

Pengenalan Pemrograman Delphi dan OOP

Adenisa Refitasari 165100034, 785567865

Fakultas Komputer

adenisarefitasari.student@umitra.ac.id

Abstract

Delphi suatu program berbasis bahasa Pascal yang berjalan dalam lingkungan Windows. Delphi telah memanfaatkan suatu teknik pemrograman yang disebut RAD yang telah membuat pemrograman menjadi lebih mudah. Delphi adalah suatu bahasa pemrograman yang telah memanfaatkan metode pemrograman Object Oriented Programming (OOP).

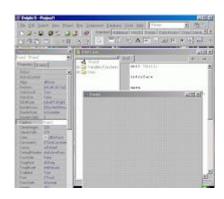
Lingkungan kerja Borland Delphi dapat dilihat pada gambar di bawah ini.

Kata Kunci: Delphi dan Object Oriented Programming (OOP).

A. INTRODUCTION

Pengenalan Pemrograman Delphi dan OOP.

Delphi adalah suatu program berbasis bahasa Pascal yang berjalan dalam lingkungan Windows. Delphi telah memanfaatkan suatu teknik pemrograman yang disebut RAD yang telah membuat pemrograman menjadi lebih mudah. Delphi adalah suatu bahasa pemrograman yang telah memanfaatkan metode pemrograman Object Oriented Programming (OOP). Lingkungan kerja Borland Delphi dapat dilihat pada gambar di bawah ini.



Fungsi dari elemen-elemen di samping adalah :

Object Inspector: suatu window yang berguna untuk mengatur suatu object baik properti, events dan method.

Form : Digunakan sebagai layar/window yang digunakan sebagai lembar kerja kita. Di form-lah semua komponen seperti tombol dan komponen lainnya disimpan.

Window Unit/Source Code: Window/layar yang berisi perintah-perintah yang akan dieksekusi oleh komputer. Di layar inilah kita mengisikan program-program.

Component Palette: Layar yang berisikan komponen-komponen yang dipakai dipakai dalam program kita.

Object Oriented Programming (OOP)

Perbedaan konsep pemrograman berbasis objek dengan konsep pemrograman biasa adalah sebagai berikut:

- 1. Dalam pemrograman biasa, suatu benda hanya memiliki properti (ciri) yang membentuk dirinya.
 - Contoh: Objek manusia memiliki nama, tinggi, berat, warna kulit dan lain-lain.
- 2. Dalam pemrograman berorientasi objek, suatu benda tidak hanya memiliki properti (ciri) tetapi juga memiliki method dan event.

Contoh: Objek manusia tidak hanya memiliki *properti* nama, tinggi, berat, warna kulit tetapi juga mempunyai metode / langkah yang dapat dikerjakan dan juga event (suatu kejadian yang bisa terjadi pada objek).

- > Event dapat berupa : Event ketika lapar, ketika haus, ketika ngantuk dan lain-lain.
- ➤ *Method* dapat berupa : Makan, minum, tidur.

Suatu method bisa saja dipanggil ketika suatu event terjadi. Contoh ketika manusia lapar, maka dia akan melakukan method makan.

Contoh nyata dalam pemrograman adalah sebagai berikut: Misalkan program yang kita buat memiliki sebuah form dan sebuah tombol. Kita dapat memberikan suatu perintah ke masing-masing event dari kedua objek tersebut. Misalkan ketika form diklik, maka warna dari form berganti menjadi merah, dan ketika tombol ditekan maka program akan ditutup.

B. CONTENT

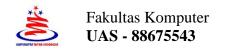
Contoh sederhana Studi Kasus:

Membuat daftar laporan berupa nama peminjam, no id, jenis id yang digunakan untuk meminjam, jumlah buku, total harga serta list buku (jumlah buku, kode dan nama buku) yang dipinjam. Ketentuan-ketentuan sebagai berikut : tiap peminjam harus menyebutkan nama, No ID, jenis ID. Peminjam berhak meminjam buku maksimal lima buah. Tiap buku yang akan dipinjam dikenakan biaya Rp100,00.

```
public class Peminjam
{
    private String nama, noId;
    private String jenisId;
    private int total;
    private int jumlah;
    private ListBuku [] lb = new
ListBuku [5];

public Peminjam (){
    this.nama = "";
    this.noId = "";
}
```

```
}
                                                total = jumlah * 100;
                                                System.out.println("Total
    public Peminjam(String nama,
                                           "+total);
String noId){
    this.nama = nama;
                                                  System.out.println("+++++++
    this.noId = noId;
                                          Daftar
                                                           List
                                                                          Buku
                                           +++++++++++");
    public Peminjam(String
                                               for(int i = 0; i < 5; i++){
                              nama,
String nold, String jenisld, int jumlah,
                                                     System.out.println((i+1)+".
int total){
                                           "+lb[i].Laporan());
    this.nama = nama;
                                                    System.out.println("+++++
    this.noId = noId;
                                           this.jenisId = jenisId;
    this.total = total;
                                             }
    this.jumlah = jumlah;
  public void setNama(String nama)
                                           public class ListBuku
                                             private String kodeBuku;
    this.nama = nama;
                                             private String namaBuku;
  public void setNoId(String noId)
                                             public ListBuku(String kode, String
    this.noId = noId;
                                           nama){
                                               this.kodeBuku = kode;
  public String getJenisId()
                                               this.namaBuku = nama;
                                             public String getKode(){
    return jenisId;
                                               return this.kodeBuku;
  public String getNama (){
    return " ";
                                             public String getNama(){
                                               return this.namaBuku;
   public void inputListBuku(int idx,
ListBuku lb){
    this.lb[idx] = lb;
                                             public String Laporan(){
                                                 return ("Buku dengan kode
                                           "+getKode()+"
    public void printDaftar(){
                                                                      judulnya
     System.out.println("*********
                                           "+getNama()+"");
System.out.println("NO
                                           }
         : "+noId+"");
ID
    System.out.println("Jenis ID
"+getJenisId());
            System.out.println("Nama
Peminjam : "+nama+"");
     System.out.println("Jumlah
: "+jumlah);
```



C. CONCLUSION

Kesimpulan dari materi ini adalah

Kesimpulan dari pembahasan diatas ivalah bahwa OOP adalah metode pemrograman yang berorientasi kepada objek. Tujuan dari OOP diciptakan adalah untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di kehidupan seharihari. Jadi setiap bagian dari suatu permasalahan adalah objek, nah objek itu sendiri merupakan gabungan dari beberapa objek yang lebih kecil lagi. Saya ambil contoh Pesawat, Pesawat adalah sebuah objek. Pesawat itu sendiri terbentuk dari beberapa objek yang lebih kecil lagi seperti mesin, roda, baling-baling, kursi, dll. Pesawat sebagai objek yang terbentuk dari objek-objek yang lebih kecil saling berhubungan, berinteraksi. berkomunikasi dan saling mengirim kepada objek-objek pesan lainnya. Begitu juga dengan program, sebuah objek yang besar dibentuk dari beberapa objek yang lebih kecil, objek-objek itu saling berkomunikasi, dan saling berkirim pesan kepada objek yang lain.

Pemrograman berorientasi objek merupakan paradigma pemrograman yang berorientasikan kepada objek. Semua data dan fungsi dalam paradigma ini dibungkus dalam kelas-kelas atau objek-objek. Bandingkan dengan logika pemrograman terstruktur. Setiap objek dapat menerima pesan, memproses data, dan mengirim pesan ke objek lainnya.

D. DISCUSSION

Hasil diskusi dari materi ini adalah

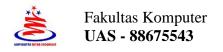
Dari hasil diaskusi materi ini bahwa OOP adalah suatu metode pemrograman yang berorientasi kepada objek. Karna dengan ada mempermudah nya ini pengembangan program dengan cara mengikuti model yang telah ada di kehidupan sehari-hari. Jadi setiap bagian dari suatu permasalahan adalah objek, nah objek itu sendiri merupakan gabungan dari beberapa objek yang lebih kecil lagi.



E. REFERENCE

- [1] O. M. Febriani and A. S. Putra, "Sistem Informasi Monitoring Inventori Barang Pada Balai Riset Standardisasi Industri Bandar Lampung," *J. Inform.*, vol. 13, no. 1, pp. 90–98, 2014.
- [2] A. S. Putra, "Paperplain: Execution Fundamental Create Application With Borland Delphi 7.0 University Of Mitra Indonesia," 2018.
- [3] A. S. Putra, "2018 Artikel Struktur Data, Audit Dan Jaringan Komputer," 2018.
- [4] A. S. Putra, "ALIAS MANAGER USED IN DATABASE DESKTOP STUDI CASE DB DEMOS."
- [5] A. S. Putra,
 "COMPREHENSIVE SET OF
 PROFESSIONAL FOR
 DISTRIBUTE COMPUTING."
- [6] A. S. Putra, "DATA ORIENTED RECOGNITION IN BORLAND DELPHI 7.0."
- [7] A. S. Putra, "EMBARCADERO DELPHI XE 2 IN GPU-POWERED FIREMONKEY APPLICATION."
- [8] A. S. Putra, "HAK ATAS KEKAYAAN INTELEKTUAL DALAM DUNIA TEKNOLOGY BERBASIS REVOLUSI INDUSTRI 4.0."
- [9] A. S. Putra, "IMPLEMENTASI PERATURAN PERUNDANGAN UU. NO 31 TAHUN 2000 TENTANG DESAIN INDUSTRI BERBASIS INFORMATION TECHNOLOGY."

- [10] A. S. Putra, "IMPLEMENTATION OF PARADOX DBASE."
- [11] A. S. Putra,
 "IMPLEMENTATION OF
 TRADE SECRET CASE
 STUDY SAMSUNG MOBILE
 PHONE."
- [12] A. S. Putra,
 "IMPLEMENTATION
 PATENT FOR APPLICATION
 WEB BASED CASE STUDI
 WWW. PUBLIKLAMPUNG.
 COM."
- [13] A. S. Putra,
 "IMPLEMENTATION
 SYSTEM FIRST TO INVENT
 IN DIGITALLY INDUSTRY."
- [14] A. S. Putra, "MANUAL REPORT & INTEGRATED DEVELOPMENT ENVIRONMENT BORLAND DELPHI 7.0."
- [15] A. S. Putra, "PATENT AS RELEVAN SUPPORT RESEARCH."
- [16] A. S. Putra, "PATENT FOR RESEARCH STUDY CASE OF APPLE. Inc."
- [17] A. S. Putra, "PATENT PROTECTION FOR APPLICATION INVENT."
- [18] A. S. Putra, "QUICK REPORT IN PROPERTY PROGRAMMING."
- [19] A. S. Putra, "REVIEW CIRCUIT LAYOUT COMPONENT REQUIREMENT ON ASUS NOTEBOOK."
- [20] A. S. Putra, "REVIEW TRADEMARK PATENT FOR INDUSTRIAL TECHNOLOGY BASED 4.0."
- [21] A. S. Putra, "TOOLBAR



- COMPONENT PALLETTE IN OBJECT ORIENTED PROGRAMMING."
- [22] A. S. Putra, "WORKING DIRECTORY SET FOR PARADOX 7."
- [23] A. S. Putra, "ZQUERY CONNECTION IMPLEMENTED PROGRAMMING STUDI CASE PT. BANK BCA Tbk."
- [24] A. S. Putra, D. R. Aryanti, and I. Hartati, "Metode SAW (Simple Additive Weighting) sebagai Sistem Pendukung Keputusan Guru Berprestasi (Studi Kasus: SMK Global Surya)," in *Prosiding Seminar Nasional Darmajaya*, 2018, vol. 1, no. 1, pp. 85–97.
- [25] A. S. Putra and O. M. Febriani, "Knowledge Management Online Application in PDAM Lampung Province." **Prosiding International** conference Information on*Technology* **Business** and (ICITB), 2018, pp. 181–187.
- [26] A. S. Putra, O. M. Febriani, and B. Bachry, "Implementasi Genetic Fuzzy System Untuk Mengidentifikasi Hasil Curian Kendaraan Bermotor Di Polda Lampung," *SIMADA (Jurnal Sist. Inf. dan Manaj. Basis Data)*, vol. 1, no. 1, pp. 21–30, 2018.
- [27] A. S. Putra, H. Sukri, and K. Zuhri, "Sistem Monitoring Realtime Jaringan Irigasi Desa (JIDES) Dengan Konsep Jaringan Sensor Nirkabel," *IJEIS (Indonesian J. Electron. Instrum. Syst.*, vol. 8, no. 2, pp. 221–232.

[28] D. P. Sari, O. M. Febriani, and A. S. Putra, "Perancangan Sistem Informasi SDM Berprestasi pada SD Global Surya," in *Prosiding Seminar Nasional Darmajaya*, 2018, vol. 1, no. 1, pp. 289–294.