



## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Konsep Dasar Sistem**

Istilah “sistem” masih populer. Terminologi ini digunakan untuk mendeskripsikan banyak hal. Usaha yang dilakukan pada masa lampau dalam pemrosesan data terfokus pada pengembangan mesin yang menjalankan operasi secara efisien. Kemudian, penemuan *punched cord* menegaskan bahwa pengkonversian data menjadi informasi adalah suatu proses. Selain itu, pengembangan komputer digital berikut teknologi yang menyertainya juga meningkatkan kepopuleran penggunaan terminologi “sistem” (Tata Sutabri, 2016)

##### **2.1.1 Pengertian Sistem**

Menurut Eddy Prahasta (2014) Sistem adalah sekumpulan komponen yang saling berhubungan satu sama lainnya dan bekerja sama untuk mencapai suatu tujuan.

##### **2.1.2 Karakteristik Sistem**

Konsep sebuah sistem yang sangat sederhana terdiri atas *input*, proses, dan *output*, sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran (Tata Sutabri, 2016)

Selain itu, sebuah sistem memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik sistem dapat dijelaskan sebagai berikut (Tata Sutabri, 2016) :

### 1. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi artinya saling bekerja sama membentuk satu kesatuan.

### 2. Batasan Sistem

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya.

### 3. Lingkungan Luar Sistem

Bentuk apa pun yang ada di luar ruang lingkup atau batasan sistem yang memengaruhi operasi sistem tersebut disebut lingkungan luar sistem.

### 4. Penghubung Sistem

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lain.

### 5. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

### 6. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain.

### 7. Pengolah Sistem (Proses)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

## 8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

### 2.1.3 Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada di dalam sistem tersebut. Oleh karena itu, sistem dapat diklasifikasikan dari beberapa sudut pandangan, di antaranya (Tata Sutabri, 2016)

:

#### 1. Sistem Abstrak dan Sistem Fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik sedangkan sistem fisik merupakan sistem yang ada secara fisik.

#### 2. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, seperti sistem perputaran bumi, terjadinya siang malam, pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan interaksi manusia dengan mesin, yang disebut *human machine system*.

### 3. Sistem Deterministik dan Sistem Probabilistik

Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem deterministik. Sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilistik.

### 4. Sistem Terbuka dan Tertutup

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak berpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa campur tangan pihak luar. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

#### 2.1.4 Daur Hidup Sistem

Siklus hidup sistem (*system life cycle*) adalah proses evolusioner yang diikuti dalam menerapkan sistem atau subsistem informasi berbasis komputer. Siklus hidup sistem terdiri dari serangkaian tugas yang erat mengikuti langkah-langkah pendekatan sistem karena tugas-tugas tersebut mengikuti pola yang teratur dan dilakukan secara *top down*. Siklus hidup sistem sering disebut sebagai pendekatan air terjun (*waterfall approach*) bagi pembangunan dan pengembangan sistem. Daur hidup sistem dapat dijelaskan sebagai berikut (Tata Sutabri, 2016) :

#### 1. Mengenali Adanya Kebutuhan

Sebelum segala sesuatunya terjadi, timbul suatu kebutuhan atau problema yang harus dapat dikenali sebagaimana adanya. Kebutuhan dapat terjadi sebagai hasil perkembangan dari organisasi dan volume

yang meningkat melebihi kapasitas dari sistem yang ada. Semua kebutuhan ini harus dapat didefinisikan dengan jelas. Tanpa adanya kejelasan dari kebutuhan yang ada, pembangunan sistem akan kehilangan arah dan efektivitasnya.

## 2. Pembangunan Sistem

Suatu proses atau seperangkat prosedur yang harus diikuti untuk menganalisis kebutuhan yang timbul dan membangun suatu sistem untuk dapat memenuhi kebutuhan tersebut.

## 3. Pemasangan Sistem

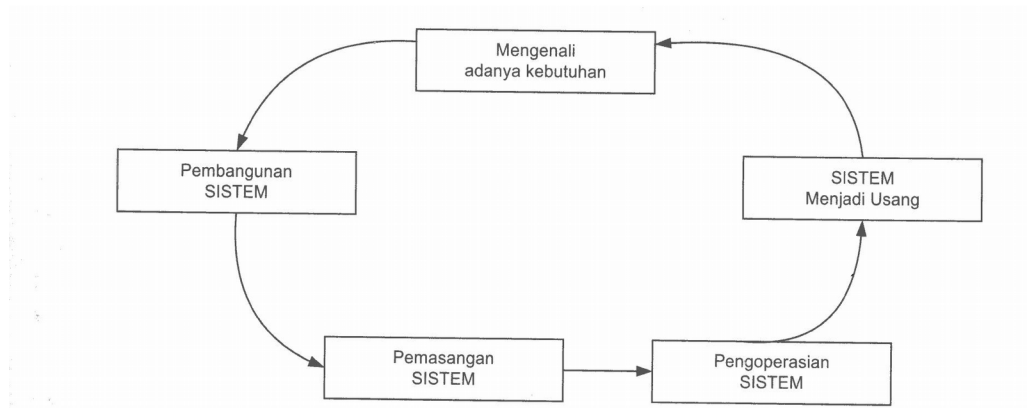
Setelah tahap pembangunan sistem selesai. Sistem kemudian akan dioperasikan. Pemasangan sistem merupakan tahap yang penting pula dalam daur hidup sistem. Peralihan dari tahap pembangunan menuju tahap operasional terjadi pemasangan sistem yang sebenarnya, yang merupakan langkah akhir dari suatu pembangunan sistem.

## 4. Pengoperasian Sistem

Program-program komputer dan prosedur-prosedur pengoperasian yang membentuk suatu sistem informasi semuanya bersifat statis, sedangkan organisasi ditunjang oleh sistem informasi tadi. Ia selalu mengalami perubahan-perubahan itu karena pertumbuhan kegiatan bisnis, perubahan peraturan, dan kebijaksanaan ataupun kemajuan teknologi. Untuk mengatasi perubahan-perubahan tersebut, sistem harus diperbaiki atau diperbarui.

## 5. Sistem Menjadi Usang

Kadang perubahan yang terjadi begitu drastis sehingga tidak dapat di atasi hanya dengan melakukan perbaikan-perbaikan pada sistem yang berjalan. Tibalah saatnya secara ekonomis dan teknis sistem yang ada sudah tidak layak lagi untuk dioperasikan dan sistem yang baru perlu dibangun untuk menggantikannya. Untuk lebih jelasnya mengenai daur hidup sistem dapat di lihat pada Gambar 2.1 :



Sumber : (Tata Sutabri, 2016)

**Gambar 2.1 Daur Hidup Sistem**

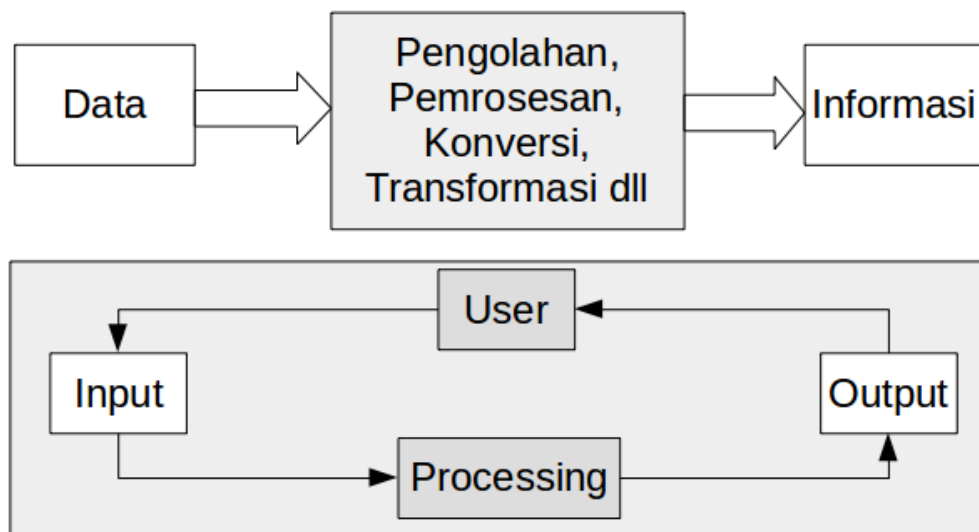
## 2.2 Konsep Informasi

Analisis mengenai sistem informasi pada umumnya di mulai dengan deskripsi mengenai “data” dan “informasi. Informasi berasal dari data maka dari itu keduanya tidak dapat dipisahkan (Eddy Prahasta, 2014)

### 2.2.1 Pengertian Data dan Informasi

Menurut Eddy Prahasta (2014) Data merupakan bahasa, *mathematical*, dan atau simbol pengganti lain yang disepakati secara umum dalam menggambarkan

suatu objek, manusia, peristiwa, aktivitas, konsep, atau objek penting lainnya. Singkatnya, data merupakan suatu kenyataan apa adanya. Sedangkan informasi adalah data yang telah ditempatkan pada konteks yang penuh arti oleh penerimanya. Untuk lebih jelasnya mengenai hubungan antara data dan informasi dapat dilihat pada Gambar 2.2 :



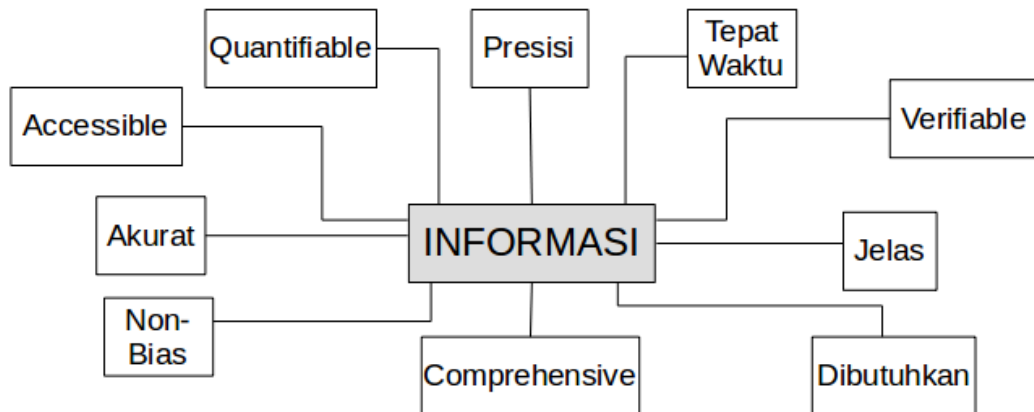
Sumber : (Eddy Prahasta, 2014)

**Gambar 2.2 Hubungan Antara Data dan Informasi**

### 2.2.2 Atribut Informasi

Banyak atribut atau kualitas yang berkaitan dengan konsep informasi dapat membantu perancang dalam mengidentifikasi kebutuhan informasi. Untuk lebih jelasnya mengenai atribut-atribut informasi dapat dilihat pada Gambar 2.3 :





Sumber : (Eddy Prahasta, 2014)

**Gambar 2.3 Atribut-Atribut Informasi**

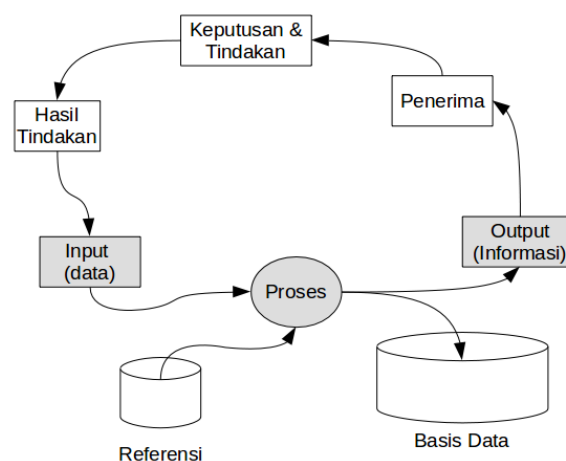
Dari Gambar 2.3 dapat dijelaskan sebagai berikut :

1. Akurat: tingkat kebebasan informasi dari kesalahan.
2. Quantifiable: tingkat kemampuan menyatakan informasi dalam bentuk numerik.
3. Presisi: ukuran detil untuk menyediakan informasi.
4. Tepat waktu: penerimaan informasi masih dalam toleransi waktu yang dibutuhkan oleh penerima; tidak kadaluarsa.
5. Jelas: tingkat kebebasan informasi dari keraguan.
6. Dibutuhkan: tingkat relevansi informasi dengan kebutuhan pengguna.
7. Verifiable: tingkat kesamaan nilai sebagai hasil pengujian informasi yang sama oleh berbagai pengguna.
8. Accessible: tingkat kemudahan dan kecepatan memperoleh informasi.
9. Non-bias: tingkat perubahan untuk memodifikasi informasi dengan tujuan mempengaruhi para penerimanya.
10. Comprehensive: tingkat kelengkapan informasi.

Pada saat mengidentifikasi kebutuhan informasi, sedapat mungkin kebutuhan tersebut dideskripsikan dalam terminologi atribut informasi. Analisis yang benar akan menunjukkan keterkaitan erat antara kebutuhan informasi beserta atributnya pada perancangan sistem informasi. Jadi, yang diperlukan adalah penyediaan informasi yang benar pada waktu yang tepat (Eddy Prahasta, 2014)

### 2.2.3 Siklus Data dan Informasi

Menurut Eddy Prahasta (2014) Aliran fakta simbol, string alfabetis/kumpulan angka rekaman kejadian disebut sebagai “data”. Pada diagram, aliran ini masuk ke proses. Oleh karena itu, data disebut masukan/ *input*. Setelah aliran ini diproses (hingga menghasilkan keluaran dalam bentuk yang dimengerti), bentuk ini disebut “informasi”. Oleh karena itu, dipandang dari sisi pemroses data dikenal sebagai *input*, sementara informasi disebut sebagai *output*. Untuk lebih jelasnya mengenai siklus data dan informasi dapat dilihat pada Gambar 2.4 :



Sumber : (Eddy Prahasta, 2014)

**Gambar 2.4 Siklus Data dan Informasi**

Di sisi pemrosesan, fenomena *input-proses-output* bersifat relatif. Suatu proses bisa menerima *input* hasil olahan (*output*) proses lainnya. Hal ini nampak

pada sekelompok manusia dengan perannya sebagai pemroses. Pihak mana yang melakukan proses pertama, dan pihak lain mana) melakukan proses berikutnya. Tidak jarang, sekelompok pemroses tidak hanya berkemampuan melakukan pemrosesan yang sekarang menjadi tugasnya, mereka juga mampu melakukan pemrosesan yang sebenarnya menjadi tugas kelompok lain pada saat ini. Selain itu, seiring dengan perubahan waktu & proses belajar, suatu kelompok juga bisa melakukan proses lainnya. Oleh karena itu, kemampuan ganda & daya-adaptasi manusia bisa “mengaburkan” pengertian data & informasi (Eddy Prahasta, 2014)

### **2.3 Konsep Dasar Sistem Informasi**

Didalam sebuah perusahaan biasanya terdapat seorang manajer. Seorang manajer yang dulunya mengambil kuliah sistem informasi manajemen sudah pasti bisa menjadi seorang manajer praktisi. Jika seandainya manajer tersebut tidak mengambil kuliah sistem informasi manajemen dan tidak mampu menggunakan komputer untuk kebutuhan perusahaannya, maka manajer tersebut menjadi tidak berguna dan tak bermanfaat lagi sebagai pengambil keputusan (Tata Sutabri, 2016)

#### **2.3.1 Pengertian Sistem Informasi**

Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan (Tata Sutabri, 2016)

### 2.3.2 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan yang saling bekerjasama. Adapun komponen sistem informasi, (Tata Sutabri, 2016) yaitu :

1. Blok Masukan

*Input* mewakili data yang masuk kedalam sistem informasi. *Input* disini termasuk metode dan media untuk menangkap data yang akan dimasukkan.

2. Blok Model

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran

Produk dari sistem informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi

Teknologi merupakan *toolbox* dalam sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 bagian yaitu teknisi, perangkat lunak dan perangkat keras.

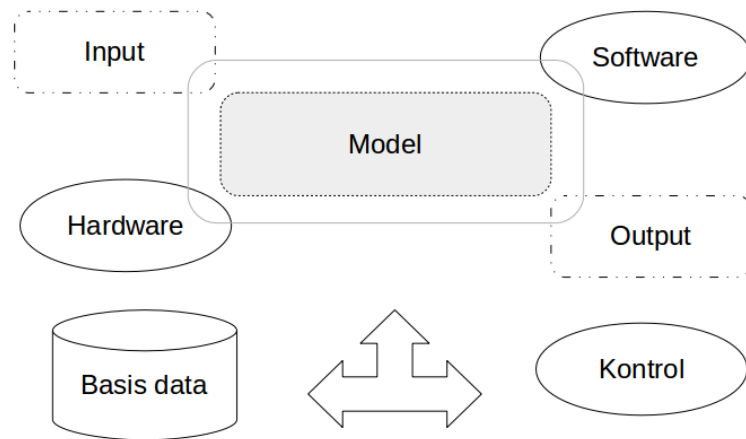
## 5. Blok Basis Data

Basis data (*database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut DBMS (*database management system*)

## 6. Blok Kendali

Banyak hal yang dapat merusak sistem informasi seperti bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, ketidakefisienan, sabotase, dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat di atasi.

Untuk lebih jelas mengenai hubungan blok-blok pada sistem informasi dapat dilihat pada Gambar 2.5 :



Sumber : (Tata Sutabri, 2016)

### Gambar 2.5 Komponen Sistem Informasi

Berdasarkan komponen fisik penyusunnya, komponen sistem informasi terdiri atas (Tata Sutabri, 2016) :

#### 1. Perangkat Keras

Meliputi perangkat-perangkat yang digunakan oleh sistem komputer untuk masukan dan keluaran, *memory*, *modem*, pengolah dan periferal lainnya.

#### 2. Perangkat Lunak

Berupa program-program komputer yang meliputi sistem operasi dan program-program aplikasi.

#### 3. Berkas Basis Data

Merupakan sekumpulan data dalam basis data yang disimpan dengan cara-cara tertentu sehingga dapat digunakan kembali dengan mudah dan cepat.

#### 4. Prosedur

Meliputi prosedur pengoperasian untuk sistem informasi, manual dan dokumen-dokumen yang memuat aturan-aturan yang berhubungan dengan sistem informasi dan lainnya.

#### 5. Manusia

Manusia yang terlibat dalam suatu sistem informasi meliputi operator, *programmer*, *system analyst*, manajer sistem informasi, manajer pada tingkat operasional, manajer pada tingkat manajerial, manajer pada tingkat strategis, teknisi, administrator basis data serta individu lain yang terlibat didalamnya .

### 2.3.3 Perencanaan Sistem Informasi

Pada umumnya, setiap terjadi perencanaan atau perubahan pada sistem, baik besar maupun kecil, selalu akan melalui tingkatan-tingkatan sebagai berikut (Tata Sutabri, 2016) :

1. Tingkat I : Ide, mengetahui perlu adanya perubahan.
2. Tingkat II : Desain, merancang cara pemecahannya.
3. Tingkat III : Pelaksanaan, menerapkan desain kedalam sistem.
4. Tingkat IV : Kontrol, memeriksa tingkat pelaksanaan dijalankan sesuai dengan desain.
5. Tingkat V : Evaluasi, memeriksa apakah perubahan yang terjadi sesuai dengan tujuan semula.
6. Tingkat VI : Tindak lanjut, melaksanakan perubahan sesuai dengan hasil evaluasi yang ada.

## 2.4 *Computer Assisted Test (CAT)*

*Computer Assisted Test* didefinisikan sebagai suatu metode ujian dengan menggunakan alat bantu komputer yang digunakan untuk mendapatkan standar minimal kompetensi dasar maupun standar kompetensi kepegawaian (BKN, 2014). Adapun tahapan proses dalam perancangan sistem CAT diawali dengan penelitian dan pengumpulan data, kemudian perencanaan, pembuatan prototipe, pelaksanaan uji coba, dan diikuti perbaikan dan pengembangan (I Ketut Buana dan Made Gede Wirakusuma, 2015)

Menurut Khusnul Khotimah (2016), tujuan dari penggunaan *Computer Assisted Test* adalah sebagai berikut :

1. Mempercepat proses pemeriksaan dan laporan hasil ujian
2. Menciptakan standarisasi hasil ujian secara nasional
3. Menetapkan standar nilai.

Menurut BKN dalam jurnal I Ketut Buana dan Made Gede Wirakusuma (2015) CAT memiliki prinsip sebagai berikut :

1. Sistem CAT dirancang semudah mungkin agar pengguna dapat dengan mudah memahaminya.
2. Pengoperasian yang sangat mudah. Yaitu dengan menggerakkan *mouse* untuk memilih, menampilkan dan menjawab soal.
3. Soal ditampilkan secara acak pada setiap peserta.
4. Pemeriksaan hasil tes langsung dilakukan oleh aplikasi CAT secara otomatis.



## 2.5 Konsep Basis Data

Menurut Eddy Sutanta (2011), istilah basis data dapat dipahami sebagai suatu kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data (Kalaupun ada maka kerangkapan data tersebut harus seminimal mungkin dan terkontrol), data disimpan dengan cara-cara tertentu sehingga mudah digunakan; data dapat digunakan oleh satu atau lebih program-program aplikasi secara optimal; data disimpan tanpa mengalami ketergantungan dengan program yang akan menggunakannya; data disimpan sedemikian rupa sehingga proses penambahan, pengambilan, dan modifikasi data dapat dilakukan dengan mudah dan terkontrol.

Sedangkan menurut Sri Rahmawati (2014) Basis data atau *database* adalah sebuah tempat untuk menyimpan data-data penting dan disamping itu data-data yang ada didalam database dapat dipanggil kembali kapan saja saat kita membutuhkannya. Dalam kehidupan sehari-hari basis data ini dapat dicontohkan seperti lemari penyimpanan arsip-arsip penting yang telah diberi tanda atau kode tertentu agar saat dilakukan pencarian menjadi lebih mudah dan cepat.

### 2.5.1 Sistem Basis Data

Menurut Eddy Sutanta (2011) sistem basis data dapat diartikan sebagai sekumpulan basis data dalam suatu sistem yang mungkin tidak ada hubungan satu sama lain, tetapi secara keseluruhan mempunyai hubungan sebagai sebuah sistem dengan didukung oleh komponen lainnya.

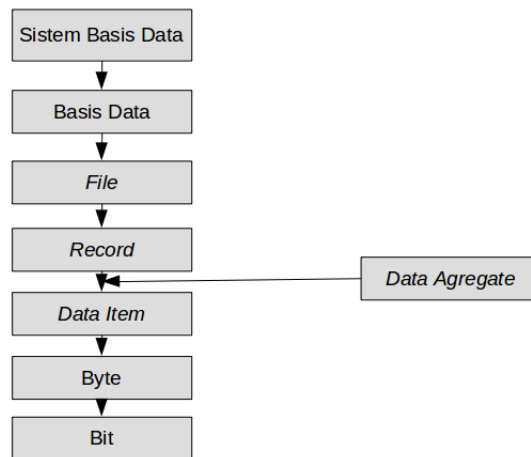
### 2.5.2 Hierarki Data

Hierarki data yang dimulai dari paling kompleks hingga yang paling sederhana dapat dijelaskan sebagai berikut (Eddy Sutanto, 2011) :

1. Sistem Basis Data, merupakan sekumpulan basis data dengan para pemakai yang menggunakan basis data tersebut secara bersama-sama.
2. Basis Data, merupakan sekumpulan dari bermacam-macam tipe *record* yang memiliki hubungan antar-*record* dan rincian data terhadap obyek tertentu.
3. *File*, merupakan sekumpulan *record* sejenis secara relasi yang tersimpan dalam media penyimpanan sekunder.
4. *Record*, merupakan sekumpulan *field*/attribut/data item yang saling berhubungan terhadap obyek tertentu.
5. *Field*, merupakan unit terkecil yang disebut data, yaitu sekumpulan *byte* yang mempunyai makna.
6. *Data Agregate*, merupakan sekumpulan data *field*/item/attribut dengan ciri khas tertentu dan diberi nama.
7. *Byte*, merupakan bagian terkecil yang dialamatkan pada memori.
8. *Bit*, merupakan sistem biner yang terdiri atas dua macam nilai, yaitu 0 dan 1.

Untuk lebih jelasnya mengenai Hierarki data dapat dilihat pada Gambar

2.6 :



Sumber : (Eddy Sutanta, 2011)

**Gambar 2.6 Hierarki Data**

### 2.5.3 Tujuan Pengembangan Basis Data

Setiap tindakan pasti memiliki tujuan yang akan dicapai. Tujuan utama dari pengembangan basis data adalah sebagai berikut (Eddy Sutanta, 2011) :

1. Data-data dalam basis data dapat digunakan oleh banyak pengguna.
2. Menjaga investasi intelektual.
3. Penekanan biaya.
4. Menghilangkan pengembangan sistem ganda.
5. Kinerja untuk memenuhi kebutuhan informasi.
6. Kejelasan data yang disimpan.
7. Kemudahan pemakaian.
8. Fleksibilitas penggunaan.
9. Memenuhi kebutuhan data yang tidak terantisipasi.
10. Perubahan yang muda.
11. Akurasi dan konsistensi.

12.Privasi.

13.Keamanan.

14.Ketersediaan.

#### 2.5.4 Keuntungan Pengembangan Basis Data

Basis data dikembangkan untuk mengatasi permasalahan-permasalahan pada saat pengolahan data. Basis data yang dikembangkan secara benar akan memberikan keuntungan, diantaranya (Eddy Sutanto, 2011) :

1. Kerangkapan data dapat diminimalkan.
2. Inkonsistensi data yang dapat dihindari.
3. Data dapat digunakan secara bersama.
4. Standarisasi data dapat dilakukan.
5. Pembatasan untuk keamanan data dapat diterapkan.
6. Integritas data dapat dipelihara.
7. Perbedaan kebutuhan data dapat diseimbangkan.

#### 2.5.5 PostgreSQL

*Postgresql* adalah *ORDBMS (Object Relational Database Management System)* yang paling maju didunia. *Postgresql* merupakan perangkat lunak *open source*. *Postgresql* dikembangkan oleh *PostgreSql Global Development Group* yang terdiri dari segelintir relawan yang dipekerjakan dan diawasi oleh perusahaan seperti *Red Hat* dan *Enterprise DB* (Sukhdeep Kaur, 2016)

*PostgreSql* tersedia untuk hampir semua sistem operasi seperti: *Linux* (semua distribusi terbaru), *Windows*, *UNIX*, *Mac OSX*, *FreeBSD*, *OpenBSD*,

*Solaris*, dan lainnya. Sistem seperti *Unix* bekerja pada semua mayoritas arsitektur seperti: *x86*, *x86-64*, *IA64*, *PowerPC*, *Sparc*, *Alpha*, *ARM*, *MIPS*, *PA-RISC*, *VAX*, *M32R* (Sukhdeep Kaur, 2016)

*MySQL* dan *PostgreSql* keduanya bersaing kuat di bidang *database* relasional karena keduanya memiliki fungsi yang canggih dan juga kinerja dan kecepatan yang sebanding. Yang terpenting kedua *database* tersebut bersifat *open source* (Sukhdeep Kaur, 2016)

Menurut Mochamad Alfian Rosid (2016) *Postgresql* sebagai *ORDBMS* (*Object Relational Database Management System*) berbagai macam kemampuan yang dimiliki oleh *database* komersil umum lainnya, seperti dukungan akan perintah-perintah *SQL*, dimana dengan menggunakan perintah-perintah *SQL* memungkinkan *database administrator* lebih mudah berinteraksi dengan *database PostgreSQL*, baik dalam manipulasi data seperti *insert*, *update*, ataupun *delete*. Selain bersifat *open source*, keunggulan *database PostgreSQL* ini dapat mendukung sebagai media penyimpanan pada banyak bahasa pemrograman yang ada, baik itu pemrograman berbasis *desktop* seperti : *Java*, *Gambas*, dll ataupun pemrograman berbasis web, seperti : *Phyton*, *PHP*, *Java*, *server Pages*, *Perl*, dll.

## 2.6 Bahasa Pemrograman *Javascript*

*Javascript* adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokumen *HTML* yang ditampilkan dalam *browser* menjadi lebih interaktif. *Javascript* memberikan beberapa fungsionalitas kedalam halaman *web*, sehingga dapat menjadi sebuah program yang disajikan dengan menggunakan antarmuka web (Betha Sidik, 2011)

### 2.6.1 NodeJS

*NodeJS* adalah sistem perangkat lunak yang didesain untuk pengembangan aplikasi web. Aplikasi ini ditulis dalam bahasa *JavaScript*, menggunakan basis *event* dan *asynchronous I/O*. Tidak seperti kebanyakan bahasa *JavaScript* yang dijalankan pada peramban, *NodeJS* dieksekusi sebagai aplikasi *server*. Aplikasi ini terdiri dari *V8 JavaScript Engine* buatan *Google* dan beberapa modul bawaan yang terintegrasi (Muhammad Iqbal C. R., Muchammad Husni dan Hudan Studiawan, 2012)

### 2.7 Unified Modelling Language (UML)

Bahasa yang digunakan untuk memodelkan perangkat lunak yang akan dibuat dan perlu adanya standarisasi agar orang di berbagai negara dapat mengerti pemodelan perangkat lunak. Sebagaimana kita ketahui bahwa menyatukan banyak kepala untuk menceritakan sebuah ide dengan tujuan untuk memahami hal yang sama tidaklah mudah, oleh karena itu diperlukan sebuah bahasa pemodelan perangkat lunak yang dapat dimengerti oleh banyak orang (Rosa A.S – M. Shalahuddin, 2015)

Banyak orang yang telah membuat bahasa pemodelan pembangunan perangkat lunak sesuai dengan teknologi pemrograman yang berkembang pada saat itu, misalnya yang sempat berkembang dan digunakan oleh banyak pihak adalah *Data Flow Diagram* (DFD) untuk memodelkan perangkat lunak yang menggunakan pemrograman prosedural atau struktural, kemudian juga ada *State Transition Diagram* (STD) yang digunakan untuk memodelkan sistem *real time* (waktu nyata) (Rosa A.S – M. Shalahuddin, 2015)

Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung (Rosa A.S – M. Shalahuddin, 2015)

### **2.7.1 Use Case Diagram**

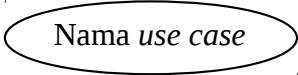
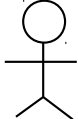

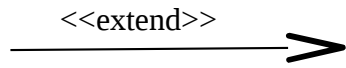
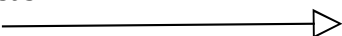
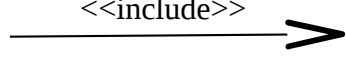
Menurut Rosa A.S dan M. Shalahuddin (2015), *use case* merupakan pemodelan yang digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakannya.

Ada dua hal utama pada *use case* yaitu pendefinisian apa yang akan disebut aktor dan *use case*. Berikut adalah penjelasan dari aktor dan *use case* :

1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Dalam menggambarkan *use case*, digunakan simbol-simbol yang memiliki makna-makna tertentu. Simbol-simbol yang digunakan pada *use case* dapat dilihat pada Tabel 2.1 :

Tabel 2.1: Simbol-simbol pada diagram *use case*

No	Simbol	Deskripsi
1	<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal di awal frase nama <i>use case</i> .
2	Aktor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
3	Asosiasi 	komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4	Ekstensi 	relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek.
5	Generalisasi 	Hubungan generalisasi dan spesialisasi antara dua buah <i>use case</i> dimanafungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6	Menggunakan <i>include</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use cse</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.



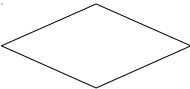


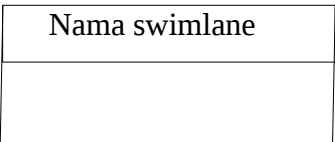
Sumber : Rosa A.S dan M. Shalahudin, 2015



### 2.7.2 Activity Diagram

Diagram aktivitas menggambarkan aliran kerja dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Simbol-simbol pada diagram aktivitas dapat dilihat pada Tabel 2.2 :

**Tabel 2.2 Simbol-simbol pada *activity diagram***

No	Simbol	Deskripsi
1	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2	aktivitas 	Aktivitas yang dilakukan sistem, biasanya diawali dengan kata kerja
1	2	3
3	Percabangan 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4	penggabungan 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5	Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
6	Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap yang terjadi.

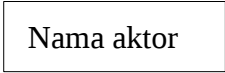

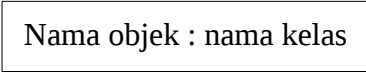


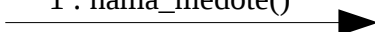

Sumber : Rosa A.S dan M. Shalahudin, 2015

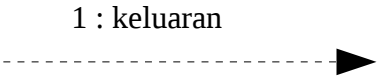
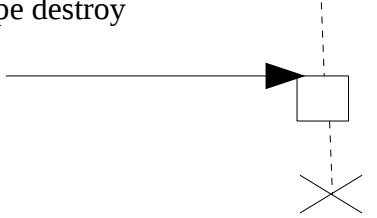
### 2.7.3 Sequence Diagram

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus

diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansi menjadi objek itu. Banyaknya sekuen diagram minimal sama dengan banyaknya *use case* (Rosa A.S dan M. Shalahudin, 2015). Simbol-simbol pada diagram sekuen dapat dilihat pada Tabel 2.3 :

**Tabel 2.3: Simbol-simbol pada diagram sekuen**

No	Simbol	Deskripsi
1	2	3
1	Aktor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri.
2	Garis hidup 	Menyatakan kehidupan suatu objek.
3	Objek 	Menyatakan objek yang berinteraksi pesan.
4	Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.
5	Pesan tipe create 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
6	Pesan tipe call 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
7	Pesan tipe send 	Menyatakan bahwa suatu objek mengirimkan data ke objek lainnya, arah panah mengarah pada objek yang dikirim.

1	2	3
8	Pesan tipe return 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
9	Pesan tipe destroy 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

Sumber : Rosa A.S dan M. Shalahudin, 2015

#### 2.7.4 Class Diagram

Diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem, Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas. Dan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas (Rosa A.S – M. Shalahudin, 2015)

Menurut Rosa A.S dan M. Shalahudin (2015) Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut :

1. Kelas *main*

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.

2. Kelas yang menangani tampilan sistem

Kelas yang mendefinisikan dan mengatur tampilan ke pemakai.

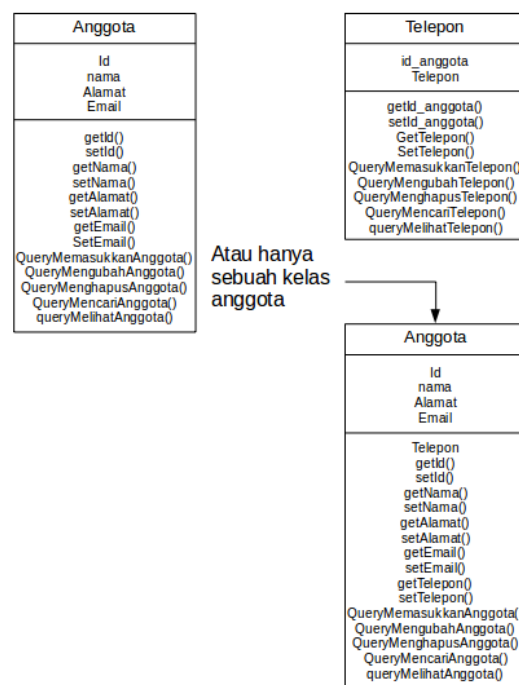
### 3. Kelas yang diambil dari pendefinisian *use case*

Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.

### 4. Kelas yang diambil dari pendefinisian data

Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data. Semua tabel yang dibuat di basis data dapat dijadikan kelas, namun untuk tabel dari hasil relasi atau atribut *multivalued* pada ERD dapat dijadikan kelas tersendiri dapat juga tidak asalkan pengaksesannya dapat dipertanggungjawabkan atau tetap ada di dalam perancangan kelas.

Untuk lebih jelasnya contoh kelas diagram dapat dilihat pada Gambar 2.7 :



Sumber : (Rosa A.S – M. Shalahudin, 2015)

**Gambar 2.7 Contoh kelas diagram**