BAB II

TINJAUAN PUSTAKA

2.1 Profil PJTV

PJTV (PadjadjaranTelevisi) berdiri pada tahun 2005 di Bandung dengan kekuatan pemancar 3Kw. Didukung dengan tenaga SDM yang muda dan kreatif serta mempunyai visi yang sama sampai saat ini PJTV menjadi stasiun televisi yang eksis di kota Bandung dan sekitarnya.

Dengan semangat untuk mencapai Visinya, selain telah banyaknya program program baru yang menarik, menghibur serta mendidik, PJTV adalah stasiun TV lokal pertama yang di bandung yang sudah mengudara kurang lebih 18 jam per hari dan saat ini telah menambah daya pancarnya menjadi 10kw yang tentunya akan menjangkau sebagian besar wilayah Jawa Barat.

2.1.1 Sejarah PJTV

PJTV lahir pada bulan Juni 2005 dengan nama Padjadjaran Televisi, dengan konsep Televisi Lokal yang mengutamakan kontent lokal dan budaya masyarakat. Pada bulan Maret 2006, Padjadjaran Televisi bergabung di bawah naungan PT. JAWA POS GROUP (kelompok media terbesar di Indonesia yang telah menjadi market leader untuk Koran dan televisi lokal di Jawa Timur dan beberapa kota besar lainnya) membuat Padjadjaran Televisi semakin yakin untuk mengukuhkan dirinya dan dapat bersaing dengan televisi lokal yang telah ada Bandung.

Terletak di kawasan Asri Nuasa Mas Estate B-12 Riung Bandung, Padjadjaran Televisi awalnya berlogo PTV, lalu diubah menjadi PJTV (Padjadjaran Televisi) dan merubah *Style* siaran dari format "beriman dan bertaqwa" menjadi "siaran keluarga", yang dapat dinikmati oleh segala usia, dan mejadi televisi milik masyarakat Bandung.

Tulisan PJTV yang diawali dengan huruf P adalah singkatan dari Padjadjaran dan tulisan JTV yang digabungkan merupakan stasiun televisi jawa TV sebagai pemilik dari PJTV.Warna Silver adalah bahwa PJTV sebagai stasiun televisi yang cemerlang. Tulisan BANDUNG EUY merupakan *tag line* sebagai ciri khas orang Bandung.PJTV saat ini mengusung sebuah konsep yang sangat berbeda dengan televisi lokal yang telah lebih dulu ada di Bandung, yaitu dengan mengutamakan konsep 3L (*Lokal, Leisure, Lifestyle*).

Lokal : Karena memang acaranya mengusung tema-tema lokal.

Leisure : Karena konsep acara yang akan menyajikan program-program untuk mereka yang ingin bersantai di rumah, dengan menjauhkan program-program TV dari unsur kekerasan dan penyiksaan.

Lifestyle : Bandung adalah kota yang sangat khas dengan lifestyle. Bandung menjadi panutan kota-kota lain di Indonesia. Dengan tim kreatif kru yang telah berpengalaman untuk format televise local, PJTV yakin dalam waktu yang tidak terlalu lama akan dapat tampil menjadi televisi kebanggaan masyarakat Bandung dan sekitarnya.

2.1.2 Visi Dan Misi

2.1.2.1 Visi

1. Menjadi model televisi lokal yang dapat mengerti keinginan masyarakat Bandung yang sangat bangga akan kotannya dan kemudian juga bangga memiliki stasiun televisi yang benar-benar milik Kota Bandung.

2. Membuat PJTV sebagai stasiun televisi yang dapat menghibur masyarakat Bandung sekaligus memberi kesempatan kepada masyarakat untuk juga terlibat dalam program-program PJTV sehingga mereka dapat menjadikan PJTV sebagai contoh televisi yang selalu menjadi sahabat dan memberikan kesempatan luas bagi masyarakat.

2.1.2.2 Misi

1. Membangun dan mengembangkan PJTV, sebagai sahabat dan tempat masyarakat Bandung mencari hiburan yang mendidik disebut stasiun televisi.

2. Membentuk masyarakat yang edukatif dan selektif dalam hal selera tontonan sehingga mengerti bahwa konsep sebuah televisi lokal adalah mengangkat tema kehidupan dan ritme budaya sehari-hari masyarakat Bandung dan mereka bangga menjadi masyarakat selektif di Bandung.

Misi Program PJTV

1. Program yang dapat memberikan hiburan-hiburan cerdas untuk masyarakat yang telah penat menjalani keseharian kehidupannya.
2. Menjadi sarana informasi yang terpercaya.
3. Dapat melahirkan bintang-bintang baru dari Bandung yang dapat menjadi kebanggan Kota Bandung.
4. Menyuguhkan tayangan yang kreatif dan inovatif.

2.1.3 Format PJTV

1. Target Segmentasi PJTV

a. *All Segment*

b. Keluarga (*Primary Target : 10 – 55 years old*)

c. Remaja (*Secondary Target : 15 – 25 years old*)

d. *SES LEVEL* : A1, A2, B1, C1, C2.

2. Format Tayangan

1. *Live*

2. *Live Interaktif*

3. *Taping*

2.1.4 Konsep Dan Format PJTV

2.1.4.1 Konsep Program

1. Selalu mengedepankan konsep Lokal, *leisure* dan *life style*.
2. Sebanyak mungkin melibatkan masyarakat.
3. Memberikan kesungguhan yang berbeda untuk kepausan para pemasang iklan dengan memberikan masukan innovatif dari *team* kreatif sehingga pemasang iklan dapat mencapai tujuan dalam beriklan.

2.1.4.2 Format Program

1. *Entertainment Program*
2. *Edutaimnet*
3. *Variety Show*
4. *Soft News*
5. *Talent Searching*

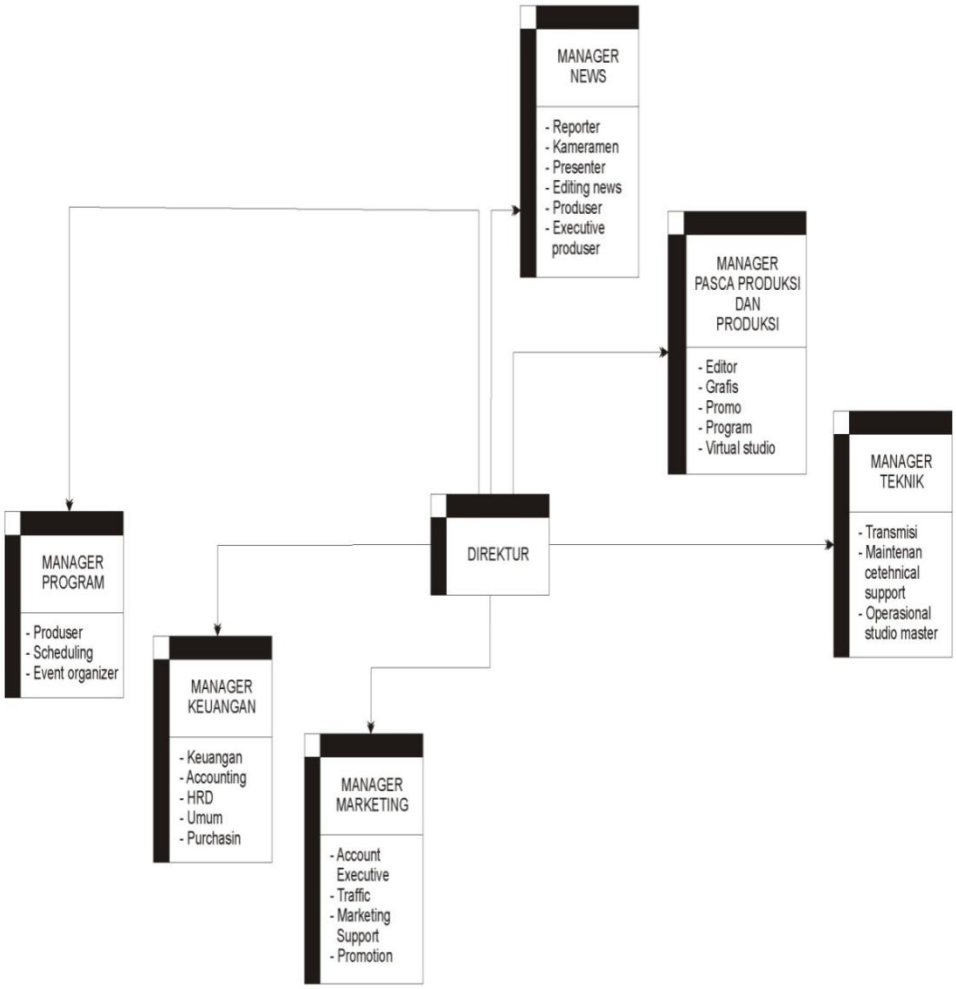
2.1.4.3 Klasifikasi Program

1. *Soft News*
2. *Talkshow*
3. *Otomotif*
4. *Life Style*
5. *Education Entertaiment*
6. *Quis Unteraktif*
7. *Tantangan*
8. *Break The Record*
9. *Filler*

10. Variety Show

2.1.5 Struktur Organisasi

Menjalankan stasiun televisi, maka diperlukan suatu struktur organisasi agar berjalan dengan baik dan terencana.Untuk menjelaskan struktur organisasi PJTV dan juga divisi yang menjadi sasaran kegiatan penelitian, maka struktur organisasi dari PT ESA VISUAL PADJADJARAN TIVI (PJTV) Bandung di kategorikan menjadi dua struktur yaitu, struktur besar dan struktur divisi.Untuk lebih jelasnya, dapat dilihat pada gambar 2.1.



Gambar 2.1 Sturktur Organisasi Di PJTV

2.1.5.1 Job Descriptions Struktur Besar

Kedudukan-kedudukan atau posisi yang terdapat pada Struktur Besar di dalam PJTV antara lain :

a. *President Director* (Komisaris Utama)

Memilki tanggung jawab atas produktivtas stasiun PJTV secara menyeluruh dalam mengembangkan acara, arah dan strategi bsisni perusahaan yang keseluruhannnya akan dipertanggung jawabkan kepada para pemegang saham.

b. Executive Secretary

Kedudukannya tepat dibawah *President Director* yang memiliki tugas untuk membantu *President Director* dalam menyusun jadwal atau program kerja *President Director* yang sifatnya harian ataupun jangka waktu yang tepat.

c. Coorporet Research & Development

Memilki tanggung jawab dalam menetukan efektifitas dan kualitas sumber daya manusia yang dimilki perusahaan.

d. Management Information System

Bertanggung jawab untuk menyediakan sarana dan prasarana yang berkaitan dengan pengelolaan data – data informasi yang dibutuhkan perusahaan.

e. Director Operation

Mengawasi ketersediaan perangkat operasional serta pemanfaatnya dalam keseharian aktivitas perusahaan berkala.

f. Director Sales, Marketing & Programming

Bertanggung jawab dalam perencanaan, penetapan strategi usaha pemasaran hasil produksi perusahaan dan menetukan target penapatan sesuai dengan anggaran yang tersedia dan pengembangannya.

g. Director Finance

Bertugas mengatur adminitrasi keuangan dan pengelolaan anggaran yang tersedia.

h. General Manager Production

Mengkoordinasi, mengawasi dan mengevaluasi produksi dan penyiaran program tayangan sesuai dengan ketentuan standard *Operation Process* yang ada.

i. General Manager News

Mengkoordinir, mengawasi dan mengevaluasi pelaksanaan produksi, penyiaran dan operasional siaran berita sesuai dengan standard *OperationProcess.*

2.1.5.2 Job Descriptions News Departement

a. *General Manager News* (Pimpinan Redaksi)

Bertanggung jawab terhadap mekanisme dan aktivitas kerja keredaksian sehari-hari, mengkoordinasi, mengawasi dan mengevaluasi pelaksanaan produksi, penyiaran dan operasional siaran berita.

b. Produser *Eksekutif*

Bertugas memikirkan *setting*, dekor, latar belakang atau tampilan suatu program berita yang akan menjadi ciri khas program berita itu, dan juga melakukan pengawasan terhadap kerja reporter dan produser dan memastikan staf redaksi memenuhi *style* yang telah ditetapkan. Ia juga harus memikirkan cara memperbaiki mutu suatu program.

c. Asisten Produser

Bertugas membantu reporter menyiapkan paket berita jika reporter berada dalam waktu yang mendesak atau jika reporter tidak bisa menyelesaikan paket beritanya karena harus berangkat lagi menjalankan tugas.

d. Sekretaris Redaksi

Mengurus administrasi dan realisasi peliputan. Anggaran untuk dalam kota dan luar kota.

e. Produser Program

Bertanggung jawab terhadap proses penciptaan dan pengembangan suatu program sesuai dengan tema yang telah ditentukan dan disepakati oleh produser eksekutif. Serta menyediakan konsep atau ide dan materi program sesuai dengan bidang keahlian dan memutuskan berita apa yang akan disiarkan.

f. Presenter

Bertugas menyampaikan informasi dan pesan yang telah disiapkan dalam bentuk naskah kepada pemirsa.

g. Scripwriter

Bertugas untuk menulis dan menyusun naskah (*scrip*) acara.

h. Koordinator Kamera

Orang yang bertugas mengkoordinir kegiatan kameramen secara keseluruhan saat menyiarkan suatu program acara.

i. Juru Kamera

Di studio juru kamera bertugas menyediakan visual untuk paket acara (*package*) yang lengkap sesuai dengan kaidah visual berita. Sedangkan dilapangan juru kamera bertugas meliput terutama dalam hal pengambilan gambar berita.

j. Tata Cahaya

Bertanggung jawab mengatur penempatan sumber cahaya, mengarahkannya, sehingga memperoleh efek yang diinginkan.

k. Tata Artistik

Bertanggung jawab untuk merencanakan setting dekorasi, mengawasi konstruksi set, dan penataan grafik.

l. Sub Kontrol

Bertanggung jawab pada bagian *audio* dan mengatur *balance* suara dari berbagai sumber juga mengatur penempatan mikrofon.

m. Penyunting Gambar (*editor*)

Menggabungkan gambar-gambar yang diperoleh juru kamera.

n. Penata Rias

Mempersiapkan segala sesuatu keperluan *presenter* yang berhubungan dengan rias dan busana saat pengambilan gambar.

2.1.6 Standard Procedure Operational

2.1.6.1 Reporter

a. Datang tepat waktu, lihat papan tulis dan baca baik-baik tegas yang diberikan.
b. Hubungi juru kamera, dan siapkan peralatan (kamera, kaset) yang dibutuhkan.
c. Berangkat meliput.
d. Ketika liputan, perhatikan alam dan kondisi sekitarnya, jangan Cuma inti berita itu saja, elemen lingkungan dapat memperkaya isi tulisan.
e. Jika sudah selesai langsung pulang ke kantor, bukan ke rumah.
f. Sampai di kantor ketik naskah berita sesuai dengan gambar yang diliput dan berita yang diliput.
g. Jika sudah selesai mengetikan naskah berita, naskah berita langsung diserahkan pada produser.

2.1.6.2 Juru Kamera

a. Untuk juru kamera yang menggunakan peralatan logistik *news* dimohon mengisi formulir pengambilan alat.
b. Untuk pengambilan peralatan logistik *news* hanya juru kamera, selain juru kamera (*reporter, driver*, dll) tidak diperbolehkan.
c. Juru kamera diwajibkan untuk bertanggung jawab atas peralatan logistik *news* yang digunakan saat liputan.
d. Juru kamera saat liputan diwajibkan untuk menjaga peralatan.
e. Juru kamera harus jujur dan bertanggung jawab apabila terjadi kerusakan peralatan.
f. Bila terjadi kerusakan pada alat yang habis digunakan, juru kamera harus melaporkan kepada logistik *news* atau koordinator kamera secepatnya.
g. Kamera yang disediakan koordinator news hanya bisa digunakan untuk meliput, tidak boleh digunakan untuk yang lainnya.

2.1.6.3 Persiapan Reporter dan Juru Kamera Sebelum Meliput

a. Mencari informasi dari berbagai sumber tentang kasus atau berita yang akan diliput.
b. Disaat yang sama juru kamera juga harus mempersiapkan peralatan untuk meliput (kamera *full batre*, kaset beserta cadangannya satu, *microfon*, dll).
c. Bersiap untuk berangkat ketempat tujuan yang akan diliput, dan tidak lupa membawa ongkos bensin yang sudah disediakan oleh pihak kantor.

2.1.6.4 Reporter Saat Dilapangan

a. Sampai ditempat liputan, seorang reporter dan juru kamera harus *stand by* atau bersiap saat berlebih disaat *hunting*.
b. Saat liputan seorang reporter sudah harus siap dengan pertanyaan-pertanyaan yang akan ditanyakan atau mewawancarai narasumber.
c. Sebelum melakukan wawancara seorang reporter harus memperkenalkan diri dahulu, sebelum melakukan tugas.
d. Selain mendengarkan narasumber, reporter juga harus siap dengan alat tulis untuk menulis informasi yang didapat dari narasumber.
e. Mulailah dengan pertanyaan yang mudah terlebih dahulu.

2.2 Landasan Teori

Konsep dasar sistem ini meliputi pengertian dasar dari sistem, bentuk umum sistem, karakteristik sistem dan analisis sistem.

2.2.1 Pengertian Sistem

Pengertian sistem adalah kumpulan dari unsur atau elemen-elemen yang saling berkaitan atau berinteraksi dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan tertentu.

Didalam mendefinisikan sistem, terdapat dua pendekatan yang menekankan pada prosedurnya dan yang menekankan pada komponen atau elemennya. Pendekatan sistem yang lebih menekankan pada prosedur mendefinisikan sistem sebagai berikut :

"Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersam-sama unutk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu".

Sedangkan pendekatan sistem yang lebih menekankan pada elemen atau Komponennya mendefinisikan sistem sebagai berikut :

"Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai tujuan tertentu ".

2.2.2 Bentuk Umum Sistem

Bentuk umum dari suatu sistem terdiri atas masukan (*input*), proses, dan keluaran (*output*). Dalam bentuk umum sistem ini biasa melakukan satu atau lebih masukan yang akan diproses dan menghasilkan keluaran sesuai dengan rencana yang telah ditentukan sebelumnya.



Gambar 2.2 Bentuk Umum Sistem

2.2.3 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, antara lainakan dijelaskan dibawah ini.

2.2.4 Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang berikut:

1. Sistem dapat diklasifikasikan sebagai sistem abstrak *(abstrac system)* dan sistem phisik *(physical system)*. Sistem abstrak adalah sistem yang berupa

pemikiran atau ide-ide yang tidak tampak secara fisik. Sistem phisik merupakan sistem yang ada secara fisik.

2. Sistem dapat diklasifikasikan sebagai sistem alamiah *(natural system)* dan sitem buatan manusia *(human made system)*. Sistem alami adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia. Sistem buatan manusia adalah sistem yang dirancang oleh manusia. Sistem buatan manusia yang melibatkan interaksi antara manusia dengan mesin disebut *human-machine system* atau ada yang menyebutnya dengan *man machine system.*
3. Sistem dapat diklasifikasikan sebagai sistem tertentu *(deterministic system)* dan system tidak tentu *(probabilistic system).* Sistem tertentu beroprasi dengan tingkah laku yang sudah diprediksi. Interaksi antara bagian dengan tingkah laku yang sudah dapat diprediksi dengan pasti sehingga keluaran sistem dapat diramalkan. Sistem tidak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsure probabilistik. Sistem diklasifikasikan sebagai sistem penutup *(closed system)* dan sistem terbuka *(open system)*. Sistem tertutup merupakan sistem yang tidak berhubungan dengan lingkungan luarnya. Secara teoritis sistem tertutup ini ada, tetapi kenyataannya tidak ada sistem yang benar-benar tetutup, yang ada hanyalah *relativelly closed system* (secara relatif tertutup, tidak benar-benar tertutup). Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya.

Sistem ini menerima dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lainnya.

2.2.5 Analisis Sistem

Analisis sistem *(sistem analis)* merupakan tahapan yang sangat penting dalam membangun sebuah aplikasi, karena kesalahan dalam tahap ini dapat mempengaruhi tahapan selanjutnya. Analisa sistem didefinisikan sebagai berikut :

"Penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya".

Dalam tahap analisis terdapat langkah-langkah dalam pengerjaannya, sebagai berikut:

a. *Identify* yaitu mengidentifikasi masalah.
b. *Understand* yaitu memahami kerja dari sistem yang ada.
c. *Analyze* yaitu menganalisis sistem.
d. *Report* yaitu membuat laporan hasil analisis.

2.2.6 Desain Sistem dan Tujuannya

Desain sistem dapat didefinisikan sebagai berikut*:*

"Penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi".

Selain itu Desain sistem memunyai maksud dan tujuan sebagai berikut:

1. Untuk memenuhi kebutuhan pengguna sistem.
2. Memberikan gambaran dalam perancangan sistem.

Sedangkan sasaran-sasaran yang harus dicapai agar desain sistem dapat mencapai tujuan:

1. Desain sistem harus berguna, mudah dipahami dan nantinya mudah digunakan.
2. Desain sistem harus dapat mendukung tujuan utama perusahaan atau instansi.
3. Desain harus efektif dan efisien serta mendukung keputusan yang akan dilakukan oleh manajemen.

2.2.7 Definisi Sistem Informasi

Informasi merupakan hal yang sangat penting bagi manajemen dalam mengambil suatu keputusan.Suatu informasi dapat diperoleh dari sistem informasi atau juga disebut dengan *processing system* atau *information processing system* atau *information-generating system*.Sistem informasi didefinisikan oleh Robert A. Leitchdan K. Roscoe Davis adalah *"suatu system didalam sebuah organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung orperasi, bersifat menejerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan".*

2.2.8 Konsep dan Dasar Sistem Informasi

Sebuah sistem informasi merupakan kumpulan dari perangkat keras dan perangkat lunak komputer serta perangkat manusia yang akan mengolah data menggunakan perangkat keras dan perangkat lunak tersebut. Informasi merupakan hal yang sangat penting dalam pengambilan keputusan, permasalahannya adalah dimana informasi tersebut didapat.Informasi dapat diperoleh dari sistem informasi. Robert A Leitch dan K. Roscoe Davis mendefinisikan sistem informasi sebagai berikut:

"Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan."

2.2.9 Definisi Basis Data

Basis Data adalah kumpulan data yang saling berhubungan yang disimpan secara bersama dengan tanpa adanya pengulangan (redundansi) yang tidak perlu untuk memenuhi berbagai kebutuhan. Terdapat konsep dasar pada basis data, konsep dasar tersebut yaiut :

a. Field, merupakan unit terkecil dari data yang berarti, yang disimpan dalam suatu file atau basis data.
b. Record, merupakan kumpulan dari field-field yang disusun dalam format yang telah ditentukan.
c. File dan Tabel, merupakan ekivalen basis data relasional dari sebuah file.

Selain itu terdapat operasi dasar yang dimiliki oleh basis data, yaitu :

a. Menambah data.
b. Membaca data.
c. Mengubah data.
d. Menghapus data.

2.2.10 Tujuan Basis Data

Basis data dimaksudkan untuk mengatasi problem pada sistem yang memakai pendekatan berbasis berkas. Basis Data pada prinsipnya ditujukan untuk pengaturan data agar terdapat kemudahan dalam pengambilan kembali data tersebut. Berikut ini adalah tujuan dari basis data, yaitu :

1. Lebih cepat dan mudah dalam melakukan penyimpanan, perubahan dan manipulasi.
2. Efisiensi ruang penyimpanan.
3. Memungkinkan untuk menjaga keakuratan data dalam basis data.
4. Memungkinkan integrasi semua basis data yang ada.
5. Memungkinkan penambahan jenis data baru dalam basis data yang telah ada.
6. Memiliki sistem keamanan.
7. Memungkinkan pemakaian secara bersama dalam satu waktu.

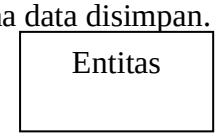
2.2.11 Tahap Perancangan Basis Data

Perancangan basis data merupakan langkah untuk menentukan basis data yang diharapkan dapat mewakili seluruh kebutuhan pengguna. Berikut tahapan-tahap yang biasa digunakan dalam perancangan basis data :

a. Entity Relationalship Diagram (ERD)

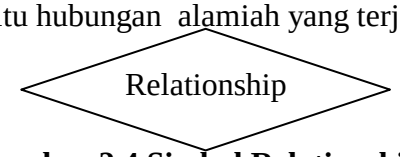
Entity Relationalship Diagram adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak.Model data E-R didasarkan pada persepsi terhadap dunia nyata yang tersusun atas kumpulan objek-objek dasar yang disebut entitas dan relasi. Berikut adalah elemen-elemen yang terdapat pada ERD, yaitu :

1. Entity (Entitas), yaitu sesuatu apa saja yang ada didalam sistem, baik nyata maupun abstrak dimana data disimpan.



Gambar 2.3 Simbol Entitas

2. Relationship, yaitu hubungan alamiah yang terjadi antara entitas.



Gambar 2.4 Simbol Relationship

3. Atribut, yaitu deskripsi kelompok data yang mempunyai karakteristik yang sama, merupakan field yang akan disimpan.



Gambar 2.5 Simbol Atribut

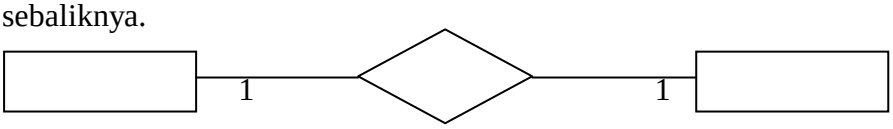
4. **Garis**, sebagai penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.

Gambar 2.6 Simbol Alur Data

5. Kardinalitas, yaitu menyatakan jumlah anggota entitas yang terlibat didalam relasi yang terjadi. Terdapat tiga macam kardinalitas relasi, yaitu :

   a. Satu Ke Satu (one to one / 1-1)

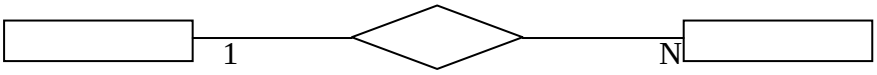
      Setiap entitas pada himpunan entitas A dapat berelasi dengan paling banyak satu entitas pada himpunan entitas B, demikian juga sebaliknya.



Gambar 2.7 ERD Entitas Satu Ke Satu

   b. Satu Ke Banyak (one to many / 1-N)

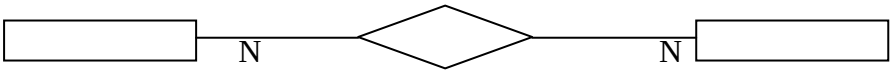
      Setiap entitas pada himpunan entitas A dapat berelasi dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya.



Gambar 2.8 ERD Entitas Satu Ke Banyak

   c. Banyak Ke Banyak (many to many / N-N)

      Setiap entitas pada himpunan entitas A dapat berelasi dengan banyak entitas pada himpunan entitas B, demikian juga sebaliknya.



Gambar 2.9 ERD Banyak Ke Banyak

b. Normalisasi

Normalisasi adalah suatu proses untuk mengubah suatu tabel yang memiliki masalah tertentu ke dalam dua buah tabel atau lebih, yang tidak lagi memiliki masalah tersebut. Masalah tersebut biasanya merupakan suatu ketidak

konsistenan (tidak normal) apabila dilakukan penghapusan *(delete)*, pengubahan *(update)* dan pembacaan *(retrieve)* pada suatu basis data.

Bentuk normalisasi adalah suatu aturan yang dikenakan pada tabel-tabel dalam basis data dan harus dipenuhi oleh tabel-tabel tersebut pada level-level normalisasi. Ada macam-macam bentuk normalisasi, diantaranya adalah bentuk tidak normal, bentuk normal pertama, bentuk normal kedua dan bentuk normal ketiga. Aturan-aturan dalam masing-masing bentuk normalisasi tersebut adalah sebagai berikut :

a. Bentuk tidak normal

Bentuk ini merupakan kumpulan data yang akan disimpan, tidak ada keharusan mengikuti suatu format tertentu, dapat saja data tidak lengkap atau terduplikasi dan data dikumpulkan apa adanya.

b. Bentuk normal pertama (1NF)

Suatu tabel dikatakan dalam bentuk normal pertama (1NF) bila setiap kolom bernilai tunggal untuk setiap baris. Ini berarti bahwa nama kolom yang berulang cukup diwakili oleh sebuah nama kolom (tidak perlu ada indeks dalam memberi nama kolom).

c. Bentuk normal kedua (2NF)

Suatu tabel berada dalam bentuk normal kedua (2NF) jika tabel berada dalam bentuk normal pertama, semua kolom bukan kunci primer tergantung sepenuhnya terhadap kunci primer. Suatu kolom disebut tergantung sepenuhnya terhadap kunci primer jika nilai pada suatu kolom selalu bernilai sama untuk suatu nilai kunci primer yang sama.

d. Bentuk normal ketiga (3NF)

Suatu tabel berada dalam bentuk normal ketiga (3NF) jika tabel berada dalam bentuk normal kedua, setiap kolom bukan kunci primer tidak memiliki ketergantungan secara transitif terhadap kunci primer.

2.2.12 Database Management System (DBMS)

Database Management System (DBMS) diartikan sebagai suatu programkomputer yang digunakan untuk memasukan, mengubah, menghapus,memanipulasi dan memperoleh data atau informasi dengan praktis dan efisien. Dibandingkan dengan sistem yang berbasis kertas, DBMS memiliki empatkeunggulan :

1. Kepraktisan. Sistem yang berbasis kertas akan menggunakan kertas yang sangat banyak untuk menyimpan informasi, sedangkan DBMS menggunakan media penyimpan sekunder yang berukuran kecil tetapi padat informasi.
2. Kecepatan. Mesin dapat mengambil atau mengubah data jauh lebih cepat dari pada manusia.
3. Mengurangi kejemuan. Orang cenderung menjadi bosan kalau melakukan tindakan-tindakan berulang yang menggunakan tangan.
4. Kekinian. Informasi yang tersedia pada DBMS akan bersifat mutakhir dan akurat setiap saat.

Di dalam lingkungan basis data, data lebih mudah digunakan pada beberapa DBMS tersedia fasilitas *Query* yang memudahkan pengguna untuk

memperoleh informasi. Pengguna yang tidak memiliki kemampuan pemrograman pun dapat dengan mudah menggunakan fasilitas *query* tersebut.

2.2.13 Pengolahan Data

Pengolahan data adalah suatu proses untuk menghasilkan bentuk yang lebih berguna berarti dari suatu kejadian berupa informasi yang diperlukan.Sedangkan definisi dari Sistem Pengolahan adalah suatu kesatuan yang saling berhubungan yang terdiri dari peralatan, tenaga pelaksana dan prosedur prosedur yang akan menghasilkan suatu informasi.

2.2.13.1 Flowmap

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

2.2.13.2 Basis Data

Basis data adalah kumpulan dari data yang saling berhubungan antara satuvtabel dengan tabel lainnya yang tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Selain itu Basis data (*database*) juga merupakan kumpulan data yang disusun atau diatur sedemikian rupa sehingga kita dapat memperoleh informasi yang tepat dengan secepat mungkin. Tujuan dari perancangan basis data adalah menentukan data-data yang dibutuhkan dalam sistem sehingga kebutuhan informasi dapat terpenuhi dan dapat diperoleh dengan tepat dan secepat mungkin.

2.2.14 Diagram Konteks (Contexts Diagram)

Diagram konteks menggambarkan hubungan antara sistem dengan entitas luarnya. Diagram konteks berfungsi sebagai transformasi dari satu proses yang melakukan transformasi data input menjadi data output. Entitas yang dimaksud adalah entitas yang mempunyai hubungan langsung dengan sistem. Suatu diagram konteks selalu mengandung satu dan hanya satu proses saja. Diagram konteks ini menggambarkan hubungan input atau output antara sistem dengan dunia luarnya (kesatuan luar).

2.2.14.1 DFD (Data Flow Diagram)

DFD atau singkatan dari *Data Flow Diagram* merupakan representasigrafik dari suatu sistem yang menunjukan proses atau fungsi, aliran data, tempatpenyimpan data dan *entitas eksternal*. DFD juga digunakan untukmenggambarkan suatu sistem yang telah ada atau sistem baru yang akandikembangkan. Dengan menggunakan DFD, rancangan yang akan kita buat akanlebih terarah dan lebih rinci. Sehingga kita tidak akan mengalami kesulitan dalammelakukan perancangan. *Data flow diagram* memiliki empat komponen, antaralain akan dijelaskan dibawah ini.

1. *Terminator (external entity)*

Terminator mewakili *entity external* yang berkomunikasi dengan sistem yang sedang dikembangkan.*Terminator* merupakan kesatuan dilingkungan sistem. Biasanya terminator ini dikenal dengan nama entitas (*external*) sumber atau tujuan

(*source* dan *sink*). Terminator dapat juga berupa departemen, divisi atau sistem diluar sistem yang berkomunkasi dengan sistem yang dikembangkan.

2. Proses

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

a. Identifikasi Proses

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

b. Nama Proses

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

3. Penyimpanan data *(data store)*

Data store digunakan sebagai sarana untuk mengumpulkan data. Data storedisimbolkan dengan dua garis horizontal yang paralel dimana tertutup padasalah satu ujungnya atau dua garis horizontal saja. *Data store* ini biasanyaberkaitan dengan penyimpanan-pemyimpanan seperti *file* atau *database* yangberkaitan dengan penyimpanan secara komputerisasi, contohnya *file* pitamagnetic, *file* disket atau *file harddisk*. *Data store* juga berkaitan denganpenyimpanan data.

4. Alur Data *(Data Flow)*

Suatu data *flow* atau alur data dapat dipresentasikan dengan anak panah yang menunjukan arah menuju ke dan keluar dari suatu proses. Alur data inidigunakan untuk menerangkan perpindahan data atau satu paket data atauinformasi dari suatu bagian sistem ke bagian lainnya. Selain menunjukanarah, alur data pada model yang dibuat dapat merepresentasikan bit, karakter,pesan, formulir, bilangan *real* dan macam-macam informasi yang berkaitandengan komputer. Ada tiga tipe DFD, yaitu *context diagram* (diagram konteks), *data flow diagram fisik*, dan *data flow diagram logika*.

2.2.14.2 Data Flow Diagram Fisik

Data flow diagram fisik adalah representasi dari sebuah sistem yang menunjukan *entitas-entitas internal* dan *eksternal* dari sistem tersebut,*Entitasentitasinternal* adalah *personal*, tempat atau mesin dalam sistem tersebut yang *mentransformasi* data. Maka data *flow* diagram fisik tidak menunjukkan apa yang dilakukan, tetapi menunjukan dimana, bagaimana, dan siapa proses-proses dalam sebuah sistem dilakukan.

2.2.14.3 Data Flow Diagram Logika

Data flow diagram logika digunakan untuk menggambarkan sistem yangakan diusulkan (sistem yang baru). *Data flow diagram logika* tidak menekankanpada bagaimana sistem diterapkan, tetapi penekanannya hanya pada logika padakebutuhan-kebutuhan sistem, yaitu proses-proses apa secara logika yangdibutuhkan oleh sistem.

2.2.15 Kamus Data (Data Directory)

Menurut JOG mendefinisikan kamus data adalah sebagai berikut :

"Kamus data atau data directory adalah catalog data tentang fakta dan kebutuhan-kebutuhan informasi dari suatu sistem informasi"

Dengan menggunakan kamus data, analisis sistem dapat mendefinisikan data yang mengalir di sistem dengan lengkap.Pada tahap perancangan sistem, kamus data dapat digunakan untuk merancang input, output (laporan-laporan) dan merancang database program.

2.2.16 Software Pendukung

Dalam membangun sistem informasi diperlukan software pendukung yang menunjang dalam proses pembuatannya.*Software* pendukung yang digunakan dalam pembuatan aplikasi ini adalah Borland Delphi 7 dan Mysql sebagai database nya.

2.2.16.1 Borland Delphi 7.0

Bahasa Pemrograman di Delphi disebut bahasa prosedural artinya bahasa/sintaknya mengikuti urutan tertentu/prosedur.Ada jenis pemrograman non-prosedural seperti pemrograman untuk kecerdasan buatan seperti bahasa Prolog.Delphi termasuk keluarga visual sekelas Visual Basic atau Visual C, artinya perintah-perintah untuk membuat objek dapat dilakukan secara visual.Pemrogram tinggal memilih objek apa yang ingin dimasukkan kedalam *form/window*, lalu tingkah laku objek tersebut saat menerima *event*/aksi tinggal dibuat programnya. Delphi merupakan bahasa berorentasi objek, artinya nama

objek, properti dan metode/prosedur dikemas menjadi satu kemasan (*encapsulate*).

Delphi adalah sebuah perangkat lunak (bahasa pemrograman) untuk membuat program/aplikasi komputer berbasis Windows.Delphi merupakan bahasa pemograman berbasis objek, artinya semua komponen yang ada merupakan objek-objek. Ciri sebuah objek adalah memiliki nama, properti dan metode/prosedur. Delphi disebut juga *visual programming* artinya komponen-komponen yang ada tidak hanya berupa teks (yang sebenarnya program kecil) tetapi muncul berupa gambar-gambar.

Berikut ini adalah IDE yang ada pada *Borland Delphi 7.0* :

1. *Project*

   Project adalah sekumpulan modul. Jadi *project* (projek) adalah program aplikasi itu sendiri. *Project* disimpan dalam *file* berakhiran .DPR. *File* ini menyimpan seluruh komponen program, termasuk pilihan projek, pilihan *Environment*, pilihan file EXE dan segala sesuatu yang berhubungan dengan projek.

2. *Form Designer*

   Form Designer adalah suatu objek yang dipakai sebagai tempat bekerja program aplikasi. *Form* berbentuk jendela dan dapat dibayangkan sebagai kertas atau meja kerja yang dapat dilukisi atau diletakkan ke dalamnya objek-objek lain.

3. *Object Inspector*

Object Inspector adalah kotak alat yang berisi *icon-icon* untuk memasukkan objek tertentu ke dalam jendela *form*. Ketika anda mulai membuat suatu projek, Delphi 7.0 akan otomatis menyediakan *icon-icon* objek yang sering dipakai, sesuai kategori aplikasi yang anda buat, misalnya DLL dan sebagainya.

4. *Properties*

Properties digunakan untuk menentukan properti suatu objek. Suatu objek biasanya mempunyai beberapa properti yang dapat diatur langsung dari jendela *properties* atau lewat kode program. Setting properti akan menentukan cara kerja dari objek yang bersangkutan saat program aplikasi dijalankan, misalnya menentukan warna, bingkai, pengambilan data dan lain-lain.

5. *Code Editor*

Code Editor adalah fasilitas yang disediakan oleh delphi untuk mengetikan kode program. Kode program ini akan mengontrol dan menentukan jalannya suatu objek.

6. *Event*

Event adalah peristiwa atau kejadian yang diterima oleh suatu objek, misalnya klik seret, tunjuk dan lain-lain.

2.2.16.2 MYSQL

MySQL adalah sebuah aplikasi *Relational Database Management Server* (RDBMS) yang sangat cepat dan kokoh. Dengan menggunakan

MySQL Server maka data dapat diakses oleh banyak pemakai secara bersamaan sekaligus dapat membatasi akses para pemakai berdasarkan *privilege* (hak akses) yang diberikan. MySQL menggunakan bahasa SQL (*Structured Query Language*) yaitu bahasa standar yang digunakan untuk pemrograman *database*. MySQL dipublikasikan sejak 1996, tetapi sebenarnya dikembangkan sejak 1979. MySQL telah memenangkan penghargaan Linux Journal Reader's Choice Award selama tiga tahun. MySQL sekarang tersedia dibawah izin *open source*, tetapi juga ada izin untuk penggunaan secara komersial. Keunggulan dari MySQL adalah:

a. Bersifat *open source*.
b. Sistem yang digunakan oleh perangkat lunak ini tidak memberatkan kerja dari *server*, karena dapat bekerja di *background*.
c. Mempunyai koneksi yang stabil dan kecepatan yang tinggi.

3.1.1 *Structured Query Language* (SQL)

Structured Query Language (SQL) adalah bahasa yang digunakan untuk berkomunikasi dengan *database*. Perintah-perintah SQL digunakan untuk melakukan pekerjaan-pekerjaan tertentu pada *database* seperti *update* data atau untuk mengambil data dari *database*. Pada subbab-subbab berikut

ini akan dijelaskan mengenai perintah *select, insert, update dan delete* pada SQL.

a. Perintah *Select*

Perintah *select* digunakan untuk mengambil dan menampilkan data dari *database* sesuai dengan syarat-syarat yang ditentukan. Berikut ini adalah format perintah *select* yang umum digunakan:

```
SELECT  column1  [,column2,etc]
FROM  table name
[WHERE condition];
```

keterangan: [] = optional

Column menunjukkan nama kolom dari tabel yang ingin diambil. *Column* dapat lebih dari satu atau menggunakan "*" untuk mengambil semua kolom. *Tablename* menunjukkan nama tabel yang ingin digunakan. Klausa *where* menyaring data berdasarkan kondisi yang dituliskan setelah kata *where*. Operator-operator yang digunakan dalam klausa *where* antara lain dapat dilihat pada tabel berikut:

Tabel 2.1 Operator Untuk Perintah SQL

Lambang	Arti
=	Sama dengan
>	Lebih besar dari
<	Lebih kecil dari
>=	Lebih besar sama dengan
<=	Lebih kecil sama dengan
<>	Tidak sama dengan
LIKE	(dijelaskan di bawah ini)

Operator *LIKE* dapat menyeleksi data sesuai dengan *string* yang diinginkan. Simbol "%" digunakan untuk mencocokkan karakter yang muncul sebelum atau sesudah karakter yang diinginkan. Contohnya adalah sebagai berikut:

```
SELECT  nama_depan, kota
FROM  karyawan
WHERE nama_depan  LIKE  'Sam%';
```

Perintah SQL di atas memiliki *output* semua nama depan yang diawali dengan "Sam". Simbol "%" dapat pula diletakkan di depan atau di belakang. Selain seperti yang dijabarkan di atas, perintah *select* juga memiliki format lain yang lebih lengkap yang mendukung fungsi agregasi, yaitu:

```
SELECT   [ALL | DISTINCT]  column1  [,column2]
FROM  table1  [,table2]
[WHERE"conditions"]
```

[GROUPBY"column-list"]

[HAVING "conditions]

[ORDER BY "column-list" [ASC | DESC]]

Tabel 2.2 Fungsi Agregasi

Nama Fungsi	**Kegunaan**
SUM( )	Untuk menghitung jumlah nilai pada suatu kolom
AVG( )	Untuk menghitung nilai rata-rata pada suatu kolom
MAX( )	Untuk mencari nilai maksimum pada suatu kolom
MIN( )	Untuk mencari nilai minimal pada suatu kolom
COUNT( )	Untuk menghitung jumlah baris pada suatu kolom

Keyword *ALL* digunakan untuk mengambil semua data sedangkan *DISTINCT* digunakan untuk mengambil data yang unik dimana baris yang sama akan dijadikan satu baris saja.

b. Klausa *Group By*

Klausa *group by* akan mengumpulkan baris data dari tabel sesuai dengan kolom yang dispesifikasikan dan memperkenankan

adalah fungsi agregasi pada satu atau lebih kolom. Untuk lebih jelas dapat dilihat pada contoh berikut:

```
SELECT MAX  (gaji), departemen
FROM karyawan
GROUP BY departemen;
```

Query ini akan mengambil dan menampilkan gaji maksimum dari karyawan dari tiap departemen dan dikelompokkan berdasarkan nama departemennya.

c. Klausa *Having*

Klausa *having* dapat menspesifikasikan kondisi dari baris untuk tiap *group*, dengan kata lain tiap baris akan diambil berdasarkan kondisi yang dispesifikasikan. Jika menggunakan klausa *having* harus diikuti dengan klausa *group by*. Contoh penggunaan dari klausa *having* adalah sebagai berikut:

```
SELECT   departemen,avg(gaji)
FROM   karyawan
GROUP  BY departemen
HAVING  avg(gaji) > 20000;
```

Pada *query* di atas jika pada *query* tersebut hanya menggunakan *group by* saja maka akan mengambil rata-rata gaji karyawan pada tiap departemen. Tetapi bila ditambahkan dengan klausa *having* maka dapat dispesifikasikan rata-rata gaji yang diambil adalah yang bernilai lebih dari 20000.

d. Klausa *Order By*

Klausa *order by* merupakan klausa *optional* yang digunakan jika kita ingin menampilkan hasil dari *query* secara urut, baik urut secara *ascending* (*ASC*)maupun secara *descending* (*DESC*), sesuai dengan kolom yang dispesifikasikan pada *order by*. Contoh penggunaannya jika diinginkan hasil urut data karyawan berdasarkan umurnya secara *ascending* adalah sebagai berikut:

```
SELECT  kode_karyawan, nama_depan, umur, gaji
FROM   karyawan
WHERE   departemen=‘Penjualan’
ORDER   BY umur ASC;
```

Bila ingin mengurutkan berdasarkan dua atau lebih kolom, maka harus dipisahkan dengan tanda koma, seperti dijabarkan di bawah ini:

```
SELECT  kode_karyawan, nama_depan, umur, gaji
FROM  karyawan
WHERE   departemen=‘Penjualan’
ORDER  BY  umur, gaji  DESC;
```

Query di atas akan mengurutkan hasil *query* berdasarkan umur, baru kemudian mengurutkan berdasarkan gaji secara *descending*.

e. Join Table

Dalam kenyataannya, *query* tidak hanya dari satu tabel saja, mungkin dua atau lebih. Untuk itu diperlukan *join table* untuk menggabungkan dua atau lebih tabel yang berelasi dalam suatu *database relational*. Dengan *join table* maka hanya dengan satu *query* yang menggunakan perintah *select* dapat menggabungkan data dari dua atau lebih tabel ke dalam satu hasil *query*.

Antar dua tabel yang berelasi pasti terdapat satu kolom yang sama. Kolom inilah yang digunakan untuk menggabungkan dua tabel tersebut. Caranya adalah sebagai berikut:

```
SELECT  konsumen.nama_depan, penjualan.no_faktur
FROM  konsumen,penjualan
WHERE
konsumen.kode_konsumen=penjualan.kode_konsumen;
```

Join table di atas merupakan *inner join* dimana akan menghasilkan semua nama depan dari konsumen (yang diambil dari tabel konsumen) dan nomor faktur penjualan (yang diambil dari tabel penjualan) dari semua baris dimana kode konsumen di tabel konsumen sama dengan kode konsumen di tabel penjualan.

f. Perintah *Insert*

Perintah *insert* digunakan untuk memasukkan atau menambah baris data ke dalam sebuah tabel. Format penulisan perintah *insert* adalah sebagai berikut:

```
INSERT   INTO tablename (first_column, …, last_column)
VALUES  (first_value, …, last value);
```

Parameter setelah *tablename* adalah daftar kolom-kolom yang dipisahkan dengan koma, diikuti dengan kata *values* lalu diikuti lagi dengan daftar nilai yang urutannya sesuai dengan daftar kolom sebelum kata *values*. Untuk nilai yang bernilai *string* harus diapit dengan tanda petik satu sedangkan untuk yang bernilai integer tidak. Contohnya adalah sebagai berikut:

```
INSERT INTO karyawan (nama_depan, nama_belakang, umur, kota)VALUES (‘Sam’, ‘Ali’, 21, ‘Surabaya’);
```

Query di atas akan menambahkan baris baru pada tabel karyawan dengan *field* nama_depan bernilai “Sam”, *field* nama_belakang bernilai “Ali”, *field* umur bernilai 21 dan *field* kota bernilai “Surabaya”.

g. Perintah *Update*

Perintah *update* digunakan untuk mengubah data yang sudah ada berdasarkan kondisi tertentu yang dijabarkan oleh *keyword where*. Format perintah lengkapnya adalah sebagai berikut:

```
UPDATE  tablename
SET  columnname =newvalue[,nextcolumn = newvalue2..]
WHERE  columnname <OPERATOR> value
[and|or column" <OPERATOR> "value"];
```

keterangan : [] = optional

Contoh penggunaan :

```
UPDATE  karyawan
SET  umur = 15
WHERE  nama_depan = ‘Sam’;
```

h. Perintah *Delete*

Perintah delete digunakan untuk menghapus baris dari suatu tabel. Formatnya adalah sebagai berikut:

```
DELETE  FROM tablename
WHERE column  name <OPERATOR> value [and|or column
<operator> value]
```

keterangan: [] = optional

Contoh penggunaannya adalah seperti di bawah ini:

```
DELETE FROM karyawan
WHERE  nama_depan = ‘Sam’;
```

Contoh di atas akan menghapus baris dari tabel “karyawan” yang field “nama_depan”-nya bernilai “Ali”. Untuk menghapus seluruh baris dari suatu tabel, kosongkan klausa *where*.