

BAB III

LANDASAN TEORI

Landasan teori digunakan untuk menyelesaikan masalah secara sistematis. Pada bab ini akan membahas landasan teori yang meliputi landasan teori mengenai hal-hal dari permasalahan yang ada dan landasan teori yang membahas tentang ilmu yang terkait dalam permasalahan tersebut.

3.1 Konsep Dasar Sistem Informasi

Konsep dasar dari Sistem Informasi terbagi atas tiga pengertian. Yang pertama adalah sistem, informasi dan sistem informasi itu sendiri.

3.1.1 Sistem

Konsep dasar tentang sistem lebih menekankan pada pemeriksaan terhadap seluruh bagian sistem, dan seringkali seorang analis terlalu memusatkan perhatian hanya pada satu komponen sistem, yang berarti dia telah mengambil tindakan yang mungkin tidak efektif, karena beberapa komponen yang penting diabaikan. Suatu sistem memiliki beberapa komponen, diantaranya pekerjaan, aktivitas, misi atau unsur-unsur sistem yang dibentuk untuk mewujudkan tujuan, untuk komponen misi atau tujuan, seringkali sukar untuk dilihat sehingga manajemen suatu sistem mengarahkan aktivitas-aktivitas pada perencanaan dan pengendalian yaitu berupa feedback (umpan balik).

“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.” (Jerry Fitzgerald : Hal 5: 1960).

Pendekatan sistem yang lebih menekankan pada elemen atau komponennya mendefinisikan sistem sebagai berikut :

“Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu.” (Jogiyanto H: Hal 2: 1999).

Pendekatan sistem yang merupakan kumpulan dari elemen-elemen atau komponen merupakan definisi yang lebih luas. Definisi ini lebih banyak diterima, Karena kenyataannya suatu sistem dapat terdiri dari beberapa subsistem atau

sistem-sistem bagian. Komponen-komponen dalam suatu sistem tidak dapat berdiri sendiri. Melainkan saling berinteraksi atau saling berhubungan membentuk suatu kesatuan sehingga tujuan atau sasaran dari sistem tersebut tercapai. Model umum sebuah sistem terdiri dari masukan, pengolah dan keluaran.

3.1.2 Informasi

Data adalah fakta-fakta atau kejadian-kejadian yang dapat berupa angka-angka atau kode-kode tertentu. Data masih belum mempunyai arti bagi penggunanya. Untuk dapat mempunyai arti data diolah sedemikian rupa sehingga dapat digunakan oleh penggunanya.

Kegunaan informasi adalah untuk mengurangi ketidakpastian di dalam proses pengambilan keputusan tentang suatu keadaan. Informasi yang digunakan di dalam suatu sistem informasi umumnya digunakan untuk beberapa kegunaan. Informasi digunakan tidak hanya digunakan oleh satu orang pihak dalam suatu organisasi. Nilai sebuah informasi ditentukan dari dua hal yaitu manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya untuk mendapatkan informasi tersebut.

Informasi dalam sebuah organisasi sangat penting peranannya. Suatu sistem yang kurang mendapatkan informasi akan menjadi lemah dan akhirnya berakhir. Informasi itu sendiri dapat didefinisikan sebagai berikut :

“Informasi merupakan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya”.

Sumber dari informasi adalah data. Data merupakan bentuk jamak dari bentuk tunggal data atau data item. Data adalah kenyataan yang menggambarkan sesuatu yang terjadi pada saat tertentu.

3.1.3 Sistem Informasi

Informasi dapat diperoleh dari sistem informasi (Information System) atau disebut juga dengan processing system atau information generating system. Dari uraian diatas maka sistem informasi dapat diartikan sebagai berikut :

1. Suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan tertentu yaitu menyajikan informasi.
2. Sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan atau untuk mengendalikan organisasi.

Sistem informasi sendiri memiliki sejumlah komponen tertentu, yang terdiri dari beberapa komponen yang berbeda yaitu, manusia, data, hardware, dan software. Sebagai suatu sistem, setiap komponen tersebut berinteraksi satu dengan lainnya membentuk satu kesatuan untuk mencapai sarannya.

3.2 Analisis Sistem

Menurut Jogiyanto (1990:129) analisa sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan - permasalahan, kesempatan - kesempatan, hambatan – hambatan yang terjadi dan kebutuhan - kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan – perbaikannya. Tahap analisis dilakukan setelah tahap perencanaan sistem dan sebelum tahap desain sistem.

3.3 Desain Sistem

Setelah tahap analisa sistem selesai dilakukan, maka analisis sistem telah mendapatkan gambaran yang jelas apa yang harus dikerjakan. Kemudian memikirkan bagaimana membentuk sistem tersebut. Menurut Jogiyanto (1990:197) desain sistem dapat diartikan sebagai berikut :

- a. Tahap setelah analisis dari siklus pengembangan sistem.
- b. Pendefinisian dari kebutuhan – kebutuhan fungsional.
- c. Persiapan untuk rancang bangun implementasi.
- d. Menggambarkan bagaimana suatu sistem dibentuk.
- e. Berupa gambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi.

- f. Menyangkut konfigurasi dari komponen – komponen perangkat lunak dan perangkat keras dari suatu sistem.

3.4 Teori Yang Terkait Dengan Penyelesaian Masalah

3.4.1 Microsoft Visual Basic. Net 2005


Visual Basic. Net 2005 merupakan bahasa yang benar – benar berorientasi objek dengan mendukung empat pilar utama dari *Object Oriented Programming* yaitu *Abstraction*, *Inheritance*, *Polymorphism* dan *Encapsulation*.




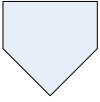

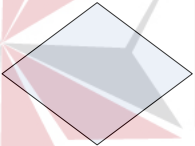

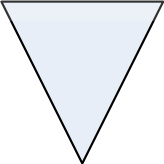
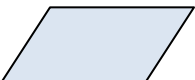
Dengan tersedianya berbagai komponen atau objek yang memiliki wadah inputan dalam suatu *form* membuat aplikasi lebih baik dan *user friendly*, sehingga aplikasi dapat dimanfaatkan semaksimal mungkin guna menghasilkan informasi yang cepat dan akurat. Selain itu, bahasa program ini memiliki kompatibilitas dengan berbagai macam aplikasi pendukung lainnya, baik itu yang mengarah kepada program yang berbasis *network*, *website* dan aplikasi lainnya. (Yuswanto, 2008)

3.4.2 System Flow

Bagan alir sistem (*system flowchart*) merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem. Bagan alir sistem menunjukkan apa yang dikerjakan di sistem. Bagan alir sistem digambar dengan menggunakan simbol-simbol yang tampak sebagai berikut ini.

Tabel 3.1 Simbol – simbol System Flow

NO	SIMBOL	NAMA SIMBOL	FUNGSI
1.		<i>Document</i>	Untuk menunjukkan dokumen input dan output baik untuk proses manual, mekanik, atau komputer.

2.		<i>Process Computer</i>	Menunjukkan kegiatan dari operasi program komputer.
3.		<i>Database</i>	Untuk menyimpan data.
4.		<i>On Page Connector</i>	Menunjukkan hubungan di halaman yang sama.
5.		<i>Off Page Connector</i>	Menunjukkan hubungan di halaman lain.
6.		<i>Terminator</i>	Menandakan awal /akhir dari suatu sistem.
7.		<i>Decision</i>	Menggambarkan logika keputusan dengan nilai <i>true</i> atau <i>false</i> .
8.		<i>Manual input</i>	Untuk menunjukkan pekerjaan yang dilakukan secara manual.
9.		Arsip	Untuk menunjukkan file non komputer yang diarsip urutan angka .
10.		Catatan	Menunjukkan data catatan

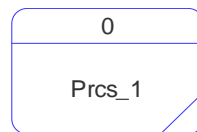
3.4.3 Data Flow Diagram (DFD)

Menurut Andri Kristanto (2004), Data Flow Diagram (DFD) adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data tersebut

disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan, dan proses yang dikenakan pada data tersebut.

Data Flow Diagram merupakan suatu metode pengembangan sistem yang terstruktur (*structured analysis and design*). Penggunaan notasi dalam data flow diagram sangat membantu untuk memahami suatu sistem pada semua tingkat kompleksitas. Pada tahap analisis, penggunaan notasi ini dapat membantu dalam berkomunikasi dengan pemakai sistem untuk memahami sistem secara logika.

Di dalam data flow diagram terdapat empat simbol yang digunakan yaitu *process*, *external entity*, *data store*, dan *data flow*. Simbol *process* digunakan untuk melakukan suatu perubahan berdasarkan data yang diinputkan dan menghasilkan data dari perubahan tersebut. Simbol *process* dapat digambarkan sebagai bentuk berikut:



Gambar 3.1 Simbol Process

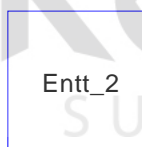
Pada bentuk gambar *process*, bagian atas berisi nomor untuk identitas proses. Suatu proses dengan nomor 0 (nol atau kosong) menandakan bahwa proses tersebut adalah sebuah *context diagram*. Diagram ini merupakan level tertinggi dari DFD yang menggambarkan hubungan sistem dengan lingkungan luarnya. Pembuatan *context diagram* dapat dilakukan dengan terlebih dahulu menentukan nama sistemnya, menentukan batasan dari sistem, dan menentukan *terminator* yang diterima atau diberikan daripada sistem untuk kemudian dilakukan penggambaran.

Nomor 1, 2, 3, dan seterusnya menandakan bahwa proses tersebut diartikan sebagai proses level-0 (nol) yang merupakan hasil turunan atau *decompose* dari proses *context diagram*. Proses level-0 membahas sistem secara lebih mendetil, baik dipandang dari segi kegiatan dari sebuah bagian, alur data yang ada, maupun *database* yang digunakan didalamnya. Pembuatannya dapat dilakukan dengan cara menentukan proses utama yang ada dalam sistem, menentukan alur data yang diterima dan diberikan masing-masing proses daripada sistem sambil memperhatikan konsep keseimbangan (alur data yang masuk atau keluar dari suatu level harus sama dengan alur data yang masuk dan keluar pada

level berikutnya), memunculkan *data store* sebagai sumber maupun tujuan data (*optional*), menggambarkan diagram level-0, menghindari perpotongan arus data, dan melakukan pemberian nomor pada proses utama (nomor tidak menunjukkan urutan proses).

Nomor 1.1, 1.2, 2.1, 2.2, dan seterusnya merupakan sebuah proses turunan atau *decompose* dari proses level-0 yang disebut sebagai proses level-1 (satu). Proses level-1 menggambarkan detail kerja dari sebuah bagian dalam sebuah sistem. Penggambarannya dilakukan dengan cara menentukan proses yang lebih kecil (sub-proses) dari proses utama yang ada di level-0, menentukan apa yang diterima atau diberikan masing-masing sub-proses daripada sistem dan tetap memperhatikan konsep keseimbangan, memunculkan *data store* sebagai sumber maupun tujuan alur data (*optional*), menggambar DFD level-1, dan berusaha untuk menghindari perpotongan arus data. Hasil turunan akhir disebut sebagai *the lowest level*, dimana hasil akhir ini tergantung dari kompleksitas sistem yang ada.

a. *External entity* disimbolkan dengan bentuk persegi yang digunakan untuk menggambarkan pelaku-pelaku sistem yang terkait, dapat berupa orang-orang, organisasi maupun instansi. *External entity* dapat memberikan masukan kepada *process* dan mendapatkan keluaran dari *process*. *External entity* digambarkan dalam bentuk sebagai berikut:



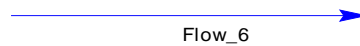
Gambar 3.2 Simbol External Entity

b. *Data store* digunakan sebagai media penyimpanan suatu data yang dapat berupa *file* atau *database*, arsip atau catatan manual, lemari *file*, dan tabel-tabel dalam *database*. Penamaan *data store* harus sesuai dengan bentuk data yang tersimpan pada *data store* tersebut, misalnya tabel pelamar, tabel pendidikan, tabel lulus seleksi, dan lain-lain. *Data store* digambarkan dalam bentuk simbol sebagai berikut:



Gambar 3.3 Simbol Data Store

- c. *Data flow* merupakan penghubung antara *external entity* dengan *process* dan *process* dengan *data store*. *Data flow* menunjukkan aliran data dari satu titik ke titik lainnya dengan tanda anak panah mengarah ke tujuan data. Penamaan *data flow* harus menggunakan kata benda, karena di dalam *data flow* mengandung sekumpulan data. *Data flow* digambarkan dengan bentuk simbol sebagai berikut:



Gambar 3.4 Simbol Data Flow

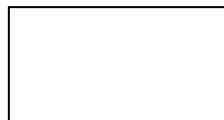
3.4.4 Entity Relationship Diagram (ERD)

Proses *reverse engineering* terhadap suatu basis data menjadi suatu kebutuhan bagi perancang basis data untuk mengetahui struktur dari sebuah basis data. Struktur tersebut biasanya dimodelkan dalam bentuk *Entity Relationship Diagram* (ERD).

ERD dibagi menjadi 2 macam yaitu: *Conceptual Data Model* (CDM), dan *Physical Data Model* (PDM). Simbol-simbol yang sering digunakan adalah:

1. Entity

Entity merupakan sesuatu yang mudah diidentifikasi. Sebuah *entity* bisa berupa obyek, tempat, orang, konsep, atau aktivitas. *Entity* dinyatakan dalam simbol persegi panjang. Simbol *entity* pada Gambar 3.5.



Gambar 3.5 Entity.

2. Atribut

Atribut merupakan penjelasan-penjelasan dari *entity* yang membedakan *entity* satu dengan yang lain. Sebuah atribut juga merupakan sifat-sifat dari sebuah *entity*. Atribut dinyatakan dalam simbol ellips. Simbol atribut pada Gambar 3.6.



Gambar 3.6 Atribut.

3. Relationship

Relationship adalah penghubung antara suatu *entity* dengan *entity* yang lain dan merupakan bagian yang sangat penting di dalam mendesain *database*. Ada tiga tipe *relationship* yang dikenal yaitu :

a. One-to-One Relationship

Jenis hubungan antar tabel yang menggunakan secara bersama sebuah kolom *primary key*. Jenis hubungan ini tergolong jarang digunakan, kecuali untuk alasan keamanan atau kecepatan akses data.

b. One-to-Many Relationship

Jenis hubungan antar tabel dimana satu *record* pada satu tabel terhubung dengan beberapa *record* pada tabel lain.

c. Many-to-Many Relationship

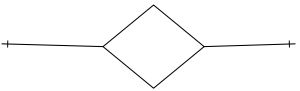
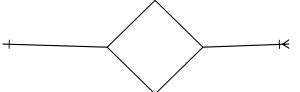
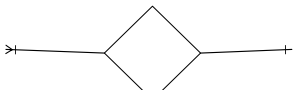
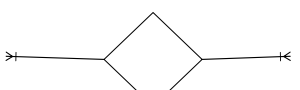
Jenis hubungan antar tabel dimana beberapa *record* pada satu tabel terhubung dengan beberapa *record* pada tabel lain.

d. Many-to-One Relationship

Jenis hubungan antar tabel dimana beberapa *record* pada satu tabel terhubung dengan satu *record* pada tabel lain.

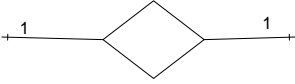
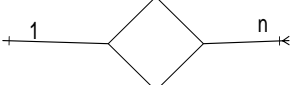

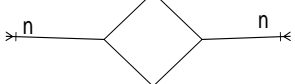
Menurut Sutanta (2004), relasi antar entitas dapat digambarkan melalui salah satu dari pilihan di bawah ini:

1. Pilihan 1

Jenis relasi	Simbol yang digunakan
1-ke-1 :	
1-ke-n :	
n-ke-1 :	
n-ke-n :	

Gambar 3.7 Simbol relasi antar entitas (pilihan 1)

2. Pilihan 2

Jenis relasi	Simbol yang digunakan
1-ke-1 :	
1-ke-n :	
n-ke-1 :	
n-ke-n :	

Gambar 3.8 Simbol relasi antar entitas (pilihan 2)

ERD dapat digambarkan menggunakan salah satu dari pilihan di atas, namun penggunaannya harus konsisten. Jika menggunakan simbol pilihan 1, maka untuk seluruh bagian ERD harus menggunakan simbol kelompok pilihan 1.

3. Kunci relasi

Kunci relasi atau *key* adalah suatu properti yang menentukan apakah suatu kolom pada table sangat penting atau tidak. Berdasarkan macamnya, kunci relasi terdiri dari:

a. Kunci kandidat

Yaitu satu atau atau gabungan minimal atribut yang bersifat unik yang dapat digunakan untuk mengidentifikasi setiap *record* dalam relasi.

b. Kunci primer

Yaitu bagian atau salah satu dari kunci kandidat yang digunakan sebagai kunci utama untuk membedakan setiap *record* dalam relasi. Kunci primer biasa disebut sebagai *primary key*.

c. Kunci alternatif

Yaitu bagian dari kunci kandidat yang tidak digunakan sebagai kunci utama.

d. Kunci penghubung

Kunci penghubung atau *foreign key* yaitu satu atau gabungan sembarang atribut yang menjadi kunci utama dalam relasi lain yang mempunyai hubungan secara logik. Kunci penghubung dan kunci utama harus memiliki tipe dan ukuran

data yang sama.

3.5 SQL Server 2005

Microsoft SQL Server adalah sebuah sistem manajemen basis data relasional (RDBMS) produk Microsoft. Bahasa kueri utamanya adalah Transact-SQL yang merupakan implementasi dari SQL standar ANSI/ISO yang digunakan oleh Microsoft dan Sybase. Umumnya SQL Server digunakan di dunia bisnis yang memiliki basis data berskala kecil sampai dengan menengah, tetapi kemudian berkembang dengan digunakannya SQL Server pada basis data besar.

Microsoft SQL Server dan Sybase/ASE dapat berkomunikasi lewat jaringan dengan menggunakan protokol TDS (Tabular Data Stream). Selain dari itu, Microsoft SQL Server juga mendukung ODBC (Open Database Connectivity), dan mempunyai driver JDBC untuk bahasa pemrograman Java. Fitur yang lain dari SQL Server ini adalah kemampuannya untuk membuat basis data mirroring dan clustering. Pada versi sebelumnya, MS SQL Server 2000 terserang oleh cacing komputer SQL Slammer yang mengakibatkan kelambatan akses Internet pada tanggal 25 Januari 2003.

3.6 Sistem Informasi Absensi

Absensi adalah suatu pendataan kehadiran, bagian dari pelaporan aktifitas suatu institusi, atau komponen institusi itu sendiri yang berisi data-data kehadiran yang disusun dan diatur sedemikian rupa sehingga mudah untuk dicari dan dipergunakan apabila sewaktu-waktu diperlukan oleh pihak yang berkepentingan.

Beberapa jenis absensi tersebut, yang membedakan jenis-jenis absensi tersebut adalah cara penggunaannya, dan tingkat daya guna. Secara umum jenis-jenis absensi dapat dikelompokkan menjadi dua, yaitu:

- a. Absensi manual adalah cara pengentrian kehadiran dengan cara menggunakan tulisan, biasanya berupa tanda tangan dll.
- b. Absensi non manual adalah suatu cara pengentrian kehadiran dengan menggunakan system terkomputerisasi, bisa menggunakan kartu dengan barcode, finger print ataupun dengan mengentrikan nik dll.

Kelebihan dalam memanfaatkan teknologi informasi pada absensi ini adalah:

- Absensi dapat dilakukan dengan lebih mudah dan cepat
- Informasi hasil pencarian yang disajikan lebih akurat
- Mempermudah dalam pembuatan laporan dan rekapitulasi
- Terdapat fasilitas informasi

Dengan demikian sistem yang terkomputerisasi akan mempermudah kerja bagian HRD, serta meningkatkan kedisiplinan pada karyawan.

