

LAPORAN TUGAS PROYEK

(Menghitung Gaji Kariawan)

Di Susun

OLEH :



Heryanti

13020180074

Teknik Informatika

JURUSAN TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS MUSLIM INDONESIA

2020

KATA PENGANTAR

Alhamdulillahirabbil'alamin, Puji syukur kehadiran Allah SWT, atas segala rahmat dan hidayah-Nya. Shalawat serta salam senantiasa tercurahkan kepada junjungan Nabi Agung Muhammad SAW yang selalu kita nantikan syafa'atnya di akhirat nanti.

Penulis mengucapkan syukur kepada Allah SWT atas limpahan nikmat sehat-Nya, baik itu berupa sehat fisik maupun akal pikiran, sehingga penulis mampu untuk menyelesaikan pembuatan makalah sebagai tugas akhir dari mata kuliah Hukum Acara Peradilan Agama dengan judul “Prosedur Pengajuan Gugatan”.

Penulis tentu menyadari bahwa makalah ini masih jauh dari kata sempurna dan masih banyak terdapat kesalahan serta kekurangan di dalamnya. Untuk itu, penulis mengharapkan kritik serta saran dari pembaca untuk makalah ini, supaya makalah ini nantinya dapat menjadi makalah yang lebih baik lagi. Demikian, dan apabila terdapat banyak kesalahan pada makalah ini penulis mohon maaf yang sebesar-besarnya.

Enrekang, 20 juni 2020

Penyusun

DAFTAR ISI

Kata pengantar.....	i
Daftar isi.....	ii
Bab I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	1
1.3 Tujuan.....	1
Bab II PEMBAHASAN	
Bab III DESAIN DAN IMPLEMENTASI	
2.1 Algoritma flowchart menghitung gaji kariawan.....	2
2.2 implementasi dan script java menghitung gaji kariawan.....	2
Bab IV PENUTUP	
3.1 Kesimpulan	10
3.2 Saran.....	10
DAFTAR PUSTAKA	

BAB I

PENDAHULUAN

1.1.Latar Belakang

Sistem penggajian dapat digolongkan dalam dua sistem, yaitu sistem skala tunggal dan sistem skala ganda. Sistem skala tunggal adalah system penggajian yang memberikan gaji yang sama kepada pegawai yang berpangkat sama dengan tidak atau kurang memperhatikan sifat pekerjaan yang dilakukan dan beratnya tanggung jawab pekerjaannya. Sistem skala ganda adalah sistem penggajian yang menentukan besarnya gaji bukan saja didasarkan pada pangkat, tetapi juga didasarkan pada sifat pekerjaan yang dilakukan, prestasi kerja yang dicapai dan beratnya tanggung jawab pekerjaannya. Selain kedua sistem penggajian tersebut dikenal juga sistem penggajian ketiga yang disebut sistem skala gabungan, yang merupakan perpaduan antara sistem skala tunggal dan sistem skala ganda. Dalam sistem skala gabungan, gaji pokok ditentukan sama bagi pegawai negeri yang berpangkat sama, di samping itu diberikan tunjangan kepada Pegawai Negeri yang memikul tanggung jawab yang lebih berat, prestasi yang tinggi atau melakukan pekerjaan tertentu yang sifatnya memerlukan pemusatan perhatian dan pengerahan tenaga secara terus-menerus.

1.2 Rumusan Masalah

- Deskripsi Gaji Pokok & Tunjangan Pengabdian Berdasarkan Golongan
- Analisis desain dan prosedur atau mekanisme kerja menggunakan algoritma atau flowcart dan menerapkan konsep pemrograman berorientasi objek Enkapsulasi,inheritance,polimorphsim, dalam program.
- Penjelasan hasil implementasi berisi capture dan penggalan kode pada fungsi utama beserta penjelasannya.
- Penjelasan pegujian program disertai dengan hasil pengujiannya berupa capture luaran.

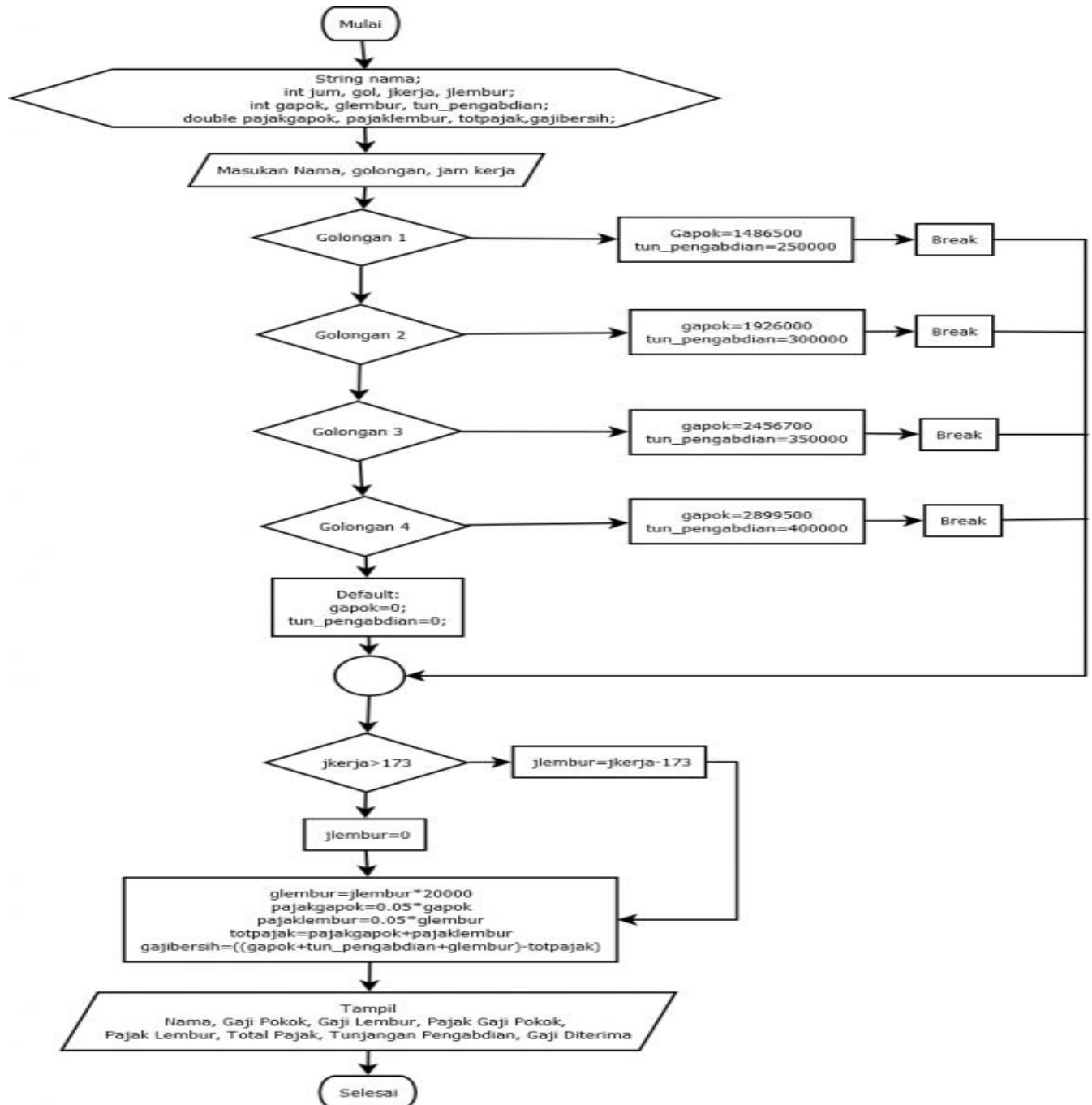
1.3 Tujuan

1. Mengetahui dan memahami aplikasi menghitung gaji kariawan yang telah dirancang
2. Mengetahui dan memahami proses kerja aplikasi menghitung gaji kariawan menggunakan neatbens.

BAB III

DESAIN DAN IMPLEMENTASI

2.1 Algoritma flowchart menghitung gaji kariawan



Program akan berjalan secara sekuensial, dimana pada bagian pertama kita lakukan inisialisasi semua variabel yang dibutuhkan, user akan memasukan nama, golongan dan jumlah jam kerja, gaji pokok dan tunjangan pengabdian akan ditentukan berdasarkan golongan yang dimasukan.

Jumlah jam kerja normal didalam sebulan diasumsikan adalah **173** jam, jika melebihi jam tersebut jam berikutnya akan dihitung sebagai jam lembur dengan rumus jam lembur=jam kerja-173. Apabila jam kerja sama dengan 173 atau kurang dari jam tersebut artinya bahwa jam lembur karyawan tersebut 0.

2.2 implementasi dan script java menghitung gaji kariawan

2.2.1.

2.2.2.

```

27
28
29 public final class Form_Gaji {
30     private DefaultTableModel model;
31     String nip, nama, golongan;
32     int gapok, jamkerja, lembur;
33 }
34
35 public void loadData() {
36     txtnip.getText();
37     txtnama.getText();
38     cmbgol.getSelectedItem();
39 }

```

Membuat variable baru untuk menampung nilai sementara dari Nip, Nama, golongan, Gaji Pokok, lembur dan jam kerja, masukkan variable tersebut pada class utama yaitu class.

2.2.3 Membuat class loadGaji

```

public void loadGaji() {
    golongan = "" + cmbgol.getSelectedItem();
    switch(golongan){
        case "I":
            gapok = 500000;
            break;
        case "II":
            gapok = 300000;
            break;
        case "III":
            gapok = 250000;
            break;
        case "IV":
            gapok = 100000;
            break;
    }
    jamkerja = (int) (gapok * 0.1);
    lembur = gapok + jamkerja;
    txtgaji.setText(""+gapok);
    txtjam.setText(""+jam kerja);
}

```

2.2.4. Membuat class datatable

```

private void datatable(){
    DefaultTableModel tbl = new DefaultTableModel();
    tbl.addColumn("nip");
    tbl.addColumn("Nama pegawai");
    tbl.addColumn("golongan");
    tbl.addColumn("gaji pokok");
    tbl.addColumn("jam kerja");
    tbl.addColumn("lembur");

    try{
        Statement statement = (Statement) koneksi.getConnection().createStatement();
        ResultSet res = statement.executeQuery("select * from tabel_mahasiswa");
        while(res.next()){
            tbl.addRow(new Object[]{
                res.getString("nip"),
                res.getString("Nama pegawai"),
                res.getString("golongan"),
                res.getString("gaji pokok"),
                res.getString("jam kerja"),
                res.getString("lembur")
            });
        }
    } catch (Exception e) {
        JOptionPane.showMessageDialog(rootPane, "Gagal");
    }
}

```

2.2.5 Memanggil class save

Pada tahap ini merupakan tahap pembuatan class saveData(), fungsi class ini adalah untuk menyimpan data berdasarkan input dari user, berikut kode program pada class saveData().

```

88 private void BkeluarActionPerformed(java.awt.event.ActionEvent evt) {
89     dispose();
90 }
91 private void btnSaveActionPerformed(java.awt.event.ActionEvent evt) {
92     saveData();
93 }
94 public void saveData() {
95     loadData();
96     try{
97         Statement stat = (Statement) Db_Connection.getKoneksi().createStatement();
98         String sql = "Insert into gaji (Nip, Nama, golongan, gapok, jam kerja, lembur)"
99             + " values ('"+ nip + "', '"+ nama + "', '"+ golongan + "', '"+ gapok + "', "
100             + "'"+ jam kerja + "', '"+ lembur + "')";
101         PreparedStatement p = (PreparedStatement) Db_Connection.getKoneksi().prepareStatement(sql);
102         p.executeUpdate();
103         getData();
104     } catch (SQLException err){
105         JOptionPane.showMessageDialog(null, err.getMessage());
106     }
107     datatable();
108 }
109 }
110

```

2.2.6 Membuat tombol reset

```
private void btnResetActionPerformed(java.awt.event.ActionEvent evt){
    Reset();
}

public void dataSelect(){
    int i = tblGaji.getSelectedRow();
    if (i == -1){
        //tidak ada data terpilih
        return;
    }
    txtnip.setText(""+model.getValueAt(i, 0));
    txtnama.setText(""+model.getValueAt(i, 1));
    cmbgol.setSelectedItem(""+model.getValueAt(i, 2));
    txtgaji.setText(""+model.getValueAt(i, 3));
    txtjam.setText(""+model.getValueAt(i, 4));
    txtlembur.setText(""+model.getValueAt(i, 5));
}
```

2.2.7 Membuat button Update

```
public void updateData(){
    /* memanggil class loadData() untuk menentukan kondisi atau variable nip
       yang akan diubah, berdasarkan yang dipilih
    */
    loadData();
    try{
        Statement stat = (Statement) Db_Connection.getConnection().createStatement();
        String sql = "UPDATE gaji SET Nama = '"+ nama + "',"
                    + "golongan = '"+ golongan + "',"
                    + "Gapok = '"+ gapok + "',"
                    + "jam kerja = '"+ jamkerja + "',"
                    + "lembur = '"+ lembur + "',"
                    + "WHERE Nip = '"+ nip + "' ";
        PreparedStatement p = (PreparedStatement) Db_Connection.getConnection().prepareStatement(sql);
        p.executeUpdate();

        getData();
        Reset();
        JOptionPane.showMessageDialog(null, "Update berhasil...");
    } catch (SQLException err){
        JOptionPane.showMessageDialog(null, err.getMessage());
    }
}
```

2.2.8 Membuat button delete

```
public void deleteData(){
    loadData();
    //pesan konfirmasi untuk menghapus data
    int pesan = JOptionPane.showConfirmDialog(null, "Anda yakin menghapus data"+ nip, "?", "Konfirmasi", JOptionPane.OK_CANCEL_OPTION);
    //jika user mengklik tombol OK maka proses delete dilakukan
    if (pesan == JOptionPane.OK_OPTION){
        try{
            Statement stat = (Statement) Db_Connection.getConnection().createStatement();
            String sql = "DELETE FROM gaji WHERE Nip = '"+ nip + "' ";
            PreparedStatement p = (PreparedStatement) Db_Connection.getConnection().prepareStatement(sql);
            p.executeUpdate();
            getData();
            Reset();
            JOptionPane.showMessageDialog(null, "Delete berhasil");
        } catch (SQLException err){
            JOptionPane.showMessageDialog(null, err.getMessage());
        }
    }
}
```

2.2.9 Membuat button keluar

```
}

private void btnKeluarActionPerformed(java.awt.event.ActionEvent evt){
    this.dispose();
}

}
```


BAB IV

PENUTUP

3.1 Kesimpulan

Dari perancangan dan implementasi aplikasi yang telah dijelaskan pada bab bab sebelumnya dapat disimpulkan bahwa aplikasi menghitung gaji kariawan merupakan suatu aplikasi yang dirancang untuk membantu para pengguna/ user untuk melakukan proses perhitungan ideal tanpa menggunakan perhitungan manual.

3.2 Saran

Dengan melakukan percobaan ke tiap tiap pengguna dapat diberi saran untuk aplikasi menghitung gaji kariawan bahwa aplikasi didalam penawaran interface yang masih sederhana dan aplikasi ini sudah berjalan sesuai kebutuhan, fungsi dan tujuan.

DAFTAR PUSTAKA

<https://kodingwae.blogspot.com/2016/06/aplikasi-penggajian-karyawan-crud.html?m=1>

<https://kelasprogrammer.com/contoh-program-java-menghitung-gaji-karyawan/>

BAB I

PENDAHULUAN

- A. Latar Belakang
- B. Rumusan Masalah

BAB II

PEMBAHASAN