Bab 2
Landasan Teori

2.1. Pengertian Sistem Informasi

Sistem pada dasarnya adalah pengelolaan data untuk menghasilkan informasi yang berguna dan mendukung dalam suatu organisasi untuk memberikan kemudahan terhadap proses-proses yang ada. Secara sederhana sistem adalah suatu kumpulan dari unsur, komponen, atau variabel-variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu. (Jeffry L. Whitten, 2004)

Manfaat sistem adalah untuk mengintegrasikan semua unsur yang ada dalam suatu ruang lingkup, dimana komponen-komponen yang ada tidak dapat berdiri sendiri. Komponen atau sub sistem harus saling berintegrasi untuk membentuk suatu kesatuan sehingga tujuan dari sebuah sistem dapat tercapai.

Informasi adalah hasil pengolahan data yang diperoleh dari setiap elemen sistem yang ada dan dibuat dalam bentuk yang mudah dipahami oleh penerimanya. Informasi menggambarkan kejadian-kejadian nyata untuk menambah pemahaman terhadap fakta-fakta yang ada, sehingga dapat digunakan untuk pengambilan suatu keputusan yang bersumber dari data.

Kualitas dari suatu informasi tergantung pada tiga hal, yaitu: (Jeffry L. Whitten, 2004)

1. Akurat (*accurate*)

   Informasi harus bebas dari kesalahan dan tidak boleh menyesatkan. Akurat juga berarti bahwa informasi harus jelas mencerminkan maksud pembuatannya.

2. Tepat waktu (*timelines*)

   Informasi yang sampai pada penerima tidak boleh tertunda. Informasi yang sudah usang nilainya akan berkurang. Karena informasi merupakan landasan didalam pengambilan suatu keputusan.

3. Relevan (*relevance*)

   Informasi tersebut mempunyai manfaat untuk penggunanya. Relevansi informasi untuk setiap orang, antara satu dan lainnya pasti memiliki perbedaan.

Manfaat adanya sistem informasi yaitu untuk menyajikan informasi yang tepat dan dapat mendukung pengambilan sebuah keputusan, menyajikan informasi untuk mendukung operasi atau aktivitas harian dan menyajikan informasi yang sesuai dengan kebutuhan. Beberapa komponen sistem informasi dapat diklasifikasikan sebagai berikut:

1. Perangkat keras (*hardware*) dan perangkat lunak (*software*) yang berfungsi sebagai mesin.
2. Manusia (*people*) dan prosedur yang merupakan sumber daya manusia dan tata cara penggunaan sebuah mesin.
3. Data yang merupakan jembatan penghubung antara manusia dan mesin agar dapat terjadi proses pengolahan data.

2.2. Pengertian Administrasi

Administrasi adalah penyusunan dan pencatatan data serta informasi secara sistematis baik internal maupun eksternal dengan maksud menyediakan keterangan serta memudahkan memperoleh kembali informasi baik hanya sebagian maupun menyeluruh. (Zaka, 2014)

Administrasi memiliki beberapa karakteristik atau ciri-ciri antara lain:

1. Terdapat kelompok manusia yang terdiri dari 2 orang atau lebih
2. Terdapat kerja sama
3. Terdapat proses atau usaha
4. Terdapat bimbingan, kepemimpinan dan pengawasan
5. Terdapat tujuan

Adapun fungsi dari administrasi diantaranya adalah sebagai berikut:

1. *Planning* (Perencanaan), penyusun rencana memerlukan kegiatan administrasi, seperti pengumpulan data, pengolahan data dan penyusunan perencanaan.
2. *Organizing* (Pengorganisasian), aktivitas menyusun dan membentuk hubungan-hubungan kerja antara orang-orang sehingga terwujud suatu kesatuan usaha dalam mencapai tujuan-tujuan yang telah ditetapkan.
3. *Staffing*, salah satu fungsi dari manajemen yang menyusun personalia pada suatu organisasi mulai dari merekrut tenaga kerja, pengembangannya sampai dengan usaha untuk setiap tenaga kerja memberi daya guna yang maksimal terhadap organisasi.
4. *Directing* (Pengarahan atau bimbingan), fungsi manajemen yang berhubungan dengan usaha memberi bimbingan, saran, perintah-perintah untuk tugas yang dilaksanakan dengan baik dan benar-benar tertuju dari yang telah ditetapkan semula.
5. *Coordinating*, sebagian dari fungsi manajemen untuk melakukan sejumlah kegiatan agar berjalan baik dengan menghindari terjadinya kekacauan, percekcokan, kekosongan kegiatan yang dilakukan dengan menghubungkan, menyatukan dan menyelaraskan pekerjaan bawahan sehingga terdapat kerja sama yang terarah dalam usaha untuk mencapai tujuan organisasi.
6. *Reporting*, manajemen yang berada pada penyampaian perkembangan atau hasil dari kegiatan dengan pemberian keterangan dari tugas dan fungsi para pejabat

yang lebih tinggi baik lisan maupun tulisan sehingga dalam menerima laporan dapat memperoleh gambaran tentang pelaksanaan tugas orang yang memberi laporan.

7. *Budgeting*, suatu kegiatan yang mengelola dan melaksanakan perencanaan yang berkelanjutan mengenai keuangan atau anggaran.

2.3. Analisis Sistem

Analisis sistem (*system analyst*) adalah sebuah teknik pemecahan masalah yang menguraikan sebuah sistem menjadi bagian-bagian komponen dengan tujuan mempelajari seberapa bagus bagian-bagian komponen tersebut bekerja dan berinteraksi untuk meraih tujuan mereka. (Jeffry L Whitten, 2004)

Pada analisis sistem, dapat diketahui solusi seperti apa yang dibutuhkan dan mendukung dalam suatu pekerjaan. Analisis sistem juga berisi tentang langkah-langkah pemecahan masalah yang dapat menghasilkan suatu informasi.

2.3.1. Pendekatan-Pendekatan Analisis Model-Driven

Anslisis terstruktur, teknik informasi dan analisis berorientasi objek adalah contoh-contoh dari analisis model-driven. Analisis model-driven adalah analisis yang menggunakan gambar untuk mengkomunikasikan masalah-masalah, persyaratan dan solusi-solusi bisnis. (Jeffry L Whitten, 2004)

Analisis model-driven ini biasanya melakukan analisis sistem terstruktur dalam bentuk *flowchart* yang dibuat dengan menggunakan program aplikasi *grafis generatepurpose*, seperti *Microsoft Visio*. Hal ini dapat memudahkan *user* dalam mengetahui dan menganalisis komponen pembangun sistem.

2.3.2. Pendekatan-Pendekatan Analisis Sistem Terakselerasi

Discovery prototyping adalah sebuah teknik yang digunakan untuk mengidentifikasi persyaratan-persyaratan bisnis pengguna dengan membuat para pengguna bereaksi

pada implementasi *quick-and-dirty* atau bijaksana dan efektif tanpa cacat dan efek yang tidak diinginkan dalam persyaratan-persyaratan tersebut. (Jeffry L. Whitten, 2004)

Prototype harus memiliki fungsi yang sesuai dengan kebutuhan dan dapat membangun model kerja kecil dari persyaratan para pengguna atau desain yang diusulkan untuk sistem informasi. Tipe *prototyping* ini biasanya merupakan teknik desain, tapi pendekatannya dapat diaplikasikan lebih awal dalam siklus hidup pengembangan sistem untuk melakukan penemuan fakta dan analisis persyaratan.

2.3.3. FAST (*Framework for The Application of Sistem Thinking*)

FAST (*Framework for the Apllication of System Thinking*) adalah kerangka cerdas fleksibel untuk menyediakan tipe-tipe berbeda tipe proyek dan strategi. (Jeffry L. Whitten, 2004)

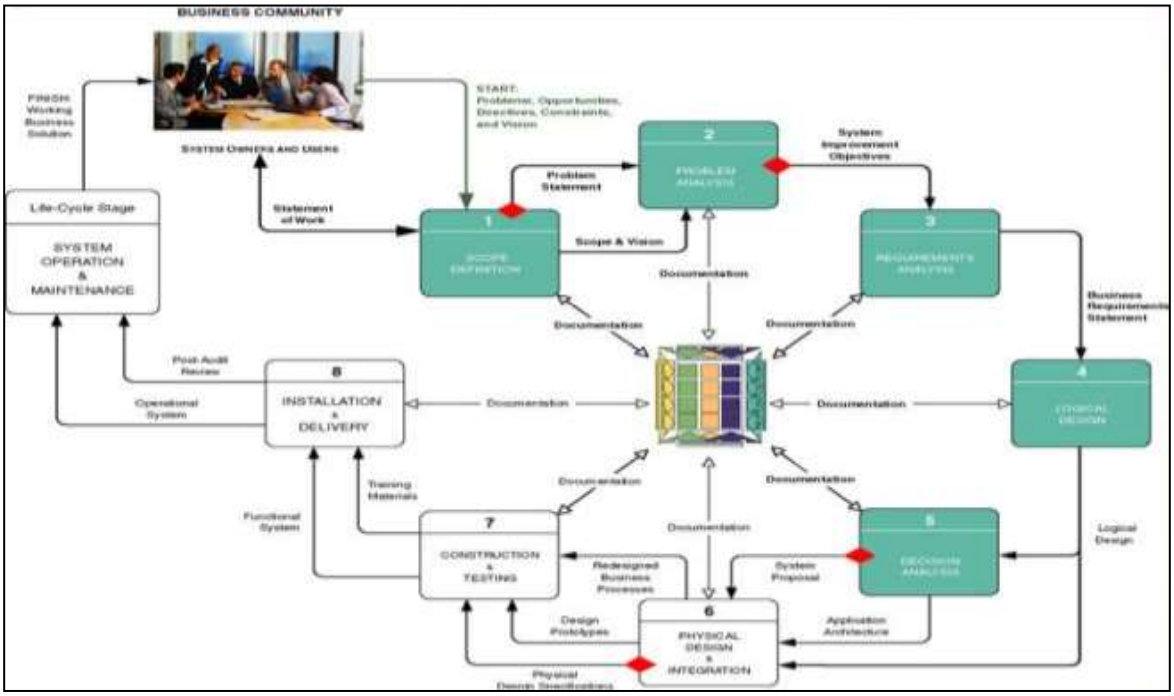
FAST tidak seperti metodologi lainnya karena tidak bersifat preskriptif atau tidak bergantung pada ketentuan resmi yang berlaku. Dalam FAST juga hal pertama yang harus dilakukan adalah melakukan pendekatan pemecahan masalah seperti memahami dan mengidentifikasi dahulu masalah seperti apa yang dihadapi dan menentukan solusinya.

Terdapat beberapa fase dalam metode FAST, yaitu:

1. Definisi Lingkup (*Scope Definition Phase*)
2. Analisis Masalah (*Problem Analysis Phase*)
3. Analisis Persyaratan (*Requirements Analysis Phase*)
4. Desain Logis (*Logical Design Phase*)
5. Analisis Keputusan (*Decision Analysis Phase*)

6. Desain Fisik dan Integrasi (*Physical Design* & *Integration*)
7. Konstruksi dan Pengujian (*Construction* & *Testing*)
8. Instalasi dan Pengiriman (*Installation* & *Delivery*)

Seperti pada gambar 2.1 berikut yang menunjukkan hubungan antara blok-blok pembangun sistem informasi yang berisi blok pembangun untuk pengetahuan, proses dan komunikasi dengan fase-fase dalam metodologi FAST.



Gambar 2.1. Sistem Diagram Konteks Analisis dan Perancangan Sistem

2.3.3.1. Definisi Lingkup (*Scope Definition*)

Fase pertama proyek yang umum adalah definisi lingkup. Ada dua tujuan dari definisi lingkup. Pertama yaitu menjawab pertanyaan, "Apakah proyek ini pantas diperhatikan?" kedua, mengasusmsikan bahwa masalah tersebut memang pantas diperhatikan, menentukan ukuran dan batasan-batasan proyek, visi proyek, semua

batasan atau limit, partisipan proyek yang dibutuhkan, dan akhirnya, anggaran dan jadwal. (Jeffry L. Whitten, 2004)

Pada definisi ruang lingkup akan diketahui masalah awal dan pernyataan lingkup untuk proyek, sehingga seorang analis dapat menyediakan staf untuk tim proyek, memperkirakan anggaran pengembangan, dan meyiapkan jadwal untuk fase-fase lainnya Setelah itu, akan didapatkan hasil berupa keputusan pada lingkup, anggaran dan jadwal pelaksanaan proyek berupa penerimaan atau penolakkan proyek.

2.3.3.2. Analisis Masalah (*Problem Analysis*)

Fase analisis masalah mempelajari sistem yang ada dan menganalisa temuan-temuan untuk menyediakan tim proyek dengan pemahaman yang lebih mendalam akan masalah-masalah yang memicu proyek. (Jeffry L. Whitten, 2004)

Prasyarat analisis masalah adalah lingkup dan persyaratan masalah seperti didefinisikan dan disetujui dalama fase analisis lingkup. Produk jadi fase analisis masalah adalah satu tujuan perbaikan sistem yang diperoleh dari pemahaman menyeluruh terhadap masalah-masalah bisnis. Tujuan-tujuan ini tidak mendefinisikan *input*, *output*, atau proses, melainkan mendefinisikan kriteria bisnis tempat semua sistem baru akan dievakuasi.

2.3.3.3. Analisis Persyaratan (*Requirement Analysis*)

Fase analisis persyaratan adalah mendefinisikan dan memprioritaskan persyaratan-persyaratan bisnis. (Jeffry L. Whitten, 2004)

Jika kita salah mendefinisikan persyaratan yang dibutuhkan oleh pengguna maka akan berakibat ketidakpuasan dari pengguna dan proses perbaikan akan memakan waktu dan biaya yang lebih banyak lagi. Peneliti atau analis mendekati pengguna

sistem dan mencari tahu apa saja yang mereka perlukan dan butuhkan dari sistem baru yang akan dikembangkan dari suatu informasi. Informasi ini bertujuan untuk memperbaiki sistem yang telah ada dan sedang berjalan.

Terdapat dua jenis persyaratan yang diterjemahkan dalam fase analisis persyaratan yaitu:

1. Persyaratan fungsional

   Persyaratan fungsional merupakan deskripsi mengenai aktivitas, layanan dan fungsi yang harus disediakan oleh sebuah sistem. Contoh persyaratan fungsional adalah daftar input, proses, output dan data yang dibutuhkan untuk memenuhi sasaran.

2. Persyaratan non fungsional

   Persyaratan nonfungsional adalah deskripsi mengenai fitur, fasilitas, karakteristik dan batasan lainnya yang menentukan apakah sistem yang dibuat memuaskan atau tidak.

2.3.3.4. Desain Logis *(Logical Design)*

Desain logis merupakan proses dalam menterjemahkan berbagai persyaratan sebuah sistem ke dalam suatu model sistem tetapi hanya menggambarkan persyaratan bisnis saja bukan desain teknik atau implementasi dari persyaratan tersebut. Desain logis menggambarkan persyaratan dengan menggunakan struktur data, proses bisnis, aliran data, dan antar muka dengan pengguna serta mensahkan persyaratan tersebut berdasarkan fase sebelumnya yaitu fase analisis persyaratan. (Jeffry L. Whitten, 2004)

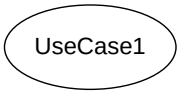
Salah satu model sistem yang paling sering digunakan dalam desain logis adalah *use case diagram*. *Use case diagram* adalah sebuah diagram yang menggambarkan

berbagai macam interaksi antara aktivitas dan pelaku yang terdapat di dalam sebuah sistem. Secara grafis, *use case diagram* menggambarkan siapa yang akan menggunakan sistem tersebut.

Terdapat tiga buah unsur utama yang terdapat di dalam *use case diagram* antara lain:

1. *Use-case* (Urutan Langkah atau Skenario)

   Use-case merupakan urutan dari langkah-langkah yang saling terkait secara tindakan dengan tujuan untuk melengkapi tugas bisnis tunggal. *Use-case* merupakan hasil dari persyaratan fungsional yang telah disusun kembali menjadi banyak sistem yang lebih kecil.



Gambar 2.2. Contoh Simbol *Use-case* dalam Diagram *Use-case*

2. *Actor* (Pelaku)

   Pelaku merupakan individu yang perlu berinteraksi dan menginisiasi kegiatan di dalam sebuah sistem untuk melakukan pertukaran informasi. Terdapat empat tipe dari pelaku di dalam diagram *use-case*, yaitu:

   a. Pelaku bisnis utama.
   b. Pelaku sistem utama.
   c. Pelaku server eksternal.
   d. Pelaku penerima eksternal.



Gambar 2.3. Contoh Simbol *Actor* dalam Diagram *Use-case*

3. *Relationship* (Hubungan)

Hubungan yang terdapat di dalam diagram *use case* berbeda-beda tergantung pada penggunaannya. Jenis hubungan tersebut adalah:

a. *Association* (Gabungan)

Hubungan antara *actor* dan satu *use case* atau aktivitas yang menggambarkan interaksi antara keduanya.

b. *Extend/Extension* (Perluasan)

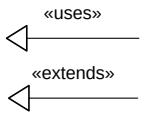
Hubungan ini berfungsi untuk mempermudah dalam memahami *use case* jika sebelumnya terlalu rumit dengan mengekstrak langkah yang lebih komplek ke dalam *use case* tersendiri.

c. *Uses* atau *Include*

Satu atau lebih *use case* yang melakukan berbagai langkah fungsional yang identik. Langkah tersebut biasanya disebut dengan abstrak yang mempresentasikan suatu betuk penggunaan ulang yang baik untuk mengurangi kejadian berulang.

d. *Depend on* (Tergantung Pada)

Menggambarkan hubungan dimana *use-case* yang satu bergantung pada *use case* yang lain dalam aktivitas atau interaksinya.

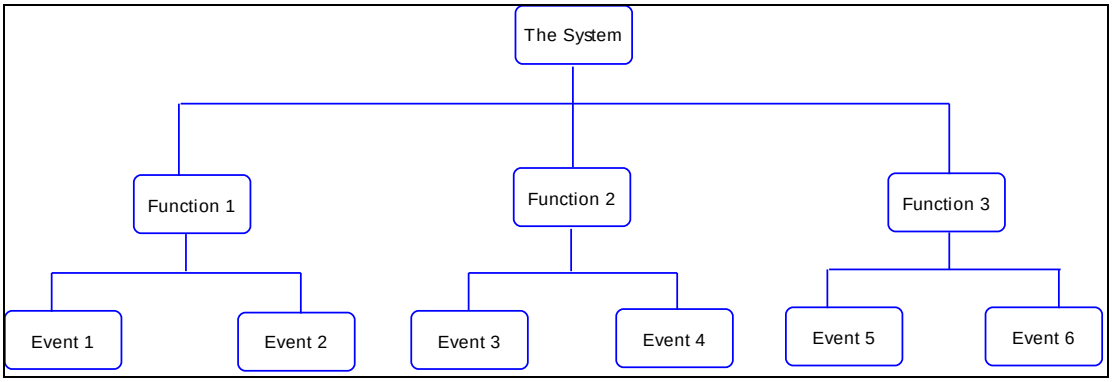


Gambar 2.4. Contoh Simbol *Relationship* dalam Diagram *Use-case*

2.3.3.5. Dekomposisi Diagram

Dekomposisi diagram adalah kegiatan menguraikan sistem menjadi sub sistem, proses dan sub proses komponennya. (Jeffry L. Whitten, 2004)

Diagram dekomposisi yang juga dinamakan bagan hierarki, menunjukan dekomposisi fungsional *top down* dan struktur sistem. Diagram dekomposisi pada dasarnya adalah alat perencanaan untuk model proses yang lebih detail, yang disebut diagram aliran data. Berikut ini merupakan contoh diagram dekomposisi pada gambar 2.1.

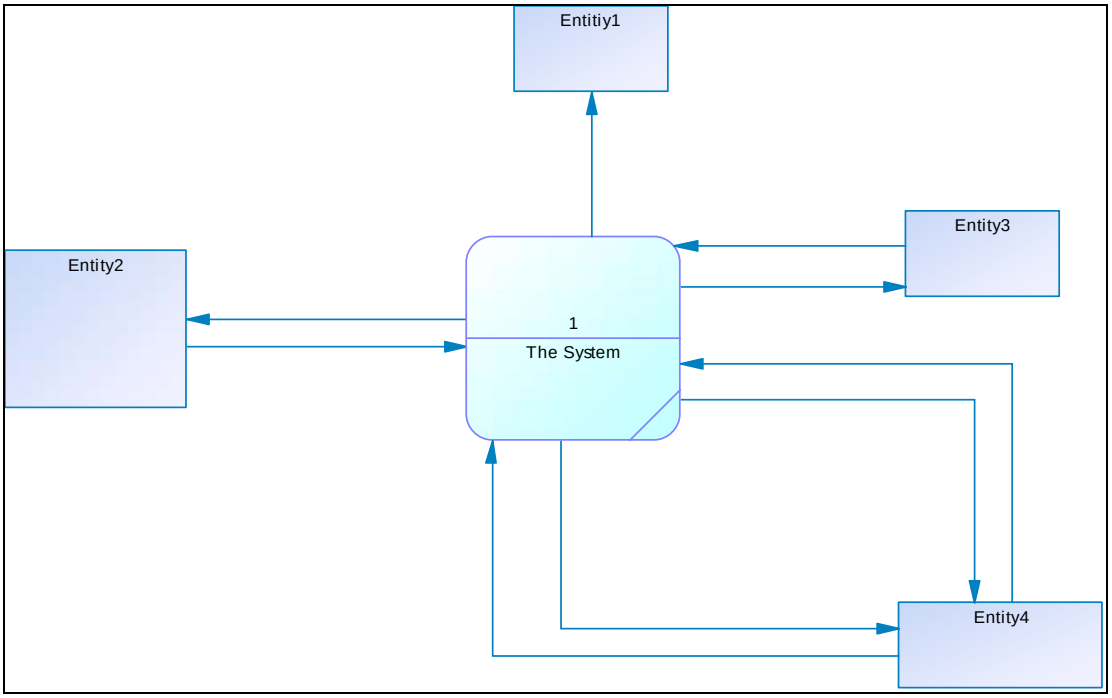


Gambar 2.5. Diagram Dekomposisi

2.3.3.6. Diagram Alir

Diagram alir adalah proses merespon *input* dan menghasilkan *output*. Data dalam pergerakan aliran data adalah data dalam pergerakan. (Jeffry L. Whitten, 2004)

Data aliran/*flow* adalah menunjukan *input* data ke proses atau *output* data dari proses. Aliran data juga digunakan untuk menunjukan pembuatan, pembacaan, penghapusan, atau pembaharuan data dalam *file* atau *database*. Berikut ini merupakan contoh diagram alir pada gambar 2.6.



Gambar 2.6. Diagram Alir

2.3.3.7. Analisis Keputusan

Persyaratan dari sebuah sistem yang akan kita buat dapat digunakan untuk menentukan alternatif-alternatif yang mungkin dapat digunakan misalnya basis komputer atau teknologi. Fase analisis keputusan menuntun untuk mengenali, menganalisis, merekomendasikan, dan mengimplementasikan solusi-solusi kandidat. Penting untuk mengenali solusi kandidat sehingga didapatkan solusi yang lebih baik dari alternatif-alternatif solusi yang ada. (Jeffry L. Whitten, 2004)

Pertanyaan-pertanyaan yang berhubungan dengan analisis keputusan misalnya:

1. Seberapa banyak sistem harus diotomasikan dengan teknologi informasi?
2. Haruskah kita membeli atau membangun perangkat lunak yang akan digunakan?
3. Teknologi informasi apakah yang mungkin berguna untuk aplikasi ini?

Fase analisis keputusan terletak pada bagian pengembangan dan dipicu oleh persyaratan bisnis yang tervalidasi ditambah dengan model sistem logis. Calon dari alternatif yang ada dievaluasi dengan kriteria kepraktisan teknis, kepraktisan operasional, kepraktisan ekonomis, kepraktisan jadwal, dan kepraktisan resiko.

2.3.3.8. Desain dan Intregasi Fisik (*Physical Design and Integration*)

Transformasi persyaratan bisnis ke dalam spesifikasi desain fisik yang akan memandu konstruksi sistem merupakan tujuan dari fase desain fisik dan integrasi. Beberapa persyaratan bisnis memang sudah tergambar pada fase desain logis yang menggunakan model sistem yang dipilih analis. Fase desain fisik juga mengilustrasikan implementasi teknis dari berbagai persyaratan bisnis tersebut. (Jeffry L. Whitten, 2004)

Perbedaan desain fisik dengan desain logis adalah desain fisik mewakili solusi teknik dengan spesifik sedangkan desain logis masih secara eksklusif berurusan dengan persyaratan bisnis yang terpisah dari solusi teknis. Fase desain fisik juga berhubungan dengan pandangan yang berbasis teknologi, yaitu:

1. Spesifikasi desain *database* fisik.
2. Proses bisnis fisik dan spesifikasi desain perangkat lunak.
3. Spesifikasi antar muka pengguna dan sistem fisik.

Terdapat dua filosofi ekstrim dari fase desain fisik, tetapi dalam pelaksanaannya kedua filosofi tersebut serikali dikombinasikan, yaitu:

1. Desain dengan spesifikasi

   Model sistem fisik dan spesifikasi terinci dihasilkan sebagai serangkaian cetak biru tertulis atau komputer untuk dijadikan instruksi

2. Desain dengan *prototyping* (pembuatan *prototype*)
   Aplikasi atau subsistem yang tidak lengkap tetapi dapat berfungsi dan disempurnakan berdasarkan umpan balik dari para pengguna dan pembuat sistem lain.

2.3.3.9. Konstruksi dan Pengujian (*Construction and Testing*)

Setelah diberikan tingkat model dan spesifikasi desain fisik, kita dapat mulai kontruksi dan menguji komponen-komponen sistem untuk desain tersebut. Fase kontruksi dan pengujian adalah sistem fungsional yang siap untuk di implementasikan. Ada dua fase kontruksi dan pengujian yaitu:

1. Membangun dan menguji sebuah sistem yang memenuhi persaratan bisnis dan spesifikasi desain fisik.

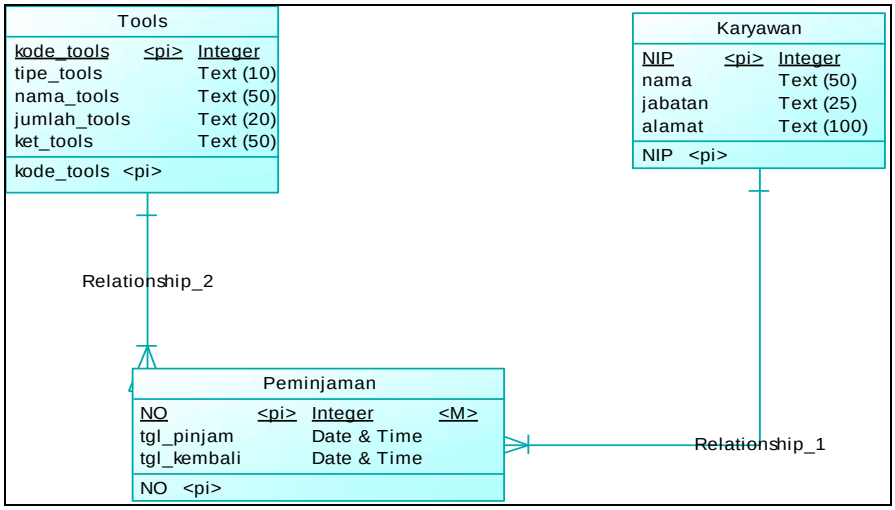
2. Mengimplementasikan antarmuka-antarmuka antara sistem antara sistem baru dan sistem yang telah ada. Sebagai tambahan, dokumentasi final (misalnya, sistem bantuan, manual pelatihan, dukungan help desk, intruksi control produksi) akan dikembangkan sebagai persiapan pelatihan dan operasi sistem. (Jeffry L. Whitten, 2004)

2.4. CDM (*Conceptual Data Model*)

CDM (*Conceptual Data Model*) atau model konsep data merupakan konsep yang berkaitan dengan pandangan pemakai terhadap data yang disimpan dalam basis data. CDM dibuat sudah dalam bentuk tabel-tabel tanpa tipe data yang menggambarkan relasi antar tabel untuk keperluan implementasi ke basis data.(M.Shalahuddin, 2014)

CDM digunakan untuk menentukan konsep awal dalam membuat sebuah sistem, CDM merupakan rancangan yang ada dalam sebuah *database* dan dapat menentukan

bagaimana sebuah rancangan sistem yang akan dibuat. Berikut merupakan contoh CDM pada studi kasus sistem informasi administrasi *tools* yang ada pada gambar 2.3.

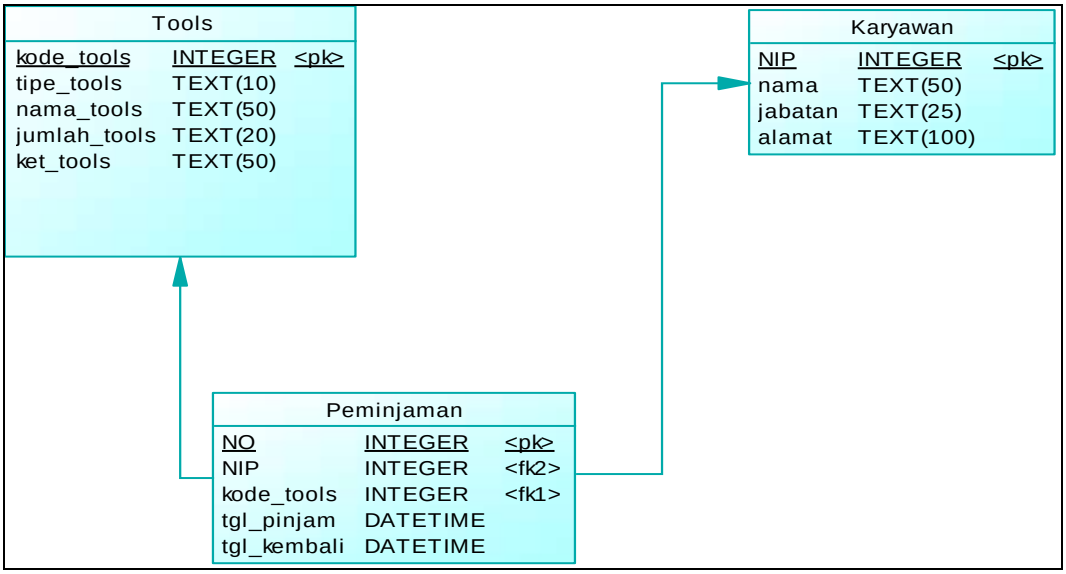


Gambar 2.7. CDM (*Conceptual Data Model*)

2.5. PDM (*Physical Data Model*)

Model relasional atau PDM (*Physical Data Model*) adalah model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data. Setiap tabel mempunyai sejumlah kolom dimana setiap kolom memiliki nama yang unik beserta tipe datanya. (M.Shalahudin, 2014)

PDM merupakan konsep yang menjelaskan detail dari proses bagaimana data disimpan didalam basis data. PDM sudah merupakan bentuk fisik perancangan basis data yang sudah siap diimplementasikan ke dalam DBMS sehingga nama tabel juga sudah merupakan nama asli tabel yang akan diimplementasikan ke dalam DBMS. Berikut merupakan contoh PDM pada studi kasus sistem informasi administrasi *tools* yang ada pada gambar 2.8.



Gambar 2.8. PDM (*Physical Data Model*)

2.6. *Microsoft Access*

Microsoft Access adalah suatu program aplikasi basis data komputer relasional yang digunakan untuk merancang, membuat dan mengolah berbagai jenis data dengan kapasitas yang besar. Aplikasi ini menggunakan mesin basis data *Microsoft Engine* dan juga menggunakan tampilan grafis yang intuitif sehingga memudahkan pengguna. *Microsoft Access* dapat menggunakan data yang disimpan didalam format *Microsoft Access*, *Microsoft Jet Database Engine*, *Microsoft SQL Server*, *Oracle Database* atau semua kontainer basis data yang mendukung ODBC. Komponen-komponen yang terdapat dalam *Microsoft Access* (Reddit, wacanakomputer.wordpress.com) adalah:

1. Tabel

   Tabel adalah objek utama dalam database yang digunakan untuk menyimpan sekumpulan data sejenis dalam sebuah objek. Tabel terdiri dari *field name* dan *record*.

2. *Query* (*SQL* atau *Structured Query Language*)

   Query adalah bahasa untuk melakukan menipulasi terhadap *database* yang digunakan untuk menampilkan, mengubah dan menganalisa sekumpulan data. *Query* dibedakan menjadi 2, yaitu:

   a. DDL (*Data Definition Language*), digunakan untuk membuat atau mendefinisikan objek-objek *database* seperti membuat tabel, relasi antar tabel dan sebagainya.

   b. DML (*Data Manipulation Language*), digunakan untuk menipulasi *database*, seperti menambah, mengubah atau menghapus data, serta mengambil informasi yang diperlukan dari *database*.

3. *Form*

   Form digunakan untuk mengontrol proses masukan data (*input*), menampilkan data (*output*), memeriksa dan memperbaharui data.

4. *Report*

   Report digunakan untuk menampilkan data yang sudah dirangkum dan mencetak data secara efektif.

   Terdapat tipe-tipe data dalam *Microsoft Access*, diantaranya adalah:

   a. *Text*

      Text digunakan untuk *field alfanumeric*, misalkan untuk menulis data biasa seperti alamat rumah, nomor telepon dan kode pos. Biasanya dalam *text* terdapat 225 karakter pada tiap *field*.

b. *Memo*

Memo dapat menampung 64000 karakter untuk tiap *field*, namun tidak dapat diurutkan.

c. *Number*

Number digunakan untuk menyimpan data *numeric* yang akan digunakan untuk proses perhitungan matematis.

d. *Date/Time*

Date/Time biasanya digunakan untuk *field* waktu atau tanggal pada sebuah data.

e. *Currency*

Currency digunakan untuk menentukan *field* pada nilai suatu *numeric* atau untuk menentukan nilai mata uang.

f. *Auto Number*

g. *Yes/No*

h. OLE *Object*

OLE *Object* digunakan untuk eksternal objek, seperti bitmap atau *file* suara.

i. *Hyperlink*

Hyperlink digunakan untuk menghubungkan antara satu *database* dengan *database* lainnya.

j. *Lookup Wizard*

Jika menggunakan tipe data ini untuk sebuah *field*, maka bisa memilih sebuah nilai dari tabel lain atau dari sebuah daftar nilai yang ditampilkan dalam *combo box*.

k. Tabel *relationship*

Tabel *relationship* adalah relasi atau hubungan antara beberapa tabel. Relasi antar tabel dihubungkan oleh *primary key* dan *foreign key* untuk membuat *relationship* maka masing-masing tabel harus memiliki *perimary key* dan *foreign key* untuk dapat menghubungkan antara tabel induk dengan tabel anak. Diperlukan teknik normalisasi terlebih dahulu sebelum membuat *relationship* antar tabel. Normalisasi merupakan proses pengelompokkan data elemen menjadi tabel yang menunjukkan entitas dan relasinya. *Primary key* yaitu suatu atribut yang tidak hanya mengidentifikasi secara unik suatu kejadian tetapi juga mewakili setiap kejadian dari suatu entitas. Sedangkan *foreign key* yaitu atribut yang melengkapi *relationship* dan menunjukkan hubungan antara tabel induk dengan tabel anak, *foreign key* ditempatkan pada tabel anak.