

## BAB III

### LANDASAN TEORI

#### 3.1 Sistem Informasi

Menurut Jogiyanto HM (2003), sistem Informasi merupakan suatu sistem yang tujuannya menghasilkan informasi sebagai suatu sistem, untuk dapat memahami sistem informasi, karena lebih baik jika konsep dari sistem itu dipahami terlebih dahulu. Demikian juga sebagai sistem penghasil informasi, maka konsep informasi perlu dipahami terlebih dahulu.

Komponen-komponen dari sistem informasi tidak boleh kurang, karena jika komponennya kurang, maka sistem informasi tersebut tidak akan mencapai tujuannya. Komponen-komponen dari sistem informasi harus tepat jumlah dan macamnya. Komponen sistem informasi di ilustrasikan 5 komponen, yaitu:

##### *a. Hardware*

Menurut Jogiyanto HM (2003), *hardware* atau biasa disebut dalam Bahasa Indonesia “Perangkat Keras” merupakan kumpulan elemen atau komponen fisik yang menyusun suatu sistem komputer. Secara garis besar, Hardware dibagimenjadi tiga kelompok berdasarkan cara kerjanya, yaitu *input* (masukan), *process* (pemrosesan), *output* (keluaran).

##### *b. Software*

Menurut Jogiyanto HM (2003), *software* merupakan sekumpulan dari perintah/fungsi yang ditulis dengan aturan tertentu untuk memerintahkan komputer melaksanakan tugas tertentu.

### c. Data

Menurut Jogiyanto HM (2003), data merupakan komponen dasar dari informasi yang akan diproses lebih lanjut untuk menghasilkan informasi.

Himpunan data tersebut akan bersifat unik, antara lain:

- a) Saling berkaitan (*Interrelated*)
- b) Kebersamaan (*Shared*)
- c) Terkendali (*Controlled*)
- d) Prosedur

Dokumentasi proses sistem, buku penuntun operasional dan teknis. Prosedur menghubungkan berbagai perintah, dan aturan yang akan menentukan rancangan penggunaan aplikasi sistem informasi. *User* dari sistem dan staff akan mengatur serta merancang sistem informasi berdasarkan prosedur-prosedur yang didokumentasikan.

### 3.2 Dashboard

Menurut Malik (2005), *dashboard* dinyatakan dalam beberapa istilah yang berbeda pada pustaka-pustaka yang ada. Shaden Malik menggunakan istilah *enterprise dashboard* yang didefinisikan sebagai sebuah antar muka komputer yang banyak menampilkan bagan, laporan, indikator visual, dan mekanisme alert, yang dikonsolidasikan ke dalam *Platform* informasi yang dinamis dan relevan. *Enterprise dashboard* berperan sebagai *live console* untuk mengelola inisiatif bisnis.

Istilah *information dashboard*, yang didefinisikan sebagai tampilan visual dan informasi penting, yang diperlukan untuk mencapai suatu atau beberapa tujuan, dengan mengkonsolidasikan dan mengatur informasi dalam satu layar (*single screen*), sehingga kinerja organisasi dapat dimonitor secara sekilas. Tampilan visual disini mempunyai pengertian bahwa penyajian informasi harus dirancang sebaik mungkin, sehingga manusia dapat menangkap informasi tersebut dengan cepat dan mudah dipahami maknanya dengan benar.

### 3.3 System Flow

Menurut Sutabri (2012), *system flow* adalah bagan alir sistem menunjukkan arus pekerjaan secara keseluruhan dari sistem. *System flow* menunjukkan sistematika dari prosedur yang ada di dalam sistem dan menunjukkan apa yang dilakukan sistem.

Simbol-simbol yang digunakan dalam *system flow* sebagai berikut:

1. Simbol dokumen

Menunjukkan dokumen input dan output untuk proses manual atau komputer.



Gambar 3.1 Lambang Dokumen

2. Simbol kegiatan manual

Menunjukkan pekerjaan manual yang terdapat pada sistem.



Gambar 3.2 Kegiatan Manual

3. Simbol simpanan *offline*

Menunjukkan file *non*-komputer yang diarsip.



Gambar 3.3 Simpanan Offline

## 4. Simbol proses

Menunjukkan kegiatan proses dari operasi program komputer.



Gambar 3.4 Lambang Proses

5. Simbol *database*

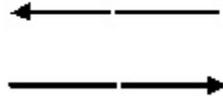
Menunjukkan tempat untuk menyimpan data hasil operasi komputer.



Gambar 3.5 Lambang Database

## 6. Simbol garis alir

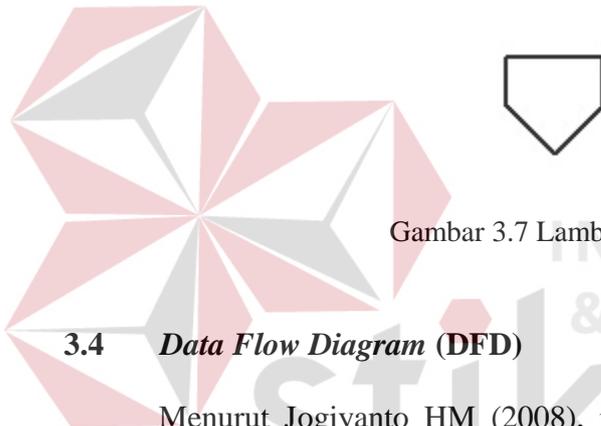
Menunjukkan arus dari proses.



Gambar 3.6 Garis Alir

## 7. Simbol penghubung

Menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain.



Gambar 3.7 Lambang Penghubung

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

Menurut Jogiyanto HM (2008), menyarankan untuk menggunakan *data flow diagram* (DFD) dalam menggambarkan atau membuat model sistem. Meskipun namanya *data flow diagram*, yang seakan-akan mencerminkan penekanan pada data, namun sebenarnya DFD lebih menekankan pada segi proses.

Menurut Sutabri (2012), pengertian secara umum dari *data flow diagram* ini adalah suatu *network* yang menggambarkan suatu sistem automat/komputersasi, manualisasi, atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya. Keuntungan penggunaan DFD adalah memungkinkan untuk menggambarkan sistem dari level yang paling tinggi

kemudian menguraikannya menjadi level yang lebih rendah (dekomposisi). Sedangkan kekurangan penggunaan DFD adalah tidak menunjukkan proses pengulangan (*looping*), proses keputusan, dan proses perhitungan. Simbol-simbol yang digunakan DFD:

a) *External Entity atau Boundary*

Menurut Sutabri (2012), *external entity* atau kesatuan luar merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lain yang berada di lingkungan luar yang akan memberikan input atau menerima output dari sistem. *External entity* disimbolkan dengan notasi kotak.

b) Arus Data

Menurut Sutabri (2012), arus Data (*data flow*) di DFD diberi simbol panah. Arus data ini mengalir di antara proses, simpanan data (*data store*) dan kesatuan luar (*external entity*). Arus data ini menunjukkan arus data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

c) Proses

Menurut Sutabri (2012), suatu proses adalah kegiatan yang dilakukan oleh orang, mesin, atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk menghasilkan arus data yang akan keluar dari proses. Simbol proses berupa lingkaran atau persegi panjang bersudut tumpul.

#### d) Simpanan Data

Menurut Sutabri (2012), simpanan data merupakan simpanan dari data yang dapat berupa hal-hal sebagai berikut, sebagai gambaran:

1. Suatu *file* atau *database* di sistem komputer.
2. Suatu arsip atau catatan manual.
3. Suatu kotak tempat data di meja seseorang.
4. Suatu tabel acuan manual.

Simpanan data di DFD disimbolkan dengan sepasang garis horizontal paralel yang tertutup di salah satu ujungnya.

#### 3.4.1 Context Diagram

Menurut Rosa (2013), *Context Diagram* merupakan langkah pertama dalam pembuatan DFD. Pada *context diagram* dijelaskan sistem apa yang dibuat dan *external entity* apa saja yang terlibat. Dalam *context diagram* harus ada arus data yang masuk dan arus data yang keluar.

#### 3.4.2 Data Flow Diagram Level 0

Menurut Rosa (2013), DFD *level 0* adalah langkah selanjutnya setelah *context diagram*. Pada langkah ini, digambarkan proses-proses yang terjadi dalam sistem informasi.

#### 3.4.3 Data Flow Diagram Level 1

Menurut Rosa (2013), DFD *Level 1* merupakan penjelasan dari DFD *level 0*. Pada proses ini dijelaskan proses apa saja yang dilakukan pada setiap proses yang terdapat di DFD *level 0*.

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

Menurut Rosa (2013), *Conceptual Data Model (CDM)* 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. CDM merupakan hasil penjabaran lebih lanjut dari ERD.

### **3.6 *Physical Data Model (PDM)***

Menurut Rosa (2013), model relasional atau *Physical Data Model (PDM)* adalah model yang menggunakan sejumlah table untuk menggambarkan data serta hubungan antara data. Setiap table mempunyai sejumlah kolom di mana setiap kolom memiliki nama yang unik beserta tipe datanya. PDM merupakan konsep yang menerangkan detail dari bagaimana data di simpan di dalam basis data. PDM sudah merupakan bentuk fisik perancangan basis data yang sudah siap diimplementasikan ke dalam DBMS sehingga nama *table* juga sudah merupakan nama asli *table* yang diimplementasikan ke dalam DBMS.

### **3.7 *Entity Rational Diagram***

Menurut Rosa (2013), *Entity Relational Diagram (ERD)* merupakan penggambaran hubungan antara beberapa *entity* yang digunakan untuk merancang *database* yang akan diperlukan.

### **3.8 *Konsep Dasar Basis Data***

Menurut Rosa (2013), basis data adalah kumpulan data (*elemen*) yang secara logika berkaitan dalam merepresentasikan fenomena. Fakta secara

terstruktur dalam *domain* tertentu untuk mendukung aplikasi pada sistem tertentu. Data yang ada biasanya saling terhubung untuk merefleksikan fakta-fakta yang terdapat di organisasi.

### 3.8.1 Sistem Basis Data

Menurut Rosa (2013), Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.

Kebutuhan basis data dalam sistem informasi meliputi:

- a. Memasukkan, menyimpan, dan mengambil data.
- b. Membuat laporan berdasarkan data yang telah disimpan. (Rosa, 2013)

#### 3.8.1.1 Kelebihan Sistem Basis Data

1. Pengendalian terhadap redudansi data.
2. Mencegah ketidakkonsistenan data.
3. Keamanan data dapat terjaga, yaitu data dapat dilindungi dari pemakai yang tidak berwenang.
4. Integritas data dapat dipertahankan.
5. Data dapat dipergunakan bersama-sama.
6. Menyediakan *recovery*.
7. Memudahkan penerapan standarisasi.
8. Data bersifat mandiri (*data independence*).

9. Keterpaduan data terjaga, memelihara keterpaduan data berarti data harus akurat. Hal ini sangat erat hubungannya dengan pengontrolan kerangkapan data dan pemeliharaan keselarasan data. (Rosa,2013)

### 3.8.1.2 Kekurangan Sistem Basis Data

- 1 Diperlukan tempat penyimpanan yang besar.
- 2 Diperlukan tenaga yang terampil dalam mengolah data.
- 3 Kerusakan sistem basis data dapat mempengaruhi departemen yang terkait.
- 4 Kompleksitas yang tinggi
- 5 Ongkos konversi dari sistem yang lama ke sistem baru. (Rosa,2013)

## 3.9 Database Management System

Menurut Rosa (2013), DBMS (*Database Management System*) atau dalam bahasa Indonesia sering disebut sebagai Sistem Manajemen Basis Data adalah suatu sistem aplikasi yang digunakan untuk menyimpan, mengelola, dan menampilkan data. Suatu sistem aplikasi disebut DBMS jika memenuhi persyaratan minimal sebagai berikut:

- a. Menyediakan fasilitas untuk mengelola akses data
- b. Mampu menangani integritas data
- c. Mampu menangani akses data yang dilakukan secara terus menerus
- d. Mampu menangani *backup* data

### 3.9.1 Bahasa-Bahasa yang Terdapat Dalam DBMS

#### 1 *Data Definition Language* (DDL)

Pola skema basis data di spesifikasikan dengan satu set definisi yang di ekspresikan dengan satu bahasa khusus yang disebut DDL. Hasil

kompilasi perintah DDL adalah satu set tabel yang disimpan di dalam *file* khusus yang disebut data *dictionary/directory*. (Betha Sidik, 2005)

## 2 *Data Manipulation Language* (DML)

Bahasa yang memperbolehkan pemakai mengakses atau memanipulasi data sebagai yang diorganisasikan sebelumnya model data yang tepat. (Betha Sidik, 2005)

## 3 *Query*

Pernyataan yang diajukan untuk mengambil informasi. Merupakan bagian DML yang digunakan untuk pengambilan informasi. (Betha Sidik, 2005)

### 3.9.2 Fungsi DBMS

#### 1. *Data Definition*

DBMS harus dapat mengolah data *definition* atau pendefinisian data.

#### 2. *Data Manipulation*

DBMS harus dapat menangani permintaan-permintaan dari pemakai untuk mengakses data.

#### 3. *Data Security dan Integrity*

DBMS dapat memeriksa *security* dan *integrity* data yang didefinisikan oleh DBA.

#### 4. *Data Recovery dan Concurrency*

- a. DBMS harus dapat menangani kegagalan-kegagalan pengaksesan basis data yang dapat disebabkan oleh kesalahan sistem, kerusakan *disk*, dan sebagainya.

- b. DBMS harus dapat mengontrol pengaksesan data yang konkuren yaitu bila satu data diakses secara bersama-sama oleh lebih dari satu pemakai pada saat yang bersamaan.

#### 5. *Data Dictionary*

DBMS harus menyediakan *data dictionary* atau kamus data. (Betha Sidik, 2005)

### 3.10 *Tools Program*

Menurut Rosa (2013), Dalam perancangan sebuah sistem informasi, pastinya membutuhkan sebuah alat bantu berupa bahasa pemrograman untuk mensejajarkan bahasa dengan komputer. *Tools* dalam bahasa pemrograman yang dipakai saat ini adalah XAMPP yang didalamnya terdapat *MySQL* serta menggunakan *notepad ++*.

#### 3.10.1 *Xampp*

Menurut Betha (2005), Xampp adalah sebuah aplikasi yang berisi *MySQL* dan *Apache*. Aplikasi ini dapat membantu untuk membangun sebuah aplikasi berbasis web tanpa harus menginstal *MySQL* dan *Apache* secara terpisah/sendiri-sendiri.

#### 3.10.2 *MySql*

Menurut Betha (2005), *MySQL* merupakan software sistem manajemen database yang sangat populer di kalangan pemrograman web, terutama dilingkungan *Linux* dengan menggunakan script *PHP* dan *Perl*. Software database ini kini telah tersedia juga pada platform sistem operasi windows (98/ME hingga windows 8).

MySQL merupakan database yang paling populer digunakan untuk membangun aplikasi web yang menggunakan database sebagai sumber dan pengelola datanya. Kepopuleran MySQL dimungkinkan karena kemudahannya untuk digunakan, cepat secara kinerja query, dan mencukupi untuk kebutuhan database perusahaan-perusahaan skala menengah kecil. MySQL merupakan database yang digunakan oleh situs-situs terkemuka di internet untuk menyimpan datanya.

### 3.10.3 Apache

Menurut Betha (2005), Apache merupakan sebuah web *server* yang digunakan karena handal dan stabil, hampir semua web *master* menggunakan apache karena perkembangan dari *server* ini cepat menyesuaikan dengan kebutuhan maupun perkembangan zaman. Cara instalasi juga sangat mudah tanpa harus konfigurasi yang rumit.

### 3.10.4 Notepad++

Menurut Rosa (2013), notepad++ adalah aplikasi gratisan pengembangan dari Notepad. Notepad++ ini juga mempunyai fitur yang sangat memudahkan kita membedakan alur syntax yaitu fitur highlighting. Fitur ini akan menandai sintaks dan variable yang di gunakan dalam source code.

Lainnya, tersedianya fitur tab yang dapat membantu anda mengelola beberapa kode dalam waktu yang bersamaan.