dari orang – orang, fasilitas, teknologi, media, prosedur - prosedur dan pengendalian yang di tujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian –

kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan yang cerik. Dari kutipan tersebut, dapat diketahui bahwa sistem informasi adalah sekumpulan komponen dari informasi yang saling terintegrasi untuk mencapai tujuan yang spesifik. Komponen yang dimaksud adalah komponen *input*, model, *output*, teknologi, basis data (*database*), kontrol atau komponen pengendali. (Jogiyanto HM, 2005)

## **2.2 Metodologi Pengembangan Sistem**

Metodologi pengembangan sistem adalah metode-metode, prosedur-prosedur, konsep-konsep, pekerjaan, aturan-aturan yang akan digunakan untuk mengembangkan suatu sistem informasi (Jogiyanto, 2005)

Di dalam penulisan skripsi ini menggunakan metodologi analisis dan desain sistem terstruktur SSAD (*Structured Systems Analysis and Design*) untuk digunakan pada pengembangan sistem. Metodologi ini dapat digunakan pada tahap analisis dan tahap desain dan metodologi ini menggunakan pendekatan pengembangan sistem terstruktur yang dilengkapi dengan alat-alat (*tools*) dan teknik-teknik (*techniques*) yang dibutuhkan dalam pengembangan sistem, sehingga hasil akhir dari sistem yang dikembangkan akan didapatkan sistem yang strukturnya didefinisikan dengan baik dan jelas.

Metodologi ini secara umum didasarkan pada pemecahan dari sistem ke dalam modul-modul berdasarkan dari tipe elemen data dan tingkah laku logika modul tersebut di dalam sistem. Dengan metodologi ini, sistem secara logika dapat digambarkan secara logika dari arus data dan hubungan antar fungsinya di dalam modul-modul sistem (Jogiyanto, 2005).

### **2.2.1 Perlunya Pengembangan Sistem**

Pengembangan sistem dapat berarti membuat suatu sistem yang baru atau memperbaiki sistem yang telah ada. (Jogiyanto HM, 2005). Sistem yang lama diperbaiki atau diganti disebabkan oleh beberapa hal berikut ini.

1. Adanya permasalahan pada sistem yang lama seperti keluhan dari pelanggan, file – file yang tidak teratur, serta tanggapan yang lambat terhadap pelanggan.

2. Untuk meraih kesempatan – kesempatan artinya teknologi yang semakin berkembang menyebabkan perusahaan ingin merasakan bahwa teknologi informasi perlu digunakan untuk meningkatkan kinerja para pegawai dan dalam pemrosesan data sehingga perusahaan tersebut dapat memperoleh kesempatan yang berupa peluang – peluang pasar, pelayanan yang meningkat kepada pelanggan dan lain sebagainya.
3. Adanya instruksi – instruksi maksudnya pembuatan sistem baru atau memperbaiki sistem yang lama dikarenakan adanya instruksi dari pimpinan atau luar organisasi seperti peraturan pemerintah.

### **2.2.2 Prinsip Pengembangan Sistem**

Beberapa prinsip dalam proses pengembangan system adalah sebagai berikut.

- a. Sistem yang dikembangkan adalah untuk manajemen.
- b. Sistem yang dikembangkan adalah investasi modal yang besar.
- c. Sistem yang dikembangkan memerlukan orang terdidik.
- d. Tahapan kerja dan tugas-tugas yang harus dilakukan dalam proses pengembangan sistem.
- e. Proses pengembangan sistem tidak harus urut.
- f. Jangan takut membatalkan proyek.

### **2.2.3 Siklus Hidup Pengembangan Sistem**

Pengembangan sistem informasi yang berbasis komputer dapat merupakan tugas kompleks yang membutuhkan banyak sumber daya dan dapat memakan waktu berbulan-bulan bahkan bertahun-tahun untuk menyelesaikannya. Proses pengembangan sistem melewati beberapa tahapan dari mulai sistem itu direncanakan sampai dengan sistem tersebut diterapkan, dioperasikan dan dipelihara. Bila operasi sistem yang sudah dikembangkan masih timbul kembali permasalahan, maka perlu dikembangkan kembali suatu sistem untuk mengatasinya dan proses ini kembali ke tahap yang pertama. Siklus ini disebut dengan siklus hidup pengembangan sistem (*system life cycle*). (Jogiyanto, 2005). Langkah – langkah utama dalam siklus hidup pengembangan sistem adalah sebagai berikut :

a. Kebijakan dan Perencanaan Sistem (System Policy and Planning)

Sebelum suatu sistem informasi dikembangkan, umumnya terlebih dahulu dimulai dengan adanya suatu kebijakan dan perencanaan untuk mengembangkan sistem itu. Tanpa adanya perencanaan sistem yang baik, pengembangan sistem tidak akan dapat berjalan sesuai dengan yang diharapkan. Tanpa adanya kebijakan pengembangan sistem oleh manajemen puncak (top management), maka pengembangan sistem tidak akan mendapat dukungan dari manajemen puncak ini. Padahal dukungan dari manajemen puncak sangat penting artinya. Kebijakan sistem (systems policy) merupakan landasan dan dukungan dari manajemen puncak untuk membuat perencanaan sistem. Perencanaan sistem (systems planning) merupakan pedoman untuk melakukan pengembangan sistem. Kebijakan untuk mengembangkan sistem informasi dilakukan oleh manajemen puncak karena manajemen menginginkan untuk meraih kesempatan-kesempatan yang ada yang tidak dapat diraih oleh sistem yang lama atau sistem yang lama mempunyai banyak kelemahan-kelemahan yang perlu diperbaiki (misalnya untuk meningkatkan efektifitas manajemen, meningkatkan produktivitas atau meningkatkan pelayanan yang lebih baik kepada langganan).

b. Analisis Sistem (System Analysis)

Penelitian atas sistem yang telah ada dengan tujuan untuk merancang sistem yang baru atau diperbarui.

c. Desain Sistem secara Umum (General System Design)

Tujuan dari desain sistem secara umum adalah untuk memberikan gambaran secara umum kepada user tentang sistem yang baru.

d. Seleksi Sistem (System Evaluation)

Hasil desain sistem secara umum tentunya harus menjadi pertimbangan pihak manajemen apakah melanjutkan pengembangan sistem yang baru berdasarkan gambaran desain sistem secara umum atau menolak rancangan baru tersebut.

e. Desain Sistem Terinci (Detailed System Design)

Dengan memahami sistem yang ada dan persyaratan-persyaratan sistem baru, selanjutnya adalah penentuan proses dan data yang diperlukan oleh sistem

baru. Jika sistem itu berbasis komputer, rancangan harus menyertakan spesifikasi jenis peralatan yang akan digunakan.




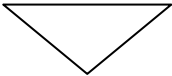
f. Implementasi Sistem (System Implementation)

Merupakan kegiatan memperoleh dan mengintegrasikan sumber daya fisik dan konseptual yang menghasilkan suatu sistem yang bekerja.

#### 2.2.4 Alat Pengembangan Sistem




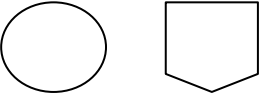
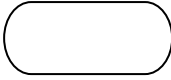
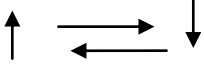
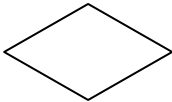
Agar dapat melakukan langkah-langkah sesuai dengan metodologi pengembangan analisis dan desain sistem terstruktur, maka dibutuhkan alat dan teknik untuk melaksanakannya. Adapun alat pengembangan sistem yang digunakan yaitu bagan alir dokumen. Bagan alir dokumen (*Document Flowchart*) yaitu bagan alir yang menunjukkan arus dari laporan formulir termasuk tembusan-tembusannya (Jogiyanto, 2005). Bagan alir dokumen ini menggunakan simbol-simbol seperti pada Tabel 2.1 berikut.

Tabel 2.1 Bagan Alir Dokumen

Simbol	Keterangan
<p><b>Dokumen</b></p> 	Menunjukkan dokumen yang digunakan untuk <i>input</i> dan <i>output</i> baik secara manual maupun komputerisasi.
<p><b>Proses manual</b></p> 	Menunjukkan pekerjaan yang dilakukan secara manual.
<p><b>Proses komputerisasi</b></p> 	Menunjukkan proses dari operasi program komputer.
<p><b>Simpanan</b></p> 	Menunjukkan arsip.




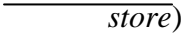


Tabel 2.1 Bagan Alir Dokumen (lanjutan)

Simbol	Keterangan
<p data-bbox="516 360 646 394"><i>Keyboard</i></p> 	<p data-bbox="894 360 1360 465">Menunjukkan input yang menggunakan <i>keyboard</i>.</p>
<p data-bbox="492 584 621 618"><i>Hard disk</i></p> 	<p data-bbox="894 584 1360 689">Media penyimpanan, menggunakan perangkat <i>hard disk</i>.</p>
<p data-bbox="500 763 662 797"><b>Keterangan</b></p> 	<p data-bbox="894 763 1360 869">Digunakan untuk memberikan keterangan yang lainnya.</p>
<p data-bbox="492 949 670 983"><b>Penghubung</b></p> 	<p data-bbox="894 949 1360 1272">Simbol yang digunakan untuk menunjukkan sambungan dari bagan alir yang terputus di halaman yang sama maupun di halaman yang lain.</p>
<p data-bbox="475 1319 638 1352"><b>Terminator</b></p> 	<p data-bbox="894 1319 1360 1424">Digunakan untuk memberikan awal dan akhir suatu proses.</p>
<p data-bbox="467 1476 605 1509"><b>Garis alir</b></p> 	<p data-bbox="894 1476 1360 1581">Digunakan untuk menunjukkan arus dari proses.</p>
<p data-bbox="500 1626 621 1659"><i>Decision</i></p> 	<p data-bbox="894 1626 1360 1805">Digunakan untuk suatu penyeleksian kondisi di dalam program.</p>

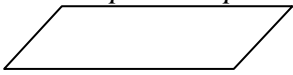
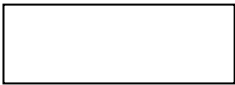
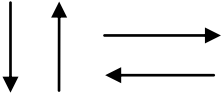

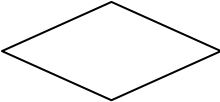

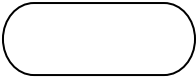
*Data flow diagram* (DFD) merupakan salah satu alat pengembangan sistem yang juga digunakan dalam metode ini. *Data flow diagram* menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan (Jogiyanto, 2005). Simbol yang digunakan dalam DFD adalah seperti pada Tabel 2.2 berikut.

Tabel 2.2 Simbol *Data Flow Diagram*

Simbol	Keterangan
<p><i>External Entity</i></p> 	Merupakan sumber atau tujuan dari aliran data dari atau ke sistem.
<p>Arus data (<i>data flow</i>)</p> 	Menggambarkan aliran data .
<p>Proses (<i>process</i>)</p> 	Proses atau fungsi yang mentransformasikan data masukan menjadi keluaran.
<p>Simpanan data (<i>data store</i>)</p> 	Komponen yang berfungsi untuk menyimpan data atau <i>file</i> .

Selain bagan alir dokumen dan *data flow diagram*, bagan alir program (*program flowchart*) merupakan alat yang penting bagi pengguna atau *user* untuk dapat memahami logika program secara terperinci. Bagan alir program ini dibuat dari derivikasi bagan alir sistem. Bagan alir program (*program flowchart*) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program (Jogiyanto, 2005). Bagan alir logika program digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika. Untuk simbol-simbol *program flowchart* hampir sama dengan bagan alir dokumen, hanya terdapat perbedaan pada simbol *process*, penyimpanan dan *input-output* data. Adapun simbol-simbol *program flowchart* sebagai berikut :

Tabel 2.3 Simbol Bagan Alir Program

Simbol	Keterangan
<p><i>Input / Output</i></p> 	Simbol <i>input/output</i> digunakan untuk mewakili data <i>input/output</i>
<p>Proses</p> 	Simbol proses ( <i>processing symbol</i> ) atau simbol pengolah yang digunakan suatu proses.
<p>Garis alir</p> 	Simbol garis alir ( <i>flow lines symbol</i> ), digunakan untuk menunjukkan arus dari proses
<p>Penghubung</p> 	Simbol penghubung ( <i>connector symbol</i> ), digunakan untuk menunjukkan sambungan dari bagan alir yang terputus dihalaman yang sama / dihalaman yang lain.
<p>Keputusan</p> 	Simbol keputusan ( <i>decision symbol</i> ), digunakan untuk suatu penyelesaian kondisi didalam program.
<p>Proses Terdefinisi</p> 	Simbol proses terdefinisi digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan ditempat lain.
<p>Terminal</p> 	Simbol terminal digunakan untuk menunjukkan awal dan akhir dari suatu program.

### 1. Jenis – Jenis Pengkodean

Kode digunakan untuk tujuan mengklasifikasikan data, memasukan data kedalam komputer dan untuk mengambil bermacam-macam informasi yang berhubungan dengannya. Kode dapat dibentuk dari kumpulan angka, huruf dan karakter - karakter khusus. (Jogiyanto HM, 2005).

Tipe kode yang digunakan diantaranya sebagai berikut.

a. Kode *mnemonik*

Kode mnemonik digunakan untuk tujuan supaya mudah diingat dengan dasar singkatan.

b. Kode urut (*sequential code*)

Kode urut disebut juga dengan kode seri, merupakan kode yang nilainya urut antara satu kode dengan kode berikutnya.

c. Kode blok (*block code*)

Kode blok mengklasifikasikan item kedalam kelompok blok tertentu yang mencerminkan suatu klasifikasi tertentu atas dasar pemakaian maksimum yang diharapkan.

d. Kode group

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

## 2. Kamus Data

Kamus data adalah suatu daftar elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga *user* dan analisis sistem mempunyai pengertian yang sama tentang *input*, *output* dan komponen *data store*. Pembentukan kamus data didasarkan pada alur data yang terdapat pada DFD. Alur data pada DFD bersifat global (hanya menunjukkan alur datanya tanpa menunjukkan struktur dari alur data). Untuk menunjukkan struktur dari alur data secara rinci maka dibentuklah kamus data (Jogiyanto HM, 2005). Berikut contoh tampilan format kamus data pada gambar 2.1.

Nama Database	:		
Nama Tabel	:		
Primary Key	:		
Foreign Key	:		
Nama Field	Type	Size	Keterangan

Gambar 2.1 Format Kamus Data

Untuk menghasilkan sistem informasi yang baik maka dibutuhkan beberapa pendukung yang mampu mengembangkan aplikasi yang akan digunakan sehingga aplikasi yang dibuat benar - benar bisa dimanfaatkan dengan baik.

### **3. Basis Data**

Basis data adalah kumpulan file-file yang mempunyai kaitan antara satu file dengan file yang lain sehingga membentuk satu bangunan data untuk menginformasikan satu perusahaan, instansi dalam bahasa tertentu (Kristanto, 1994). Penyusunan basis data digunakan untuk mengatasi masalah-masalah pada penyusunan data antara lain sebagai berikut :

#### **a. Redudansi dan Inkonsistensi data**

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

#### **b. Kesulitan Pengaksesan Data**

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

#### **c. Isolasi Data Standarisasi**

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

#### **d. Banyak Pemakaian (Multiple User)**

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

#### **e. Masalah Keamanan (Security)**

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

#### **4. Relasi Antar Tabel**

Suatu file yang terdiri dari beberapa grup elemen yang berulang-ulang perlu diorganisasikan kembali. Proses untuk mengorganisasikan file untuk menghilangkan grup elemen yang berulang-ulang dan digambarkan dalam bentuk database struktur hubungan disebut Relasi Antar Tabel (Jogiyanto, 2005).

Database dengan struktur data hubungan dapat digambarkan dalam bentuk dua dimensi. Kolom dari tabel menunjukkan atribut dari file. Atribut ini menunjukkan item data atau field. Kumpulan nilai dari field atau item data disebut dengan istilah domain. Masing-masing baris dari record di dalam tabel disebut dengan istilah tuple. Suatu tuple (record) yang mempunyai dua domain disebut dengan 2 tuple. Suatu tuple yang mempunyai 3 domain disebut dengan 3-tuple dan seterusnya. Tiap-tiap tuple atau record ini dapat mempunyai suatu kunci yang unik dengan cara mana tuple ini dapat diidentifikasi. Field yang menjadi kunci yang unik ini disebut dengan field kunci (key field). Database didukung oleh beberapa komponen antara lain :

##### **a. Tabel**

Pengelompokan secara logika dari system yang berhubungan, table- table dibentuk dari baris- baris dan kolom -kolom, baris-baris disebut sebagai record - record, sedangkan kolom-kolom disebut sebagai field.

##### **b. Record**

Suatu record mengandung informasi dari suatu isian tunggal dari suatu table.

##### **c. Field**

Suatu record dibentuk oleh banyak field dalam satu record mengandung suatu potongan tunggal dari informasi tentang record.

##### **d. Primary key**

Primary key adalah suatu field atau kombinasi dari field-field yang mana nilainya bersifat untuk tiap baris atau record dalam table.

##### **e. Foreign key**

Atribut yang menjadi atribut penghubung antara satu file dengan file lainnya.

### 2.3 Teori Pengembangan Aplikasi

Pada sub ini akan menjelaskan beberapa teori mengenai pengembangan aplikasi yang akan mendukung sistem informasi, seperti bahasa pemrograman yang akan digunakan serta aplikasi pendukung, *database* yang akan digunakan dan *report generator* yang akan digunakan.

#### 2.3.1 Macromedia Dreamweaver

Macromedia dreamweaver adalah editor yang komplit yang dapat digunakan untuk membuat animasi sederhana yang berbentuk layer. Macromedia dreamweaver juga mempunyai dua bentuk layer yaitu bentuk halaman *design* dan halaman *Code*, Hal ini akan mempermudah kita dalam menambahkan scrip yang berbasis PHP maupun Javascrip. (Nugroho, 2004)

#### 2.3.2 Xampp

Menurut Wicaksono (2008:7) menjelaskan bahwa “XAMPP adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis PHP dan menggunakan pengolah data MYSQL di komputer lokal”. XAMPP berperan sebagai *server web* pada komputer lokal. XAMPP juga dapat disebut sebuah *Cpanel server virtual*, yang dapat membantu melakukan *preview* sehingga dapat dimodifikasi *website* tanpa harus *online* atau terakses dengan *internet*.

#### 2.3.3 PHP

Menurut Nugroho (2004), *PHP* adalah sebuah bahasa pemrograman yang berbentuk *scripting*. Sistem kerja ini adalah *interpreter* bukan sebagai kompiler. Bahasa *interpreter* adalah bahasa yang *script* program tidak harus diubah kedalam bentuk *sourcecode*, sedangkan bahasa kompiler adalah bahasa yang mengubah script2 program kedalam *sourcecode* , selanjutnya dari bentuk *source code* akan diubah menjadi *object code*, bentuk dari *object code* akan menghasilkan *file* yang lebih kecil dari file mentahsebelumnya.

#### 2.3.4 PHPMYAdmin

*PHPMYAdmin* merupakan sebuah program yang berbasis *web* yang dibuat menggunakan aplikasi *PHP*. Program ini adalah untuk mengakses database

*MySQL*, intinya adalah digunakan untuk menjadi administrator dari *server MySQL*. Dengan adanya program ini akan mempermudah dan mempersingkat kinerja, dengan kelebihan<sup>2</sup> yang ada mengakibatkan pengguna awam tidak harus mampu mengetahui *scriptSQL* dalam pembuatan database dan tabel (Nugroho, 2004)

### 2.3.5 Database

Setelah mengetahui alat-alat yang digunakan dalam metode terstruktur, peranan *database* merupakan hal yang penting untuk dipahami. Menurut Jogiyanto (2005) berpendapat basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di simpanan luar komputer dan digunakan perangkat lunak tertentu untuk memanipulasinya.

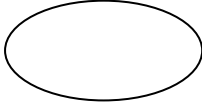
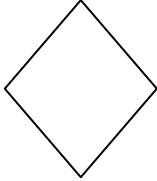

Dari pendapat tersebut dapat disimpulkan bahwa *database* adalah sekumpulan informasi yang berhubungan dengan subjek atau tujuan tertentu seperti melacak pesanan konsumen, *maintaining inventory* atau pada umumnya mengumpulkan semua data yang berhubungan pada satu tempat. Data tersebut dapat disimpan, dimanipulasi dan digunakan dalam banyak cara.

Dalam perancangan *database* terdiri dari 2 teknik yang paling sering digunakan, yaitu teknik normalisasi dan teknik *entity relationship diagram*. Teknik normalisasi yaitu proses untuk mengubah suatu tabel yang memiliki masalah tertentu ke dalam 2 tabel atau lebih yang tidak memiliki masalah tersebut. Sedangkan *Entity Relationship Diagram* (ERD) adalah suatu pemodelan konseptual yang didesain secara khusus untuk mengidentifikasi entitas yang menjelaskan data dan hubungan antar data. Adapun simbol-simbol yang terdapat pada *entity relationship diagram* sebagai berikut :

Tabel 2.4 Simbol *Entity Relationship Diagram*

Simbol	Keterangan
<p style="text-align: center;"><i>Entity</i></p> <div style="text-align: center; border: 1px solid black; width: 150px; height: 20px; margin: 0 auto;"></div>	<p style="text-align: center;">Menyatakan tabel, <i>file</i> atau entitas.</p>

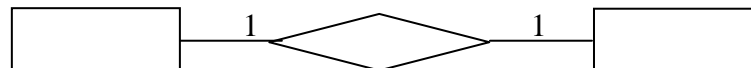


<p><i>Atributte</i></p> 	Menyatakan atribut atau <i>field</i> .
<p><i>Relationship</i></p> 	Menyatakan relasi antar entitas.
<p>Penghubung</p> 	Sebagai penghubung antara entitas dengan atribut dan antara entitas dengan relasi.

Dalam ERD terdapat hubungan atau relasi yang bisa terjadi antara dua *entitas* adalah sebagai berikut.

**a. One to one relationship (2 file)**

Hubungan antara *file* pertama dengan *file* kedua adalah satu banding satu. Hubungan tersebut dapat digambarkan dengan tanda lingkaran untuk menunjukkan tabel dan relasi antara keduanya, diwakili dengan panah tunggal.

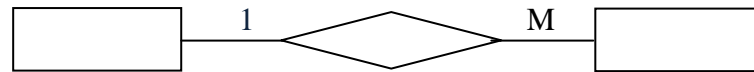


Hubungan relasi satu ke satu (*one to one*).

Gambar 2.2 Hubungan Antar Atribut One To One

**b. One to many relationship (2 file)**

Hubungan antara *file* pertama dan *file* kedua adalah satu berbanding banyak atau dapat pula sebaliknya. Hubungan tersebut dapat digambarkan dengan tanda lingkaran untuk menunjukkan tabel dan relasi antara keduanya, diwakili dengan panah tunggal untuk satu (*one*) dan panah ganda untuk banyak (*many*).

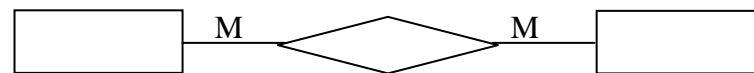


Hubungan relasi satu ke banyak (*one to many*).

Gambar 2.3 Hubungan Antar Atribut One To Many

**c. Many to many relationship (2 file)**

Hubungan antara file pertama dan file kedua adalah banyak berbanding banyak. Hubungan tersebut dapat digambarkan dengan tanda lingkaran untuk menunjukkan tabel dan relasi antara keduanya, diwakili dengan panah ganda.



Hubungan relasi banyak ke banyak (*many to many*).

Gambar 2.4 Hubungan Antar Atribut Many To Many

*Database* yang akan digunakan oleh penulis kali ini adalah *MySQL*. Berikut penjelasan mengenai beberapa hal yang ada pada *MySQL*.

**A. MySQL**

*MySQL* adalah sebuah program *database server* yang mampu menerima dan mengirimkan datanya dengan sangat cepat, *multi user* serta menggunakan perintah standar SQL (*Structured Query Language*). *MySQL* juga telah mendukung bahasa pemrograman berfitur API seperti Java sehingga memudahkan para programmer java untuk berkoneksi dengan menggunakan *MySQL*. (Wahana Komputer, 2010).

Berikut beberapa keistimewaan yang dimiliki *MySQL* antara lain:

1. *Portabilitas* yaitu *MySQL* dapat berjalan secara stabil pada berbagai macam sistem operasi.
2. *Open Source* yaitu *MySQL* di distribusikan secara gratis.
3. *Multi User* yaitu *MySQL* dapat digunakan dengan banyak *user* tanpa memiliki masalah.
4. *Performance Turning* yaitu *MySQL* memiliki tipe kolom yang sangat kompleks.

5. Perintah dan fungsi yaitu *MySQL* operator dan fungsi yang secara penuh perintah *Select* dan *Where* dalam *query*.
6. Keamanan yaitu *MySQL* memiliki beberapa lapisan keamanan dengan sistem perijinan yang detail serta sandi yang terenkripsi.
7. Skalabilitas dan pembatasan yaitu *MySQL* mampu menangani *database* berskala besar dengan jumlah *record* lebih dari 50 juta, tabel lebih dari 60 ribu dan baris lebih dari 5 milyar.
8. *Konektivitas* yaitu *MySQL* dapat melakukan koneksi dengan klien menggunakan *TCP/IP*, *UNIX* dan *Named Pipes*.
9. Lokalisasi yaitu *MySQL* dapat mendeteksi kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa meskipun bahasa Indonesia belum masuk di dalamnya.
10. Antarmuka yaitu *MySQL* memiliki antarmuka yang mendukung berbagai bahasa pemrograman yang menggunakan fungsi API.
11. Klien dan Peralatan yaitu *MySQL* dilegkapi dengan berbagai *tool* yang dapat digunakan secara *online*.
12. Struktur tabel yaitu *MySQL* memiliki struktur tabel yang *fleksibel* dalam menangani *alter table* jika dibandingkan dengan bahasa pemrograman lainnya.

## **B. Perintah dalam MySQL**

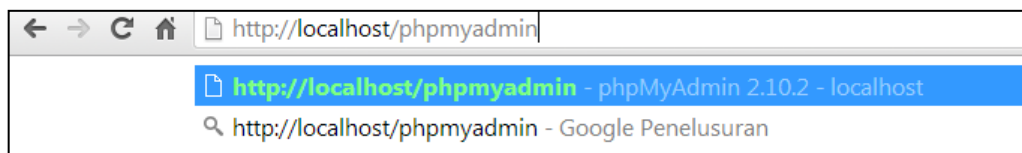
Berikut adalah beberapa intruksi bantuan yang ada pada *MySQL* dan semua perintah ini dapat langsung di ketik pada *prompt MySQL*.

1. (;) merupakan tanda yang menyatakan bahwa semua *query* harus diakhiri dengan tanda titik koma (;). Tanda ini menunjukkan bahwa *query* telah berakhir dan siap dieksekusi.
2. *Help* (\h) tanda ini digunakan untuk menampilkan *file* bantuan.
3. ? (\?) perintah ini sama memiliki fungsi sama dengan *help*.
4. *Clear* (\c) tanda ini berguna untuk membersihkan semua perintah yang telah berjalan dalam satu *prompt*, baik perintah salah maupun benar.
5. *Connect* (\r) tanda ini untuk melakukan penyegaran koneksi kedalam *database* yang ada pada *server host*.

6. *Ego* (\G) tanda ini berguna untuk menampilkan data secara *horizontal* (kekanan).
7. *Go* (\g) tanda ini memberi perintah *server* untuk mengeksekusi.
8. *Note* (\t) tanda ini berguna untuk mendokumentasikan semua *query* dan hasilnya kedalam sebuah *file* yang akan disimpan kedalam direktori *server*.
9. *Print* (\p) tanda ini berguna untuk mencetak semua *query* yang telah diperintahkan kelayar.
10. *Prompt* (\R) tanda ini memiliki perintah mengubah *prompt* standar *MySQL* menjadi *prompt* yang dikehendaki.
11. *Quit* (\q) tanda ini memiliki perintah untuk keluar dari *server MySQL*.
12. *Source* (\.) tanda ini untuk mengeksekusi *query* yang berasal dari *file* luar.
13. *Status* (\s) tanda ini untuk melihat status *server* yang sedang digunakan.

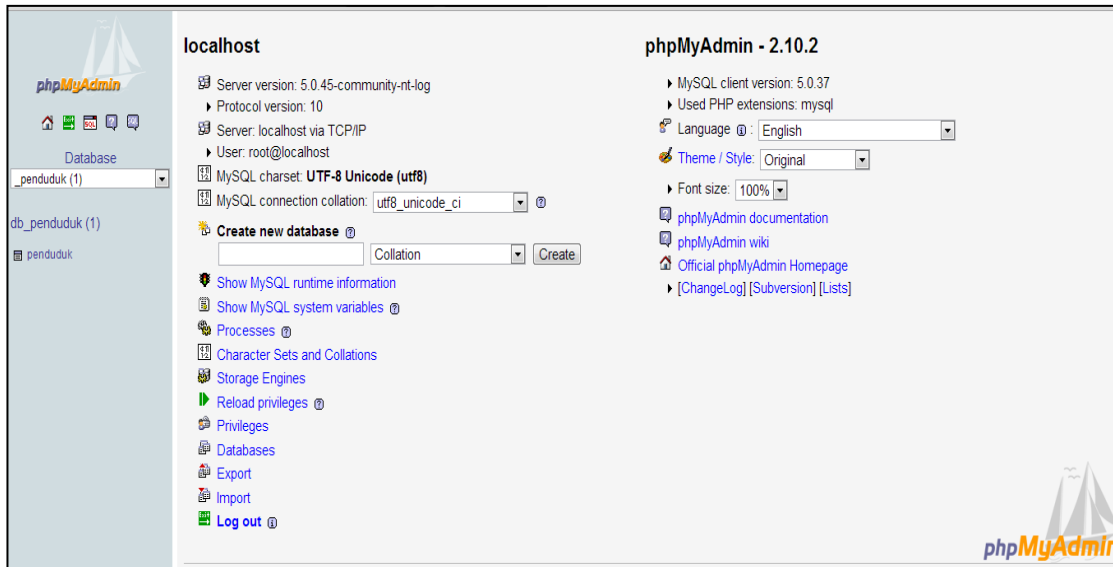
### C. Langkah – Langkah Menjalankan *MySQL*

1. Sebelum menjalankan *MySQL* pastikan aplikasi *appserv* telah terinstal pada PC atau komputer.
  - a. Ketikkan alamat <http://localhost/phpmyadmin> pada *browser* seperti gambar 2.5 berikut ini



Gambar 2.5 Tampilan Alamat Localhost pada Browser

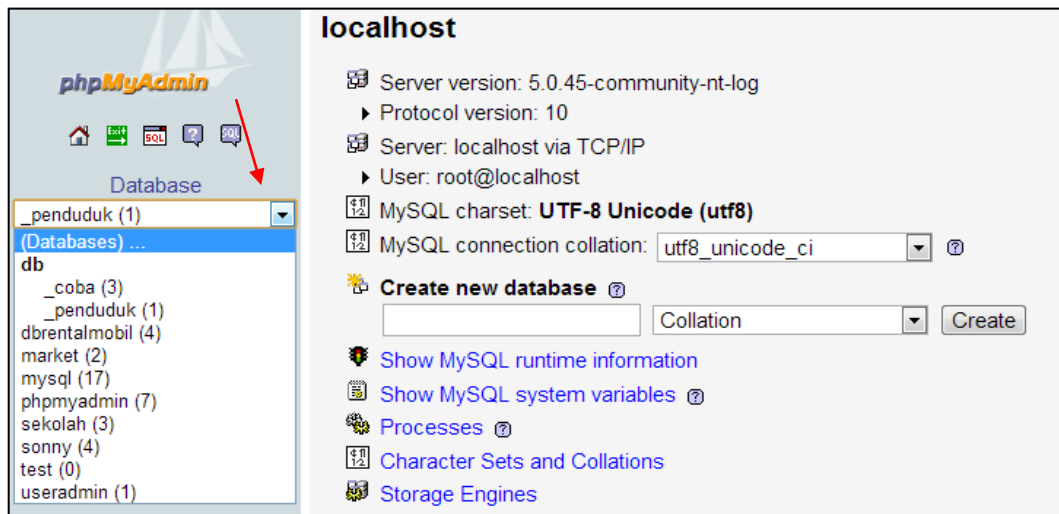
- b. Lalu masukan *username* dan *password* yang telah ditentukan pada saat instalasi.
- c. Kemudian akan tampil lembar kerja *MySQL* seperti gambar 2.6 dibawah ini.



Gambar 2.6 Tampilan Lembar Kerja *MySQL*

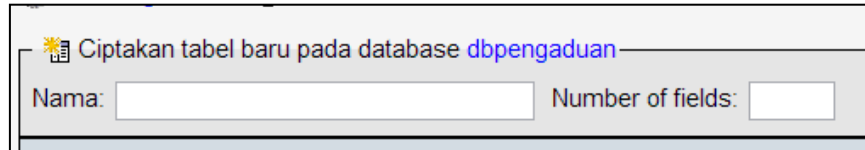
## 2. Berikut langkah – langkah dalam membuat database pada *MySQL*

- a. Ketikkan nama *database* pada kolom *create new database*
- b. Lalu klik *create*
- c. Setelah itu akan muncul nama *database* yang telah dibuat tadi pada daftar yang ada disebelah kiri halaman awal *MySQL* seperti gambar 2.7 dibawah ini.



Gambar 2.7 Tampilan Daftar Nama *Database*

3. Berikut langkah – langkah dalam membuat *table*
  - a. Ketikkan nama *table* pada kolom *create new table* pada *database* yang telah dibuat sebelumnya seperti gambar 2.8 dibawah ini



Ciptakan tabel baru pada database dbpengaduan

Nama:  Number of fields:

Gambar 2.8 Tampilan Membuat Table Baru

- b. Setelah itu isikan *field* sesuai dengan kebutuhan, lalu klik *go*,
- c. Kemudian isikan nama field, tipe data, lenght, primary key dengan sesuai, lalu klik *button* simpan.

#### D. Tipe Data pada *MySQL*

Tipe data adalah suatu bentuk pemodelan data yang dideklarisikan pada saat melakukan pembuatan tabel (Wahana Komputer, 2010). Jenis tipe data pada *MySQL* dapat dibagi kedalam beberapa bentuk yaitu.

##### 1. Data *numeric*.

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

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

## 2. Data untuk penanggalan dan waktu

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

- a. *Datetime* : bentuk ini merupakan tipe data yang dapat menyimpan dua bentuk tipe data sekaligus, yaitu penanggalan dan waktu.
- b. *Date* : bentuk ini digunakan untuk menyimpan data penanggalan dan memiliki bentuk penulisan.
- c. *Timestamp* : bentuk penanggalan *timestamp* dapat diciptakan berjajar tanpa ada pembatasnya.
- d. *Time* : bentuk *time* digunakan untuk menyimpan data berbentuk penanggalan dan dimulai dari tahun dan diikuti bulan dan tanggal.
- e. *Year* : bentuk yang paling sederhana adalah *year*,

## 3. Data *string*

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