

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Dalam kaitannya dengan sistem informasi, sistem adalah sekelompok komponen yang terkait, yang bekerja sama untuk suatu tujuan umum dengan menerima masukkan data dan menghasilkan keluaran dari proses transformasi yang terorganisir. Sistem adalah sekelompok elemen-elemen yang terintegrasi dengan maksud untuk pencapaian suatu tujuan tertentu. Berdasarkan definisi tersebut dapat disimpulkan bahwa sistem merupakan kerangka yang *terintegrasi* yang mempunyai satu atau lebih tujuan, serta mengkoordinasi berbagai sumber daya yang ada untuk kemudian mengubah masukan (*input*) menjadi keluaran (*output*). (Jogiyanto H.M., 2005).

2.1.1 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu, yaitu mempunyai komponen-komponen, batas sistem, lingkungan luar sistem, penghubung, masukan keluaran, pengolah, dan sasaran atau tujuan.

a) Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem dapat berupa subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

b) Batas Sistem

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menunjukkan ruang lingkup dari sistem tersebut.

c) Lingkungan Luar Sistem

Lingkungan luar sistem adalah apapun batas dari sistem yang mempengaruhi operasi sistem, lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Lingkungan luar yang merugikan harus ditahan dan dikendalikan, jika tidak maka akan mengganggu kelangsungan hidup dari sistem.

d) Penghubung Sistem

Penghubung merupakan media penghubung antara subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya.

e) Masukan Sistem

Masukan adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa perawatan dan masukan sinyal. Masukan perawatan adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Masukan sinyal adalah energi yang diproses untuk didapatkan keluaran.

f) Keluaran Sistem

Keluaran adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat menjadi masukan untuk subsistem yang lain.

g) Pengolah Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran. Suatu sistem produksi akan mengolah masukan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran yang berupa barang.

h) Sasaran Sistem

Suatu sistem pasti mempunyai tujuan atau sasaran, sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem.

2.1.2 Klasifikasi Sistem

Suatu sistem dapat diklasifikasikan sebagai berikut :

a) Sistem Abstrak dan Sistem Fisik

Sistem abstrak adalah suatu sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, sedangkan sistem fisik adalah sistem yang ada secara fisik.

b) Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam sedangkan sistem buatan manusia adalah sistem yang dirancang oleh manusia.

c) Sistem Tertentu dan Sistem Tak Tentu

Sistem tertentu adalah suatu sistem yang operasinya dapat diprediksi secara tepat sedangkan sistem tak tentu adalah sistem dengan perilaku kedepan yang tidak dapat diprediksi.

d) Sistem Tertutup dan Sistem Terbuka

Sistem tertutup adalah sistem yang tidak terpengaruh oleh lingkungan luar sedangkan sistem terbuka adalah sistem yang berhubungan dan terpengaruh oleh lingkungan luar.

2.2 Konsep Dasar Informasi

Informasi merupakan bagian terpenting dari sebuah organisasi. Data yang diolah melalui suatu model menjadi informasi, kemudian diberikan oleh penerima untuk membuat suatu keputusan dan melakukan tindakan yang berarti akan membuat sejumlah data kembali. Informasi dapat diartikan sebagai hasil dari pengolahan data kedalam bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang digunakan untuk pengambilan keputusan. (Jogiyanto H.M., 2005). Informasi yang baik adalah informasi yang berkualitas. Informasi yang berkualitas dapat ditentukan oleh hal-hal sebagai berikut(Jogiyanto.HM,2005):

a) Relevan

Berarti informasi tersebut mempunyai manfaat untuk pemakainya. *Relevansi* informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda. Misalnya informasi mengenai sebab-musabab kerusakan mesin produksi kepada akuntan perusahaan adalah kurang *relevan* dan akan lebih *relevan* bila ditujukan kepada ahli teknik perusahaan. Sebaliknya informasi mengenai harga pokok produksi untuk ahli teknik merupakan informasi yang kurang *relevan*, tetapi relevan untuk akuntan.

b) Dapat dipercaya (*Reability*)

Informasi yang akan diberikan dapat dipercaya kebenarannya dan mempunyai data-data yang lengkap dan jelas sumber-sumber datanya.

c) Tepat Waktu (*Timelines*)

Informasi yang dihasilkan atau dibutuhkan tidak boleh terlambat, karena nantinya tidak mempunyai nilai yang baik, sehingga apabila dijadikan dasar dalam pengambilan keputusan akan berakibat fatal atau kesalahan pengambilan keputusan dan tindakan.

d) Akurat

Berarti informasi tersebut harus bebas dari kesalahan-kesalahan dan tidak menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dapat merubah atau merusak informasi tersebut.

e) Ekonomis

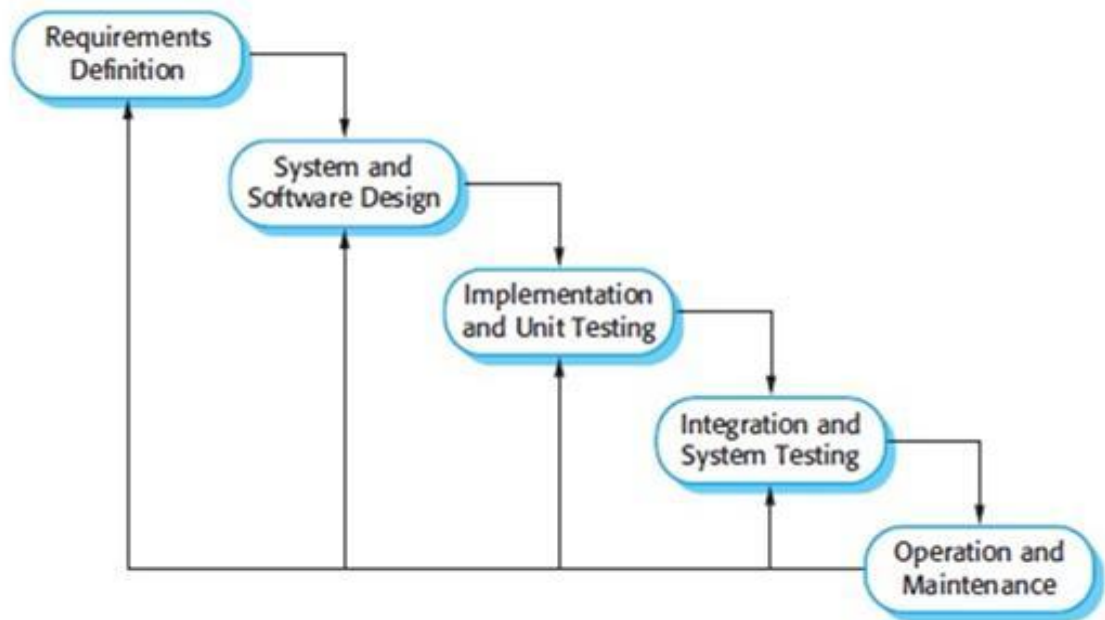
Maksud ekonomis disini adalah dilihat dari segi biaya pembuatan informasi tersebut, diusahakan murah dan memberikan manfaat yang besar bagi pemakai.

2.3 Metodologi Pengembangan Sistem

Dalam penggunaan metode pengembangan sistem baru ini penulis menggunakan Metode *waterfall*, dimana semua proses yang dilakukan secara berurutan sesuai dengan urutan yang ada. Model *waterfall* menyediakan pendekatan alur hidup

perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap *maintenance*. Berikut ini tahapan pengembangan sistem (Sommerville,2011).

Model tahapan pengembangan sistem *Waterfall* ada pada Gambar 2.1 berikut :



Gambar 2.1 *Waterfall*

(Sumber : Sommerville, 2011)

2.3.1 *Requirement Definition*

Requirement definition merupakan tahap pertama yang menjadi dasar pembuatan sistem. Kebutuhan fungsional dan kebutuhan nonfungsional. kebutuhan fungsional adalah kebutuhan yang berisi proses-proses apa saja yang nantinya dilakukan oleh sistem. Sedangkan kebutuhan nonfungsional adalah kebutuhan yang menitik beratkan pada properti perilaku yang dimiliki oleh sistem

2.3.2. System and Software Design

System and Software Design merupakan tahapan perancangan software yang digunakan untuk mendukung sistem untuk membantu perancangan pemodelan sistem.

2.3.3. Implementation and Unit Testing

Pada tahap ini menghasilkan sistem informasi kepegawaian dengan aplikasi berbasis web sebagai penerapan rancangan sistem dan pengujian unit.

2.3.4. Integration and System Testing

Pada tahap ini sistem yang dipakai akan digunakan sebagai sistem pengarsipan pada UPT Dinas Pendidikan Kec. Rebang Tangkas, kemudian dengan menggunakan data-data dari dokumen perusahaan ini mengintegrasikan sistem lama yang konvensional menjadi terkomputerisasi.

2.3.5. Operation and Maintenance

Setelah semua tahapan dilalui, tahapan ini merupakan tahapan akhir sebagai penggunaan sistem ini, apabila ada kesalahan pada sistem ini. Akan ada perancangan kembali pada tahapan awal.




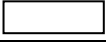



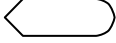
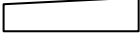
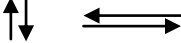
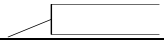
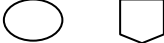


2.4 Alat Bantu Perancangan Sistem

Alat Bantu perancangan sistem yang dapat digunakan adalah :

a) Bagan Alir Dokumen

Bagan alir dokumen (*Document flowchart*) merupakan bagan alir yang menunjukkan arus data dari laporan dan formulir termasuk tembusan-tembusannya. Simbol yang digunakan dalam bagan alir dokumen dapat dilihat pada tabel dibawah ini.

Tabel 2.1 Simbol-simbol Bagan Alir Dokumen

Simbol	Keterangan
Dokumen 	Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan
Kegiatan manual 	Menunjukkan pekerjaan yang dilakukan tidak dengan program komputer.
Simpanan / arsip 	Menunjukkan dokumen yang diarsipkan (arsip manual)
Proses 	Menunjukkan kegiatan proses yang dilakukan dengan program komputer
Hard Disk 	Menunjukkan <i>input / output</i> menggunakan <i>hard disk</i>
Terminasi 	Menunjukkan awal atau akhir dari suatu proses.
Keputusan 	Menunjukkan suatu penyeleksian kondisi di dalam program.
Display 	Menunjukkan <i>output</i> yang ditampilkan di <i>monitor</i>
Pemasukan 	Menunjukkan <i>input data</i> secara manual melalui <i>keyboard</i>
Garis alir 	Menunjukkan arus dari data
Penjelasan 	Menunjukkan penjelasan dari suatu proses
Penghubung 	Menunjukkan penghubung kehalaman yang masih sama atau ke halaman lain.
Produk 	Menunjukkan Barang, bias dalam bentuk baju, batik, dan lain-lain
	Menunjukkan komunikasi menggunakan telepon

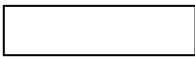


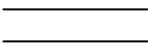
b) Diagram Konteks

Diagram konteks adalah suatu diagram yang menggambarkan suatu proses pengolahan data secara umum dalam satu lingkungan dan hubungan dengan *entitas* luar.

c) DFD (*Data Flow Diagram*)

DFD adalah penjelasan lebih rinci dari diagram konteks dan proses fungsional yang ada dalam sistem. DFD menjelaskan tentang aliran masuk, aliran keluar, proses serta penyuntingan *file* yang digunakan. Keuntungan menggunakan DFD adalah memudahkan pemakai (*user*) yang kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan atau dikembangkan.

Tabel 2.2 Simbol – symbol DFD

Simbol	Nama Simbol	Keterangan
	Simbol Terminator (<i>Eksternal/Internal Entity</i>)	Suatu kesatuan luar dapat disimbolkan dengan <i>notasi</i> kotak.
	Simbol Proses	Suatu Proses dapat ditunjukkan dengan simbol lingkaran.
	Simbol arus data (<i>Data Flow</i>)	Arus data diberi simbol suatu panah
	Simpanan data (<i>Data Store</i>)	Simpanan data dapat disimbolkan dengan sepasang garis <i>horizontal paralel</i> yang tertutup disalah satu ujungnya.

Berikut ini keterangan simbol yang digunakan dalam *Data flow Diagram* (DFD) :

1. Kesatuan Luar (*External Entity*)

Setiap sistem mempunyai batas sistem (*boundary*) yang memisahkan suatu sistem dengan lingkungan luarnya (*Eksternal Entity*) merupakan kesatuan (*entity*) dilingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainya yang berada di lingkungan luarnya yang akan memberikan *input* atau menerima *output* dari sistem.

2. Arus Data (*Data Flow*)

Arus Data (*Data Flow*) di DFD diberi simbol suatu panah, arus data ini mengalir diantara proses, simpanan data, dan kesatuan luar.

3. Proses (*Process*)

Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk kedalam proses untuk dihasilkan arus data yang akan keluar dari proses.

4. Simpanan Data (*Data Store*)

Simpanan Data (*Data Store*) merupakan simpanan dari data.

d) Kamus Data

Kamus Data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Kamus data dibuat berdasarkan arus data yang ada di diagram aliran data. Dengan menggunakan kamus data, *analisis* sistem dapat mendefinisikan data yang mengalir di sistem secara lengkap. Kamus data digunakan untuk merancang *input*, merancang laporan dan *database*.

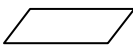
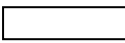
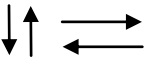
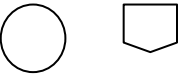
Format Kamus Data				
Nama database :				
Nama tabel :				
Primary Key :				
Foreign Key :				
Nama Field	Type	Size	Kondisi	Keterangan
Keterangan: Kondisi berisi (contoh: NULL/NOT NULL)				

Gambar 2.2 *Format Kamus Data*

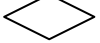



e) Bagan alir program (*program flowchart*)

Bagan alir program (*program flowchart*) adalah bagian *flowchart* yang menggambarkan arus logika dari data yang akan diproses kedalam suatu program mulai dari awal sampai akhir. Bagan alir merupakan alat yang berguna bagi *programmer* untuk mempersiapkan program yang rumit. Simbol – simbol program yang digunakan antara lain sebagai berikut:

Tabel 2.3 Simbol-simbol Bagan alir program (*Program Flowchart*)

Simbol	Keterangan
<i>Input / Output</i> 	Simbol <i>input/output</i> digunakan untuk mewakili data <i>input/output</i>
Proses 	Simbol proses digunakan untuk mewakili suatu proses.
Garis Alir 	Simbol garis alir (<i>flow lines symbol</i>) digunakan untuk menunjukkan arus dari proses
Penghubung 	Simbol penghubung (<i>connector symbol</i>) digunakan untuk menunjukkan sambungan dari bagan alir yang terputus dihalaman yang sama / dihalaman yang lain

Tabel 2.3 Simbol-simbol Bagan alir program (Lanjutan)

Simbol	Keterangan
Keputusan 	Simbol keputusan (<i>decision symbol</i>) digunakan untuk suatu penyelesaian kondisi di dalam program
Proses terdefinisi 	Simbol proses terdefinisi digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain
Persiapan 	Simbol persiapan (<i>preparation symbol</i>) digunakan untuk memberi nilai awal suatu besaran.
<i>Terminal</i> 	Simbol <i>terminal</i> (<i>terminal symbol</i>) digunakan untuk menunjukkan awal dan akhir dari suatu proses / program

2.5. Sistem Basis Data

Sistem basis data adalah kumpulan subsistem yang terdiri atas basis data dengan para pemakai yang menggunakan basis data secara bersama–sama, personal–personal yang merancang dan mengolah basis data, teknik–teknik untuk merancang dan mengelola basis data, serta sistem komputer untuk mendukungnya. Dari pengertian tersebut dapat disimpulkan bahwa sistem basis data mempunyai beberapa elemen penting yaitu :

- Basis data sebagai inti dari sistem basis data
- Perangkat lunak (*software*) untuk perancangan dan pengelolaan basis data
- Perangkat keras (*Hardware*) sebagai pendukung operasi pengolahan data
- Manusia (*Brainware*) yang mempunyai peran penting dalam sistem tersebut yaitu sebagai pemakai atau para *spesialis* informasi yang mempunyai fungsi sebagai perancangan atau pengelola

Istilah Dalam *Database*

Beberapa istilah dalam *database* yang sering dipakai antara lain :

- Entity*

Entity adalah konsep informasi yang direkam, meliputi orang, kejadian dan tempat.

b) *Atribut* atau *Field*

Atribut atau *Field* adalah sesuatu yang mewakili *entity*.

c) *Data Value*

Data Value adalah informasi yang tersimpan dalam setiap *atribut*.

d) *Record*

Record adalah kumpulan *atribut* yang saling berkaitan satu dengan yang lain dan menginformasikan suatu *entity* secara lengkap.

e) *File*

File adalah kumpulan *record* yang mempunyai panjang atribut yang sama tetapi berbeda data *valuenya*.

f) *Basis data* atau *Database*

Database adalah kumpulan *file* satu dengan *file* yang lainnya yang membentuk satu informasi sistem secara keseluruhan.

Field kunci memegang peranan yang sangat penting dalam pembuatan tabel yang berisi *entity* dan *relasinya*. *Field* kunci merupakan satu *field* atau satu *set field* yang terdapat dalam satu *file* yang merupakan kunci dan mewakili *record*. Kunci disini akan sangat penting apabila dalam program nanti terdapat fasilitas pencarian, karena *field* yang merupakan kunci akan menjadi penentu dalam pencarian program. *Field* kunci dibagi menjadi beberapa bagian yaitu :

a) *Kunci Kandidat*

Kunci Kandidat adalah satu *atribut* atau *field* yang mengidentifikasi secara unik dari suatu kejadian yang sifatnya khusus dari suatu *entity*.

b) *Kunci Primer*

Kunci Primer adalah kunci kandidat yang dipilih untuk mewakili setiap kejadian dari suatu *entity*.

c) *Kunci Alternatif*

Kunci Alternatif adalah kunci kandidat yang tidak dipakai sebagai kunci *primer*.

d) *Kunci Tamu*

Kunci Tamu adalah kunci *primer* yang ditempatkan pada *file* lain dan biasanya menunjukan dan melengkapi suatu hubungan antara *file* satu dengan *file* yang lainnya.

Manfaat dari *database* sebagai berikut :

- a) Media permanen penyimpana pengolahan data.
- b) Petunjuk dan penjelasan bagaimana hasil pengolahan data disimpan.
- c) Data yang tersimpan dapat diubah dan dihapus.
- d) Rujukan pembuatan laporan

Sedangkan tujuan dari konsep *database* sebagai berikut :

- a) Meminimumkan terjadinya pengulangan data yang sama (*redudancy data*).
- b) Mencapai *interpendensi data*.

2.6 Normalisasi

a) Teknik *Normalisasi*

Teknik *normalisasi* adalah suatu teknik yang digunakan untuk membantu dalam merancang struktur basis data secara rinci. Walaupun tanpa teknik *normalisasi*, proses pengembangan aplikasi tetap dapat dilakukan, namun untuk efisiensi media penyimpanan data yang digunakan dalam aplikasi tersebut maka teknik ini menjadi sangat penting untuk dipergunakan. Pengalaman menunjukan bahwa struktur basis data yang dirancang dengan menggunakan teknik *normalisasi* akan lebih stabil terhadap perubahan. Langkah-langkah dalam perancangan struktur basis data dengan menggunakan teknik *normalisasi* adalah sebagai berikut :

1. Membuat bentuk un-normal (*un-normalized form*)

Yaitu dengan cara memasukan seluruh *atribut* yang diperlukan ke dalam satu *file* atau *relasi* kemudian tentukan *atribut* kuncinya (*key field*).

2. Membuat bentuk normal pertama (*1NF*)

Menurut James Martin bentuk normal pertama adalah suatu *relasi* yang tidak mengandung grup berulang (*repeating group*). Untuk mendapatkan normal pertama adalah dengan cara memisahkan grup berulang ke dalam relasi baru,

kemudian kunci utamanya (*primary key*) adalah kunci utama grup berulang ditambah kunci utama relasi asal.

3. Membuat bentuk normal kedua (*2NF*)

Sebuah *relasi* dikatakan dalam bentuk normal kedua bila relasi tersebut dalam bentuk normal pertama serta seluruh *atribut* (yang bukan kunci utama) tergantung secara fungsional sepenuhnya pada kunci utama (tidak hanya pada sebagian kunci utama). Untuk mendapatkan bentuk normal kedua adalah dengan cara memisahkan *atribut* yang tergantung secara fungsional pada sebagian kunci utama kedalam *relasi* baru, kemudian tambahkan *atribut* yang mengidentifikasikannya.

4. Membuat bentuk normal ketiga (*3NF*)

Suatu *relasi* dikatakan dalam bentuk normal ketiga jika *relasi* tersebut dalam bentuk normal kedua dan setiap *atributnya* tidak tergantung secara *transitif* pada kunci utama. Untuk mendapatkan bentuk normal ketiga adalah dengan cara memisahkan *atribut* yang mempunyai ketergantungan *transitifitas* ke dalam *relasi* baru, kemudian tambahkan *atribut* yang mengidentifikasikannya.


5. Membuat bentuk normal keempat (*4NF*)

Suatu *relasi* dikatakan dalam bentuk normal keempat jika relasi tersebut dalam bentuk normal ketiga dan seluruh *atribut* (yang bukan kunci utama) tidak tergantung bernilai banyak pada kunci utama (*multi valued dependencies*). Untuk mendapatkan normal keempat adalah dengan cara memisahkan *atribut* yang mempunyai ketergantungan nilai banyak ditambah kunci utama *relasi* asal menjadi kunci utama *relasi* baru.


b) *Diagram Relasi Entitas (Entity Relationship Diagram)*

Diagram relasi entitas adalah suatu alat yang dapat digunakan untuk mengorganisasikan data yang dikumpulkan dimana dalam *diagram* ini dapat diperlihatkan *entitas* beserta hubungan antar *entitas* tersebut. *Diagram relasi entitas* memperlihatkan seluruh *entitas* yang ada dalam sistem berikut dengan *relasinya*. Untuk membantu memberi gambaran *relasi* secara lengkap terdapat tiga macam *relasi* dalam hubungan antara *entitas* yaitu :

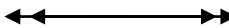
1. *One To One Relationship*

Hubungan antara entitas pertama dengan kedua adalah satu berbanding satu, hubungan tersebut dapat diwakilkan dengan  tanda panah tunggal.

2. *One To Many Relationship*

Hubungan antara *entitas* pertama dengan *entitas* kedua adalah satu berbanding banyak. *Relasi* antara keduanya diwakilkan dengan  panah ganda untuk menunjukkan hubungan banyak dan panah tunggal untuk hubungan satu.

3. *Many To Many Relationship*

Hubungan antara *entitas* pertama dan *entitas* kedua adalah banyak berbanding banyak, keduanya diwakilkan dengan  panah ganda.

2.7 Pengenalan Internet

Internet adalah sebuah jaringan komputer yang terdiri dari berbagai macam ukuran jaringan komputer diseluruh dunia mulai dari sebuah *PC*, jaringan lokal berskala kecil, jaringan kelas menengah hingga jaringan utama yang menjadi tulang punggung internet.(Jogiyanto HM, 2005).

Internet yang kita kenal saat ini pertama kali dikembangkan tahun 1969 oleh Departemen Pertahanan Amerika Serikat dengan nama *ARPAnet (US Defense Advanced Research Projects Agency)*. *ARPAnet* dibangun dengan sasaran untuk membuat suatu jaringan komputer yang tersebar untuk menghindari pemusatan informasi di satu titik yang dipandang rawan untuk dihancurkan apabila terjadi peperangan. Dengan cara ini diharapkan apabila satu bagian dari jaringan terputus, maka jalur yang melalui jaringan tersebut dapat secara *otomatis* dipindahkan ke saluran lainnya. Di tahun 1986 lahir *National Science Foundation Network (NSFNET)*, yang menghubungkan para periset di seluruh negeri dengan 5 buah pusat super komputer. Jaringan ini kemudian berkembang untuk menghubungkan berbagai jaringan akademis lainnya yang terdiri atas universitas dan *konsorsium-konsorsium* riset. *NSFNET* kemudian mulai menggantikan *ARPANET* sebagai

jaringan riset utama di Amerika hingga pada bulan Maret 1990 *ARPANET* secara resmi dibubarkan. Pada saat *NSFNET* dibangun, berbagai jaringan internasional didirikan dan dihubungkan ke *NSFNET*. Australia, negara-negara Skandinavia, Inggris, Perancis, Jerman, Kanada dan Jepang segera bergabung kedalam jaringan ini. Pada awalnya, internet hanya menawarkan layanan berbasis teks, meliputi *remote access*, *email/messaging*, maupun diskusi melalui *newsgroup (Usenet)*. Layanan berbasis *grafis* seperti *World Wide Web (WWW)* saat itu masih belum ada, yang ada hanyalah layanan yang disebut *Gopher* yang dalam beberapa hal mirip seperti *web* yang kita kenal saat ini, kecuali sistem kerjanya yang masih berbasis teks. Kemajuan berarti dicapai pada tahun 1990 ketika *World Wide Web* mulai dikembangkan oleh *CERN* (Laboratorium Fisika Partikel di Swiss) berdasarkan proposal yang dibuat oleh Tim *Berners-Lee*. Namun demikian, *WWW browser* yang pertama baru lahir dua tahun kemudian, tepatnya pada tahun 1992 dengan nama *Viola*. *Viola* diluncurkan oleh Pei Wei dan didistribusikan bersama *CERN WWW*. Tentu saja *web browser* yang pertama ini masih sangat sederhana, tidak secanggih browser modern yang kita gunakan saat ini.

Terobosan berarti lainnya terjadi pada 1993 ketika *InterNIC* didirikan untuk menjalankan layanan pendaftaran *domain*. Bersamaan dengan itu, Gedung Putih (*White House*) mulai *online* di Internet dan pemerintah Amerika Serikat meloloskan *National Information Infrastructure Act*. Penggunaan internet secara komersial dimulai pada 1994 dipelopori oleh perusahaan *Pizza Hut*, dan *Internet Banking* pertama kali diaplikasikan oleh *First Virtual*. Setahun kemudian, *CompuServe*, *America Online*, dan *Prodigy* mulai memberikan layanan akses ke Internet bagi masyarakat umum. Sementara itu, kita di Indonesia baru bisa menikmati layanan Internet komersial pada sekitar tahun 1994. Sebelumnya, beberapa perguruan tinggi seperti Universitas Indonesia telah terlebih dahulu tersambung dengan jaringan internet melalui *gateway* yang menghubungkan universitas dengan *network* di luar negeri.

2.8 Perangkat Lunak Penunjang

Bahasa pemrograman yang menunjang dalam pembuatan *website* antara lain sebagai berikut:

c) Bahasa Pemrograman HTML (*hypertext Markup Language*)

HTML adalah suatu format data yang digunakan untuk membuat dokumen *hypertext* yang dapat dibaca dari satu *platform* ke *platform* lainnya tanpa melakukan suatu perubahan apapun. Dokumen HTML sebenarnya adalah suatu dokumen teks biasa, sehingga di *platform* apapun dokumen tersebut dapat dibaca. Diawal tahun 80-an, ketika IBM masih menjadi penentu standar utama dunia komputer dan sistem informasi, IBM memiliki suatu ide yang cenderung tersebut berdasarkan pada pengetahuan bahwa setiap dokumen mempunyai ciri-ciri umum, seperti judul, alamat, isi dan elemen-elemen lain yang sangat mirip dari satu dokumen ke dokumen lain. IBM kemudian mengembangkan suatu jenis bahasa yang menggunakan tanda-tanda sebagai basisnya. IBM menambahkan sistem ini dengan *Markup Language*. IBM menamakan sistem ini dengan *Generalizes Markup Language (GML)*. Untuk membuat aplikasi *web*, dalam hal ini menggunakan HTML, maka kita membutuhkan suatu *editor* guna mengetikkan, mengedit atau menyimpan dokumen-dokumen HTML. *Editor* untuk mendesain suatu *web* dibagi 2 yaitu yang bersifat teks murni dan *WYSIWYG (Graphic)*. *Editor* untuk teks, antara lain *Notepad* dan *Ultra Edit*. Sedang editor *WYSIWYG* adalah *Netscape* dan *Front Page*, dan *Macromedia Dreamweaver*.

d) Bahasa pemrograman *PHP*

PHP adalah bahasa *server - side scripting* yang menyatu dengan *HTML* untuk membuat halaman *web* yang dinamis. Maksud dari *server - side scripting* adalah *sintaks* dan perintah-perintah yang diberikan akan sepenuhnya dijalankan di *server* tetapi disertakan pada dokumen *HTML*. Pembuatan *web* ini merupakan kombinasi antara *PHP* sendiri sebagai bahasa pemrograman dan *HTML* sebagai pembangun halaman *web*. Ketika seorang pengguna internet akan membuka suatu *situs* yang menggunakan fasilitas *server - side scripting PHP*, maka terlebih

dahulu *server* yang bersangkutan akan memproses semua perintah PHP di *server* lalu mengirimkan hasilnya dalam *format* HTML ke web *browser* pengguna internet tadi. Dengan demikian seorang pengguna internet tidak dapat melihat kode program yang ditulis dalam PHP sehingga keamanan dari halaman *web* menjadi lebih terjamin. Kode-kode bahasa PHP dalam penulisannya menyatu dengan *tag-tag HTML* dalam suatu *file* kode PHP diletakkan antara tanda `<? atau <? Php` dan diakhiri dengan tanda `?>`, berikut ini contoh penulisan kode PHP yang menyatu dengan *tag HTML*:

“Coba1.PHP

<HTML>

<HEAD>

<TITLE> Program PHP </TITLE>

</HEAD>

<BODY>

<? Echo “ mari belajar PHP”?>

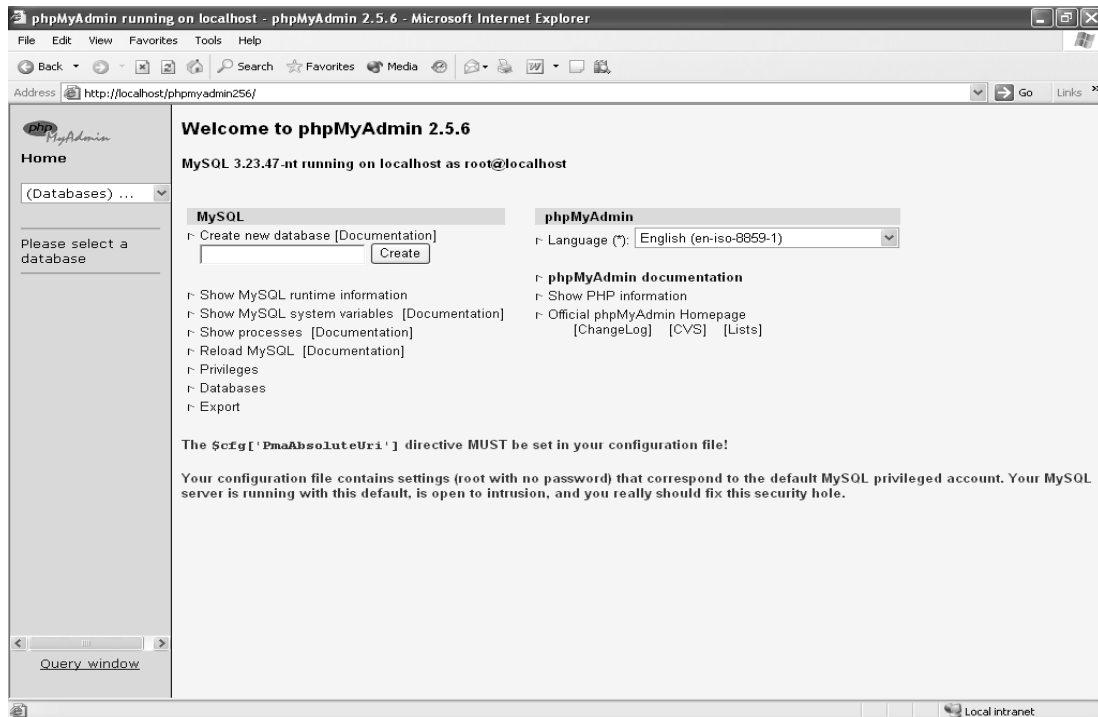
</BODY>

</HTML>.

e) Database MySQL

MySQL adalah *multi user database* yang menggunakan bahasa *Structured Query Language (SQL)*. MySQL dalam operasi *client - server* melibatkan *server daemon MySQL* di sisi *server* dan berbagai macam program serta *library* yang berjalan di sisi *client*. MySQL mampu menangani data yang cukup besar. Perusahaan yang mengembangkan MySQL yaitu TcX, mengaku menyimpan data lebih dari 40 *database*, 10.000 tabel dan sekitar 7 juta baris, totalnya kurang lebih 100 *Gigabyte data*. SQL adalah bahasa standar yang digunakan untuk mengakses *database server*. Bahasa ini pada awalnya dikembangkan oleh IBM, namun telah diadopsi dan digunakan sebagai standard industri. Dengan menggunakan SQL, proses akses *database* menjadi lebih *user – friendly*. dibandingkan dengan menggunakan *dBASE* atau *Clipper* yang masih menggunakan perintah-perintah pemrograman. MySQL merupakan *software database* yang paling populer di

lingkungan *Linux*, kepopuleran ini karena ditunjang performansi *query* dari *databasenya* yang saat ini bisa dikatakan paling cepat dan jarang bermasalah. MySQL ini juga sudah dapat berjalan pada lingkungan *Windows*.



Gambar 2.3 tampilan DBMS MySql

Menurut Didik Dwi Prasetyo (2003) keistimewaan MySQL adalah :

1. *Portability*

MySQL dapat berjalan stabil pada berbagai sistem operasi diantaranya seperti *windows, linux, free BDS* dan lain-lain.

2. *Open Source*

MySQL didistribusikan secara gratis (*open source*).

3. *Multi User*

MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

4. *Performance Tuning*

MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL.

5. *Column Types*

MySQL memiliki tipe kolom yang sangat kompleks.

6. *Command dan Functions*

MySQL memiliki operator dan fungsi secara penuh mendukung perintah *SELECT* dan *WHERE* dalam *query*.

7. *Security*

MySQL memiliki beberapa lapisan sekuritas seperti *level subnetmark*, nama *host* dan izin akses *user* dengan sistem perizinan yang mendetail secara *password terenkripsi*.

8. *Scalability dan Limits*

MySQL mampu menangani *database* dalam skala besar, dengan jumlah *record* lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada setiap tabelnya.

9. *Connectivity*

MySQL dapat melakukan koneksi dengan *client* menggunakan protokol TCP/IP, *Unix Scket (UNIX)*, atau *Named Tipes (NP)*.

10. *Localisation*

MySQL dapat mendeteksi pesan kesalahan (*error code*) pada *client* dengan menggunakan lebih dari dua puluh bahasa.

11. *Inteface*

MySQL memiliki *interface* (antar muka) terdapat berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi APL (*Aplication Programing Interface*).

12. *Client dan Tools*

MySQL dilengkapi dengan berbagai *tool* yang dapat digunakan untuk *administrasi database* dan pada setiap *tool* yang ada disertakan petunjuk *online*.

13. *Struktur Table*

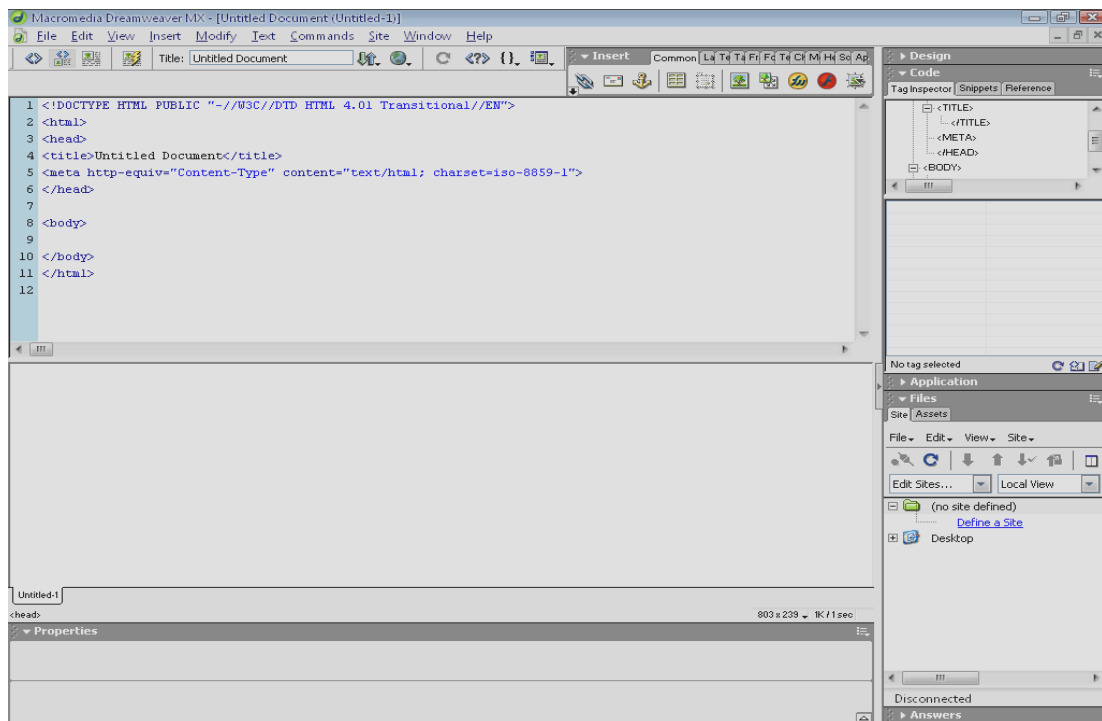
MySQL memiliki struktur table yang lebih fleksibel dalam mengenai *ALTER TABLE*, dibandingkan *database* lainnya, semacam *Postgre SQL* ataupun *Oracle*.

f) Xampp

Xampp adalah *software* yang digunakan sebagai *web server*. Dengan adanya *software* tersebut atau menginstal Xampp kedalam komputer, maka komputer akan berfungsi sebagai *web server*.

g) *Dreamweaver*

Dreamweaver merupakan program paket macromedia yang berisi *flash*, *couldfusion*, *firework* dan *freehand*. *Editor* ini digunakan untuk merancang dan membuat halaman *web* atau *website* yang dapat digunakan oleh orang yang tidak cukup mengerti pemrograman teknis, (Nugroho, 2004). Kemudian lainnya adalah aplikasi ini memudahkan pengguna untuk membuat hal *web* tanpa harus harus menuliskan semua *script* dari tampilan, teks, gambar, animasi, *database*, *audio*, video dan objek lainnya. Hal ini disebabkan terdapatnya *tools*, *properties* atau *panel* untuk membuat objek pada halaman *web*.



Gambar 2.4 Editor HTML Macromedia Dreamweaver