

BAB 2

KAJIAN TEORI

2.1 Pengertian Sistem Informasi

Pengertian dari sistem informasi merupakan gabungan pengertian dari sistem dan informasi :

Pengertian sistem menurut para ahli :

1. Menurut Mustakini (2009:34), bahwa Sistem (system) dapat didefinisikan dengan pendekatan prosedur dan dengan pendekatan komponen. “Dengan pendekatan prosedur, sistem dapat didefinisikan sebagai kumpulan dari prosedur-prosedur yang mempunyai tujuan tertentu” . Contoh sistem yang didefinisikan dengan pendekatan ini adalah sistem akuntansi. Sistem ini didefinisikan sebagai kumpulan dari prosedur-prosedur penerimaan kas, pengeluaran kas, penjualan, pembelian dan buku besar [1]
2. Menurut Andri Kristanto (2008:1) “Sistem merupakan jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu”. [2]

Dari pengertian sistem diatas dapat kita simpulkan bahwa Sistem adalah suatu kumpulan peraturan yang saling berhubungan untuk digunakan dalam menyelesaikan suatu kegiatan atau sasaran.

Pengertian informasi menurut para ahli :

1. Menurut Prasajo dan Riyanto, “bahwa pengertian informasi sering disamakan dengan pengertian data. Data adalah sesuatu yang belum diolah dan belum dapat digunakan sebagai dasar yang kuat dalam pengambilan keputusan”. Beberapa contoh data adalah data nama mahasiswa, jumlah kursi, jumlah siswa, dan lain-lain. Data mahasiswa relatif belum berarti, jika digunakan untuk mengambil keputusan tertentu. Data nama mahasiswa ditambah data IPK mahasiswa, dan persentase nilai “D” dapat digunakan untuk menentukan bahwa mahasiswa tersebut dapat mengambil

bebas teori atau tidak. Data bebas teori dan nilai skripsi dapat digunakan untuk mengambil keputusan bahwa mahasiswa tersebut berhak lulus atau tidak. Gabungan dari data nama mahasiswa, IPK, persentasi nilai “D”, nilai skripsi merupakan sebuah informasi”. [3]

2. Menurut Nova, “bahwa informasi dapat di produksi dan dipasarkan sebagai sebuah produk, pada dasarnya informasi merupakan suatu yang diproduksi dan didistribusikan, baik oleh sebuah lembaga pendidikan, radio, televisi, penerbit buku, koran dan majalah. Ketidak akuratan informasi akan menyebabkan perusahaan yang bergerak dibidang informasi dapat kehilangan reputasi dan kredibilitasnya”. [4]

Dari pengertian diatas dapat kita simpulkan bahwa informasi itu adalah kumpulan data yang telah dipelajari atau telah diproses dan siap diberikan bagi penerima maupun pemberi informasi.

Pengertian sistem informasi menurut para ahli :

1. Pengertian sistem informasi menurut Laudon, Kenneth, Jane, “Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, pihak luar tertentu dengan laporan-laporan yang diperlukan”. [5]
2. Menurut O’brian dalam Yakub bahwa “sistem informasi (*information system*) merupakan kombinasi teratur dari orang-orang, perangkat keras (*hardware*), perangkat lunak (*software*), jaringan komunikasi, dan sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi”. [6]

Dari pengertian diatas dapat kita uraikan bahwa sistem informasi adalah suatu sistem yang digunakan dalam mengumpulkan informasi dari semua sumber, dan transaksi-transaksi yang dilakukan dalam suatu bidang bisnis.

2.2 Flowchart(Bagan Alir)

Flowchart adalah baga(*chart*) yang menunjukkan alir(*flow*) di dalam suatu program atau prosedur sistem secara logika. [7]

Dari pengertian menurut Jogiyanto, dapat kita uraikan bahwa *flowchart* adalah suatu bagan air yang digunakan untuk menggambarkan sebuah kerangka kerja atau proses bisnis secara *pyshical* didalam suatu perusahaan.

2.3 ERD (*EntityRelationshipDiagram*)

Pengertian *ERD(entity relationship diagram)* menurut para ahli :

1. Menurut Brady dan Loonam (2010), "*Entity Relationship Diagram (ERD)* merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh sistem analisis dalam tahap analisis persyaratan proyek pengembangan sistem. Sementara seolah-olah teknik diagram atau alat peraga memberikan dasar untuk desain sebuah *database* relasional yang mendasari sistem informasi yang dikembangkan. *ERD* bersama-sama dengan detail pendukung merupakan model data yang gilirannya digunakan sebagai spesifikasi untuk database". [8]
2. Menurut Whitten dan Bentley (2007), "*ERD* adalah sebuah model konseptual yang mendeskripsikan hubungan antar penyimpanan dalam *DFD (Data Flow Diagram)*. *ERD* digunakan untuk memodelkan struktur data dan hubungan antar data. Dengan *ERD* model dapat diuji dengan mengabaikan proses yang harus dilakukan. Selain itu dengan *ERD* kita akan dapat menjawab pertanyaan mengenai data apa yang kita perlukan serta bagaimana data yang satu berhubungan dengan data yang lain". [9]

ERD memiliki 4 komponen yaitu :

1. Entitas
Entitas adalah objek dalam dunia nyata yang dapat dibedakan dengan objek lain.
2. Atribut
Atribut adalah setiap entitas pasti mempunyai elemen yang disebut dengan atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dari atribut mempunyai sesuatu yang dapat

mengidentifikasi isi elemen satu dengan yang lain. Gambar atribut diwakili oleh simbol elips.

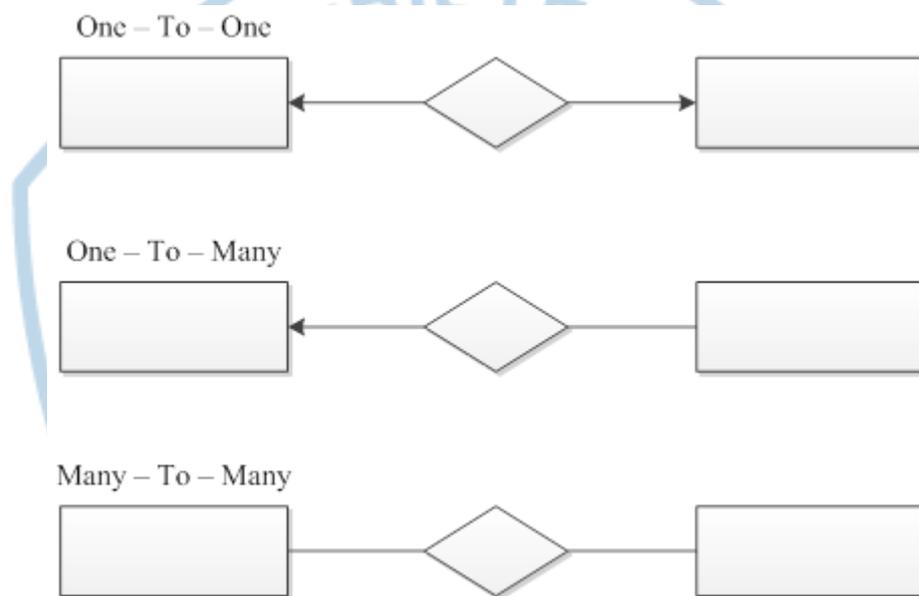
3. Relasi

Relasi adalah hubungan antara beberapa himpunan entitas.

4. Arus Data atau Kardinalitas

Arus data adalah simbol yang menunjukkan arah, dimana dapat diartikan hubungan yang dimiliki antara suatu entitas dengan entitas lainnya.

Berikut adalah hubungan antar relasi :



Gambar 2.1 ERD Hubungan antar relasi

Dari pengertian para ahli diatas dapat kita uraikan bahwa *ERD (Entity relationship diagram)* adalah suatu diagram yang digunakan dalam sebuah bidang bisnis yang menggambarkan aktivitas-aktivitas yang sedang berjalan.

2.4 Implementasi Tabel

1. Menurut Brady dan Loonam, "Implementasi merupakan perluasan aktivitas yang saling menyesuaikan atau implementasi sebagai evaluasi. [8]"

2. Menurut Whitten dan Bentley, “Implementasi adalah bermuara pada aktivitas, aksi, tindakan, atau adanya mekanisme suatu sistem. Implementasi bukan sekedar aktivitas, tetapi suatu kegiatan yang terencana dan untuk mencapai tujuan kegiatan”. [9]

Dari pengertian para ahli di atas bisa kita uraikan bahwa implementasi adalah suatu aktivitas yang dilakukan untuk secara luas untuk mencapai suatu tujuan tertentu.

Implementasi tabel adalah sebuah gambaran yang digunakan oleh suatu organisasi dalam memperlihatkan aktivitas-aktivitas yang telah didetailkan dari *ERD*. Implementasi *ERD* to tabel merupakan gambaran aktivitas yang ditransformasikan menjadi tabel dari himpunan entitas dan relasi yang memiliki atribut masing-masing. [7]

2.5 UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah penggabungan dari bahasa pemodelan *object-oriented*. UML divisualisasikan oleh struktur dari sistem *software* yang biasa disebut *4+1 view model* yang terdiri dari :

1. *Use Case View*
2. *Design View*
3. *Implementation View*
4. *Deployment View*
5. *Process View*

Dalam *UML*, sebuah model dipresentasikan sebagai diagram. Diagram adalah tampilan grafik dari kumpulan elemen model. Diagram yang berbeda merepresentasikan informasi yang berbeda, secara tipikal baik struktur maupun *behavior* dari elemen model yang diagram tersebut gambarkan. *UML* menetapkan 9 tipe diagram. *UML (Unified Modeling Language)* adalah Metodologi kolaborasi antara metoda-metoda *Booch*, *OMT (Object Modeling Technique)*, serta *OOSE (Object Oriented Software Engineering)* dan beberapa metoda lainnya, merupakan metodologi yang paling sering digunakan saat ini untuk analisa dan perancangan

sistem dengan metodologi berorientasi objek mengadaptasi maraknya penggunaan bahasa “pemrograman berorientasi objek” (*OOP*). [10]

Berikut adalah contoh tabel *UMLDiagram*

Tabel 2.1 *UML Diagram*

No.	Diagram	View
1	Use case diagram	Use case
2	Object diagram	Use case dan desain
3	Sequence diagram	Use case dan desain
4	Collaboration diagram	Desain
5	Class diagram	Desain
6	Statechart diagram	Desain
7	Activity diagram	Desain
8	Component diagram	Implementasi
9	Deployment diagram	Deployment

2.5.1 Use Case Diagram

Use case diagram menurut Satzinger et al., “*use case* merupakan suatu aktivitas yang dilakukan oleh sistem, biasanya merupakan sebuah respon untuk permintaan dari pengguna sistem”. Satzinger et al menjelaskan bahwa actor tidak selalu sama, karena *use case* merupakan orang yang berinteraksi dengan sistem yang mana sistem harus meresponnya. [11]:

1. *Usecase*
2. *Actor*
3. *Relationship*
4. *Systemboundary*

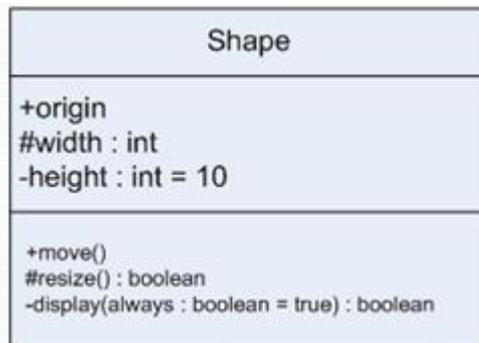
2.5.2 Activity Diagram

Activitydiagram menggambarkan proses bisnis dan urutan aktivitas dalam sebuah proses. *Activitydiagram* dibuat berdasarkan sebuah use case atau beberapa use case pada use case diagram.

2.5.3 Class Diagram

Class diagram menggambarkan struktur dan deskripsi *class*, *package*, dan objek beserta hubungan satu sama lainnya seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class* memiliki 3 area pokok :

1. Nama
2. Atribut



3. Metoda

2.6 Java

Sebagian besar pemrograman modern berdiri di atas class library yang telah ada untuk mendukung fungsionalitas. Pada bahasa *Java*, kelompok-kelompok kelas yang berkaitan erat dimasukkan di satu paket, bervariasi sesuai edisi *Java*. Masing-masing paket diperuntukkan untuk maksud tertentu yaitu , aplikasi standar, skala *enterprise*, dan produk konsumen. *Java* adalah bahasa yang dapat dijalankan di sembarang platform, di beragam lingkungan. *TheJava 2 Platform* tersedia dalam tiga edisi untuk keperluan berbeda berikut :

1. *Java standar edition (JSE)*

JSE menyediakan lingkungan pengembangan yang kaya figure, stabil, aman, dan cross-platform. Edisi ini mendukung konektivitas basis data, rancangan antar muka pemakai, input/output, dan pemrograman jaringan dan termasuk berbagai paket-paket dasar bahasa java.

2. *Java Enterprise Edition (JEE)*

Jee Menyediakan kakas untuk membangun dan menjalankan *multiple tier enterprise applications*. *J2EE* berisi paket-paket di *J2SE* ditambah paket-paket untuk mendukung pengembangan *Enterprisen JavaBeans, Java Servlets Pages,XML* dan kendali transaksi yang fleksibel.

3. Java Micro Edition (JME)

JME digunakan untuk beragam *consumer electronic product*, seperti *pager, smart card, cell phone, handheld PDA*, dan *set-top box*. JME menyediakan bahasa *java* yang sama unggulnya dalam portabilitas (kemampuan dijalankan di manapun) dan *safe network delivery* seperti JSE dan JEE. [12].

2.6.1 Java Programming

Aplikasi java adalah sebuah program computer yang dieksekusi ketika *java command* digunakan untuk menjalankan *Java Virtual Machine (JVM)*. Contoh pemrograman java yang paling sederhana adalah menampilkan teks "Hello World".

```
public class MyFirstJavaProgram {
    /* This is my first java program.
     * This will print 'Hello World' as the output
     */
    public static void main(String []args) {
        System.out.println("Hello World"); // prints Hello World
    }
}
```

Element penting lainnya pada pemrograman java adalah fungsi control yang terdiri dari :

1. Percabangan

a) If else

Statement if mengambil keputusan percabangan berdasarkan nilai *true* atau *false*.

```
If (ekspresi Boolean)
{
    Statement1;
    Statement2;
}
Else
{
    Statement3;
}
```

b) Switch ...case

Selain *if...else* juga terdapat percabangan lain yakni *switch.. case*

```
switch (ekspresi)
{
    Case ekspresi-konstan
    statement
    break atau statemen untuk meloncat ke blok case lain
    [default : statement]
}
```

2. Pengulangan

a) *While*

Pengulangan menggunakan *while* mengandung arti selama (*while*) kondisinya bernilai benar.

```
While (ekspresi)
{
    Statement;
    Statement;
}
```

b) *Do..while..*

Statement ini menjalankan ekspresi di akhir setelah aksi dilakukan.

```
do statement while (ekspresi)
```

c) *For*

Statement pengulangan ini mempunyai elemen ekspresi inisialisasi *variable* dan *counter* perulangan. Semua hal tersebut dapat dijadikan satu pada *statement* perulangan *for*.

```
for ([inisialisasi variabel], [ekspresi], [iterator],)
{
    Statement;
}
```

2.6.2 JDBC (*Java Database Connectivity*)

JDBC adalah sebuah nama dari sebuah produk yang ada dalam Java, atau yang dikenal sebagian orang sebagai Java Database Conectivity, seperti dokumentasi yang dikeluarkan oleh Sun Microsystem sendiri dalam situs resminya. Sebenarnya lengkapnya adalah JDBC API yaitu Application Programming Interface, yang menyediakan akses data secara universal dan independent. [12]

2.6.3 Pemrograman Berorientasi Objek

Ada 4 prinsip dasar dari pemrograman berorientasi obyek, yaitu abstraksi, enkapsulasi, modularitas, dan hirarki. Abstraksi memfokuskan perhatian pada karakteristik obyek yang paling dominan yang bias digunakan untuk membedakan obyek tersebut dari obyek lainnya. Dengan abstraksi ini developer bias menerapkan konsep *KIS (Keep It Simple)* pada obyek yang di dunia nyata memiliki kerumitan yang tinggi. Contoh abstraksi adalah obyek dosen yang diabstraksikan sebagai yang mengajar di perguruan tinggi, sementara mahasiswa adalah orang yang terdaftar belajar di perguruan tinggi.

Enkapsulasi menyembunyikan banyak hal yang terdapat dalam obyek yang tidak perlu diketahui oleh obyek lain. Dalam praktek pemrograman enkapsulasi diwujudkan dengan membuat suatu kelas interface. Obyek lain hanya tahu dia memanggil kelas interface dalam kelas implementasinya dan untuk menuntaskan proses tersebut perlu berhubungan dengan obyek mana saja. Dengan demikian bila terjadi perubahan pada proses implementasi maka obyek pemanggil tidak akan terpengaruhi secara langsung.

Modularitas membagi system yang rumit menjadi bagian-bagian yang lebih kecil yang bias mempermudah developer memahami dan mengelola obyek tersebut. Contohnya adalah Sistem Akademis yang bisa dibagi menjadi kemahasiswaan, katalog mata kuliah, dan pembayaran kuliah.

Hirarki berhubungan dengan abstraksi dan modularitas, yaitu pembagian berdasarkan urutan dan pengelompokan tertentu. Misalnya untuk menentukan objek mana yang berada pada kelompok yang sama, objek mana yang merupakan komponen dari objek yang memiliki hirarki lebih tinggi. Semakin rendah hirarki obyek berarti semakin jauh abstraksi dilakukan terhadap suatu obyek. Prinsip ini terjabar kemudian dalam konsep inheritance dan polymorphism.

Ke 4 prinsip dasar ini merupakan hal yang mendasari teknologi obyek dan perlu ditanamkan dalam cara berpikir *developer* berorientasi obyek. [13]

2.7 MySQL

MySQL adalah sebuah system manajemen database relasi (relational database management system) yang bersifat “terbuka” (open source). *MySQL* termasuk *RDMS (Relational Database Management System)*. Pada *MySQL* sebuah database terdiri atas tabel-tabel. Sebuah tabel terdiri atas baris dan kolom *MySQL* dapat didefinisikan sebagai :

- a) *MySQL* merupakan sistem manajemen *database*. *Database* merupakan struktur penyimpanan data. Untuk menambah, mengakses, dan memproses data yang disimpan dalam sebuah *database* komputer, diperlukan sistem manajemen *database* seperti *MySQL server*.
- b) *MySQL* merupakan sistem manajemen atau basis data yang terhubung (*relational database management system*). *Database* terhubung menyimpan data pada tabel-tabel terpisah. Hal tersebut akan menambah kecepatan dan fleksibilitasnya. Kata *SQL* pada *MySQL* merupakan bahasa standar yang digunakan untuk mengakses *database* dan ditetapkan oleh *ANSI/ISO SQL Standard*.
- c) *MySQL* merupakan *Software open source*. *Open source* berarti semua orang diizinkan menggunakan dan memodifikasi *software*. Semua orang dapat Mengunduh tanpa membayar. Anda dapat mempelajari *source code* dan akan menggunakannya sesuai kebutuhan.
- d) *Server database MySQL* mempunyai kecepatan akses tinggi, mudah digunakan dan handal. *MySQL* dikembangkan untuk menangani data yang besar secara cepat dan telah sukses digunakan selama bertahun-tahun. Konektifitas, kecepatan, dan keamanannya memuat *server MySQL* cocok untuk mengakses database di internet.
- e) *MySQL server* bekerja di *client/server* atau *system embedded*. *Software database MySQL* merupakan sistem *client/server* yang terdiri atas multithread *SQL server* yang mendukung *software client* dan *library* yang berbeda, *tool administrative*, dan sejumlah *application programming interpface (APIs)*.
- f) *MySQL* tersedia dalam beberapa macam bahasa. [12]

2.7.1 DDL

DDL (DataDefinitionLanguage) digunakan untuk mendefinisikan, mengubah, serta menghapus basis data dan objek-objek yang diperlukan dalam basis data, misalnya table, view, user, dsb. Secara umum, *DDL* yang digunakan adalah *CREATE* untuk membuat objek, *ALTER* untuk mengubah objek yang sudah ada, *USE* untuk menggunakan objek, dan *DROP* untuk menghapus objek.

2.7.2 DML

DML (Data Manipulation Language) digunakan untuk memanipulasi data yang ada dalam suatu table. Adapun perintah manipulasi yang umum dilakukan pada data adalah :

1. *SELECT* untuk menampilkan data.
2. *INSERT* untuk menambahkan data baru.
3. *UPDATE* untuk mengubah data yang sudah ada.
4. *DELETE* untuk menghapus data.

2.8 Kamus Data

Kamus Data ialah sebuah atau suatu aplikasi khusus yang digunakan sebagai referensi mengenai data yang telah disusun oleh seorang analis sistem untuk digunakan dalam melakukan analisa dan desain. [7]

1. Kelengkapan dan keakuratan dalam memvalidasi diagram alir data.
2. Persediaan yang lengkap dalam mengembangkan suatu layar dan laporan-laporan.
3. Pengembangan logika untuk proses-proses DAD.

Fungsi kamus data ialah suatu katalog yang secara detail menjelaskan tentang DFD yang mencakup proses-proses, *data flow* dan *data store*. Ada hal-hal yang harus di muat dalam kamus data yaitu : [7]

1. Nama arus data
2. Alias
3. Bentuk data
4. Arus data
5. Penjelasan

6. Periode
7. Volume
8. Struktur data

Ada juga notasi pada kamus data yaitu :

1. Notasi tipe data
2. Notasi struktur data

2.8.1 Notasi tipe data

Notasi tipe data digunakan dalam pembuatan spesifikasi format dari masukan dan keluaran suatu data. Contoh :

1. X untuk setiap karakter
2. 9 Angka numberik
3. A karakter alphabet
4. Z angka nol yang ditampilkan dalam spasi kosong
5. . Pemisahan pecahan
6. , Pemisahan pecahan
7. _ Tanda penghubung
8. / Pembagian

2.8.2 Notasi struktur data

Notasi struktur data digunakan untuk keterangan\

1. = Terdiri dari
2. + Dan atau *and*
3. () Pilihan *optional*
4. {} iterasi (perulangan proses)
5. [] Pilih salah satu yang ada
6. @ Kunci *field*

2.9 iReport

iReport merupakan salah satu *visual designer tool* untuk *jasperReports library* dan *Jasper Reports server*. Dalam *iReport user* dapat melihat *preview report* yang telah dibuat dalam sekali klik dengan integrasi *export* ke berbagai format seperti : [14]

1. Integrasi dengan *PDF, HTML, XHTML, XLSX, RTF, DOCX, TEXT, ODT, XML preview*.
2. *Support exporting* ke dalam *PDF, HTML, XHTML, XLS, RTF, DOCS, TEXT, CSV, OpenOffice, XML, PNG, JPEG*.
3. *Support* untuk *multiple character encodings*.

2.10 Black Block Testing

Black box testing adalah sebuah metode pengujian hasil program yang sudah dibuat dengan cara menjalankan semua persyaratan fungsional pada program. Pengujian program sangat dianjurkan karena demi mengetahui performa program yang sudah dibuat agar tidak ada masalah pada saat pengguna menggunakan program.

