# BAB II

# LANDASAN TEORI

## 2.1. Definisi Sistem

Sistem dapat didefinisikan sebagai suatu kesatuan yang terdiri dari dua atau lebih komponen atau subsistem yang berinteraksi untuk mencapai tujuan suatu tujuan.(Jogiyanto. HM,1999 : 1).

UNIT PELAKSANA PERPUSTAKAAN UNIKOM

Sistem yang terdiri dari sejumlah komponen atau elemen-elemen kemudian saling berinteraksi dan membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan agar tercapai suatu tujuan.

## 2.2. Definisi Basis Data (Database)

Basis dapat diartikan sebagai gudang atau markas, tempat bersarang / berkumpul. Sedangkan Data adalah kumpulan dari kejadian-kejadian yang diangkat berdasarkan fakta (kenyataan). Data juga merupakan Sumber Daya untuk memperoleh informasi.

Jadi Basis Data dapat didefinisikan sebagai himpunan kelompok data yang saling berhubungan yang diorganisir sedemikian rupa dan tanpa pengulangan (*redundancy*) agar dapat dimanfaatkan kembali dengan cepat dan mudah sesuai dengan kebutuhan, yang tersimpan dalam media elektronik yaitu komputer.

Tujuan dibuat basis data adalah untuk mengorganisir kumpulan data yang ada menjadi suatu bentuk kesatuan data yang saling terkait satu sama lain untuk memperoleh suatu informasi. Untuk lebih lengkapnya pemanfaatan basis data dilakukan untuk memenuhi sejumlah tujuan (objektif) sebagai berikut :

1. Efisiensi Ruang Penyimpanan (*Space*)
2. Kecepatan dan Kemudahan (*Speed*)
3. Keakuratan (*Accurancy*)
4. Ketersediaan (*Availability*)
5. Kelengkapan (*Completeness*)
6. Kebersamaan Pemakaian (*Sharability*)

## 2.3. Sistem Manajemen Basis Data (Database Management System / DBMS)

Data perlu disimpan dalam database untuk keperluan penyediaan informasi selanjutnya. Data di dalam database perlu diorganisasikan sedemikian rupa, supaya informasi yang dihasilkan berkualitas. Database diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut dengan *Database Management system (DBMS)*.

Sistem Manajemen Database (*Database Management System / DBMS*) adalah satu rangkaian program-program yang mengelola sebuah database dan menyediakan mekanisme-mekanisme dimana jenis-jenis data dapat disimpan, dicari kembali, dan diubah.

Tiap Sistem Manajemen Database tergantung pada empat modul perangkat lunak utama yang bertindak sebagai penghubung antara pemakai dan komputer. Modul-modul ini adalah: definisi data, manipulasi data, laporan, dan kamus data.

1. Modul Definisi Data

   Modul definisi data adalah modul dimana struktur data atau format data didefinisikan. Kolom (*field*) nama, panjang kolom, dan jenis data dan informasi lain, semuanya dimasukkan melalui suatu bahasa definisi data (*data definition language*/DDL) ketika database dibuat.

2. Modul Manipulasi Data

   Modul ini sama saja dengan modul pertama dimana modul manipulasi data membuatnya dapat menggunakan database melalui suatu bahasa manipulasi data (*data manipulation language*/DML). Bahasa ini adalah bahasa perintah dimana pendesain atau pemakai dapat mengakses, memproses, dan memanipulasi database.

3. Modul Laporan

   Modul laporan ini menyediakan kerangka kerja dan bimbingan dalam mendesain laporan.

4. Modul Kamus Data

   Kamus data adalah suatu file yang terpisah yang menyimpan informasi. Kamus data berguna khusus bagi perlindungan timbulnya kelebihan data/ penggandaan data.

Perangkat lunak yang termasuk DBMS seperti dBase III+, dBase IV, FoxBase, Rbase, Ms-Access dan Borland-Paradox (untuk kelas sederhana) atau

Borland-Interbase, MS-SQLServer, CA-Open Ingres, Oracle, Infomix dan Sysbase(untuk kelas kompleks). DBMS dapat diakses oleh programmer yang membuat program aplikasi, pemakai yang mendefinisikan kebutuhan/keperluannya, dan *operating system*. Sebagai perangkat lunak, DBMS dapat dipergunakan untuk :

1. Memadukan data sehingga dapat digunakan untuk keperluan yang lebih luas
2. Menghindari penggandaan data
3. Memproteksi sistem dari pemakai yang tidak diperbolehkan
4. Pemutakhiran data.

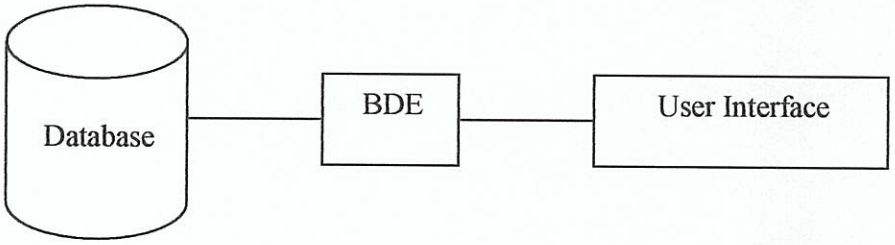
Pemakaian DBMS memerlukan spesialis yang disebut ***Database Administrator*** yaitu seorang ataupun beberapa orang yang mempunyai tanggung jawab untuk memelihara basis data, menentukan data dapat diatur secara efektif, dan mengatur sistem pengamanan untuk mencegah masuknya pemakai yang tidak dikehendaki.

## 2.4. Borland C++ Builder

Borland C++ Builder adalah suatu alat pengembang aplikasi berbasis Microsoft Windows yang menerapkan konsep visualisasi. Bahasa yang digunakan dalam *compiler* C++ Builder adalah bahasa C++ yang terkenal dengan keampuhannya dalam pembuatan program.

Sebagai salah satu alat pengembang aplikasi yang bersifat RAD (*Rapid Application Development*), C++ Builder juga dapat membuat aplikasi database (basis data) secara mudah dan cepat. Adapun yang dinamakan dengan aplikasi

database adalah aplikasi yang didalamnya terdapat proses pengaksesan dan pemanipulasian data dari sebuah database (kumpulan table) dimana data-data tersebut akan dihubungkan dengan sebuah *User Interface* (UI) atau form-form. Lingkungan database sering dikenal dengan istilah *Back End*, Sedangkan Lingkungan *User Interface* dikenal dengan istilah *Front End*. Berikut ini gambaran secara global dari sebuah aplikasi database pada C++ Builder.




**BACK END** **FRONT END**

**Gambar 2.1** Gambaran global aplikasi database pada C++ Builder

BDE (*Borland Database Engine*) adalah alat (*software*) yang disediakan C++ Builder untuk membuat komunikasi antara database dengan *User Interface*. Selain itu terdapat beberapa alat atau fitur yang perlu diketahui dalam mengembangkan aplikasi database, diantaranya:

1. Database Explorer

   Fitur ini berupa *software* dan berguna untuk melihat isi database, mengatur alias BDE dan juga untuk membuat kamus data (*data dictionary*).

2. Database Desktop

   Fitur ini juga berupa *software* dan berguna untuk melakukan pembuatan dan perubahan terhadap isi database.

3. Data Modul

   Fitur ini berupa unit yang berguna untuk menempatkan komponen-komponen seperti TTable, TQuery, dan TStoredProc.

4. Quick Report

   Fitur ini berupa komponen-komponen yang terdapat pada *Component Pallete* di dalam IDE C++ Builder. Fitur ini berfungsi untuk membuat laporan-laporan database di dalam aplikasi yang tentunnya sangat dibutuhkan oleh pengguna (*user*).

Jenis database itu bermacam-macam diantaranya OO (*Object Oriented*), RDB (*Relational Database*), dan lain-lain. Sistem yang digunakan untuk pengaturan database jenis RDB dinamakan RDMS (*Relational Database Management System*).

## 2.5. Konsep OOP (Object Oriented Programming)

Konsep utama pemrograman berorientasi objek yaitu melakukan permodelan objek dari kehidupan nyata ke dalam tipe data abstrak. Jelasnya, pemrograman berorientasi objek merupakan konsep pemrograman untuk memodelkan objek yang kita gunakan dalam kehidupan sehari-hari, dan konsep seperti ini membawa perubahan yang mendasar dalam konsep pemrograman terstruktur.

Pemrograman prosedural murni yang tidak menerapkan konsep *object oriented* banyak menitikberatkan pada arah pembentukan fungsi-fungsi, sehingga dalam program akan terdapat banyak fungsi dan variabel yang tidak beraturan.

Dari kelemahan tersebut, dibentuklah bahasa yang menerapkan OOP untuk menyederhanakan fungsi-fungsi dan variabel-variabel ke dalam bentuk objek.

Sebuah objek dapat didefinisikan sebagai suatu yang mempunyai data (berupa variable atau konstanta) dan perilaku (berupa fungsi). Jika C++ Builder adalah suatu lingkungan / objek, data-data milik objek tersebut disebut *properti* dan perilaku yang dimilikinya disebut *method*, dan juga kejadian-kejadian yang dapat menimpa objek tersebut adalah *event*.

Objek dapat dideklarasikan sebagai suatu yang diminati untuk diselidiki adalah *objek* yang ada diruang kerja tersebut. Meja, kursi, lemari adalah kelas furniture yang mempunyai *properti* seperti, nama, berat, tinggi, ukuran, bahan dan lain-lain. Kemudian Bapak, ibu dan anda adalah kelas manusia yang mempunyai *properti* juga, nama, berat, tinggi, dan bentuk pisik seperti mata, hidung, bibir, telinga, tangan, kaki dan lain-lain. Selain itu, manusia mempunyai *method*, misalnya marah, sedih, makan, tidur dan perilaku lain. Sedangkan *event* contohnya bila ketika kita merasa lapar tentu kita akan makan.

Yang dimaksud dengan OOP sebenarnya hanyalah metode pemrograman yang menirukan keadaan dunia yang sebenarnya, yaitu dengan membuat objek-objek untuk masalah yang dihadapi.

Secara garis besar yang menjadi ciri dari OOP adalah terdapatnya proses abstraksi (*abstraction*), pengkapsulan / pemodulan (*encapsulation*), penurunan sifat (*inheritance*) dan polimorfisme (*polymorphism*) pada objek-objek yang dibentuk.

1. Abstraksi

   Kelas adalah sebuah contoh abstraksi dari suatu objek, artinya data-data dan fungsi-fungsi yang terdapat pada suatu kelas baru dapat digunakan setelah melalui proses instansiasi. Hal ini merupakan abstraksi salah satu dasar dari OOP.

2. Pengkapsulan / pemodulan

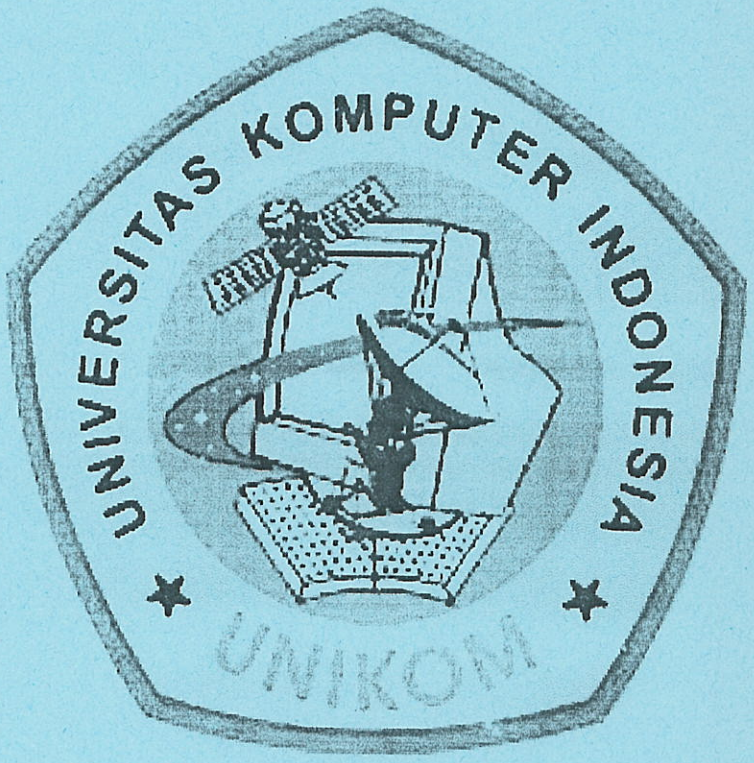
   Ciri kedua dari OOP adalah pengkapsulan yaitu proses pembungkusan atau penyederhanaan dari beberapa data serta fungsi menjadi sebuah objek/kelas.

3. Penurunan sifat

   Penurunan Sifat merupakan ciri utama dari OOP dimana sifat-sifat yang terdapat pada kelas induk akan dimiliki oleh kelas keturunannya. Hal ini tergantung kepada *access specifier* yang diberikan dalam proses penurunan kelas.

4. Polimorfisme

   Polimorfisme/kebanyakrupaan adalah kemampuan program untuk mengungkap banyak hal yang berbeda dengan menggunakan cara yang sama. Polimorfisme berarti objek-objek yang berbeda-beda tetapi berasal dari satu orang tua dapat mempunyai metode yang sama tetapi cara pelaksanaannya berbeda-beda.


UNIVERSITAS KOMPUTER INDONESIA
UNIKOM