

BAB II

LANDASAN TEORI

A. Penelitian Terkait

Sandy Kosasi (2015) , dalam penelitiannya yang berjudul “Perancangan Sistem Informasi Integrasi Aplikasi Point Of Sale Mini Market” dijelaskan perancangan sistem informasi integrasi aplikasi point of sale untuk mini market Selera Niaga sudah dapat memberikan informasi mengenai transaksi penjualan untuk setiap harinya. Sistem informasi integrasi ini dapat melakukan proses pencatatan dan pengecekan mengenai pengelolaan dan informasi persediaan barang. Pengelolaan data barang juga menjadi lebih akurat karena adanya informasi antar proses bisnis yang saling berkaitan satu dengan lainnya. Perancangan integrasi aplikasi point of sale mini market Selera Niaga memiliki sejumlah fitur-fitur berupa dapat mengelola data barang, konsumen, pencarian produk tertentu, faktor pemesanan, pemrosesan transaksi dan faktur penjualan, penanganan persediaan minimal, dapat mengeluarkan nota penjualan dan sejumlah laporan dalam pengelolaan transaksi sehari-hari.

Silvester Dian Handy Permana (2015), dengan judul penelitian “Analisa Dan Perancangan Aplikasi Point Of Sale (POS) Untuk Mendukung Manajemen Hubungan Pelanggan”, menjelaskan rancangan aplikasi point of sales (POS) ini mampu mengatasi permasalahan dan dapat menyajikan informasi secara lebih baik dan terkomputerisasi. Dengan diterapkannya aplikasi point of sales (POS) ini dapat membantu tugas-tugas pihak-pihak

terkait atau seluruh stakeholder yang berhubungan langsung dengan aplikasi POS ini. Aplikasi POS ini didesain dengan antar muka interaktif sehingga baik administasi maupun pemilik dapat langsung menggunakan aplikasi POS ini dengan sebaik-baiknya.

Jodhi Sugihartono (2015), dalam penelitian “Pembuatan Aplikasi Point of Sale Toko Cabang Perusahaan Torani Menggunakan Framework CodeIgniter”, menjelaskan aplikasi *point of sale* ini dibuat untuk meminimalisasi pendataan barang dan transaksi secara manual serta penggunaan telepon untuk melakukan pemesanan terhadap distributor. Dengan adanya aplikasi ini, maka setiap pendataan barang di toko dan pemesanan barang yang habis di toko dapat dilakukan secara cepat dengan mengacu pada database yang ada pada perusahaan, sehingga ini akan membuat pekerja melakukan pekerjaan secara efektif dan efisien. Tujuan dilakukan pembuatan aplikasi point of sale penelitian dalam Tugas Akhir ini adalah untuk merancang sistem pendataan barang yang dapat digunakan untuk memudahkan pengecekan jumlah stok barang di toko, merancang sistem pemesanan barang di toko cabang untuk memudahkan pemesanan barang yang habis di toko, merancang sistem kasir untuk memudahkan proses transaksi jual beli dengan pelanggan toko, dan membagi sistem menjadi 3 bagian menu untuk membagi tugas antara admin, pegawai, dan petugas kasir secara terstruktur.

B. *Point Of Sales (POS)*

Point of sales atau disebut POS secara sederhananya, dapat diartikan sebagai perangkat lunak untuk kegiatan mencatat transaksi penjualan sehari-hari. Sebelum sistem POS dikembangkan. Orang-orang banyak yang menggunakan cash register, atau bahkan melakukan pencatatan secara manual dalam menjalankan usaha ritelnya. Untuk saat ini fungsi-fungsi dari cash register ini sendiri sudah tidak memadai lagi karena jumlah item barang yang semakin banyak dengan harga yang semakin bervariasi. Kondisi inilah yang merupakan alasan mendasar mengembangkan aplikasi POS. *Point of sales* (POS) adalah suatu sistem untuk mengelola penjualan barang secara ritel. Istilah ini digunakan untuk merujuk ke perangkat lunak dan perangkat keras yang berhubungan dengan check out stands, dan semua fitur yang disertakan. Kebanyakan pengecer menggunakan sistem POS dan beberapa produsen utama menawarkan sistem POS yang dirancang untuk berbagai jenis usaha, mulai dari toko kelontong untuk butik pakaian. Aplikasi POS membuat bisnis bekerja secara lebih efisien, menurunkan biaya menjalankan bisnis sekaligus meningkatkan layanan pelanggan dan pengelolaan bisnis menjadi lebih efisien dan efektif. Sebuah aplikasi sistem *point of sales* senantiasa dapat menyajikan laporan penjualan terbaru setiap hari. *Point of sales* memungkinkan transaksi di dalamnya diproses lebih cepat juga efisien dalam mengelola sistem dan tingkat persediaan barang ritel (Anonymous, 1999).

Tomy mendefinisikan sistem *Point of Sales* (POS) adalah sebuah sistem aplikasi yang diterapkan pada bisnis minimarket ataupun pertokoan untuk menangani pengolahan data transaksi pembelian (*purchases*), transaksi

penjualan eceran (*retails*), transaksi hutang (*liabilities*), transaksi retur pembelian (*purchase returns*), dan pelaporan transaksi (*reporting*) yang secara umum penting dibutuhkan dalam pengambilan keputusan strategis oleh para pebisnis swalayan, organisasi, atau perusahaan yang berskala kecil dan menengah. (Tommy, 2008).

Pengertian lain dari *Point Of Sales (POS)* yaitu merupakan kegiatan yang berorientasi pada penjualan serta sistem yang membantu proses transaksi. Setiap *POS* terdiri dari hardware berupa (Terminal/PC, *Receipt Printer*, *Cash Drawer*, Terminal pembayaran, *Barcode Scanner*) dan *software* berupa (*Inventory Management*, *Pelaporan*, *Purchasing*, *Customer Management*, Standar Keamanan Transaksi, *Return Processing*) dimana kedua komponen tersebut digunakan untuk setiap proses transaksi. *POS* akan menjadi sangat penting di dunia bisnis karena *POS* diibaratkan berupa terminal uang dimana tempat menerima pembayaran dari pembeli kepada pedagang, karena pembayaran tersebut merupakan indikator bagi pebisnis untuk mengukur tingkat pendapatan mereka (Axopos, 2015).

C. Teknologi Arsitektur *Client/Server*

Arsitektur *client/server* merupakan arsitektur yang mengacu pada sejauhmana semua komponen software berinteraksi dan membentuk sebuah sistem baru dengan sistem pembagian dan pendistribusian yang efektif. Arsitektur *client/server* dirancang untuk memisah layanan basis data dari *client*, dengan penghubungnya menggunakan jalur komunikasi data. Layanan basis data diimplementasikan pada sebuah komputer yang berdaya guna, yang

memungkinkan manajemen tersentralisasi, keamanan, dan berbagai sumber daya. Aplikasi-aplikasi client diimplementasikan pada berbagai platform, menggunakan berbagai kakas pemrograman. Biasanya server diletakkan di tempat yang berbeda dengan client dan dihubungkan dengan suatu jaringan lokal (Monshouwer & Valverde, 2011).

Sistem pendistribusian informasi dan pemeliharaan jaringan akan menjadi lebih mudah dan cepat dan dapat mengharapkan keakuratan yang lebih signifikan (Turban & Volonino, 2010). Client dapat menampung permintaan pengguna, mengecek sintaks dan menerjemahkan permintaan basis data ke dalam bahasa SQL atau bahasa basis data lainnya. Dalam konteks basis data, client bertugas untuk mengatur antarmuka pengguna (user interface) dan aplikasi logika (application logic), berperan seperti workstation yang mutakhir untuk menjalankan aplikasi basis data. Kemudian pesan tersebut dikirim kepada server, menunggu respon, dan membentuk respon untuk pengguna akhir (end-user). Server menerima dan memroses permintaan basis data tersebut kemudian mengirimkan kembali hasilnya kepada client. Proses tersebut meliputi checking authorization, ensuring integrity, maintaining the system catalog, performing query dan update processing. Untuk tambahan proses tersebut juga menyediakan concurrency dan recovery control. Semua proses ini akan berjalan sebagaimana mestinya melalui mekanisme pengaturan sistem tersebut (Connolly & Begg, 2010).

D. Pengertian Sistem Informasi

1. Pengertian Sistem.

Sistem adalah sekumpulan unsur atau elemen yang saling berkaitan serta saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan. Model umum sebuah sistem biasanya terdiri dari masukan, proses dan keluaran yang dilakukan secara beraturan. Masukan itu sendiri dapat berupa sesuatu yang mempunyai wujud dan dampak yang berpengaruh terhadap sistem tersebut, sedangkan proses itu sendiri adalah suatu bentuk atau format yang diinginkan atau atribut dari yang dihasilkan.

Jogiyanto H.M (2001), analisis sistem dapat didenifikasikan sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud mengidentifikasi dan mengevaluasi masalah-masalah, kesempatan, hambatan, yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

Davis (dalam, Al Bahra Bin Ladjamudin, 2005:3), sistem adalah himpunan sesuatu benda nyata atau abstrak yang terdiri dari bagian atau komponen yang saling berkaitan, berhubungan ketergantungan, dan saling mendukung, yang secara keseluruhan bersatu dalam satu kesatuan (*unity*) untuk mencapai tujuan tertentu secara efisien dan efektif.

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, antara lain sebagai berikut :

a. Batasan sistem (*Boundary*)

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainya atau dengan lingkungan luarnya. Batas

sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan.

b. Lingkungan luar sistem

Lingkungan luar (*environment*) adalah sesuatu atau semua yang berada di luar lingkungan sistem yang dapat mempengaruhi kerja dari sistem itu sendiri.

c. Komponen sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan.

d. Masukan sistem

Masukan adalah energi yang dimasukkan kedalam sistem, masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Sedangkan signal input adalah energi yang diproses untuk mendapatkan keluaran.

e. Keluaran sistem

Keluaran adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna.

f. Penghubung sistem

Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari suatu sub sistem ke sub sistem yang lainnya. Sedangkan penghubung itu sendiri merupakan media penghubung antara satu sub sistem dengan sub sistem lainnya, sehingga dapat berinteraksi membentuk suatu kesatuan.

g. Sasaran dan tujuan sistem jika sistem tidak mempunyai tujuan atau sasaran maka operasi sistem tidak berguna, jadi sistem dikatakan berhasil bila mengenai sasaran dan tujuannya.

Dari definisi diatas penulis dapat menyimpulkan bahwa sistem adalah seperangkat elemen yang mana elemen tersebut mempunyai sub-sub elemen yang saling berintegrasi untuk mencapai suatu tujuan.

2. Pengertian Informasi

Berikut adalah pengertian tentang informasi menurut beberapa pakar. Menurut Gordon B. Davis (dalam, Al Bahra Bin Ladjamudin, 2005:8) informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam mengambil keputusan saat ini atau masa yang akan datang.

Menurut Raymond Mc. Leod, Jr (dalam, Al Bahra Bin Ladjamudin, 2005:9) Informasi adalah data yang sudah diolah menjadi suatu bentuk yang penting bagi penerima dan punya nilai yang nyata dan dapat digunakan untuk mengambil keputusan, baik sekarang maupun yang akan datang.

Menurut Jogiyanto H.M (2001) Informasi adalah yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

Dari ketiga definisi diatas dapat disimpulkan bahwa informasi adalah:

a. Data yang telah diolah dan mempunyai arti.

- b. Data yang telah diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang digunakan untuk mengambil keputusan. Menggambarkan suatu kejadian dan kesatuan nyata bagi penerimanya.

3. Pengertian Sistem Informasi

Sistem informasi adalah sebuah rangkaian prosedur formal dimana data dikelompokkan, diproses menjadi informasi dan didistribusikan kepada pemakai. (Hall, 2001)

Sistem informasi mencakup sejumlah komponen (manusia, komputer, teknologi informasi dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi) dan dimaksudkan untuk mencapai suatu sasaran atau tujuan. (Abdul Kadir, 2014).

Sebuah sistem informasi adalah sistem buatan manusia yang berisi himpunan berintegrasi dari komponen-komponen manual dan komponen-komponen terkomputerisasi yang bertujuan untuk mengumpulkan data, memproses data dan menghasilkan informasi untuk pemakai. (Lani Sidharta, 1995).

Dari berbagai pendapat diatas dapat disimpulkan bahwa sistem informasi adalah suatu sistem di dalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal pada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan.

E. PHP (*Hypertext Preprocessor*)

1. Definisi PHP (*Hypertext Preprocessor*)

PHP adalah bahasa pemrograman *script* yang paling banyak dipakai saat ini. *PHP* banyak dipakai untuk memrogram situs *web* dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain. Contoh terkenal dari aplikasi *PHP* adalah forum (*phpBB*) dan *MediaWiki* (*software* di belakang *Wikipedia*). *PHP* juga dapat dilihat sebagai pilihan lain dari *ASP.NET/C#/VB.NET Microsoft*, *ColdFusion Macromedia*, *JSP/Java Sun Microsystems*, dan *CGI/Perl*. Contoh aplikasi lain yang lebih kompleks berupa *CMS* yang dibangun menggunakan *PHP* adalah *Mambo*, *Joomla*, *Postnuke*, *Xaraya*, dan lain-lain.

2. Sejarah PHP (*Hypertext Preprocessor*)

Pada awalnya *PHP* merupakan kependekan dari *Personal Home Page* (Situs Personal). *PHP* pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu *PHP* masih bernama *FI* (*Form Interpreted*), yang wujudnya berupa sekumpulan *script* yang digunakan untuk mengolah data *form* dari *web*. Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya *PHP/FI*. Dengan perilsan kode sumber ini menjadi *open source*, maka banyak programmer yang tertarik untuk ikut mengembangkan *PHP*. Pada November 1997, dirilis *PHP/FI 2.0*. Pada rilis ini interpreter *PHP* sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan *PHP/FI* secara

signifikan. Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter *PHP* menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk *PHP* dan meresmikan rilis tersebut sebagai *PHP 3.0* dan singkatan *PHP* dirubah menjadi akronim berulang *PHP: Hypertext Preprocessing*. Pada pertengahan tahun 1999, Zend merilis interpreter *PHP* baru dan rilis tersebut dikenal dengan *PHP 4.0*. *PHP 4.0* adalah versi *PHP* yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi *web* kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi. Pada Juni 2004, Zend merilis *PHP 5.0*. Dalam versi ini, inti dari interpreter *PHP* mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam *PHP* untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek.

3. Kelebihan *PHP (Hypertext Preprocessor)*

1. Bahasa pemrograman *PHP* adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. *Web Server* yang mendukung *PHP* dapat ditemukan dimana - mana dari mulai *apache*, *IIS*, *Lighttpd*, hingga *Xitami* dengan konfigurasi yang relatif mudah.
3. *PHP* adalah bahasa *open source* yang dapat digunakan di berbagai mesin (*Linux*, *Unix*, *Macintosh*, *Windows*) dan dapat dijalankan

secara runtime melalui console serta juga dapat menjalankan perintah-perintah sistem.

F. *MySQL*

MySQL (baca: *mai-se-kyu-el*) merupakan *software* yang tergolong sebagai *DBMS (Database Managemen Sistem)* yang bersifat *Open Source*. *Open Source* menyatakan bahwa *software* ini dilengkapi dengan *source code* (kode yang dipakai untuk membuat *MySQL*), selain tentu saja bentuk *executable*-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi, dan bisa dengan cara *men-download* (mengunduh) di internet secara gratis.

MySQL awalnya dibuat oleh perusahaan konsultan bernama *TcX* yang berlokasi di Swedia. Saat ini pengembangan *MySQL* berada dibawah naungan perusahaan *MySQL AB*.

Sebagai *software DBMS*, *MySQL* memiliki sejumlah fitur seperti yang dijelaskan dibawah ini.

1. **Multiplatform.**

MySQL tersedia pada beberapa *platform* (Windows, Linuk, Unix, dan lain-lain).

2. **Handal, cepat, dan mudah digunakan.**

MySQL tergolong sebagai *database server* (server yang melayani permintaan terhadap database) yang handal, dapat menangani database yang besar dengan kecepatan tinggi, mendukung banyak sekali fungsi untuk mengakses database, dan sekaligus mudah untuk digunakan.

Berbagai *tool* pendukung juga tersedia (walaupun dibuat oleh pihak lain). Perlu diketahui, *MySQL* dapat menangani sebuah table yang berukuran dalam *terabyte* (1 *terabyte* = 1024 *gigabyte*). Namun, ukuran yang sesungguhnya sangat bergantung pada batasan sistem operasi. Sebagai contoh, pada sistem *solaris 9/10*, batasan ukuran file sebesar 16 *terabyte*.

3. Jaminan keamanan akses.

MySQL mendukung pengamanan *database* dengan berbagai kriteria pengaksesan. Sebagai gambaran, dimungkinkan untuk mengatur *user* tertentu agar bisa mengakses data yang bersifat rahasia (misalnya gaji pegawai), sedangkan *user* lain tidak boleh. *MySQL* juga mendukung konektivitas ke berbagai *software*. Sebagai contoh, dengan menggunakan *ODBC (Open Database Connectivity)*, *database* yang ditangani *MySQL* dapat diakses melalui program yang dibuat dengan *Visual Basic*. *MySQL* juga mendukung program klien melalui *JDBC (Java Database Conectivity)*. *MySQL* juga bisa diakses melalui aplikasi berbasis *Web* : misalnya dengan menggunakan *PHP*.

4. Dukungan *SQL*.

Seperti tersirat dalam namanya, *MySQL* mendukung perintah *SQL (Structured Query Language)*. Sebagai mana diketahui, *SQL* merupakan standart dalam pengaksesan *database relasional*. Pengetahuan akan *SQL* akan memudahkan siapa pun untuk menggunakan *MySQL*. (Abdul Kadir; 2008; 2-3).

G. Web

1. Definisi Web

Web adalah salah satu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, suara, animasi, video) di dalamnya yang menggunakan protokol HTTP (*hypertext transfer protocol*) dan untuk mengaksesnya menggunakan perangkat lunak yang disebut *browser*.

Browser (perambah) adalah aplikasi yang mampu menjalankan dokumen-dokumen *web* dengan cara diterjemahkan. Prosesnya dilakukan oleh komponen yang terdapat di dalam aplikasi *browser* yang biasa disebut *web engine*. Semua dokumen *web* ditampilkan oleh *browser* dengan cara diterjemahkan.

Situs *Web* adalah dokumen-dokumen *web* yang terkumpul menjadi satu kesatuan yang memiliki *Unified Resource Locator* (URL atau *domain*) dan biasanya di-*publish* di *internet* atau *intranet*. (Arief, 2011)

2. Definisi Situs Web

Situs *web* merupakan kumpulan dari halaman *web* yang sudah dipublikasikan di jaringan *internet* dan memiliki domain atau URL (*Uniform Resource Locator*) yang dapat diakses semua pengguna *internet* dengan cara mengetikkan alamatnya. Berikut adalah contoh alamat situs *web*: www.amiko.ac.id, <http://rudyantoarief.com>. (Arief, 2011).

H. *Flowchart* (Diagram Alur)

1. Pengertian *flowchart* (Diagram Alur).

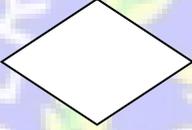
Karena komputer membutuhkan hal-hal yang rinci, maka bahasa pemrograman bukanlah alat baik untuk merancang sebuah *algoritma* awal. Alat yang banyak dipakai untuk membuat *algoritma* adalah diagram alur (*flowchart*).

Diagram alur dapat menunjukkan secara jelas arus pengendalian suatu *algoritma*, yakni melaksanakan suatu rangkaian kegiatan secara *logis* dan *sistematis*. Suatu diagram alur dapat memberi gambaran dua *dimensi* berupa simbol-simbol grafis. Masing-masing simbol telah ditetapkan lebih dahulu fungsi dan artinya. Simbol-simbol tersebut dipakai untuk menunjukkan berbagai kegiatan operasi dan jalur pengendalian. Arti khusus dari sebuah *flowchart* adalah simbol-simbol yang digunakan untuk menggambarkan urutan proses yang terjadi di dalam suatu program komputer secara *sistematis* dan *logis*. (Sutabri, 2004).

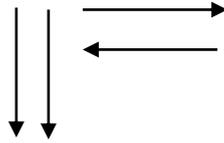
2. Simbol-simbol *flowchart*.

Sudah dikemukakan di atas bahwa diagram alur atau *flowchart* memiliki beberapa simbol yang biasa digunakan untuk menggambarkan rangkaian proses yang harus dilaksanakan. Simbol-simbol tersebut dijelaskan di bawah ini: (Sutabri, 2004).

Tabel 2.1. Simbol *Flowchart*

Simbol <i>Flowchart</i>	Fungsi
<i>TERMINAL</i>	
	Simbol ini digunakan untuk mengawali atau mengakhiri suatu proses/kegiatan.
<i>PREPARATION</i>	
	Simbol ini digunakan untuk mempersiapkan harga awal/nilai awal suatu variabel yang akan diproses.
<i>DECISION</i>	
	Simbol ini digunakan untuk pengujian suatu kondisi yang sedang diproses.
<i>PROSES</i>	
	Simbol ini digunakan untuk menggambarkan suatu proses yang sedang dieksekusi.
<i>INPUT/OUTPUT</i>	
	Digunakan untuk menggambarkan proses input (<i>read</i>) maupun proses output (<i>print</i>).
<i>SUBROUTINE</i>	
	Digunakan untuk menggambarkan proses pemanggilan subprogram dari main program.

FLOW LINE



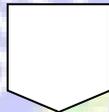
Simbol ini digunakan untuk menggambarkan arus proses dari suatu kegiatan ke kegiatan lain.

CONNECTOR



Simbol ini digunakan sebagai penghubung antara suatu proses dengan proses lainnya yang ada di dalam suatu lembar halaman.

PAGE CONNECTOR



Simbol ini digunakan sebagai penghubung antara suatu proses dengan proses lainnya, tetapi berpindah halaman.

MANUAL OPERATION



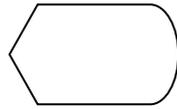
Simbol ini digunakan untuk menggambarkan suatu kegiatan atau proses yang bersifat manualisasi.

PRINTER



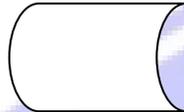
Digunakan untuk menggambarkan suatu kegiatan mencetak suatu informasi dengan mesin printer.

CONSOLE



Simbol ini digunakan untuk menggambarkan suatu kegiatan menampilkan data atau informasi melalui monitor atau *CRT (Cathode Ray Tube)*.

DISK



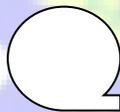
Simbol ini digunakan untuk menggambarkan suatu kegiatan membaca atau menulis data menggunakan media *magnetic disk*.

MANUAL INPUT



Simbol ini digunakan untuk menggambarkan proses pemasukan data melalui media keyboard.

TAPE



Simbol ini digunakan untuk menggambarkan suatu kegiatan membaca atau menulis data menggunakan media *magnetic tape*.

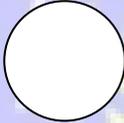
Sumber : *Analisis Sistem Informasi* (Sutabri. 2004)

I. Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah representasi grafik dari sebuah sistem. *DFD* menggambarkan komponen-komponen sebuah sistem, aliran-

aliran data di mana komponen-komponen tersebut, dan asal, tujuan, dan penyimpanan dari data tersebut. Kita dapat menggunakan *DFD* untuk dua hal utama, yaitu untuk membuat dokumentasi dari sistem informasi yang ada, atau untuk menyusun dokumentasi untuk sistem informasi yang baru. Empat simbol yang digunakan :

Tabel 2.2. Simbol *Data Flow Diagram (DFD)*

Simbol	Fungsi
	Simbol <i>entitas eksternal</i> atau <i>terminator</i> menggambarkan asal atau tujuan data di luar sistem
	Simbol lingkaran menggambarkan entitas atau proses dimana aliran data masuk ditransformasikan ke aliran data keluar
	Simbol aliran data menggambarkan aliran data
	Simbol <i>file</i> menggambarkan tempat data disimpan

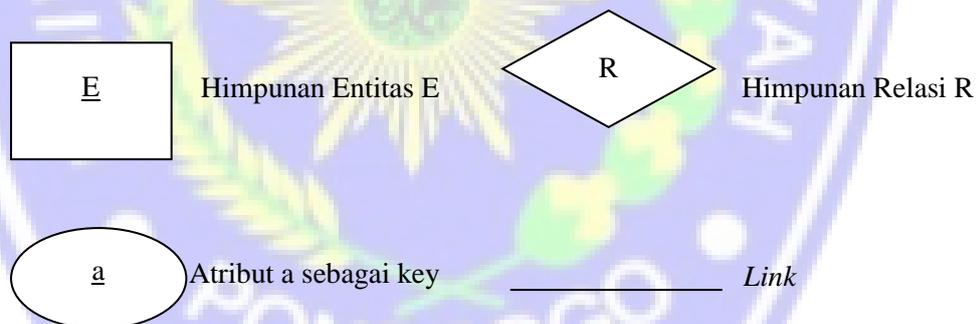
Sumber : *Analisis Sistem Informasi* (Sutabri. 2004)

J. *Entity Relationship Diagram (ERD)*

Model *entity-relationship* yang berisi komponen-komponen. Himpunan *entitas* dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang mempresentasikan seluruh fakta dari dunia nyata yang kita

tinjau, dapat digambarkan dengan lebih sistematis dengan menggunakan diagram *entity-relationship* (diagram *E-R*). Notasi-notasi simbolik didalam diagram *E-R* yang dapat kita gunakan adalah :

1. Persegi panjang, menyatakan himpunan *entitas*.
2. Lingkaran/*ellips*, menyatakan atribut. Atribut yang berfungsi sebagai *key* digaris bawahhi.
3. Belah ketupat, menyatakan himpunan relasi.
4. Garis, sebagai penghubung antara himpunan relasi dengan himpunan *entitas* dengan atributnya.
5. Kardinalitas relasi dapat dinyatakan dengan banyaknya garis cabang atau dengan pemakaian angka (1 dan 1 untuk relasi satu-ke-satu, 1 dan N untuk relasi satu-ke-banyak atau N dan N untuk relasi banyak-ke-banyak).



Gambar 2.1 Kardinalitas Relasi

K. Basis Data (*Database*)

1. Definisi Basis Data

Basis dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Sedangkan data merupakan representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa,

pembeli, pelanggan), barang, hewan peristiwa, konsep, keadaan, dan sebagainya yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya. Basis data (*database*) merupakan kumpulan data yang saling berhubungan (punya relasi). (Yakub, 2012). Menurut Janner (2007), Basis Data adalah koleksi data yang bisa mencari secara menyeluruh dan secara sistematis memelihara dan *me-retrieve* informasi.

2. Manfaat Basis Data

- a. Kecepatan dan kemudahan (*Speed*), pemanfaatan basis data memungkinkan untuk dapat, menyimpan, merubah, dan menampilkan kembali data tersebut dengan lebih cepat dan mudah.
- b. Efisiensi ruang penyimpanan (*space*), dengan basis data efisiensi atau optimalisasi penggunaan ruang penyimpanan dapat dilakukan, karena penekanan jumlah redundansi data, baik dengan sejumlah pengkodean atau dengan membuat label-label yang saling berhubungan.
- c. Keakuratan (*accuracy*), pembentukan relasi antar data bersama dengan penerapan aturan atau batasan (*constraint*) tipe, domain dan keunikan data dapat diterapkan dalam sebuah basis data.
- d. Ketersediaan (*availability*), dapat memilah data utama atau master, transaksi, data histori hingga data kadaluwarsa. Data yang jarang atau tidak digunakan lagi dapat diatur dari sistem basis data yang aktif.

- e. Keamanan (*security*), untuk menentukan siapa-siapa yang berhak menggunakan basis data beserta objek-objek di dalamnya dan menentukan jenis-jenis operasi apa saja yang boleh dilakukan.

Kebersamaan pemakai (*sharebility*), basis data dapat digunakan oleh beberapa pemakai dan beberapa lokasi. Basis Data yang dikelola oleh sistem (aplikasi) yang mendukung *multiuser* dapat memenuhi kebutuhan, akan tetapi harus menghindari inkonsistensi data. (Yakub, 2012).

3. Operasi Basis Data

Pada sebuah *disk (harddisk)*, basis dapat diciptakan dapat pula ditiadakan. Pada sebuah *disk* juga dapat menempatkan beberapa basis data, misalnya basis data kepegawaian, akademik, penjualan, perpustakaan dan lain-lain. Sementara dalam sebuah basis data dapat ditempatkan pada satu *file* atau tabel barang, faktur, pelanggan dan transaksi barang. Operasi-operasi dasar yang dapat dilakukan basis data adalah :

- a. Pembuatan basis data baru (*Create Database*)
- b. Penghapusan basis data (*Drop Database*)
- c. Pembuatan file atau tabel baru ke suatu basis data (*Create Table*)
- d. Penghapusan file atau tabel dari suatu basis data (*Drop Table*)
- e. Penambahan atau pengisian data baru di sebuah basis data (*Insert*)
- f. Pengambilan data dari sebuah file atau tabel (*Retrieve* atau *Search*)
- g. Pengubahan data dalam sebuah file atau tabel (*Update*)
- h. Penghapusan data dari sebuah file atau tabel (*Delete*)

Operasi pembuatan basis data dan tabel merupakan operasi awal yang hanya dilakukan sekali dan berlaku seterusnya. Sedangkan untuk operasi pengisian data merupakan operasi rutin yang dilakukan berulang ulang. (Yakub, 2012).

4. Persyaratan Basis Data

Ketentuan yang harus diperhatikan pada pembuatan *file* basis data agar dapat memenuhi kriteria sebagai basis data, yaitu: *redudansi* data, inkonsistensi data, pengaksesan data, data terisolasi untuk standarisasi, masalah keamanan, masalah integritas data, data *multiuser*.

a. *Redudansi* dan Inkonsistensi Data

Penyimpanan data yang sama di beberapa tempat disebut *redudansi*, hal ini akan menyebabkan pemborosan dan menimbulkan inkonsistensi data (data tidak konsisten) karena bila terjadi maka data harus dirubah pada beberapa tempat, hal ini tentunya tidak efisien.

b. Pengaksesan Data

Data di dalam basis data harus siap diakses oleh siapa saja yang membutuhkan dan mempunyai hak untuk mengaksesnya. Oleh karena itu perlu dibuat suatu program pengelolaan atau suatu aplikasi untuk mengakses data yang dikenal sebagai *Database Management System (DBMS)*.

c. Data Terisolasi untuk Standarisasi

Jika data tersebar dalam beberapa *file* dalam bentuk *format* yang tidak sama, maka akan menyulitkan dalam menulis program

aplikasi, baik untuk mengambil dan menyimpan data. Oleh karena itu ada dalam satu *database* harus dibuat satu *format* yang sama, sehingga mudah dibuat program aplikasinya.

d. Masalah Keamanan atau *Security*

Setiap pemakai sistem basis data tidak semua bagian diperbolehkan untuk mengakses semua data, misalnya data mengenai gaji pegawai hanya boleh dibuka oleh bagian keuangan, sedang bagian gudang dan bagian lain tidak diperkenankan untuk membukanya. Keamanan dapat diatur dan disesuaikan baik ditingkat basis data atau aplikasinya.

e. *Multiple User*

Salah satu alasan basis data dibangun karena nantinya data tersebut akan digunakan oleh banyak orang, baik dalam waktu berbeda maupun bersamaan. Oleh karena itu diperlukan basis data yang handal dan dapat mendukung banyak pemakai atau multiuser. (Yakub, 2012).

5. Sistem Basis Data

a. Pengertian Sistem Basis Data

Sistem basis data (*database*) merupakan sistem yang terdiri dari kumpulan *file* atau tabel yang saling berhubungan dan memungkinkan beberapa mengakses dan memanipulasinya. Sistem basis data juga merupakan suatu sistem yang menyusun dan mengelola data organisasi perusahaan, sehingga mampu menyediakan informasi yang diperlukan oleh pemakai. Istilah sistem basis data tentu saja berbeda dengan istilah

basis data, sistem basis data merupakan lingkup yang lebih luas daripada basis data. (Yakub, 2012).

b. Komponen Sistem Basis Data

Sistem basis data terdapat komponen-komponen utama yaitu; perangkat keras (*hardware*), sistem operasi (*operating system*), basis data (*database*), program aplikasi (*application program*), *Database Management System (DBMS)*, dan pemakai (*user*).

c. *Database Management System (DBMS)*

Database Management System (DBMS) merupakan kumpulan program aplikasi yang digunakan untuk membuat dan mengelola basis data. *DBMS* berisi suatu koleksi data dan satu set program untuk mengakses data. *DBMS* merupakan perangkat lunak (*software*) yang menentukan bagaimana data tersebut diorganisasi, disimpan, diubah dan diambil kebalik. Perangkat ini juga menerapkan mekanisme pengamanan data, pemakaian data bersama, dan konsistensi data.

d. Pemakai (*Users*)

Pemakai atau *users* adalah beberapa jenis atau tipe pemakai pada sistem basis data, berdasarkan cara mereka berinteraksi pada basis data, diantaranya program aplikasi, pemakai mahir, pemakai umum dan pemakai khusus.

1) *Programmer Aplikasi*

Programmer aplikasi adalah pemakai yang berinteraksi dengan basis data melalui *DML (Data Manipulation Language)*, yang disertakan dalam program yang ditulis dalam bahasa

pemrograman induk (seperti *pascal*, *cobol*, *clipper*, *foxpro*, dan lain – lain).

2) *User Mahir (Casual User)*

Casual user adalah pemakai yang berinteraksi dengan sistem tanpa menulis modul program. Pemakai menggunakan *query* (untuk akses data), dengan bahasa *query* yang telah disediakan oleh suatu *DBMS*.

3) *User Umum (Naive User)*

Naive user adalah pemakai yang berinteraksi dengan sistem basis data melalui pemanggilan satu program aplikasi permanen, yang telah ditulis atau disediakan sebelumnya.

4) *User Khusus (Specialized User)*

Specialized User adalah pemakai yang menulis aplikasi basis data non konvensional untuk keperluan khusus, seperti untuk aplikasi sistem pakar, pengolahan citra, dan lain-lain.

6. **Administrator Basis Data**

Sebuah lingkup basis data seharusnya mempunyai satu orang atau sekelompok orang pada bagian struktur basis data untuk menangani administrasi basis data yang biasa disebut administrator basis data atau *Database Administrator (DBA)*. *Administrator* basis data adalah orang yang bertanggung jawab dan bekerjasama dengan analisis sistem dan *user-user* lain guna melengkapi berbagai tugas seperti; mendefinisikan data, pemodelan data, desain basis data, serta menjamin keserasian integritas data. (Yakub, 2012).

L. XAMPP

1. Definisi XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. *XAMPP* merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstall *XAMPP* maka tidak perlu lagi melakukan instalasi dan konfigurasi *web server Apache*, *PHP* dan *MySQL* secara manual. *XAMPP* akan menginstallasi dan mengkonfigurasikannya secara otomatis untuk anda atau auto konfigurasi.

XAMPP merupakan salah satu paket instalasi *Apache*, *PHP* dan *MySQL* instant yang dapat kita gunakan untuk membantu proses instalasi ketiga produk tersebut. Selain paket instalasi instant *XAMPP* versi 1.6.4 juga memberikan fasilitas pilihan penggunaan *PHP4* atau *PHP5*. Untuk berpindah versi *PHP* yang ingin digunakan juga sangat mudah dilakukan dengan menggunakan bantuan *PHP-Switch* yang telah disertakan oleh *XAMPP*, dan yang terpenting *XAMPP* bersifat *free* atau gratis untuk digunakan.

Dukungan terhadap pemrograman berbasis obyek (*object oriented programming/OOP*) ditambahkan pada *Perl 5*, yang pertama kali dirilis pada tanggal 31 Juli 1993. Proyek pengembangan *Perl 6* dimulai pada tahun 2000, dan masih berlangsung hingga kini tanpa tanggal yang jelas kapan mau dirilis. Ini dikatakan sendiri oleh *Larry Wall* dalam satu pidatonya yang dikenal dengan seri *The State of the Onion*. Dua di antara karakteristik utama *Perl* adalah penanganan teks dan berbagai jalan pintas

untuk menyelesaikan persoalan-persoalan umum. Perl sangat populer digunakan dalam program-program *CGI (Common Gateway Interface)* dan berbagai protokol Internet lainnya. Seperti diketahui, *TCP/IP* sebagai basis bagi semua protokol Internet yang dikenal sekarang ini menggunakan format teks dalam komunikasi data. Seperti juga bahasa populer lainnya, Perl menerima banyak kritikan. Meski banyak di antaranya hanya berupa mitos, atau berlebih-lebihan, tapi terdapat juga sejumlah kritikan yang valid. Salah satunya adalah, sintaksnya susah dibaca, karena banyak menggunakan simbol-simbol yang bukan huruf dan angka.

2. Bagian Penting dari XAMPP

Mengenal bagian *XAMPP* yang biasa digunakan pada umumnya: *htdocs* adalah *folder* tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas *PHP*, *HTML* dan *script* lain. *PHPMyAdmin* merupakan bagian untuk mengelola basis data *MySQL* yang ada dikomputer. *Control Panel* yang berfungsi untuk mengelola layanan *service XAMPP*. Seperti menghentikan *stop* layanan, ataupun memulai *start*.

XAMPP merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstall *XAMPP* maka tidak perlu lagi melakukan instalasi dan konfigurasi *web server Apache*, *PHP* dan *MySQL* secara manual. *XAMPP* akan menginstallasi dan mengkonfigurasikannya secara otomatis untuk anda atau auto konfigurasi. *Software XAMPP* versi ini terdiri atas:

- a. *Apache* versi 2.0.54

- b. *MySQL* versi 4.1.12
- c. *PHP* versi 5.0.4
- d. *PHPMyAdmin* versi 2.6.2-p11 dan lain-lain

M. Adobe Dreamweaver CS6

1. Pengertian Adobe Dreamweaver CS6

Adobe Dreamweaver CS6 merupakan *HTML editor* professional yang berfungsi mendesain, melakukan *editing* dan mengembangkan aneka *website*. Salah satu kelebihan *Adobe Dreamweaver CS6* yaitu ruang kerja *Adobe Dreamweaver CS6* beserta *tools* yang tersedia dapat digunakan dengan sangat mudah dan cepat sehingga anda bisa membangun suatu *website* dengan cepat dan tanpa harus melakukan *coding*. Selain itu, *Adobe Dreamweaver CS6* juga mempunyai integrasi dengan produk *macromedia* lainnya, seperti *flash* dan *firework*, *flash* sudah sangat terkenal sebagai program untuk membuat animasi yang berbasis *web* dengan perkembangan kebutuhan dan teknologi, *flash* akhir-akhir ini juga digunakan untuk membuat animasi dan video.

2. Fasilitas Adobe Dreamweaver CS6

Adobe dreamweaver CS6 memiliki peningkatan dalam kemampuan *toolbar*, yaitu dapat digunakan untuk memodifikasi tampilan *toolbar* atau menambahkan fungsi baru. Selain antarmuka pengguna baru, *adobe dreamweaver CS6* memiliki kemampuan untuk menyunting kode dengan lebih baik. *Adobe dreamweaver CS6* juga dapat melakukan print kode pada jendela *code view*, serta memiliki fasilitas *code hints* yang membantu

dalam urutan tag-tag, serta *tag inspector* yang sangat berguna dalam menangani *tag-tag HTML*.

