BAB 2
TINJAUAN PUSTAKA

2.1 Tinjauan Perusahaan

2.1.1 Sejarah Instansi

PT. Telview Technology didirikan oleh Ir. Darwin Lestari pada tahun 1992 yang bermula bernama PT. Suryaniaga Loka Lestari yang beralamat di Jl. Sukamaju No. 25–27 Sukajadi Bandung. Pada tanggal 10 Juli 2011 menempati kantor baru di Jalan Padjadjaran no. 26. Awal pendirian Perusahaan ini yaitu bergerak dibidang komputer dan seiring berkembangnya jaman Ir. Darwin Lestari melihat peluang bisnis di bidang *security* yaitu CCTV *security* dan Alarm Security. PT. Telview Technology merupakan suatu perusahaan yang bergerak dibidang Penjualan dan Pemasangan CCTV *security Camera*, *Alarm* dan *Access Control System* sebagai Distributor Sentra Penjualan dan Jasa service baik pemasangan maupun jasa *service*. Kantor pusat berada di Jl. Padjadjaran No. 26 Bandung dengan *Outlet* yang tersebar di seluruh cabang seluruh Indonesia. PT. Telview Technology membuat suatu ikon sebagai ciri khas dari perusahaan yaitu memasang sebuah CCTV berdiameter kurang lebih dua meter dengan berat mencapai satu ton. Kamera CCTV ini masuk ke Museum Rekor Indonesia (MURI) dan Guinness Book Of Record dalam kategori *Technology* dan *Sains*.

CCTV ini dapat berfungsi layaknya kamera CCTV standar sehingga CCTV ini bukan hanya sekedar pajangan melainkan dapat digunakan juga sebagai *security* untuk memonitoring jalan raya pajajaran khususnya didepan PT. Telview Technology. Ide CCTV raksasa ini menurut direktur utama PT. Telview Technology, Ir. Darwin Lestari yaitu sebagai *security awareness* dan pengingat bahwa lingkungan sekitar kita perlu perhatian khusus baik itu secara langsung maupun tidak langsung dengan asumsi apabila setiap tempat mempunyai CCTV maka tidak menutup kemungkinan rasa aman akan semakin meningkat di kalangan masyarakat.

2.1.2 Visi dan Misi PT. Telview Technology

Visi perusahaan yang akan menjadi sebuah pandangan tentang tujuan jangka panjang perusahaan atau rencana yang akan dicapai oleh perusahaan. Misi merupakan kegiatan atau aktifitas yang mengarahkan perusahaan agar dapat tercapainya tujuan yang menjadi impian perusahaan. Visi dan misi ini yang menjadi dasar perusahaan.

2.1.2.1 Visi Perusahaan

Visi dari perusahaan PT.Telview Technology adalah " Perusahaan kelas dunia dalam teknologi dan keamanan yang terpercaya dan mampu mengeksplorasi sumber daya teknologi yang berkembang".

2.1.2.2 Misi Perusahaan

Misi dari perusahaan PT.Telview Technology adalah :

1. Mempertahankan standar kinerja terbaik dalam industri, dan mendapatkan reputasi yang sangat baik.
2. Menghadirkan produk berkualitas tinggi, layanan profesional dan kualitas.
3. Menawarkan produk terbaik dan pelayanan kepada pelanggan kami adalah komitmen kami.
4. Menyediakan produk berkualitas tinggi dan layanan purna jual bagi pelanggan kami.
5. Menjadikan CCTV sebagai suatu teknologi keamanan.

2.1.3 Logo Instansi

Perusahaan CCTV security camera PT. Telview Technology mempunyai sebuah logo perusahaan sebagai berikut :



Gambar 2.1 Logo Perusahaan

Filosofi Logo :

- Tulisan dan Logo berwarna biru muda merupakan warna khas dari PT. Telview Technology.
- Lingkaran penuh didalam melambangkan suatu kepala atau inti seseorang yang sangat membutuhkan suatu perlindungan.
- Lingkaran busur biru didalam melambangkan suatu tangan seseorang yang membutuhkan suatu perlindungan.
- Lingkaran busur biru diluar melambangkan suatu perlindungan menyeluruh.

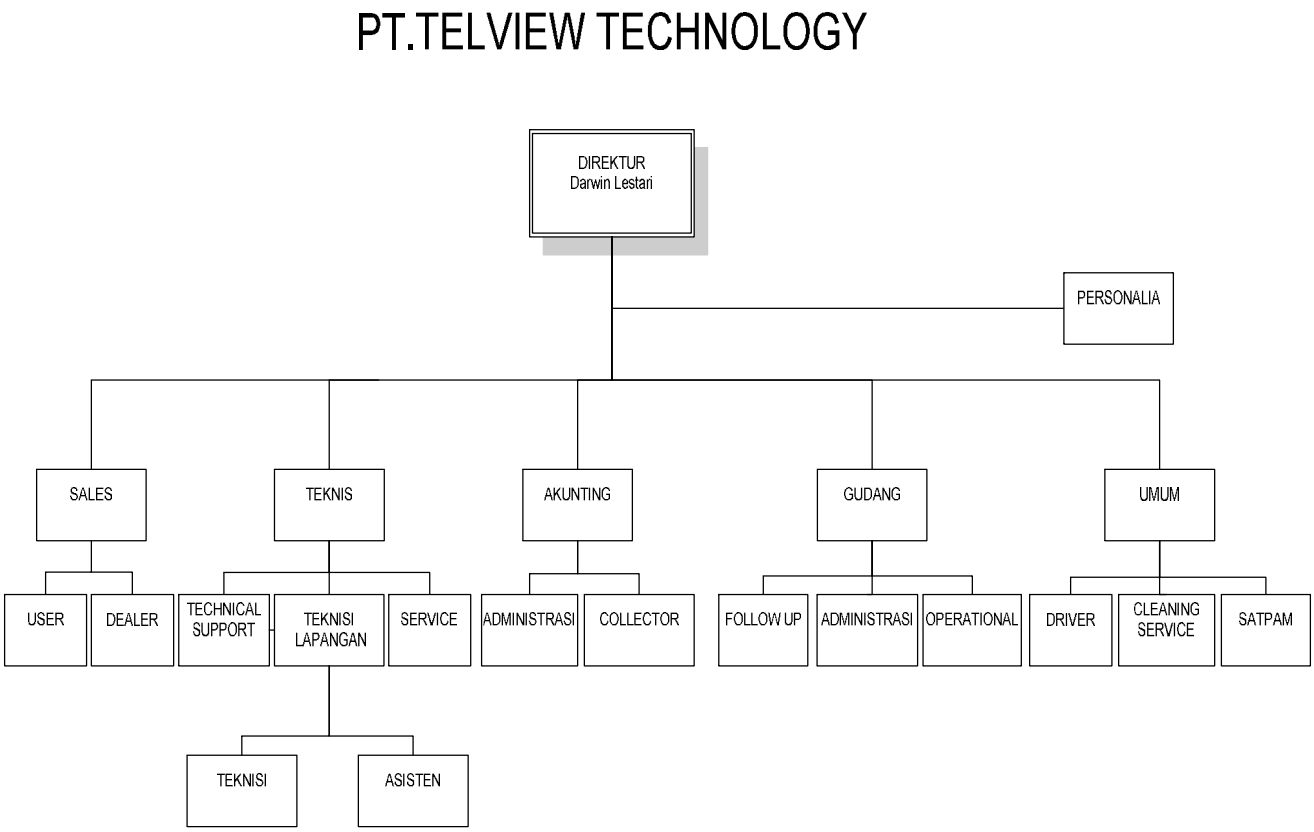
Arti Logo itu sendiri dapat didefinisikan bahwa “Sesuatu yang berharga diperlukan suatu perlindungan yang maksimal yang menjadikan rasa aman dan perlindungan itu sendiri dapat diperoleh dengan memasang CCTV kamera di PT. Telview Technology”.

2.1.4 Struktur Organisasi dan *Job Description*

Dalam suatu perusahaan perlu adanya pembagian kerja agar lebih terstruktur dalam melaksanakan kegiatan perusahaan. Struktur Organisasi perusahaan dapat membantu terselesaikannya pekerjaan dengan lebih baik dan lebih cepat karena menggambarkan hierarki tanggung jawab. Dengan adanya penempatan mengenai kemampuan dan keahlian setiap orang diharapkan

keunggulan seseorang dapat menutupi kekurangan orang lain sehingga dapat terwujud suatu tim yang solid.

2.1.4.1 Struktur Organisasi



Gambar 2.2 Struktur Organisasi

2.1.4.2 Job description

Dalam melaksanakan tugasnya setiap karyawan mempunyai tanggung jawabnya masing – masing. Pada gambar 2.2 merupakan pengorganisasian tanggung jawab setiap karyawan perusahaan. Berikut adalah penjelasan mengenai bagian dari struktur organisasi PT.Telview Technology Bandung :

1. Direktur

Memimpin perusahaan dengan menerbitkan kebijakan - kebijakan perusahaan serta menerima laporan mengenai perkembangan perusahaan.

2. Personalia / HRD

Mengecek report harian karyawan, *salary* dan evaluasi kinerja dari para karyawan.

3. Marketing

a. Sales User

Melakukan aktifitas penjualan barang kepada *customer* yang memang untuk dipakai pribadi.

b. Sales Dealer

Melakukan aktifitas penjualan barang kepada *customer* apabila customer tersebut membeli barang akan tetapi untuk dijual kembali.

c. Sales Counter

Melakukan aktifitas penjualan barang kepada *customer* di beberapa cabang *counter* yang ada.

4. Teknis

a. Teknikal *Support*

Melakukan Penelitian terhadap fitur barang dan spesifikasi yang dibutuhkan untuk menunjang pengoperasian barang serta melayani *service by phone* terhadap masalah dari *customer*.

b. Teknisi Lapangan

Melakukan pekerjaan *Survey*, Instalasi dan *service* di tempat *customer*.

c. Asisten

Melakukan pekerjaan Instalasi di tempat *customer* sesuai dengan arahan dari Teknisi.

d. Service

Melakukan perbaikan terhadap barang – barang yang mengalami kerusakan *Hardware* dan *Software*.

5. Akunting

Melakukan pekerjaan pembukuan serta administrasi faktur pembelian,

penjualan, pajak dan faktur lainnya.

6. Gudang

Melakukan pengecekan stok barang dan data keluar masuk barang sejak pembelian hingga penjualan.

7. Umum

Melakukan pekerjaan yang berhubungan dengan kebersihan, keamanan dan penagihan.

2.1.5 Badan Hukum Instansi

PT. Telview Technology merupakan suatu perusahaan milik perorangan yaitu Bpk. Darwin Lestari yang berperan pula sebagai Direktur di PT. Telview Technology. Badan Hukum perusahaan tercatat dalam :

Nomor Pokok Wajib Pajak (NPWP) : 21.131.831.6-423.000

Surat Izin Usaha Perdagangan (SIUP) : 51013-6629-BPMPPT

2.2 Landasan Teori

Teori – teori pendukung dari beberapa ahli dan literatur sangat diperlukan dalam sebuah penelitian yang digunakan sebagai dasar pedoman penelitian. Landasan teori bermanfaat untuk memberi suatu gambaran umum tentang latar suatu penelitian dan pembahasan hasil penelitian. Teori yang akan dijelaskan dalam landasan teori ini diantaranya Konsep dasar Data, Konsep dasar Sistem, Konsp dasar Informasi, metode pengembangan sistem yang digunakan dan *tools* yang digunakan dalam pembangunan aplikasi *targeting checklist ini*.

2.2.1 Konsep dasar Data

Data merupakan kumpulan dari fakta – fakta atau kejadian – kejadian yang dapat berupa angka, simbol, huruf dan lain – lain yang berguna untuk proses pengolahan data sebagai suatu masukan bagi suatu proses.

Data dapat diartikan sebagai suatu kumpulan fakta atau kejadian yang belum mempunyai arti bagi penerimanya dan masih memerlukan pengolahan. Data menggambarkan sebuah representasi fakta yang tersusun secara terstruktur.

2.2.2 Konsep dasar Informasi

Informasi merupakan suatu hasil dari pengolahan dari sebuah model, formasi, organisasi ataupun perubahan suatu bentuk dari data yang memliki nilai tertentu dan bisa digunakan untuk menambah pengetahuan bagi yang menerimanya.. Data yang sudah tersedia dikemas sedemikian rupa sehingga menjadi sebuah informasi yang berguna bagi penerimanya. Informasi adalah suatu hasil dari pemrosesan data menjadi sesuatu yang bermakna bagi yang menerimanya. Informasi dapat diartikan sebagai hasil dari pengolahan data dalam suatu bentuk yang menggambarkan suatu kejadian yang nyata menjadi lebih berguna dan lebih berarti.

2.2.3 Konsep dasar Sistem

Sistem merupakan kumpulan elemen yang saling berinteraksi untuk mencapai suatu tujuan tertentu. Menurut Jerry Fitch Gerald, Sistem adalah suatu jaringan kerja dari prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu.

Suatu sistem memiliki karakteristik tertentu yaitu memiliki komponen - komponen, batas sistem, lingkungan luar sistem, penghubung, masukan, keluaran, pengolahan, sasaran dan tujuan.

1. Memiliki Komponen *(Component)*

Suatu sistem terdiri dari sejumlah komponen yang saling berhubungan dan berinteraksi serta bekerja sama untuk membentuk satu kesatuan. Komponen sistem dapat berupa bagian-bagian dari sistem ataupun subsistem dari sistem tersebut. Setiap sub sistem tersebut mempunyai sifat untuk menjalankan fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Sistem dapat memiliki suatu sistem yang lebih besar lagi yang disebut sebagai *suprasystem.*

2. Memiliki Batas Sistem *(System Boundary)*

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batasan sistem ini dapat menunjukkan ruang lingkup dari sistem tersebut.

3. Lingkungan Luar Sistem *(System Environment)*

Lingkungan luar sistem merupakan semua sistem apapun yang ada diluar batas dari sistem yang mempengaruhi operasi sistem yang dapat bersifat menguntungkan atau merugikan sistem.

4. Penghubung Sistem *(System Interprest)*

Suatu media penghubung sistem antara satu subsistem dengan subsistem lainnya disebut dengan penghubung sistem Dengan adanya media penghubung ini akan terjadi interaksiantar subsistem sehingga membentuk suatu satu kesatuan yang memungkinkan mengalirkan sumber-sumber daya dari sub sistem satu ke sub sistem lainnya.

5. Masukan Sistem *(System Input)*

Masukan Sistem merupakan suatu energi yang dimasukkan ke dalam 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 didapatkan keluaran.

6. Keluaran Sistem *(System Output)*

Keluaran Sistem merupakan hasil masukan yang diolah serta diklasifikasikan menjadi suatu keluaran yang berguna.

7. Pengolahan Sistem *(System Process)*

Pengolahan Sistem merupaan bagian pengolah sistem yang berfungsi untuk mengubah masukan menjadi keluaran.

8. Sasaran dan Tujuan Sistem *(System Goal)*

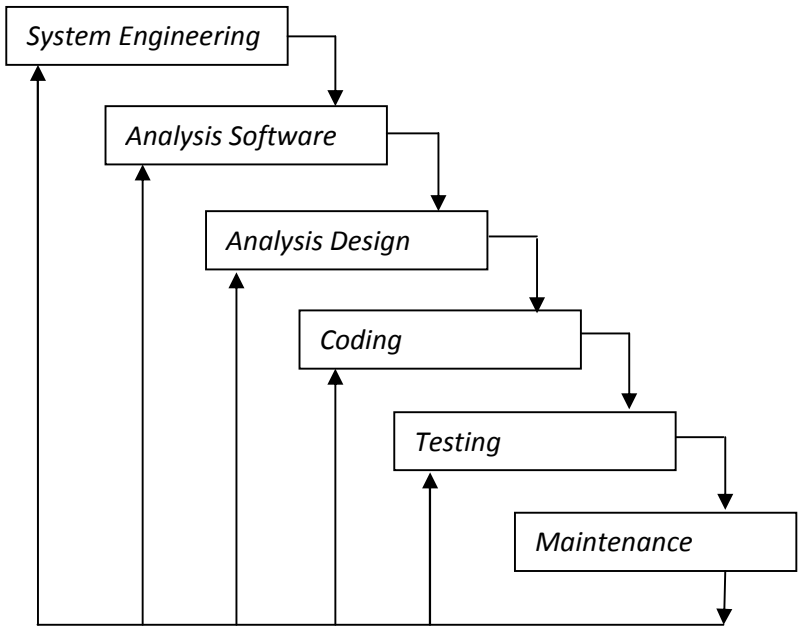
Sasaran Sistem menentukan masukan yang dibutuhkan serta keluaran yang akan dihaslkan oleh sistem tersebut. Sistem tersebut dikatakan sebagai sistem yang *Goal* apabila operasi sistem tersebut berguna dan berhasil mencapai sasaran tujuannya.

2.2.4 Metode Pengembangan Sistem

Metode yang digunakan pada aplikasi ini adalah metode *waterfall* dengan pengembangan *System Development Life Cycle* (SDLC) yaitu proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model – model atau metodologi yang digunakan orang lain untuk mengembangkan sistem – sistem perangkat lunak sebelumnya. Metode pembangunan aplikasi berorientasi objek *Unified Modeling Language* (UML) yaitu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadap perangkat lunak tersebut.

2.2.4.1 Metode *Waterfall*

Model *waterfall* sering disebut juga model *sekuensial linear*. Model *waterfall* menyediakan pendekatan alur hidup perangkat lunak secara terurut mulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung.
Berikut adalah gambar dari model *waterfall* :



Gambar 2.3 Model *Waterfall*

Tahap – tahap pada metode *waterfall* adalah sebagai berikut :

1. Tahap Pemodelan dan Rekayasa *(System Engineering)*

Dalam Tahap Rekayasa ini merupakan tahapan awal untuk merumuskan sistem yang akan dibangun. Tahap ini bertujuan agar memahami sistem yang akan dibangun.

2. Tahap Analisis Perangkat Lunak *(Analysis Software.)*

Tahap ini dilakukan terhadap permasalahan yang terjadi untuk mengetahui kebutuhan perangkat lunak dari aplikasi yang akan dibangun. Fungsi *Input*, fungsi proses dan fungsi *output* dirancang pada tahap ini. Proses pengumpulan kebutuhan dilakukan secara *intensif* untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

3. Tahap Perancangan Perangkat Lunak *(Analysis Design)*.

Tahap perancangan perangkat lunak ini adalah perancangan aplikasi yang dilakukan berdasarkan data yang telah dikumpulkan sebelumnya agar data lebih mudah dipahami oleh pengguna. Pembentukan perancangan struktur file, struktur menu, struktur program, *input* dan *output*. Pada tahap ini proses translasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

4. Tahap Implementasi Perangkat Lunak *(Coding)*.

Tahap ini adalah kegiatan menterjemahkan data perancangan dalam suatu bahasa pemrograman agar dapat dipahami oleh bahasa mesin. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

5. Tahap Pengujian *(Testing)*.

Tahap pengujian ini dilakukan terhadap perangkat lunak yang telah dibangun apakah sesuai dengan maksud dan tujuan yang akan dicapai atau tidak. Dalam pengujian ini terfokus pada perangkat lunak dari segi lojik dan fungsional serta memastikan semua bagian sudah diuji. Hal ini dilakukan untuk

meminimalisir kesalahan dan memastikan keluaran yang dihasilkan sesuai dengan yang diharapkan.

6. Tahap Pemeliharaan *(Maintenance)*.

Tahap ini adalah tahap penerapan secara keseluruhan disertai pemeliharaan suatu perangkat lunak yang telah dibangun sehingga dapat dilakukan perubahan dan penambahan data sesuai dengan permintaan pengguna. Tahap ini dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang baru.

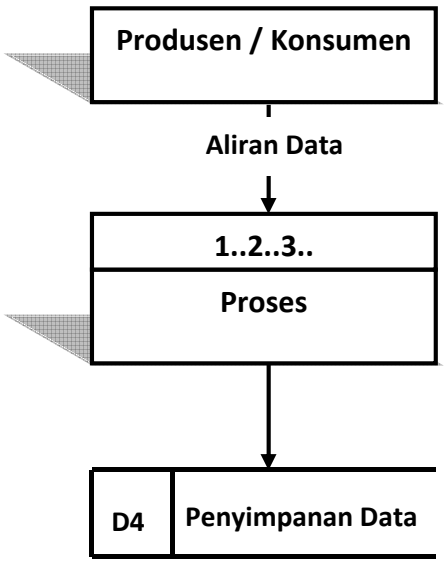
2.2.4.2 Konsep Perancangan Sistem

Pemrograman terstruktur adalah paradigma pemrograman yang membagi-bagi program berdasarkan fungsi atau prosedur – prosedur yang dibutuhkan program komputer. Pembagian program biasanya dengan mengelompokan fungsi-fungsi dan prosedur-prosedur yang diperlukan sebuah proses tertentu. Fungsi dan prosedur ditulis secara terurut dari atas ke bawah sesuai urutan ketergantungan antar fungsi atau prosedur dimana fungsi atau prosedur yang dapat dipakai dibawahnya harus yang sudah dideklarasikan diatasnya.

Pemodelan pada pemrograman terstruktur lebih fokus bagaimana memodelkan data dan fungsi yang harus dibuat. Jenis paradigma pemrograman yag digunakan dapat dideteksi dari bahasa pemrograman apa yang akan digunakan untuk membuat program, baru setelah itu ditentukan paradigma pemrograman apa yag akan digunakan.

2.2.4.2.1 Data Flow Diagram (DFD)

Data Flow Diagram awalnya dikembangkan oleh Chris Gane dan Triash Sarson pada tahun 1979 yang termasuk dalam Structured Systems Analysis and Design Methodology (SSADM) yang ditulis oleh Chris Gane dan Triash Sarson. Sistem yang dikembangkan ini berbasis pada dekomposisi fungsional dari sebuah sistem. Berikut adalah contoh DFD yang dikembangkan oleh Chris Gane dan Triash Sarson :



Gambar 2.4 Contoh DFD

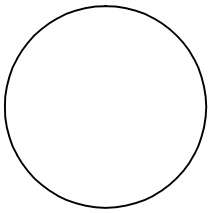
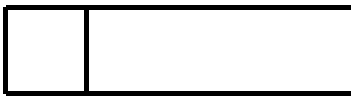
Edward Yourdon dan Tom DeMarco memperkenalkan metode yang lain pada tahun 1980an dimana mengubah persegi dengan sudut lengkung pada DFD Chris Gane dan Triash Sarson dengan lingkaran untuk menotasikannya. DFD Edward Yourdon dan Tom DeMarco populer digunakan sebagai model analisis sistem perangkat lunak untuk sistem perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur.

Informasi yang ada didalam perangkat lunak dimodifikasi dengan beberapa transformasi yang dibutuhkan. Data Flow Diagram (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan dan keluaran.

DFD dapat digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level abstraksi. DFD dapat dibagi menjadi beberapa level yang lebih detail. DFD menyediakan mekanisme untuk pemodelan fungsional ataupun pemodelan aliran informasi. Oleh karena itu, DFD lebih sesuai digunakan untuk memodelkan fungsi-fungsi perangkat lunak yang akan diimplementasikan menggunakan pemrograman terstruktur karena pemrograman terstruktur membagi-bagi bagiannya dengan fungsi-fungsi dan prosedur-prosedur.

DFD tidak sesuai untuk memodelkan sistem perangkat lunak yang akan dibangun menggunakan pemrograman berorientasi objek. Paradigma pemrograman terstruktur dan pemrograman berorientasi objek merupakan hal yang berbeda. Notasi-notasi pada DFD Edward Yourdon dan Tom DeMarco adalah sebagai berikut :

Tabel 2.1 Notasi DFD

Notasi	Keterangan
	Notasi ini menggambarkan suatu proses atau fungsi atau prosedur pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur. Nama yang diberikan biasanya berupa kata kerja.
	Notasi ini menggambarkan suatu basis data atau penyimpanan pada pemodelan perangkat lunak dengan pemrograan terstruktur maka tabel-tabel inilah yang harus dibuat menjadi tabel basis data yang dibutuhkan. Tabel ini juga harus sesuai dengan perancangan tabel pada basis data ERD. Nama yang diberikan biasanya berupa kata benda.
	Notasi ini menggambarkan Entitas luar (masukan atau keluaran atau orang yang berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan. Nama yang diberikan biasanya berupa kata benda.
	Notasi ini menggambarkan aliran data yang dikirim antar proses. Nama yang

	digunakan pada aliran data ini biasanya berupa kata benda dan diawali dengan kata data.

Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD level 0 atau sering disebut juga Context Diagram.

Diagram level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orag maupun sistem lain. DFD level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.

2. Membuat DFD level 1

DFD level 1 digunakan ntuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD level 1 merupakan hasil breakdown DFD level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD level 2

Modul-modul pada DFD level 1 dapat dibreakdown menjadi DFD level 2. Modul mana saja yang harus di breakdown lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk dibreakdown lagi. Untuk sebuah sistem, jumlah DFD level 2 sama dengan jumlah modul pada DFD level 1 yang dibreakdown.

4. Membuat DFD level 3 dan seterusnya.

DFD level 3, 4, 5 dan seterusnya merupakn breakdown dari modul pada DFD level diatasnya. Breakdown pada level 3, 4 , 5 dan seterusnya aturannya sama persis dengan DFD level 1 atau level 2.

Pada satu diagram DFD sebaiknya jumlah modul tidak boleh lebih dari 20 buah modul, diagram akan terlihat rumit dan susah untuk dibaca sehingga menyebabkan sistem yang dikembangkan juga menjadi rumit.

2.2.4.2.2 Kamus Data

Kamus data dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD. Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan dan keluaran dapat dipahami secara umum. Kamus data biasanya berisi :

1. Nama-nama dari data.
2. Penggunaan yang merupakan proses-proses yang terkait dengan data.
3. Deskripsi data
4. Informasi tambahan seperti tipe data, nilai data, batas nilai data dan komponen yang membentuk data.

Kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut :

Tabel 2.2 Simbol Kamus Data

Simbol	Keterangan
=	Disusun atau terdiri dari
+	Dan
[|]	Baik ... atau...
$\{\}^n$	n kali diulang atau bernilai banyak
( )	Data opsional
...	Batas komentar

Kamus data pada DFD harus dapat dipetakan dengan hasil perancangan basis data yang dilakukan sebelumnya. Jika ada kamus data yang tidak dapat dipetakan pada tabel hasil perancangan basis data berarti hasil perancangan basis data dengan perancangan dengan DFD masih belum sesuai sehingga harus ada yang diperbaiki baik perancangan basis datanya, perancangan DFD atau keduanya.

2.2.5 Perangkat Lunak Pendukung

2.2.5.1 Pengertian *World Wide Web*

World Wide Web adalah sekelompok dokumen multimedia yang saling terkoneksi menggunakan *hypertext link*. Dan dengan mengklik pada *hot link*, anda dapat berpindah dari suatu dokumen ke dokumen lainnya. (Bustami, Ahmad 1999) Ada dua hal penting yang harus diketahui bila belajar dengan *WWW* yaitu *software web browser* dan *software web server*. Dimana *software web browser* ini bertindak sebagai *client* yang memungkinkan anda untuk melihat dan mendapatkan informasi dari *server web,* sedangkan *software web server* bertindak sebagai *server* yang memberikan/menyedikan informasi yang diminta oleh *browser*. (Bustami, Ahmad 1999)

2.2.5.2 Pengenalan HTML

HTML atau yang memiliki kepanjangan *Hypertext Markup Language* adalah *script* di mana kita bisa menampilkan informasi dan daya kreasi kita lewat internet. HTML sendiri adalah suatu dokumen teks biasa yang mudah dimengerti dibandingkan dengan bahasa pemrograman lainnya, dan karena bentuknya itumaka HTML dapat dibaca oleh berbagai platform seperti: Windows, Linux, Macintosh. Kata *"Markup Language"* pada HTML menunjukkan fasilitas yang berupa tanda tertentu dalam skrip HTML dimana kita bisa mengatur judul, garis, tabel, gambar, dan lain-lain dengan perintah yang telah ditentukan pada elemen HTML. Penamaan dokumen *HTML* dapat dilakukan dengan memilih suatu nama, sembarang nama, kemudian menambahkan sebuah ekstensi *".htm"* atau *".html"* . HTML sendiri dikeluarkan oleh *W3C (World Wide Web Consortium)*, setiap terjadi perkembangan level HTML, harus dievaluasi ketat dan disetujui oleh W3C. HTML terdiri dari beberapa bagian yang fungsinya sebagai penanda suatu kelompok perintah tertentu, misalnya kelompok perintah form yang ditandai dengan kode < *form*>, judul dengan <*title*> dan sebagainya. Untuk mengetahui lebih lanjut mengenai bagian-bagian HTML perhatikan skema di bawah ini:

<html>
<head>
<title> </title>
</head>
<body> isi dari halaman web </body>
</html>

Keterangan:

1. Dokumen HTML selalu diawali dengan tanda tag pembuka <html> dan diakhiri dengan tanda tag penutup </html>.
2. Pada elemen head <head></head>, dapat kita sisipkan kode-kode untuk menuliskan keterangan tentang dokumen HTML. Atau dapat juga kita sisipkan kode-kode pemrograman web seperti *JavaScript*, *VBScripts*, atau *CSS* untuk menambah daya tarik pada situs yang kita buat agar lebih menarik dan dinamis.
3. Elemen body <body></body> berisi tag-tag untuk isi atau layout tampilan situs kita, seperti: , <table></table>, <form></form>

Jadi jelas bahwa elemen adalah suatu bagian yang besar yang terdiri dari kode tag tersebut. Sedangkan tag hanyalah merupakan bagian dari elemen. *Tag* adalah kode-kode yang digunakan untuk men-setting dokumen HTML. Secara garis besar bentuk umum *tag* adalah sebagai berikut : <tag-awal>TEKS</tag-akhir> Namun ada juga *tag* yang tidak perlu ada penutup seperti <br>, <hr>, <img>, dan lain-lain.

2.2.5.3 Pengenalan PHP

2.2.5.3.1 Sejarah PHP

PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP bernama FI (*Form Interpreted*). Pada saat tersebut PHP adalah sekumpulan *script* yang digunakan untuk mengolah data *form* dari *web*. Perkembangan selanjutnya adalah Rasmus melepaskan kode sumber tersebut dan menamakannya PHP/FI, pada saat tersebut kepanjangan dari PHP/FI adalah *Personal Home Page/Form Interpreter*. Dengan pelepasan kode sumber ini

menjadi *open source*, maka banyak *programmer* yang tertarik untuk ikut mengembangkan PH.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini *interpreter* sudah diimplementasikan dalam 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 nama rilis tersebut menjadi PHP 3.0. 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. Versi ini banyak dipakai sebab versi ini mampu dipakai untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan proses dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Versi ini adalah versi mutakhir dari PHP. Dalam versi ini, inti dari *interpreter* PHP mengalami perubahan besar. Dalam versi ini juga dikenalkan model pemrograman berorientasi objek baru untuk menjawab perkembangan bahas pemrograman kearah pemrograman berorientasi objek. PHP memiliki kelebihan dibandingkan dengan bahasa pemograman lainnya, diantaranya adalah:

1. Bahasa pemrograman php adalah sebuah bahasa *script* yang tidak melakukan sebuah *kompilasi* dalam penggunaanya.
2. *Web Server* yang mendukung php dapat ditemukan dimana-mana dari mulai IIS sampai dengan *apache*, dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan *developer* yang siap membantu dalam pengembangan.
4. Dalam sisi pemahamanan, php adalah bahasa *scripting* yang paling mudah karena referensi yang banyak.

PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin (*linux, unix, windows*) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah sistem.

2.2.5.3.2 Contoh Pemrograman dengan Menggunakan PHP

Untuk menjalankan program PHP ini, komputer harus telah memiliki sebuah sistem yang telah terinstalasi dan terkonfigurasi dengan baik Apache Web Server, PHP, dan *database* MySQL. Ketiganya adalah program *open source* yang tersedia secara gratis di internet dan dapat berjalan di berbagai *platform* (Windows maupun UNIX/Linux). Program untuk menampilkan tulisan *Hello World* yang dibuat dengan menggunakan PHP adalah sebagai berikut:

```
<?php
echo “<b>Hello World</b>”;
?>
```

Kode program tersebut dituliskan pada sebuah *text editor* serta disimpan dengan ekstensi .php (contoh: hello.php) pada direktori C:\xampp\htdocs\, kemudian hasilnya dapat dilihat melalui web browser seperti Internet Explorer, arahkan alamat pada file hello.php yang telah kita buat, misalkan alamatnya adalah http://localhost/Hello.php

Berikut ini adalah cara menyisipkan kode PHP pada halaman HTML biasa :

```
<html>
<head><title></title>
</head>
<body>
<script language="php">
. . . . kode PHP . . . .
</script>
</body>
</html>
```

Cara yang lebih singkat adalah:

```
<?php
  kode PHP
?>
```

Atau bisa juga:

```
<?
  kode PHP
?>
```

Variabel dapat disebut sebagai sebuah tempat penyimpanan data bagi nilai numeris maupun non numeris, agar dapat digunakan pada bagian lain dari *script* program PHP. PHP mendukung berbagai jenis *variabel* yaitu:

a. *Integer* (bilangan bulat)
b. Bilangan *floating point* (presisi tunggal, ganda)
c. *Boolean*
d. *Null* (untuk variabel yang belum diset)
e. *String*
f. *Array*
g. *Object*
h. *Resource*

Jika menggunakan bahasa pemrograman C atau Pascal, maka kita harus mendefinisikan *variabel* tiap elemen, namun pada PHP kita tidak perlu mendefinisikan terlebih dahulu jenis *variabel* sebelum menggunakannya. PHP memiliki kepandaian untuk membedakan jenis *variabel* secara otomatis berdasarkan konteks yang sedang berlaku bagi *variabel* tersebut. Setiap *variabel* dalam PHP selalu dimulai dengan tanda dolar (" $") dan harus dimulai dengan huruf dan dapat diikuti oleh huruf dan angka. Dengan demikian, $warpspeed, $impuls_speed, $LCAR dan $Dilithium1 merupakan suatu *variabel* PHP yang

valid. Setiap *variabel* dalam PHP peka terhadap perbedaan huruf kapital dan non kapital, sehingga $subspace, $SubSpace, dan $SUBSPACE yang berbeda. adalah tiga buah *variabel*

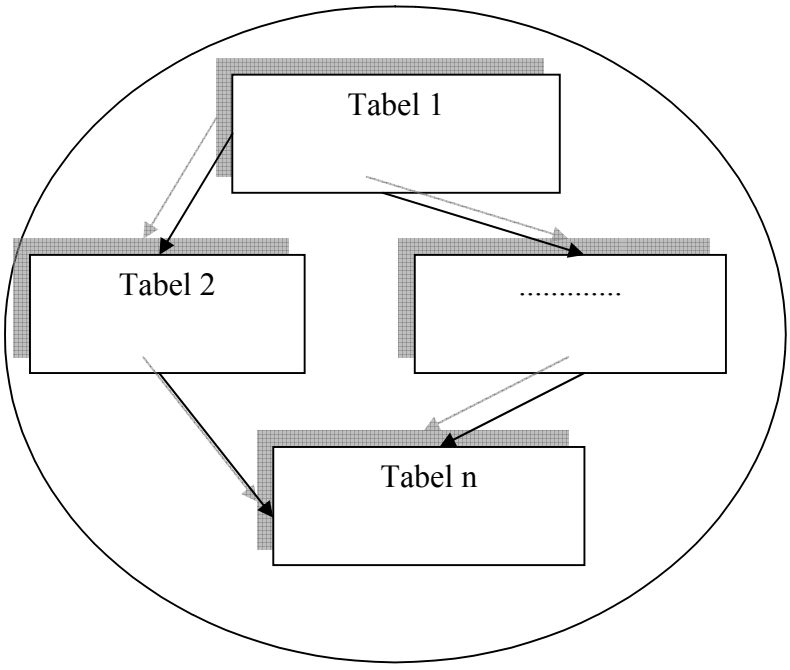
2.2.5.4 Pengenalan Basis data

Sistem Basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yag sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat. Sistem informasi tidak dapat dipisahkan dengan kebutuhan akan basis data apapun bentuknya, baik itu berupa file teks ataupun Database Management System.

Kebutuhan Basis data dalam sistem informasi meliputi :

1. Memasukkan, menyimpan dan mengambil data.
2. Membuat laporan berdasarkan data yang telah disimpan.

Dalam menggunakan basis data relasional yang diimplementasikan dengan tabel-tabel yang saling memiliki relasi seperti pada gambar berikut ini :

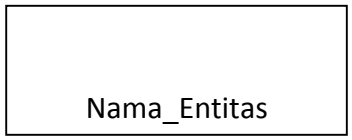
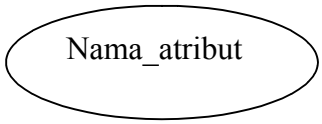
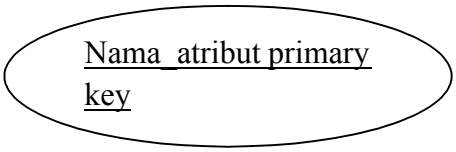
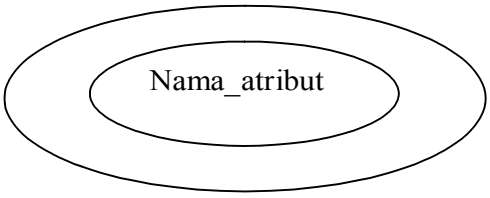


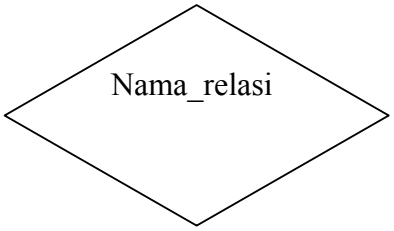
Gambar 2.5 Contoh Relasi antar Tabel

2.2.5.4.1 Entity Relationship Diagram (ERD)

Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan Entity Relationship Diagram (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional sehingga jika penyimpanan basis data menggunakan OODBMS maka perancanga basis data tidak perlu menggunakan ERD. Berikut adalah simbol-simbol yang digunakan pada ERD :

Tabel 2.3 Simbol ERD

Simbol	Deskripsi
Entitas / *Entity* 	Entitas merupakan data inti yang akan disimpan pada basis data.
Atribut 	Field atau kolom data yang perlu disimpan dalam suatu entitas.
Atribut *primary key* 	Field atau kolom data yang perlu disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan biasanya berupa id.
Atribut multinilai 	Field atau kolom data yang perlu disimpan dalan suatu entitas yang dapat memiliki nilai lebih dari satu.
Relasi	Relasi yang menghubungkan antar

	entitas biasanya diawali dengan kata kerja.
Asosiasi 1 0..* _____	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki *multiplicity kemungkinan jumlah pemakaian.*

2.2.5.4.2 Pengertian *Database* MySQL

Database merupakan sekumpulan data yang saling terintegrasi satu sama lain dan terorganisasi berdasarkan sebuah skema atau struktur tertentu dan tersimpan pada sebuah hardware komputer. *Database* terdiri dari beberapa tabel yang saling terorganisasi. Tabel digunakan untuk menyimpan data dan terdiri dari baris dan kolom. Setiap pemakai atau user yang diberikan wewenang yang dapat melakukan akses terhadap data tersebut.

MySQL merupakan sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License) . Setiap pengguna dapat secara bebas menggunakan MySQL. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya; SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Kelebihan dari MySQL diantaranya sebagai berikut :

a. *Portability* yaitu MySQL dapat digunakan dengan stabil tanpa kendala, berarti pada berbagai sistem operasi diantaranya seperti Windows, Linux, Mac OS X Server, Solaris, Amiga HP-UX dan masih banyak lagi.
b. *Open source* yaitu MySQL didistribusikan secara *open source* di bawah lisensi GPL, sehingga dapat memperoleh menggunakannya secara cuma-cuma tanpa dipungut biaya sepeserpun.
c. *Multiuser* yaitu MySQL dapat digunakan untuk menangani beberapa user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik. Hal ini akan memungkinkan sebuah *database server* MySQL dapat diakses *client* secara bersamaan dalam waktu yang bersamaan pula.
d. *Performance Tuning* yaitu MySQL memiliki kecepatan yang cukup menakjubkan dalam menangani *query* sederhana, serta mampu memproses lebih banyak SQL persatuan waktu.
e. *Column Types* yaitu MySQL didukung tipe kolom (tipe data) yang sangat kompleks.
f. *Command dan Functions* yaitu MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah SELECT dan WHERE dalam query.
g. *Scalability dan Limits* yaitu dalam hal batas kemampuan, MySQL terbukti mampu menangani *database* dalam skala yang besar dengan jumlah record lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. Selasin itu batas indeks yang dapat ditampung mencapai 32 indeks pada setiap tabelnya.
h. *Interface* yaitu sama halnya dengan *software database* lainnya, MySQL memiliki *interface* (antarmuka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Aplication Programming Interface).*
i. *Struktur tabel* yaitu Struktur tabel MySQL cukup baik, serta cukup fleksibel. Misalnya ketika menangani Alter Table, dibandingkan database lainnya semacam ProgresSQL ataupun Oracle.

2.2.5.5 Pengenalan Dreamweaver 8

Macromedia Dreamweaver 8 merupakan sebuah editor HTML profesional untuk mendesain secara visual dan mengolah sebuah situs *website* maupun halaman suatu *website*. Versi terbaru di tahun 2005 adalah *Macromedia Dreamweaver* 8 yang merupakan penyempurnaan dari versi sebelumnya dan semakin mudah dalam penggunaannya. *Macromedia Dreamweaver* 8 mendukung pemrograman *Client Side* yang terkenal, yaitu *JavaScript. Macromedia Dreamweaver* 8 juga mendukung pemrograman *Script Server Side*, seperti *Personal Home Page (*PHP), *Active Server Pages* (ASP), *ASP.NET.ColdFusion* dan *Java Server Page* (JSP). Seperti halnya dengan program-program lain, untuk menjalankan Macromedia Dreamweaver 8, mulailah dengan memilih tombol *Start* pada *taskbar*, kemudian pilih *All Program* pada tampilan Start Menu Program, pilih folder *Macromedia* dan klik *Macromedia Dreamweaver 8*. Pada saat pertama kali menggunakan Dreamweaver setelah penginstalan, akan muncul tampilan kotak dialog pilihan layout ruang kerja yang akan digunakan.

Komponen yang terdapat pada ruang kerja Dreamweaver 8 adalah:

1. *Document Window* berfungsi untuk menampilkan dokumen yang sedang bekerja.
2. *Menu Utama* berisi semua perintah yang dapat digunakan untuk bekerja pada Dreamweaver.
3. *Insert Bar* terdiri dari tombol-tombol untuk menyisipkan berbagai macam objek seperti hyperlink, image/gambar, tabel, dan lain-lain.
4. *Document Toolbar* berisi tombol-tombol dan menu pop-up dari dokumen yang sedang digunakan.
5. *Panel Groups* adalah sekumpulan panel window yang saling berkaitan satu sama lain yang dikelompokkan di bawah satu judul untuk membantu bekerja dengan Dreamweaver.
6. *Tag Selector* adalah tag-tag HTML yang terpilih sesuai dengan objek yang ada di area kerja.

7. *Properti Inspector* digunakan untuk melihat dan mengubah berbagai properti dari tiap objek terpilih yang ada di area kerja. Setiap objek memiliki properti yang berbeda-beda.