BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Perusahaan

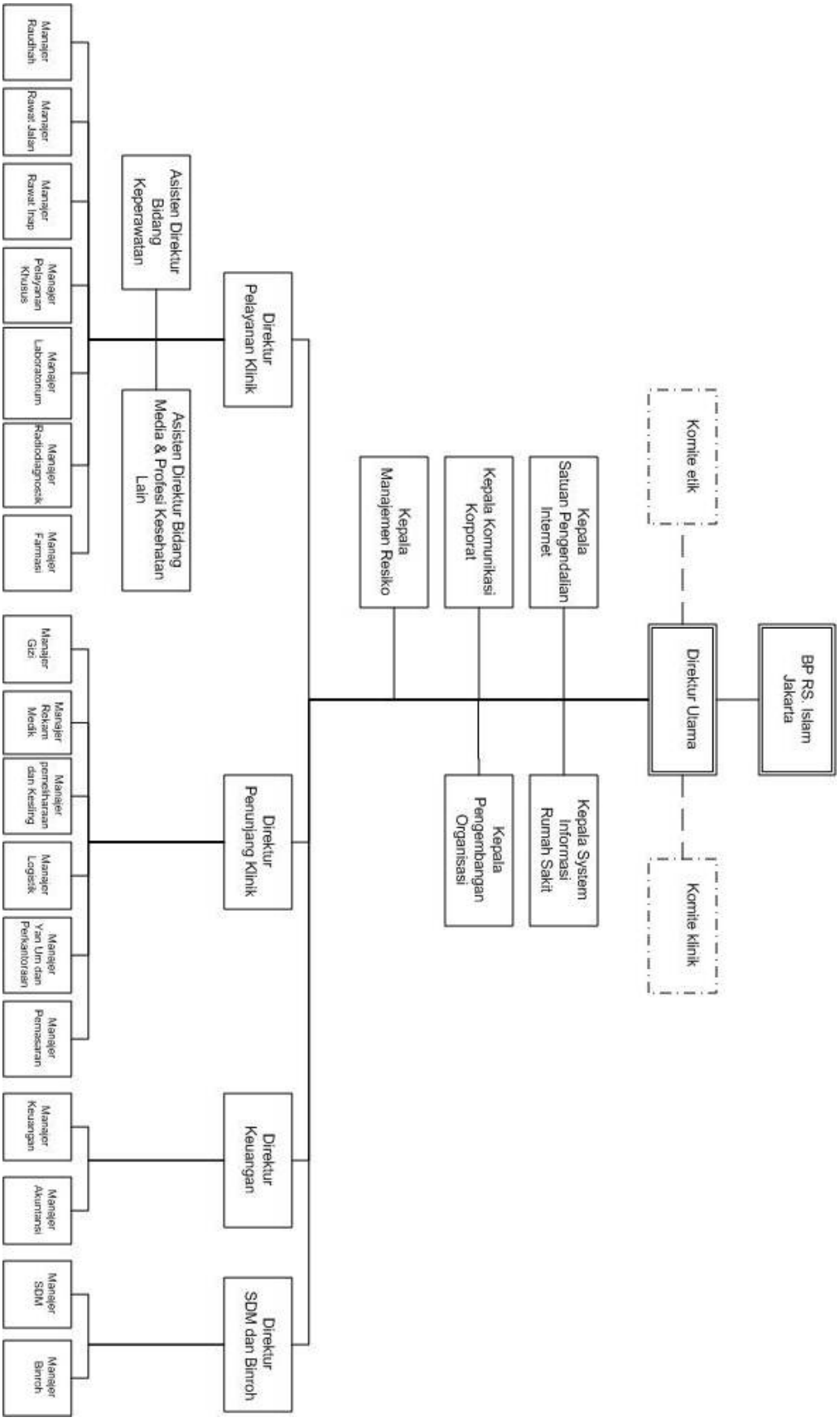
Awal berdirinya Rumah Sakit Islam adalah bermulai dari dirasakanya kebutuhan akan pelayanan kesehatan yang bernapaskan islam. Dalam pembangunannya itu pendanaan didapatkan sebagian melalui bantuan sebuah lembaga dari Departemen Luar Negeri Belanda yaitu SCCFA (*State Committee for coordinating Foreign Aid)*. Akhirnya pada tanggal 23 Juni 1971, Rumah Sakit Islam berdiri dengan kokohnya yang diresmikan oleh Presiden Soeharto dengan acara pembukaan yang cukup meriah dan khidmat.

Bila dibandingkan dengan sekarang keadaanya masih sangat sederhana, antara lain tampak dari gedung perawatan yang hanya memiliki kapasitas 56 tempat tidur, ruang kantor, poliklinik, laboratorium, apotik dan dapur. Ditambah satu asrama putri dan rumah dinas Dokter. Setelah itu perkembangan dari tahun ketahun dilakukan untuk mencapai Rumah Sakit yang lebih baik lagi, seperti pada tahun 1972, dengan didapatnya bantuan dari Presiden Soeharto Rumah Sakit Islam dapat membangun sebuah ruang operasi yang cukup memadai. Kemudian pada tahun 1973, dibangun gedung perawatan untuk kelas I dengan kapasitas 16 tempat tidur. Pada tanggal 23 Desember 1973 bekerjasama dengan Universitas Yarsi (dahulu Sekolah Tinggi Kedokteran Yarsi) dalam mengelola dua ruang perawatan untuk kelas III yang berada dalam di dalam komplek STK Yarsi. Lalu pada tahun 1975 dibangunlah sebuah gedung perawatan kelas utama yang berkapasitas 26 tempat tidur.

Pada tahun 1982, Rumah Sakit Islam berhasil membangun ruang perawatan untuk *Intensif Care Unit* (ICU) dengan kapasitas delapan tempat tidur yang dilengkapi dengan fasilitas gas medik sentral. Tahun 1983/1984 Rumah Sakit Islam telah dapat memperluas diri dengan bangunan *medical check up* (unit uji medik dan diagnostik), ruang senam hamil, ruang bayi sehat, ruang perpustakaan, pos keamanan dan musholla. Sementara pada tahun 1984/1985 Rumah Sakit Islam merampungkan pembangunan gedung Akademi keperawatan Kesehatan dan koperasi “Sejahtera”. Bersamaan dengan itu juga dilakukan renovasi dapur menjadi paviliun badar untuk perawatan pasien anak-anak. Pembangunan ini didukung dana sumbangan dari PT. ASTRA, dan dibangun pula dapur gizi dan tempat cuci/*laundry* yang baru. Tahun 1986/1987 dibangun sebuah gedung perawatan kelas I, yaitu Paviliun Multazam dengan kapasitas 48 tempat tidur, bersamaan dengan itu dibangun juga mushalla Asyifa dan renovasi instalasi rekam medik.

Tahun 1995 Rumah Sakit Islam dinyatakan sebagai Rumah Sakit kelas Utama yang merupakan klasifikasi tertinggi dan disejajarkan dengan Rumah Sakit Pemerintah tipe B. Dalam Kurun Waktu 1998-2000 Rumah Sakit Islam juga berhasil merenovasi sarana parkir, sarana umum dan sarana lainnya yang menambah citra positif bagi Rumah Sakit Islam Jakarta.

Kini dalam kurun waktu 37 tahun Rumah Sakit Islam telah berkembang dengan mantap secara bertahap, baik dari segi manajemen, fisik dan pelayanannya apalagi setelah didapatnya ISO 9001:2000, hal ini semakin menambah citra positif bagi Rumah Sakit.



Gambar 2.1 Struktur Organisasi Rumah Sakit Islam Jakarta

Adapun *Job description* dari Dept. Collector bagian penagihan, yaitu

1. Kolektor merekapitulasi data pasien yang berasal dari perusahaan-perusahaan yang melakukan kerja sama dengan pihak rumah sakit.
2. Melakukan rekapitulasi tagihan pasien per perusahaan.
3. Mencetak hasil rekapitulasi tagihan pasien per perusahaan.
4. Melakukan penagihan terhadap perusahaan-perusahaan sebagai penanggung pasien yang memiliki piutang per periode.
5. Mengolah data piutang per perusahaan.
6. Mengolah data pelunasan piutang per perusahaan.

2.2 Landasan Teori

Pada landasan teori ini akan menerangkan mengenai teori-teori yang berhubungan dengan aplikasi pengolahan tagihan piutang perusahaan pada Rumah Sakit Islam Jakarta baik mengenai sistem informasi, database dan aplikasi pembangun aplikasi.

2.2.1 Pengertian Piutang

Piutang adalah adalah salah satu jenis transaksi akuntansi yang mengurusi penagihan konsumen yang berhutang pada seseorang, suatu perusahaan, atau suatu organisasi untuk barang dan layanan yang telah diberikan pada konsumen tersebut. Pada sebagian besar entitas bisnis, hal ini biasanya dilakukan dengan membuat tagihan dan mengirimkan tagihan tersebut kepada konsumen yang akan dibayar dalam suatu tenggat waktu yang disebut termin kredit atau pembayaran. [7]

Pada dasarnya piutang dapat dikelompokkan menjadi 3 jenis, yakni:

1. Piutang dagang

Piutang dagang adalah tagihan perusahaan kepada pelanggan sebagai akibat adanya penjualan barang atau jasa secara kredit, dalam hal ini tagihan tersebut tidak disertai dengan surat perjanjian yang formal, melainkan karena unsur kepercayaan dan kebijakan perusahaan di mana dalam penjualannya telah ditetapkan syarat penjualan misalnya 2/10,n/30. Hal ini berarti piutang yang timbul diharapkan akan dapat diterima dalam jangka waktu paling lama 30 hari sejak tanggal transaksi. Apabila pelanggan membayar dalam jangka waktu kurang dari 10 hari setelah tanggal transaksi maka akan diberikan potongan/diskon sebesar 2% dari harga jual. Piutang dagang umumnya berjangka waktu kurang dari satu tahun sehingga dilaporkan sebagai aktiva lancar. [6]

2. Piutang nondagang

Piutang non dagang terdiri atas macam-macam tagihan yang tidak termasuk dalam piutang dagang maupun piutang wesel. Misalnya piutang kepada karyawan perusahaan, direksi perusahaan, dan piutang kepada cabang-cabang perusahaan. [6]

3. Piutang wesel

Piutang wesel merupakan piutang yang lebih formal dibandingkan piutang dagang karena didalamnya memerlukan perjanjian tertulis debitur kepada kreditur untuk membayar sejumlah uang yang tercantum dalam surat janji tersebut pada waktu tertentu di masa yang akan datang. Umumnya piutang wesel berjangka waktu lebih dari 60 hari, apabila piutang wesel

berjangka waktu kurang dari satu tahun dilaporkan dalam neraca sebagai aktiva lancar sedangkan untuk piutang wesel berjangka waktu lebih dari satu tahun diperlakukan sebagai piutang jangka panjang. [6]

2.2.2 Konsep Dasar Sistem

Sistem dapat didefinisikan menurut dua kelompok pendekatan sistem, yaitu yang melakukan pada prosedurnya dan yang menekankan pada komponen atau elemen. Definisi sistem berdasarkan pendekatan yang menekankan pada prosedur adalah sebagai berikut :

> "Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu" [2]

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]

Suatu sistem memiliki karakteristik atau sifat-sifat tertentu, yaitu memiliki komponen-komponen *(components)*, batas sistem *(boundary)*, lingkungan luar sistem *(environment)*, penghubung *(interprest)*, masukan *(input)*, keluaran *(output)*, pengolah *(process)* dan sasaran *(objective)* dan tujuan *(goal)*.

1. Komponen Sistem (*System Components*)

Komponen sistem atau elemen-elemen sistem dapat berupa suatu sub sistem atau bagian-bagian dari sistem. Setiap sistem baik besar maupun kecil, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai suatu yang lebih besar yang disebut *supra system.*

2. Batas Sistem (*System Boundary*)

Batas sistem merupakan daerah-daerah yang membatasi antara satu sistem dengan sistem lainnya dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menujukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem (*System Environment*)

Lingkungan luar sistem dari suatu sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem yang dapat bersifat menguntungkan dan dapat pula merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem yang harus dijaga dan dipelihara. Sedangkan yang merugikan harus ditahan dan dikendalikan, karena akan mengganggu kelangsungan hidup sistem.

4. Penghubung Sistem *(System Interprest)*

Penghubung sistem merupakan media penghubung antara satu sub sistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke

subsistem yang lainnya. Dengan penghubung akan terjadi interaksi antar subsistem, sehingga membentuk satu kesatuan.

5. Masukan Sistem *(System Input)*

Masukan adalah suatu energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Contoh *maintenance input* di dalam sistem komputer adalah program, yang digunakan untuk mengoperasikan komputer. Sedangkan *signal input* adalah energi yang diproses untuk mendapatkan keluaran. Contoh *signal input* di dalam sistem komputer adalah data, yang dapat diolah menjadi Informasi.

6. Keluaran Sistem *(System Output)*

Keluaran (*Output*) merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna.

7. Pengolah Sistem (*System Process*)

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya, yang bertugas untuk merubah masukan menjadi keluaran.

8. Sasaran Sistem (*System Objective*)

Suatu sistem pasti memiliki tujuan (*goal*) atau sasaran (*objective*). Suatu operasi sistem akan berguna dan berhasil apabila mencapai sasaran atau tujuannya. Sasaran sistem sangat menentukan masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem.

2.2.3 Konsep Dasar Data dan Informasi

Konsep Informasi tidak akan lepas dari keterhubungan dengan data. Untuk mendapatkan Informasi yang valid maka dibutuhkan data-data yang berhubungan antara satu dengan yang lain agar *output* sesuai dengan yang diharapkan. [4]

2.2.3.1 Data

Data adalah kumpulan dari fakta-fakta, kejadian-kejadian yang dapat berupa simbol, angka, huruf, dan lain-lain yang berguna bagi suatu pengolahan data (proses) atau sebagai masukan (*input*) bagi suatu proses. Data merupakan deskripsi dari sesuatu dan kejadian yang kita hadapi. [4]

2.2.3.2 Informasi

Informasi merupakan hasil pengolahan dari sebuah model, formasi, organisasi, ataupun suatu perubahan bentuk dari data yang memiliki nilai tertentu, dan bisa digunakan untuk menambah pengetahuan bagi yang menerimanya. Dalam hal ini, data bisa dianggap sebagai obyek dan Informasi adalah suatu subyek yang bermanfaat bagi penerimanya. Informasi juga bisa disebut sebagai hasil pengolahan atau pemrosesan data.[4]

Informasi didefinisikan oleh Robert A.Leitch dan K. Roscoe Davis sebagai berikut :

> “ Sistem Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi yang mendukung

operasi bersifat manajerial dan kegiatan strategi-strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan."

Sumber dari Informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata, tetapi data merupakan bentuk yang masih mentah dan belum dapat memberikan arti banyak bagi pemakai, sehingga perlu diolah lebih lanjut untuk menghasilkan Informasi yang dibutuhkan. Dengan kata lain Informasi adalah hasil dari pengolahan data.

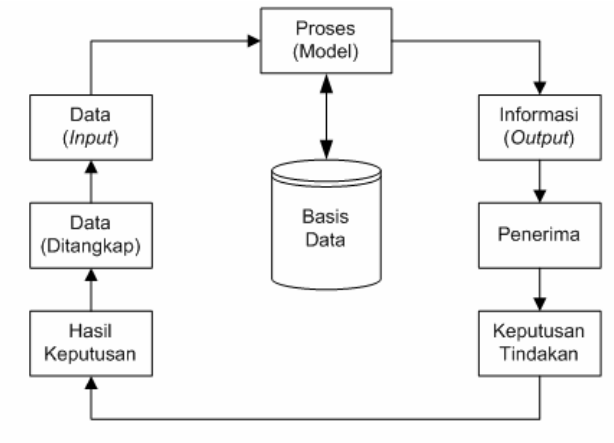
1. Nilai Informasi

Nilai dari Informasi *(value of information)* ditentukan dari dua hal, yaitu manfaat dan biaya mendapatkannya. Suatu Informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Kegunaan Informasi adalah untuk mengurangi hal ketidakpastian di dalam proses pengambilan keputusan tentang suatu keadaan.

2. Siklus Informasi

Data agar menjadi lebih berarti dan berguna dalam bentuk Informasi, maka perlu diolah melalui suatu model tertentu. Data yang telah diolah tersebut kemudian diterima oleh penerima, lalu penerima membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai *input*, dan diproses kembali lewat suatu model dan seterusnya sehingga membentuk suatu siklus. Siklus ini disebut dengan siklus

Informasi (*information cycle*) atau disebut pula siklus pengolahan data (*processing cycles*).



Gambar 2.2 Siklus Informasi [4]

3. Pengertian Sistem Informasi

Sistem Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung operasi bersifat managerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan yang diperlukan.

4. Konsep Dasar Sistem Informasi

Sistem Informasi dapat didefinisikan sebagai suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan Informasi. Pengertian lain dari sistem Informasi adalah sekumpulan prosedur organisasi yang pada saat dilaksanakan dan memberikan Informasi bagi pengambil keputusan dan/atau untuk mengendalikan Informasi.

5. Komponen Sistem Informasi

Dalam membangun suatu sistem Informasi diperlukan penggabungan elemen-elemen pendukung tersebut antara lain :

1. *Software*, merupakan suatu program komputer, struktur data, dan dokumen-dokumen yang saling berhubungan yang digunakan dalam metode logika dan prosedur yang dibutuhkan.
2. *Hardware*, merupakan perangkat elektronik yang memiliki kemampuan untuk melakukan proses komputerisasi.
3. *Brainware*, adalah pengguna dan operator perangkat keras atau perangkat lunak.
4. Data, berupa salinan-salinan manual dan deskripsi Informasi yang menggambarkan operasi sistem.

6. Tujuan Sistem Informasi

Sistem Informasi memiliki beberapa tujuan, yaitu:

1. Integrasi sistem
   a. Menghubungkan sistem individu/kelompok
   b. Pengkolektifan data dan penyambungan secara otomatis
   c. Peningkatan koordinasi dan pencapaian sinergi
2. Efisiensi pengelolaan
   a. Penggunaan basis data dalam upaya kesamaan pengadministrasian data
   b. Pengelolaan data berkaitan dengan karakteristik Informasi
   c. Penggunaan dan pengambilan Informasi

3. Dukungan keputusan untuk manajemen
    a. Melengkapi Informasi guna kebutuhan proses pengambilan keputusan.
    b. Akuisisi Informasi eksternal melalui jaringan komunikasi.
    c. Ekstraksi dari Informasi internal yang terpadu.

7. Manfaat Sistem Informasi

Sistem Informasi memiliki beberapa manfaat, yaitu:

1. Menghemat tenaga kerja
2. Peningkatan efisiensi
3. Mempercepat proses
4. Perbaikan dokumentasi
5. Pencapaian standar
6. Perbaikan keputusan

2.2.4 Konsep Pengembangan Sistem

Pengembangan sistem secara umum adalah suatu tahap dimana di dalamnya terdapat identifikasi komponen-komponen sistem Informasi yang akan dirancang secara rinci yang bertujuan untuk memberikan gambaran kepada pengguna atau *user* mengenai sistem yang baru. Sedangkan desain sistem secara terinci dimaksudkan untuk pembuat program komputer dan ahli teknik lainnya yang akan mengimplementasikan sistem. [1]

Penggambaran dan rancangan model sistem Informasi secara logika dapat dibuat dalam bentuk Diagram Konteks dan Diagram Alir Data (DAD) atau *Data Flow Diagram* (DFD).

2.2.4.1 Diagram Konteks

Diagram Konteks adalah diagram tingkat tinggi dari Diagram Alir Data yang merupakan gambaran global dari sistem Informasi yang menggambarkan aliran-aliran data ke dalam maupun keluar suatu sistem dan merupakan alat yang digunakan untuk melihat batasan antara sistem dengan *eksternal entity* . [1]

2.2.4.2 Diagram Alir Data

Diagram Alir Data atau *Data Flow Diagram* (DFD) adalah suatu model yang menjelaskan arus data mulai dari pemasukan sampai dengan keluaran data. Tingkatan DFD dimulai dari diagram konteks yang menjelaskan secara umum suatu sistem atau batasan sistem aplikasi yang akan dikembangkan. Kemudian DFD dikembangkan menjadi DFD tingkat 0 atau level 0 dan kemudian DFD level 0 dikembangkan lagi menjadi level 1 dan selanjutnya sampai sistem tersebut tergambarkan secara rinci menjadi tingkatan-tingkatan lebih rendah lagi. [1]

DFD merupakan penurunan atau penjabaran dari diagram konteks. Dalam pembuatan DFD harus mengacu pada ketentuan sebagai berikut :

1. Setiap penurunan level yang lebih rendah harus mempresentasikan proses tersebut dalam spesifikasi proses yang jelas.
2. Penurunan dilakukan apabila memang diperlukan.
3. Tidak semua bagian dari sistem harus ditunjukkan dengan jumlah level yang sama.

Simbol-simbol yang digunakan pada diagram alir data atau *data flow diagram* antara lain :

1. Kesatuan Luar (*External Entity*)

   Kesatuan luar atau *external entity* merupakan kesatuan luar di lingkungan sistem, bisa berupa orang, organisasi, atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan masukan bagi sistem atau keluaran dari sistem.

2. Proses (*Process*)

   Proses adalah kegiatan kerja yang dilakukan oleh manusia, mesin atau komputer dari hasil suatu arus data yang masuk kedalam proses untuk arus data yang akan keluar dari proses.

3. Arus Data (*Data Flow*)

   Arus data mengalir diantara proses, simpanan data dan kesatuan luar. Arus data ini menunjukkan masukan dan keluaran dari suatu proses.

4. Penyimpanan Data (*Data Storage*)

   Simpanan data merupakan simpanan dari data yang dapat berupa suatu arsip atau *database* pada sistem komputer, suatu arsip atau catatan manual, suatu kotak tempat data di meja seseorang, atau suatu agenda/buku.

2.2.4.3 Kamus Data

Kamus data atau *data dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan Informasi dari suatu sistem Informasi. Kamus data

berisikan beberapa tabel yang memuat nama dari data , tipe dari data yang akan kita gunakan di dalam menyusun program. Kamus data digunakan untuk menghindari duplikasi elemen-elemen dan menghindari konflik antara elemen-elemen tersebut. Dengan kamus data , dapat melihat elemen-elemen yang terdapat dalam sebuah sistem. [1]

2.2.5 Konsep Basis Data

Basis data adalah kumpulan *file* yang saling berinteraksi, relasi tersebut biasa ditunjukan dengan kunci dari tiap *file* yang ada. Satu basis data menunjukan satu kumpulan data yang dipakai dalam satu lingkup perusahaan atau instansi.

Terdapat dua buah teknik perancangan basis data, yaitu dengan membuat *Entity Relationship Diagram* atau dengan menerapkan normalisasi terhadap struktur tabel yang telah diketahui.

1. *Entity Relationship Diagram* (Diagram E-R)

Model E-R yang berisi komponen-komponen entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang merepresentasikan seluruh fakta dari yang kita tinjau, dapat digambarkan dengan lebih sistematis dengan menggunakan diagram E-R. [1]

Objektif utama dari pembuatan diagram E-R adalah untuk menunjukan objek-objek apa saja yang dilibatkan dalam sebuah basis data dan bagaimana hubungan yang terjadi diantara objek-objek tersebut. Pada sebuah sistem yang ruang lingkupnya lebar dan kompleks, penggambaran atribut-atribut dalam

sebuah diagram E-R sering kali mengganggu objek yang ingin dicapai. Maka dapat dipisahkan pendeklarasian atribut-atribut dari diagram E-R dan dinyatakan dalam kamus data. Atribut yang berfungsi sebagai *key* juga dibedakan dengan yang bukan *key* dengan menggarisbawahi atribut tersebut. Model data diagram E-R dibentuk dari empat komponen dasar, yaitu :

1. Entitas (*Entity*)

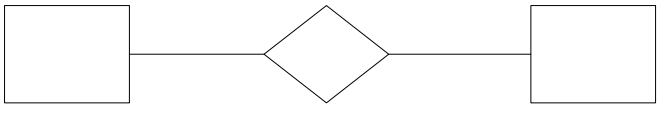
   Entitas adalah sesuatu yang ada dan dapat berupa orang,tempat, benda, peristiwa atau konsep yang bisa memberi atau mengandung Informasi

2. Relasi (*Relationship*)

   Jenis hubungan untuk entitas yang ada di dalam diagram adalah sebagai berikut :

   a. *One to One Relationship* (Satu ke Satu)

      Setiap entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu pula sebaliknya.

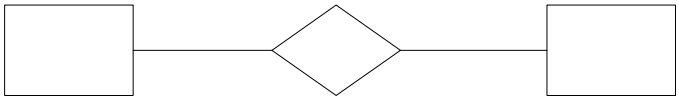


Gambar 2.3 *One to One Relationship*

   b. *One to Many Relationship* (Satu ke Banyak)

      Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak

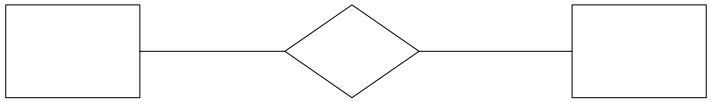
sebaliknya dimana setiap entitas pada himpunan B berhubungan paling banyak dengan satu entitas pada entitas A.



Gambar 2.4 *One to Many Relationship*

c. *Many to One Relationship* (Banyak ke Satu)

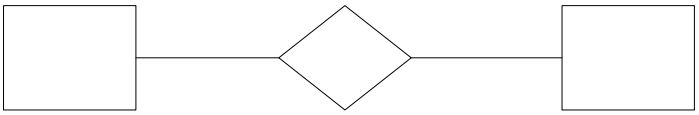
Setiap entitas pada himpunan A berhubungan paling banyak satu entitas pada himpunan entitas B, tetapi tidak sebaliknya dimana setiap entitas pada himpunan entitas B berhubungan dengan banyak entitas pada entitas himpunan B.



Gambar 2.5 *Many to One Relationship*

d. *Many to Many Relationship* (Banyak ke Banyak)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, demikian pula sebaliknya.



Gambar 2.6 *Many to Many Relationship*

3. Atribut

Setiap entitas memiliki atribut-atribut atau elemen data yang mencirikan entitas tersebut.

4. *Key* (Kunci)

Sebuah atribut atau set atribut yang nilainya mengidentifikasikan entitas secara unik dalam suatu entitas. *Key* memiliki beberapa jenis sesuai dengan kegunaannya masing-masing, yaitu *primary key* (kunci utama), *foreign key* (kunci tamu).

a. *Primary Key* (Kunci Utama)

Primary key atau disebut juga kunci utama merupakan suatu kunci unik yang mengidentifikasi masing-masing baris di suatu tabel. Isi dari atribut yang menjadi kunci utama harus unik, dalam artian bahwa data yang satu dengan yang lain tidak boleh sama. Hal ini dimaksudkan agar tidak terjadi redudansi data. Contoh dari kunci utama ini adalah kode perusahaan dimana data dari kode perusahaan tidak sama antara perusahaan satu dengan yang lainnya. Gabungan primary key dari himpunan entitas yang berpartisipasi dalam sebuah relasi membentuk *super key* dari himpunan relasi tersebut. [1]

*b. Foreign Key (*Kunci Tamu)

Foreign key adalah *Primary Key* yang ada pada tabel lain yang akan dihubungkan dengan tabel yang lainnya, dikarenakan datanya saling berhubungan. [1]

*c. Super Key (*Kunci Super)

Super key dari sebuah himpunan entitas adalah sekumpulan satu atau lebih atribut yang memiliki nilai unik terhadap setiap entitasnya. [1]

2.2.6 *Database Management System* (DBMS)

Database adalah kumpulan data yang saling berkaitan, berhubungan yang disimpan secara bersama-sama sedemikian rupa tanpa pengulangan yang tidak perlu, untuk memenuhi berbagai kebutuhan. Data-data ini harus mengandung semua Informasi untuk mendukung semua kebutuhan sistem.

Proses dasar yang dimiliki oleh *database* ada 4, yaitu :

1. Pembuatan data-data baru (*create database*)
2. Penambahan data (*insert*)
3. Mengubah data (*edit*)
4. Menghapus data (*delete*)

Sistem manajemen *database* (*Database Management System*) merupakan sistem pengoperasian dan sejumlah data pada komputer. Dengan sistem ini dapat merubah data, memperbaiki data yang salah dan menghapus

data yang tidak dapat dipakai. Sistem manajemen database merupakan suatu perluasan *software* sebelumnya mengenai *software* pada generasi komputer yang pertama.

Salah satu tujuan DBMS adalah untuk menyediakan fasilitas atau antarmuka (*interface*) dalam melihat atau menikmati data kepada pemakai. Untuk itu, sistem tersebut seringkali akan menyembunyikan detail tentang bagaimana data disimpan, dipakai atau dipelihara. Karena itu, seringkali data yang dilihat oleh pemakai sebelumnya berbeda dengan yang tersimpan secara fisik.

Perangkat lunak yang termasuk DBMS adalah : dBase III+, dBaseIV, Fox Base, Interbase, Ms.Access dan Borland Paradox atau Borland-Interbase

2.2.7 Perangkat Lunak

Perangkat lunak merupakan suatu kebutuhan yang harus dipenuhi oleh suatu sistem. Kebutuhan ini diperlukan untuk mencapai suatu tujuan.

2.2.7.1 Microsoft Access

Microsoft Access adalah sebuah program aplikasi basis data komputer relasional yang ditujukan untuk kalangan rumahan dan perusahaan kecil hingga menengah. Aplikasi ini merupakan anggota dari beberapa aplikasi Microsoft Office, selain tentunya Microsoft Word, Microsoft Excel, dan Microsoft PowerPoint. Aplikasi ini menggunakan mesin basis data Microsoft Jet Database Engine, dan juga menggunakan tampilan grafis yang intuitif sehingga memudahkan pengguna.

Microsoft Access dapat menggunakan data yang disimpan di dalam *form*at Microsoft Access, Microsoft Jet Database Engine, Microsoft SQL Server, Oracle Database, atau semua kontainer basis data yang mendukung standar ODBC. Para pengguna/*programmer* yang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang kompleks, sementara para *programmer* yang kurang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang sederhana. Access juga mendukung teknik-teknik pemrograman berorientasi objek, tetapi tidak dapat digolongkan ke dalam perangkat bantu pemrograman berorientasi objek.

1. Pengguna Microsoft Access

Microsoft Access digunakan kebanyakan oleh bisnis-bisnis kecil dan menengah, di dalam sebuah organisasi yang kecil bahkan mungkin juga digunakan oleh perusahaan yang cukup besar, dan juga para *programmer* untuk membuat sebuah sistem buatan sendiri untuk menangani pembuatan dan manipulasi data.

Beberapa pengembang aplikasi profesional menggunakan Microsoft Access untuk mengembangkan aplikasi secara cepat (digunakan sebagai *Rapid Application Development*/*RAD tool*), khususnya untuk pembuatan purwarupa untuk sebuah program yang lebih besar dan aplikasi yang berdiri sendiri.

2. Fitur Microsoft Access

Salah satu keunggulan Microsoft Access dilihat dari perspektif *programmer* adalah kompatibilitasnya dengan bahasa pemrograman *Structured Query Languag*e (SQL), *query* dapat dilihat dan disunting sebagai

statemen-statemen SQL, dan statemen SQL dapat digunakan secara langsung di dalam *Macro* dan VBA *Module* untuk secara langsung memanipulasi tabel data dalam Access. Para pengguna dapat mencampurkan dan menggunakan kedua jenis bahasa tersebut (VBA dan *Macro*) untuk memprogram *form* dan logika dan juga untuk mengaplikasikan konsep berorientasi objek.

Tidak seperti sebuah sistem manajemen basis data relasional yang komplit, Microsoft JET Database Engine tidak memiliki fitur *trigger* dan *stored procedure*. Dimulai dari Microsoft Access 2000 yang menggunakan Microsoft Jet Database Engine versi 4.0, ada sebuah sintaksis yang mengizinkan pembuatan *query* dengan beberapa parameter, dengan sebuah cara seperti halnya sebuah *stored procedure*, meskipun prosesur tersebut dibatasi hanya untuk sebuah pernyataan tiap prosedurnya. Access juga mengizinkan *form* untuk mengandung kode yang dapat dieksekusi ketika terjadi sebuah perubahan terhadap tabel basis data, seperti halnya *trigger*, selama modifikasi dilakukan hanya dengan menggunakan *form* tersebut, dan merupakan sesuatu hal yang umum untuk menggunakan *query* yang akan diteruskan (*pass-through* dan teknik lainnya di dalam Access untuk menjalankan *stored procedure* di dalam RDBMS yang mendukungnya.

3. Pengembangan dengan Access

Access mengizinkan pengembangan yang relatif cepat karena semua tabel basis data, kueri, *form*, dan report disimpan di dalam berkas basis data miliknya (*.MDB). Untuk membuat *query*, Access menggunakan *Query Design Grid*, sebuah program berbasis grafis yang mengizinkan para penggunanya untuk membuat *query* tanpa harus mengetahui bahasa

pemrograman SQL. DI dalam *Query Design Grid*, para pengguna dapat memperlihatkan tabel basis data sumber dari *query*, dan memilih *field-field* mana yang hendak dikembalikan oleh proses dengan mengklik dan menyeretnya ke dalam *grid*. *Join* juga dapat dibuat dengan cara mengklik dan menyeret *field-field* dalam tabel ke dalam *field* dalam tabel lainnya. Access juga mengizinkan pengguna untuk melihat dan memanipulasi kode SQL jika memang diperlukan. [8]

2.2.7.2 Borland Delphi 7

Borland delphi merupakan suatu bahasa pemrograman yang memberikan berbagai fasilitas pembuatan aplikasi visual. Keunggulan bahasa pemrograman ini terletak pada produktivitas, kualitas, pengembangan perangkat lunak, kecepatan kompilasi, pola desain yang menarik serta diperkuat dengan pemrograman yang terstruktur. Keunggulan lain delphi adalah dapat dipergunakan untuk merancang program aplikasi yang memiliki tampilan seperti program aplikasi lain yang berbasis windows.

1. Komponen Delphi

Component palette terdiri dari beberapa komponen yang dapat dipilih yang digunakan untuk menangani beberapa tugas pemrograman. Komponen-komponen yang terletak pada bagian *component palette* sudah ditata dalam beberapa tab yang masing-masing menunjukan maksud dan fungsi. Masing-masing tab ditampilkan dalam konfigurasi *default* yang semua juga tergantung pada versi delphi yang digunakan.

Tabel berikut menunjukan daftar tab *default* dan beberapa komponen yang terdapat di dalamnya. [5]

Tabel 2.1 Tabel Komponen Delphi

Nama Tab	Isi
Standart	Kontrol-kontrol standar program windows dan menu
Additional	Kontrol-kontrol tambahan
Win32	Kontrol-kontrol umum windows 9x/NT 4.0
System	Komponen dan kontrol-kontrol dari sistem komputer termasuk *timer*, multimedia dan DDE
Data Access	Komponen-komponen non-visual yang digunakan untuk mengakses tabel-tabel *database*, *query*, dan *report*
Data Controls	Komponen-komponen visual, dan kontrol-kontrol data-*aware*
dbExpress	Komponen-komponen non-visual yang digunakan aplikasi untuk berhubungan dengan *database* dengan menggunakan *dbExpress*
DataSnap	Komponen dan kontrol-kontrol non-visual yang digunakan untuk membuat aplikasi *database* bertingkat (*multi-tiered*)
BDE	Komponen dan kontrol-kontrol non-visual yang digunakan untuk menghubungkan Informasi *database* dengan menggunakan *Borland Database Engine* (BDE)
ADO	Komponen dan kontrol-kontrol non-visual yang digunakan untuk menghubungkan Informasi *database* dengan

	menggunakan *ActiveX Data Object* (ADO)
InterBase	Komponen dan kontrol-kontrol non-visual yang digunakan untuk menghubungkan secara langsung *database interbase* tanpa menggunakan BDE ataupun ADO
InternetExpress	Komponen yang digunakan untuk membangun aplikasi *InternetExspress* yang simultan dengan *Web Server* dan klien dari suatu aplikasi *database* bertingkat

2. Fitur Pada Delphi 7

   Fitur baru dan perbaikan yang ada pada Borland Delphi 7 ini adalah :

   a. IDE (Interface Development Environtment)

      Lingkungan pengembangan aplikasi (IDE) Borland Delphi 7 telah mengalami perubahan dari versi sebelumnya. Diantaranya, terdapat *Compiler Message*, perubahan pada *Component Pallete*, *Code Insight* dan *Debugger*.

   b. Web

      Borland Delphi 7 menyediakan Intraweb buatan AtoZed Software, yang dapat digunakan untuk membuat aplikasi *web server* dengan sarana standar. Borland Delphi 7 juga mendukung pada Apache 2. Borland juga menghilangkan Win-CGI sebagai target aplikasi *Web server* dan *web service*. Fasilitas untuk *Web server* juga mengalami perbaikan - perbaikan .

c. COM

Sekarang ini dengan Delphi 7.0, dapat membuat CoClass wrapper bagi pengembangan – pengembangan .NET dengan cara menggunakan kotak dialog *Import Type Library*. Dengan adanya *resulting wrapper*, maka fitur interoperabilitas dari Microsoft's .NET *Framework* dapat digunakan.

d. Database

Pada Delphi 7.0 driver dbExpress telah diupdate bagi Informix SE, Oracle 9i, DB2 7.2, InterBase 6.5, dan MySQL 3.23.49. Driver baru bagi MSSQL 2000 juga tersedia. Disamping itu, beberapa hal baru dan perubahan juga dilakukan pada komponen *database*. Borland juga telah membuang SQL Links. Borland merekomendasikan pemakaian dbExpress bagi *database* SQL Server yang diakses di Delphi.

e. Component Library

Jika ditelusuri komponen librari Delphi 7.0, maka akan ditemukan komponen baru, unit baru, komponen yang berubah, komponen yang hilang dan komponen yang mendukung bagi tema Windows XP.

f. Runtime Library

Beberapa perubahan di Runtime Library antara lain ialah perubahan pada unit Classes, Math, StdConv,StrUtils, SysUtils, VarCmplx, dan Variants.

g. Compiler

Kompiler Delphi dcc32 sekarang ini support terhadap tiga warning kompiler tambahan, yaitu *Unsafe_Type*, *Unsafe_Code*, dan

Unsafe_Cast. Warnings tersebut defaultnya adalah *disabled,* tetapi dapat di-*enabled*. Fitur ini sangat membantu kita ketika akan mem-port kode ke lingkungan eksekusi terkendali di *platform* Microsoft's .NET.

h. Model Maker

Sarana baru yang disebut *Model Maker* dapat membantu memudahkan proses desain, konstruksi, dan pengelolaan *class* dan *interface*. *Model Maker* juga memiliki sarana untuk pembuatan diagram UML-style, yang dapat dipakai untuk membuat dan memodifikasi *source code project*. [3]