BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan perusahan

Tinjuanan Perusahaan adalah untuk mengetahui keadaan diperusahaan diantaranya adalah sejarah perusahaan dan struktur organisasi perusahaan

2.1.1 Sejarah Perusahan

Counter adalah sebuah perusahaan dalam bidang penjualan, counter tersebut menjual berbagai macam-macam produk. Counter di sini adalah untuk penjualan handphone yang berbagai macam merek berserta kartu perdananya, di counter tesebit menjual voucher elektronik maupun fisik beserta aksesorisnya.

Counter Amadjid Cell Bandung berdiri pada tahun 2004, berawal dari penjualan voucher bergabung dengan counter yang lebih besar, kemudian pada tahun 2005 counter tersebut mulai berkembang dan banyak pelanggan yang sering membeli di counter tersebut dan meminta untuk bekerja sama.

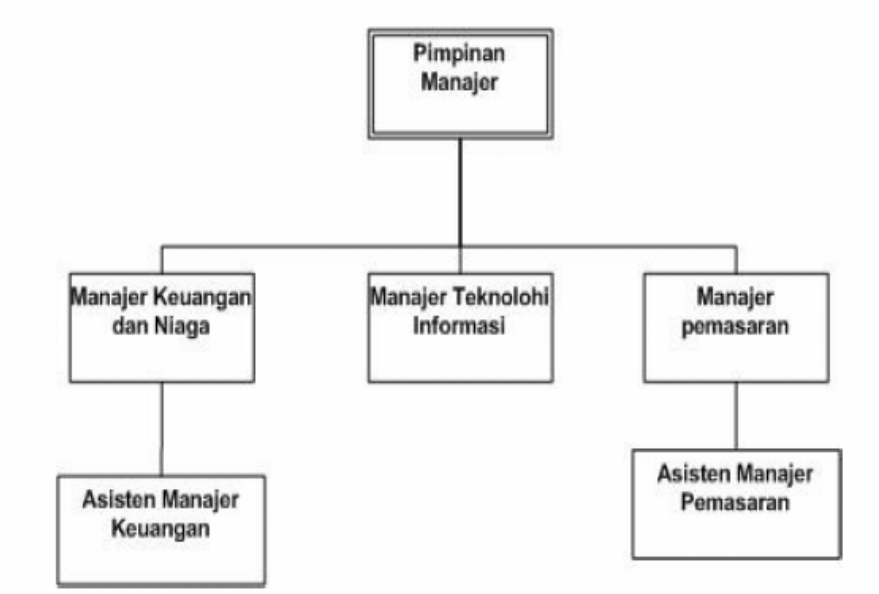
Perkembangan Counter Amadjid Cell Bandung semakin lama semakin maju sehingga perusahaan tersebut berdiri sendiri hingga sekarang.

2.1.2 Struktur Organisasi

Untuk memudahkan pendistribusian data, kewajiban, kewenangan, dan tanggung jawab dari setiap posisi pada perusahaan dalam bidang penjualan voucher dan handphone, maka sebuah perusahaan dalam bidang penjualan voucher dan handphone tentunya memerlukan sebuah struktur organisasi.

Sinergis antara lini dalam struktur organisasi merupakan kunci kesuksesan perusahaan dalam mengeksploitasi dan mengintegritasikan visi dan misi

perusahaan. Tujuan perusahaan akan sia-sia apabila struktur organisasi tidak tertata dengan baik, jadi struktur organisasi akan hanya berfungsi apabila ada transformasi yang baik antara tujuan perusahaan dengan setiap unit struktur organisasi.



Gambar 2.1 Struktur Organisasi

2.1.2.1 Deskripsi Jabatan

Berikut uraian tugas dan wewenang dari struktur organisasi Amadjid Cellular Bandung:

1. Manajer

Bertanggung jawab kepada Pemilik Modal dan mempunyai tugas dan wewenang untuk mengelola dan menjalankan tugas sebagai pimpinan perusahaan.

2. Manajer Keuangan

Manajer Keuangan dan Niaga mempunyai wewenang sebagai berikut:

1. Mengembangkan sistem manajemen keuangan yang berlaku di Perusahaan,
2. Menyusun kebijakan pengelolaan keuangan yang efektif dan efisien.
3. Menyususn rencana pendapatan dan biaya sesuai dengan *Business Plan Company.*

Manajer Keuangan dan Niaga dibantu oleh satu orang asisten yaitu **Asisten Manajer Keuangan,** yang bertanggung jawab atas pengelolaan keuangan dari anggaran serta menyiapkan informasi akuntasi untuk kepentingan manajemen.

3. Manajer Teknologi Informasi

Mempunyai tugas utama jabatan sebagai penanggungjawab atas seluruh kegiatan *engineering* yang meliputi fungsi rencana kerja, kajian pengembangan perangkat lunak, dan teknis implementasi teknologi jaringan. Manajer *engineering* mempunyai wewenang sebagai berikut :

1. Menyusun rencana pengembangan.
2. Mengkaji pengembangan perangkat lunak.
3. Menentukan atau membuat *design* yang berhubungan dengan sistem pengembangan perangkat lunak, instalasi dan implementasi teknologi jaringan dan keamanan sistem.

4. Manajer Pemasaran

Mempunyai tugas utama jabatan yaitu melakukan proses strategi pemasaran produk dan riset tentang kemungkinan produk yang akan dikembangkan nantinya. Selain itu, menyusun kegiatan pemasaran untuk memenuhi kegiatan dan kepuasan pelanggan.

2.2 E-Commerce

2.2.1. Pengertian E-Commerce

Menurut David Baum pengertian *e-commerce* dapat kita artikan sebagai berikut :

" E-Commerce adalah satu set dinamis teknologi, aplikasi dan proses bisnis yang menghubungkan perusahaan, pelanggan dan komunitas tertentu melalui transaksi elektronik dan perdagangan barang, pelayanan dan informasi yang dilakukan secara elektronik, jika dilihat dari segi ekonominya."

Selain dari pernyataan diatas penulis mencoba mengartikan *e-commerece* itu merupakan suatu aplikasi dari teknologi *online* (internet) pada bidang komersil.

2.2.2. Internet dan E-Commerce

Dalam beberapa waktu terakhir ini, dengan begitu merebaknya media internet dimana-mana khususnya di Indonesia, dengan semakin menjamurnya warnet yang bertebaran hampir seperti wartel, ternyata telah membuat banyak perusahaan mulai mencoba menawarkan berbagai macam produk-produk mereka dengan menggunakan media ini.

Kehadiran internet yang masih merupakan industri baru yang dalam fase pertumbuhan dan masih terus berubah serta penuh ketidakpastian telah memperkokoh keyakinan akan pentingnya peranan teknologi dalam pencapaian

tujuan *financial* perusahaan melalui modifikasi dan efisiensi proses bisnis, yaitu dengan memanfaatkan *E-Commerce*. Kemanapun internet untuk menjangkau pelanggan baru dan penghematan biaya yang cukup signifikan untuk distribusi dan pelayanan merupakan keuntungan yang bisa didapat perusahaan dengan memindahkan roda nilai *commerce* ke media *internet*.

Perkembangan teknologi internet yang sangat cepat berubah menjadikan strategi atau model bisnis hari ini mungkin tidak lagi relevan untuk diterapkan besok. Didalam lingkungan internet yang kompetitif ini, disamping perencanaan yang baik serta harapan yang realistis terhadap teknologi baru, juga diperlukan kesiapan infrastruktur yang cukup memadai dan visi yang komprehensif untuk bisa berjalan melewati ketidakpastian dan perubahan yang begitu cepat.

E-Commerce merupakan salah satu keunggulan dari internet, hingga akhirnya di era sekarang ini (banyak disebut dengan era digital) nampaknya tiada hari yang terlewatkan tanpa mendengar atau membaca kata *E-Commerce* di berbagai media informasi.

Sebenarnya apakah yang dimaksud dengan *e-commerce* itu ? sebenarnya dalam *e-commerce* banyak sebutan yang dipakai untuk memudahkan orang untuk mengucapkannya. Ada beberapa sebutan untuk *e-commerce* yaitu *commerce* atau *Ecom* atau *E-commerce* atau *Immerce,* yang pada dasarnya semua sebutan diatas mempunyai makna yang sama. Istilah-istilah tersebut berarti membeli atau menjual secara elektronik, dan kegiatan ini dilakukan pada jaringan internet. *E-Commerce* juga dapat berarti pemasangan iklan web shop 24 jam sehari bagi seluruh pelanggannya.

Sebuah web shop (toko pada web) dapat dijalankan sementara pelaku bisnis dapat melakukan aktivitas lain (misalnya sambil beristirahat atau tidur) dan tidak terbatas pada letak geografisnya.

Jika anda bergerak dalam bidang retailer yang berkembang dengan pesat, anda mungkin akan merasa khawatir untuk pergi dari belakang kursi yang *online* ke .com *e-commerce* anda yang sibuk transaksi. Hal yang perlu diketahui disini ialah bahwa selama web melakukan kegiatan perdagangan tidak ada gugatan terhadap setiap produk perusahaan jika anda sudah siap menjalankan sebuah mail *order service* (pelayanan order melalui surat) atau jika pelanggan menempatkan ordernya via *telephone* atau faksimili. Dengan *E-Commerce* segalanya dapat berkembang secara alami.

2.2.3. Elemen-elemen *E-Commerce*

Yang termasuk dalam elemen-elemen *e-commerce* yaitu :

1. *Networking* atau jaringan merupakan beberapa komputer yang saling berhubungan dan saling tukar menukar informasi dan terkoneksi melalui sebuah kartu jaringan dan jalur komunikasi, yang terdiri dari jaringan local atau lebih dikenal dengan LAN dan jaringan internet yang banyak digunakan untuk jaringan LAN atau lokal terdapat berbagai jenis jaringan seperti jaringan bus, token, star dan yang lain yang cocok untuk digunakan, selain itu juga jaringan merupakan bagian yang penting untuk menentukan baik atau tidaknya suatu jaringan, untuk dapat diakses dalam jaringan baik jaringan lokal maupun lewat jaringan internet.

2. *Security* (keamanan)

   Security atau keamanan merupakan bagian yang penting, karena menyangkut masalah keamanan data *user* dan juga keamanan server kita, masalah seperti enkripsi data pada saat transaksi memakai kartu kredit, aplikasi yang sudah banyak dipakai yaitu SSL (*Scure Socet Layer*).

3. *Web programming* dan *web design*

   Kedua masalah ini dalam pelaksanaannya dapat dikerjakan oleh satu orang atau dibagi dua, untuk *web programming* dikhususkan dalam pembuatan dalam bahasa pemrograman dan untuk *web design*, khusus untuk mendesain halaman web atau pengaturan gambar, warna maupun tata letak suatu web dari segi keindahan dan dinamis, dalam *web programming* biasanya digunakan script server seperti PHP, ASP, CGI dan yang lainnya. Untuk designnya banyak software-software yang mendukung seperti Macromedia, Photoshop, Frontpage, Office publisher, dan masih banyak lagi yang lainnya.

4. Business *online*

   Maksud dari business *online* disini yaitu menyangkut bagaimana cara pengelolaan perusahaan mulai dari manajemen, administrasi keuangan dan lainnya, yang semua itu merupakan suatu strategi dalam menarik customer atau pelanggan untuk datang ke toko kita.

5. *Cyberlaw*

   Cyberlaw merupakan hukum yang mengatur pelanggan di dunia internet, seperti mencuri nomor kartu kredit yang merupakan kejahatan internet. *Cyberlaw* ini harus didukung oleh pihak yang berwajib dinegara kita, karena apabila dinegara kita belum diterapkan sistem *cyberlaw*, maka para pelanggan akan berpindah tempat untuk menjaga keamanan mereka.

2.3. Internet

2.3.1. Pengertian Internet

Menurut Robert J. Verzello yang diterjemahkan oleh Jogiyanto HM (1998:23)

"Internet adalah suatu jaringan komputer global yang terbentuk dari jaringan-jaringan komputer lokal dan regional, dengan adanya jaringan ini memungkinkan komunikasi data antar komputer-komputer yang terhubung ke jaringan tersebut".

2.3.2. Layanan Aplikasi di Internet

Terdapat banyak sekali layanan aplikasi di internet dan masih terus akan berkembang sejalan dengan perkembangan teknologi informasi, dalam laporan ini hanya akan dibahas beberapa contoh aplikasi yang banyak digunakan saja, diantaranya adalah :

1. *Electronic mail (E-Mail)*

Aplikasi ini adalah yang paling banyak digunakan, dan termasuk salah satu dari aplikasi pertama di Internet. Dengan E-Mail, anda dapat mengirim dan

menerima surat, pesan, dokumen secara elektronik dengan pemakai lain di Internet yang mempunyai alamat e-mail.

2. *News-USENET*

Digunakan sebagai sarana untuk berdiskusi antar pemakai jaringan Internet. Aplikasi ini hampir serupa dengan suatu papan pengumuman, dimana setiap orang dapat mengirim, melihat dan menanggapi suatu berita atau suatu topik diskusi dengan fasilitas yang hampir sama dengan e-mail. Topik diskusi dipisahkan oleh group, dan pemakai yang berminat dapat melihat isi diskusi pada newsgroup tersebut.

3. *File Transfer Protocol (FTP)*

FTP merupakan suatu protocol untuk aplikasi pengiriman data berupa file, Dengan adanya aplikasi ini, dimungkinkan untuk *upload* da*n download* data dalam format data berbentuk file seperti misalnya data aplikasi, gambar, database dan sebagainya.

4. *Remote Login – Telnet*

Telnet adalah suatu aplikasi remote login Internet yang memungkinkan anda untuk *log-in* atau menggunakan komputer yang berbeda pada jaringan secara interaktif. Untuk *login* dibutuhkan login *account* pada komputer tujuan, jika anda bukan user terdaftar maka tidak dapat *login* ke komputer tersebut. Aplikasi ini biasanya digunakan untuk mengakses komputer berbasis sistem operasi UNIX dari tempat yang berbeda dari servernya.

5. *World Wide Web (WWW)*

Awalnya informasi dapat dicari pada internet dengan menggunakan fasilitas *information service* berbasis *archie*, *gopher* dan *WAIS (Wide Area Information System).* Pencarian informasi berdasarkan menu-menu pada sistem-sistem tersebut dan output yang dihasilkan berbasis teks. Saat ini dengan teknologi *World Wide Web*, dimungkinkan untuk mengakses informasi secara interaktif, dan bentuk informasinya berupa tampilan grafis maupun teks. Hal ini dimungkinkan dengan adanya *Hypertext Transfer Protocol* (HTTP) yang digunakan untuk mengakses suatu informasi yang disimpan pada suatu situs web(*website*). Untuk dapat menggunakan sarana ini, dibutuhkan aplikasi *Web Browser.*

6. Universal Resource Locater (URL)

Pada penggunaan *World Wide Web*, penunjukan suatu sumber informasi menggunakan metode *Universal Resource Locater* (URL), yang merupakan konsep penamaan lokasi standar dari suatu file, direktori, komputer, lokasi komputernya dan metoda yang digunakan. URL tidak hanya dapat menunjuk ke suatu file tapi dapat juga menunjuk suatu query, dokumen dalam suatu database, atau hasil dari perintah atau yang lainnya. Dengan URL ini didefinisikan lokasi dan metoda pengaksesan file tersebut.

7. *Hyper Text Transfer Protocol* (HTTP)

Web merupakan terobosan baru bagi teknologi system informasi yang menghubungkan data dari banyak sumber dan layanan yang beragam macamnya di internet. Pengguna tinggal mengklik tombol mousenya pada link-link *hypertext*

yang ada untuk melompat ke dokumen-dokumen di berbagai lokasi di internet. Link-link sendiri bisa mengacu kepada dokumen web, server FTP *(File Transfer Protokol)*, e-mail ataupun layanan-layanan lainnya.

Server dan browser web berkomunikasi satu sama lain dengan protocol yang memang dibuat khusus untuk ini, yaitu HTTP. HTTP bertugas menangani permintaan-permintaan *(request)* dari browser untuk mengambil dokumen-dokumen web.

HTTP bisa dianggap sebagai system yang bermodel *Client-Server*. Browser web sebagai *Clientnya*, mengirimkan permintaan kepada *Server Web* untuk mengirimkan dokumen-dokumen web yang dikehendaki pengguna. *Server Web* lalu memenuhi permintaan ini dan mengirimkannya melalui jaringan kepada *browser*. Setiap permintaan akan dilayani dan ditangani sebagai suatu koneksi terpisah yang berbeda.

2.4. *Hyper Text Markup Language* (HTML)

HTML dewasa ini dikenal sebagai bahasa standar untuk membuat dokumen web. Sesungguhnya *Hypertext Markup language* (HTML) justru tidak dibuat untuk mempublikasikan informasi di web, namun oleh karena kesederhanaan serta kemudahan penggunaannya, HTML kemudian dipilih orang untuk mendistribusikan informasi di web.

Perintah perintah HTML diletakkan dalam file berekstensi *.html dan ditandai dengan mempergunakan *tag* (tanda) berupa karakter “< “dan “>”. Tidak seperti bahas pemrograman berstruktur seperti pascal atau C, HTML tidak

mengenal *jumping* atau *looping*. Kode-kode html dibaca oleh *browser* dari atas kebawah tanpa adanya lompatan-lompatan.

Struktur sebuah dokumen HTML pada dasarnya dibagi menjadi dua bagian besar, yaitu *header* dan body. Masing-masing ditandai oleh pasangan *container tag* <head> dan <body>. Bagian *head* berisikan judul dokumen dan informasi-informasi dasar lainnya sedangkan bagian *body* adalah data dokumennya. Pengaturan format teks dan pembentukan *link* dilakukan terhadap objeknya langsung dengan ditandai oleh *tag-tag* HTML.

HTML diatur oleh konsorsium WWW (W3C) semua perubahan atas standar bahasa HTML harus disahkan terlebih dahulu oleh WWW. Sejauh ini, HTML telah mengalami berbagai revisi sepanjang hidupnya, standar paling akhir yang sekarang diperkenalkan adalah standar HTML 4.0, yang mendukung antara lain CSS *(Cascading Style Sheet)*, *Dynamic Content Positioning* (Penempatan isi secara dinamis), hingga kini, tidak semua browser web telah disesuaikan untuk mendukung standar HTML terbaru ini.

2.5 Pengertian Basis data

Basis data dapat diartikan sebagai kumpulan dari beberapa file yang sejenis. *Menurut Hannafin dan Peck (1998:76) basis data mengandung arti "Sekumpulan data yang saling berhubungan dan berkaitan satu dengan yang lainnya digunakan oleh suatu organisasi"*.

Dalam hal ini data dan informasi merupakan kesatuan yang saling berhubungan dan bekerjasama yang terdiri dari : peralatan, tenaga pelaksana, dan prosedur data. Sehingga pengolahan data ini membentuk sistem pengolahan data.

Peralatan dalam hal ini berupa perangkat keras (*hardware*) yang digunakan, tenaga pelaksana yaitu bagian keuangan yang menggunakan sistem informasi ini, dan prosedur data yaitu berupa perangkat lunak (*software*) yang digunakan dan dipakai untuk dialokasikan dalam pembuatan sistem informasi pengolahan *database*.

2.5.1 Sistem Basis Data

Sistem basis data merupakan sistem yang terdiri dari sekumpulan *table* yang saling berhubungan (dalam sebuah basis data di sebuah sistem komputer) dan sekumpulan program (DBMS) yang memungkinkan beberapa pemakai lain untuk mengakses dan memanipulasi tabel-tabel tersebut.

Sistem basis data pada dasarnya dibuat untuk tujuan sebagai berikut yaitu:

1. Kecepatan dan kemudahan

   Dengan sistem basis data diharapkan pengolahan data dapat lebih cepat dan lebih mudah dibandingkan dengan system manual

2. Effisiensi ruang penyimpanan

   Dengan basis data, efisiensi/optimalisasi penggunaan ruang penyimpanan dapat dilakukan, karena kita dapat melakukan penekanan jumlah redundansi data.

3. Keakuratan

   Pemanfaatan pengkodean atau pembentukan relasi antar data bersam a dengan penerapan atauran atau batasan tipe data, domain data, keunikan data, dan sebagainya yang secara ketat dapat diterapkan dalam sebuah basis data, sangat berguna untuk menekan ketidakakuratan masukan ataupun penyimpanan data.

4. Ketersediaan

   Dengan pertumbuhan data yang semakin banyak, maka diperlukan tempat penyimpanan yang sangat besar, karena itu kita perlu memilah data dalam bentuk kategori-kategori tertentu sehingga data yang tidak terlalu penting dapat dihapus.

5. Kelengkapan

   Untuk mengakomodasi kebutuhan kelengkapan data yang semakin berkembang maka kita tidak hanya dapat menambah *record-record* data, tetapi juga melakukan perubahan struktur dalam basis data, baik dalam bentuk penambahan objek baru (tabel) atau dengan penambahan field-field baru pada suatu tabel.

6. Keamanan

   Dengan sistem basis data kita dapat memproteksi semua objek yang ada dengan berbagai tingkatan kemanan.

7. Kebersamaan pemakain

   Pemakai basis data biasanya tediri dari banyak pemakai, sehingga hendaknya basis data harus bisa diakses oleh banyak pemakai dalam waktu yang bersamaan.

Dalam sebuah sistem basis data secara lengkap akan terdapat komponen-komponen utama, yaitu:

1. *Hardware* (Perangkat keras)
2. *Operating system* (Sistem Operasi)
3. *Database* (Basis Data)
4. *System* (Aplikasi/perangkat lunak) pengelola basis data (DBMS)
5. *User* (Pemakai)

2.5.2 Model Hubungan (*Entity Realtionship Model*).

Entity Relationship Model didasarkan pada persepsi dunia nyata yang terdiri dari himpunan obyek dasar yang disebut entitas dan relasi antar entitas. Entitas adalah objek yang dapat diidentifikasi secara unik. Entitas dikarakterisasi dan dipresentasikan dengan suatu gugus atribut.

Sekelompok entitas yang memiliki karakterisasi entitas disebut gugus entitas (*entity set*). Setiap entiti dari gugus tersebut disebut anggota gugus (*member of set*).

Berdasarkan jumlah gugus yang terlibat maka relasi antar entiti dibedakan menjadi :

a. Relasi biner (*binary*), yaitu relasi antar 2 gugus entitas.
b. Relasi trio (*ternary*), yaitu relasi antar 3 gugus entitas.
c. Relasi N-ary, yaitu relasi antar n gugus entitas.

Khusus untuk relasi biner maka relasi antar anggota dari dua gugus yang terlibat (kardinalitas relasi biner) dapat bersifat :

a. Relasi 1-1 (*one-to-one relationship*).

Adalah satu entiti anggota gugus diasosiasikan dengan tepat satu entiti anggota gugus yang lain.

b. Relasi 1-banyak (*one-to-many relationship*).

Adalah satu entiti anggota gugus diasosiasikan dengan satu atau lebih entiti anggota gugus yang lain. Sebaliknya satu entiti anggota gugus yang lain tersebut diasosiasikan dengan tepat satu entiti anggota gugus pasangannya.

c. Relasi banyak-1 (*many-to-one relationship*).

Adalah satu entitas anggota gugus diasosiasikan dengan satu atau lebih entiti anggota gugus yang lain dan berlaku pula sebaliknya.

Terdapat beberapa pengertian tentang key sehubungan dengan normalisasi dan ERD, antara lain :

a. *Superkey* adalah gugus dari sejumlah atribut entitas yang dapat digunakan untuk mengidentifikasi obyek secara *unik*.

b. *Candidate key* adalah superkey dengan jumlah atribut minimal dan dapat berdiri sendiri.

c. *Primary key* adalah superkey yang dipilih oleh desainer atau administrator basis data.

2.5.3.1 Normalisasi

Adalah proses yang berkaitan dengan model data relational untuk mengorganisasi himpunan data dengan ketergantungan dan keterkaitan yang tinggi atau erat. Hasil dari proses normalisasi adalah himpunan-himpunan data dalam bentuk normal (*normal form*). Ada beberapa bentuk normal, yaitu :

a. Bentuk Normal I (*First Normal Form* / 1-NF).

Suatu relasi memenuhi 1-NF jika dan hanya jika setiap atribut dari relasi tersebut hanya memiliki nilai tunggal dalam satu baris atau record.

b. Bentuk Normal II (*Second Normal Form* / 2-NF).

Bentuk Normal tahap I terpenuhi, dan semua atribut yang bukan memiliki *dependensi* sepenuhnya terhadap kunci *primer*.

Notasi Dependensi Total :

$$X \longleftrightarrow A$$

c. Bentuk Normal III (*Third Normal Form* / 3-NF).

Berada Pada bentuk Normal II, dan setiap atribut bukan kunci tidak memiliki *dependensi transitif* terhadap kunci *primer.*

Notasi dependensi transitif : $X \rightarrow Y$, $Y \rightarrow Z$ Maka: $X \rightarrow Z$

d. Bentuk Normal *Boyce-Codd* (*Boyce-Codd Normal Form* / BCNF).

Suatu relasi memenuhi BCNF jika dan hanya jika setiap determinan yang ada pada relasi tersebut adalah kunci kandidat (*candidate keys*). Determinan adalah gugus atribut dimanaa satu atau lebih atribut lain tergantung secara fungsional.

Kegunaan normalisasi :

1. Meminimasi pengulangan informasi.
2. Memudahkan indentifikasi entitas / obyek.

2.5.4 Diagram ER

Diagram ER digunakan untuk membantu proses perancangan basis data

Notasi yang digunakan berupa:

1. Persegi panjang yang menyatakan himpunan entitas
2. Lingkaran/ elips yang menyatakan atribut
3. Belah ketupat yang menyatakan himpunan relasi
4. Kardinalitas yang dapat dinyatakan dengan garis cabang atau angka
5. Garis sebagai penghubung antara himpunan entitas dengan himpunan entitas yang lain

2.5.5 *Flow map* (Diagram alir)

Flow map merupakan diagram alir yang menunjukan arus bagi dokumen, aliran data fisik entitas-antitas sistem informasi dan kegiatan operasi yang berhubungan dengan sistem informasi. Penggambaran biasanya diawali dengan mengapati dokumen apa yang menjadi media data atau informasi dan selanjutnya ditelusuri bagaimana dokumen tersebut termasuk ke bagian atau entitas mana dokumen tersebut, proses apa yang terjadi terhadap dokumen tersebut dan seterusnya.

2.5.6 *Data Flow Diagram* (Diagram alir data)

Data flow diagram adalah representasi graphis dari suatu sistem yang menggambarkan komponen-komponen sebuah sistem, aliran data diantara komponen-komponen tersebut beserta asal, tujuan dan penyimpanan datanya.

Data flow diagram yang harus digambarkan dalam analisis maupun perancangan system adalah sebagai berikut :

1. *Context diagram* adalah *data flow diagram* tingkat paling atas dari sebuah sistem informasi yang menggambarkan keseluruhan proses dalam suatu sistem.
2. *Data flow diagram level physic* (diagram alir data level fisik) adalah data *flow diagram* yang menggambarkan entitas internal dan eksternal dalam suatu sistem beserta aliran dan tempat penyimpanan datanya.

DFD memiliki 2 (dua) macam bentuk, yaitu *physical DFD* (DFD fisik) dan *logical DFD* (DFD logis). DFD fisik lebih menekankan pada bagaimana proses dari sistem diterapkan, sedangkan DFD logis lebih menekankan pada proses-proses apa yang terdapat dalam sistem.

Diagram alir data level fisik ialah menunjukan dimana bagaimana dan oleh siapa proses-proses dalam sistem dilakukan.

1. *Data flow diagaram level logic* adalah *data flow diagram* yang menggambarkan fungsi atau proses yang dilakukan setipa entitas.
2. Diagram alir data level *logic* lebih menunjukan dimana dan oleh siapa proses tersebut dilakukan.

2.5.7 *Data dictionary* (kamus data)

Data dictionary adalah daftar organisasi semua elemen yang ada dalam sistem secara lengkap dengan definisi yang baku sehingga user dan analisis sistem akan memiliki pengertian yang sama untuk input, output, komponen penyimpanan dan penghitungannya.

Kamus data dapat digunakan pada saat analisis sistem atau pada saat perancangan sistem pada saat entitas sistem, kamus data digunakan untuk

mencatat terminologi bisnis, aturan standar (batasan panjang karakter, nilai, *system field).*

Untuk membuat spesifikasi elemen data, digunakan notasi struktur data yaitu :

Notasi	**Keterangan**
=	Terdiri dari, sama dengan, diuraikan
+	dan
()	Pilihan, boleh atau tidak
N { } M	Iterasi atau pengukuran mulai N kali sampai M kali
[ ]	Pilih salah satu pilihan
|	Pemisalan dalam notasi [ ]
*	Keterangan, komentar atau saran
@	*Key field*

2.6 *Structure Query Language* (SQL)

SQL adalah bahasa standar yang digunakan oleh *Database Management System* untuk memanipulasi dan memperoleh data dari sebuah database relasional. Dengan SQL kita dapat melakukan hal-hal berikut terhadap basisdata :

a. Pengubahan struktur sebuah *database*

b. Pengubahan pengaturan keamanan sistem

c. Memberikan hak akses kepada pengguna

d. Memperoleh informasi dari *database*

e. Mengupdate database

Perintah-perintah SQL dikelompokkan menjadi lima macam yaitu:

1. *Data Definitions Language* (DDL)

   Adalah perintah SQL yang digunakan untuk mendefinisikan kerangka basisdata, perintah-perintahnya adalah:

   a. *create* : untuk membuat atau menciptakan objek basisdata
   b. *alter* : untuk memodifikasi atau mengubah objek basisdata
   c. *drop* : untuk menghapus objek basisdata
   d. objek *database* yang dimaksud adalah basisdata, *table*, index

2. *Data Manipulations Language* (DML)

   Adalah perintah yang digunakan untuk mengoperasikan atau memanipulasi isi basisdata, SQL menyediakan 4 perintah DML yaitu:

   a. *select* : digunakan untuk mengambil data dari basisdata
   b. *delete* : digunakan untuk menghapus data pada basisdata
   c. *insert* : digunakan untuk menambahkan data kedalam table
   d. *update* : digunakan untuk memodifikasi data pada basisdata

3. Security

   Adalah perintah-perintah yang digunakan untuk menjamin keamanan data. Perintahnya antara lain:

   a. *grant* : digunakan untuk memberikan akses kepada user tertentu ke basisdata
   b. revoke : digunakan untuk mencabut hak akses dari user

4. *Integrity*

   Adalah perintah-perintah yang digunakan untuk menjaga kesatuan data. Contoh: *recover table*: digunakan untuk memperbaiki *table* pada basisdata

5. *Auxilliary*

   Adalah perintah-perintah pelengkap atau tambahan seperti : *unload* dan *rename*.

2.7 Personal Home Page (PHP)

2.7.1 Sejarah PHP

PHP merupakan bahasa pemrograman untuk *script web server-side*. Bahasa pemrograman PHP diciptakan pertama kali oleh Rasmus Lerdorf, seorang pemrogram C yang sangat handal. Semula PHP hanya digunakan untuk mencatat seberapa jumlah pengunjung pada *hompage*-nya. Rasmus adalah salah seorang pendukung *open source*. Karena itulah ia mengeluarkan *Personal Home Page Tools* versi 1.0 secara gratis atau *freeware* pada tahun 1995. Setelah mempelajari YACC dan GNU Bison, Rasmus menambah kemampuan pada PHP 1.0 dan menertbitkan PHP 2.0 sebagai pengembangan dari PHP 1.0 yang telah ada sebelumnya.

PHP 2.0 mampu berhubungan dengan *database* dan dapat diintegrasikan dengan HTML. Pada tahun 1996, PHP telah digunakan oleh banyak *website* di dunia. Sebuah kelompok pengembang software yang terdiri dari Rasmus, Zeew Suraski, Andi Gutman, Stig Bakken, Shane Carveo dan Jim Winstead bekerja selama tujuh bulan untuk melakukan penyempurnan terhadap PHP 2.0, akhirnya pada tanggal 6 juni 1998, PHP 3.0 resmi dikeluarkan ke dunia pemrograman.

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. PHP versi 4.2 telah diterbitkan pada tanggal 22 April 2002 dengan log kelompok fungsi, sampai dengan versi 4.3.7 tercatat 125 kelompok fungsi yang dimiliki oleh PHP. Pada juni 2004, Zend merilis PHP 5.0. Versi ini adalah versi yang paling 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 bahasa pemrograman kearah pemrograman berorientasi objek.

2.7.2 Kelebihan PHP

PHP memiliki beberapa kelebihan yang tidak dimiliki oleh bahasa-bahasa sejenisnya, yaitu:

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.

5. 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.

2.8 *Personal Home Page tool* (PHP)

PHP adalah salah satu bahasa *Server-side* yang didesain khusus untuk aplikasi web. PHP dapat disisipkan diantara bahasa HTML dan karena bahasa *Server side*, maka bahasa PHP akan dieksekusi di *server*, sehingga yang dikirimkan ke *browser* adalah "hasil jadi" dalam bentuk HTML, dan kode PHP anda tidak akan terlihat.

2.8.1 Keistimewaan PHP

Sebagai bahasa skrip yang bersifat server side PHP memiliki beberapa keunggulan antara lain:

1. Tidak diperlukan kompabilitas browser atau harus menggunakan *browser* tertentu karena serverlah yang akan mengerjakan skrip PHP tersebut
2. *Freeware* artinya dapat didistribusikan dengan bebas
3. Memiliki kemampuan koneksi keberbagai macam *database* seperti: MySQL, PostgreSQL, Oracle, dBase, Sybase dan banyak lagi.

2.9. MySQL

MySQL adalah perangkat lunak *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). setiap orang bebas untuk menggunakannya, tapi tidak boleh dijadikan produk turunan yang bersifat *Closed Source* atau komersial.

MySQL sebenarnya merupakan turunan salah satu konsep utama dalam *database* sejak lama, yaitu SQL (*Structur Query Language*). SQL adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Keandalan suatu sistem *database* (DBMS) dapat diketahui dari cara kerja *Optimizer*-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh *user* maupun program-program aplikasinya. Sebagai *database server*, MySQL dapat dikatakan lebih unggul dibanding *database server* lainnya dalam *query* data. Hal ini terbykti untuk *query* yang dilakukan oleh *single user*, kecepatan *query* MySQL bisa sepuluh kali lipat lebih cepat dari PostgreSQL dan Lima kali lebih cepat dibanding Interbase.

2.9.1 Keistimewaan MySQL

Sebagai *database server* yang memiliki konsep *database modern*, MySQL memiliki banyak keistimewaan. Berikut ini beberapa keistimewaan yang dimiliki oleh MySQL.

1. *Portability*

   MySQL dapat berjalan stabil pada berbagai system operasi seperti Windows, Linux, FreeBSD, Mac OS X Server, dan lain-lain.

2. *Open Source*

   MySQL didistribusikan secara *Open Source*, sehingga dapat digunakan secara bebas.

3. *Multi-user*

   MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah. Hal ini memungkinkan sebuah *database server* MySQL dapat diakses *client* secara bersamaan.

4. *Performance Tuning*

   MySQL memiliki kecepatan yang tinggi dalam menangani *query*, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

5. *Column Types*

   MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed/unsigned integer, float, double, char, varchar, text, blob, date, time, timestamp, year, set* dan *enum.*

6. *Command dan Function*

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

7. *Security*

   MySQL memiliki lapisan lapisan sekuritas seperti level subnetmask, nama *host*, dan izin akses *user* dengan *system* perizinan yang mendetail serta *password* terenkripsi.

8. *Scalability dan Limits*

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

9. *Connectivity*

   MySQL dapat melakukan koneksi dengan *client* menggunakan *protocol* TCP/IP, Unix Socet (UNIX), atau Named Pipes (NT).

10. *Localisation*

    Deteksi pesan kesalahan pada *client* dengan menggunakan lebih dari 20 bahasa

11. *Interface*

    Interface terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*)

12. *Client dan Tools*

    Dilengkapi dengan berbagai *Tool* yang dapat digunakan untuk administrasi *database*, dan pada setiap *tool* yang ada disertakan petunjuk *online*.

13. Struktur Tabel

    Struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibanding *database* lainnya semacam PostgreSQL ataupun Oracle.

2.10 Model-model Perangkat lunak

2.10.1 Model Dasar *Squensial* atau *Waterfull*

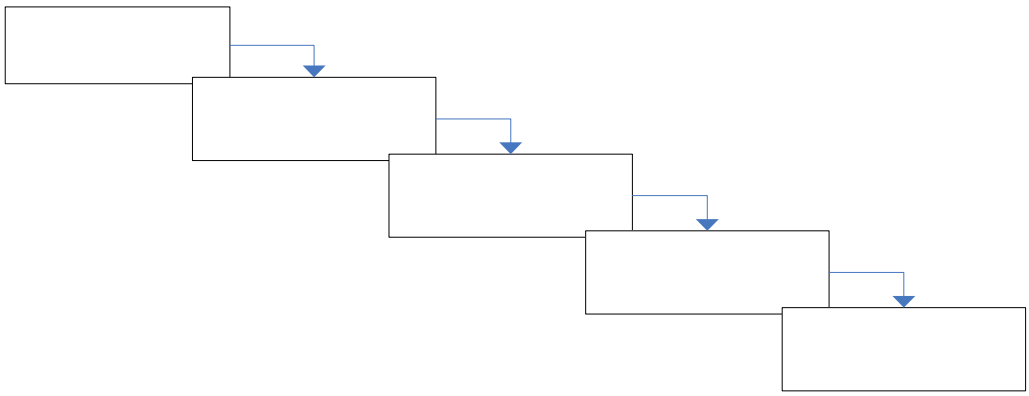
Model Dasar Squensial linier untuk software engineering, sering disebut juga dengan siklus kehidupan klasik atau model air terjun, model ini mengusulkan sebuah pendekatan kepada perkembangan software yang sistematik dan squensial yang mulai pada tingkat dan kemajuan sistem kepada seluruh analisis, desain, kode, pengujian, dan pemeliharaan dimodelkan setelah siklus rekayasa konversional, model squensial, linier melingkupi aktifitas-aktifitas

2.10.2 Model Prototipe

Prototyping paradigma dimulai dengan pengumpulan kebutuhan. Secara ideal prototipe berfungsi sebagai sebuah mekanisme untuk mengidentifikasi kebutuhan perangkat lunak. Bila prototipe yang sedang bekerja dibangun, pengembang harus mempergunakan fragmen-fragmen program yang ada atau mengaplikasikan alat-alat bantu (contohnya *report generator*, *window manager*, dll) yang memungkinkan program yang bekerja untuk dimunculkan secara tepat.

2.10.3 Model RAD

Rapid Aplication Development (RAD) adalah sebuah model proses perkembangan perangkat lunak sekuensial linier yang menekankan siklus perkembangan yang sangat pendek. Pendekatan RAD melingkupi fase-fase, yaitu fase *bussiness modeling, data modeling, process modeling, aplication generation, dan testing and turnover* lihat gambar dibawah ini (Gambar 2.3).



Gambar 2.2 Model RAD

2.11 Bahasa Basis Data

Semua DBMS memiliki dua macam bahasa yang digunakan untuk mengelola dan mengorganisasikan data, yaitu:

1. Bahasa Definisi Data (DDL/*Data Definition Language*)
2. Bahasa Manipulasi Data (DML/*Data Manipulation Language)*

Selain kedua bahasa di atas, pada kebanyakan DBMS sekarang juga terdapat bahasa yang tergolong DCL (*Data Control Language*) yang berkaitan dengan pengaturan sekuritas basis data.

A. Bahasa Definisi Data (DDL)

DDL adalah perintah-perintah yang biasa digunakan oleh administrator basis data (DBA) untuk mendefinisikan skema ke DBMS. Skema adalah deskripsi lengkap tentang struktur medan, rekaman, dan hubungan data pada basis data. Tugas utama skema adalah menjabarkan struktur basis data kepada DBMS. Hasil kompilasi dari pernyataan-pernyataan DDL disimpan dalam berkas-berkas spesial yang disebut katalog sistem.

Pernyataan SQL seperti CREATE TABLE dan DROP TABLE merupakan contoh perintah yang termasuk dalam kategori DDL.

B. Bahasa Manipulasi Data (DML)

DML adalah perintah-perintah yang digunakan untuk mengubah, memanipulasi, dan mengambil data pada basis data. Tindakan seperti menghapus, mengubah, dan mengambil data menjadi bagian dari DML. DML pada dasarnya dibagi menjadi dua:

1. DML Prosedural adalah perintah-perintah yang memungkinkan pengguna menentukan data apa saja yang diperlukan dan bagaimana cara mendapatkannya.
2. DML Nonprosedural adalah perintah-perintah yang memungkinkan pengguna menentukan data apa saja yang diperlukan, tetapi tidak perlu menyebutkan cara mendapatkannya.

2.11.1 Produk DBMS

Pasar perangkat lunak DBMS, dalam hal ini adalah RDBMS merupakan pasar yang sangat luas dan besar untuk diperebutkan [1]. Saat ini, beragam produk RDBMS telah hadir untuk beragram ukuran, bentuk pemodelan yang diikuti, komersial atau *free*, *open source* dan sebagainya.

Terdapat beragam RDBMS terpadu untuk aplikasi skala kecil, misalnya Microsoft Access. Demikian juga terdapat sangat banyak RDBMS untuk aplikasi skala besar (*enterprise*), misalnya Oracle, Microsoft SQL Server, Sysbase dan Borland Interbase Server. RDBMS *open source* untuk skala kecil yang telah menerapkan SQL yaitu MySQL. Sedangkan RDBMS *open source* yang telah menerapkan objek relasional yaitu ProgresSQL.

2.11.2 Konsep Merancang Basis Data

2.11.2.1 Definisi

Istilah basis data banyak menimbulkan interpretasi yang berbeda. Anthoni J. Fabbri dan A. Robert Schwab, mendefinisikan basis data sebagai berikut:

> “Basis data adalah sistem berkas terpadu yang dirancang terutama untuk meminimalkan pengulangan data.

Menurut George Tsu-der Chou, basis data dapat didefinisikan sebagai berikut:

> “Basis data sebagai kumpulan informasi bermanfaat yang diorganisasikan ke dalam tatacara yang khusus”.[3]

Basis data dimaksudkan untuk mengatasi problem pada sistem yang memakai pendekatan berbasis berkas.

Sistem basis data adalah suatu sistem menyusun dan mengelola *record-record* menggunakan komputer untuk menyimpan atau merekam serta memelihara data operasional lengkap sebuah organisasi/perusahaan sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakai untuk proses mengambil keputusan.

Untuk mengelola basis data diperlukan perangkat lunak yang disebut DBMS. DBMS adalah perangkat lunak sistem yang memungkinkan para pemakai membuat, memelihara, mengontrol, dan mengakses basis data dengan cara yang praktis dan efisien.

2 .11.2.2 Bahasa Basis Data

Semua DBMS memiliki dua macam bahasa yang digunakan untuk mengelola dan mengorganisasikan data, yaitu:

1. Bahasa Definisi Data (DDL/*Data Definition Language*)
2. Bahasa Manipulasi Data (DML/*Data Manipulation Language)*

Selain kedua bahasa di atas, pada kebanyakan DBMS sekarang juga terdapat bahasa yang tergolong DCL (*Data Control Language*) yang berkaitan dengan pengaturan sekuritas basis data.

2.11.2.3 Pengendalian Data

Fungsi pengendalian data mempunyai dua tujuan dsar yaitu:

1 untuk menjaga dan menjamin keamanan aset perusahan yang sedang dijalankan, termasuk data

2. untuk menjamin bahwa data yang diperoleh akurat dan lengkap serta diproses dengan benar. Berbagai teknik dan prosedur dapat dipakai untuk menyelenggarakan pengendalian dan keamanan yang memadai.

A. Bahasa Definisi Data (DDL)

DDL adalah perintah-perintah yang biasa digunakan oleh administrator basis data (DBA) untuk mendefinisikan skema ke DBMS. Skema adalah deskripsi lengkap tentang struktur medan, rekaman, dan hubungan data pada basis data. Tugas utama skema adalah menjabarkan struktur basis data kepada DBMS. Hasil kompilasi dari pernyataan-pernyataan DDL disimpan dalam berkas-berkas spesial yang disebut katalog sistem.

Pernyataan SQL seperti CREATE TABLE dan DROP TABLE merupakan contoh perintah yang termasuk dalam kategori DDL.

B. Bahasa Manipulasi Data (DML)

DML adalah perintah-perintah yang digunakan untuk mengubah, memanipulasi, dan mengambil data pada basis data. Tindakan seperti menghapus, mengubah, dan mengambil data menjadi bagian dari DML. DML pada dasarnya dibagi menjadi dua:

1. DML Prosedural adalah perintah-perintah yang memungkinkan pengguna menentukan data apa saja yang diperlukan dan bagaimana cara mendapatkannya.
2. DML Nonprosedural adalah perintah-perintah yang memungkinkan pengguna menentukan data apa saja yang diperlukan, tetapi tidak perlu menyebutkan cara mendapatkannya.

2.11.2.4 Tahap Perancangan Basis Data

Perancangan basis data merupakan langkah untuk menentukan basis data yang diharapkan dapat mewakili seluruh kebutuhan pengguna. Perancangan basis data terdiri atas perancangan basis data secara konseptual, perancangan basis data secara logis, dan perancangan basis data secara fisis.

Beberapa komponen yanng terdapat pada perancangan basis data secara konseptual antara lain:

1. Entitas

   Entitas adalah sesuatu atau objek pada dunia nyata yang dapat dibedakan satu terhadap yang lain, dan bermanfaat bagi aplikasi yang sedang kita kembangkan.

2. Atribut

   Atribut adalah komponen untuk mendeskripsikan entitas yang digambarkan dengan bentuk elips

2 . Hubungan

   Hubungan adalah asosiasi atau kaitan antara dua entitas. Hubungan dapat berupa:

a. 1-1, yakni satu data pada suatu tabel berpasangan dengan hanya satu data tabel lain.
b. 1-N, yakni satu data pada suatu tabel berpasangan dengan banyak data pada tabel lain.
c. N-M, yakni banyak data pada suatu tabel berpasangan dengan banyak data pada tabel lain.

3 . Kekangan

Kekangan digunakan untuk melindungi integritas data (misalnya, melindungi kesalahan sewaktu pengisian data).

4 . Domain

Domain adalah himpunan yang berlaku bagi suatu atribut. Kekangan domain mendefinisikan nama, tipe, format, panjang, dan nilai masing-masing *item* data.

5 . Integritas Referensial

Integritas referensial adalah aturan-aturan yang mengatur hubungan antara kunci primer dengan kunci tamu milik tabel-tabel yang berbeda dalam suatu basis data relasional untuk menjaga konsistensi data.

6 . Relasi

Relasi adalah hubungsn dari beberapa entitas yang digambarkan dengan jajaran genjang relasi antara tabel dapat dikategorikan menjadi 3 macam.

7. Pengolahan

Pengolahan adalah suatu dapat mempunyai suatu bagian pengolahan yang merubah masukan menjadi keluaran

8. Integritas Referensial

   Integritas referensial adalah aturan-aturan yang mengatur hubungan antara kunci primer dengan kunci tamu milik tabel-tabel yang berbeda dalam suatu basis data relasional untuk menjaga konsistensi data.

9. Pemodelan Analisis

   Model analisis, yang sebenarnya merupakan serangkaian model, merupakan representasi teknis yang pertama dari sistem. Tetapi saat ini ada dua yang mendominasi landasan pemodelan analisis. Yang pertama, analisis terstruktur, adalah metode pemodelan klasik, dan analisis berorientasi objek.

Analisis terstruktur adalah aktivitas pembangunan model. Dengan menggunakan notasi yang sesuai dengan prinsip analisis operasional dapat menciptakan model yang menggambarkan muatan dan aliran informasi (data kontrol), membagi sistem secara fungsional dan secara *behavioral*, dan menggambarkan esensi dari apa yang harus dibangun.

Entity-relationship Diagram adalah notasi yang digunakan untuk melakukan aktivitas pemodelan data. Atribut dari masing-masing objek data yang ditulis pada ERD dapat digambarkan dengan menggunakan deskripsi objek data. Sedangakan *data flow diagram* (DFD) memberikan informasi tambahan yang digunakan selama analisis domain informasi dan berfungsi sebagai dasar bagi pemodelan fungsi.

2.11.2.5 Entity Relationship Diagram (ERD)

ERD hanya berfokus pada data, dengan menunjukkan “jaringan data” yang ada untuk suatu sistem yang diberikan. ERD sangat berguna bagi aplikasi di mana data dan hubungan yang mengatur data sangatlah kompleks. ERD pada mulanya diusulkan oleh Peter Chen untuk desain sistem *database* relasional dan telah dikembangkan oleh yang lainnya. Serangkaian komponen utama diindentifikasikan untuk ERD: objek data, atribut, hubungan, dan berbagai tipe indikator. Tujuan utama dari ERD adalah untuk mewakili objek data dan hubungan mereka.

2.11.2.6 Data Flow Diagram (DFD)

Data Flow Diagram (DFD –DAD/Diagram Alir Data) memperlihatkan hubungan fungsional dari nilai yang dihitung oleh sistem, termasuk nilai masukan, nilai keluaran, serta tempat penyimpanan internal. DAD adalah gambaran grafis yang memperlihatkan aliran data dari sumbernya dalam objek kemudian melewati proses yang mentransformasinya ke tujuan yang lain, yang ada pada objek lain. DAD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangan lingkungan fisik dimana data tersebut mengalir. DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur (*structured analysis and design*). DFD merupakan alat yang cukup populer sekarang ini, karena dapat menggambarkan arus data di dalam sistem dengan terstruktur jelas.

Beberapa simbol yang digunakan dalam *Data Flow Diagram* (DFD) antara lain:

1. *External Entity* (kesatuan luar) atau *boundary* (batas sistem)

   Setiap sistem pasti mempunyai batas sistem (*boundary*) yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima *input* dan menghasilkan *output* kepada lingkungan luarnya. Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan *input* atau menerima *output* dari sistem.

2. *Data Flow* (arus data)

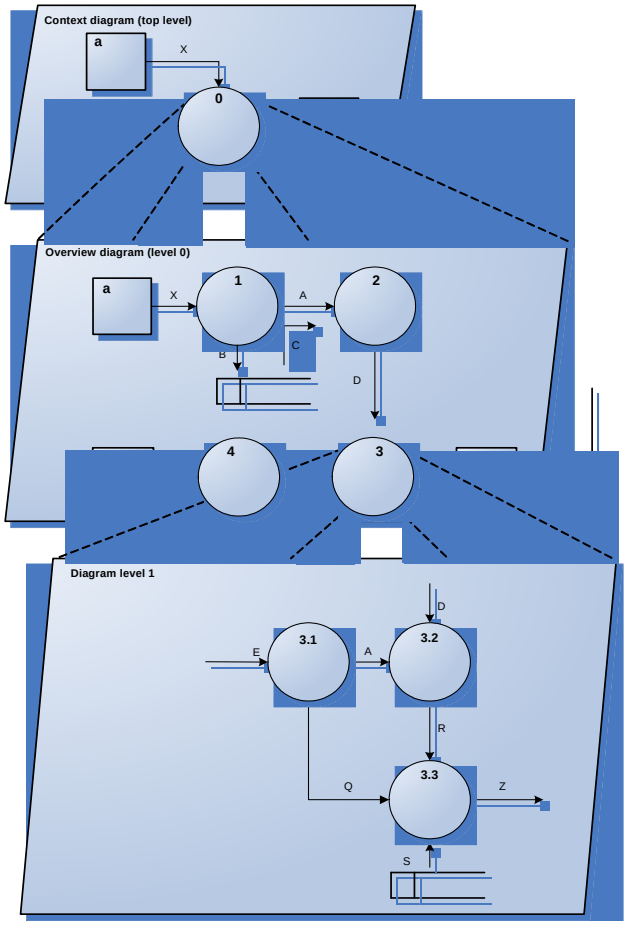
   Arus data (*data flow*) di DFD diberi simbol suatu panah. Arus data ini mengalir diantara proses (*process*), simpanan data (*data strore*) dan kesatuan luar (*external entity*). Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

3. *Process* (proses)

   Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses. Setiap proses harus diberi penjelasan yang lengkap meliputi identifikasi proses, nama proses dan pemroses.

4. *Data Store* (simpanan luar)

   Simpanan data (*data store*) merupakan simpanan dari data yang dapat berupa, yaitu suatu file atau *database* di sistem komputer, suatu arsip atau catatan manual, suatu kotak tempat data di meja seseorang, suatu tabel acuan manual, dan suatu agenda atau buku



Gambar 2.3 Bentuk *Data Flow Diagram* (DFD)

2.12 Kamus Data

Kamus data (KD) atau *data dictionary* (DD) atau disebut juga dengan istilah *system data dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data, analisis sistem dapat mendefinisikan data yang mengalir di sistem dengan lengkap. Kamus data dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem.

Pada tahap analisis, kamus data dapat digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir di

sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.

Pada tahap perancangan, kamus data digunakan untuk merancang *input*, merancang laporan-laporan dan *database*. Kamus data dibuat berdasarkan arus data yang ada di DFD. Arus data di DFD sifatnya global, hanya ditunjukkan nama arus datanya saja. Keterangan lebih lanjut tentang struktur data dari arus data di DFD secara lebih rinci dapat dilihat di kamus data. Kamus data harus dapat mencerminkan keterangan yang jelas tentang data yang dicatatnya, maka kamus data harus memuat nama arus data, alias, bentuk data, arus data, penjelasan, periode, volume, dan struktur data.[1]

Tabel 2.1 Notasi Kamus Data

Notasi	Arti
=	Terbentuk dari (is composed) atau terdiri dari (consist of) atau sama dengan (is equivalent of)
+	AND
[]	Salah satu dari (memilih dari salah satu dari elemen-elemen data di dalam kurung bracket ini)
|	Sama dengan simbol []
M{ }M	Iterasi (elemen data di dalam kurung brace beriterasi mulai minimum N kali dan maksimum M kali
( )	Optional (elemen data di dalam kurung parenthesis sifatnya optional, dapat ada dan dapat tidak ada)
*	Keterangan setelah tanda ini adalah komentar

2.12.1 Intranet

2.12.1.1 Definisi

Intranet (*Internal Network*) mulai terdengar pada pertengahan tahun 1995 oleh beberapa penjual produk jarinagn yang mengacu pada kebutuhan informasi dalam bentuk *web* dalam perusahaan. Intranet adalah jaringan komputer dalam perusahaan yang menggunakan komunikasi data standar seperti dalam internet. Artinya, kita dapat menggunakan semua fasilitas internet untuk kebutuhan dalam perusahaan. Dengan kata lain intranet dapat dikatakan berinternet dalam lingkungan perusahaan. Standar internet yang digunakan adalah standar protokol TCP/IP (*Transport Control Protocol*, ada yang menyebutnya *Transmission Control Protocol/Internetworking Protocol*), TCP/IP memungkinkan protokol jaringan untuk berkomunikasi, menerima dan mengirimkan data ke terminal yang lain. Standar yang lain adalah FTP (*File Transfer Protocol*) yang merupakan pelayanan *resource sharing*, sebuah fasilitas untuk dapat mengambil file yang ada di internet. SMTP (*Simple Mail Transfer Protocol*) yang merupakan dasar dari e-mail untuk berkomunikasi serta MIME (*Multipurpose internet Mail Extensions*) yang merupakan standar untuk mendefinisikan format biner, grafik dan suara agar dapat ditransmisikan dengan e-mail. Protokol lainnya adalah NNTP (*Network News Transfer Protocol*) dan POP (*Post Office Protocol*).

Web server adalah jaringan *client server* interaktif yang menggunakan teknologi *World Wide Web* (penggunaan *hypertext* dan *graphics* secara bersamaan untuk menampilkan informasi). Sedangkan *Extended Intranet* adalah intranet yang

mewakili kesempatan perusahaan untuk dapat berkomunikasi dan bertransaksi dengan pelanggan dan *supplier* dan proses bisnis perusahaannya.[6]

2.12.2 Aplikasi Web

Pada awalnya aplikasi *web* dibangun hanya dengan menggunakan bahasa yang disebut HTML (*HyperText Markup Language)*. Pada perkembangan berikutnya, sejumlah skrip dan objek dikembangkan untuk memperluas kemampuan HTML. Pada saat ini, banyak skrip seperti itu, antara lain yaitu PHP dan ASP, sedangkan contoh yang berupa objek adalah *applet*. Aplikasi *web* itu dapat dibagi menjadi *web* statis dan *web* dinamis.

Web statis dibentuk dengan menggunakan HTML saja. Kekurangan aplikasi seperti ini terletak pada keharusan untuk memelihara program secara terus-menerus untuk mengikuti setiap perubahan yang terjadi. Kelemahan ini diatasi dengan model *web* dinamis.

Dengan menggunakan pendekatan *web* dinamis, dimungkinkan untuk membentuk sistem informasi berbasis *web*. Dari sisi teknologi yang digunakan untuk membentuk *web* dinamis terdapat dua pengelompokan, yaitu teknologi pada sisi klien dan teknologi pada sisi server.

Teknologi *web* pada sisi klien diimplementasikan dengan mengirimkan kode perluasan HTML atau program tersendiri dan HTML ke klien. Klienlah yang bertanggung jawab dalam melakukan proses terhadap seluruh kode yang diterima. Kelemahan pendekatan seperti ini adalah terdapat kemungkinan bahwa *browser* pada klien tidak mendukung fitur kode perluasan HTML. Kelebihan teknologi

pada sisi klien, yaitu memungkinkan penampilan yang bersifat dinamis. Contoh teknologi pada sisi klien, yaitu Kontrol ActiveX, Java Applet, dan Skrip sisi-klien

Teknologi *web* pada sisi server memungkinkan pemrosesan kode di dalam server sehingga kode yang sampai pada pemakai berbeda dengan kode asli pada server. Contoh teknologi yang berjalan di server, yaitu CGI, ASP, JSP, PHP dan lain sebagainya. Keuntungan penggunaan teknologi pada sisi server adalah sebagai berikut:

1. Mengurangi lalu lintas jaringan dengan cara menghindari percakapan bolak-balik antara klien dan server.
2. Mengurangi waktu pembuatan kode, mengingat klien hanya mengambil kode HTML saja.
3. Mencegah masalah ketidakkompatibelan *browser*.
4. Klien dapat berinteraksi dengan data yang ada pada server.
5. Mencegah klien mengetahui rahasia kode (mengingat kode yang diberikan ke klien berbeda dengan kode asli pada server).