

LAPORAN TUGAS PROYEK PBO
“PROGRAM PEMINJAMAN BUKU”



Dipersiapkan oleh :

Nama : Widya Wardani

Stambuk : 13020180022

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS MUSLIM INDOESIA
MAKASSAR
2020

DAFTAR ISI

DAFTAR ISI.....

BAB I PENDAHULUAN

I.1 Latar Belakang

I.2 Rumusan Masalah

I.3 Batasan Permasalahan

BAB II PEMBAHASAN

II.1 Deskripsi Kasus.....

II.2 Analisis dan Desain Prosedur

II.3 Penjelasan Hasil Implementasi

II.4 Penjelasan Pengujian Program.....

BAB III PENUTUP

III.1 Kesimpulan

III.2 Saran

DAFTAR PUSTAKA

BAB I

PENDAHULUAN

I.1 Latar Belakang

Teknologi komputer merupakan salah satu teknologi yang paling banyak digunakann dalam pengembangan informasi. Karena memiliki banyak fungsi seperti sebagai sarana komunikasi, dokumentasi, pengolahan citra images dan sebagainya. Komputer dulu hanya digunakan sebagai alat untuk membantu manusia dalam proses untuk menghitung. Perkembangan zaman membuat komputer berubah sebagai alat untuk pengolah informasi yang sangat canggih.

Perkembangan teknologi komputer dari waktu ke waktu semakin berkembang denga pesat, mungkin saat ini dan seterusnya perkembangan teknologi komputer akan menunjukkan peningkatan lagi dan lagi kearah yang mungkin diluar jangkauan pemikiran kita sekarang. Seiring dengan semakin pesat dan makin vitalnya kegunaan akan teknologi komputer, maka manusia mulai mengembangkan teknologi komputer ke berbagai bidang, bahkan sampai ke hal – hal yang paling terkecil.

Dapat di lihat begitu banyak mahasiswa yang ingin melakukan peminjaman buku dan memiliki pengawas dengan jumlah yang banyak, akan tetapi system yang digunakan masih menggunakan system manual.

Karena data yang sangat banyak dan masih menggunakan system manual, maka timbul beberapa masalah sebagai berikut :

- Dokumen mahasiswa sangat banyak sehingga memerlukan waktu yang banyak untuk menginput semua data peminjaman.

Agar permasalahan tersebut dapat diatasi, maka saya akan mencoba membuat “Aplikasi Pinjaman Buku dan Denda ” program aplikasi ini diharapkan dapat memecahkan masalah peminjaman buku dan perthitungan denda.

I.2 Rumusan Masalah

- Bagaimana membuat aplikasi untuk menghitung denda peminjaman buku.

I.3 Batasan Masalah

- Aplikasi di batasi menggunakan PHP, MySQL dan NetBeans.
- System memudahkan menginput peminjaman buku dan menghitung denda.

BAB II

PEMBAHASAN

II.1 Deskripsi Kasus

Program untuk untuk mencatat buku yang dipinjam dan menghitung besar denda buku. Pengunjung yang akan meminjam buku harus terdaftar sebagai anggota perpustakaan dengan ketentuan sebagai berikut :

- a. Pengguna memasukkan jumlah buku dan identitas buku dan peminjam dimasukkan melalui keyboard
- b. Waktu peminjaman selama 1 minggu (5 hari kerja), Denda diberikan kepada peminjam yang terlambat mengembalikan buku melebihi dari lama peminjaman (1 minggu).

Jika terlambat setiap 1 minggu maka setiap buku didenda sebesar Rp. 1000/buku dan kelipatannya (jika 2 minggu terlambat mengembalikan maka setiap buku didenda 2 X Rp.1000/buku dst..)

Jika buku tersebut hilang, maka pengunjung wajib mengganti buku atau membayar sebesar harga buku.

II.2 Analisis dan Desain Prosedur

- Flowchart dan Algoritma

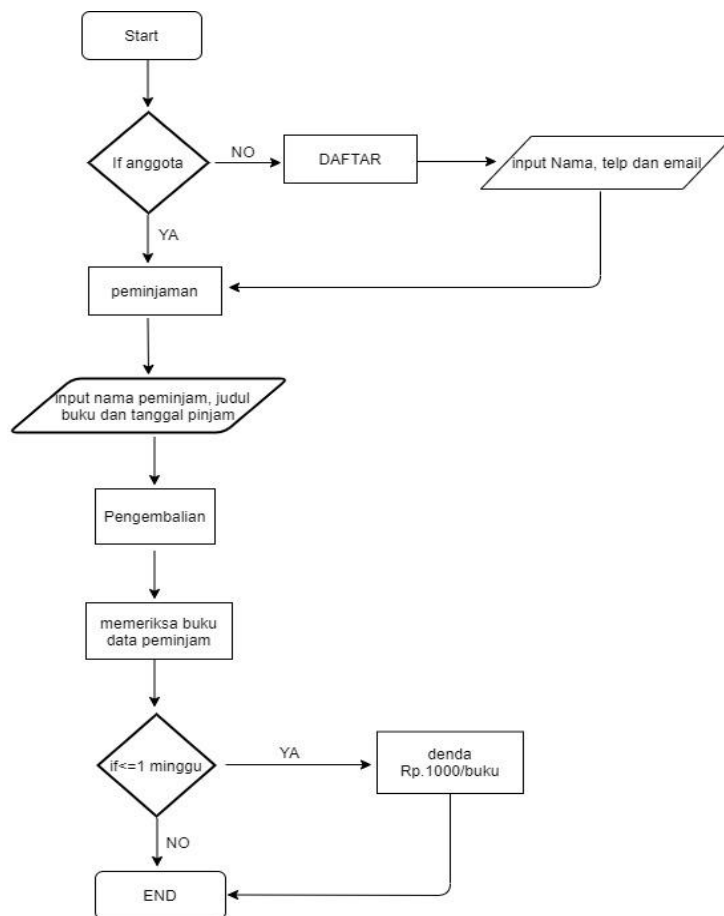
Algoritma Flowchart:

1. Menu start
2. Anggota :
 - Menginput data anggota.
 - Jika sudah terdaftar sebagai anggota, maka sudah bisa meminjam buku.
 - Jika belum terdaftar maka calon anggota harus mengisi data.
3. Peminjaman buku : Menginput kode buku, nama buku, id anggota, nama anggota, tgl pinjam, tgl kembali.
4. Pengembalian buku :

- Periksa data anggota.
- Jika pengembalian buku lewat dari 7 hari maka anggota wajib membayar denda Rp. 1000/buku.

5. Selesai

FLOWCHART



- konsep pemrograman berorientasi objek:

Encapsulation :

```
package com.database.program;
import java.awt.HeadlessException;
import java.sql.Connection;
import javax.swing.table.DefaultTableModel;
import java.sql.SQLException;
import javax.swing.JOptionPane;

/**
 *
 * @author user
 */
public class DataAnggota extends javax.swing.JFrame {

    private void kosongkan_form(){
        txtIdAnggota.setEditable(true);
        txtIdAnggota.setText(null);
        txtNama.setText(null);
        txtJurusan.setText(null);
        txtTelp.setText(null);
        txtEmail.setText(null);
    }

    private void tampilkan_data(){

        DefaultTableModel model = new DefaultTableModel();
        model.addColumn("No");
```

```
model.addColumn("IdAnggota");
model.addColumn("Nama");
model.addColumn("Jurusan");
model.addColumn("Telp");
model.addColumn("Email");

try{
    int No = 1;
    String sql = "SELECT * FROM DataAnggota where
IdAnggota like '%"
        + txtcari.getText() + "%"
        + "or Nama like '%" + txtcari.getText()
        + "%" + "or Jurusan like '%" + txtcari.getText()
        + "%" + "or Telp like '%" + txtcari.getText()
        + "%" + "or Email like '%" + txtcari.getText() + "%";
    java.sql.Connection conn =
(Connection)DatabasePerpustakaan.configDB();
    java.sql.Statement stm = conn.createStatement();
    java.sql.ResultSet res = stm.executeQuery(sql);

    while(res.next()){
        model.addRow(new
Object[]{No++,res.getString(1),res.getString(2),res.getString(3),res.
getString(4),res.getString(5)});
    }
    tabelanggota.setModel(model);
}
```



```

    }catch (SQLException e){
        System.out.println("Error : " + e.getMessage());
    }

```

II.3 Penjelasan Hasil Implementasi

Ada 3 tabel yang saya buat dan tampilannay semua sama kecuali tampilan hitung denda seperti berikut:

The screenshot shows a Java Swing window titled "DATA ANGGOTA". It features a form with five text input fields labeled "IdAnggota", "Nama", "Jurusan", "Telp", and "Email". Below the form are five buttons: "save", "cancel", "edit", "delete", and "exit". At the bottom of the window, there is a table with 4 columns labeled "Title 1", "Title 2", "Title 3", and "Title 4", and 5 rows. To the right of the table is a search bar with a "search" button.

PENJELASAN:

- menginput id anggota, nama, jurusan, telp dan email dengan menekan tombol save.

Sintax :

```

private void btnSaveActionPerformed(java.awt.event.ActionEvent
    evt) {

```

```

        // TODO add your handling code here:
        try{
            String sql = "INSERT INTO dataanggota VALUES (" +
txtIdAnggota.getText()+"," +txtNama.getText()+"," +txtJurusan.get
Text()+"," +txtTelp.getText()+"," +txtEmail.getText()+")";
            java.sql.Connection conn =
(Connection)DatabasePerpustakaan.configDB();
            java.sql.PreparedStatement pstm =
conn.prepareStatement(sql);
            pstm.execute();
            JOptionPane.showMessageDialog(null, "Proses Simpan
Data Berhasil..");
            tampilkan_data();
            kosongkan_form();

        }catch(HeadlessException | SQLException e){
            JOptionPane.showMessageDialog(this, e.getMessage());
        }
    }
}

```

- syntax menu cancel

```

private void txtcancelActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here:
    kosongkan_form();
}

```

- syntax menu edit :

```
private void beditActionPerformed(java.awt.event.ActionEvent evt)
{
    // TODO add your handling code here:
    try {
        String sql = "UPDATE dataanggota SET IdAnggota=" +
txtIdAnggota.getText()
        + ",Nama=" + txtNama.getText()
        + ",Jurusan=" + txtJurusan.getText()
        + ",Telp=" + txtTelp.getText()
        + ",Email=" + txtEmail.getText()
        + " WHERE IdAnggota=" + txtIdAnggota.getText() +"";
        java.sql.Connection conn = (Connection)
DatabasePerpustakaan.configDB();
        java.sql.PreparedStatement pstm = conn.prepareStatement(sql);
        pstm.execute();
        JOptionPane.showMessageDialog(null, "Edit Data
Berhasil...");
    } catch (HeadlessException | SQLException e){
        JOptionPane.showMessageDialog(this, e.getMessage());
    }
    tampilkan_data();
    kosongkan_form();
}
```

- syntax menu delete :

```
private void btdeleteActionPerformed(java.awt.event.ActionEvent
evt) {
    try{
        String sql = "Delete FROM dataanggota WHERE IdAnggota
        =" + txtIdAnggota.getText() + """;
        java.sql.Connection conn = (Connection)
        DatabasePerpustakaan.configDB();
        java.sql.PreparedStatement pstmt =
        conn.prepareStatement(sql);
        pstmt.execute();
        JOptionPane.showMessageDialog(null, "Hapus Data
        Berhasil..");

        }catch (HeadlessException | SQLException e){
            JOptionPane.showMessageDialog(this, e.getMessage());
        }
        tampilkan_data();
        kosongkan_form();
    }
```

- syntax menu exit :

```
private void btxitActionPerformed(java.awt.event.ActionEvent evt)
{
    System.exit(0);
}
```

II.4 Penjelasan Pengujian Program

Hasil pengujian program

The screenshot shows a window titled "DATA ANGGOTA" with the following components:

- Input fields for: IdAnggota, Nama, Jurusan, Telp, and Email.
- Buttons: save, cancel, edit, delete, and exit.
- A table with 6 columns: No, IdAnggota, Nama, Jurusan, Telp, and Email.
- A search bar with a "search" button.

| No | IdAnggota | Nama | Jurusan | Telp | Email |
|----|-------------|---------|--------------------|--------------|-----------------|
| 1 | 13020180044 | andre | TeknikInformati... | 081529487345 | andremaulana... |
| 2 | 13020180055 | sindy | farmasi | 082134876213 | sindyariska.com |
| 3 | 13020190029 | indahhh | ekonomi | 83493392023 | iindah@gmail... |

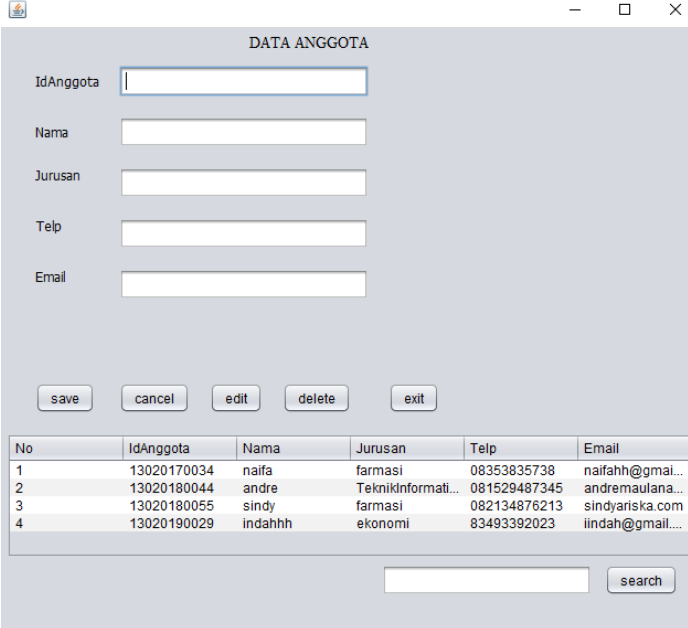
Output di atas ada table beserta isi nya karena pada MySQL sudah di inputkan data, maka tampilkan seperti di atas

- hasil pengujian menu save :

This screenshot shows the same "DATA ANGGOTA" window after a new record has been saved. A "Message" dialog box is displayed with the text "Proses Simpan Data Berhasil.." and an "OK" button. The input fields now contain the new data: IdAnggota: 13020170034, Nama: naifa, Jurusan: farmasi, Telp: 08353835738, Email: naifahh@gmail.com. The table below remains unchanged from the previous screenshot.

| No | IdAnggota | Nama | Jurusan | Telp | Email |
|----|-------------|---------|--------------------|--------------|-----------------|
| 1 | 13020180044 | andre | TeknikInformati... | 081529487345 | andremaulana... |
| 2 | 13020180055 | sindy | farmasi | 082134876213 | sindyariska.com |
| 3 | 13020190029 | indahhh | ekonomi | 83493392023 | iindah@gmail... |

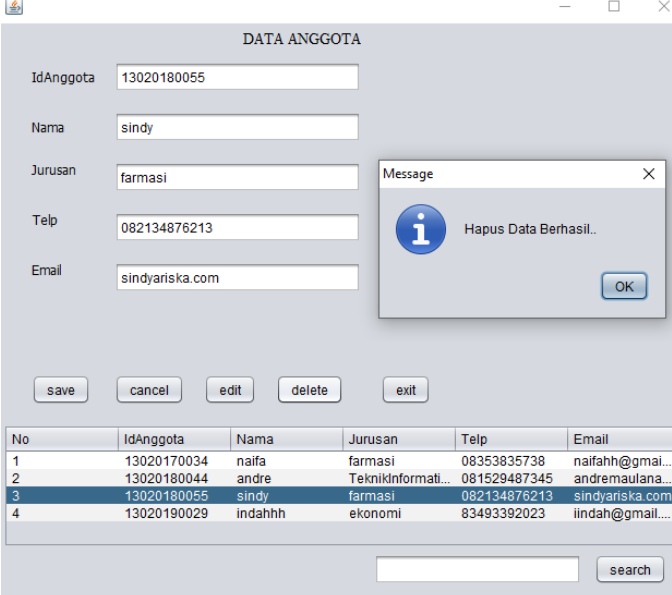
Apabila muncul kata “Proses simpan data berhasil “ maka data yang di inputkan tadi masuk di table, seperti berikut :



The screenshot shows a window titled "DATA ANGGOTA". It contains a form with the following fields: IdAnggota, Nama, Jurusan, Telp, and Email. Below the form are buttons for "save", "cancel", "edit", "delete", and "exit". At the bottom, there is a table with 6 columns: No, IdAnggota, Nama, Jurusan, Telp, and Email. The table contains 4 rows of data. Below the table is a search bar with a "search" button.

| No | IdAnggota | Nama | Jurusan | Telp | Email |
|----|-------------|---------|--------------------|--------------|------------------|
| 1 | 13020170034 | naifa | farmasi | 08353835738 | naifahh@gmail... |
| 2 | 13020180044 | andre | TeknikInformati... | 081529487345 | andremaulana... |
| 3 | 13020180055 | sindy | farmasi | 082134876213 | sindyariska.com |
| 4 | 13020190029 | indahhh | ekonomi | 83493392023 | iindah@gmail.... |

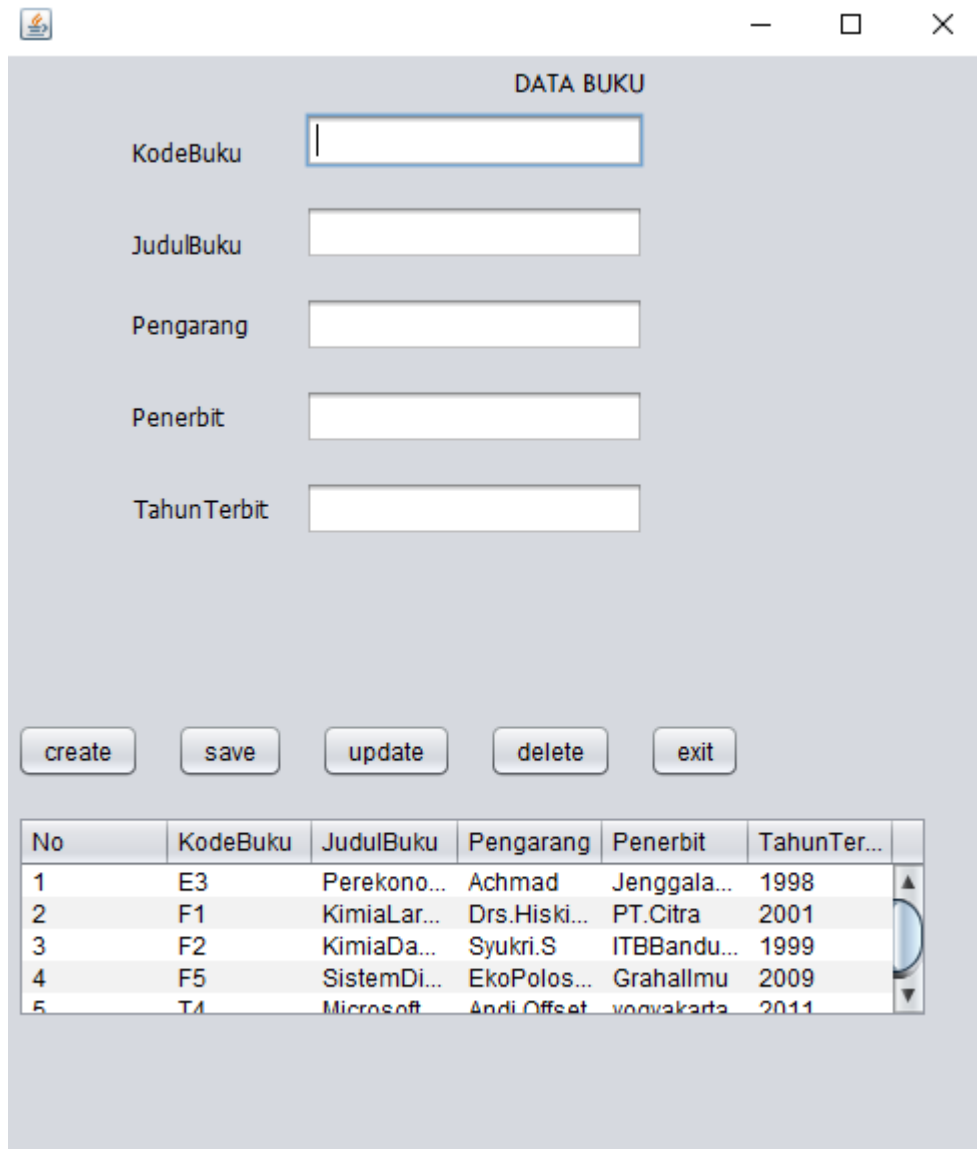
- hasil pengujian menu delete



The screenshot shows the same "DATA ANGGOTA" window, but with a "Message" dialog box open. The dialog box has a blue information icon and the text "Hapus Data Berhasil..". There is an "OK" button at the bottom right of the dialog box. The form fields are filled with: IdAnggota: 13020180055, Nama: sindy, Jurusan: farmasi, Telp: 082134876213, Email: sindyariska.com. The table below the form shows the same data as the previous screenshot, but the row with IdAnggota 13020180055 is highlighted in blue.

| No | IdAnggota | Nama | Jurusan | Telp | Email |
|----|-------------|---------|--------------------|--------------|------------------|
| 1 | 13020170034 | naifa | farmasi | 08353835738 | naifahh@gmail... |
| 2 | 13020180044 | andre | TeknikInformati... | 081529487345 | andremaulana... |
| 3 | 13020180055 | sindy | farmasi | 082134876213 | sindyariska.com |
| 4 | 13020190029 | indahhh | ekonomi | 83493392023 | iindah@gmail.... |

- output data buku :



DATA BUKU

KodeBuku

JudulBuku

Pengarang

Penerbit

TahunTerbit

| No | KodeBuku | JudulBuku | Pengarang | Penerbit | TahunTerbit |
|----|----------|-------------|--------------|-------------|-------------|
| 1 | E3 | Perekono... | Achmad | Jenggala... | 1998 |
| 2 | F1 | KimiaLar... | Drs.Hiski... | PT.Citra | 2001 |
| 3 | F2 | KimiaDa... | Syukri.S | ITBBandu... | 1999 |
| 4 | F5 | SistemDi... | EkoPolos... | Grahallmu | 2009 |
| 5 | T4 | Microsoft | Andi Offset | voovakarta | 2011 |

- hasil hitung denda

PROGRAM MENGHITUNG DENDA PEMINJAMAN BUKU

Input Nilai

IdAnggota: 13020190070

Nama: hendra

JumlahBuku: 5

Denda: 1000

Output Nilai

IdAnggota :

Nama :

TotalDenda:

Hitung Reset Simpan Keluar

Apabila kita menekan menu hitung maka total denda akan muncul.

- Sintax menu hitung :

```
private void bthitungActionPerformed(java.awt.event.ActionEvent  
evt) {  
    // TODO add your handling code here:  
    String IdAnggota = txtIdAnggota.getText();  
    String Nama = txtNama.getText();  
    if (txtIdAnggota.getText().isEmpty() ||  
    txtNama.getText().isEmpty()  
        || txtJumlah.getText().isEmpty()  
        || txtDenda.getText().isEmpty()){
```



```
        JOptionPane.showMessageDialog(null, "Form Masih  
Kosong!!");  
    }  
    else {  
        Double TotalDenda =  
(Double.parseDouble(txtDenda.getText())+Double.parseDouble(t  
xtJumlah.getText()));  
        outTotalDenda.setText(TotalDenda.toString());  
        outIdAnggota.setText(IdAnggota);  
        outNama.setText>Nama);  
    }  
}
```

BAB III

PENUTUP

III.1 Kesimpulan

Program untuk untuk mencatat buku yang dipinjam dan menghitung besar denda buku. Pengunjung yang akan meminjam buku harus terdaftar sebagai anggota perpustakaan dengan ketentuan sebagai berikut :

- a. Pengguna memasukkan jumlah buku dan identitas buku dan peminjam dimasukkan melalui keyboard
- b. Waktu peminjaman selama 1 minggu (5 hari kerja), Denda diberikan kepada peminjam yang terlambat mengembalikan buku melebihi dari lama peminjaman (1 minggu).

Jika terlambat setiap 1 minggu maka setiap buku didenda sebesar Rp. 1000/buku dan kelipatannya (jika 2 minggu terlambat mengembalikan maka setiap buku didenda 2 X Rp.1000/buku dst..)

Jika buku tersebut hilang, maka pengunjung wajib mengganti buku atau membayar sebesar harga buku.