

## BAB III LANDASAN TEORI

Landasan teori merupakan suatu dasar mengenai pendapat penelitian atau penemuan yang didukung oleh data dan argumentasi. Landasan teori digunakan untuk menjelaskan teori-teori yang mendukung penyusunan laporan kerja praktik ini yang antara lain:

### 3.1 Sistem Informasi

#### 3.1.1 Definisi Sistem dan Informasi

Menurut (Romney & Steinbart, 2006) pendefinisian sistem dibagi menjadi tiga, yaitu (1) *Entitas* yang terdiri dari dua atau lebih komponen atau subsistem yang berinteraksi untuk mencapai suatu tujuan. (2) Perlengkapan dan program yang terdiri dari instalasi komputer lengkap. (3) Program dan prosedur terkait yang menjalankan suatu tugas dalam sebuah komputer.

Data adalah fakta yang menggambarkan suatu kejadian. Data yang diolah menjadi bentuk yang lebih berguna dan dapat diambil suatu keputusan disebut informasi. Sistem informasi adalah suatu sistem kerja yang terorganisir dalam pengolahannya hingga menghasilkan informasi yang berfungsi dalam pengambilan keputusan. Sedangkan Menurut Robert A. Leitch dan K. Roscoe Davis dalam buku Jogiyanto HM., (1999: 11), “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”.

### 3.1.2 Sekilas tentang Sistem

Sistem adalah salah satu bagian dari istilah sistem informasi di mana sistem berperan penting dalam sebuah instansi. Untuk lebih memahami definisi dari sistem, Murdick (1991:27) mengatakan bahwa sistem adalah seperangkat elemen yang membentuk kumpulan atau prosedur-prosedure/bagan-bagan pengolahan yang mencari suatu tujuan bagian atau tujuan bersama dengan mengoperasikan data dan/atau barang pada waktu rujukan tertentu untuk menghasilkan informasi dan/atau energi dan/atau barang. Sistem itu sendiri memiliki beberapa karakteristik atau sifat-sifat tertentu. Menurut Hartono (2005:3) sistem mempunyai karakteristik atau sifat-sifat tertentu, yakni:

#### 1. Komponen

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerjasama membentuk satu kesatuan. Komponen - komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu mempengaruhi proses sistem secara keseluruhan.

#### 2. Batasan sistem.

Batasan sistem (*boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batasan suatu sistem menunjukkan ruang lingkup dari sistem tersebut.

#### 3. Lingkungan Luar Sistem.

Lingkungan luar (*environment*) dari suatu sistem adalah apapun diluar batas sistem yang mempengaruhi operasi. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan berupa energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Sedang lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

#### 4. Penghubung Sistem

Penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

#### 5. Masukan Sistem

Masukan (*Input*) sistem adalah energi yang masukan kedalam sistem. Masukan dapat berupa masukan perawatan (*maintenance Input*), dan masukan sinyal (*signal Input*). *Maintenance Input* adalah energi yang dimasukan supaya tersebut dapat beroperasi. *Signal Input* adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh didalam komputernya dan data adalah *signal Input* untuk diolah menjadi informasi.

#### 6. Keluaran Sistem

Keluaran (*output*) sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Misalnya untuk sistem komputer, panas yang dihasilkan adalah keluaran

yang tidak berguna dan merupakan hasil sisa pembuangan, sedang informasi adalah keluaran yang dibutuhkan.

#### 7. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan mengubah masukan menjadi keluaran. Suatu sistem produksi akan mengolah

masukan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi.

#### 8. Sasaran Sistem

Sebuah sistem sudah tentu mempunyai sasaran ataupun tujuan. Dengan adanya sasaran sistem, maka kita dapat menentukan masukan yang dibutuhkan sistem dan keluaran apa yang akan dihasilkan sistem tersebut dapat dikatakan berhasil apabila mencapai/mengenai sasaran atau pun tujuan

#### 3.1.3 Informasi

Informasi merupakan salah satu kebutuhan di dalam suatu instansi, perusahaan, organisasi, lembaga serta lingkungan yang berada di luar sistem. Informasi dianggap sangat penting karena dengan adanya informasi dapat menambah pengetahuan, mengurangi ketidakpastian dan resiko kegagalan serta dapat membantu para pemimpin dalam mengambil suatu kesimpulan dan keputusan yang efektif dan efisien.

Salah satu informasi yang penting dari suatu perusahaan ataupun instansi pemerintah adalah tentang kepegawaiannya yang terkait dengan apa saja yang termasuk dalam melengkapi data pegawai tersebut. Mulai dari Pangkat dan Jabatan, Lembur, Penggajian dan lain-lain.

Sistem Penggajian Pegawai pada CV. Rahayu Sentosa adalah cara baru dalam pengolahan data penggajian karyawan. Sistem penggajian ini merupakan dasar dan konsekuen logis dari perkembangan teknologi informasi dan komunikasi. Karena dahulunya hanya memakai pencatatan manual dengan buku yang memiliki kelemahan dalam hal waktu yang diperlukan cukup lama pada proses penginputan data serta kesalahan perhitungan. Seiring dengan perkembangan ilmu komputer, dibutuhkan sistem informasi yang terintegrasi sehingga menghasilkan laporan penggajian yang lebih efektif dan efisien.

#### **3.1.4 Penggajian (*Payroll*)**

Menurut Alimansyah (2003), Gaji adalah pembayaran atas penyerahan jasa yang dilakukan oleh karyawan yang mempunyai jenjang jabatan Manajer, Pegawai Administrasi, Supervisor dan lain-lain, dan pada umumnya gaji dibayarkan secara tetap tiap bulan.

Menurut Amir Abadi Yusuf (2000), Penggajian (*Payroll*) adalah sistem yang menyajikan cara-cara penggajian pegawai secara memadai dan akurat, menghasilkan laporan-laporan penggajian yang diperlukan, dan menyajikan informasi kebutuhan pegawai kepada manajemen.

#### **3.1.5 Absensi**

Fungsi absensi sering dianggap sebagai bagian yang paling penting dan berpengaruh dalam menghitung total gaji, bahkan bisa dikatakan sebagian besar proses menghitung total gaji berasal dari kegiatan absensi. Karena output yang akan di keluarkan dari proses absensi merupakan penghitungan untuk total gaji agar dapat di jumlahkan.

### 3.1.6 Menghitung Total gaji

Fungsi Menghitung total gaji ini merupakan suatu fungsi yang baru untuk menggantikan sebuah proses atau cara yang ada sebelumnya yang telah ada di CV. Rahayu Sentosa, walaupun sudah terkomputerisasi tapi masih menggunakan *microsoft excel* yang dimana kesalahan masih sering terjadi karena kesalahan Inputan absensi yang dilakukan berulang-ulang di setiap harinya. Oleh karena itu fungsi hitung total gaji memperbaiki fungsi yang ada. Dengan memberikan Inputan berupa berapa jumlah bolos maka dihitung berapa potongan yang akan di berikan kepada karyawan yaitu gaji pokok di kurangi total potongan, dan penambahan diberikan saat Inputan lembur yang disesuaikan dengan berapa jam karyawan tersebut melakukan lembur setelah semua dihitung, gaji pokok di kurangi total potongan dan penambahan jumlah lembur akan menghasilkan total gaji. Rumus untuk menghitung total gaji adalah sebagai berikut.

$$\text{Total Gaji} = \text{Gaji Pokok} + ((\text{Tunjangan}) - (\text{Terlambat}))$$

**Gambar 3.1 Rumus Perhitungan Gaji**

### 3.2 Analisis dan Perancangan Sistem

Analisis sistem dilakukan dengan tujuan dapat mengidentifikasi dan mengevaluasi permasalahan yang terjadi dan hasil/tujuan yang diharapkan, sehingga dapat diusulkan perbaikannya. Perancangan sistem merupakan penguraian suatu sistem informasi yang utuh ke dalam bagian komputerisasi yang dimaksud, mengidentifikasi dan mengevaluasi permasalahan, menentukan kriteria, menghitung konsistensi terhadap kriteria yang ada, serta mendapatkan hasil atau

tujuan dari masalah tersebut serta mengimplementasikan seluruh kebutuhan operasional dalam membangun aplikasi.

Menurut Kendall (2003), analisa dan perancangan sistem dipergunakan untuk menganalisis, merancang, dan mengimplementasikan peningkatan-peningkatan fungsi bisnis yang dapat dicapai melalui penggunaan sistem informasi terkomputerisasi. Berikut ini adalah proses dalam analisis dan perancangan sistem:

### **3.2.1 Data Flow Diagram (DFD)**

*Data Flow Diagram* adalah alat pembuatan model yang menggambarkan sistem dari suatu jaringan proses fungsional yang dihubungkan dengan alir data secara manual atau terkomputerisasi. Menurut (Kendall, 2003) DFD merupakan suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut, dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut. Penggambaran DFD disusun berdasarkan tingkatan di bawah ini :

#### *a. Context Diagram*

Merupakan diagram awal yang terdiri dari suatu proses dan menggambarkan ruang lingkup proses.

#### *b. Diagram Level 0*

Merupakan diagram yang menggambarkan proses utama dari sistem serta hubungan entity, proses, alur data, dan data source.

#### *c. Diagram Detail*

Merupakan penguraian dalam proses yang ada pada diagram level 0. Diagram ini adalah diagram yang paling rendah dan tidak dapat diuraikan lagi.

*Data Flow Diagram (DFD)* memiliki empat komponen, yaitu :

a. *Terminator* atau *External Entity*

External entity merupakan kesatuan di lingkungan sistem yang dapat berupa orang atau sistem yang berada di lingkungan luar sistem yang memberikan masukan atau menerima keluaran dari sistem. Dilambangkan dengan persegi dan diberi nama orang atau sistem tersebut sebagai penjelas.

b. Proses

Merupakan komponen yang berfungsi untuk mentransformasikan sistem dari *input* menuju ke *output*. Dilambangkan dengan lingkaran atau persegi panjang dengan sudut tumpul. Proses diberi nama untuk menerangkan proses atau kegiatan yang akan dilaksanakan.

c. Alur Data

Alur data digambarkan dengan anak panah yang menuju ke dalam proses maupun ke luar proses. Alur data digunakan untuk menerangkan perpindahan data atau informasi dari suatu bagian ke bagian lainnya.

d. *Data Store*

Merupakan tempat pengumpulan data (data tersimpan) yang disimbolkan dengan dua garis horisontal *paralel*. *Data store* perlu diberikan nama untuk menjelaskan nama dari *file*-nya. *Data store* berkaitan dengan penyimpanan data secara terkomputerisasi.



### 3.2.2 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram (ERD)* adalah diagram yang menggambarkan hubungan antara *entitas* dan *relasinya*. Entitas merupakan objek dari suatu organisasi dan biasanya memiliki atribut. Relasi adalah hubungan antar entitas. Menurut (Marlinda, 2004), *attribute* adalah kolom di sebuah relasi. Macam-macam *attribute* yaitu :

a. *Simple Attribute*

Atribut yang unik dan tidak dimiliki oleh attribute lainnya.

b. *Composite Attribute*

Atribut yang memiliki dua nilai harga.

c. *Single Value Attribute*

Atribut yang hanya memiliki satu nilai harga.

d. *Multi Value Attribute*

Atribut yang banyak memiliki nilai harga.

e. *Null Value Attribute*

Atribut yang tidak memiliki nilai harga.

*ERD* digunakan untuk menguji model dan mengabaikan proses apa yang harus dilakukan. *ERD* dapat dikategorikan menjadi tiga bagian, yaitu :

1. *One to one relationship*

Jenis hubungan antar tabel yang menggunakan bersama sebuah kolom *primary key*. Jenis hubungan ini tergolong jarang digunakan, kecuali untuk alasan keamanan atau kecepatan akses data. Misalnya, satu departemen hanya mengerjakan satu jenis pekerjaan saja dan satu pekerjaan hanya dikerjakan oleh satu departemen saja.

## 2. *One to many relationship*

Jenis hubungan antar tabel dimana satu *record* pada satu tabel terhubung dengan beberapa *record* pada tabel lain. Jenis hubungan ini merupakan yang paling sering digunakan. Misalnya, suatu pekerjaan hanya dikerjakan oleh satu departemen saja, namun suatu departemen dapat mengerjakan beberapa macam pekerjaan sekaligus.

## 3. *Many to many relationship*

Jenis hubungan ini merupakan hubungan antar tabel dimana beberapa *record* pada satu tabel terhubung dengan beberapa *record* pada tabel lain. Misalnya, satu departemen mampu mengerjakan banyak pekerjaan, juga satu pekerjaan dapat ditangani oleh banyak departemen.

*Entity Relationship Diagram* dibagi menjadi dua jenis model, yaitu :

### a. *Conceptual Data Model (CDM)*

*Conceptual Data Model* adalah jenis model data yang menggambarkan hubungan antar tabel secara konseptual.

### b. *Physical Data Model (PDM)*

*Physical Data Model* adalah adalah jenis model data yang menggambarkan hubungan antar tabel secara fisikal.

## 3.3 Data

Pengertian data menurut beberapa ahli yaitu:

- a. Jogyanto (Analisis dan Desain Sistem Informasi; 8) Data merupakan kenyataan yang menggambarkan suatu kejadian – kejadian dan kesatuan nyata.
- b. Menurut Nuzulla Agustina data adalah keterangan mengenai sesuatu hal yang sudah sering terjadi dan berupa himpunan fakta, angka, grafik,

tabel, gambar, lambang, kata, huruf-huruf yang menyatakan sesuatu pemikiran, objek, serta kondisi dan situasi.

- c. Menurut Inmon (2005, p493), data adalah kumpulan dari fakta, konsep, atau instruksi pada penyimpanan yang digunakan untuk komunikasi, perbaikan dan diproses secara otomatis yang mempresentasikan informasi yang dapat di mengerti oleh manusia.
- d. Menurut Turban (2010, p41), data adalah deskripsi dasar dari benda peristiwa, aktivitas dan transaksi yang direkam, dikelompokkan, dan disimpan tetapi belum terorganisir untuk menyampaikan arti tertentu.

Jenis-jenis data dapat dibagi berdasarkan sifatnya, sumbernya, cara memperolehnya, dan waktu pengumpulannya. Menurut sifatnya, jenis-jenis data yaitu:

- a. Data Kualitatif: data kualitatif adalah data yang tidak berbentuk angka, misalnya: Kuesioner Pertanyaan tentang suasana kerja, kualitas pelayanan sebuah rumah sakit atau gaya kepemimpinan, dll.
- b. Data Kuantitatif: data kuantitatif adalah data yang berbentuk angka, misalnya: harga saham, besarnya pendapatan, dll.

Jenis-jenis data menurut sumbernya, antara lain:

- a. Data Internal: data internal adalah data dari dalam suatu organisasi yang menggambarkan keadaan organisasi tersebut. Contohnya: suatu perusahaan, jumlah karyawannya, jumlah modalnya, atau jumlah produksinya, dll.
- b. Data Eksternal: data eksternal adalah data dari luar suatu organisasi yang dapat menggambarkan faktor-faktor yang mungkin mempengaruhi hasil kerja suatu organisasi. Misalnya: daya beli masyarakat mempengaruhi hasil penjualan suatu perusahaan.

Jenis-jenis data menurut cara memperolehnya, antara lain:

- a. Data Primer (*primary data*): data primer adalah data yang dikumpulkan sendiri oleh perorangan/suatu organisasi secara langsung dari objek yang diteliti dan untuk kepentingan studi yang bersangkutan yang dapat berupa *interview, observasi*.
- b. Data Sekunder (*secondary data*): data sekunder adalah data yang diperoleh/ dikumpulkan dan disatukan oleh studi-studi sebelumnya atau yang diterbitkan oleh berbagai instansi lain. Biasanya sumber tidak langsung berupa data dokumentasi dan arsip-arsip resmi.

Jenis-jenis data menurut waktu pengumpulannya, antara lain:

- a. Data *cross section*, yaitu data yang dikumpulkan pada suatu waktu tertentu (*at a point of time*) untuk menggambarkan keadaan dan kegiatan pada waktu tersebut. Misalnya; data penelitian yang menggunakan kuesioner.
- b. Data berkala (*time series data*), yaitu data yang dikumpulkan dari waktu ke waktu untuk melihat perkembangan suatu kejadian/kegiatan selama periode tersebut. Misalnya, perkembangan uang beredar, harga 9 macam bahan pokok penduduk.

### **3.4 Microsoft Visual Studio 2010**

*Visual Studio 2010* pada dasarnya adalah sebuah bahasa pemrograman komputer. Dimana pengertian dari bahasa pemrograman itu adalah perintah-perintah atau instruksi yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu.

*Visual Studio 2010* (yang sering juga disebut dengan *VB .Net 2010*) selain disebut dengan bahasa pemrograman, juga sering disebut sebagai sarana (*tool*)

untuk menghasilkan program-program aplikasi berbasis windows. Beberapa kemampuan atau manfaat dari *Visual Studio 2010* diantaranya seperti:

- a. Untuk membuat program aplikasi berbasis *windows*.
- b. Untuk membuat objek-objek pembantu program seperti, misalnya: *control ActiveX, file Help*, aplikasi Internet dan sebagainya.
- c. Menguji *program (debugging)* dan menghasilkan program berakhiran *EXE* yang bersifat *executable* atau dapat langsung dijalankan.

*Visual Studio 2010* adalah bahasa yang cukup mudah untuk dipelajari. Bagi programmer pemula yang baru ingin belajar program, lingkungan Visual Studio dapat membantu membuat program dalam sekejap mata. Sedang bagi programmer tingkat lanjut, kemampuan yang besar dapat digunakan untuk membuat program-program yang kompleks, misalnya lingkungan *net-working* atau *client server*.

Bahasa *Visual Studio* cukup sederhana dan menggunakan kata-kata bahasa Inggris yang umum digunakan. Kita tidak perlu lagi menghafalkan sintaks-sintaks maupun format-format bahasa yang bermacam-macam, di dalam Visual Basic semuanya sudah disediakan dalam pilihan-pilihan yang tinggal diambil sesuai dengan kebutuhan. Selain itu, sarana pengembangannya yang bersifat visual memudahkan kita untuk mengembangkan aplikasi berbasis Windows, bersifat mouse-driven (digerakkan dengan mouse) dan berdaya guna tinggi.

Beberapa keistimewaan *Visual Studio 2010* ini diantaranya seperti:

- a. Menggunakan platform pembuatan program yang dinamakan *developer studio*, yang memiliki tampilan dan sarana yang sama dengan *Visual C++* dan *Visual J++*. Dengan begitu Anda dapat bermigrasi atau belajar bahasa pemrograman lainnya dengan mudah dan cepat.

- b. Memiliki compiler handal yang dapat menghasilkan file executable yang lebih cepat dan lebih efisien dari yang sebelumnya.
- c. Memiliki beberapa tambahan wizard yang baru. Wizard adalah sarana yang mempermudah di dalam pembuatan aplikasi dengan mengotomisasi
- d. Tugas - tugas tertentu.
- e. *Visual Studio 2010* mempunyai beberapa fitur untuk pengembangan berbagai macam aplikasi yang diantaranya; *Windows Development, Web Development, Office Development, Sharepoint Development, Cloud Development (Windows Azure), Silverlight Tooling, Multi-Core Development, Customizable IDE.*

### **3.5 Microsoft SQL Server 2008**

Pada dasarnya pengertian dari *SQL Server* itu sendiri adalah bahasa yang dipergunakan untuk mengakses data dalam basis data relation. Bahasa ini secara defacto adalah bahasa standar yang digunakan dalam manajemen basis data relasional. Saat ini hampir semua server basis data yang ada mendukung bahasa ini dalam manajemen datanya. *SQL server 2005* merupakan salah satu produk dari *Relational Database Management System (RDBMS)*.

*SQL Server 2005 (codenamed Yukon)*, dirilis pada bulan Oktober 2005, adalah penerus *SQL Server 2000*. Ini termasuk dukungan untuk mengelola data *XML*, di samping data relasional. Untuk mendefinisikan tipe data xml yang dapat digunakan baik sebagai tipe data dalam kolom *database* atau sebagai literal dalam *query*. kolom *XML* dapat dikaitkan dengan skema *XSD*; *XML* data yang disimpan adalah diverifikasi terhadap skema. *XML* dikonversi ke tipe data internal biner sebelum disimpan dalam database. metode pengindeksan khusus yang dibuat tersedia untuk data *XML*. *query* data *XML* menggunakan *XQuery*; *Common*

*Language Runtime (CLR)* integrasi adalah fitur utama dengan edisi ini, memungkinkan seseorang untuk menulis kode *SQL* sebagai Kode Dikelola oleh *CLR*. *SQL Server 2005* menambahkan beberapa ekstensi untuk bahasa *T-SQL* untuk memungkinkan penyisipan query *XQuery* di *T-SQL*. Selain itu, juga mendefinisikan sebuah ekstensi baru untuk *XQuery*, yang disebut *XML DML*, yang memungkinkan modifikasi query berbasis data *XML*. *SQL Server 2005* juga memungkinkan *server database* yang akan terkena terhadap layanan web menggunakan paket-paket *TDS* dikemas dalam *SOAP (protokol)* permintaan. Bila data yang diakses melalui layanan web, hasil yang dikembalikan sebagai *XML*.

Berdasarkan tipe datanya, database server bisa digolongkan sebagai:

- a. *OLTP: Online Transaction Processing* (Transaksi)
- b. *OLAP: Online Analyze Processing* (Analisa)

Contoh *OLTP* adalah transaksi transfer uang di atm, mengambil no. urut saat antri di telkom dsb. Contoh *OLAP* adalah *summary* laporan penjualan saham, contoh lain report penjualan bulanan. *SQL Server 2005* dapat digunakan sebagai *OLTP* atau *OLAP server*.

### **3.6 Power Designer 6.0**

*Power designer 6.0 32 bit* adalah sebuah paket program desain tools yang digunakan untuk membuat Visualisasi, dokumentasi dan mendesain suatu sistem perangkat lunak. Paket ini meliputi:

#### **3.6.1 Process Analyst Model**

*Process analyst* adalah suatu *Powerdesigner tool* yang memungkinkan untuk menggambarkan aliran informasi dari suatu proses di dalam sistem informasi. Suatu *PAM* menunjukkan bagaimana data diubah dan menunjukkan proses fungsional yang berlaku untuk data itu.

Produk ini adalah merupakan suatu alat (tools) yang dapat digunakan untuk membuat pemodelan aliran informasi suatu sistem perangkat.

Dengan menggunakan produk software ini diharapkan dapat:

- a. Membangun model fungsional ( *OMT*) dan diagram arus data
- b. Menggunakan lambang sesuai dengan berbagai metoda didukung:  
*OMT, Yourdon/Demarco, Gane & Sarson, dan SSADM*
- c. Menciptakan suatu hirarki proses
- d. *Customize* dan mencetak laporan model
- e. Menggunakan data item dari suatu *Data Conceptual Model ( CDM)*
- f. Menggunakan *OLE* teknologi untuk menghubungkan rancangan model ke aplikasi lain.

Suatu *PAM* memungkinkan untuk:

- a. Meneliti proses dari suatu sistem
- b. Menghadirkan proses ini dalam wujud suatu model
- c. Menciptakan suatu struktur pohon hirarkis yang mewakili proses ini

### **3.6.2 Data architect**

*Data architect* adalah sebuah tools yang digunakan untuk mendesain suatu database. Tools ini akan memberikan keuntungan yaitu akan dapat memberikan dua tingkat pendekatan dalam desain pemodelan data yaitu tingkat konseptual atau logikal (*CDM ; Conceptual data Model*) dan tingkat fisik (*PDM : Physical data Model*)

Dengan menggunakan data architect kita akan dapat :



- a. Membuat Model suatu sistem informasi dengan menggunakan suatu *entity-relationship* diagram. Model ini disebut *Conceptual Data Model (CDM)*
- b. Menghasilkan suatu *Physical Data Model (PDM)* untuk suatu target database management sistem (DBMS), mempertimbangkan pokok-pokok dari DBMS
- c. Meng-*customize* PDM untuk menyesuaikan performa tingkah laku dan fisik
- d. Menghasilkan *script database* untuk target DBMS.
- e. Menghasilkan suatu referensi integritas jika mereka didukung oleh target database
- f. meng-*customize* dan mencetak laporan model
- g. Merencanakan balik aplikasi dan *database* dan aplikasi
- h. Menggambarkan atribut yang diperluas untuk *PDM object*

### 3.6.2.1 CDM (*Conceptual Data Model*)

Suatu CDM menghadirkan keseluruhan struktur logis dari suatu database, yang mana tidak terikat pada jenis apapun perangkat lunak atau penyimpanan data struktur. Suatu model konseptual sering berisi object data yang belum diterapkan didalam fisik database itu memberi suatu penyajian yang formal menyangkut data yang diperlukan untuk menjalankan perusahaan atau suatu aktivitas bisnis.

*CDM* mempunyai beberapa peran sebagai berikut :

- a. Menghadirkan organisasi data di (dalam) suatu format grafis.
- b. Memverifikasi kebenaran disain data.

- c. Menghasilkan *PDM* yang (mana) menetapkan implementasi fisik (menyangkut) *database*.

### 3.6.2.2 *PDM (Physical data Model)*

*PDM* adalah merupakan suatu implementasi fisik dari *database*. Dengan *PDM*, kita dapat mempertimbangkan secara detail tentang implementasi fisik nyata. *PDM* dimasukkan kedalam laporan perangkat lunak atau penyimpanan data struktur. Kamu dapat memodifikasi *PDM* untuk menyesuaikan batasan fisik (*physical constrain*) atau hasil rancangan

*PDM* mempunyai beberapa peran sebagai berikut :

- a. Menghadirkan organisasi fisik data di (dalam) suatu format grafis
- b. Menghasilkan catatan untuk modifikasi dan pembuatan *database*
- c. Menggambarkan batasan (*constrain*) dan referensi integritas
- d. Menghasilkan *extended* atribut
- e. Merekayasa balik *database* yang ada
- f. Memperbaharui suatu *CDM*

### 3.6.3 *Metaworks*

*Metaworks* merupakan kamus adalah suatu alat untuk me-manage pengembangan aplikasi di dalam suatu kelompok lingkungan perangkat lunak

Dengan menggunakan *Metaworks* kamus kita dapat:

- a. Menciptakan profil dan rekening/tg-jawab untuk para pemakai kamus

- b. Menciptakan proyek, model dan submodels, dan menugaskan [mereka/nya] ke para pemakai
- c. Mengunci dan membuka kunci model
- d. Menyuling/Menyadap dan memperkuat model dan submodels
- e. Membandingkan definisi obyek dan mengatur konflik data
- f. Membuat kamus *query*
- g. Memodifikasi object di (dalam) kamus

### 3.7 Microsoft Office Visio 2007

*Microsoft Visio* (atau sering disebut *Visio*) adalah sebuah program aplikasi komputer yang sering digunakan untuk membuat *diagram*, *diagram* alir (*flowchart*), *brainstorm*, dan skema jaringan yang dirilis oleh *Microsoft Corporation*. Aplikasi ini menggunakan *grafik vector* untuk membuat *diagram-diagramnya*. *Visio* aslinya bukanlah buatan *Microsoft Corporation*, melainkan buatan *Visio Corporation*, yang diakuisisi oleh *Microsoft* pada tahun 2000. Versi yang telah menggunakan nama *Microsoft Visio* adalah *Visio 2002*, *Visio 2003*, dan *Visio 2007* yang merupakan versi terbaru. *Visio2007 Standard* dan *Professional* menawarkan antarmuka pengguna yang sama, tapi seri *Professional* menawarkan lebih banyak pilihan *template* untuk pembuatan *diagram* yang lebih lanjut dan juga penataan letak (*layout*). Selain itu, edisi *Professional* juga memudahkan pengguna untuk mengoneksikan *diagram-diagram* buatan mereka terhadap beberapa sumber data dan juga menampilkan informasi secara *visual* dengan menggunakan *grafik*.