

LAPORAN TUGAS PROYEK

“APLIKASI MENGHITUNG NILAI AKHIR MAHASISWA”



Sitti Ajriana A.
13020180051
B1

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

PENDAHULUAN

A. Deskripsi Aplikasi

Aplikasi ini membahas mengenai cara membuat program java GUI berbasis MVC menggunakan database dalam menghitung sebuah nilai mahasiswa, dari nilai UTS,UAS, Tugas Mandiri/Kuis,Kehadiran serta Tugas Proyek sehingga mendapatkan sebuah hasil berupa nilai akhir, dan nilai huruf.

Presentasi nilai pada program ini ialah:

Nilai UTS (20%), Nilai UAS(25%),Nilai Tugas Mandiri(20%),Nilai Tugas Proyek(20%) dan Kehadiran(15%)

Dimana ketentuan penilaian tersebut sebagai berikut:

1. apabila final score nilai akhir mahasiswa >85 maka memperoleh nilai A(4.00), $(\text{nilai} < 85) \&\& (\text{nilai} \geq 81)$ maka memperoleh nilai A-(3.75), $(\text{nilai} \leq 80) \&\& (\text{nilai} \geq 76)$ maka memperoleh nilai B+(3.50), $(\text{nilai} \leq 75) \&\& (\text{nilai} \geq 71)$ maka memperoleh nilai B(3.00), $(\text{nilai} \leq 70) \&\& (\text{nilai} \geq 66)$ maka memperoleh nilai B-(2.75), $(\text{nilai} \leq 65) \&\& (\text{nilai} \geq 61)$ maka memperoleh nilai C+(2.50), $(\text{nilai} \leq 60) \&\& (\text{nilai} \geq 51)$ maka memperoleh nilai C(2.00), $(\text{nilai} \leq 50) \&\& (\text{nilai} \geq 45)$ maka memperoleh nilai D(1.00), Dan apabila final score nilai akhir mahasiswa <45 maka memperoleh nilai E(0.00).

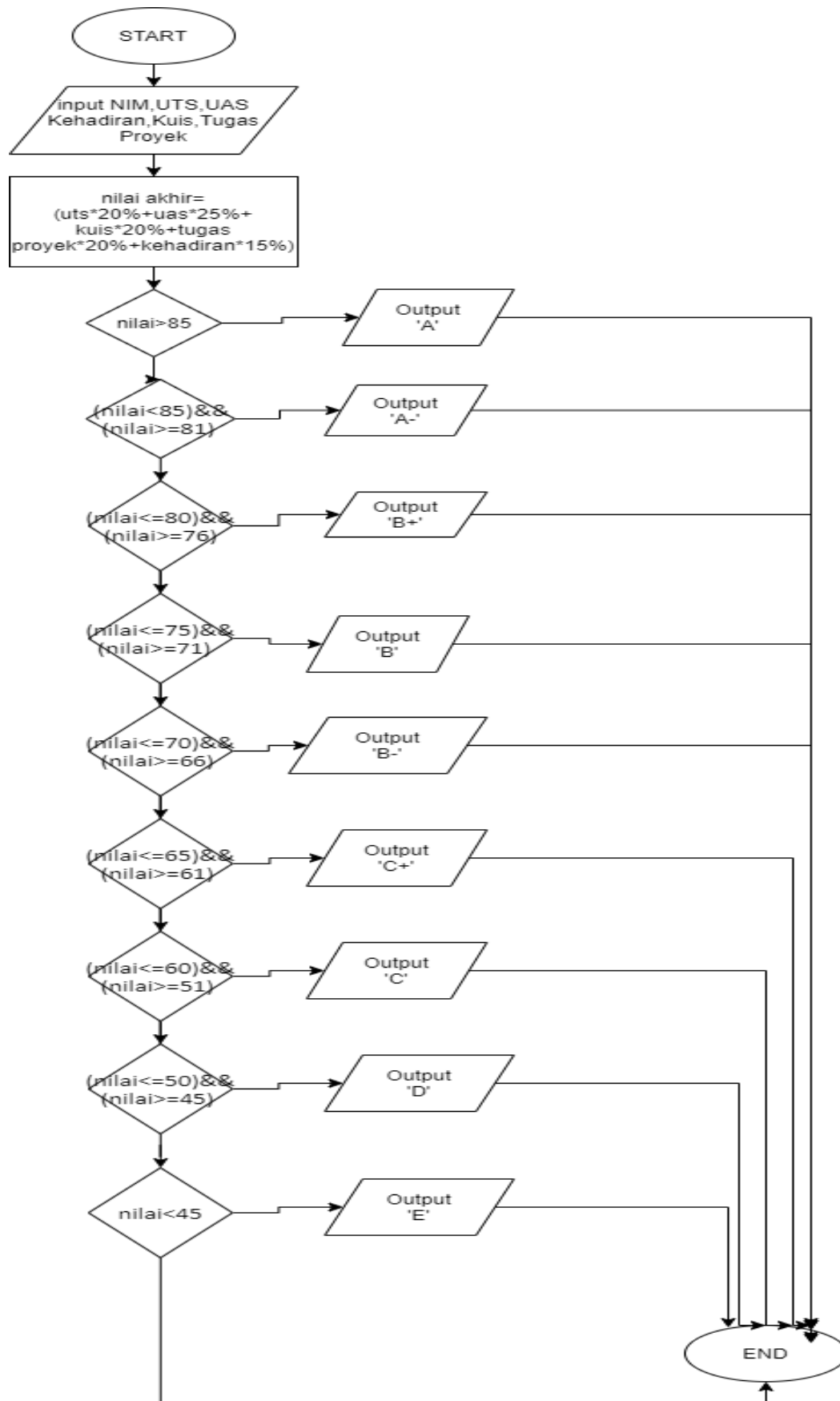
Proses Inputnya adalah:

1. NIM Mahasiswa
2. Kehadiran
3. Nilai UTS
4. Nilai UAS
5. Nilai Kuis
6. Tugas Proyek

Proses Outputnya:

1. Nilai Akhir
2. Nilai Huruf

B. Flowchart:



C. Penjelasan hasil Implementasi :

1. Pembuatan database:

```
1 • create database Nilai;
2 • use Nilai;
3
4 • create table Mhs ( nim varchar(12) primary key, uts int, kuis int, uas int, tp int, kehadiran int, nilai_akhir int,
5   grade varchar(12));
6
7 • drop table Mhs;
8
9 • select * from Mhs;
```

2. Koneksi Db untuk mengkoneksikan antara database dan program

```
public class KoneksiDb {
    private static java.sql.Connection mysqlconfig;
    public static java.sql.Connection configDB() throws SQLException {
        try {
            String url="jdbc:mysql://localhost:3306/Mahasiswa"; //url database
            String user="root"; //user database
            String pass=""; //password database
            DriverManager.registerDriver(new com.mysql.jdbc.Driver());
            mysqlconfig=DriverManager.getConnection(url, user, pass);
        } catch (Exception e) {
            System.err.println("koneksi gagal "+e.getMessage()); //perintah menampilkan error pada koneksi
        }
        return mysqlconfig;
    }
}
```

3. Membuat Library agar tidak terjadi error ketika pemanggilan karena library belum dipanggil

```
import java.awt.Frame;
import java.sql.Connection;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import javax.swing.event.ListSelectionEvent;
import javax.swing.event.ListSelectionListener;
import javax.swing.table.DefaultTableModel;
import javax.swing.table.JTableHeader;
```

4. Membuat Table Model

```
public class FormNilai extends javax.swing.JFrame {
    DefaultTableModel model;
```

5. Membuat Method setTable(); untuk mensetting isi table yang akan kita buat, membuat header table, variable field itu untuk menampung nama kolom lalu itu dijadikan table model

```
private void setTable(){
    Object[] field = {"Mata Kuliah", "Kehadiran", "UTS", "KUIS", "UAS", "TUGAS PROYEK", "Nilai Akhir", "Nilai Huruf"};
    model = new DefaultTableModel(field, 0);
    jTable1.setModel(model);
    JTableHeader th = jTable1.getTableHeader();
    th.setReorderingAllowed(false);
}
```

6. Membuat Method calculate (String... utm) untuk menghitung semua nilai sesuai dengan presentasi nilai yang ada.

Arti String... utm pada parameter method adalah berapapun parameter yang dibutuhkan maka sebenarnya program tetap bisa berjalan dengan lancar selama nilai parameter pemanggilnya bertipe string.

```
private String calculate(String... utm){  
    for (int i = 0; i < utm.length; i++){  
        if(utm[i].isEmpty()){  
            utm[i] = "0";  
        }  
    }  
    jTable1.setRowHeight(30);  
}
```

Tetapi pada program ini tidak saya buat seperti itu, karena disitu ada code double t1=0,t2=0,t3=0,t4=0,t5=0; lalu switch case dan double jumlah = t1+t2+t3+t4+t5; yang sudah jelas parameternya hanya bisa menampung 5 nilai. Math Round yang digunakan untuk membulatkan bilangan float(pecahan)

```
,  
double t1=0,t2=0,t3=0,t4=0,t5=0;  
for (int i = 0; i < utm.length; i++) {  
    double cv = Double.valueOf(utm[i]);  
    switch (i) {  
        case 0:  
            t1 = cv * 0.15;  
            break;  
        case 1:  
            t2 = cv * 0.2;  
            break;  
        case 2:  
            t3 = cv * 0.2;  
            break;  
        case 3:  
            t4 = cv * 0.25;  
            break;  
        case 4:  
            t5 = cv * 0.2;  
            break;  
        default:  
            break;  
    }  
}  
},  
}  
double jumlah = t1+t2+t3+t4+t5;  
double as = Math.round(jumlah);  
String cvJumlah = indexOF(as);  
return cvJumlah;  
}
```

7. Membuat Method indexOF(double b) untuk mendeteksi bilangan, jika bilangan itu bernilai 0 maka hilangkan selain itu hanya merubah tipe data saja, method ini digunakan pada method calculate

```
public String indexOF(double b){
    String hsl;
    if(b %1 == 0){
        hsl = String.valueOf(b).substring(0,String.valueOf(b).lastIndexOf("."));
    }else{
        hsl = String.valueOf(b);
    }
    return hsl;
}
```

8. Membuat Method calAbjad(String jumlah) untuk memberikan nilai berupa huruf yang sudah diprogram pada method ini.

```
private String calAbjad(String jumlah){
    String grade = "";
    double nilai = Double.valueOf(jumlah);
    if ( nilai >= 85 ) {
        grade = "A";
    } else if ((nilai<85)&&(nilai>=81)){
        grade = "A-";
    } else if ((nilai<=80)&&(nilai>=76)){
        grade = "B+";
    } else if ((nilai<=75)&&(nilai>=71)) {
```

9. Pemanggilan method setTable(); untuk mensetting table saat awal program dijalankan, jadi code pada method setTable() seperti kolom kolom dibutuhkan akan disetting disini.

```
public FormNilai() throws SQLException {
    initComponents();
    KoneksiDb.configDB();
    Services = new impl();
    ambil_data_tabel();
    setTable();
}
```

10. Membuat actionPerformed pada Button hitung

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    String JM = txtJM.getText();  
    String hadir = txtKehadiran.getText();  
    String uts = txtuts.getText();  
    String kuis = txtNK.getText();  
    String uas = txtuas.getText();  
    String tp = txtTP.getText();  
  
    if(JM.isEmpty()){  
        JOptionPane.showMessageDialog(this, "Masukan Jumlah Mahasiswa Terlebih Dahulu");  
        txtJM.requestFocus();  
    }else{  
        String[] data = {hadir, uts, kuis, uas, tp};  
  
        String jml = calculate(data);  
        Object[] data2 = {JM, hadir, uts, kuis, uas, tp, jml, calAbjad(jml)};  
        model.addRow(data2);  
    }  
}
```

11. ActionPerformed “Clear Field” artinya membersihkan/menghapus semua isi field yang sebelumnya ada tulisannya. Menggunakan setText karena akan mensetting variabel dari pallet text field untuk mensettingnya menjadi kosong

```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    txtJM.setText("");  
    txtKehadiran.setText("");  
    txtuts.setText("");  
    txtNK.setText("");  
    txtuas.setText("");  
    txtTP.setText("");  
}
```

12. ActionPerformed “Clear Row” untuk menghapus row (baris) pada tabel.

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    int[] pilih = jTable1.getSelectedRows();  
    if(pilih.length > 0){  
        for(int i = pilih.length-1; i >= 0; i--){  
            model.removeRow(pilih[i]);  
        }  
    }else{  
        JOptionPane.showMessageDialog(this, "Pilih row yang ingin dihapus");  
    }  
}
```


13.Actionperformed “Edit” untuk mengedit field pada tabel

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try {  
        // TODO add your handling code here:  
        profil p = new profil();  
  
        if (txtJM.getText().equals("") && txtuts.getText().equals("") && txtNK.getText().equals("") &&  
            txtuas.getText().equals("") && txtTP.getText().equals("") && txtKehadiran.getText().equals("")) {  
            JOptionPane.showMessageDialog(this, "silahkan isi data yang akan diubah");  
        } else {  
            // p.setId(Integer.parseInt(txtId.getText()));  
            p.setNim(txtJM.getText());  
            p.setUts(Integer.parseInt(txtuts.getText()));  
            p.setUas(Integer.parseInt(txtuas.getText()));  
            p.setKuis(Integer.parseInt(txtNK.getText()));  
            p.setTp(Integer.parseInt(txtTP.getText()));  
            p.setAbsen(Integer.parseInt(txtKehadiran.getText()));  
            Services.update(p);  
  
            ambil_data_tabel();  
  
            JOptionPane.showMessageDialog(this, "data berhasil diubah");  
        }  
    } catch (SQLException ex) {  
        JOptionPane.showMessageDialog(this, "data gagal diubah");  
    }  
}
```

14.Actionperformed “Keluar” untuk keluar dari program

```
private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    int replay = JOptionPane.showConfirmDialog(  
        null,  
        "yakin anda ingin keluar?",  
        "konfirmasi keluar aplikasi",  
        JOptionPane.YES_NO_OPTION);  
    if (replay==JOptionPane.YES_OPTION){  
        System.exit(0);  
    }  
}
```

D. Pengujian Program

NIM

Kehadiran

Nilai UTS

Nilai Kuis

Nilai UAS

Tugas Proyek

PROSES

EDIT

CLEAR ROW

CLEAR FIELD

KELUAR

Mata K...	Kehadi...	UTS	KUIS	UAS	TUGAS...	Nilai Ak...	Nilai H...

NIM

Kehadiran

Nilai UTS

Nilai Kuis

Nilai UAS

Tugas Proyek

PROSES

EDIT

CLEAR ROW

CLEAR FIELD

KELUAR

Mata K...	Kehadi...	UTS	KUIS	UAS	TUGAS...	Nilai Ak...	Nilai H...
123	80	86	90	90	85	87	A

PENUTUP

KESIMPULAN

Pada Pemrograman Java ini saya menggunakan Netbeans IDE 8.0.1 sebagai editor. Dari hasil pembuatan ini, Saya dapat memahami dan mengerti cara merancang tampilan form dari sebuah aplikasi dan mengerti fungsi dari source kode masing-masing form dan menghubungkannya ke database.

SARAN

Semoga praktikum bahasa pemograman java lebih di tingkatkan lagi. Agar mahasiswa dapat membangun sebuah aplikasi atau sistem informasi yang kemudian bisa diterapkan dalam kehidupan sebagai kebutuhan akan informasi.

DAFTAR PUSTAKA

<https://www.ismynr.xyz/2019/02/program-java-gui-menghitung-nilai-mahasiswa.html>
waktu akses : 16 juni 2020