

**PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBYEK
SEMESTER GENAP T.A 2020/2021
LAPORAN PROYEK AKHIR**



DISUSUN OLEH :

NIM	: 123180069 123180073
NAMA	: FEBI DWI ANDRIANSYAH HILMY HAIDAR
KELAS/PLUG	: A
NAMA ASISTEN	: MONICA AMEILIANA ANDISTA AMIRA SALSABILA

**PROGRAM STUDI INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK INDUSTRI
UNIVERSITAS PEMBANGUNAN NASIONAL "VETERAN"
YOGYAKARTA
2020**

HALAMAN PENGESAHAN

LAPORAN PROYEK AKHIR

Disusun oleh :

Febi Dwi Andriansyah
123180069

Hilmy Haidar
123180073

Telah Diperiksa dan Disetujui oleh Asisten Praktikum Pemrograman Berorientasi Obyek
Pada Tanggal :.....

Asisten Praktikum

Asisten Praktikum

Monica Ameiliana Andista M
NIM. 124180041

Amira Salsabila
NIM. 124180067

Mengetahui,
Ka. Lab. Geoinformatika

Budi Santosa, S.Si.,M.T.
NIK. 2 7009 02 0234 1

KATA PENGANTAR

Asslamualaikum warrohmatullahi wabarrokatuuh

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa yang senantiasa mencurahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan praktikum Pemrograman Berorientasi Objek serta laporan proyek akhir praktikum yang berjudul Pendataan Service Laptop. Adapun laporan ini berisi tentang proyek akhir yang penulis pilih dari hasil pembelajaran selama praktikum berlangsung.

Tidak lupa ucapan terimakasih kepada asisten laboratorium yang selalu membimbing dan mengajari penulis dalam melaksanakan praktikum dan dalam menyusun laporan ini. Laporan ini masih sangat jauh dari kesempurnaan, oleh karena itu kritik serta saran yang membangun penulis harapkan untuk menyempurnakan laporan akhir ini.

Atas perhatian dari semua pihak yang membantu penulisan ini, penulis ucapkan terimakasih. Semoga laporan ini dapat dipergunakan seperlunya.

Yogyakarta, 15 Mei 2020

Penyusun

DAFTAR ISI

HALAMAN PENGESAHAN	ii
KATA PENGANTAR	iii
DAFTAR ISI	iv
BAB I PENDAHULUAN	1
1.1 Dasar Teori.....	1
BAB II PEMBAHASAN	3
2.1 Listing Program	3
2.2 Pembahasan	44
2.3 Output	47
BAB III JADWAL Pengerjaan dan Pembagian Tugas	57
3.1 Jadwal Pengerjaan dan Pembagian	57
BAB IV KESIMPULAN DAN SARAN	58
4.1 Kesimpulan	58
4.2 Saran	58
DAFTAR PUSTAKA	59

BAB I

PENDAHULUAN

1.1 Dasar Teori

1.1.1 Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal. Aplikasi-aplikasi berbasis java umumnya dikompilasi ke dalam p-code (*bytecode*) dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM). Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (*general purpose*), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin.

1.1.2 OOP (OBJECT ORIENTED PROGRAMING)

Pengertian OOP (Object Oriented Programming) adalah suatu metode pemrograman yang berorientasi kepada objek. Tujuan dari OOP diciptakan adalah untuk mempermudah 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.

1.1.3 MVC (Model View Controller)

Model View Controller atau MVC adalah sebuah metode untuk membuat sebuah aplikasi dengan memisahkan data (Model) dari tampilan (View) dan cara bagaimana memprosesnya (Controller). Dalam implementasinya kebanyakan kerangka kerja (*framework*) dalam aplikasi web adalah berbasis arsitektur MVC. MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, antarmuka pengguna, dan bagian yang menjadi kontrol dalam sebuah aplikasi web.

1.1.4 INTELLIJ IDEA COMMUNITY EDITION 2019.3.2

IntelliJ Idea Community Edition adalah sebuah proyek software OpenSource. Java IDE komersial oleh JetBrains. IntelliJ IDEA Community Edition menyediakan semua alat yang Anda butuhkan untuk Java, Groovy, Kotlin, Scala, dan Android. Menawarkan penyelesaian kode instan dan pintar, analisis kode on-the-fly, dan alat refactoring yang andal. Alat-alat yang sangat penting untuk misi seperti sistem kontrol versi terintegrasi dan berbagai bahasa dan kerangka kerja yang didukung sudah tersedia - tidak ada plugin yang sibuk.

1.1.5 XAMPP

Xampp adalah sebuah paket kumpulan software yang terdiri dari Apache, Mysql, phpmyadmin, PHP, Perl, Freetype2, dan lain-lain. Instalasi Mysql dan Apache langsung dapat dilakukan melalui Xampp tanpa menginstall aplikasi-aplikasi tersebut satu persatu. Setelah berhasil menginstall Xampp, kita dapat langsung mengaktifkan Mysql dengan cara mengaktifkan Xampp.

1.1.6 Database MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang multialur, multipengguna, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus di mana penggunaannya tidak cocok dengan penggunaan GPL.

BAB II

PEMBAHASAN

2.1 Listing Program

2.1.1 Package Controller

```
package Controller;

import MVC.MVCTransaksi;
import Model.Model;
import View.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import javax.swing.JOptionPane;
import javax.swing.JTable;

public class Controller {
    String nama, id_bagian, username, password;
    String nama_service, id_jenis, harga;

    public Controller(tampilanSebelumMain tampilansebelummain, Model model, tampilanMain
        tampilanmain, tampilanMain2 tampilanmain2, tampilanCreatePegawai
        tampilancreatepegawai, tampilanLihatPegawai tampilanlihatpegawai,
        tampilanUpdatePegawai tampilanupdatepegawai, tampilanCreateService
        tampilancreateservice, tampilanLihatService tampilanlihatservice,
        tampilanUpdateService tampilanupdateservice) {

        //ngisi data
        if (model.getContact() != 0) {
            // menampilkan tabel di awal
            String datakontak[][] = model.readContact();
            tampilanlihatpegawai.tabel.setModel((new JTable(datakontak,
                tampilanlihatpegawai.namaKolom)).getModel());
            tampilanlihatpegawai.tabel.getColumnModel().getColumn(4).setMinWidth(0);
            tampilanlihatpegawai.tabel.getColumnModel().getColumn(4).setMaxWidth(0); {

            else {
                int input = JOptionPane.showConfirmDialog(null, "Kontak belum ada,
                    "Pilih opsi", JOptionPane.YES_NO_OPTION);
                if (input == 0) {
                    tampilanmain.dispose();
                    ;
                    tampilancreatepegawai.setVisible(true);
                } else {
                    JOptionPane.showMessageDialog(null, "Tidak jadi buat kontak");
                }
            }
        }
    }
}
```

Listing Program 2.1.1.1 Listing Program Controller

```

if (model.getContacts() != 0) {
    // menampilkan tabel di awal
    String datakontak[][] = model.readContacts();
    tampilanlihat.service.tabel.setModel((new JTable(datakontak,
    tampilanlihat.service.namaKoloms)).getModel());
    tampilanlihat.service.tabel.getColumnModel().getColumn(3).setMinWidth(0);
    tampilanlihat.service.tabel.getColumnModel().getColumn(3).setMaxWidth(0);
} else {
    int input = JOptionPane.showConfirmDialog(null, "Kontak belum ada, apakah anda ingin membuat
    kontak baru ? ", "Pilih opsi", JOptionPane.YES_NO_OPTION);
    if (input == 0) {
        tampilanmain2.dispose();
        ;
        tampilancreateservice.setVisible(true);
    }
} else {
    JOptionPane.showMessageDialog(null, "Tidak jadi buat kontak");
}}
tampilansebelummain.btnPegawai.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilanmain.setVisible(true);
        tampilansebelummain.setVisible(false);
    }
});

tampilansebelummain.btnService.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilanmain2.setVisible(true);
        tampilansebelummain.setVisible(false);
    }
});

tampilansebelummain.btnTransaksi.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        MVCTransaksi mvcTransaksi = new MVCTransaksi();
        /* view.setVisible(true); */
        tampilansebelummain.setVisible(false);
    }
});

tampilanmain.btnbuatKontak.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilancreatepegawai.setVisible(true);
        tampilanmain.setVisible(false);
    }
});

```



```

tampilanmain2.btnbuatKontaks.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilancreateservice.setVisible(true);
        tampilanmain.setVisible(false);
    }
});

tampilancreatepegawai.btntambah.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        String nama = tampilancreatepegawai.tfnama.getText();
        String id_bagian = tampilancreatepegawai.tfid_bagian.getText();
        String username = tampilancreatepegawai.tfusername.getText();
        String password = tampilancreatepegawai.tfpasword.getText();
        model.insertContact(nama, id_bagian, username, password);
        String contact[][] = model.readContact();
        tampilanlihatpegawai.tabel.setModel(new JTable(contact,
            tampilanlihatpegawai.namaKolom).getModel());
        tampilanlihatpegawai.tabel.getColumnModel().getColumn(4).setMinWidth(0);
        tampilanlihatpegawai.tabel.getColumnModel().getColumn(4).setMaxWidth(0);
        tampilancreatepegawai.setVisible(false);
        tampilanlihatpegawai.setVisible(true);
        tampilancreatepegawai.tfpasword.setText(null);
        tampilancreatepegawai.tfnama.setText(null);
        tampilancreatepegawai.tfid_bagian.setText(null);
        tampilancreatepegawai.tfusername.setText(null);
    }
});

tampilancreateservice.btntambahs.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        String nama_service = tampilancreateservice.tfnama_service.getText();
        String id_jenis = tampilancreateservice.tfid_jenis.getText();
        String harga = tampilancreateservice.tfharga.getText();
        model.insertContacts(nama_service, id_jenis, harga);
        String contact[][] = model.readContacts();
        tampilanlihatservice.tabel.setModel(new JTable(contact,
            tampilanlihatservice.namaKoloms).getModel());
        tampilanlihatservice.tabel.getColumnModel().getColumn(3).setMinWidth(0);
        tampilanlihatservice.tabel.getColumnModel().getColumn(3).setMaxWidth(0);
        tampilancreateservice.setVisible(false);
        tampilanlihatservice.setVisible(true);
        tampilancreateservice.tfnama_service.setText(null);
        tampilancreateservice.tfid_jenis.setText(null);
        tampilancreateservice.tfharga.setText(null);
    }
});

tampilancreatepegawai.btnbatal.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilancreatepegawai.setVisible(false);
        tampilanmain.setVisible(true);
    }
});

```

Listing Program 2.1.1.2 Lanjutan *Listing Program Controller*

```

tampilancreateservice.btnbatal.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilancreateservice.setVisible(false);
        tampilanmain2.setVisible(true);
    }
});

tampilanmain.btnlihatKontak.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilanmain.setVisible(false);
        tampilanlihatpegawai.setVisible(true);
    }
});

tampilanmain2.btnlihatKontaks.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilanmain2.setVisible(false);
        tampilanlihatservice.setVisible(true);
    }
});

tampilanlihatpegawai.btntambah.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilanlihatpegawai.setVisible(false);
        tampilancreatepegawai.setVisible(true);
    }
});

tampilanlihatservice.btntambahs.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilanlihatservice.setVisible(false);
        tampilancreateservice.setVisible(true);
    }
});

tampilanlihatpegawai.btnkembali.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilanlihatpegawai.setVisible(false);
        tampilanmain.setVisible(true);
    }
});

```

Listing Program 2.1.1.3 Lanjutan *Listing Program Controller*

```

tampilanlihatpegawai.tabel.addMouseListener(new MouseAdapter() {

    @Override
    public void mouseClicked(MouseEvent e) {
        super.mousePressed(e);
        int baris = tampilanlihatpegawai.tabel.getSelectedRow();
        int kolom = tampilanlihatpegawai.tabel.getSelectedColumn();
        String dataterpilih = tampilanlihatpegawai.tabel.getValueAt(baris, 0).toString();
        System.out.println(dataterpilih);
        tampilanlihatpegawai.lnamaisi.setText(tampilanlihatpegawai.tabel.
            getValueAt(baris, 0).toString());
        tampilanlihatpegawai.lid_bagianisi.setText(tampilanlihatpegawai.
            tabel.getValueAt(baris, 1).toString());
        tampilanlihatpegawai.lusernameisi.setText(tampilanlihatpegawai.
            tabel.getValueAt(baris, 2).toString());
        tampilanlihatpegawai.lpasswordisi.setText(tampilanlihatpegawai.
            tabel.getValueAt(baris, 3).toString());
        nama = tampilanlihatpegawai.lnamaisi.getText();
        id_bagian = tampilanlihatpegawai.lid_bagianisi.getText();
        username = tampilanlihatpegawai.lusernameisi.getText();
        password = tampilanlihatpegawai.lpasswordisi.getText();
        tampilanlihatpegawai.btnupdate.setEnabled(true);
        tampilanlihatpegawai.btnhapus.setEnabled(true); });

tampilanlihatservice.tabel.addMouseListener(new MouseAdapter() {
    @Override
    public void mouseClicked(MouseEvent e) {
        super.mousePressed(e);
        int baris = tampilanlihatservice.tabel.getSelectedRow();
        int kolom = tampilanlihatservice.tabel.getSelectedColumn();
        String dataterpilih = tampilanlihatservice.tabel.getValueAt(baris, 0).toString();
        System.out.println(dataterpilih);
        tampilanlihatservice.lnama_serviceisi.setText(tampilanlihatservice.
            .tabel.getValueAt(baris, 0).toString());
        tampilanlihatservice.lid_jenisisi.setText(tampilanlihatservice.tabel.
            getValueAt(baris, 1).toString());
        tampilanlihatservice.lhargaisi.setText(tampilanlihatservice.tabel.
            getValueAt(baris, 2).toString());
        nama_service = tampilanlihatservice.lnama_serviceisi.getText();
        id_jenis = tampilanlihatservice.lid_jenisisi.getText();
        harga = tampilanlihatservice.lhargaisi.getText();
        tampilanlihatservice.btnupdates.setEnabled(true);
        tampilanlihatservice.btnhapuss.setEnabled(true); });

tampilanlihatpegawai.btnsearch.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        String cari = tampilanlihatpegawai.tfsearch.getText();
        String data[][] = model.getContactSearch(cari);
        tampilanlihatpegawai.tabel.setModel(new JTable(data,
            tampilanlihatpegawai.namaKolom).getModel());
        tampilanlihatpegawai.tabel.getColumnModel().getColumn(4).setMinWidth(0);
        tampilanlihatpegawai.tabel.getColumnModel().getColumn(4).setMaxWidth(0);
    }
});

```

Listing Program 2.1.1.4 Lanjutan *Listing* Program Controller

```

tampilanlihat.service.btnsearchs.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        String cari = tampilanlihat.service.tfsearchs.getText();
        String datas[][] = model.getContactSearchs(cari);
        tampilanlihat.service.tabel.setModel(new JTable(datas,
            tampilanlihat.service.namaKoloms).getModel());
        tampilanlihat.service.tabel.getColumnModel().getColumn(3).setMinWidth(0);
        tampilanlihat.service.tabel.getColumnModel().getColumn(3).setMaxWidth(0); } });

tampilanlihat.pegawai.btnhapus.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {

        String id = tampilanlihat.pegawai.lid.getText();
        model.hapusKontak(username);
        String datakontak[][] = model.readContact();
        tampilanlihat.pegawai.tabel.setModel(new JTable(datakontak,
            tampilanlihat.pegawai.namaKolom).getModel());
        tampilanlihat.pegawai.tabel.getColumnModel().getColumn(4).setMinWidth(0);
        tampilanlihat.pegawai.tabel.getColumnModel().getColumn(4).setMaxWidth(0);
        tampilanlihat.pegawai.lnamaisi.setText(null);
        tampilanlihat.pegawai.lpasswordisi.setText(null);
        tampilanlihat.pegawai.lusernameisi.setText(null);
        tampilanlihat.pegawai.lid_bagianisi.setText(null);
        tampilanlihat.pegawai.tfsearch.setText(null); } });

tampilanlihat.service.btnhapuss.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        String id = tampilanlihat.pegawai.lid.getText();
        model.hapusKontaks(id_jenis);
        String datakontak[][] = model.readContacts();
        tampilanlihat.service.tabel.setModel(new JTable(datakontak,
            tampilanlihat.service.namaKoloms).getModel());
        tampilanlihat.service.tabel.getColumnModel().getColumn(3).setMinWidth(0);
        tampilanlihat.service.tabel.getColumnModel().getColumn(3).setMaxWidth(0);
        tampilanlihat.service.lnama_serviceisi.setText(null);
        tampilanlihat.service.lhargaisi.setText(null);
        tampilanlihat.service.lid_jenisisi.setText(null);
        tampilanlihat.service.tfsearchs.setText(null); } });

tampilanlihat.pegawai.btnupdate.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {

        tampilanupdate.pegawai.setVisible(true);
        tampilanlihat.pegawai.setVisible(false);
        tampilanupdate.pegawai.tfnama.setText(nama);
        tampilanupdate.pegawai.tfid_bagian.setText(id_bagian);
        tampilanupdate.pegawai.tfusername.setText(username);
        tampilanupdate.pegawai.tfpasword.setText(password);
    } });

```

Listing Program 2.1.1.5 Lanjutan *Listing Program Controller*

```

tampilanlihat.service.btnupdates.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {

        tampilanupdateservice.setVisible(true);
        tampilanlihat.service.setVisible(false);
        tampilanupdateservice.tfnama_service.setText(nama_service);
        tampilanupdateservice.tfid_jenis.setText(id_jenis);
        tampilanupdateservice.tfharga.setText(harga);

    }
});

tampilanlihatpegawai.btnrefresh.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        String contact[][] = model.readContact();

        tampilanlihatpegawai.tabel.setModel(new JTable(contact,
            tampilanlihatpegawai.namaKolom).getModel());
        tampilanlihatpegawai.tabel.getColumnModel().getColumn(4).setMinWidth(0);
        tampilanlihatpegawai.tabel.getColumnModel().getColumn(4).setMaxWidth(0);
        tampilanlihatpegawai.lnamaisi.setText(null);
        tampilanlihatpegawai.lid_bagianisi.setText(null);
        tampilanlihatpegawai.lpasswordisi.setText(null);
        tampilanlihatpegawai.lusernameisi.setText(null);
        tampilanlihatpegawai.btnupdate.setEnabled(false);
        tampilanlihatpegawai.btnhapus.setEnabled(false);
        tampilanlihatpegawai.tfsearch.setText(null);

    }
});

tampilanlihat.service.btnrefreshs.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        String contacts[][] = model.readContacts();

        tampilanlihat.service.tabel.setModel(new JTable(contacts,
            tampilanlihat.service.namaKoloms).getModel());
        tampilanlihat.service.tabel.getColumnModel().getColumn(3).setMinWidth(0);
        tampilanlihat.service.tabel.getColumnModel().getColumn(3).setMaxWidth(0);
        tampilanlihat.service.lnama_serviceisi.setText(null);
        tampilanlihat.service.lid_jenisisi.setText(null);
        tampilanlihat.service.lpasswordisi.setText(null);
        tampilanlihat.service.lhargaisi.setText(null);
        tampilanlihat.service.btnupdates.setEnabled(false);
        tampilanlihat.service.btnhapuss.setEnabled(false);
        tampilanlihat.service.tfsearchs.setText(null);

    }
});

```

Listing Program 2.1.1.6 Lanjutan *Listing Program Controller*

```

tampilanmain.btnexit.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        //JOptionPane.showMessageDialog(null, "Shutting Down...");
        tampilansebelummain.setVisible(true);
        tampilanmain.dispose();
        tampilancreatepegawai.dispose();
        tampilanlihatpegawai.dispose();
        tampilanupdatepegawai.dispose();
    }
});

tampilanmain2.btnexit.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        //JOptionPane.showMessageDialog(null, "Shutting Down...");
        tampilansebelummain.setVisible(true);
        tampilanmain2.dispose();
        tampilancreateservice.dispose();
        tampilanlihatservice.dispose();
        tampilanupdateservice.dispose();
    }
});

tampilanupdatepegawai.btnkembali.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilanupdatepegawai.setVisible(false);
        tampilanlihatpegawai.setVisible(true);
    }
});

tampilanupdateservice.btnkembalis.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilanupdateservice.setVisible(false);
        tampilanlihatservice.setVisible(true);
    }
});

tampilanupdatepegawai.btncancel.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        String nama = tampilanupdatepegawai.tfnama.getText();
        String id_bagian = tampilanupdatepegawai.tfid_bagian.getText();
        String username = tampilanupdatepegawai.tfusername.getText();
        String password = tampilanupdatepegawai.tfpasword.getText();
        String id = tampilanlihatpegawai.lid.getText();
        model.updateKontak(nama, id_bagian, username, password);
        String contact[][] = model.readContact();
        tampilanlihatpegawai.tabel.setModel(new JTable(contact,

```

Listing Program 2.1.1.7 Lanjutan *Listing Program Controller*

```

tampilanlihatpegawai.namaKolom).getModel());
tampilanlihatpegawai.tabel.getColumnModel().getColumn(4).setMinWidth(0);
tampilanlihatpegawai.tabel.getColumnModel().getColumn(4).setMaxWidth(0);
tampilanlihatpegawai.lnamaisi.setText(nama);
tampilanlihatpegawai.lid_bagianisi.setText(id_bagian);
tampilanlihatpegawai.lusernameisi.setText(username);
tampilanlihatpegawai.lpasswordisi.setText(password);
tampilanupdatepegawai.setVisible(false);
tampilanlihatpegawai.setVisible(true); } });

tampilanupdateservice.btnefits.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        String nama_service = tampilanupdateservice.tfnama_service.getText();
        String id_jenis = tampilanupdateservice.tfid_jenis.getText();
        String harga = tampilanupdateservice.tfharga.getText();

        model.updateKontaks(nama_service, id_jenis, harga);
        String contacts[][] = model.readContacts();
        tampilanlihatservice.tabel.setModel(new JTable(contacts,
            tampilanlihatservice.namaKoloms).getModel());
        tampilanlihatservice.tabel.getColumnModel().getColumn(3).setMinWidth(0);
        tampilanlihatservice.tabel.getColumnModel().getColumn(3).setMaxWidth(0);
        tampilanlihatservice.lid_jenisisi.setText(id_jenis);
        tampilanlihatservice.lnama_serviceisi.setText(nama_service);
        tampilanlihatservice.lhargaisi.setText(harga);
        tampilanupdateservice.setVisible(false);
        tampilanlihatservice.setVisible(true);
    } });
}
}

```

Listing Program 2.1.1.8 Lanjutan *Listing* Program Controller

```

package Controller;

import MVC.MVC;
import Model.DAOLogin;
import Model.ModelLogin;
import View.*;

import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class ControllerLogin {
    ModelLogin modelLogin;
    ViewLogin viewLogin;
    DAOLogin daoLogin;

    public ControllerLogin(ModelLogin modelLogin, ViewLogin viewLogin, DAOLogin daoLogin) {
        this.modelLogin = modelLogin;
        this.viewLogin = viewLogin;
        this.daoLogin = daoLogin;

        viewLogin.bLogin.addActionListener(new ActionListener() {

```

```

@Override
public void actionPerformed(ActionEvent e) {
    String username = viewLogin.getUser();
    String password = viewLogin.getPassword();
    if (username.isEmpty() || password.isEmpty()) {
        JOptionPane.showMessageDialog(null, "Harap isi semua field");
    } else {
        modelLogin.setModelLogin(username, password);
        boolean login = daoLogin.login(modelLogin);
        if (login) {
            MVC mvc = new MVC();
            viewLogin.setVisible(false);
            JOptionPane.showMessageDialog(null, "Login Berhasil");
        } else {
            JOptionPane.showMessageDialog(null, "Username atau Password Salah");
        }
    }
}
}
}
}
}

```

Listing Program 2.1.1.9 *Listing Program ControllerLogin*

```

package Controller;

import Model.ModelTransaksi;
import View.*;
import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;

public class ControllerTransaksi {
    ModelTransaksi modelTransaksi;
    tampilanSebelumMain tampilanSebelumMain;
    View view;
    /*ViewLogin viewLogin;*/
    String dataterpilih = null;
    String dataterpilih1 = null;
    String dataterpilih2 = null;
    int baris,kolom;

    public ControllerTransaksi(ModelTransaksi modelTransaksi, View view){
        this.modelTransaksi = modelTransaksi;
        this.view = view;
    }
    if (modelTransaksi.getBanyakData() != 0) {
        String data[][] = modelTransaksi.readData();
        String data1[][] = modelTransaksi.readService();
        String data2[][] = modelTransaksi.readPegawai();
        view.tabel.setModel((new JTable(data, view.namaKolom)).getModel());
        view.tabel1.setModel((new JTable(data1, view.namaKolom1)).getModel());
        view.tabel2.setModel((new JTable(data2, view.namaKolom2)).getModel());
    } else {
        JOptionPane.showMessageDialog(null, "Data Tidak Ada");
    }
}

```

Listing Program 2.1.1.10 *Listing Program ControllerTransaksi*


```

view.tabel.addMouseListener(new MouseAdapter() {
    @Override
    public void mouseClicked(MouseEvent e) {
        int pilih = view.tabel.getSelectedRow();
        if (pilih == 1) {
            return;
        }
        String username = (String) view.tabel.getValueAt(pilih, 1);
        view.tfUsername.setText(username);
        String nama_pelanggan = (String) view.tabel.getValueAt(pilih, 2);
        view.tfNama_pelanggan.setText(nama_pelanggan);
        String no_hp = (String) view.tabel.getValueAt(pilih, 3);
        view.tfNo_hp.setText(no_hp);
        String total_biaya = (String) view.tabel.getValueAt(pilih, 4);
        view.tfTotal_biaya.setText(total_biaya);
        super.mousePressed(e);
        baris = view.tabel.getSelectedRow();
        kolom = view.tabel.getSelectedColumn();

        dataterpilih = view.tabel.getValueAt(baris, 0).toString();

        System.out.println(dataterpilih);

        //int input = JOptionPane.showConfirmDialog(null, "Apa anda ingin menghapus NIM" +
        dataterpilih + "?", "Pilih Opsi...", JOptionPane.YES_NO_OPTION);

        if (dataterpilih != null) {
            view.bDelete.setEnabled(true);
            /* view.bUpdate.setEnabled(true);*/
        }
    }
});

view.tabel1.addMouseListener(new MouseAdapter() {
    @Override
    public void mouseClicked(MouseEvent e) {
        int pilih = view.tabel1.getSelectedRow();
        if (pilih == 1) {
            return;
        }
        String id_jasa = (String) view.tabel1.getValueAt(pilih, 1);
        view.tfId_jenis.setText(id_jasa);
        String harga = (String) view.tabel1.getValueAt(pilih, 2);
        view.tfTotal_biaya.setText(harga);
        super.mousePressed(e);
        baris = view.tabel1.getSelectedRow();
        kolom = view.tabel1.getSelectedColumn();

        dataterpilih1 = view.tabel1.getValueAt(baris, 0).toString();

        System.out.println(dataterpilih1);
    }
});

```

Listing Program 2.1.1.11 Listing Program ControllerTransaksi

```

view.tabel2.addMouseListener(new MouseAdapter() {
    @Override
    public void mouseClicked(MouseEvent e) {
        int pilih = view.tabel2.getSelectedRow();
        if (pilih == 1) {
            return;
        }
        String username = (String) view.tabel2.getValueAt(pilih, 1);
        view.tfUsername.setText(username);
        super.mousePressed(e);
        baris = view.tabel2.getSelectedRow();
        kolom = view.tabel2.getSelectedColumn();

        dataterpilih2 = view.tabel2.getValueAt(baris, 0).toString();

        System.out.println(dataterpilih2); });

view.bCreate.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        String username = view.getUsername();
        String id_jenis = view.getId_jenis();
        String nama_pelanggan = view.getNama_pelanggan();
        String no_hp = view.getNo_hp();
        int total_biaya = view.getTotal_biaya();
        modelTransaksi.insertData(username, id_jenis, nama_pelanggan, no_hp, total_biaya);

        String data[][] = modelTransaksi.readData();
        view.tabel.setModel(new JTable(data, view.namaKolom.getModel()));
    }
});

view.bDelete.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent ae) {
        int baris = view.tabel.getSelectedRow();

        if (baris >= 0) {
            String dataTerpilihX = view.tabel.getValueAt(baris, 0).toString();
            int dataTerpilih = Integer.parseInt(dataTerpilihX);
            int input = JOptionPane.showConfirmDialog(null, "Apa anda ingin menghapus ID : "
                + dataTerpilih + "?", "Pilih Opsi...", JOptionPane.YES_NO_OPTION);

            if (input == 0) {
                modelTransaksi.deleteData(dataTerpilih);
            } else {
                JOptionPane.showMessageDialog(null, "Tidak Jadi Dihapus");
            }
        } else {
            JOptionPane.showMessageDialog(null, "Pilih data yang akan didelete dahulu");
        }
    }
});

```

Listing Program 2.1.1.12 *Listing Program ControllerTransaksi*

```

view.bUpdate.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        int baris = view.tabel.getSelectedRow();

        if (baris >= 0) {
            String dataTerpilihX = view.tabel.getValueAt(baris, 1).toString();
            int dataTerpilih = Integer.parseInt(dataTerpilihX);

            int input = JOptionPane.showConfirmDialog(null, "Apa anda ingin UPDATE ID :
            " + dataTerpilih + "?", "Pilih Opsi...", JOptionPane.YES_NO_OPTION);

            if (input == 0) {
                int no_transaksi = view.getno_transaksi();
                String nama_pelanggan = view.getNama_pelanggan();
                String no_hp = view.getNo_hp();
                int total_biaya = view.getTotal_biaya();

                modelTransaksi.updateData(dataTerpilih, nama_pelanggan, no_hp, total_biaya);
            } else {
                JOptionPane.showMessageDialog(null, "Tidak Jadi Diupdate");
            }
        } else {
            JOptionPane.showMessageDialog(null, "Pilih data yang akan diupdate dahulu");
        }
    }
});

view.bSearch.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        String search = view.tfSearch.getText();
        String datas[][] = modelTransaksi.search(search);
        view.tabel.setModel((new JTable(datas, view.namaKolom)).getModel());
    }
});

view.bRefresh.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        String datas[][] = modelTransaksi.readData();
        view.tfno_transaksi.setText("");
        view.tfNama_pelanggan.setText("");
        view.tfNo_hp.setText("");
        view.tfTotal_biaya.setText("");
        view.tabel.setModel((new JTable(datas, view.namaKolom)).getModel());
    }
});

view.bExit.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        int input = JOptionPane.showConfirmDialog(null, "Apa anda yakin ingin keluar?",
        "Pilih Opsi...", JOptionPane.YES_NO_OPTION);
        if (input == 0) {
            System.exit(0);
        }
    }
});
}

```

Listing Program 2.1.1.13 Listing Program ControllerTransaksi

```
private void updateTable() {
    String data[][] = modelTransaksi.readData();
    view.tabel.setModel(new JTable(data, view.namaKolom.getModel()));
}
}
```

Listing Program 2.1.1.14 *Listing Program ControllerTransaksi*

2.1.3 Package Model

```
package Model;

import com.mysql.jdbc.Connection;
import com.mysql.jdbc.Statement;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import javax.swing.JOptionPane;

public class Model {
    Connection koneksi ;
    Statement statement ;
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/service_laptop";
    static final String USER = "root";
    static final String PASS = "";

    public Model() {
        try {
            Class.forName(JDBC_DRIVER);
            koneksi = (Connection) DriverManager.getConnection(DB_URL, USER, PASS);
        }
        catch (Exception e) {
            JOptionPane.showMessageDialog(null, e.getMessage());
            System.out.println("Koneksi Gagal");
        }
    }

    public void insertContact (String nama , String id_bagian, String username , String password) {
        try {
            String query = "INSERT INTO `pegawai` (`username`,`id_bagian`,`nama`,`password`) "
                + "VALUES ('" + username + "','" + id_bagian + "','" + nama + "','" + password + "')";
            statement = (Statement) koneksi.createStatement();
            statement.executeUpdate(query);
            JOptionPane.showMessageDialog(null, "Berhasil Menambahkan Kontak!");
        }
        catch(SQLException e) {
            JOptionPane.showMessageDialog(null, e.getMessage());
        }
    }
}
```

Listing Program 2.1.2.1 *Listing Program Model*

```

public void insertContacts (String nama_service , String id_jenis, String harga ) {
    try {
        String query = "INSERT INTO `jenis_service` (`id_jenis`,`nama_service`,`harga`) "
            + "VALUES ('" + id_jenis + "','" + nama_service + "','" + harga + "')";
        statement = (Statement) koneksi.createStatement();
        statement.executeUpdate(query);
        JOptionPane.showMessageDialog(null, "Berhasil Menambahkan Service!");
    }
    catch(SQLException e) {
        JOptionPane.showMessageDialog(null, e.getMessage());
    }
}

public String[][] readContact() {
    int jumlah = 0 ;
    try {
        String data[][] = new String [ getContact() ] [5];
        String query = "Select * FROM pegawai";
        ResultSet rs = statement.executeQuery(query);
        while (rs.next()) {
            data[jumlah][0] = rs.getString("nama");
            data[jumlah][1] = rs.getString("id_bagian");
            data[jumlah][2] = rs.getString("username");
            data[jumlah][3] = rs.getString("password");
            jumlah++;
        }
        return data;
    }
    catch (SQLException e) {
        System.out.println(e.getMessage());
        System.out.println("SQL Error");
        return null;
    }
}

public String[][] readContacts() {
    int jumlah = 0 ;
    try {
        String data[][] = new String [ getContacts() ] [5];
        String query = "Select * FROM jenis_service";
        ResultSet rs = statement.executeQuery(query);
        while (rs.next()) {
            data[jumlah][0] = rs.getString("nama_service");
            data[jumlah][1] = rs.getString("id_jenis");
            data[jumlah][2] = rs.getString("harga");
            jumlah++;
        }
        return data;
    }
    catch (SQLException e) {
        System.out.println(e.getMessage());
        System.out.println("SQL Error");
        return null;
    }
}

```

Listing Program 2.1.2.2 Lanjutan *Listing Program Model*

```

public int getContact () {
    int jumlahData =0 ;
    try {
        statement = (Statement) koneksi.createStatement();
        String query = "SELECT * FROM pegawai ";
        ResultSet rs = statement.executeQuery(query);
        while (rs.next() ) {
            jumlahData ++ ;
        }
        return jumlahData;
    }
    catch(SQLException e) {
        System.out.println(e.getMessage());
        System.out.println("SQL Error");
        return 0;
    }
}

public int getContacts () {
    int jumlahData =0 ;
    try {
        statement = (Statement) koneksi.createStatement();
        String query = "SELECT * FROM jenis_service ";
        ResultSet rs = statement.executeQuery(query);
        while (rs.next() ) {
            jumlahData ++ ;
        }
        return jumlahData;
    }
    catch(SQLException e) {
        System.out.println(e.getMessage());
        System.out.println("SQL Error");
        return 0;
    }
}

public void hapusKontak(String username) {
    try{
        statement = (Statement) koneksi.createStatement();
        String query = "DELETE FROM pegawai WHERE `username` = '" +username + "'";
        statement.executeUpdate(query);
        JOptionPane.showMessageDialog(null, "Berhasil di hapus");
    }
    catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e.getMessage());
        System.out.println("Tidak ada data");
    }
}

public void hapusKontaks(String id_jenis) {
    try{
        statement = (Statement) koneksi.createStatement();
        String query = "DELETE FROM jenis_service WHERE `id_jenis` = '" +id_jenis + "'";
        statement.executeUpdate(query);
        JOptionPane.showMessageDialog(null, "Berhasil di hapus");
    }
    catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e.getMessage());
        System.out.println("Tidak ada data");
    }
}
}

```

Listing Program 2.1.2.3 Lanjutan *Listing Program Model*

```

public void updateKontak (String nama , String id_bagian, String username , String password){
    try{
        statement = (Statement) koneksi.createStatement();
        String query = "UPDATE `pegawai` SET `nama`='"+nama+"', `id_bagian`='"+id_bagian+"',
        `password`='"+password+"' "+"WHERE `username`='"+username+"'";
        statement.executeUpdate(query);
        JOptionPane.showMessageDialog(null, "Data Berhasil di update");
    }
    catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e.getMessage());
    }
}

public void updateKontaks (String nama_service , String id_jenis, String harga ){
    try{
        statement = (Statement) koneksi.createStatement();
        String query = "UPDATE `jenis_service` SET `nama_service`='"+nama_service+"',
        `harga`='"+harga+"' "+"WHERE `id_jenis`='"+id_jenis+"'";
        statement.executeUpdate(query);
        JOptionPane.showMessageDialog(null, "Data Berhasil di update");
    }
    catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e.getMessage());
    }
}

public int getBanyakContactSearch (String cari) {
    int jmlh=0;
    try {
        statement = (Statement) koneksi.createStatement();
        String query = "SELECT * FROM pegawai WHERE nama like '%" +cari+"%' ";
        ResultSet rs = statement.executeQuery(query);
        while (rs.next()) {
            jmlh ++;
        }
        return jmlh ;
    }
    catch (SQLException e) {
        JOptionPane.showMessageDialog(null, "Tidak ada kontak yang di cari");
        return 0 ;
    }
}

public int getBanyakContactSearchs (String cari) {
    int jmlh=0;
    try {
        statement = (Statement) koneksi.createStatement();
        String query = "SELECT * FROM jenis_service WHERE nama_service like '%" +cari+"%' ";
        ResultSet rs = statement.executeQuery(query);
        while (rs.next()) {
            jmlh ++;
        }
        return jmlh ;
    }
    catch (SQLException e) {
        JOptionPane.showMessageDialog(null, "Tidak ada kontak yang di cari");
        return 0 ;
    }
}

```

Listing Program 2.1.2.4 Lanjutan *Listing Program Model*

```

public String[][] getContactSearch (String cari) {
    try {
        int jmlh = 0 ;
        String data[][] = new String [getBanyakContactSearch(cari)][5] ;
        String query = "SELECT * FROM pegawai WHERE nama like '%" + cari + "%' OR id_bagian like '%"
            + cari + "%' OR username like '%" + cari + "%' OR password like '%" + cari + "%' ";
        ResultSet rs = statement.executeQuery(query) ;
        while (rs.next()) {
            data[jmlh][0] = rs.getString("nama");
            data[jmlh][1] = rs.getString("id_bagian");
            data[jmlh][2] = rs.getString("username");
            data[jmlh][3] = rs.getString("password");
            jmlh++;
        }
        return data;
    }
    catch (SQLException e) {
        return null;
    }
}

public String[][] getContactSearchs (String cari) {
    try {
        int jmlh = 0 ;
        String data[][] = new String [getBanyakContactSearchs(cari)][5] ;
        String query = "SELECT * FROM jenis_service WHERE nama_service like '%" + cari + "%' OR
            id_jenis like '%" + cari + "%' OR harga like '%" + cari + "%' ";
        ResultSet rs = statement.executeQuery(query) ;
        while (rs.next()) {
            data[jmlh][0] = rs.getString("nama_service");
            data[jmlh][1] = rs.getString("id_jenis");
            data[jmlh][2] = rs.getString("harga");

        }
        return data;
    }
    catch (SQLException e) {
        return null;
    }
}
}

```

Listing Program 2.1.2.5 Lanjutan *Listing Program Model*

```

package Model;

public class ModelLogin {
    private String username;
    private String password;

    public void setModelLogin(String username, String password) {
        this.username = username;
        this.password = password;
    }

    public String getUsername() {
        return username;
    }
}

```



```

public void setUsername(String username) {
    this.username = username;
}

public String getPassword() {
    return password;
}

public void setPassword(String password) {
    this.password = password;
}
}
public String getName() {
    return name; } }

```

Listing Program 2.1.2.6 *Listing Program ModelLogin*

```

package Model;

import javax.swing.*.*;
import java.sql.*.*;

public class ModelTransaksi {
    //mengkoneksikan ke database
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/service_laptop";
    static final String USER = "root";
    static final String PASS = "";
    Connection koneksi;
    Statement statement;//untuk perintah query

    public ModelTransaksi() {
        try{
            Class.forName(JDBC_DRIVER);
            koneksi = (Connection) DriverManager.getConnection(DB_URL, USER, PASS);
            System.out.println("Koneksi Berhasil");
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(null, ex.getMessage());
            System.out.println("Koneksi Gagal");
        }
    }

    public void insertData(String username, String id_jenis, String nama_pelanggan, String no_hp,
        int total_biaya){
        try {
            String query = "INSERT INTO `transaksi`(`username`,`id_jenis`,`nama_pelanggan`,`no_hp`,`total_biaya`) VALUES ('"+username+"','"+id_jenis+"','"+nama_pelanggan+"','"+no_hp+"','"+total_biaya+"')"; //value 1 (id diskip)
            statement = (Statement) koneksi.createStatement();
            statement.executeUpdate(query); //execute querynya
            System.out.println("Berhasil ditambahkan");
            JOptionPane.showMessageDialog(null, "Data Berhasil");
        } catch (Exception sql) {
            System.out.println(sql.getMessage());
            JOptionPane.showMessageDialog(null, sql.getMessage());
        }
    }
}

```

Listing Program 2.1.2.7 *Listing Program ModelTransaksi*

```

public int getBanyakData(){//menghitung jumlah baris
    int jmlData = 0;
    try{
        statement = koneksi.createStatement();
        String query = "Select * from transaksi"; //pengambilan dara dalam java dari database
        ResultSet resultSet = statement.executeQuery(query);
        while (resultSet.next()){ //lanjut kedata selanjutnya jmlData bertambah
            jmlData++;
        }
        return jmlData;
    }catch(SQLException e){
        System.out.println(e.getMessage());
        System.out.println("SQL Error");
        return 0;
    }
}

public String[][] readData(){
    try{
        int jmlData = 0;//menampung jumlah data
        int nomor = 1;

        String data[][] = new String[getBanyakData()][7]; //baris, kolom nya ada 3

        String query = "Select * from transaksi "; //pengambilan dara dalam java dari database
        ResultSet resultSet = statement.executeQuery(query);
        while (resultSet.next()){ //lanjut kedata selanjutnya jmlData bertambah
            data[jmlData][0] = resultSet.getString("no_transaksi");
            data[jmlData][1] = resultSet.getString("username");
            data[jmlData][2] = resultSet.getString("id_jenis");
            data[jmlData][3] = resultSet.getString("nama_pelanggan");
            data[jmlData][4] = resultSet.getString("no_hp");
            data[jmlData][5] = resultSet.getString("total_biaya");
            nomor++;
            jmlData++; //barisnya berpindah terus
        }
        return data;
    }catch(SQLException e){
        System.out.println(e.getMessage());
        System.out.println("SQL Error");
        return null;
    }
}

public void deleteData(int no_transaksi) {
    try{
        String query = "DELETE FROM `transaksi` WHERE `no_transaksi` = '"+no_transaksi+"'";
        statement = koneksi.createStatement();
        statement.executeUpdate(query);
        JOptionPane.showMessageDialog(null, "Berhasil Dihapus");
    }catch(SQLException sql) {
        System.out.println(sql.getMessage());
    }
}

```

Listing Program 2.1.2.8 Lanjutan *Listing Program ModelTransaksi*

```

public void updateData(int no_transaksi, String nama_pelanggan, String no_hp, int total_biaya){
    try{

        statement = koneksi.createStatement();

        String query = "UPDATE `transaksi` SET `nama_pelanggan`='"+nama_pelanggan+"',
        `no_hp`='"+no_hp+"',`total_biaya`='"+total_biaya+"' WHERE `no_transaksi`='"+no_transaksi+"'";
        statement.executeUpdate(query);

        JOptionPane.showMessageDialog(null, "Edit Data Sukses!");

    }catch(Exception ex){
        System.out.println(ex.getMessage());
    }
}

public int getJmlDataSearch(String search) {
    int jmlData = 0;
    try {
        String query = "SELECT * FROM transaksi WHERE nama_pelanggan LIKE '%" + search + "%'";
        ResultSet resultSet = statement.executeQuery(query);
        while (resultSet.next()) {
            jmlData++;
        }

        return jmlData;
    } catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e.getMessage());

        return 0;
    }
}

public String[][] search(String search) {
    try {
        int jmlData = 0;
        int nomor = 1;
        String datas[][] = new String[getJmlDataSearch(search)][8];
        String query = "SELECT * FROM transaksi WHERE nama_pelanggan LIKE '%" + search + "%'";
        ResultSet resultSet = statement.executeQuery(query);
        while (resultSet.next()) {
            datas[jmlData][0] = resultSet.getString("no_transaksi");
            datas[jmlData][1] = resultSet.getString("username");
            datas[jmlData][2] = resultSet.getString("id_jenis");
            datas[jmlData][3] = resultSet.getString("nama_pelanggan");
            datas[jmlData][4] = resultSet.getString("no_hp");
            datas[jmlData][5] = resultSet.getString("total_biaya");
            nomor++;
            jmlData++;
        }

        return datas;
    } catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e.getMessage());

        return null;
    }
}

```

Listing Program 2.1.2.9 Lanjutan *Listing Program MVCLogin*

```

public String[][] readPegawai() {
    int jumlah = 0 ;
    try {
        String data[][] = new String [ getPegawai()] [5];
        String query = "Select * FROM pegawai";
        ResultSet rs = statement.executeQuery(query);
        while (rs.next()) {
            data[jumlah][0] = rs.getString("nama");
            data[jumlah][1] = rs.getString("username");
            jumlah++;
        }
        return data;
    }
    catch (SQLException e) {
        System.out.println(e.getMessage());
        System.out.println("SQL Error");
        return null;
    }
}

public String[][] readService() {
    int jumlah = 0 ;
    try {
        String data[][] = new String [ getService()] [4];
        String query = "Select * FROM jenis_service";
        ResultSet rs = statement.executeQuery(query);
        while (rs.next()) {
            data[jumlah][0] = rs.getString("nama_service");
            data[jumlah][1] = rs.getString("id_jenis");
            data[jumlah][2] = rs.getString("harga");
            jumlah++;
        }
        return data;
    }
    catch (SQLException e) {
        System.out.println(e.getMessage());
        System.out.println("SQL Error");
        return null;
    }
}

public int getPegawai () {
    int jumlahData =0 ;
    try {
        statement = (com.mysql.jdbc.Statement) koneksi.createStatement();
        String query = "SELECT * FROM pegawai ";
        ResultSet rs = statement.executeQuery(query);
        while (rs.next() ) {
            jumlahData ++ ;
        }
        return jumlahData;
    }
    catch(SQLException e) {
        System.out.println(e.getMessage());
        System.out.println("SQL Error");
        return 0;
    }
}

```

Listing Program 2.1.2.10 Lanjutan *Listing Program Model Transaksi*

```

public int getService () {
    int jumlahData =0 ;
    try {
        statement = (com.mysql.jdbc.Statement) koneksi.createStatement();
        String query = "SELECT * FROM jenis_service ";
        ResultSet rs = statement.executeQuery(query);
        while (rs.next() ) {
            jumlahData ++ ;
        }
        return jumlahData;
    }
    catch(SQLException e) {
        System.out.println(e.getMessage());
        System.out.println("SQL Error");
        return 0;
    }
}
}

```

Listing Program 2.1.2.11 Lanjutan *Listing Program ModelTransaksi*

```

package Model;

import javax.swing.*;
import java.sql.*;

public class DAOLogin {
    Connection koneksi;
    Statement statement;
    ResultSet resultSet;

    public DAOLogin() {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            String url = "jdbc:mysql://localhost/service_laptop";
            koneksi = DriverManager.getConnection(url, "root", "");
            statement = koneksi.createStatement();
        } catch (ClassNotFoundException ex) {
            JOptionPane.showMessageDialog(null, "Class Not found : " + ex);
        } catch (SQLException ex) {
            JOptionPane.showMessageDialog(null, "SQL Exception : " + ex);
        }
    }

    public boolean login(ModelLogin modelLogin) {
        try {
            String query = "SELECT * FROM pegawai WHERE username = '"+modelLogin.getUsername()
                + "' " + "AND password = '" + modelLogin.getPassword() + "'";
            resultSet = statement.executeQuery(query);
            while (resultSet.next()) {
                if (modelLogin.getUsername().equals(resultSet.getString("username"))
                    && modelLogin.getPassword().equals(resultSet.getString("password"))) {
                    return true;
                }
            }
        }
        return false;
    }
}

```

```

catch (Exception sql) {
    JOptionPane.showMessageDialog(null, sql.getMessage());
    return false;
}
}
}

```

Listing Program 2.1.2.8 *Listing Program DAOLogin*

2.1.3 Package MVC

```

package MVC;

import Controller.ControllerLogin;
import Model.DAOLogin;
import Model.ModelLogin;
import View.ViewLogin;

public class MVCLLogin {
    ViewLogin viewLogin = new ViewLogin();
    ModelLogin modelLogin = new ModelLogin();
    DAOLogin daoLogin = new DAOLogin();
    ControllerLogin controllerLogin = new ControllerLogin(modelLogin, viewLogin, daoLogin);
}

```

Listing Program 2.1.3.1 *Listing Program MVCLLogin*

```

package MVC;

import Controller.ControllerTransaksi;
import Model.ModelTransaksi;
import View.View;

public class MVCTransaksi {
    View view = new View();
    ModelTransaksi modelTransaksi = new ModelTransaksi();

    ControllerTransaksi controllerTransaksi = new ControllerTransaksi(modelTransaksi, view);
}

```

Listing Program 2.1.3.2 *Listing Program MVCTransaksi*

```

package MVC;

import Controller.Controller;
import Model.Model;
import View.tampilanMain;
import View.*;

public class MVC {

    tampilanSebelumMain tampilansebelummain= new tampilanSebelumMain();
    tampilanMain tampilanmain = new tampilanMain();
}

```

```

tampilanMain2 tampilanmain2 = new tampilanMain2();
tampilanCreatePegawai tampilancreatepegawai = new tampilanCreatePegawai();
tampilanLihatPegawai tampilanlihatpegawai = new tampilanLihatPegawai();
tampilanUpdatePegawai tampilanupdatepegawai = new tampilanUpdatePegawai();
tampilanCreateService tampilancreateservice = new tampilanCreateService();
tampilanLihatService tampilanlihatservice = new tampilanLihatService();
tampilanUpdateService tampilanupdateservice = new tampilanUpdateService();
Model model = new Model() ;
Controller controller = new Controller(tampilansebelummain, model,tampilanmain,tampilanmain2,
tampilancreatepegawai,tampilanlihatpegawai, tampilanupdatepegawai,
tampilancreateservice,tampilanlihatservice, tampilanupdateservice);
}

```

Listing Program 2.1.3.3 Lanjutan *Listing* Program MVC

2.1.4 Package View

```

package View;

import javax.swing.*.*;
import java.awt.*.*;

public class FormService extends JFrame {
    Font fjudul, fkata;
    JLabel ljudul, lnama_pelanggan, lid_bagian, lusername, lpassword, lsamadengan1, lsamadengan2,
lsamadengan3;
    JTextField tfnama_pelanggan, tfid_bagian, tfusername, tfpassword;
    JButton btntambah, btnbatal;

    public FormService() {
        setSize(300, 375);
        setVisible(false);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLayout(null);
        setLocationRelativeTo(null);
        setTitle("TAMBAH KONTAK");

        fjudul = new Font("Tahoma", Font.BOLD + Font.ITALIC, 18);
        fkata = new Font("Italic", Font.ITALIC, 14);
        ljudul = new JLabel("CREATE DATA PEGAWAI");
        ljudul.setFont(fjudul);
        lsamadengan1 = new JLabel("=====");
        lsamadengan2 = new JLabel("=====");
        lsamadengan3 = new JLabel("=====");
        lnama_pelanggan = new JLabel("Nama");
        lnama_pelanggan.setFont(fkata);
        lid_bagian = new JLabel("ID Bagian");
        lid_bagian.setFont(fkata);
        lusername = new JLabel("Username");
        lusername.setFont(fkata);
        lpassword = new JLabel("Password");
        lpassword.setFont(fkata);
        tfnama_pelanggan = new JTextField();
        tfid_bagian = new JTextField();
        tfusername = new JTextField();
        tfpassword = new JTextField();
        btnbatal = new JButton("Batal");
    }
}

```

```

        btntambah = new JButton("Buat");

        add(ljudul);
        add(lsamadengan1);
        add(lsamadengan2);
        add(lsamadengan3);
        add(lnama_pelanggan);
        add(lid_bagian);
        add(lusername);
        add(lpassword);
        add(tfnama_pelanggan);
        add(tfid_bagian);
        add(tfusername);
        add(tfpassword);
        add(btnbatal);
        add(btntambah);

        lsamadengan1.setBounds(0, 25, 300, 5);
        lsamadengan2.setBounds(0, 60, 300, 5);
        ljudul.setBounds(65, 30, 200, 30);
        lnama_pelanggan.setBounds(30, 100, 80, 20);
        tfnama_pelanggan.setBounds(100, 100, 150, 20);
        lid_bagian.setBounds(30, 130, 80, 20);
        tfid_bagian.setBounds(100, 130, 150, 20);
        lusername.setBounds(30, 160, 80, 20);
        tfusername.setBounds(100, 160, 150, 20);
        lpassword.setBounds(30, 190, 80, 20);
        tfpassword.setBounds(100, 190, 150, 20);
        btntambah.setBounds(150, 230, 100, 30);
        btnbatal.setBounds(30, 230, 100, 30);
        lsamadengan3.setBounds(0, 280, 300, 5);
    }
}

```

Listing Program 2.1.4.1 *Listing Program FormService*

```

package View;

import javax.swing.JFrame;
import java.awt.*;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JTextField;

public class tampilanCreatePegawai extends JFrame {
    Font fjudul, fkata;
    JLabel ljudul, lnama, lid_bagian, lusername, lpassword, lsamadengan1, lsamadengan2, lsamadengan3;
    public JTextField tfnama, tfid_bagian, tfusername, tfpassword ;
    public JButton btntambah, btnbatal;

    public tampilanCreatePegawai () {
        setSize(300, 375);
        setVisible(false);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLayout(null);
        setLocationRelativeTo(null);
        getContentPane().setBackground(Color.PINK);
    }
}

```



```

setTitle("Create Pegawai");
fjudul = new Font("Tahoma", Font.BOLD , 18);
fkata= new Font("Calibri", Font.BOLD,16);
ljudul = new JLabel("CREATE DATA"); ljudul.setFont(fjudul);
lsamadengan1 = new JLabel("");
lsamadengan2 = new JLabel("");
lsamadengan3 = new JLabel("");
lnama = new JLabel("Nama"); lnama.setFont(fkata);
lid_bagian = new JLabel("Bagian"); lid_bagian.setFont(fkata);
lusername = new JLabel("Username"); lusername.setFont(fkata);
lpassword = new JLabel("Password"); lpassword.setFont(fkata);
tfnama = new JTextField();
tfid_bagian = new JTextField();
tfusername = new JTextField();
tfpassword = new JTextField();
btnbatal = new JButton("Cancel");
btntambah = new JButton("Submit");

add(ljudul);
add(lsamadengan1);
add(lsamadengan2);
add(lsamadengan3);
add(lnama);
add(lid_bagian);
add(lusername);
add(lpassword);
add(tfnama);
add(tfid_bagian);
add(tfusername);
add(tfpassword);
add(btnbatal);
add(btntambah);

lsamadengan1.setBounds(0,25,300,5);
lsamadengan2.setBounds(0,60,300,5);
ljudul.setBounds(80,60,200,30);
lnama.setBounds(30,100,80,20);
tfnama.setBounds(100,100,150,20);
lid_bagian.setBounds(30,130,80,20);
tfid_bagian.setBounds(100,130,150,20);
lusername.setBounds(30,160,80,20);
tfusername.setBounds(100,160,150,20);
lpassword.setBounds(30,190,80,20);
tfpassword.setBounds(100,190,150,20);
btntambah.setBounds(150,230,100,30);
btntambah.setBackground(Color.BLUE);
btntambah.setForeground(Color.WHITE);
btnbatal.setBounds(30,230,100,30);
btnbatal.setBackground(Color.RED);
btnbatal.setForeground(Color.WHITE);
lsamadengan3.setBounds(0,280,300,5);
}
}

```

Listing Program 2.1.4.2 Listing Program tampilanCreatePegawai

```

package View;

import javax.swing.JFrame;
import java.awt.*;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JTextField;

public class tampilanCreateService extends JFrame {
    Font fjudul, fkata;
    JLabel ljudul, lnama_service, lid_jenis, lharga, lsamadengan1, lsamadengan2, lsamadengan3;
    public JTextField tfnama_service, tfid_jenis, tfharga;
    public JButton btntambahs, btnbatal;

    public tampilanCreateService () {

        setSize(300,375);
        setVisible(false);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLayout(null);
        setLocationRelativeTo(null);
        getContentPane().setBackground(Color.PINK);
        setTitle("Create Service");

        fjudul = new Font("Tahoma", Font.BOLD, 18);
        fkata= new Font("Calibri", Font.BOLD, 16);
        ljudul = new JLabel("CREATE DATA JASA"); ljudul.setFont(fjudul);
        lsamadengan1 = new JLabel("");
        lsamadengan2 = new JLabel("");
        lsamadengan3 = new JLabel("");
        lnama_service = new JLabel("Jasa"); lnama_service.setFont(fkata);
        lid_jenis = new JLabel("Kode Jasa"); lid_jenis.setFont(fkata);
        lharga = new JLabel("Harga"); lharga.setFont(fkata);
        tfnama_service = new JTextField();
        tfid_jenis = new JTextField();
        tfharga = new JTextField();
        btnbatal = new JButton("Batal");
        btntambahs = new JButton("Buat");

        add(ljudul);
        add(lsamadengan1);
        add(lsamadengan2);
        add(lsamadengan3);
        add(lnama_service);
        add(lid_jenis);
        add(lharga);

        add(tfnama_service);
        add(tfid_jenis);
        add(tfharga);

        add(btnbatal);
        add(btntambahs);

        lsamadengan1.setBounds(0,25,300,5);
        lsamadengan2.setBounds(0,60,300,5);
        ljudul.setBounds(80,60,200,30);
        lnama_service.setBounds(30,100,80,20);
        tfnama_service.setBounds(100,100,150,20);
    }
}

```

```

        lid_jenis.setBounds(30,130,80,20);
        tfid_jenis.setBounds(100,130,150,20);
        lharga.setBounds(30,160,80,20);
        tfharga.setBounds(100,160,150,20);

        btntambahs.setBounds(150,230,100,30);
        btntambahs.setBackground(Color.BLUE);
        btntambahs.setForeground(Color.WHITE);
        btnbatal.setBounds(30,230,100,30);
        btnbatal.setBackground(Color.RED);
        btnbatal.setForeground(Color.WHITE);
        lsamadengan3.setBounds(0,280,300,5);
    }
}

```

Listing Program 2.1.4.3 *Listing Program tampilanCreateService*

```

package View;

import java.awt.Font;
import javax.swing.BorderFactory;
import javax.swing.GroupLayout;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.JTextField;
import javax.swing.ScrollPaneConstants;
import javax.swing.table.DefaultTableModel;

public class tampilanLihatPegawai extends JFrame{

    JPanel ppanel= new JPanel();
    JPanel ppaneldalam = new JPanel();
    public JLabel ljudul, lmenupilihan, lnama, lid_bagian, lusername,
    lpassword, lnamaisi, lid_bagianisi, lusernameisi, lpasswordisi, lid;
    public JTextField tfsearch;
    Font fjudul, fmenupilihan;
    public JTable tabel ;
    public JButton btntambah , btnkembali, btnhapus, btnupdate, btnrefresh, btnsearch;
    DefaultTableModel tabelModel;
    JScrollPane scrollPane;
    public Object namaKolom [] = { "nama", "bagian", "username", "password", "" } ;
    GroupLayout gppanel, gpaneldalam;

    public tampilanLihatPegawai () {
        fjudul = new Font("Arial Black", Font.BOLD , 21);
        fmenupilihan = new Font("Broadway", Font.BOLD , 21);
        ljudul = new JLabel("DAFTAR PEGAWAI"); ljudul.setFont(fjudul); ljudul.setForeground(white);
        lnama = new JLabel("Nama :");
        lusername = new JLabel("username :");
        lpassword = new JLabel("password :");
        lid_bagian = new JLabel("Bagian :");
        lid = new JLabel();
    }
}

```

```

lnamaisi = new JLabel();
lusernameisi = new JLabel();
lpasswordisi = new JLabel();
lid_bagianisi = new JLabel();
tabelModel = new DefaultTableModel(namaKolom,0);
tabel = new JTable(tabelModel);
scrollPane = new JScrollPane(tabel);
btnsearch = new JButton("Cari");
tfsearch = new JTextField();
btntambah = new JButton("Tambah");
btnkembali = new JButton("Kembali");
btnhapus = new JButton("Hapus");
btnupdate = new JButton("Edit");
btnrefresh = new JButton("Refresh");
ppanel.setBorder(BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED));
gppanel = new GroupLayout(ppanel);
ppanel.setLayout(gppanel);
ppaneldalam.setBorder(javax.swing.BorderFactory.createTitledBorder(null, "Data Terpilih",
javax.swing.border.TitledBorder.CENTER, javax.swing.border.
TitledBorder.DEFAULT_POSITION));
gppaneldalam = new GroupLayout(ppaneldalam);
ppaneldalam.setLayout(gppaneldalam);
lmenupilihan = new JLabel("INFO DATA"); lmenupilihan.setFont(fmenupilihan);

setTitle("Data Pegawai");
setDefaultCloseOperation(EXIT_ON_CLOSE);
setVisible(false);
setLayout(null);
setSize(900,600);
getContentPane().setBackground(BLUE);
setLocationRelativeTo(null);

add(ppanel);
ppanel.add(lmenupilihan);
ppanel.add(btntambah);
ppanel.add(btnkembali);
ppanel.add(btnhapus);
ppanel.add(btnupdate);
ppanel.add(btnrefresh);
ppanel.add(ppaneldalam);
ppaneldalam.add(lnama);
ppaneldalam.add(lusername);
ppaneldalam.add(lpassword);
ppaneldalam.add(lid_bagian);
ppaneldalam.add(lnamaisi);
ppaneldalam.add(lusernameisi);
ppaneldalam.add(lid_bagianisi);
ppaneldalam.add(lpasswordisi);
add(ljudul);
add(scrollPane);
add(btnsearch);
add(tfsearch);

ppanel.setBounds(540,50,330,480);
ppanel.setBackground(ORANGE);
ppaneldalam.setBounds(30,70,275,200);
lmenupilihan.setBounds(105,30,170,20);
btntambah.setBounds(170,400,100,30);
btnkembali.setBounds(70,400,100,30);

```

```

    btnhapus.setBounds(170,340,100,30);
    btnupdate.setBounds(70,340,100,30);
    btnrefresh.setBounds(120,280,100,20);
    ljudul.setBounds(180,20,250,20);

    scrollPane.setBounds(50,100,470,430);
    scrollPane.setVerticalScrollBarPolicy(ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS);
    tfsearch.setBounds(50,60,370,20);
    btnsearch.setBounds(420,60,100,20);

    lnama.setBounds(20,50,80,20);
    lid_bagian.setBounds(20,80,80,20);
    lusername.setBounds(20,110,80,20);
    lpassword.setBounds(20,140,80,20);
    lnamaisi.setBounds(95,50,200,20);
    lid_bagianisi.setBounds(95,80,200,20);
    lusernameisi.setBounds(95,110,200,20);
    lpasswordisi.setBounds(95,140,200,20);
    btnhapus.setEnabled(false);
    btnupdate.setEnabled(false);
}
}

```

Listing Program 2.1.4.4 *Listing Program tampilanLihatPegawai*

```

package View;

import java.awt.Font;
import javax.swing.BorderFactory;
import javax.swing.GroupLayout;
import javax.swing.JButton;
import javax.swing.JFrame;
import static java.awt.Color.*;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.JTextField;
import javax.swing.ScrollPaneConstants;
import javax.swing.table.DefaultTableModel;

public class tampilanLihatService extends JFrame{

    JPanel ppanel= new JPanel();
    JPanel ppaneldalam = new JPanel();
    public JLabel ljudul, lmenupilihan,lnama_service,lid_jenis,lharga,
    lnama_serviceisi,lid_jenisisi,lhargaisi,lpasswordisi,lid;
    public JTextField tfsearchs;
    Font fjudul,fmenupilihan;
    public JTable tabel ;
    public JButton btntambahs , btnkembalis, btnhapuss, btnupdates,btnrefreshs,btnsearchs;
    DefaultTableModel tabelModel;
    JScrollPane scrollPane;
    public Object namaKoloms [] = { "nama service","id jenis","harga","" } ;
    GroupLayout gppanel,gpaneldalam;

    public tampilanLihatService() {

```

```

fjudul = new Font("Arial Black", Font.BOLD , 21);
fmenupilihan = new Font("Broadway", Font.BOLD , 21);
ljudul = new JLabel("DAFTAR JASA"); ljudul.setFont(fjudul); ljudul.
setFont(fjudul);ljudul.setForeground(white);
lnama_service = new JLabel("Nama  :");
lharga = new JLabel("Harga  :");
lid_jenis = new JLabel("Id Jenis  :");
lid = new JLabel();
lnama_serviceisi = new JLabel();
lhargaisi = new JLabel();
lpasswordisi = new JLabel();
lid_jenisisi = new JLabel();
tabelModel = new DefaultTableModel(namaKoloms,0);
tabel = new JTable(tabelModel);
scrollPane = new JScrollPane(tabel);
btnsearchs = new JButton("Cari");
tfsearchs = new JTextField();
btntambahs = new JButton("Tambah");
btnkembalis = new JButton("Kembali");
btnhapuss = new JButton("Hapus");
btnupdates = new JButton("Edit");
btnrefreshs = new JButton("Refresh");
ppanel.setBorder(BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED));
gppanel = new GroupLayout(ppanel);
ppanel.setLayout(gppanel);
ppaneldalam.setBorder(javax.swing.BorderFactory.createTitledBorder(null, "Data Terpilih", javax.
swing.border.TitledBorder.CENTER, javax.swing.border.TitledBorder.DEFAULT_POSITION));
gppaneldalam = new GroupLayout(ppaneldalam);
ppaneldalam.setLayout(gppaneldalam);
lmenupilihan = new JLabel("INFO DATA"); lmenupilihan.setFont(fmenupilihan);

setTitle("Data Jasa");
setDefaultCloseOperation(EXIT_ON_CLOSE);
setVisible(false);
setLayout(null);
setSize(900,600);
getContentPane().setBackground(BLUE);
setLocationRelativeTo(null);

add(ppanel);
ppanel.add(lmenupilihan);
ppanel.add(btntambahs);
ppanel.add(btnkembalis);
ppanel.add(btnhapuss);
ppanel.add(btnupdates);
ppanel.add(btnrefreshs);
ppanel.add(ppaneldalam);
ppaneldalam.add(lnama_service);
ppaneldalam.add(lharga);
ppaneldalam.add(lid_jenis);
ppaneldalam.add(lnama_serviceisi);
ppaneldalam.add(lhargaisi);
ppaneldalam.add(lid_jenisisi);
ppaneldalam.add(lpasswordisi);
add(ljudul);
add(scrollPane);
add(btnsearchs);
add(tfsearchs);

```

```

ppanel.setBounds(540,50,330,480);
ppanel.setBackground(ORANGE);
ppaneldalam.setBounds(30,70,275,200);
lmenupilihan.setBounds(105,30,170,20);
btntambahs.setBounds(170,400,100,30);
btnkembalis.setBounds(70,400,100,30);
btnhapuss.setBounds(170,340,100,30);
btnupdates.setBounds(70,340,100,30);
btnrefreshs.setBounds(120,280,100,20);
ljudul.setBounds(220,20,250,20);

scrollPane.setBounds(50,100,470,430);
scrollPane.setVerticalScrollBarPolicy(ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS);
tfsearchs.setBounds(50,60,370,20);
btnsearchs.setBounds(420,60,100,20);

lnama_service.setBounds(20,50,80,20);
lid_jenis.setBounds(20,80,80,20);
lharga.setBounds(20,110,80,20);
lnama_serviceisi.setBounds(95,50,200,20);
lid_jenisisi.setBounds(95,80,200,20);
lhargaisi.setBounds(95,110,200,20);
lpasswordisi.setBounds(95,140,200,20);
btnhapuss.setEnabled(false);
btnupdates.setEnabled(false);
}
}

```

Listing Program 2.1.4.5 *Listing Program tampilanLihatService*

```

package View;

import javax.swing.*.*;
import java.awt.*.*;
import java.awt.event.*;

public class tampilanMain extends JFrame{
    Font fjudul;
    JLabel ljudul;
    public JButton btnbuatKontak,btnlihatKontak,btnexit;

    public tampilanMain () {

        setLayout(null);
        setVisible(false);
        setSize(300,350);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLocationRelativeTo(null);
        getContentPane().setBackground(Color.PINK);
        setBackground(new java.awt.Color(204,204,204));
        setTitle("Pegawai");

        fjudul = new Font("Calibri", Font.BOLD , 21);
        ljudul = new JLabel("Silahkan Pilih!");
        ljudul.setFont(fjudul);

        btnbuatKontak = new JButton("Create Data Pegawai");
    }
}

```

```

        btnlihatKontak = new JButton("Read Data Pegawai");
        btnexit = new JButton("Exit");

        add(ljudul);
        add(btnbuatKontak);
        add(btnexit);
        add(btnlihatKontak);

        ljudul.setBounds(85,70,160,20);

        btnbuatKontak.setBounds(50,100,190,30);
        btnbuatKontak.setBackground(Color.DARK_GRAY);
        btnbuatKontak.setForeground(Color.white);
        btnlihatKontak.setBounds(50,150,190,30);
        btnlihatKontak.setBackground(Color.BLUE);
        btnlihatKontak.setForeground(Color.white);
        btnexit.setBounds(50,200,190,30);
        btnexit.setBackground(Color.RED);
        btnexit.setForeground(Color.white);
    }
}

```

Listing Program 2.1.4.6 Lanjutan *Listing Program tampilanMain*

```

package View;

import javax.swing.*.*;
import java.awt.*.*;
import java.awt.event.*;

public class tampilanMain2 extends JFrame{
    Font fjudul;
    JLabel ljudul;
    public JButton btnbuatKontaks,btnlihatKontaks,btnexit;

    public tampilanMain2 () {

        setLayout(null);
        setVisible(false);
        setSize(300,350);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLocationRelativeTo(null);
        getContentPane().setBackground(Color.PINK);
        setBackground(new java.awt.Color(204,204,204));
        setTitle("Jasa");

        fjudul = new Font("Calibri", Font.BOLD , 21);
        ljudul = new JLabel("Silahkan Pilih!");
        ljudul.setFont(fjudul);

        btnbuatKontaks = new JButton("Create Data Service");
        btnlihatKontaks = new JButton("Read Data Service");
        btnexit = new JButton("Exit");

        add(ljudul);

        add(btnbuatKontaks);
    }
}

```



```

add(btnexit);
add(btnlihatKontaks);

ljudul.setBounds(87,70,160,20);

btnbuatKontaks.setBounds(50,100,190,30);
btnbuatKontaks.setBackground(Color.darkGray);
btnbuatKontaks.setForeground(Color.white);
btnlihatKontaks.setBounds(50,150,190,30);
btnlihatKontaks.setForeground(Color.white);
btnlihatKontaks.setBackground(Color.blue);
btnexit.setBounds(50,200,190,30);
btnexit.setBackground(Color.RED);
btnexit.setForeground(Color.WHITE);
    }
}

```

Listing Program 2.1.4.7 Lanjutan *Listing Program tampilanMain2*

```

package View;

import javax.swing.*.*;
import java.awt.*.*;
import java.awt.event.*;

public class tampilanSebelumMain extends JFrame{
    Font fjudul;
    JLabel ljudul;
    public JButton btnPegawai,btnService,btnTransaksi,btnexit;

    public tampilanSebelumMain () {
        setLayout(null);
        setVisible(true);
        setSize(300,350);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLocationRelativeTo(null);
        setBackground(new java.awt.Color(204,204,204));
        getContentPane().setBackground(Color.PINK);
        setTitle("menu utama");

        fjudul = new Font("Tahoma", Font.BOLD , 21);
        ljudul = new JLabel("PILIH");
        ljudul.setFont(fjudul);

        btnPegawai = new JButton("Pegawai");
        btnService = new JButton("Service");
        btnTransaksi = new JButton("Transaksi");
        btnexit = new JButton("Exit");

        add(ljudul);
        add(btnPegawai);
        add(btnService);
        add(btnTransaksi);

        ljudul.setBounds(110,70,160,20);

        btnPegawai.setBounds(50,100,190,30);
    }
}

```

```

        btnService.setBounds(50,150,190,30);
        btnTransaksi.setBounds(50,200,190,30);
        /*btnexit.setBounds(50,200,190,30);*/
    }
}

```

Listing Program 2.1.4.8 Lanjutan *Listing* Program tampilanSebelumMain

```

package View;

import javax.swing.*.*;
import java.awt.*.*;

public class tampilanUpdatePegawai extends JFrame {
    Font fjudul ;
    JLabel ljudul, lnama, lid_bagian, lusername, lpassword, lsamadengan1, lsamadengan2;
    public JTextField tfnama, tfid_bagian , tfusername , tfpassword;
    public JButton btncancel, btnkembali ;

    public tampilanUpdatePegawai () {
        fjudul = new Font("Tahoma", Font.BOLD , 21);
        ljudul = new JLabel("EDIT PEGAWAI"); ljudul.setFont(fjudul);
        lsamadengan1 = new JLabel("");
        lsamadengan2 = new JLabel("");
        lnama = new JLabel("Nama");
        lid_bagian = new JLabel("Bagian");
        lusername = new JLabel("Username");
        lpassword = new JLabel("Password");
        tfnama = new JTextField();
        tfid_bagian = new JTextField();
        tfusername = new JTextField();
        tfpassword = new JTextField();
        btncancel = new JButton("Submit");
        btnkembali = new JButton("Cancel");

        setTitle("Data Pegawai");
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setVisible(false);
        setLayout(null);
        setSize(360,360);
        getContentPane().setBackground(Color.PINK);
        setLocationRelativeTo(null);

        add(lsamadengan1);
        lsamadengan1.setBounds(0,40,360,5);
        add(lsamadengan2);
        lsamadengan2.setBounds(0,80,360,5);
        add(ljudul);
        ljudul.setBounds(100,50,200,20);
        add(lnama);
        lnama.setBounds(20,110,60,20);
        add(tfnama);
        tfnama.setBounds(100,110,200,20);
        add(lid_bagian);
        lid_bagian.setBounds(20,140,60,20);
        add(tfid_bagian);
        tfid_bagian.setBounds(100,140,200,20);
    }
}

```

```

add(lusername);
lusername.setBounds(20,170,60,20);
add(tfusername);
tfusername.setBounds(100,170,200,20);
add(lpassword);
lpassword.setBounds(20,200,60,20);
add(tfpassword);
tfpassword.setBounds(100,200,200,20);

add(btnkembali);
btnkembali.setBounds(80,250,100,30);
btnkembali.setForeground(Color.WHITE);
btnkembali.setBackground(Color.RED);
add(btnedit);
btnedit.setBounds(180,250,100,30);
btnedit.setForeground(Color.white);
btnedit.setBackground(Color.BLUE);
}
}

```

Listing Program 2.1.4.9 *Listing Program tampilanUpdatePegawai*

```

package View;

import javax.swing.*;
import java.awt.*;

public class tampilanUpdateService extends JFrame {
    Font fjudul ;
    JLabel ljudul, lnama_service, lid_jenis, lharga, lsamadengan1,lsamadengan2;
    public JTextField tfnama_service, tfid_jenis , tfharga ;
    public JButton btnedits, btnkembalis ;

    public tampilanUpdateService() {
        fjudul = new Font("Tahoma", Font.BOLD + Font.ITALIC, 21);
        ljudul = new JLabel("EDIT JASA"); ljudul.setFont(fjudul);
        lsamadengan1 = new JLabel("");
        lsamadengan2 = new JLabel("");
        lnama_service = new JLabel("Jasa");
        lid_jenis = new JLabel("ID Jasa");
        lharga = new JLabel("Harga");
        tfnama_service = new JTextField();
        tfid_jenis = new JTextField();
        tfharga = new JTextField();
        btnedits = new JButton("Submit");
        btnkembalis = new JButton("Cancel");

        setTitle("Data Jasa");
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setVisible(false);
        setLayout(null);
        setSize(360,360);
        getContentPane().setBackground(Color.PINK);
        setLocationRelativeTo(null);

        add(lsamadengan1);
        lsamadengan1.setBounds(0,40,360,5);
        add(lsamadengan2);
    }
}

```

```

Isamadengan2.setBounds(0,80,360,5);
add(ljudul);
ljudul.setBounds(100,50,200,20);
add(lnama_service);
lnama_service.setBounds(20,110,60,20);
add(tfnama_service);
tfnama_service.setBounds(100,110,200,20);
add(lid_jenis);
lid_jenis.setBounds(20,140,60,20);
add(tfid_jenis);
tfid_jenis.setBounds(100,140,200,20);
add(lharga);
lharga.setBounds(20,170,60,20);
add(tfharga);
tfharga.setBounds(100,170,200,20);

add(btnkembalis);
btnkembalis.setBounds(80,250,100,30);
btnkembalis.setForeground(Color.WHITE);
btnkembalis.setBackground(Color.RED);
add(btnedits);
btnedits.setBounds(180,250,100,30);
btnedits.setForeground(Color.white);
btnedits.setBackground(Color.BLUE);
}
}

```

Listing Program 2.1.4.10 Lanjutan *Listing* Program tampilanUdateService

```

package View;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.JTextField;
import javax.swing.ScrollPaneConstants;
import javax.swing.table.DefaultTableModel;
import java.awt.*;

public class View extends JFrame {
    JLabel lNama_pelanggan, lStatus, lNo_hp, lTotal_biaya, lno_transaksi, lUsername,
    lid_jenis, lNama_service;
    public JTextField tfNama_pelanggan, tfNo_hp, tfTotal_biaya, tfno_transaksi, tfSearch, tfUsername,
    tfId_jenis, tfNama_service;
    //String nama_service[] = {"1. LCD = 800.000", "2. Keyboard = 500.000", "3. Software = 1000.000"};
    //JComboBox cbNama_service;
    public JButton bRefresh, bCreate, bUpdate, bDelete, bExit, bSearch;
    public JTable tabel, tabel1, tabel2;
    DefaultTableModel tableModel, tableModel1, tableModel2;
    JScrollPane scrollPane, scrollPane1, scrollPane2;
    public Object namaKolom[] = {"No. Tran", "username", "id_jenis", "Pelanggan", "No. Hp", "Biaya"};
    public Object namaKolom1[] = {"Id_Jasa", "Jasa", "Harga"};
    public Object namaKolom2[] = {"Pegawai", "Username"};
}

```

```

public View() {
    setTitle("Data Transaksi");
    tableModel = new DefaultTableModel(namaKolom, 0);
    tabel = new JTable(tableModel);
    scrollPane = new JScrollPane(tabel);

    tableModel1 = new DefaultTableModel(namaKolom1, 0);
    tabel1 = new JTable(tableModel1);
    scrollPane1 = new JScrollPane(tabel1);

    tableModel2 = new DefaultTableModel(namaKolom2, 0);
    tabel2 = new JTable(tableModel2);
    scrollPane2 = new JScrollPane(tabel2);

    lUsername = new JLabel("Pegawai Yang Melayani");
    lId_jenis = new JLabel("Id Jenis");
    lNama_pelanggan = new JLabel("Nama Pelanggan");
    lStatus = new JLabel("Jasa Service");
    lNo_hp = new JLabel("NO. Hp");
    lTotal_biaya = new JLabel("Total Biaya");
    lno_transaksi = new JLabel("no_transaksi");

    tfUsername = new JTextField();
    tfId_jenis = new JTextField();
    tfNama_pelanggan = new JTextField();
    tfNo_hp = new JTextField();
    tfTotal_biaya = new JTextField();
    tfno_transaksi = new JTextField();
    tfSearch = new JTextField();
    tfNama_service = new JTextField();

    bRefresh = new JButton("Refresh");
    bCreate = new JButton("Create");
    bUpdate = new JButton("Update");
    bDelete = new JButton("Delete");
    bExit = new JButton("Exit");
    bSearch = new JButton("Search");

    setLayout(null);
    add(scrollPane);
    add(scrollPane1);
    add(scrollPane2);
    add(lUsername);
    add(tfUsername);
    add(lId_jenis);
    add(tfId_jenis);
    add(lNama_pelanggan);
    add(tfNama_pelanggan);
    //add(lNama_service);
    //add(tfNama_service);
    add(lNo_hp);
    add(tfNo_hp);
    add(lTotal_biaya);
    add(tfTotal_biaya);
    add(lno_transaksi);
    add(tfno_transaksi);
    add(tfSearch);

    add(bSearch);

```

```

add(bRefresh);
add(bCreate);
add(bUpdate);
add(bDelete);
add(bExit);

scrollPane.setBounds(40, 60, 1000, 340);
scrollPane.setVerticalScrollBarPolicy(ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS);
scrollPane.setHorizontalScrollBarPolicy
(scrollPaneConstants.HORIZONTAL_SCROLLBAR_ALWAYS);

scrollPane1.setBounds(380, 460, 280, 100);
scrollPane1.setVerticalScrollBarPolicy(ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS);
scrollPane1.setHorizontalScrollBarPolicy
(scrollPaneConstants.HORIZONTAL_SCROLLBAR_ALWAYS);

scrollPane2.setBounds(680, 460, 280, 100);
scrollPane2.setVerticalScrollBarPolicy(ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS);
scrollPane2.setHorizontalScrollBarPolicy
(scrollPaneConstants.HORIZONTAL_SCROLLBAR_ALWAYS);

lUsername.setBounds(40, 410, 150, 30);
tfUsername.setBounds(40, 440, 150, 25);
lId_jenis.setBounds(40, 475, 100, 30);
tfId_jenis.setBounds(40, 505, 150, 25);
// lNama_service.setBounds(240, 475, 100, 30);
tfNama_service.setBounds(240, 505, 150, 25);
lNama_pelanggan.setBounds(40, 540, 100, 30);
tfNama_pelanggan.setBounds(40, 570, 150, 25);
lNo_hp.setBounds(40, 605, 100, 30);
tfNo_hp.setBounds(40, 635, 150, 25);
lTotal_biaya.setBounds(40, 665, 100, 30);
tfTotal_biaya.setBounds(40, 695, 150, 25);

tfSearch.setBounds(40, 20, 260, 25);
bSearch.setBounds(320, 18, 100, 30);

bRefresh.setBounds(260, 725, 100, 30);
bCreate.setBounds(380, 725, 100, 30);
bDelete.setBounds(500, 725, 100, 30);
bExit.setBounds(620, 725, 100, 30);

setSize(1100, 800);
setVisible(true);
setLocationRelativeTo(null);
getContentPane().setBackground(Color.PINK);
setDefaultCloseOperation(EXIT_ON_CLOSE);
}

public int getno_transaksi() {
    return Integer.parseInt(tfno_transaksi.getText());
}

public String getNama_pelanggan() {
    return tfNama_pelanggan.getText();
}

public String getUsername() {

```

```

        return tfUsername.getText();
    }

    public String getId_jenis() {
        return tfId_jenis.getText();
    }

    public String getNo_hp() {
        return tfNo_hp.getText();
    }

    public int getTotal_biaya() {
        return Integer.parseInt(tfTotal_biaya.getText());
    }

    public String getNama_service() {
        return tfNama_service.getText();
    }

    public String getSearch() {
        return tfSearch.getText();
    }
}

```

Listing Program 2.1.4.11 *Listing Program View*

```

package View;

import javax.swing.*.*;
import java.awt.*.*;

public class ViewLogin extends JFrame{
    JLabel lJudul, lUsername, lPassword;
    public JTextField tfUser;
    JPasswordField tfPassword;
    public JButton bLogin;

    public ViewLogin() {
        setTitle("LOGIN");
        lJudul = new JLabel("LOGIN");
        lUsername = new JLabel("Username : ");
        tfUser = new JTextField();
        lPassword = new JLabel("Password : ");
        tfPassword = new JPasswordField();

        bLogin = new JButton("Login");

        setLayout(null);
        add(lJudul);
        add(lUsername);
        add(lPassword);
        add(tfUser);
        add(tfPassword);
        add(bLogin);

        lJudul.setBounds(160, 30, 100, 20);
        lUsername.setBounds(70, 70, 80, 25);
    }
}

```

```

tfUser.setBounds(160, 70, 145, 25);
lPassword.setBounds(70, 110, 80, 25);
tfPassword.setBounds(160, 110, 145, 25);
bLogin.setBounds(160, 150, 80, 25);

setSize(400, 300);
setVisible(true);
setLocationRelativeTo(null);
getContentPane().setBackground(Color.PINK);
setDefaultCloseOperation(EXIT_ON_CLOSE);
}
public String getUser() {
    return tfUser.getText();
}
public String getPassword() {
    return tfPassword.getText();
}
}

```

Listing Program 2.1.4.12 Lanjutan *Listing* Program ViewLogin

2.1.3 Package Main

```

import MVC.MVCLLogin;

public class Main {
    public static void main(String[] args) {
        MVCLLogin login = new MVCLLogin();
    }
}

```

Listing Program 2.1.4.13 Lanjutan *Listing* Program Main

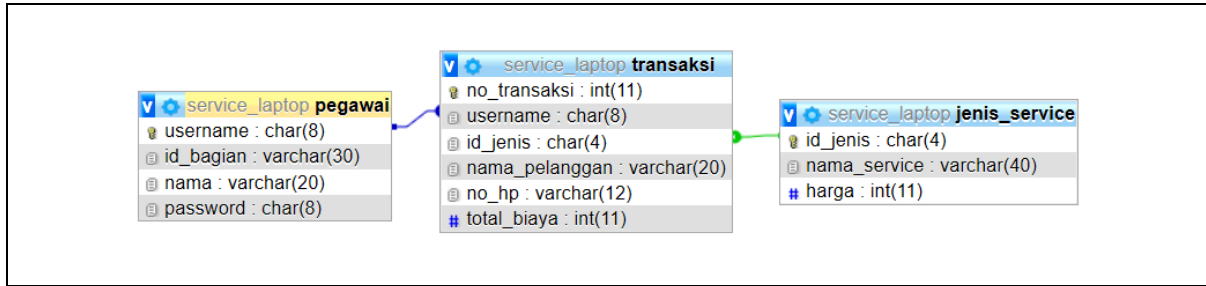
2.2 Pembahasan

Projek Pemrograman Berbasis Objek dengan menggunakan bahasa pemrograman java ini membuat sebuah aplikasi tentang Pendataan service laptop dimana aplikasi ini memiliki fitur-fitur utama yang dibutuhkan dalam perpustakaan pada umumnya yaitu adanya pelayanan jasa service, pengolahan data pelanggan, pengolahan data pegawai. Aplikasi ini akan mempermudah pendataan pelayanan service laptop, dengan mengolah data-datanya didalam aplikasi ini.

2.2.1 Database

Aplikasi Pendataan service laptop ini memiliki beberapa entitas utama yaitu pegawai sebagai orang yang melakukan pengolahan data serta aktivitas yang berkaitan dengan pendataan service laptop.

Rancangan databasenya dapat dilihat sebagai berikut



Gambar 2.2.1 Rancangan Database

Lalu untuk dapat menghubungkan database ke dalam aplikasi menggunakan JDBC driver, dengan menambahkan *library* baru dari maven dan lakukan konfigurasi seperti berikut

```

Connection koneksi ;
Statement statement ;
static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
static final String DB_URL = "jdbc:mysql://localhost/service_laptop";//nama database kita di slash terakhir
static final String USER = "root";
static final String PASS = "";

public Model() {
    try {
        Class.forName(JDBC_DRIVER);
        koneksi = (Connection) DriverManager.getConnection(DB_URL, USER, PASS);
    }
    catch (Exception e) {
        JOptionPane.showMessageDialog( parentComponent: null, e.getMessage());
        System.out.println("Koneksi Gagal");
    }
}

```

Gambar 2.2.2 Koneksi Database

2.2.2 Login

Sebelum mengakses fitur-fitur yang ada dalam aplikasi admin diminta untuk *Login* terlebih dahulu. Ketika admin melakukan pengolahan data, datanya akan disimpan dalam *database*, dan Ketika pegawai melakukan *login* maka data pengguna akan diperiksa di dalam *database* dengan cara seperti berikut.

```

public boolean login(ModelLogin modelLogin) {
    try {
        String query = "SELECT * FROM pegawai WHERE username = '" + modelLogin.getUsername() + "' AND password = '" + modelLogin.getPassword() + "'";
        resultSet = statement.executeQuery(query);
        while (resultSet.next()) {
            if (modelLogin.getUsername().equals(resultSet.getString(1, "username")) && modelLogin.getPassword().equals(resultSet.getString(2, "password"))) {
                return true;
            }
        }
        return false;
    } catch (Exception sql) {
        JOptionPane.showMessageDialog( parentComponent: null, sql.getMessage());
        return false;
    }
}

```

Gambar 2.2.3 Proses *Sign In*

Apabila *method* tersebut mengembalikan nilai 1 maka proses *sign in* berhasil dan langsung menuju tampilan utama tapi jika nilainya 0 maka proses *sign in* gagal dan akan muncul peringatan.

2.2.3 Menu

Setelah melakukan *login* pegawai akan masuk ke dalam menu-menu yaitu pegawai, *service* dan transaksi. Menu ini akan menjadi penghubung pegawai untuk berpindah dari fitur satu ke fitur lainnya. Untuk dapat melakukan itu semua jendela di bungkus dalam masing-masing panel dan digabungkan dalam panel utama untuk menampung panel-panel tersebut. Untuk bisa berpindah antar panel dapat dilakukan *code* seperti berikut.

```

tampilansebelummain.btnPegawai.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilanmain.setVisible(true);
        tampilansebelummain.setVisible(false);
    }
});

tampilansebelummain.btnService.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        tampilanmain2.setVisible(true);
        tampilansebelummain.setVisible(false);
    }
});

tampilansebelummain.btnTransaksi.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        MVCTransaksi mvcTransaksi = new MVCTransaksi();
    }
});

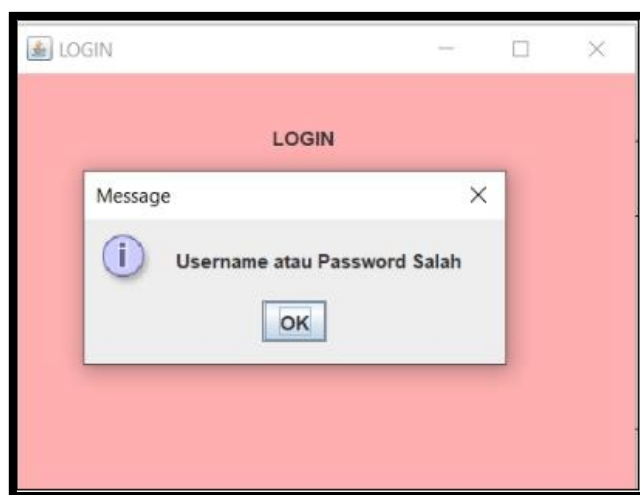
```

Gambar 2.2.4 Menu

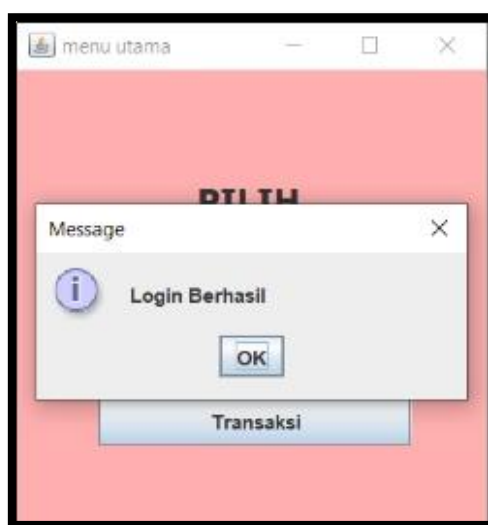
2.3 Output



Gambar 2.3.1 Gambar Login



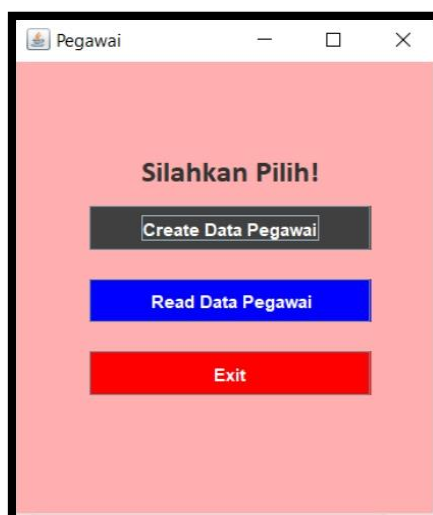
Gambar 2.3.2 Gambar login gagal



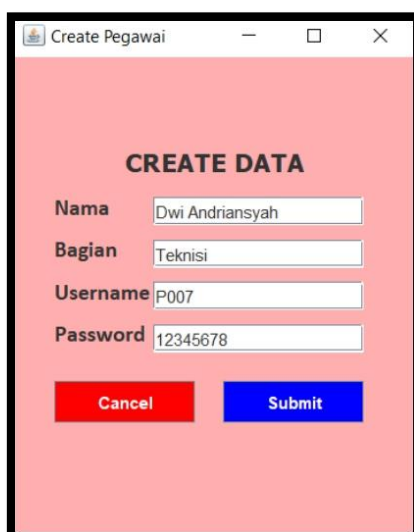
Gambar 2.3.3 Gambar login berhasil



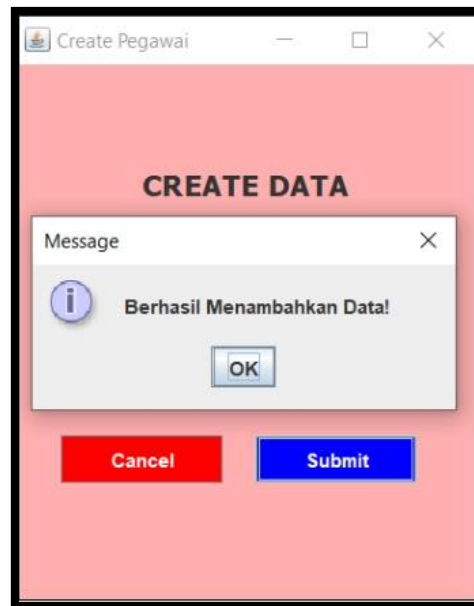
Gambar 2.3.4 Gambar menu utama



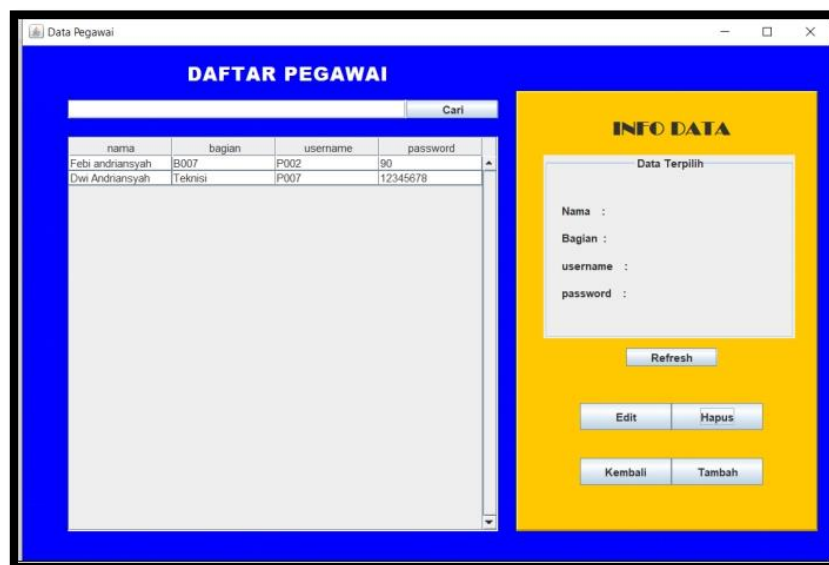
Gambar 2.3.5 Menu Pegawai



Gambar 2.3.6 Gambar form pegawai



Gambar 2.3.7 Gambar berhasil menambahkan pegawai



Gambar 2.3.8 Gambar read data pegawai



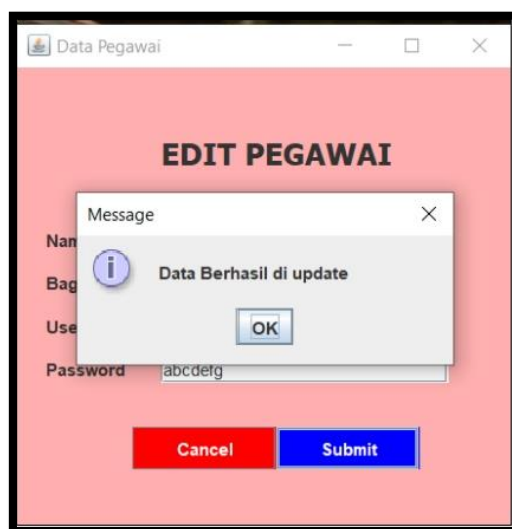
Gambar 2.3.9 Gambar cari pegawai



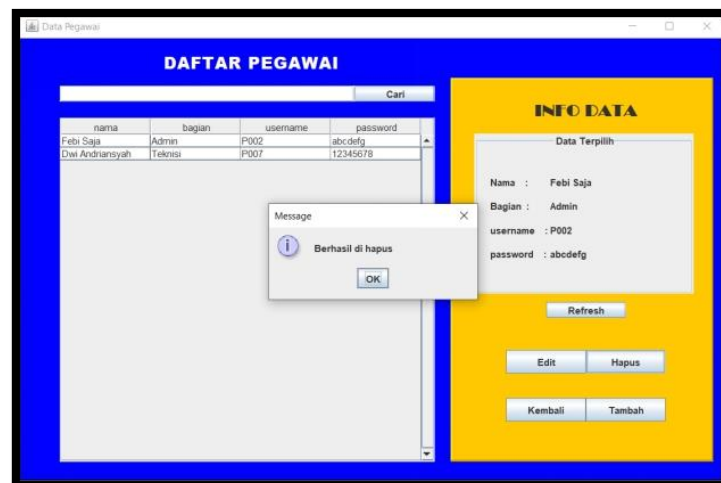
Gambar 2.3.10 Gambar Data Terpilih



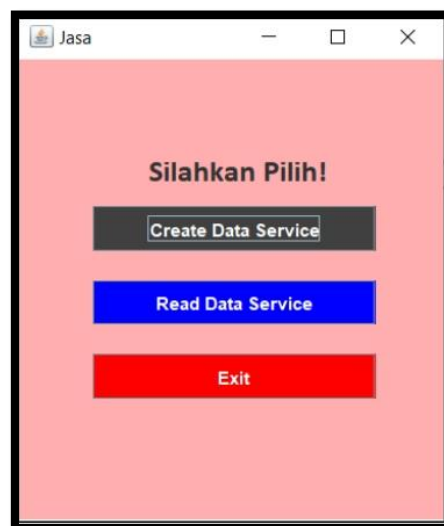
Gambar 2.3.11 Gambar form edit



Gambar 2.3.12 Gambar data berhasil di edit



Gambar 2.3.13 Gambar hapus data terpilih



Gambar 2.3.14 Gambar menu jasa

The screenshot shows a form window titled "Create Service". The background is light pink. At the top, there is a heading "CREATE DATA JASA". Below the heading are three input fields:

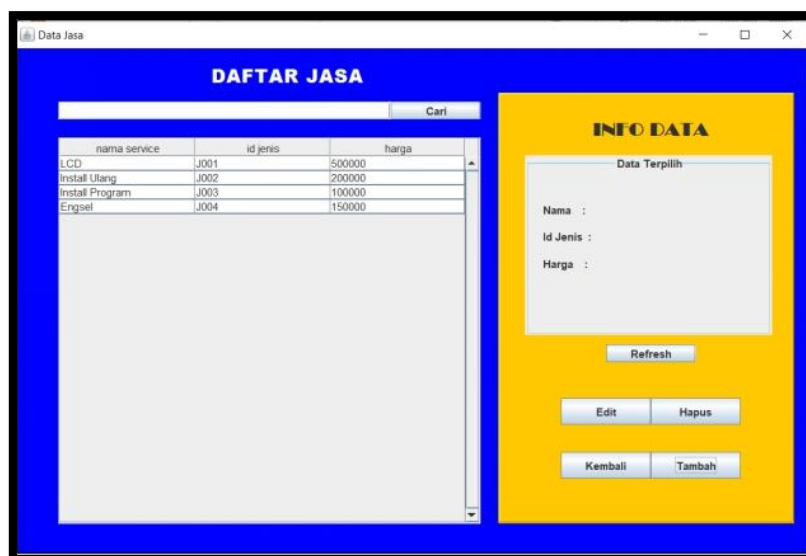
- A field labeled "Jasa" with the text "LCD" entered.
- A field labeled "Kode Jasa" with the text "J001" entered.
- A field labeled "Harga" with the text "500000" entered.

At the bottom of the form are two buttons: a red button labeled "Batal" and a blue button labeled "Buat".

Gambar 2.3.15 Gambar form jasa



Gambar 2.3.14 Gambar berhasil menambahkan service



Gambar 2.3.15 Gambar read data service



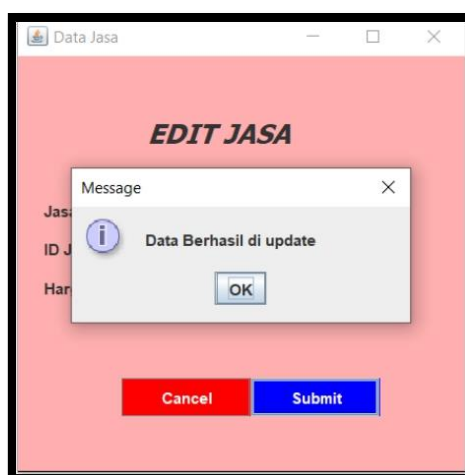
Gambar 2.3.16 Gambar cari pegawai



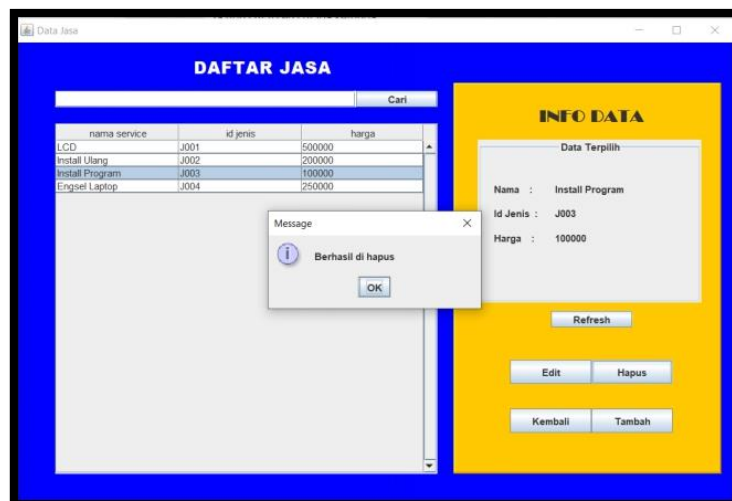
Gambar 2.3.17 Gambar Data Terpilih



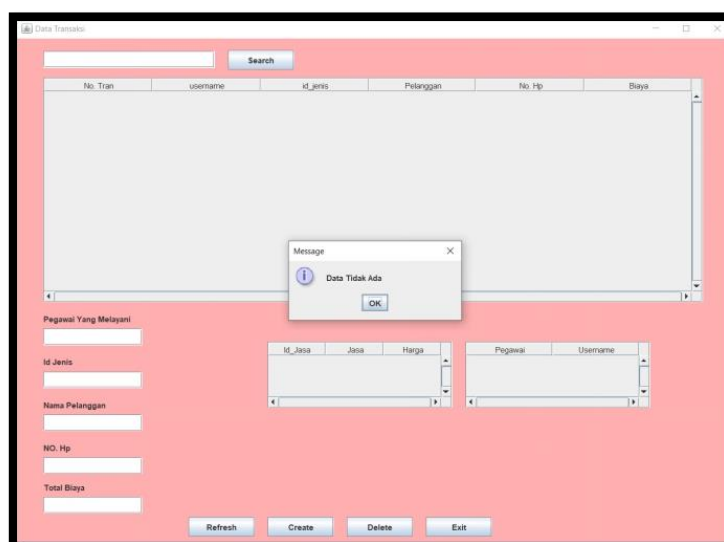
Gambar 2.3.18 Gambar form edit



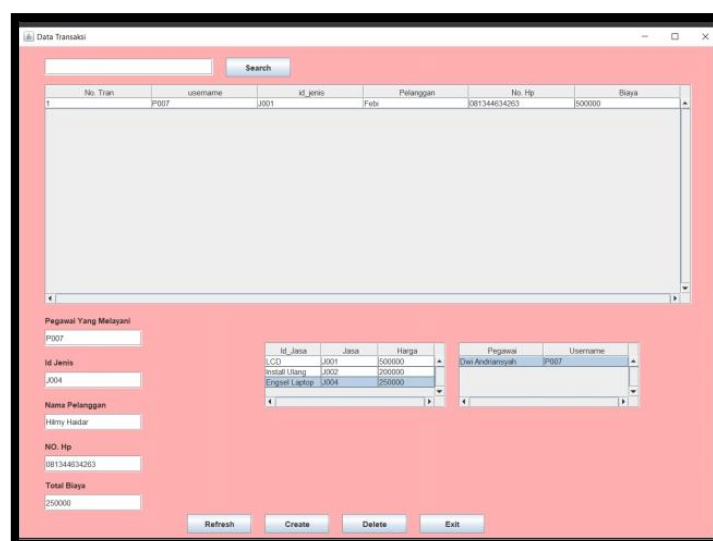
Gambar 2.3.19 Gambar data berhasil di edit



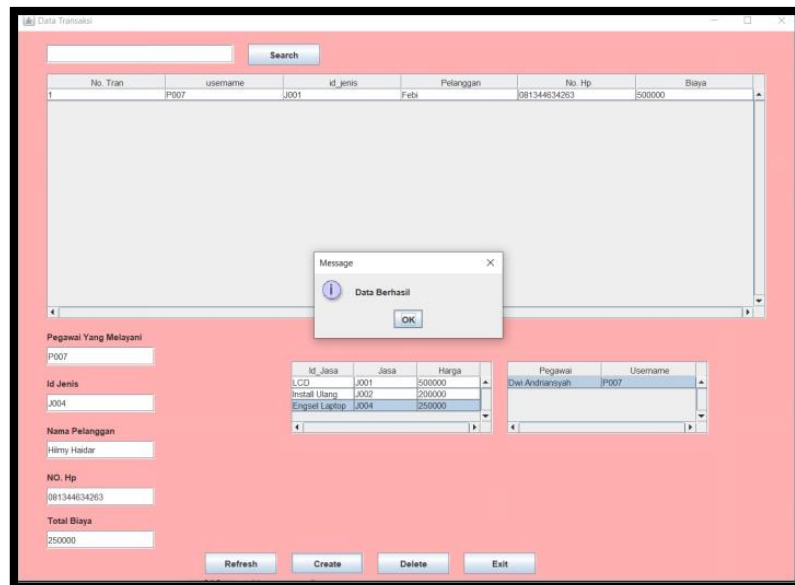
Gambar 2.3.20 Gambar hapus data terpilih



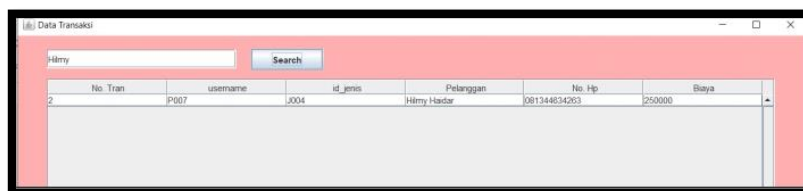
Gambar 2.3.21 Gambar transaksi



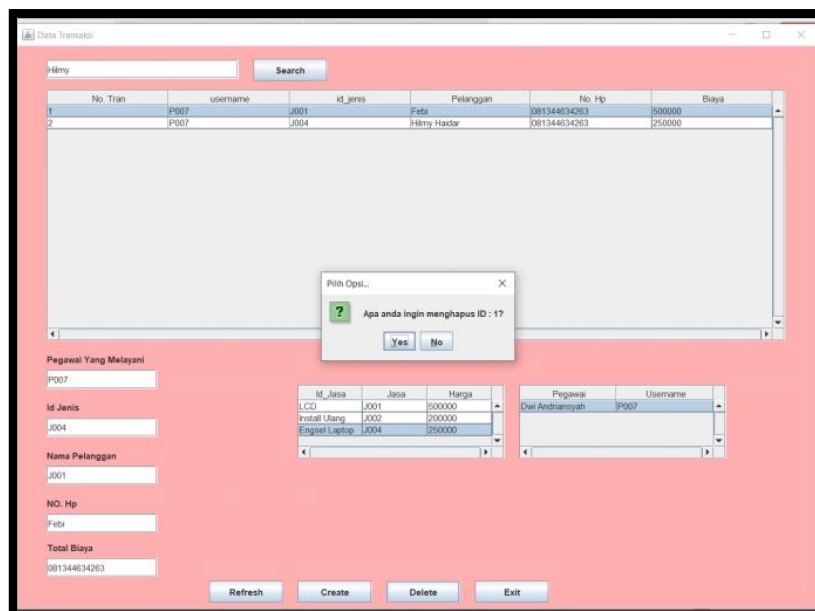
Gambar 2.3.22 Gambar create transaksi



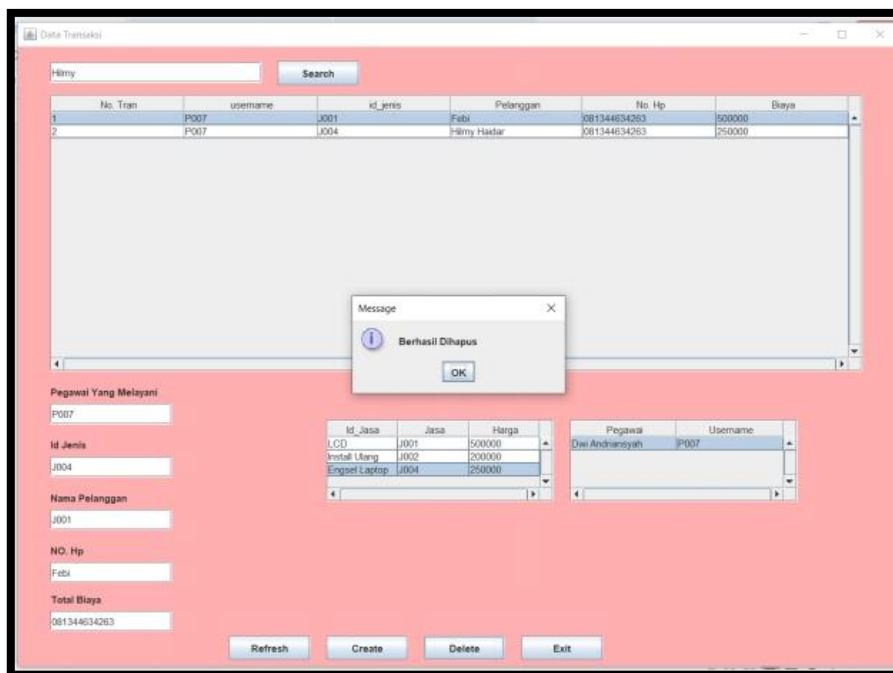
Gambar 2.3.23 Gambar berhasil create transaksi



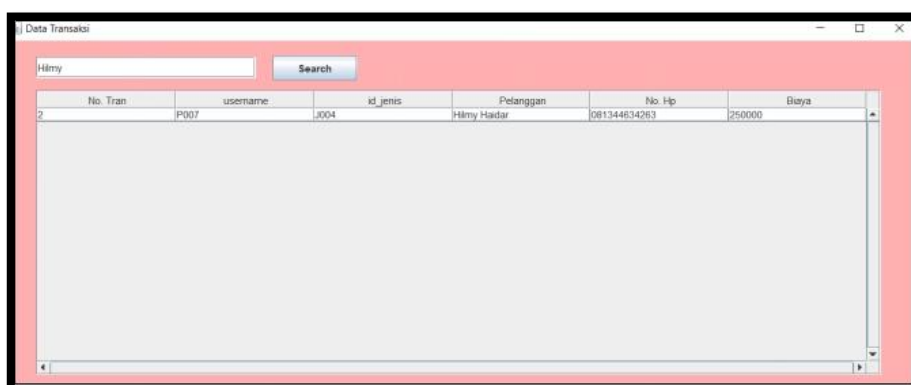
Gambar 2.3.24 Gambar cari transaksi



Gambar 2.3.25 Gambar hapus transaksi



Gambar 2.3.26 Gambar data berhasil dihapus



Gambar 2.3.27 Gambar data setelah dihapus

BAB III

JADWAL Pengerjaan dan Pembagian Tugas

3.1 Jadwal Pengerjaan dan Pembagian

Jadwal pengerjaan proyek Pemrograman Berbasis Objek tentang Pendataan Service Laptop ini sebagai berikut:

Tabel 3.1 Tabel Jadwal Pengerjaan dan Pembagian Tugas

No	Nama Tugas	Minggu ke-					Penyusun
		April			Mei		
		2	3	4	1	2	
1.	Menentukan Judul						Hilmy
2.	Merancang Database						Hilmy & Febi
3.	Merancang Desain Aplikasi						Hilmy & Febi
4.	Pembuatan View						Hilmy & Febi
5.	Pembuatan Model dan Controller						Hilmy & Febi
6.	Testing dan Penyempurnaan Program						Febi
7.	Menyusun Laporan						Hilmy & Febi

BAB IV

KESIMPULAN DAN SARAN

4.1. Kesimpulan

Pada laporan ini, dapat disimpulkan bahwa konsep OOP dan MVC sangat membantu untuk pembuatan suatu aplikasi atau program yang menggunakan bahasa pemrograman java. Dengan memanfaatkan konsep OOP dan MVC membuat sebuah alur program yang terstruktur dan sistematis. Pada proyek ini, hal tersebut berguna dalam perancangan “Aplikasi Pendataan Service Laptop”.

4.2. Saran

Pada pembuatan aplikasi ini, masih diperlukan perbaikan. Perbaikan yang bisa dilakukan diantaranya dengan menambahkan fitur detail sehingga pelanggan dapat menggunakan lebih dari satu jasa yang tersedia, sehingga dalam pendataan yang dilakukan oleh pegawai service laptop tersebut dapat menjadi lebih mudah dan efisien.

DAFTAR PUSTAKA

Anonim. 2019. *Modul Praktikum Pemrograman Berorientasi Objek Program Studi Teknik Informatika Jurusan Teknik Informatika Fakultas Teknik Industri UPN “Veteran” Yogyakarta*. Yogyakarta: UPN “Veteran” Yogyakarta.