

BAB II

LANDASAN TEORI

2.1 E-Document

E-Document atau Dokumen elektronik yang artinya segala bentuk file digital yang hanya terdapat dalam bentuk atau format elektronik seperti data yang disimpan di komputer, jaringan (*network*), *back-up* ke dalam *Compact Disk* (CD) ataupun DVD, atau media penyimpanan lainnya. Contoh bentuk dokumennya bisa berupa: email, voice mail, instans messages (IM), *e-calender*, *audio-file*, *video*, data-data hasil transfer *handphone*, animasi, grafik, foto, presentasi, *web pages*, dan data digital lainnya.

Dokumen eletronik dibuat dan diperoleh dengan berbagai cara:

1. Membuat secara langsung dokumen elektronik melalui benrbagai aplikasi komputer yang menghasilka keluaran (*output*) dalam bentuk digital, misalnya aplikasi pengolahan huruf, angka, gambar grafik, dan lainnya
2. Memindahkan dokumen, dapat berupa:
 - a) Pemindaian (*Scanning*),
 - b) Konversi File (*conversion*),
 - c) *Inporting* data dari berbagai format

Berikut ini adalah Pengelolaan *E-Document*(*Document management*)

1. Memasukkan Dokumen (*Input*), prosesnya yaitu:
 - a) Menentukan Katagori dokumen.
 - b) Penentuan kata tangkap (*caption*).
 - c) Penentuan sifat dokumen
 - d) Penentuan masa simpan
 - e) Unggah File
2. Pencarian Dokumen

Merupakan fitur yang digunakan untuk melakukan pencarian dokumen elektronik sesuai keinginan pengguna, dengan melakukan pegetikan satu atau beberapa kata tangkap.

Setelah dokumen yang dicari ditemukan, kemudian dokumen tersebut bisa didistribusikan sesuai kebutuhan, yaitu: membaca, cetak, simpan, dan kirim.

2.2 Sistem

Dalam kaitannya dengan sistem informasi, sistem adalah sekelompok komponen yang terkait, yang bekerja sama untuk suatu tujuan umum dengan menerima masukan 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).

2.3 Subsistem

Suatu sistem dapat didefinisikan sebagai satu kesatuan yang terdiri dari beberapa komponen atau subsistem yang berinteraksi untuk mencapai tujuan. Contohnya sistem komputer dapat terdiri dari subsistem perangkat keras dan subsistem perangkat lunak, yang mana masing-masing subsistem saling keterkaitan satu sama lain.

2.4 Perancangan Sistem

Perancangan sistem adalah strategi untuk memecahkan masalah dan mengembangkan solusi terbaik bagi permasalahan itu. Perancangan sistem adalah termasuk bagaimana mengorganisasi sistem ke dalam subsistem-subsistem kekomponen-komponen perangkat keras, perangkat lunak, serta prosedur-prosedur.

2.5 Karakteristik Sistem

(Tata Sutabri. 2012) suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu yaitu :

- a. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan. komponen-komponen sistem tersebut dapat berupa suatu subsistem.

b. Batasan Sistem

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisahkan.

c. Lingkungan Luar sistem

Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem. lingkungan luar sistem ini dapat menguntungkan dan dapat juga merugikan sistem tersebut.

d. Penghubung Sistem

Media yang menghubungkan sistem dengan subsistem yang lain disebut penghubung sistem atau *interface*. penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lain.

e. Masukan Sistem

Masukan adalah energi yang dimasukkan ke dalam suatu sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

f. Keluaran Sistem

Hasil dari energi atau masukan yang diolah dan diklasifikasikan menjadi keluaran yang berguna. keluaran ini merupakan masukan bagi subsistem yang lain.

g. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

h. Sasaran Sistem

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik. kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya.

2.6 Konsep Dasar Informasi

informasi merupakan proses lebih lanjut dari data yang sudah memiliki nilai tambah. Informasi dapat dikelompokkan menjadi 3 bagian, yaitu :

a. Informasi Strategis

informasi ini digunakan untuk mengambil keputusan jangka panjang, yang mencakup informasi eksternal, rencana perluasan perusahaan, dan sebagainya.

b. Informasi Taktis

informasi ini dibutuhkan untuk mengambil keputusan jangka menengah, seperti informasi tren penjualan yang dapat dimanfaatkan untuk menyusun rencana penjualan.

c. Informasi Teknis

informasi ini dibutuhkan untuk keperluan operasional sehari-hari, seperti informasi persediaan stock, retur penjualan, dan laporan kas harian.

2.7 Komponen Sistem Informasi

Terdapat 6 (enam) komponen dari sebuah sistem informasi. Sebagai suatu sistem ke enam blok tersebut masing-masing saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sasaran. Blok-blok tersebut adalah:

a. Blok Masukan (*InputBlock*)

Input mewakili data yang masuk ke dalam sistem informasi. *Input* disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

b. Blok Model (*ModelBlock*)

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

c. Blok Keluaran (*OutputBlock*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

d. Blok Teknologi (*TechnologyBlock*)

Teknologi merupakan “kotak alat” (*tool box*) dalam sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan.

e. Blok Basis data (*DatabaseBlock*)

Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

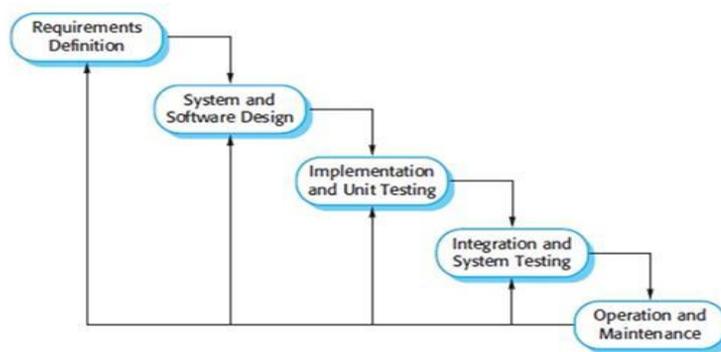
f. Blok Kendali (*ControlsBlock*)

Banyak hal yang dapat merusak sistem informasi. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

2.8 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)

2.8.1 *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

2.8.2 *System and Software Design*

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

2.8.3 *Implementation and Unit Testing*

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

2.8.4 *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

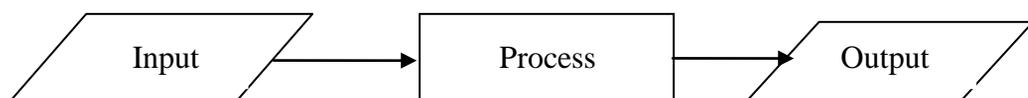
mengintegrasikan sistem lama yang konvensional menjadi terkomputerisasi.

2.8.5 *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.

2.9 Sistem Input-Output

Tiga komponen dasar sistem / subsistem adalah input, proses dan output. Berikut ini adalah gambaran secara umum mengenai input, proses, dan output pada sebuah sistem.

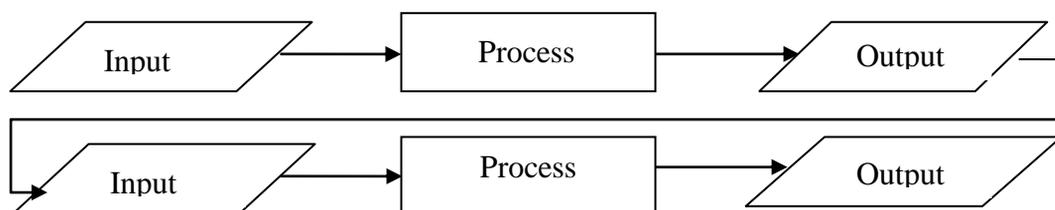


Gambar 2.2 Sistem Input Output

Input: merupakan segala sesuatu yang masuk ke dalam suatu system. Input dapat diklasifikasikan ke dalam tiga kategori yaitu:

a. Serial Input

Serial input merupakan input yang diperoleh sebagai hasil atau output sistem sebelumnya.



Gambar 2.3 Serial Input

Probable input merupakan potensial input yang dapat digunakan oleh suatu sistem Probable Input.

b. Feedback

Feedbaack adalah data atau informasi tentang kinerja (output dari suatu sistem) Feedback input merupakan input jenis ketiga, input ini merupakan bagian ouput dari sistem yang sama yang digunakan sebagai kontrol.

Feedback input dibagi menjadi dua kelompok, yaitu: Negatif feedback input dan Positif feedback input. Positif feedback input digunakan sebagai alat kontrol untuk memperbesar, sedangkan Negatif feedback input digunakan sebagai alat kontrol untuk memperkecil.

c. Pengendalian

Pengendalian meliputi monitoring (pengawasan) dan pengevaluasian untuk menentukan apakah sistem bekerja menuju pencapaian tujuan yang telah ditentukan.

Proses merupakan perubahan dari input menjadi output.

Output merupakan hasil dari suatu proses yang merupakan tujuan dari keberadaan sistem.

2.10 Sistem Informasi

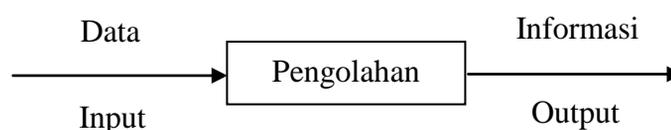
”Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan oleh pihak luar tertentu” Tata Sutabri (2012).

Dari pengertian di atas dapat ditarik kesimpulan bahwa sistem informasi adalah suatu sistem yang terdapat pada sebuah organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan merupakan kegiatan strategi dari suatu organisasi serta tersedianya pihak luar tertentu dengan laporan-laporan yang diperlukan.

2.11 Pengolahan Data

Pengolahan data adalah serangkaian operasi atas informasi yang direncanakan guna mencapai tujuan atau hasil yang digunakan dalam menangani informasi.

Proses Pengolahan Data



Gambar 2.4 Proses Pengolahan Data

Proses pengolahan data dalam hal ini dapat berjalan apabila terdapat *input* berupa data, data kemudian diolah sehingga menghasilkan suatu *output* berupa informasi. Pengolahan data dapat dirincikan ke dalam proses berikut:

- a. Proses konversi dari fakta yang terjadi dalam sekitar atau sekeliling, menjadi pengetahuan yang dimengerti (*Data Capturing*);
- b. Proses perekam dan pencatatan pengetahuan simbol-simbol yang disepakati (*Data Recording*);
- c. Proses pemeriksaan kesesuaian antara data dengan fakta atau data sebelum proses pencatatan (*Data Verifying*);
- d. Proses pemisahan data menurut jenisnya (*Data Classifying*);
- e. Proses pengurutan data (*Data Sorting*);
- f. Proses pembuatan duplikat data (*Data Copying*);
- g. Proses pembacaan kembali data dari tempat rekaman (*Data Retrieving*);
- h. Proses penyampaian dan penyebaran data kepada yang memerlukan (*Data Communication* atau *Disseminating*);
- i. Proses penggabungan data satu dengan data lainnya yang berkaitan;
- j. Proses manipulasi data, baik manipulasi aritmatik berupa penjumlahan, pengurangan, perkalian, maupun manipulasi *logic* seperti perbandingan, *and*, *or* dan sebagainya (*Data Manipulating*).

Pengolahan data merupakan proses pengumpulan data mentah berupa dokumen yang kemudian di-*input*-kan ke dalam sistem kerja komputer untuk diolah sehingga menghasilkan informasi yang berguna. Pengolahan data untuk mencapai tujuan informasi yang diinginkan.

2.12 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 ke dalam 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 *webbrowser* 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.

2.13 Perangkat Lunak Penunjang

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

a) 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. Di awal 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 *Generalized 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*.

b) 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*:

```

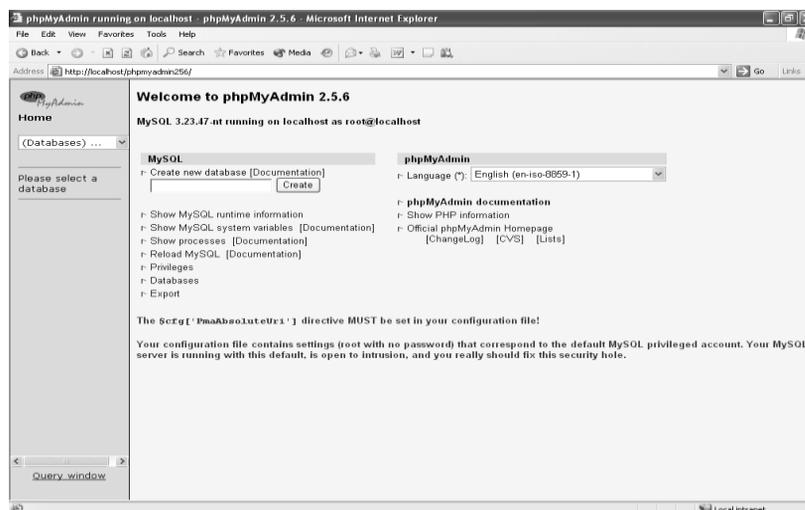
“ Coba1.PHP
<HTML>
<HEAD>
<TITLE> Program PHP </TITLE>
</HEAD>
<BODY>
<? Echo “ mari belajar PHP”?>
</BODY>
</HTML>.

```

c) 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 *databasesnya* yang saat ini bisa dikatakan paling cepat dan jarang bermasalah. MySQL ini juga sudah dapat berjalan pada lingkungan *Windows*.



Gambar 2.5 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 Pipes (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

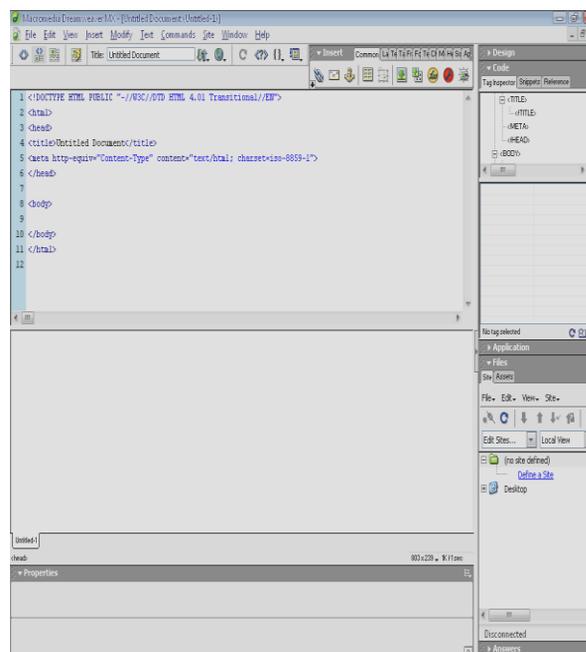
MySQL memiliki struktur table yang lebih fleksibel dalam mengenai *ALTER TABLE*, dibandingkan *database* lainnya, semacam *Postgre SQL* ataupun *Oracle*.

d) 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*.

e) 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*.

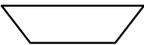
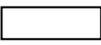
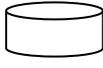
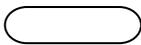
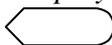
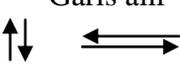


Gambar 2.6 Editor HTML Macromedia Dreamweaver

2.14 Bagan Alir Dokumen

Bagan alir dokumen (*Document flowchart*) merupakan bagan alir yang menunjukkan arus data dari laporan dan formulir termasuk tembusan-tembusannya.

Tabel 2.1 Simbol-simbol Bagan Alir Dokumen

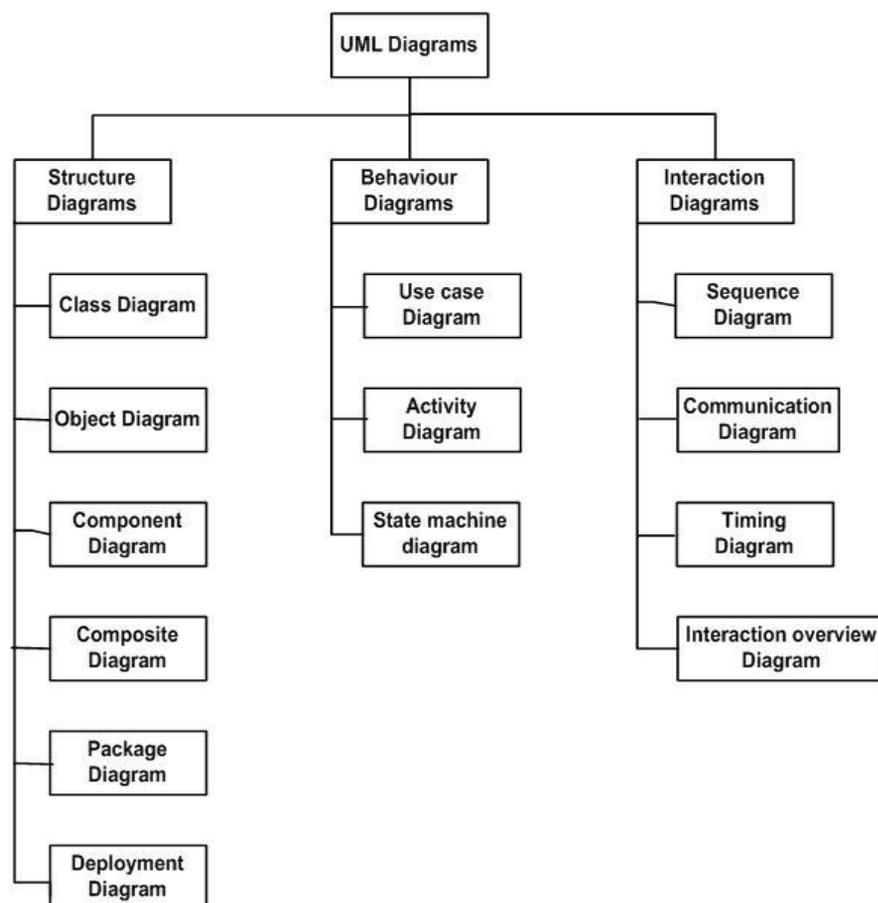
Simbol	Keterangan
Dokumen 	Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan
Kegiatan manual 	Menunjukkan pekerjaan yang dilakukan tidak dengan program komputer.
Simpanan / arsip 	Menunjukkan dokumen yang diarsipkan (arsip manual)
Proses 	Menunjukkan kegiatan proses yang dilakukan dengan program komputer
<i>Hard Disk</i> 	Menunjukkan <i>input / output</i> menggunakan <i>hard disk</i>
<i>Terminasi</i> 	Menunjukkan awal atau akhir dari suatu proses.
Keputusan 	Menunjukkan suatu penyeleksian kondisi di dalam program.
<i>Display</i> 	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

2.15 UML (*Unified Modeling Language*)

Menurut Rosa A.S. dan M. Salahuddin dan M. Salahuddin (2011:137) mendefinisikan UML:

“UML (*Unified Modeling Language*) adalah standrisasi bahasa permodelan untuk membangun pearangkat lunak yang dibangun dengan menggunakan teknik pengembangan berorientasi objek, yang muncul karena adanya kebutuhan permodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak.”

Pembagian kategori dan macam-macam diagram UML dapat dilihat pada gambar di bawah ini:



Sumber : Rosa A.S. dan M. Salahuddin dan M. Salahuddin (2011:140)

Gambar 2.7 Diagram UML

Berikut ini penjelasan singkat dari pembagian kategori *Diagram* UML:

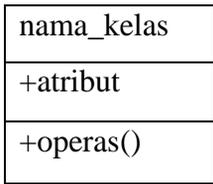
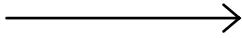
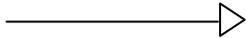
- a. *Structure diagrams* yaitu kumpulan *diagram* yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
- b. *Behavior diagrams* yaitu kumpulan *diagram* yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
- c. *Interaction diagrams* yaitu kumpulan *diagram* yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

Diagram-diagram yang digunakan dalam UML berupa *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*. *Diagram* yang merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur karena dapat menggambarkan arus data di dalam sistem dengan terstruktur dan jelas, juga merupakan dokumentasi yang baik. Tujuan menggunakan diagram-diagram UML ini adalah memberikan pandangan umum sistem, memperlihatkan sebuah proses yang berinteraksi dengan lingkungannya. empat diagram yang digunakan, yaitu terdiri dari:

2.14.1. Class Diagram

Pemodelan dari *structure diagram*. Menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

Tabel 2.2 Simbol - Simbol *Class Diagram*

No	Simbol	Deskripsi
1.	<p>Kelas</p> 	Kelas pada struktur sistem
2.	<p>Antarmuka/ <i>interface</i></p> 	Sama dengan konsep interface dalam pemrograman berorientas objek.
3.	<p>Asosiasi / <i>association</i></p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya disertai dengan <i>multiplicity</i> .
4.	<p>Asosiasi berarah/ <i>directed association</i></p> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	<p>Generalisasi / <i>generalization</i></p> 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus).
6.	<p>Kebergantungan / <i>dependency</i></p> 	Relasi antar kelas dengan makna kebergantungan antar kelas.
7.	<p>Agresiasi / <i>aggregation</i></p> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>)

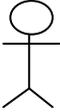
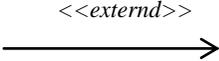
Sumber : Rosa A.S. dan M. Salahuddin (2013:146)

2.14.2. Use case diagram

Pemodelan dari *behavior diagrams* untuk kelakuan sistem informasi yang akan dibuat. Mendeskripsikan sebuah interaksi antara satu atau lebih aktor

dengan sistem informasi yang akan dibuat. Simbol-simbol yang digunakan untuk pembuatan *use case diagram* dapat dilihat pada tabel sebagai berikut :

Tabel 2.3 Simbol - Simbol *Use Case Diagram*

No	Simbol	Keterangan
1.	<i>use case</i> 	Fungsionalitas yang disediakan system sebagai unit-unit atau actor.
2.	Aktor 	seseorang/sesuatu yang berinteraksi dengan sistem yang sedang kita kembangkan.
3.	Asosiasi/ <i>Association</i> 	komunikasi antara actor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan actor.
4.	Ekstensi / <i>Extends</i> 	Relasi usecase tambaha ke sebuah usecase dimaa usecase yang ditambahkan dapat berdiri walau tapa usecase tambahan itu
5.	Generalisasi 	Hubunga generalisasi dan spesifikasi (umum-khusus) antara dua buah usecase
6.	Menggunakan / <i>include/ uses</i> 	<i>Include</i> memungkinkan suatu <i>usecase</i> untuk menggunakan fungsionalitas yang disediakan oleh <i>usecase</i> yang lainnya

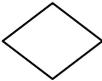
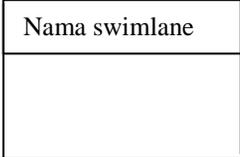
Sumber : Rosa A.S. dan M. Salahuddin (2011:156)

2.14.3. Activity Diagram

Teknik untuk menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.

Simbol-simbol yang digunakan untuk pembuatan *activity diagram* dapat dilihat pada tabel sebagai berikut:

Tabel 2.4 Simbol - Simbol *Activity Diagram*

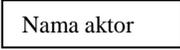
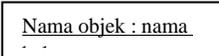
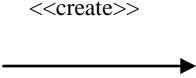
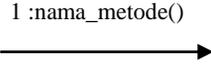
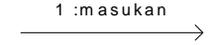
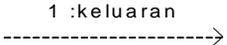
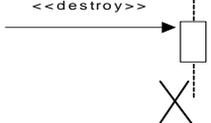
No	Simbol	Keterangan
1	Status Awal 	Status Awal aktivitas system
2	Aktivitas 	Aktivitas yang dilakukan system, biasanya diawali dengan kata kerja
3	Percabangan/decision 	asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
4	Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
5	Status Akhir 	Status Akhir yang dilakukan system
6	Swimlane 	Memisahkan organisasi yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber: Rosa A.S. dan M. Salahuddin (2011:162)

2.14.4. *Sequence Diagram*

Menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan untuk pembuatan *sequence diagram* dapat dilihat pada tabel sebagai berikut :

Tabel 2.5 Simbol - Simbol *Sequence Diagram*

No	Simbol	Keterangan
1	Aktor  nama actor atau 	Orang, proses, atau system lain yang berinteraksi dengan system informasi yang akan dibuat di luar system informasi yang akan dibuat itu sendiri.
2	Garis hidup/ <i>lifeline</i> 	menyatakan kehidupan suatu objek.
3	Objek 	menyatakan objek yang berinteraksi pesan
4	Waktu aktif 	menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
5	Pesan tipe <i>create</i> 	menyatakan objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
6	Pesan tipe <i>call</i> 	menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,
7	Pesan tipe <i>send</i> 	menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya.
8	Pesan tipe <i>return</i> 	menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau menghasilkan suatu kembalian ke objek tertentu.
9	Pesan tipe <i>destroy</i> 	menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

Sumber: Rosa A.S. dan M. Salahuddin (2011:165)