

BAB II

TINJAUAN PUSTAKA

II.1. Perancangan Sistem Informasi

Perancangan sistem informasi ,terjemahan dari *Information System Planning* (ISP),menceritakan bagaimana menerapkan pengetahuan tentang sistem informasi kedalam organisasi.Untuk dapat terus maju dan eksis bila organisasi berkembang sesuai dengan teknologi dan teori organisasi modern.Namun demikian hal ini tidak berarti bahwa sistem informasi dan teknologi informasi sebagai suatu hal yang kaku.Sistem informasi dapat dibentuk sesuai kebutuhan organisasi.Oleh karena itu untuk dapat menerapkan sistem yang efektif dan efisien diperlukan perancangan,pelaksanaan,pengaturan dan evaluasi sesuai keinginan dan nilai masing-masing organisasi.Orang tidak boleh sekedar mengadaptasi setiap sistem yang ditawarkan,akan tetapi juga tidak boleh menutup mata terhadap pengetahuan dan kesempatan yang ada diluar organisasi untuk mendapatkan sistem yang cocok.Sistem yang efektif dan efisien tidak lain untuk mendapatkan keunggulan dalam kompetisi.Semua orang dapat menggunakan sistem informasi dalam organisasi,akan tetapi faktor efisiensi setiap sistem adalah berbeda.

Ada beberapa tingkatan-tingkatan perubahan sistem baik yang besar maupun kecil yaitu:

- a. Tingkat I : Ide,mengetahui perlu adanya perubahan.
- b. Tingkat II : Design,merancang cara pemecahannya.

- c. Tingkat III : Pelaksanaan,menerapkan design kedalam sistem.
- d. Tingkat IV: Kontrol,memeriksa apakah tingkat pelaksanaan dijalankan sesuai dengan design.
- e. Tingkat V: Evaluasi,memeriksa apakah perubahan yang terjadi sesuai dengan tujuan semula.
- f. Tingkat VI: Tindak lanjut,melaksanakan perubahan sesuai dengan hasil evaluasi yang ada.(Tata Sutabri, S.Kom. ; 2004 : 39-40)

II.2. Pengertian Sistem

Sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen atau variabel-variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu. Teori sistem secara umum terutama menekankan pentingnya perhatian terhadap setiap bagian yang membentuk sebuah sistem. Teori sistem mengatakan bahwa setiap unsur pembentuk organisasi adalah penting dan harus mendapat perhatian yang utuh supaya manajer dapat bertindak lebih efektif.(Tata Sutabri, S.Kom. ; 2004 : 3)

Sebuah sistem mempunyai karakteristik atau sifat-sifat yang tertentu, yaitu mempunyai :

- a. Komponen – komponen (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan. Komponen – komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat – sifat dari sistem yang menjalankan suatu fungsi tertentu

dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar, yang disebut Supra sistem.

b. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem di pandang sebagai satu kesatuan yang tidak dapat dipisah – pisahkan.

c. Lingkungan Luar Sistem (*Environment*)

Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem. Lingkungan luar sistem ini dapat menguntungkan dan dapat juga merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi bagi sistem tersebut, yang demikian lingkungan luar tersebut harus dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus dikendalikan, karena kalau tidak maka akan mengganggu kelangsungan hidup sistem tersebut.

d. Penghubung Sistem (*Interface*)

Sebagai media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem interface. Penghubung ini memungkinkan sumber – sumber daya mengalir dari satu subsistem ke subsistem yang lain. Keluaran suatu subsistem akan menjadi masukan untuk subsistem yang lain dengan melewati penghubung. Dengan demikian terjadi suatu integrasi sistem yang membentuk satu kesatuan.

e. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam system disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Sebagai contoh, di dalam suatu unit sistem komputer, “program” adalah *maintenance input* yang digunakan untuk mengoperasikan komputer sementara “data” adalah *signal input* yang akan diolah menjadi informasi.

f. Keluaran Sistem (*Output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, yang mana informasi ini dapat digunakan sebagai masukan untuk mengambil keputusan atau hal – hal lain yang merupakan input bagi subsistem lainnya.

g. Pengolahan Sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran. Sebagai contoh, sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan – laporan yang dibutuhkan oleh pihak manajemen.

h. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai

sasaran atau tujuan yang telah direncanakan. (Tata Sutabri, S.Kom ; 2004 : 12 - 13)

II.3. Informasi

Informasi adalah sebuah istilah yang tidak tepat dala pemakainnya secara umum. Informasi dapat mengenai data mentah, data tersusun, kapasitas sebuah saluran komunikasi, dan lain sebagainya. Suatu sistem yang kurang mendapatkan informasi akan menjadi luruh, kerdil, dan akhirnya mati. Sistem pengolahan informasi data menjadi informasi atau tepatnya mengolah data dari bentuk tak berguna menjadi berguna bagi penerimanya. Nilai informasi berhubungan dengan keputusan. Bila tidak adanya pilahan atau keputusan, maka informasi menjadi tidak diperlukan. (Tata Sutabri S.Kom; 2005: 23 – 24)

Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya.

Informasi merupakan proses lebih lanjut dari data yang sudah memiliki nilai tambah. Informasi dapat dikelompokkan menjadi 3 bagian, yaitu :

- a. Informasi Strategis. Informasi ini digunakan untuk mengambil keputusan jangka panjang, mencakup informasi eksternal, rencana perluasan dan sebagainya.

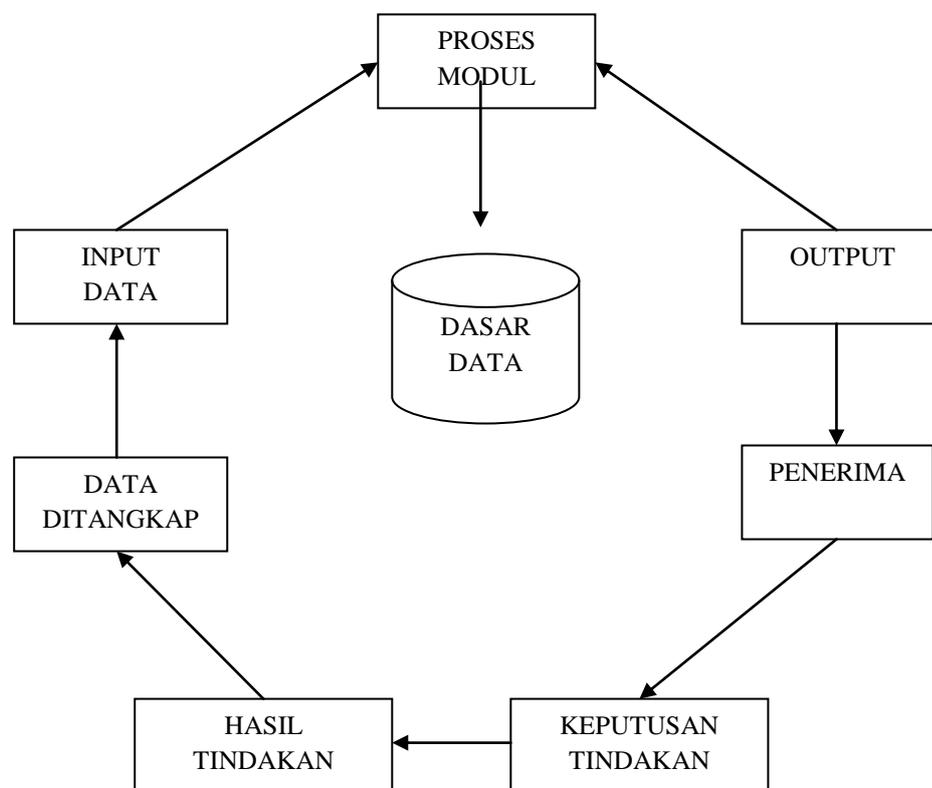
- b. Informasi Taktis. Informasi ini dibutuhkan untuk mengambil keputusan jangka menengah seperti informasi trend penjualan yang dapat dimanfaatkan untuk menyusun rencana penjualan.
- c. Informasi Teknis. Informasi ini dibutuhkan untuk keperluan operasional sehari-hari seperti informasi persediaan stock, retur penjualan dan laporan kas harian. (Tata Sutabri S.Kom; 2004 : 17-18)

II.3.1. Fungsi dan Siklus Informasi

Fungsi utama informasi adalah menambah pengetahuan atau mengurangi ketidakpastian pemakai informasi. Informasi yang disampaikan kepada pemakai kepada pemakai mungkin merupakan hasil data yang dimasukkan kedalam pengolahan. Informasi juga berfungsi untuk memberikan dasar guna melakukan seleksi. Informasi tidak mengarah kepada yang harus dilakukan, tetapi mengurangi keanekaragaman dan ketidakpastian sehingga dapat dihasilkan keputusan yang baik. Fungsi informasi yang penting lainnya adalah memberikan standar-standar, aturan-aturan ukuran dan aturan-aturan keputusan untuk penentuan dan penyebaran tanda-tanda kesalahan dan umpan balik guna mencapai tujuan kontrol.

Informasi dapat berasal dari pengamatan, percakapan dengan orang lain, rapat-rapat panitia, dari majalah, dari media surat kabar atau laporan pemerintah dan dari sistem informasi itu sendiri. Informasi diperoleh dari data yang ada, kemudian data tersebut diolah dengan menggunakan suatu proses tertentu.

Data diolah melalui suatu model informasi. Si penerima akan menerima informasi tersebut untuk membuat suatu keputusan dan melakukan tindakan yang akan mengakibatkan munculnya sejumlah data lagi. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model dan seterusnya sehingga membentuk sebuah Siklus Informasi. (Tata Sutabri S.Kom; 2004 : 19-21)



Gambar II.1 Siklus Informasi (*Information Cycle*)

Sumber : (Tata Sutabri S.Kom; 2004 : 21)

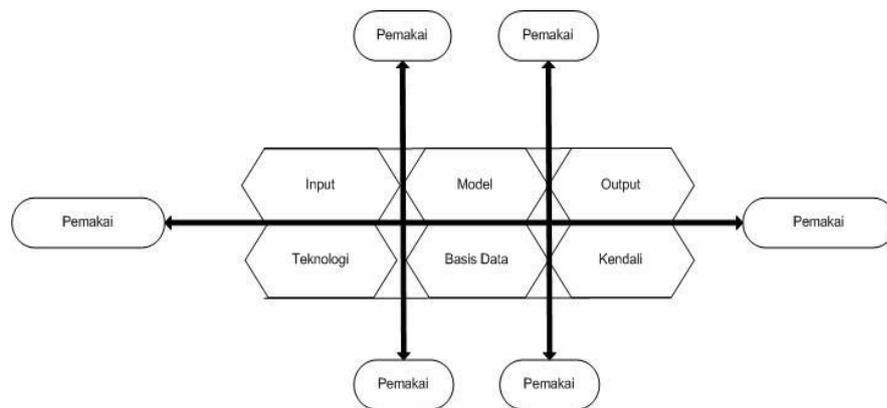
II.4. Sistem Informasi

Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan. (Tata Sutabri, S.Kom ; 2004 : 36)

Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok-blok bangunan (*building block*), yang terdiri:

- a. Blok masukan (*input block*), mewakili data yang masuk kedalam sistem informasi.
- b. Blok model (*model block*), terdiri dari komponen prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang sudah tersimpan di basis data.
- c. Blok keluaran (*output block*), produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna.
- d. Blok teknologi (*technology block*), digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirim keluaran dan membantu pengendalian dari sistem secara keseluruhan.
- e. Blok basis data (*database block*), kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lainnya, tersimpan didalam perangkat keras komputer dan perangkat keras lunak komputer yang digunakan untuk memanipulasinya.

- f. Blok kendali (*control block*), pengendalian perlu dirancang dan ditetapkan untuk menyakinkan bahwa hal-hal yang dapat merusak sistem dicegah dan bila terlanjur terjadi maka kesalahan-kesalahn dapat cepat diatasi. (Tata Sutabri S.Kom; 2004 : 36-37)



Gambar II.3 Komponen Sistem Informasi

Sumber: (Tata Sutabri S.Kom ; 2004 : 38)

II.5. Pengertian Sumber Daya Manusia

Sumber daya manusia merupakan salah satu sumber daya yang terdapat dalam organisasi, meliputi semua orang yang melakukan aktivitas. Secara umum, sumber daya yang terdapat dalam suatu organisasi bisa dikelompokkan atas dua macam yakni:

- a. Sumber daya manusia (*human resource*)
- b. Sumber daya Non-manusia (*non-human resource*) yang termasuk dalam kelompok sumber daya non-manusia ini antara lain modal, mesin, teknologi, bahan-bahan (material), dan lain-lain.

Dari keseluruhan sumber daya yang tersedia dalam suatu organisasi, baik organisasi publik maupun swasta, sumber daya manusia adalah yang paling penting dan sangat menentukan. Sumber daya manusia merupakan satu-satunya sumber daya yang memiliki akal, perasaan, keinginan, kemampuan, keterampilan, pengetahuan, dorongan, daya dan karya. Satu-satunya daya yang memiliki ratio, rasa, dan karsa. Semua potensi sumber daya manusia tersebut sangat berpengaruh terhadap upaya organisasi dalam pencapaian tujuannya. Betapa pun majunya teknologi, berkembangnya informasi, tersedianya modal dan memadainya bahan, namun jika tanpa sumber daya manusia maka akan sulit bagi organisasi untuk mencapai tujuannya. Betapapun bagusya perumusan tujuan dan rencana organisasi, agaknya akan hanya sia-sia belaka jika unsur sumber daya manusianya tidak diperhatikan, apalagi kalau ditelantarkan.

Berbagai istilah yang dipakai untuk menunjukkan manajemen sumber daya manusia antara lain: manajemen sumber daya manusia, manajemen sumber daya insani, manajemen personalia, manajemen kepegawaian, manajemen perburuhan, manajemen tenaga kerja, administrasi personalia (kepegawaian), dan hubungan perindustrial.

Manajemen sumber daya manusia sebenarnya merupakan suatu gerakan pengakuan terhadap pentingnya unsur manusia sebagai sumber daya yang cukup potensial, yang perlu dikembangkan sedemikian rupa sehingga mampu memberikan kontribusi yang maksimal bagi organisasi dan bagi pengembangan dirinya. Istilah MSDM kini semakin populer, menggantikan istilah personalia. Meskipun demikian istilah personalia ini masih tetap dipergunakan

dalam banyak organisasi untuk memahami departemen yang menangani kegiatan-kegiatan seperti rekrutmen, seleksi, pemberian kompensasi dan pelatihan karyawan. MSDM akhir-akhir ini merupakan istilah yang banyak dipergunakan dalam berbagai forum diskusi, seminar, lokakarya dan sejenisnya. Pergantian istilah dari manajemen personalia dengan manajemen SDM dianggap sebagai suatu gerakan yang mencerminkan pengakuan adanya peranan vital dan semakin pentingnya sumber daya manusia dalam suatu organisasi, adanya tantangan-tantangan yang semakin besar dalam pengelolaan SDM secara efektif, serta terjadinya pertumbuhan ilmu pengetahuan dan profesionalisme dibidang manajemen sumber daya manusia. (Dr. Faustino Cordoso Gomes, Msi; 2003: 1-3)

II.5.1. Ruang lingkup Sumber Daya Manusia

Tugas SDM berkisar pada upaya mengelolah unsur manusia dengan segala potensi yang dimilikinya seefektif mungkin sehingga dapat diperoleh sumber daya manusia yang puas (*satisfied*) dan memuaskan (*satisfactory*) bagi organisasi. SDM merupakan bagian ddari manajemen umumnya yang memfokuskan diri pada sumber daya manusia.

Ruang lingkup SDM yang berhubungan dengan organisasi adalah:

- a. Rancangan organisasi
- b. Staffing
- c. Sistem reward, tunjangan-tunjangan, dan pematuhan
- d. Manajemen performansi
- e. Pengembangan pekerja dan organisasi

f. Komunikasi dan hubungan masyarakat.

II.5.2. Meningkatnya Otomasi Dan Pengembangan MSDM

Kecenderungan ini sekedar mencerminkan dan memperluas suatu kecenderungan yang terdapat diseluruh dunia organisasi: pemakaian komputer yang meningkat. Lebih banyak program manajemen sumber daya manusia yang memakai sistem informasi sumber daya manusia yang terpadu dalam proses pembuatan keputusan.

Program-program lunak (*software*) dikembangkan, membantu program perencanaan, monitor, dan evaluasi sumber daya manusia dari program-program *software* tunggal yang memusatkan pada suatu program seperti pengembangan manajemen atau rekrutmen, ke program-program sistem informasi manajemen sumber daya manusia dengan database yang lebih banyak pada komputer-komputer yang memungkinkan penggunaan dan integrasi data. Dalam kasus yang terakhir, perusahaan-perusahaan memadukan laporan kepegawaian individual dengan data dari berbagai macam sumber. Data-data itu bisa dipergunakan untuk merencanakan program-program pengembangan manajemen, menyediakan (*feed*) suksesi-suksesi manajemen dan sistem-sistem perencanaan tenaga kerja, untuk melayani tujuan-tujuan lainnya juga. (Dr. Faustino Cordoso Gomes, Msi; 2003:17)

II.6. Visual Basic 2010

Visual Basic 2010 merupakan salah satu bagian dari produk pemrograman terbaru yang dikeluarkan oleh Microsoft, yaitu Microsoft Visual Studio 2010. Sebagai produk lingkungan pengembangan terintegrasi atau IDE andalan yang dikeluarkan oleh Microsoft, Visual Studio 2010 menambahkan perbaikan-perbaikan fitur dan fitur baru yang lebih lengkap dibandingkan versi Visual Studio sebelumnya, yaitu Microsoft Visual Studio 2008.

Visual Studio merupakan produk pemrograman andalan dari Microsoft Corporation, yang didalamnya berisi beberapa IDE pemrograman seperti Visual Basic, Visual C++, Visual Web Developer, Visual C#, dan Visual F#. Semua IDE pemrograman tersebut sudah mendukung penuh implementasi .Net Framework terbaru, yaitu .Net Framework 4.0 yang merupakan pengembangan dari .Net Framework 3.5. Adapun database standar yang disertakan adalah Microsoft SQL Server 2008 express. (Sumber: wahana Komputeri; 2010:2)

II.6.1. Instalasi Visual Basic 2010

Supaya aplikasi Visual Basic 2010 dapat berjalan dengan optimal, perlu dilakukan pengecekan apakah perangkat yang akan digunakan nantinya sudah memenuhi kriteria yang ditetapkan oleh Microsoft selaku pengembang aplikasi Visual Studio 2010. Persyaratan awal yang harus dipenuhi sebelum instalasi dapat dilakukan adalah bahwa komputer yang nantinya akan diinstal Visual Studio minimal harus terdapat sistem operasi sebagai berikut:

- a. Windows 7

- b. Windows Vista
- c. Windows XP Service Pack 3
- d. Windows 2008
- e. Windows Server 2003 Service Pack 2

Selain persyaratan sistem operasi minimal yang bisa digunakan,terdapat pula persyaratan awal library aplikasi atau *prerequisites* yang harus dipenuhi,diantaranya:

- a. Visual C Runtime 9.0*
- b. Visual C Runtime10.0*
- c. NET Framework 4*
- d. Visual Studio Macro Tools*
- e. TFS Object Model*

(*Wahana Komputer;2010:2*)

II.7. Entitas Relation Diagram (ERD)

Entitas Relation Diagram berguna untuk memodelkan sistem yang nantinya basis datanya akan dikembangkan.Model ini juga membantu perancangan/analisis sistem pada saat melakukan analisis dan perancangan basis data karena model ini dapat menunjukkan macam data yang dibutuhkan dan kerelasian antar data didalamnya.Bagi pengguna, model ini sangat membantu dalam hal pemahaman model sistem dan rancangan basis data yang akan dikembangkan oleh perancang/analisis sistem. (Andi;2011:91)

II.7.1. Jenis Kelelasan Antar Entitas (*Relationship*)

a. Kerelasiaan 1-1/satu ke satu (*one-to-one*)

Kerelasiaan jenis ini terjadi jika kejadian atau transaksi di antara dua entitas yang berhubungan hanya memungkinkan terjadi sebuah kejadian atau transaksi pada kedua entitas. Secara lebih teknis, jika nilai yang digunakan sebagai penghubung pada entitas pertama hanya dimungkinkan muncul satu kali saja pada entitas kedua yang saling berhubungan. Sebagai contoh, satu orang mahasiswa (atribut-atributnya tersimpan dalam entitas mahasiswa) hanya dimungkinkan mempunyai satu orang wali mahasiswa (atribut-atributnya tersimpan dalam entitas wali_mahasiswa).

b. Kerelasiaan jenis n-ke-1/banyak ke satu (*many to one*) atau 1-ke-n/satu ke banyak (*one to many*)

Kerelasiaan jenis ini terjadi jika kejadian atau transaksi diantara dua entitas yang berhubungan hanya memungkinkan terjadi satu kali dalam entitas pertama dan dapat terjadi lebih dari satu kali kejadian atau transaksi pada entitas kedua. Secara lebih teknis, jika nilai yang lain digunakan sebagai penghubung pada entitas pertama dimungkinkan muncul lebih dari satu kali pada entitas kedua yang berhubungan. Sebagai contoh, lebih dari satu mahasiswa (atribut-atributnya tersimpan dalam entitas mahasiswa) dapat memilih hanya satu buah program studi (atribut-atributnya tersimpan dalam entitas program_studi). Kondisi ini disebut jenis kerelasiaan n-ke-1.

c. Kerelasiaan jenis n-ke-n/ banyak-banyak (*many-to-many*)

Kerelasiaan jenis ini terjadi jika kejadian atau transaksi diantara dua entitas yang berhubungan memungkinkan terjadi lebih dari satu kali dalam entitas pertama dan entitas kedua. Secara lebih teknis, jika nilai yang digunakan sebagai penghubung pada entitas pertama dimungkinkan muncul lebih dari satu kali, baik pada entitas pertama maupun pada entitas kedua yang saling berhubungan, dan sebaliknya. Sebagai contoh, lebih dari satu mahasiswa (atribut-atributnya tersimpan pada entitas mahasiswa) dapat memilih lebih dari satu mata kuliah (atribut-atributnya tersimpan dalam entitas mata_kuliah). Kondisi ini disebut juga kerelasiaan n-ke-n. (Andi;2011:102-103)

II.8. Kamus Data

Kamus Data (KD) atau *data dictionary* (DD) seperti halnya dengan kamus bahasa berfungsi menjelaskan lebih detail suatu kata maupun kalimat, kamus data yang digunakan dalam analisis struktur dan desain sistem informasi juga merupakan suatu katalog yang menjelaskan lebih detail tentang data flow diagram yang mencakup proses, data flow data store. Kamus data dapat digunakan pada metodologi berorientasi data dengan menjelaskan lebih detail lagi hubungan entitas, seperti atribut-atribut suatu entitas. Pada metodologi objek, kamus data dapat menjelaskan lebih detail atribut maupun metode atau service suatu objek. Apabila didefinisikan, kamus data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan kamus data

sistem analisis dapat mendefinisikan data yang mengalir pada sistem dengan lengkap. (Tata Sutabri S.Kom; 2004 : 170)

II.8.1. Isi Kamus Data

Kamus Data harus dapat mencerminkan keterangan yang jelas tentang data yang dicatatnya, untuk itu maka Kamus Data harus memuat hal-hal berikut :

a. Arus Data

Arus data menunjukkan dari mana data mengalir dan kemana data akan menuju. keterangan arus data ini perlu dicatat di KD untuk memudahkan mencari arus data flow diagram(DFD).

b. Nama arus data

Karena Kamus Data dibuat berdasarkan arus data yang mengalir di DFD, maka nama dari arus data juga harus dicatat di KD, sehingga yang membaca DFD dan memerlukan penjelasan lebih lanjut tentang suatu arus data tertentu di DFD dapat mencarinya dengan mudah di KD.

c. Tipe data

Telah diketahui bahwa arus data dapat mengalir dari hasil suatu proses keproses yang lainnya. Data yang mengalir ini biasanya dalam bentuk laporan serta dokumen hasil cetakan komputer. Dengan demikian bentuk dari data yang mengalir dapat berupa dokumen dasar atau formulir, dokumen hasil cetakan komputer, laporan tercetak, tampilan layar dimonitor, variabel, parameter dan field-field. Bentuk data seperti ini perlu dicatat di kamus data.

d. Struktur data

Struktur data menunjukkan arus data yang dicatat pada kamus data yang terdiri dari item-item atau elemen-elemen data.

e. Alias

Alias atau nama lain dari data juga harus dituliskan bila. Alias perlu ditulis karena data yang sama mempunyai nama yang berbeda untuk orang atau departemen lainnya.

f. Volume

Volume yang perlu dicatat didalam kamus data adalah volume rata-rata dan volume puncak arus data. Volume rata-rata menunjukkan banyaknya arus data yang mengalir dalam satu periode tertentu sementara volume puncak menunjukkan volume yang terbanyak.

g. Periode

Periode ini menunjukkan kapan terjadi arus data. Periode perlu dicatat di kamus data karena dapat digunakan untuk mengidentifikasi kapan input data harus dimasukkan kedalam sistem, kapan proses program harus dilakukan kapan laporan –laporan harus dihasilkan.

h. Penjelasan

Untuk lebih memperjelas makna dari arus data yang dicatat di kamus data, maka bagian penjelasan dapat diisi dengan keterangan-keterangan tentang arus data tersebut. (Tata Sutabri S.Kom; 2004 : 70-71)

II.9. Normalisasi

Normalisasi diartikan sebagai suatu teknik yang menstrukturkan/mendekomposisi data dalam cara-cara tertentu untuk mencegah timbulnya permasalahan pengolahan data dalam basis data. Permasalahan yang dimaksud adalah berkaitan dengan penyimpangan-penyimpangan (*anomallisme*) yang terjadi akibat adanya kerangkapan data dalam relasi dan in-efisiensi pengolahan (Martin,1975).

Proses normalisasi menghasilkan relasi yang optimal,yaitu(Martin,1975):

- a. Memiliki struktur *record* yang konsisten secara logik.
- b. Memiliki struktur *record* yang mudah untuk dimengerti.
- c. Memiliki struktur *record* yang sederhana dalam pemeliharaan.
- d. Memiliki struktur *record* yang mudah ditampilkan kembali untuk memenuhi kebutuhan pengguna.
- e. Minimalisasi kerangkapan data guna meningkatkan kinerja sistem

(Andi;2011:174-175)

II.9.1 Level Normalisasi

Berbeda dengan kebiasaan ,yaitu teori biasanya terlebih dahulu baru kemudian diterapkan dalam praktik, teori normalisasi dibangun menurut konsep level normalisasi.Level normalisasi atau sering disebut sebagai bentuk normal suatu relasi dijelaskan berdasarkan kriteria tertentu pada bentuk normal.Bentuk normal yang kita kenal hingga saat ini meliputi bentuk 1NF, 2NF, 3NF, BCNF, 4NF, 5NF, DKNF, dan RUNF.Secara berturut-turut masing-masing level normal tersebut dibahas berikut ini, dimulai dari bentuk tidak normal,yaitu;

1. Relasi bentuk tidak normal (*un normalized form/UNF*)

Relasi-relasi yang dirancang tanpa mengindahkan batasan dalam definisi basis data dan karakteristik RDBM menghasilkan relasi UNF. Bentuk ini harus dihindari dalam perancangan relasi dalam basis data. Relasi UNF mempunyai kriteria yaitu ;

- a. jika relasi mempunyai bentuk *non flat file* (dapat terjadi akibat data disimpan sesuai dengan kedatangannya, tidak memiliki struktur tertentu, terjadi duplikasi atau tidak lengkap,
- b. jika relasi memuat set atribut berulang (*non single value*),
- c. jika relasi memuat atribut *non atomic value*.

2. Relasi bentuk normal pertama (*first norm form/1NF*)

Relasi disebut sebagai 1NF jika memenuhi kriteria sebagai berikut;

- a. Jika seluruh atribut dalam relasi bernilai atomik (*atomic value*)
- b. Jika seluruh atribut dalam relasi bernilai tunggal (*single value*)
- c. Jika relasi tidak memuat set atribut berulang
- d. Jika semua *record* mempunyai jumlah atribut yang sama.

Permasalahan dalam 1NF adalah sebagai berikut;

- a. Tidak dapat menyisipkan informasi parsial
- b. Terhapusnya informasi ketika menghapus sebuah *record*
- c. Pembaruan atribut nonkunci mengakibatkan sejumlah *record* harus diperbaharui.

Mengubah relasi UNF menjadi bentuk 1NF dapat dilakukan dengan cara sebagai berikut;

- a. Melengkapi nilai-nilai dalam atribut
 - b. Mengubah struktur relasi.
3. Bentuk normal kedua (*second norm form/2NF*)

Relasi disebut sebagai 2NF jika memenuhi kireteria sebagai berikut;

- a. Jika memenuhi kriteria 1NF
- b. Jika memenuhi atribut nonkunci FD pada PK

Permasalahan dalam 2NF adalah sebagai berikut;

- a. Kerangkapan data (*data redundancy*)
- b. Pembaruan yang tidak benar dapat menimbulkan inkonsistensi data (*data inconsistency*)
- c. Proses pembaruan data efesien
- d. Penyimpangan data saat penyisipan, penghapusan, dan pembaruan.

Kriteria tersebut mengindikasikan bahwa diantara atribut dalam 2NF masih mungkin mengalami TDF.Selain itu, relasi 2NF menuntut telah didefenisikan atribut PK dalam relasi.Mengubah relasi 1NF menjadi bentuk 2NF dapat dilakukan dengan mengubah struktur relasi dengan cara;

- a. Identifikasikan FD relasi 1NF (jika perlu gambarkan diagram ketergantungan datanya
- b. Berdasarkan informasi tersebut, dekomposisi relasi 1NF menjadi relasi-relasi baru sesuai FD-nya.Jika menggunakan diagram maka simpul-simpul yang berada pada puncak diagram ketergantungan data bertindak sebagai PK pada relasi baru.

4. Bentuk norma ketiga(*third norm form/3NF*)

Suatu relasi disebut sebagai 3NF jika memenuhi kriteria sebagai berikut;

- a. Jika memenuhi kriteria 2NF
- b. Jika setiap atribut nonkunci tidak TDF (*non transitive dependency*)

Permasalahan dalam 3NF adalah keberadaan penentu yang tidak merupakan bagian dari PK menghasilkan duplikasi rinci data pada atribut yang berfungsi sebagai FK (duplikasi berbeda dengan kerangkapan data). Mengubah relasi 2NF menjadi bentuk 3NF dapat dilakukan dengan mengubah struktur relasi dengan cara;

- a. Identifikasikan TDF relasi 2NF (jika perlu gambarkan diagram ketergantungan datanya)
- b. Berdasarkan informasi tersebut, dekomposisikan relasi 2NF menjadi relasi-relasi baru sesuai TDF-nya. Jika menggunakan diagram maka simpul-simpul yang berada pada puncak diagram ketergantungan data bertindak sebagai PK pada relasi baru. Misalnya terhadap relasi R dengan sifat berikut;

a. $R=(A,B,C)$ dengan $PK = A$

b. $FD : R.B \rightarrow R.C$

Maka relasi R perlu dikomposisikan menjadi relasi-relasi R1 dan R2 yaitu;

a. $R1 = (B,C)$

b. $R2 = (A,B)$, $FK : B$ references $R1$

5. Bentuk normal Boyce-Codd (*Boyce-Codd norm form/BCNF*)

Bentuk normal BCNF dikemukakan oleh R.F. Boyce dan E.F. Codd. Suatu relasi tersebut sebagai BCNF jika memenuhi kriteria sebagai berikut;

- a. Jika memenuhi kriteria 3NF
 - b. Jika semua atribut penentu (*determinan*) merupakan CK
6. Bentuk normal keempat (*forth norm form/4NF*)
- Relasi disebut 4NF jika memenuhi kriteria sebagai berikut;
- a. Jika memenuhi kriteria BCNF
 - b. Jika setiap atribut didalamnya tidak mengalami ketergantungan pada banyak nilai.atau dengan kalimat lain, bahwa semua atribut yang mengalami ketergantungan secara fungsional (*functionsllly dependency*).
7. Bentuk normal kelima (*fifth norm form/5NF*)
- Suatu relasi memenuhi kriteria 5NF jika kerelasian antar data dalam relasi tersebut tidak dapat direkonstruksi dari struktur relasi yang sederhana.
8. Bentuk normal kunci domain (*domain key norm form/DKNF*)
- Suatu relasi disebut sebagai DKNF jika setiap batasan dapat disimpulkan secara sederhana dengan mengetahui sekumpulan nama atribut dan domainnya selama menggunakan sekumpulan atribut pada kuncinya.Bentuk DKNF ini dikemukakan oleh R. Fagin pada 1981 dan bersifat sangat spesifik, artinya tidak semua relasi dapat mencapai level ini. (Andi;2011:176-179)

II.10.Pengenalan SQL Server

Microsoft SQL Server merupakan salah satu produk RDBMS.RDBMS memiliki kepanjangan *Relational Database Management System*.Merupakan salah satu produk andalan yang dibuat oleh Microsoft yang berfungsi sebagai reational database.Entah mengapa nama Microsoft SQL Server tersebut,lebih mengenal

dengan sebutan SQL Server. Mungkin memang karena penyebutannya yang sudah tak asing lagi ditelinga kita.

Microsoft SQL Server mendukung SQL sebagai bahasa pemroses query. Seperti yang kita ketahui, SQL merupakan bahasa standar internasional untuk proses query database dan SQL ini sudah banyak sekali digunakan pada hampir semua aplikasi, baik itu e-commerce, pendidikan, organisasi, pemerintah, atau bahkan personal sekalipun. Sehingga dengan demikian, SQL Server patut kita pertimbangkan sebagai database program kita. (Sumber: Agus Saputra; 2013:11)

II.10.1. Kelebihan Dan Kekurangan Menggunakan SQL Server

Suatu software atau bahasa pemrograman pasti memiliki kelebihan dan kekurangan masing-masing, tak terkecuali database (SQL Server). SQL Server memiliki beberapa kelebihan dan kekurangan. Berikut beberapa kelebihan dan kekurangan yang dapat penulis ambil.

Kelebihan SQL Server yaitu:

- a. Mendukung SP (*Store Procedure*). Dengan *Store Procedure*, kita akan memperoleh manfaat, terutama untuk *security* (mencegah inputan ganda). Untuk lebih jelasnya dapat anda baca buku penulis “**Panduan Praktis Menguasai Database Server MySQL**”, terbitan PT Elex Media Komputindo.
- b. Mendukung Trigger, merupakan sekumpulan perintah atau sintaks yang akan secara otomatis dijalankan jika terjadi operasi tertentu dalam suatu tabel ataupun view. Biasanya Trigger digunakan untuk memanggil satu atau beberapa perintah SQL secara otomatis sebelum dan sesudah terjadi

proses manipulasi (*create, read, update, delete*). Misalnya kita ingin menyimpan data back-up ke tabel history atau log sebelum data tersebut dihapus dari sebuah tabel.

- c. Mendukung adanya kursor, untuk melakukan mapping record terhadap tabel yang kita proses.
- d. Mendukung adanya *function* dan beberapa transact SQL yang lain. Dengan kata lain, kita dapat menggabungkan keempat fitur yang dimiliki oleh SQL Server.

Kekurangan SQL Server yaitu:

- a. Merupakan aplikasi berbayar yang mana bila kita menggunakan aplikasi tersebut secara resmi, kita harus mengeluarkan dana yang cukup besar untuk software tersebut.
- b. Mempunyai banyak BUG, sehingga jika kita tidak berhati-hati bisa-bisa dieksploitasi dari luar. Contoh, file databasenya, bisa dengan mudah di-attach ataupun di-remove dari sistem SQL Server. (Agus Saputra; 2013: 13-14)

II.11. UML (*Unified Modelling Language*)

UML (*Unified Modelling Language*) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi objek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif

untuk berbagi (sharing) dan mengkomunikasikan rancangan mereka dengan yang lain.

UML merupakan kesatuan dari bahasa pemodelan yang dikembangkan oleh *Booch*, *Object Modeling Technique* (OMT) dan *Object Oriented Software Engineering* (OOSE). Metode Booch dari Grady Booch sangat terkenal dengan nama metode *Design Object Oriented*. Metode ini menjadikan proses analisis dan design kedalam empat tahapan iterative, yaitu: identifikasi kelas-kelas dan objek-objek, identifikasi semantik dari hubungan objek dan kelas tersebut, perincian interface dan implementasi. Keunggulan metode Booch adalah pada detil dan kayanya dengan notasi dan elemen. Pemodelan OMT yang dikembangkan oleh Rumbaugh didasarkan pada analisis terstruktur dan pemodelan entity-relationship. Tahapan utama dalam metodologi ini adalah analisis, design sistem, design objek dan implementasi.

Dengan menggunakan UML, metode Booch, OMT, OSE digabungkan dengan membuang elemen-elemen yang tidak praktis ditambah elemen-elemen dari metode lain yang lebih efektif dan elemen-elemen baru yang belum ada pada metode terdahulu sehingga UML lebih ekspresif dan seragam daripada metode lainnya. (Munawar,2005:17-18)

II.11.1. *Class Diagram*

Dalam notasi UML, *Class* digambarkan dalam bentuk kotak. Nama *Class* menggunakan huruf besar diawal kalimatnya dan diletakkan diatas kotak. Bila *Class* mempunyai nama dari 2 suku kata atau lebih, maka semua suku kata digabungkan tanpa spasi dengan huruf awal tiap suku kata menggunakan huruf besar. (Munawar;2005:35).



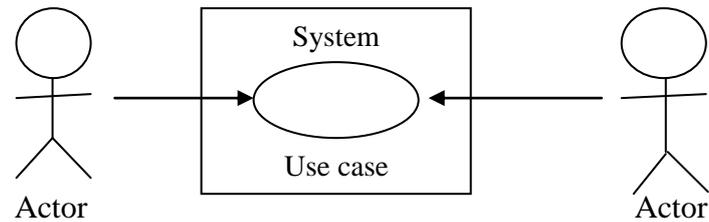
Gambar : II.2 Notasi *Class* di UML

(Sumber: Munawar;2005:35)

II.11.2. *Use Case*

Use Case adalah *deskripsi* fungsi dari sebuah sistem dari perspektif pengguna. *Use case* bekerja dengan cara mendeskripsikan tipikal interaksi antar *user* (pengguna) sebuah system dengan systemnya sendiri melalui sebuah cerita bagaimana sebuah system dipakai. Dalam pembicaraan tentang *use case*, pengguna biasanya disebut dengan *actor*. *Actor* adalah sebuah peran yang bisa dimainkan oleh pengguna dalam interaksinya dengan system.

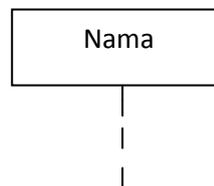
Model *use case* adalah bagian dari model *requirement* (Jacobson et all, 1992). Termasuk disini adalah problem domain objek model dan penjelasan tentang *user interface*. *Use case* memberikan spesifikasi fungsi-fungsi yang ditawarkan oleh sistem dari *perspektif user*.(Munawar.2005:63)



Gambar : II.3. Use Case Model
(Sumber: Munawar.2005:64)

II.11.3. Sequence Diagram

Sequence diagram digunakan untuk menggambarkan perilaku sebuah skenario. Komponen utama *sequence diagram* terdiri atas objek yang dituliskan dengan kotak segiempat bernama. *Message* diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan *progress vertical*. (Munawar.2005:87)



Gambar : II.4. Participant pada sebuah sequence diagram
(Sumber: Munawar.2005:87)

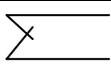
II.11.4. Activity Diagram

Activity Diagram adalah teknik untuk mendeskripsikan logika *procedural*, proses bisnis dan aliran kerja dalam banyak kasus. *Activity Diagram* mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *Activity Diagram* bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa. (Munawar.2005:109).

II.11.4.1. Simbologi

Berikut adalah symbol-simbol yang sering digunakan pada saat pembuatan *Activity Diagram*.

Tabel : II.1. Simbol-Simbol yang sering dipakai pada *activity diagram*.

Simbol	Keterangan
	Titik awal
	Titik akhir
	Activity
	Pilihan untuk pengambilan keputusan
	Fork : Digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan parallel menjadi satu.
	Rake : Menunjukkan adanya dekomposisi.
	Tanda Waktu
	Tanda Pengiriman
	Tanda Penerimaan
	Aliran Akhir (Flow Final)

(Sumber : Munawar.2005:109)