BAB II

LANDASAN TEORI

2.1. Pengertian Sistem

Terdapat dua kelompok pendekatan dalam mendefinisikan sistem, yaitu yang menekankan kepada prosedur dan yang menekankan kepada komponen atau elemen. Pendekatan sistem yang lebih menekankan pada prosedur di definisikan sebagai berikut :

Menurut Jogiyanto, H.M. (2001 : 1) *"Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran yang tertentu."*

Sedangkan pendekatan sistem yang lebih menekankan pada komponen atau elemen-elemen di definisikan sebagai berikut :

Menurut Jogiyanto, H.M. (2001 : 1) *"Sistem adalah elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu."*

2.1.1. Karakteristik Sistem

Sistem mempunyai beberapa karakteristik atau sifat tertentu, antara lain :

a. Komponen sistem (*Component*) : Suatu sistem terdiri dari komponen yang saling berinteraksi, dan bekerja sama membentuk suatu komponen sistem.

b. Batasan sistem (*Boundary*) : Merupakan daerah yang membatasi suatu sistem dengan sistem yang lain atau dengan lingkungan kerjanya.

c. Sub sistem (*Sub system*) : Bagian-bagian dari sistem yang beraktivitas dan berinteraksi satu sama lain untuk mencapai tujuan masing-masing.

d. Lingkungan luar sistem (*Environment*) : Suatu sistem yang ada di luar dari batas sistem yang dipengaruhi oleh operasi sistem.

e. Penghubung sistem (*Interface*) : Media penghubung antara suatu sub sistem dengan sub sistem lain.

f. Masukan sistem (*Input*) : Energi yang masuk ke dalam sistem, berupa perawatan dan sinyal yang di masukkan supaya sistem dapat berinteraksi.

g. Keluaran sistem (*Output*) : Hasil energi yang diolah dan di klasifikasi menjadi keluaran yang berguna dan sisa pembuangan.

h. Pengolahan sistem (*Process*) : Suatu sistem dapat mempunyai suatu bagian pengolahan yang akan mengubah masukan menjadi keluaran.

i. Sasaran sistem (*Object*) : Tujuan yang ingin dicapai oleh sistem, akan dikatakan berhasil apabila mengenai sasaran atau tujuan.

(Sumber : http://led21.blogspot.com//*pengertian-elemen-karakteristik/* 24 Maret 2013)

2.1.2. Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya adalah sebagai berikut :

1. Sistem Abstrak dan Sistem Fisik

Sistem abstrak (*abstract system*) adalah sistem yang berisi gagasan atau konsep. Sedangkan sistem fisik (*physical system*) adalah sistem yang secara fisik dapat dilihat atau sistem yang dapat terlihat.

2. Sistem Deterministik dan Sistem Probabilistik

Sistem deterministik (*deterministic system*) adalah suatu sistem yang operasinya dapat diprediksi secara tepat. Sedangkan sistem probabilistik (*probabilistic system*) adalah sistem yang tidak dapat diramalkan dengan pasti karena mengandung unsur probabilitas.

3. Sistem Tertutup dan Sistem Terbuka

Sistem tertutup (*closed system*) adalah sistem yang tidak bertukar materi, informasi, atau energi dengan lingkungan, dengan kata lain sistem ini tidak berinteraksi dan tidak dipengaruhi oleh lingkungan. Sedangkan sistem terbuka (*open system*) adalah sistem yang berhubungan dengan lingkungan dan dipengaruhi oleh lingkungan.

4. Sistem Alamiah dan Sistem Buatan Manusia

Sistem Alamiah *(natural system*) adalah sistem yang terjadi karena alam. Sedangkan sistem buatan manusia (*human made system*) adalah sistem yang dibuat oleh kecerdasan manusia, misalnya sistem komputer.

5. Sistem Sederhana dan Sistem Kompleks

Berdasarkan tingkat kerumitannya, sistem dibedakan menjadi sistem sederhana dan sistem kompleks.

(Sumber : http://rahmat.blogsot.com/*konsep-dasar-sistem-klasifikasi-sistem/* 24 Maret 2013)

2.1.3. Elemen Sistem

Ada beberapa elemen yang dapat membentuk sebuah sistem, yaitu sebagai berikut :

1. Tujuan

Setiap sistem memiliki tujuan (*Goal*), entah hanya satu atau mungkin banyak. Tujuan inilah yang menjadi pemotivasi yang mengarahkan sistem. Tanpa tujuan, sistem menjadi tak terarah dan tak terkendali.

2. Masukan

Masukan (*input*) sistem adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan yang diproses. Masukan dapat berupa hal-hal yang berwujud (tampak secara fisik) maupun yang tidak tampak (abstrak).

3. Proses

Proses merupakan bagian yang melakukan perubahan atau transformasi dari masukan menjadi keluaran yang berguna dan lebih bernilai, misalnya berupa informasi dan produk.

4. Keluaran

Keluaran (*output*) merupakan hasil dari pemrosesan. Pada sistem informasi, keluaran bisa berupa suatu informasi, saran, cetakan laporan, dan sebagainya.

5. Batas

Yang disebut batas (*boundary*) sistem adalah pemisah antara sistem dan daerah di luar sistem (lingkungan). Batas sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem.

6. Mekanisme Pengendalian dan Umpan Balik

Mekanisme pengendalian (*control mechanism*) diwujudkan dengan menggunakan umpan balik (*feedback*), yang mencuplik keluaran. Umpan balik ini digunakan untuk mengendalikan baik masukan maupun proses. Tujuannya adalah untuk mengatur agar sistem berjalan sesuai dengan tujuan.

7. Lingkungan

Lingkungan adalah segala sesuatu yang berada diluar sistem. Lingkungan bias berpengaruh terhadap operasi sistem dalam arti bisa merugikan atau menguntungkan sistem itu sendiri.

(Sumber : http://rahmat.blogsot.com/*konsep-dasar-sistem-elemen-sistem/* 24 Maret 2013)

2.2. Pengertian Informasi

Informasi sangat penting bagi setiap perusahaan ataupun badan institusi lainnya, selain itu informasi juga penting bagi sistem. Karena apabila suatu sistem tidak mempunyai informasi maka akan menjadi suatu sistem yang susah berkembang. Menurut Jogiyanto, H.M. (2001 : 8) informasi adalah *"Data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya."*

Sedangkan menurut Witarto (2004 : 9) informasi adalah *"Rangkaian data yang mempunyai sifat sementara, tergantung dengan waktu, mampu memberi kejutan atau suprise pada yang menerimanya."*

Perancangan suatu program aplikasi terdiri dari satu kesatuan sistem. Informasi yang baik adalah informasi yang berkualitas ditentukan oleh hal-hal sebagai beikut :

a. Akurat

   Informasi harus bebas dari kesalahan dan tidak menyesatkan.

b. Tepat pada waktunya.

   Informasi yang datang pada penerima tidak boleh terlambat.

c. Relevan

   Informasi tersebut mempunyai manfaat/hubungan bagi pemakainya.

d. Dapat dipercaya

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

2.3. Pengertian Sistem Informasi

Menurut Drs. Zulkifli Amsyah, MLS. (2005 : 4) *"Sistem informasi, yaitu suatu rangkaian informasi yang di dalamnya terdapat bagian-bagian yang berhubungan dan saling berketergantungan satu sama lain, mulai dari bagian besar ke bagian yang lebih kecil, yaitu dari sub, subsub, subsubsub, dan seterusnya sampai yang terkecil".*

Dari pengertian diatas dapat disimpulkan bahwa sistem informasi adalah kumpulan komponen pembentuk sistem yang mempunyai keterkaitan antara satu komponen dengan komponen lainnya yang bertujuan menghasilkan suatu informasi dalam suatu bidang tertentu.

2.3.1. Komponen Sistem Informasi

Menurut Abdul Kadir (2003 : 70) komponen-komponen dalam suatu sistem informasi adalah sebagai berikut :

1. Perangkat Keras (*Hardware*)

   Mencakup peranti-peranti fisik seperti komputer dan printer.

2. Perangkat Lunak (*Software atau Program*)

   Sekumpulan instruksi yang memungkinkan perangkat keras untuk dapat memproses data.

3. Prosedur

   Sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data dan pembangkitan keluaran yang dikehendaki.

4. Basia Data (*Database*)

   Sekumpulan tabel, hubungan, yang berkaitan dengan penyimpanan data.

5. Pemakai (*User*)

   Semua pihak yang bertanggung jawab dalam pengembangan sistem informasi, pemrosesan, dan penggunaan keluaran sistem informasi.

6. Jaringan Komputer dan Komunikasi Data

   Sistem penghubung yang memungkinkan sesumber (*resources*) dipakai secara bersama atau diakses oleh sejumlah pemakai.

2.4. Arsitektur Aplikasi

Arsitektur aplikasi diartikan sebagai suatu perancangan dalam membuat aplikasi sehingga dapat mendukung terbentuknya suatu sistem informasi berupa jaringan komputer dengan berbagai jenis topologi didalamnya dan menghasilkan sesuatu yang bermanfaat.

2.4.1. Pengertian Jaringan Komputer

Jaringan komputer adalah sebuah kumpulan komputer, printer dan peralatan keras lainnya yang saling terhubung dalam satu kesatuan. Informasi dan data dapat bergerak melalui kabel-kabel atau tanpa kabel sehingga memungkinkan para pengguna jaringan komputer dapat saling bertukar informasi, data dan sebuah dokumen, serta dapat mencetak pada printer yang sama ataupun dapat secara bersama-sama menggunakan suatu

perangkat keras/perangkat lunak yang terhubung dengan jaringan. Setiap komputer, printer atau periferal yang terhubung dengan jaringan disebut *node*.

2.4.2. Jenis – Jenis Jaringan Komputer

Secara umum jaringan komputer dibagi atas lima jenis, yaitu ;

1. *Local Area Network (LAN)*

Local Area Network (LAN), merupakan jaringan milik pribadi di dalam sebuah gedung atau kampus yang berukuran sampai beberapa kilometer. *LAN* seringkali digunakan untuk menghubungkan komputer-komputer pribadi dan *workstation* dalam kantor suatu perusahaan atau pabrik-pabrik untuk memakai bersama sumberdaya (*resouce*, misalnya printer) dan saling bertukar informasi.

2. *Metropolitan Area Network (MAN*)

Metropolitan Area Network (MAN), pada dasarnya merupakan versi *LAN* yang berukuran lebih besar dan biasanya menggunakan teknologi yang sama dengan *LAN. MAN* dapat mencakup kantor-kantor perusahaan yang letaknya berdekatan atau juga sebuah kota dan dapat dimanfaatkan untuk keperluan pribadi (*swasta*) atau umum. *MAN* mampu menunjang data dan suara, bahkan dapat berhubungan dengan jaringan televisi kabel.

3. *Wide Area Network (WAN)*

Wide Area Network (WAN), jangkauannya mencakup daerah geografis yang luas, seringkali mencakup sebuah negara bahkan benua. *WAN* terdiri dari kumpulan mesin-mesin yang bertujuan untuk menjalankan program-program (*application*) pemakai.

4. *Internet*

Sebenarnya terdapat banyak jaringan didunia ini, seringkali menggunakan perangkat keras dan perangkat lunak yang berbeda-beda. Orang yang terhubung pada jaringan sering berharap untuk bisa berkomunikasi dengan orang lain yang terhubung ke jaringan lainnya. Keinginan seperti ini memerlukan hubungan antar jaringan yang seringkali tidak kompatibel dan berbeda. Biasanya untuk melakukan hal ini diperlukan sebuah mesin yang disebut *gateway* guna melakukan hubungan dan melaksanakan terjemahan yang diperlukan, baik perangkat keras maupun perangkat lunaknya. Kumpulan jaringan yang terinterkoneksi inilah yang disebut dengan internet.

2.5. Kasus Yang Dirancang

Pada kasus dalam penelitian ini penulis akan membutuhkan penjelasan singakat tentang definisi atau pengertian dari dokter praktek, pelayanan, pasien, dan rekam medis.

2.5.1. Pengertian Dokter Praktek

Sistem informasi pelayanan kesehatan yang dilakukan oleh dokter umum atau spesialis secara mandiri pada suatu tempat sarana kesehatan yang telah memperoleh ijin yang berwenang. Kegiatan pokok pelayanan kesehatan dokter praktek meliputi:

a. Upaya preventif

   Kegiatan pelayanan kesehatan sebagai upaya pencegahan penyakit terhadap serangan penyakit dengan cara pemberian imunisasi, vaksinasi, KB .

b. Upaya promotif

   Kegiatan pelayanan kesehatan sebagai upaya peningkatan kesadaran kesehatan pasien terhadap suatu resiko dengan cara penyuluhan kesehatan pada pasien.

c. Upaya kuratif

   Kegiatan pelayanan kesehatan sebagai upaya penyumbuhan dan pengobatan terhadap penyakit, dengan mendapatkan pengobatan dan tindakan medis.

   (Sumber: http://sitiulfairmayanti.blogspot.com/*Sisteminformasipelayanandokterpraktekdipoliklinikudinus/* 2 April 2013)

2.5.2. Pengertian Pelayanan

Pelayanan adalah suatu kegiatan atau urutan kegiatan dimana terjadinya interaksi langsung antara seseorang dengan orang lain untuk kepuasan pasien/konsumen. Pelayanan selalu ada dalam setiap perusahaan atau instansi pemerintahan yang bergerak dibidang jasa. Seperti halnya pada Klinik Griya Chrysalis Bandung

2.5.3. Pengertian Pasien

Pasien adalah orang sakit yang dirawat oleh seorang dokter. Hak pasien adalah mendapatkan ganti rugi apabila pelayanan yang diterima tidak sebagaimana mestinya. Masyarakat sebagai konsumen dapat menyampaikan keluhannya kepada pihak rumah sakit atau institusi layanan kesehatan lainnya sebagai upaya perbaikan interen dalam pelayanannya atau kepada lembaga yang memberi perhatian kepada konsumen kesehatan.

2.5.4. Pengertian Rekam Medis

Rekam medis adalah berkas ataupun catatan yang berisikan informasi tentang identitas pasien, *anamnesis* (riwayat medis seseorang), penentuan fisik laboratorium, diagnosa segala pelayanan dan tindakan medik yang diberikan kepada pasien dan pengobatan.

Catatan merupakan tulisan-tulisan yang dibuat oleh seorang dokter atau tenaga medis mengenai tindakan-tindakan yang dilakukan kepada pasien dalam rangka palayanan kesehatan.

2.6. Perangkat lunak Pendukung

Adapun beberapa perangkat lunak pendukung dalam pembangunan aplikasi sistem informasi pelayanan medis dan pengelolaan data pasien pada Klinik Griya Chrysalis Bandung ini adalah sebagai berikut :

2.6.1. *Hypertext Markup Language* (HTML)

Dokumen HTML adalah file teks regular (disebut juga ASCII) yang diciptakan dengan menggunakan editor teks (misalnya, *Emacs* dalam *Unix*, *Gnotepad* pada *Linux*, atau *Notepad* pada *Windows*) atau dengan menggunakan *word processor*, tetapi harus disimpan dengan format *"text only with line breaks".*

Sesuai dengan namanya, bahasa ini menggunakan tanda (*MarkUp*) untuk menandai perintah-perintahnya. Saat ini banyak sekali aplikasi-aplikasi yang dapat digunakan untuk membuat *web page*, seperti *Microsof Frontpage, Adobe GoLive, Macromedia Dreamweaver, Ultradev* dan lain-lainnya. Namun untuk seorang *web develover* kemampuan dasar perintah HTML tetap dibutuhkan.

Homepage dibangun oleh sebuah file yang didalamnya terdapat kode-kode dan informasi dan dibaca oleh sebuah aplikasi yang bernama *browser*. Oleh *browser*, kode dan informasi akan disajikan sebagai sebuah halaman *web* yang kita kenal sebagai *homepage*. Kode tersebut haruslah ditulis menurut suatu aturan tertentu yang dapat dimengerti *browser*. Aturan tersebut dikenal dengan "bahasa pengkodean".

Untuk membuat *homepage website*, bahasa pengkodean yang digunakan adalah HTML. Sekalipun banyak orang menyebutnya sebagai salah satau bahasa pemrograman, HTML sebenarnya sama sekali bukan bahasa pemrograman karena HTML merupakan bahasa *MarkUp* (pengkodean) yang digunakan untuk menentukan format atau *style* dan teks yang ditandai. Simbol *MarkUp* yang digunakan oleh HTML ditandai dengan tanda lebih kecil (<) dan lebih besar (>), keduanya dinamai tag. Sebagai contoh akan ditampilkan teks yang tercetak tebal, maka *markup* yang digunakan adalah sebagai berikut:

<B> Teks ini bercetak tebal </B>

Untuk menandai bahwa sebuah file teks merupakan file HTML, maka ciri yang terlihat jelas adalah ekstensi filenya yaitu .htm atau .html. Namun labih jauh dari pada itu didalam file tersebut harus mengandung struktur sebagai berikut:

<HTML>

…….

…….

</HTML>

Tag <HTML> harus diletakan pada bagian awal dan tag </HTML> harus diletakan pada bagian terakhir dari suatu file HTML. Tag-tag HTML tidak bersifat case sensive artinya penggunaan huruf besar maupun huruf kecil tidak menjadi masalah. Jadi tag <HTML> akan sama dengan tag <html>.

2.6.2. JavaScript

JavaScript adalah bahasa script yang digunakan pada halaman HTML. JavaScript adalah varian dari java yang sangat terkenal dalam lingkungan pemrograman web. JavaScript mempunyai keunggulan pada sisi client, maupun diakses lebih cepat dari script server. Tetapi kelemahannya pada script ini belum tentu mampu diproses oleh browser client tergantung kompetibilitas *browser* tersebut. Cara untuk menjalankan Javascript hanya dengan *javascript - enabled* yaitu *browser* yang mampu menjalankan javascript seperti Netscape Navigator (versi 2.0 keatas) atau internet ekplorer (MSIE-versi 3.0 ke atas).

Beberapa hal mengenai JavaScript :

1) JavaScript didesain untuk menambah interaktif suatu web.
2) JavaScript merupakan sebuah bahasa *scripting* dengan bahasa pemrograman yang ringan.
3) JavaScript biasanya disisipkan (*embedded*) dalam halaman HTML.
4) JavaScript adalah bahasa interpreter (yang berarti skrip dieksekusi tanpa proses kompilasi.

Ada beberapa cara untuk menuliskan JavaScript yaitu :

a) Skrip pada bagian head

   Skrip ini dieksekusi ketika dipanggil (biasanya berbentuk function) atau dipanggil berdasarkan event tertentu. Peletakan skrip di head akan menjamin skrip di *load* terlebih dahulu sebelum digunakan.

```
<html>
<head>
<script type=”text/javascript”>
Function message ()
{
Alert (“Alert ini dipanggil saat even dijalankan”);
}
<script>
</head>
</html>
```

b) Skrip pada Body

Skrip akan dieksekusi ketika halaman di-load sampai dibagian <body> dan ketika ditempatkan berarti antara isi dan javacript dijadikan satu bagian.

```
<html>
<head>
</head>
<body>
<script type=”text/javascript”>
Document.write(“Pesan ini akan tampil”);
</script>
</body>
 </html>
```

c) External JavaScript

Untuk memudahkan menulis script yang berulang-ulang di setiap halaman maka JavaScript dapat ditulis di file secara ekternal. Jadi, antara HTML dan JavaScript dipisah, kemudian file tersebut dipanggil dari dokumen HTML. File JavaScript tersebut disimpan dengan ektensi .js.

```
<html>
<head>
</head>
<body>
<script src=”xxx.js”>
</script>
</body>
</html>
```

2.6.3. Sekilas Tentang PHP

PHP merupakan singkatan dari *"PHP: Hypertext Preprocessor"*, adalah sebuah bahasa *scripting* yang terpasang pada *HTML*. Sebagian besar *sintaks* mirip dengan bahasa *C, Java, asp* dan *Perl*, ditambah beberapa fungsi *PHP* yang spesifik.

Tujuan utama bahasa ini adalah untuk memungkinkan perancang *web* untuk menulis halaman *web* dinamik dengan cepat. Kelebihan *PHP* Pada saat ini bahasa *PERL* dan *CGI* sudah jauh ketinggalan jaman sehingga

sebagian besar *designer web* banyak beralih ke bahasa *server-side scripting* yang lebih dinamis seperti *PHP*.

Seluruh aplikasi berbasis *web* dapat dibuat dengan *PHP*. Namun kekuatan yang paling utama *PHP* adalah pada konektivitasnya dengan *system database* di dalam *web*. Sistem *database* yang dapat didukung oleh *PHP* adalah :

1. MySQL

2. Ms. Access

3. Sybase

4. PostgreSQL

PHP dapat berjalan di berbagai system operasi seperti *windows 98/NT, UNIX/LINUX*, solaris maupun *macintosh*. Keunggulan lainnya dari *PHP* adalah *PHP* juga mendukung komunikasi dengan layanan seperti *protocol IMAP, SNMP, NNTP, POP3* bahkan *HTTP*.

2.6.3.1. Dasar-dasar PHP

PHP adalah sebuah bahasa script yang sangat bagus dan cocok untuk digabungkan dengan HTML. Dengan kemampuan PHP untuk membuat website yang dinamis serta keindahan desain menggunakan HTML, maka nyaris tidak ada yang tidak dapat dilakukan gabungan kedua bahasa ini di dunia *web*.

Kelebihannya lagi, PHP ini merupakan bahasa yang sangat mudah, bagi yang pernah mempelajari C atau bahasa pemrograman lain

akan sangat merasakan perbedaaan ini. Contoh salah satu perbedaan yang paling terasa adalah bahwa kita tidak usah repot-repot deklarasi variabel di awal, kalau butuh variabel tinggal pakai. Untuk dapat menjalankan PHP Anda membutuhkan sebuah *web server*. Ada banyak Software Web Server yang dapat anda gunakan untuk menjalankan Script PHP di komputer kita. Contoh *web server* yang dapat kita pakai adalah *Apache,* software ini dapat kita *Download* di internet. Untuk lebih memudahkan *Apache* sudah tersedia satu paket dengan *MySQL* dalam Software *PHPTriad.*

2.6.3.2. Variabel pada PHP

PHP mendukung beberapa jenis Variabel sebagai berikut:

1 *Integer*, Variabel berjenis Integer bertujuan menyimpan bilangan bulat (tidak termasuk bagian pecahan atau pemangkatannya).

2 *Double*, digunakan untuk bilangan berjenis pecahan dan pemangkatannya.

3 *String*, merupakan jenis data berupa karakter yang disimpan sebagai nomor pada memori komputer. Nilai yang disimpan adalah nilai ASCII karakter string tersebut.

4 *Array*, adalah set variabel yang mempunyai jenis data yang sama. Array mengandung komponen yang dinamakan elemen.

5 *Object*, adalah jenis variabel yang berdasarkan gambaran object pada dunia nyata, yang berupa “status” dan juga “tingkah laku”.

Sebuah object menyimpan statusnya pada sebuah Variabel dan tingkah lakunya berupa parameter.

6 *PDFDoc*, (bila dukungan pada PDF diaktifkan).

7 *PDFInfo*, (bila dukungan pada PDF diaktifkan).

2.6.4. WEB Server

Web server merupakan *server internet* yang mampu melayani koneksi transfer data dalam *protocol HTTP*. *Web server* merupakan hal yang terpenting dari server di internet dibandingkan server lainnya *seperti e-mail server, ftp server* ataupun *news server*. Hal ini di sebabkan *web server* telah dirancang untuk dapat melayani beragam jenis data, dari *text* sampai grafis 3 dimensi.

2.6.5. Pembahasan MySQL

Menurut Abdul Kadir (2009 : 15) mengenai MySQL yaitu : "MySQL merupakan software yang tergolong database server dan bersifat *Open Source*."

PHP memang mendukung banyak *database*, tetapi kita mengambil yang lebih umum yaitu MySQL. MySQL dapat menyimpan semua data Website seperti Berita, Artikel, *Counter* dan sebagainya dengan mudah dan terstruktur, dan dapat membukanya kembali dengan mudah dan cepat. Yang paling disukai dari MySQL yaitu Querynya yang simple dan menggunakan

escape character yang sama dengan PHP, selain itu MySQL adalah *database* tercepat saat ini.

2.6.5.1. PhpMyAdmin

PhpMyAdmin adalah tool untuk mengadministrasi/ memanajemen database MySQL berbasis web. PhpMyAdmin mempunyai banyak kelebihan yang dapat mempermudah pekerjaan anda, hal ini sangat membantu apabila anda mempunyai server MySQL dalam komputer.

2.6.5.2. *Database* MYSQL pada Program

SQL (*Strutured Query Language*) adalah bahasa standar yang digunakan untuk memanipulasi dan memperoleh data dalam suatu *database* rasional. SQL dapat membuat seorang Administrator melakukan hal-hal berikut:

1 Mengubah suatu strutur *database*

2 Mengubah pengaturan keamanan *system*

3 Memberikan suatu hak akses kepada pengguna untuk mengakses *database* atau *table*

4 Memperoleh informasi dari database

5 Memutakhirkan isi *database*

Perintah SQL dikelompokan menjadi 3 macam:

1. *Data Defenition Language* (DDL)

Merupakan kelompok perintah yang digunakan untuk melakukan pendefenisian database dan pendefenisian tabel. Dengan kelompok perintah dalam DDL ini maka kita dapat membuat tabel, mengubah strukturnya, menghapus tabel, membuat indeks untuk tabel, dan lain-lain yang bermuara pada pembentukan struktur *database*.

2. *Data Manipulation Language* (DML)

Perintah (statement) SQL digunakan untuk melakukan manipulasi data dalam database, menambahkan (insert), mengubah (update), menghapus(delete), mengambil dan mencari data (query). Perintah SQL standar seperti : select, insert, update, delete, create, dan drop dapat digunakan untuk menyelesaikan tugas yang diberikan berhubungan dengan data suatu *database*.

3. *Data Control Language* (DCL)

Termasuk dalam DCL adalah perintah untuk melakukan pendefenisian pemakai yang boleh mengakses database dan apa saja privilegenya. Fasilitas ini tersedia pada sistem manajemen database yang memiliki fasilitas keamanan dengan membatasi pemakai dan kewenangannya.

2.7. Analisis Sistem

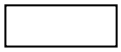
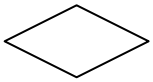
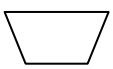
Penggambaran dan analisis model sistem yang digunakan untuk menggambarkan pembangunan perangkat lunak adalah analisis terstruktur. Analisis terstruktur adalah aktivitas pembangunan model dengan menggunakan notasi yang sesuai dengan prinsip analisis operasional. Analisis terstruktur bukan merupakan metode tunggal yang diaplikasikan secara konsisten oleh semua yang menggunakannya. Pada model ini analisis terhadap perangkat lunak dapat digambarkan dalam bentuk *Flow Map*, Diagram konteks, *Data Flow Diagram* (DFD) dan Kamus Data.

2.7.1. *Flowmap*

Flowmap adalah representasi grafik dari sistem informasi, proses-proses, aliran-aliran data logis, masukan, keluaran, dan *file-file* serta entitas-entitas sistem operasi yang berhubungan dengan sistem informasi.

Tabel 2.1

Simbol-Simbol *flowmap*

Notasi	Arti
	Simbol dokumen/arsip
	Simbol proses otomatis
	Simbol kondisi/pemilihan
	Simbol proses manual
	Simbol penyimpanan arsip manual

	Simbol penyimpanan file di Hardisk
	Simbol penghubung halaman

2.7.2. Diagram konteks

Diagram Konteks adalah bagian dari DFD yang berfungsi memetakan model model lingkungan, yang dipresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem. Diagram ini adalah diagram *level* tertinggi dari DFD. Diagram konteks menyoroti sejumlah karateristik penting sistem, yaitu :

1. Kelompok Pemakai, organisasi atau sistem lain dimana sistem melakukan komunikasi (sebagai terminator).
2. Data Masuk, yaitu data yang diterima sistem dari lingkungan dan harus diproses dengan cara tertentu.
3. Data Keluar, yaitu data yang dihasilkan sistem dan diberikan kedunia luar.
4. Penyimpanan Data (*storage*), yaitu digunakan secara bersama antara sistem dengan terminator. Data ini dapat dibuat oleh sistem dan digunakan oleh lingkungan atau sebaliknya dibuat oleh lingkungan dan digunkan oleh sistem.

   Hal ini berarti pembuatan simbol *data storage* dalam diagram konteks dibenarkan, dengan syarat simbol tersebut merupakan bagian dari diluar sistem.
5. Batasan, antara sistem dan lingkungan.

2.7.3. *Data Flow Diagram (DFD)*

DFD atau singkatan dari *Data Flow Diagram* merupakan representasi grafik dari suatu sistem yang menunjukan proses atau fungsi, aliran data, tempat penyimpan data dan *entitas eksternal*. DFD juga digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan. Dengan menggunakan DFD, rancangan yang akan kita buat akan lebih terarah dan lebih rinci. Sehingga kita tidak akan mengalami kesulitan dalam melakukan perancangan. *Data flow diagram* memiliki 4 komponen, antara lain akan dijelaskan dibawah ini :

1. *Terminator (external exitity)*

   Terminator mewakili *entity external* yang berkomunikasi dengan sistem yang sedang dikembangkan. *Terminator* merupakan kesatuan dilingkungan sistem. Biasanya terminator ini dikenal dengan nama entitas (*external*) sumber atau tujuan (*source* dan *sink*). Terminator dapat juga berupa departemen, divisi atau sistem diluar sistem yang berkomunikasi dengan sistem yang dikembangkan.

2. Proses

   Proses sering dikenal dengan nama *Bubble*, fungsi atau informasi. Komponen proses menggambarkan bagian dari sistem yang mentransformasikan *input* ke *output*, atau dapat dikatakan bahwa komponen proses menggambarkan transformasi satu *input* atau lebih menjadi *output*. Setiap proses harus diberikan penjelasan yang lengkap sebagai berikut:

a. Identifikasi Proses

Umumnya berupa angka yang menunjukan nomor dari proses atau ditulis pada bagian atas simbol proses.

b. Nama Proses

Menunjukan apa yang dikerjakan oleh proses tersebut. Nama proses harus jelas dan lengkap menggambarkan bagian prosesnya nama proses diletakan dibawah identifikasi proses.

3. Penyimpanan data *(data store)*

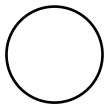
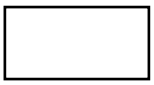
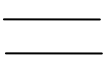
Data store digunakan sebagai sarana untuk mengumpulkan data. Data store disimbolkan dengan dua garis horizontal yang paralel dimana tertutup pada salah satu ujungnya atau dua garis horizontal saja. *Data store* ini biasanya berkaitan dengan penyimpanan-pemyimpanan seperti *file* atau *database* yang berkaitan dengan penyimpanan secara komputerisasi, contohnya *file* pita magnetic, *file* disket atau *file harddisk*. *Data store* juga berkaitan dengan pemyimpanan data.

4. Alur Data *(Data Flow)*

Suatu data *flow* atau alur data dapat dipresentasikan dengan anak panah yang menunjukan arah menuju ke dan keluar dari suatu proses. Alur data ini digunakan untuk menerangkan perpindahan data atau satu paket data atau informasi dari suatu bagian sistem ke bagian lainnya. Selain menunjukan arah, alur data pada model yang dibuat dapat merepresentasikan bit, karakter, pesan, formulir, bilangan *real* dan macam-macam informasi yang berkaitan dengan komputer.

Tabel 2.2

Simbol-Simbol *DFD*

Notasi	Arti
	Simbol proses
	Simbol kesatuan luar
	Simbol arus data
	Simbol simpanan data

2.7.4. Kamus data

Kamus data adalah suatu daftar atau kamus dari seluruh elemen-elemen data yang diperlukan oleh suatu sistem. Kamus data dibuat berdasarkan arus data yang terdapat pada DFD. Arus data dalam DFD bersifat global sehingga hanya dapat ditunjukkan nama arus datanya saja dan keterangan lebih lanjut mengenai struktur arus data dapat dilihat dari kamus data.

2.8. Perancangan Basis Data

2.8.1. Normalisasi

Normalisasi adalah suatu teknik dalam menstruktur data dalam cara-cara tertentu untuk mengurangai atau mencegah timbulnya masalah yang berhubungan dengan pengolahan data dalam *database*. Normalisasi juga diartikan sebagai proses pengelompokan data elemen menjadi tabel-tabel yang menunjukan entitas dan relasinya.

Konsep-konsep pada normalisasi, antara lain :

1. Kunci atribut (*Key field* atau *Key attribut*) yautu suatu kunci *field* yang mewakili *record* atau *tupple*
2. Kunci kandidat (*Candidate key*) yaitu suatu atribut atau satu set atribut yang mengidentifikasi secara unik suatu *entity*.
3. Kunci primer (*Primary key*) yaitu satu atribut atau satu set atribut yang yang mengidentifikasi secara unik dan mewakili setiap kejadian pada satu *entity*.
4. Kunci alternatif (*Alternate key*) yaitu kunci kandidat yang dipakai sebagai kunci primer.
5. Kunci tamu (*Foreign key*) yaitu suatu atribut atau satu set atribut dan melengkapi hubungan yang menunjukan ke induknya.

Bentuk-bentuk normalisasi yaitu :

1. Normal satu (1NF atau *First Normal Form*)

   Relasi berada pada normal kesatu jika semua atribut mempunyai nilai yang bersifat atomic.

2. Normal kedua (2NF atau *Second Normal Form*)

   Relasi berada pada normal kedua jika relasi tersebut merupakan normal satu dan atribut bukan merupakan kunci tergantung penuh pada kunci primer.

3. Normal ketiga (3NF atau *Three Normal Form*)

   Relasi berada pada normal ketiga jika relasi tersebut merupakan normal kedua dan atribut bukan kunci tidak tergantung secara transitif pada kunci primer.

4. BCNF (*Boyce Cood Normal Form*)

   Relasi berada pada BCNF jika dan hanya jika faktor penentunya adalah kunci kandidat dan relasi tersebut merupakan normal ketiga.

2.8.2. Tabel Relasi

Relasi merupakan sebuah tabel yang terdiri dari beberapa kolom dan beberapa baris. Relasi menunjukkan adanya hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Entitas merupakan individu yang mewakili sesuatu yang nyata dan dapat dibedakan dengan yang lainnya Semisal pada contoh *entity* manusia mempunyai hubungan dengan *entity* alamat yaitu ”tinggal di”. Di dalam merancang database

hendaknya seluruh *entity* yang ada mempunyai hubungan dengan *entity* yang lain, minimal satu. Jika ada *entity* dalam database yang tidak mempunyai hubungan dengan satupun *entity* yang lain, maka akan timbul kesalahan dalam desain. Biasanya *entity* yang tidak berhubungan akan dihilangkan.

2.8.3. *Entity Relationship Diagram (ERD)*

Menurut Al-Bahra, Lajamudin (2005 : 43) *Entity Relationship Diagram* adalah suatu model atau teknik pendekatan yang dapat menyatakan atau menggambarkan hubungan *entity* didalam sebuah sistem, dimana hubungan tersebut dinyatakan sebagai *one-to-one*, *one-to-many*, *many-to-one* dan *many-to-many*.

Entity relationship bertujuan untuk menentukan objek-objek data (*Entity*) dan *relationship* (hubungan) yang ada pada objek-objek tersebut.

Komponen *Entity Relationship Diagram* :

1. *Entity*

Entity adalah suatu yang dapat dibedakan atau diidentifikasikan secara unik, objek disini sifatnya berupa orang, tempat, peristiwa atau konsep yang informasinnya direkam.

2. Atribut

Atribut adalah kumpulan dari beberapa elemen data yang membentuk suatu entity. Atribut merupakan ciri sebutan atau karakteristik yang dapat

mewakili suatu simbol yang digunakan untuk menggambarkan atribut pada *entity relationship diagram.*

3. *Relationship*

Relationship adalah relasi atau hubungan yang terjadi diantara beberapa entity. Simbol yang digambarakan untuk menggambarkan *relationship* pada teknik *entity relationship diagram* adalah berupa dimendi (Belah ketupat).

Setiap *relationship* mempunyai derajat *relationship*, yaitu tingkat atau hubungan yang terdiri dari tiga derajat, yaitu :

b. *Unary* (derajat 1)

Suatu *relationship* yang dihubungkan dengan suatu *entity* set, dimana penghubungnya ada dua.

c. *Binary* (derajat 2)

Relationship derajat dua adalah *relationship* yang biasa terjadi yaitu dua *entity* set dihubungkan dengan satu *entity relationship*.

d. *Ternary* (derajat 3)

Relationship derajat tiga.

Relationship diagram juga mempunyai *Cardinalitas* yaitu batasan pada jumlah *entity* yang terdiri dari:

1. Relasi satu ke satu (*one-to-one*)

Sebuah entitas A yang berpasangan dengan tepat satu entitas B. Demikian pula entitas B juga berpasangan tepat satu dengan entitas A.

2. Relasi satu ke banyak (*one-to-many*)

Sebuah entitas A yang berpasangan secara banyak (nol atau lebih) dengan entitas di dalam B. Sedangkan entitas B hanya berpasangan dengan tepat satu entitas di dalam A.

3. Relasi Banyak ke Satu (*many-to-one)*

Sebuah entitas A yang berpasangan dengan tepat satu entitas didalam B. Sedangkan entitas B dapat berpasangan secara banyak (nol atau lebih) dengan entitas di dalam A. (sama seperti one to many hanya dibalik pengertiannya)

4. Relasi Banyak ke Banyak (*many-to-many*)

Sebuah entitas A yang berpasangan secara banyak (nol atau lebih) dengan entitas di dalam B. Dan entitas B juga dapat berpasangan secara banyak (nol atau lebih) dengan entitas di dalam A.