

## **BAB III**

### **LANDASAN TEORI**

#### **3.1 Sistem**

Menurut Hartono (1999 : 23) pada bukunya yang berjudul Analisis dan Desain Sistem Informasi, menyebutkan bahwa sistem merupakan kumpulan dari elemen-elemen yang satu dengan yang lain berinteraksi dan bersama-sama beroperasi untuk mencapai tujuan tertentu. Sistem mempunyai peran yang sangat besar dalam menentukan berjalan tidaknya suatu lembaga atau perusahaan. Hal ini dikarenakan setiap perusahaan akan selalu berdasarkan pada suatu sistem dalam menjalankan aktifitas sehari-harinya.

Suatu sistem dapat dirumuskan sebagai suatu totalitas himpunan yang terdiri dari bagian-bagian yang mana antara satu dengan yang lainnya saling berinteraksi dan bersama-sama beroperasi guna mencapai suatu tujuan tertentu didalam suatu lingkungan. Bagian-bagian atau subsistem tersebut merupakan suatu kompleksitas tersendiri, tapi dalam kebersamaan mencapai suatu tujuan berlangsung secara harmonis dalam keteraturan yang pasti.

#### **3.2 Sistem Informasi**

Sistem adalah sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai tujuan (McLeod, 2001:11), Informasi sendiri merupakan data yang telah diproses, atau data yang memiliki arti (McLeod, 2001:15).

Dari uraian diatas maka definisi Sistem Informasi adalah merupakan sekelompok elemen yang telah terintegrasi dengan maksud untuk mencapai suatu tujuan berdasarkan data yang telah diproses.

Menurut terjemahan Edwards (2001:11) ada beberapa cara berbeda untuk mengklarifikasikan sistem, yaitu melalui :

1. Tingkat Formalitasnya.
2. Tingkat penerapan otomasi yang diberikan.
3. Hubungan sistem tersebut dalam pembuatan keputusan.
4. Sifat-sifat input, proses dan outputnya.
5. Sumber dan tingkat kesesuaiannya.
6. Bobot sistem pada perusahaan.

### **3.3 Sistem Informasi Persediaan Obat**

Setiap perusahaan dagang perlu memiliki persediaan untuk menjamin kelangsungannya. Hal itu perlu dilakukan dengan menginvestasikan sejumlah uang ke dalamnya. Mereka harus mampu mempertahankan jumlah persediaan optimum untuk menjamin kebutuhan bagi kemajuan kegiatan perusahaan, baik secara kuantitas maupun kualitas.

Persediaan pada umumnya merupakan salah satu jenis aktiva lancar yang jumlahnya cukup besar dalam suatu perusahaan. Hal ini mudah dipahami karena persediaan merupakan faktor penting dalam menentukan kelancaran operasi perusahaan. Persediaan merupakan bentuk investasi, dari mana keuntungan (laba) itu bisa diharapkan melalui penjualan di kemudian hari.

Oleh sebab itu pada kebanyakan perusahaan sejumlah minimal persediaan harus dipertahankan untuk menjamin kontinuitas dan stabilitas penjualannya.

Pengertian persediaan menurut beberapa ahli antara lain sebagai berikut :

Menurut Zaki Badridwan (2000:149), menerangkan bahwa :

“Pengertian persediaan barang secara umum istilah persediaan barang dipakai untuk menunjukkan barang- barang yang dimiliki untuk dijual kembali atau digunakan untuk memproduksi barang-barang yang akan dijual”.

Menurut KepMenkes No.1027/MENKES/SK/IX/2004, apotek adalah tempat tertentu tempat dilakukannya pekerjaan kefarmasian dan penyaluran sediaan kefarmasian, perbekalan kesehatan lainnya kepada masyarakat.

Menurut PerMenkes 917/MENKES/per/x/1993, obat (jadi) adalah sediaan atau paduan- paduan yang siap digunakan untuk mempengaruhi atau menyelidiki secara fisiologis atau keadaan patologi dalam rangka penetapan diagnosa , pencegahan, penyembuhan, pemulihan, peningkatan kesehatan dan kontrasepsi.

Dalam sebuah sistem informasi persediaan obat, apotek pada umum memiliki struktur organisasi seperti: Kepala Apotek, bagian penjualan/ kasir, bagian apoteker, bagian gudang, dan bagian pembelian.

### **3.4 Database**

Database adalah suatu sistem menyusun dan mengelola *record-record* menggunakan komputer untuk menyiapkan atau merekam serta memelihara data operasional lengkap dengan sebuah organisasi/perusahaan sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakai untuk proses

pengambilan keputusan (Marlinda,2004:1). Database dapat dinyatakan sebagai suatu sistem yang memiliki karakteristik seperti berikut :

1. Merupakan suatu kumpulan interaksi data yang disimpan bersama dan tanpa mengganggu satu sama lain atau membentuk duplikat data.
2. Kumpulan data di dalam database dapat digunakan oleh sebuah program secara optimal.
3. Penambahan data baru, modifikasi dan pengambilan kembali dari data dapat dilakukan dengan mudah dan terorganisasi.

Dalam arsitektur database terdapat tiga tingkatan yang saling mendukung.

Dibawah ini adalah penjelasannya yaitu :

1. *Internal level* yaitu tingkat yang basis datanya secara fisik ditulis atau disimpan di media storage dan level yang berkaitan.
2. *External level* disebut juga individual user view, yaitu tingkat yang basis datanya dapat berdasarkan kebutuhan masing-masing aplikasi di user atau level yang berkaitan dengan para pemakai.
3. *Conceptual level* disebut juga community user view, yaitu tingkat user view dari aplikasi yang berbeda digabungkan sehingga menggunakan basis data secara keseluruhan dengan menyembunyikan penyimpanan data secara fisik yang merupakan penghubung dari *Internal level* dan *External level*.

Seluruh operasi yang dilakukan pada database didasarkan atas tabel-tabel dan hubungannya. Dalam model relasional dikenal antara lain *table*, *record*, *field*, *index*, *query*. Penjelasannya seperti dibawah ini :

1. *Table* atau *entity* dalam model relasional digunakan untuk mendukung antar muka komunikasi antara pemakai dengan profesional komputer.
2. *Record* atau baris atau dalam istilah model relasional yang formal disebut *tuple* adalah kumpulan data yang terdiri dari satu atau lebih.
3. *Field* atau kolom atau dalam istilah model relasional yang formal disebut *attribute* adalah sekumpulan data yang mempunyai atau menyimpan fakta yang sama atau sejenis untuk setiap baris pada *table*.
4. *Index* merupakan tipe dari suatu *table* tertentu yang bersis nilai-nilai *field* kunci atau *field*.
5. *Query* merupakan sekumpulan perintah *Structure Query Language (SQL)* yang dirancang untuk memanggil kelompok *record* tertentu dari satu tabel atau lebih untuk melakukan operasi pada tabel.

### 3.5 Visual Basic.NET 2005

Visual Basic (VB) 2005 merupakan bahasa pemrograman yang terdapat dalam satu paket aplikasi Visual Studio 2005. Visual Studio 2005 merupakan suatu produk Microsoft yang merupakan penerus dari Visual Studio 2003.

Budiharto (2006:1) menyebutkan, “ Visual Basic 2005 ialah bahasa pemrograman terbaru yang memudahkan programmer VB 6/ VB.Net beralih ke VB 2005”. Budiharto (2006:3-4) juga menyebutkan alasan penting lainnya untuk melakukan migrasi ke VB 2005, yaitu :

1. Visual Basic 2005 mengatasi semua masalah yang sulit di sekitar pengembangan aplikasi berbasis windows dan mengurangi penggunaan

aplikasi lainnya serta versi komponen, bahkan mewarisi sifat C++ dan berbau Java.

2. Visual Basic 2005 memiliki fasilitas penanganan bug yang hebat dan real time background compiler yang mengakibatkan developer visual C# dapat mengetahui kesalahan kode yang terjadi secara up-to-date.
3. Windows Form Designer memungkan developer memperoleh aplikasi desktop dalam waktu yang singkat.
4. Bagi developer, Visual Basic 2005 menyediakan model pemrograman data akses ActiveX data Object (ADO) yang sudah dikenal dan diminati, ditambah XML baru yang berbasis Microsoft ADO.Net dengan ADO.Net, developer akan memperoleh akses ke komponen yang lebih powerfull, seperti control DataSet.
5. Visual Basic 2005 menghasilkan web. Menggunakan form web yang baru, anda dapat dengan mudah membangun thin-client aplikasi berbasis web.
6. .NET Framework secara mendasar dibuat untuk dipasangkan dengan windows 2003 dengan keunggulan memonitor kelalaian dari aplikasi yang sedang berjalan, dan mmengisolasi setiap aplikasi yang sedang berjalan.
7. Developer dengan berbagai latar belakang dapat dengan segera menguasai .NET karena kemudahan dan kemiripan kode yang ditawarkannya.
8. Deployment/penyebaran yang mudah, baik untuk aplikasi windows maupun aplikasi web karena sudah tersedia wizard atau tool secara khusus dengan fasilitas tambahan yang menarik. Tool canggih ini tidak tersedia pada aplikasi sebelumnya.

9. Integrasi dengan sistem yang sudah ada sangat mudah, NET Framework com memungkinkan anda berinteraksi dengan sistem yang sudah ada menggunakan XML web service.
10. Mendukung lebih dari 20 bahasa pemrograman yang tidak terbayang sebelumnya.

### **3.6 Microsoft SQL Server 2005**

1. SQL Server 2005 merupakan produk dari Microsoft dalam bidang Relational Database Management System (RDBMS) yang didesain untuk mendukung proses transaksi yang besar. SQL Server 2005 dapat dijalankan pada Windows 2000 profesional service pack 4, Windows 2000 service pack 4, windows XP profesional service pack 2, atau windows 2003 service pack 1 (Budiharto,2006:21). Budiharto (2006:21) juga menyebutkan SQL Server 2005 membutuhkan windows installer 3.1 yang dapat diperoleh pada saat instalasi Visual Studio 2005.
2. SQL Server 2005 terdiri atas tujuh edisi berbeda yang tersedia pada CD yang berbeda, yaitu Standard Edition, Enterprise Edition, Personal Edition, Developer Edition, Windows CE Edition, Evaluation edition dan Microsoft desktop Engine (MDE). SQL Server 2005 mempunyai fasilitas tambahan yang membuat software tersebut memiliki kemampuan penuh dalam e-commerce. SQL Server 2005 secara otomatis akan menginstall enam database utama, yaitu master, model, tempdb, pubs, Northwind dan msdb. (Yuswanto, 2005:10).

### 3.7 Entity Relationship Diagram (ERD)

*Entity relationship diagram* menurut Hartono (1999:11) adalah suatu bentuk perencanaan *database* secara konsep fisik yang nantinya akan dipakai sebagai kerangka kerja dan pedoman dari struktur penyimpanan data. ERD digunakan untuk menggambarkan model hubungan data dalam sistem, dimana di dalamnya terdapat hubungan entitas beserta atribut relasinya dan mendokumentasikan kebutuhan-kebutuhan untuk sistem pemrosesan data. ERD memiliki beberapa jenis model yaitu :

Tabel 3.2 Jenis ERD

NO	JENIS ERD	KETERANGAN
1.	<i>Conceptual Data Model (CDM)</i>	Merupakan model <i>universal</i> dan dapat menggambarkan semua struktur logic <i>database</i> (DBMS), dan tidak bergantung dari <i>software</i> atau pertimbangan struktur <i>data storage</i> . Sebuah CDM dapat diubah langsung menjadi PDM.
2.	<i>Physical Data Model (PDM)</i>	Merupakan model ERD yang mengacu pada pemilihan <i>software</i> DBMS yang spesifik. Hal ini seringkali berbeda secara signifikan dikarenakan oleh struktur tipe <i>database</i> yang bervariasi,



		dari model schema, tipe data penyimpanan dsb.
--	--	---

ERD memiliki 4 jenis obyek, yaitu :

### 1. *Entity*

Sesuatu yang ada dan terdefiniskan bisa berupa nyata maupun abstrak yang dapat dibedakan satu dengan yang lainnya dan adanya hubungan saling ketergantungan. Ada 2 macam tipe *entity*, yaitu :

#### a. *Strong Entity*

*Strong Entity* merupakan tipe *entity* yang mempunyai *key attribute* untuk setiap individu yang ada didalamnya.

#### b. *Weak Entity*

*Strong Entity* merupakan *entity* yang tidak memiliki *key atribut*, oleh karena itu *weak entity* harus dihubungkan dengan *strong entity* untuk menggunakan atribut kunci secara bersama-sama.

### 2. *Attribute*

Setiap *entity* memiliki beberapa *attribute*, yang merupakan ciri atau karakteristik dari *entity* tersebut. *Attribute* sering disebut juga data elemen atau *data field*.

### 3. *Key*

Beberapa elemen data memiliki sifat, dengan mengetahui nilai yang telah diberikan oleh sebagian elemen data dari *entity* tertentu, dapat

diidentifikasi nilai-nilai yang terkandung dalam elemen-elemen data lain ada *entity* yang sama. Elemen penentu tersebut adalah sebagai elemen data kunci (*key*).

#### 4. *Relationship*

*Relationship* menggambarkan hubungan yang terjadi antar *entity* yang mewujudkan pemetaan antar *entity*. Bentuk *relationship* yaitu :

a. *One to One Relationship*

Hubungan satu *entity* dengan satu *entity* yang lain.

b. *One to Many Relationship*

Hubungan antar *entity* satu dengan *entity* yang lainnya adalah satu berbanding banyak.

### 3.8 Data Flow Diagram(DFD)

*Data Flow Diagram* atau yang sering disebut *Bubble Chart* atau diagram menurut Budiharto (2006:1), model proses, digram alur kerja atau model fungsi adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alir data baik secara manual maupun komputerisasi. DFD merupakan alat pembuat model yang sering digunakan untuk menjelaskan aliran informasi dan transformasi data yang bergerak dari pemasukan data hingga keluaran. Untuk memudahkan proses pembacaan DFD, maka penggambaran DFD disusun berdasarkan tingkatan atau *level* dari atas ke bawah, yaitu :

1. *Context Diagram*

Merupakan diagram paling atas yang terdiri dari suatu proses dan menggambarkan ruang lingkup proses. Hal yang digambarkan dalam diagram konteks adalah hubungan terminator dengan sistem dan juga sistem dalam suatu proses. Sedangkan hal yang tidak digambarkan dalam *Context Diagram* adalah hubungan antar *terminator* dan *data source*.

2. *Diagram Zero (Level 0)*

Merupakan diagram yang berbeda diantara diagram konteks dan diagram detail serta menggambarkan proses utama dari DFD. Hal yang digambarkan dalam *Diagram Zero* adalah proses utama dari sistem serta hubungan *entity*, proses, alur data dan *data source*.

3. *Diagram Detail (Primitif)*

Merupakan penguraian dalam proses yang ada dalam *Diagram Zero*. Diagram yang paling rendah dan tidak dapat diuraikan lagi.

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

1. Terminator

*Terminator* atau *External Entity* atau kesatuan luar mewakili entitas *external* yang berkomunikasi dengan sistem yang dikembangkan. *Terminator* merupakan kesatuan di lingkungan sistem yang dapat berupa orang atau sistem yang berada di lingkungan luar sistem yang memberikan inputan maupun yang menerima output dari sistem serta berupa bagian atau divisi diluar sistem yang berkomunikasi dengan sistem. Terminator ini sering juga disebut entitas (*external*), sumber atau tujuan (*source and sink*).

## 2. Proses

Proses sering dikenal dengan *Bubble*, fungsi atau informasi. Komponen proses menggambarkan bagian dari sistem yang mentransformasikan input ke output, atau dapat dikatakan bahwa komponen proses menggambarkan transformasi satu inputan atau lebih menjadi output dari sistem. Dilambangkan dengan lingkaran atau empat persegi panjang tegak dengan sudut tumpul. Proses diberi nama untuk menerangkan proses atau kegiatan apa yang sedang atau kegiatan yang akan dilaksanakan. Dan setiap proses harus diberi penjelasan lengkap sebagai berikut :

### a. Identifikasi Proses

Umumnya berupa angka yang menunjukkan nomor dari proses dan ditulis pada bagian atas simbol.

### b. Nama Proses

Menunjukkan apa yang sedang dikerjakan oleh proses tersebut. Nama proses harus jelas dan lengkap menggambarkan bagian prosesnya. Nama proses diletakkan dibawah identifikasi proses.

## 3. Data Store

*Data Store* digunakan sebagai sarana untuk pengumpulan data. *Data Store* disimbolkan dengan dua garis horizontal yang paralel dimana tertutup pada satu ujungnya atau dua garis horizontal. Suatu nama perlu diberikan pada *data store* menunjukkan nama dari filenya. *Data Source* biasanya berkaitan dengan penyimpanan *file* atau *database* yang dilakukan secara terkomputerisasi. *Data*

*Store* dihubungkan dengan alur data hanya pada komponen proses pengertiannya sebagai berikut :

- a. Alur data dari store yang berarti sebagai pengaksesan data untuk suatu proses.
- b. Alur data ke proses berarti meng-*update* data seperti menambah data, mengurangi data maupun mengubah data.

#### 4. Alur Data

Alur data dapat 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.

#### 5. Syarat-syarat sebuah DFD :

- a. Pemberian nama untuk setiap komponen DFD.
- b. Pemberian nomor pada proses DFD.
- c. Penggambaran DFD serapi mungkin.
- d. Menghindari pembuatan DFD yang rumit.
- e. Memastikan DFD dibangun secara konsisten.