

LAPORAN TUGAS PROYEK

“Aplikasi Perhitungan Gaji Karyawan”



Nama : Sitti Azyyati Tajuddin

Stambuk : 13020180012

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS MUSLIM INDONESIA

MAKASSAR

2019

BAB 1

PENDAHULUAN

Suatu pengolahan data karyawan dalam sebuah usaha bisa mendapatkan kendala. Seperti yang dapat dihadapi seperti yang ada di program ini. Karyawan yang menggunakan sistem konvensional karena adanya suatu aplikasi yang dapat membantu perusahaan, penggajiannya membutuhkan waktu yang dapat lama pada badan usaha.

Sistem yang dapat mempercepat dan mempermudah dalam proses perhitungan gaji, memberikan suatu data akurat untuk menghitung data penggajian karyawan dan dapat membuat laporan adalah Aplikasi Sistem Penggajian Karyawan pada suatu badan usaha.

Pada laporan ini akan menguji gaji berdasarkan akumulasi seperti tunjangan gaji pokok yang ditentukan.

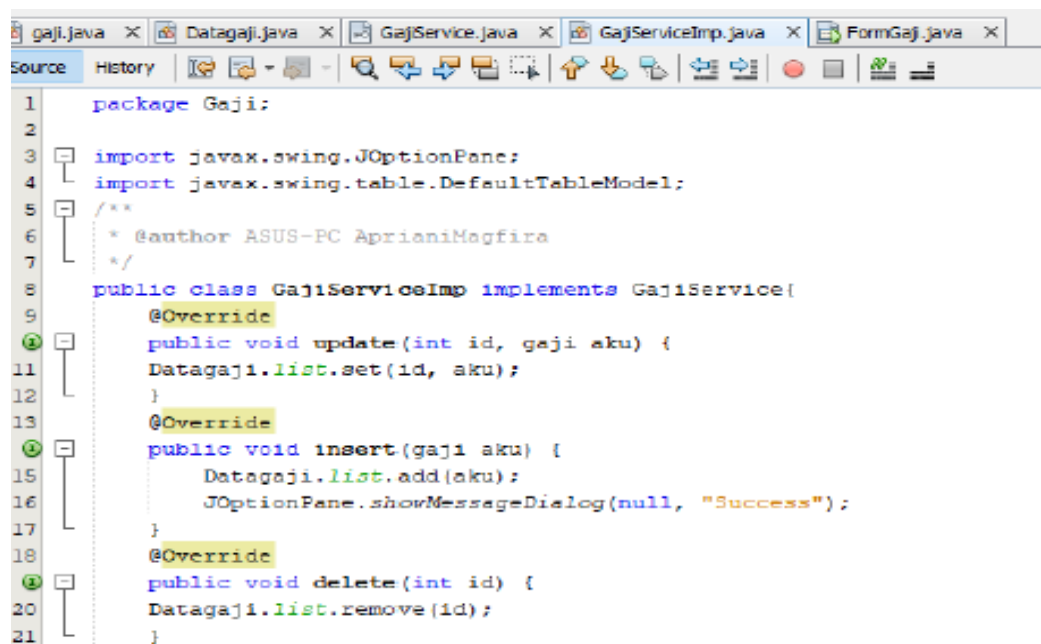
BAB 2

ANALISIS DAN DESAIN

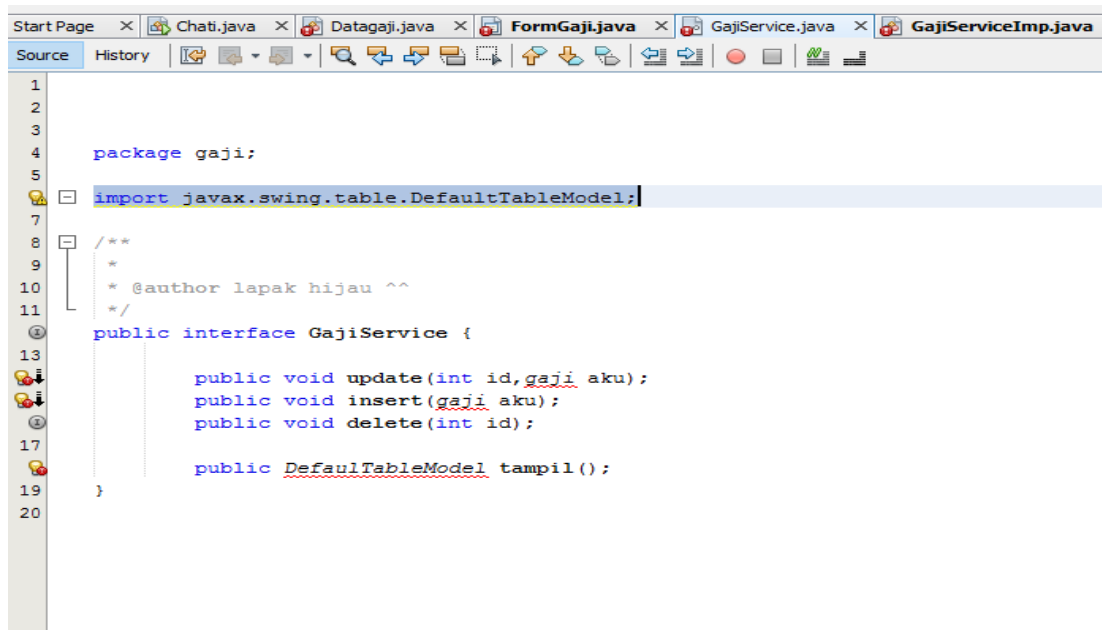
1. (Prosedur Kerja / Flowchart)
2. Konsep Pemrograman Berorientasi Object
 - a. Polimorphism

Polymorphism merupakan kemampuan suatu method untuk bekerja dengan lebih dari satu tipe argumen. Pada bahasa lain (khususnya C++), konsep ini sering disebut dengan method overloading. Pada dasarnya, Python tidak menangani hal ini secara khusus. Hal ini disebabkan karena Python merupakan suatu bahasa pemrograman yang bersifat dynamic typing yaitu tidak memerlukan deklarasi tipe.

Contohnya mengoverride method update() dengan menambahkan beberapa teks tambahan. Method update() pada class GajiServiceImp.java perilaku yang berbeda dengan class GajiServiceImp



```
1 package Gaji;
2
3 import javax.swing.JOptionPane;
4 import javax.swing.table.DefaultTableModel;
5
6 /**
7  * @author ASUS-PC AprianiMagfira
8  */
9 public class GajiServiceImp implements GajiService {
10     @Override
11     public void update(int id, gaji aku) {
12         Datagaji.list.set(id, aku);
13     }
14     @Override
15     public void insert(gaji aku) {
16         Datagaji.list.add(aku);
17         JOptionPane.showMessageDialog(null, "Success");
18     }
19     @Override
20     public void delete(int id) {
21         Datagaji.list.remove(id);
22     }
23 }
```

A screenshot of an IDE window with multiple tabs: Start Page, Chat1.java, Datagaji.java, FormGaji.java, GajiService.java, and GajiServiceImp.java. The 'GajiService.java' tab is active, showing the following code:

```
1  
2  
3  
4 package gaji;  
5  
6 import javax.swing.table.DefaultTableModel;  
7  
8 /**  
9  *  
10  * @author lapak hijau ^^  
11  */  
12 public interface GajiService {  
13  
14     public void update(int id, gaji aku);  
15     public void insert(gaji aku);  
16     public void delete(int id);  
17  
18     public DefaultTableModel tampil();  
19 }  
20
```

b. Inheritance (Pewarisan)

Inheritance atau Pewarisan/Penurunan adalah konsep pemrograman dimana sebuah class dapat ‘menurunkan’ property dan method yang dimilikinya kepada class lain. Konsep inheritance digunakan untuk memanfaatkan fitur ‘code reuse’ untuk menghindari duplikasi kode program.

Konsep inheritance membuat sebuah struktur atau ‘hierarchy’ class dalam kode program. Class yang akan ‘diturunkan’ bisa disebut sebagai class induk (parent class), super class, atau base class. Sedangkan class yang ‘menerima penurunan’ bisa disebut sebagai class anak (child class), sub class, derived class atau heir class.

c. Enkapsulasi / pengkapsulan (Encapsulation)

Enkapsulasi adalah pembungkus, maksud pembungkus disini adalah untuk menjaga suatu proses program agar tidak dapat diakses secara sembarangan atau diintervensi oleh program lain. Konsep enkapsulasi sangat penting dilakukan untuk menjaga kebutuhan program agar dapat diakses sewaktu-waktu, sekaligus menjaga program tersebut.

Begitulah konsep kerja dari enkapsulasi, dia akan melindungi sebuah program dari akses ataupun intervensi dari program lain yang mempengaruhinya. Hal ini sangat menjaga keutuhan program yang telah dibuat dengan konsep dan rencana yang sudah ditentukan dari awal.

Enkapsulasi menciptakan abstraksi untuk desain kelas. Jika Anda ingin melindungi beberapa anggota dari kelas dasar maka timbullah situasi

enkapsulasi. Dalam konsep ini ada tiga kata kunci yang digunakan. Konsep hak akses ini biasa di sebut Access Modifier.

BAB 3

HASIL IMPLEMENTASI PROGRAM

1. Luaran / Output Program
2. Penggalan Source Code Program Aplikasi

a. Class Gaji.java

13020180012_SittiAzyyatiTajuddin_TugasProyek - NetBeans IDE 8.2 RC

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

<default config>

Projects Files Services

13020180012_SittiAzyyatiTajuddin_TugasProyek

Source Packages

gaji

Datagaji.java

FormGaji.java

Gaji.java

GajiService.java

GajiServiceImp.java

pkg13020180012_sittiazyyatiTajuddin_tugasproyek

Test Packages

Libraries

JDK 1.8 (Default)

Test Libraries

chati

Source Packages

chati

setTot - Navigator

Members

getNama() : String

getNik() : String

getNohp() : String

getTot() : double

setAlamat(String alamat)

setGaji(int gaji)

setGaji(int gaji1)

setJabatan(String jabatan)

setJk(String jk)

setLembur(String lembur)

setName(String nama)

setNik(String nik)

setNohp(String nohp)

setTot(double tot)

Source History

```
6 package gaji;
7
8 /**
9  * @author lapak hijau ^^
10  */
11
12 public class Gaji {
13     private String nik,nama,jabatan,nohp,alamat,jk, lembur;
14     private int gaji,gaji1;
15     private double tot;
16     public String getNik(){
17         return nik;
18     }
19     public void setNik(String nik){
20         this.nik = nik;
21     }
22     public String getNama(){
23         return nama;
24     }
25     public void setNama(String nama){
26         this.nama = nama;
27     }
28     public String getJabatan(){
29         return jabatan;
30     }
31     public void setJabatan(String jabatan){
32         this.jabatan = jabatan;
33     }
34     public String getNohp(){
35         return nohp;
36     }
```

13020180012_SittiAzyyatiTajuddin_TugasProyek - NetBeans IDE 8.2 RC

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

<default config>

Projects Files Services

13020180012_SittiAzyyatiTajuddin_TugasProyek

Source Packages

gaji

Datagaji.java

FormGaji.java

Gaji.java

GajiService.java

GajiServiceImp.java

pkg13020180012_sittiazyyatiTajuddin_tugasproyek

Test Packages

Libraries

JDK 1.8 (Default)

Test Libraries

chati

Source Packages

chati

setTot - Navigator

Members

getNama() : String

getNik() : String

getNohp() : String

getTot() : double

setAlamat(String alamat)

setGaji(int gaji)

setGaji(int gaji1)

setJabatan(String jabatan)

setJk(String jk)

setLembur(String lembur)

setName(String nama)

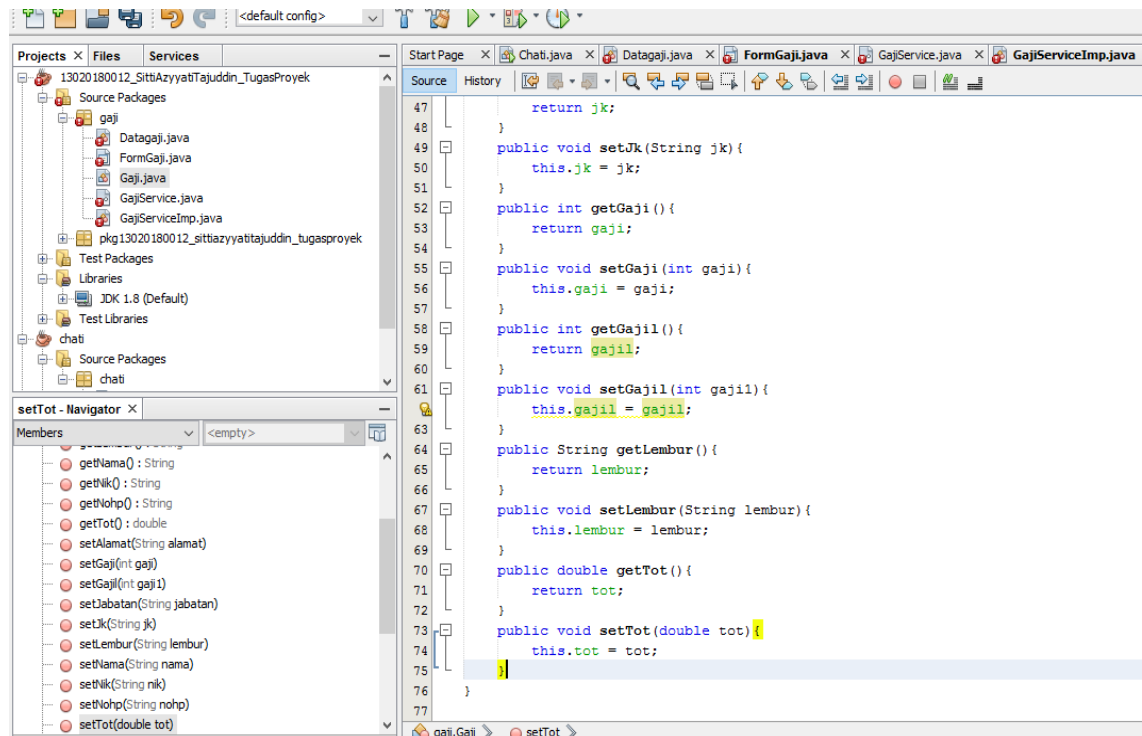
setNik(String nik)

setNohp(String nohp)

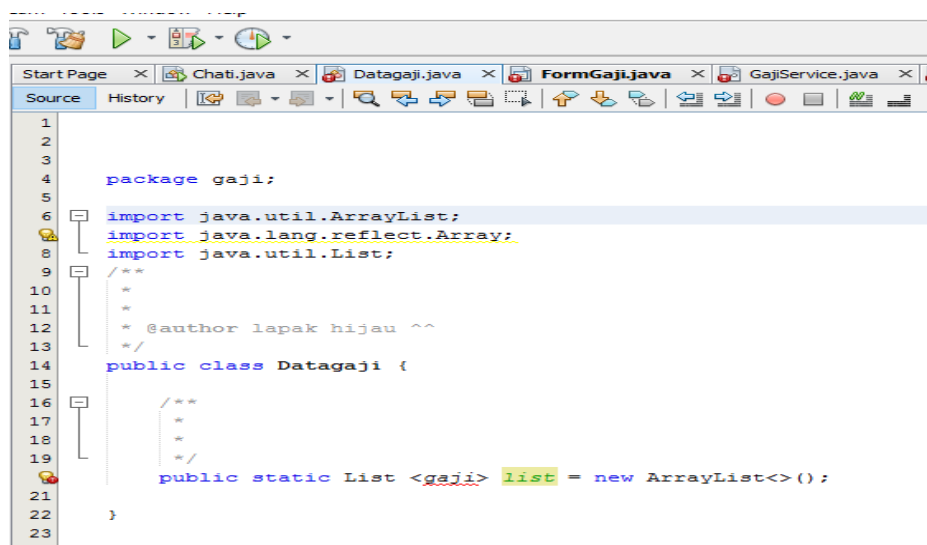
setTot(double tot)

Source History

```
34 public String getNohp(){
35     return nohp;
36 }
37 public void setNohp(String nohp){
38     this.nohp = nohp;
39 }
40 public String getAlamat(){
41     return alamat;
42 }
43 public void setAlamat(String alamat){
44     this.alamat = alamat;
45 }
46 public String getJk(){
47     return jk;
48 }
49 public void setJk(String jk){
50     this.jk = jk;
51 }
52 public int getGaji(){
53     return gaji;
54 }
55 public void setGaji(int gaji){
56     this.gaji = gaji;
57 }
58 public int getGaji1(){
59     return gaji1;
60 }
61 public void setGaji1(int gaji1){
62     this.gaji1 = gaji1;
63 }
64 public String getLembur(){
```



b. Class Datagaji.java



Untuk menyimpan data yang berbentuk array yang dikumpulkan pengguna

c. Interface GajiService.java

```

1
2
3
4 package gaji;
5
6 import javax.swing.table.DefaultTableModel;
7
8 /**
9  *
10  * @author lapak hijau ^^
11  */
12 public interface GajiService {
13
14     public void update(int id, gaji aku);
15     public void insert(gaji aku);
16     public void delete(int id);
17
18     public DefaultTableModel tampil();
19
20 }

```

d. Class GajiServiceImp.java

```

1 package Gaji;
2
3 import javax.swing.JOptionPane;
4 import javax.swing.table.DefaultTableModel;
5
6 /**
7  * @author ASUS-PC AprianiMagfira
8  */
9 public class GajiServiceImp implements GajiService{
10
11     @Override
12     public void update(int id, gaji aku) {
13         DataGaji.list.set(id, aku);
14     }
15
16     @Override
17     public void insert(gaji aku) {
18         DataGaji.list.add(aku);
19         JOptionPane.showMessageDialog(null, "Success");
20     }
21
22     @Override
23     public void delete(int id) {
24         DataGaji.list.remove(id);
25     }
26 }

```

e. Class FormGaji.java

1. Untuk menambahkan data


```

27
28
29     public void tambah() {
30         t = 0;
31         nik.setEnabled(true);
32         nama.setEnabled(true);
33         jk.setEnabled(true);
34         hp.setEnabled(true);
35         alamat.setEnabled(true);
36         jabatan.setEnabled(true);
37         gaji.setEnabled(true);
38         lembur.setEnabled(true);
39         gajil.setEnabled(true);
40         gajit.setEnabled(true);
41
42         nik.setText("");
43         nama.setText("");
44         hp.setText("");
45         alamat.setText("");
46         gaji.setText("");
47         gajil.setText("");
48         gajit.setText("");
49
50         hitung.setEnabled(true);
51         simpan.setEnabled(true);
52         hapus.setEnabled(true);
53         tambah.setEnabled(false);
54         ubah.setEnabled(false);
55     }

```

2. Untuk mengubah data

```

50         simpan.setEnabled(true);
51         hapus.setEnabled(true);
52         tambah.setEnabled(false);
53         ubah.setEnabled(false);
54     }
55
56     public void ubah() {
57         t = 1;
58         nik.setEnabled(true);
59         nama.setEnabled(true);
60         jk.setEnabled(true);
61         hp.setEnabled(true);
62         alamat.setEnabled(true);
63         jabatan.setEnabled(true);
64         gaji.setEnabled(true);
65         lembur.setEnabled(true);
66         gajil.setEnabled(true);
67         gajit.setEnabled(true);
68
69         hitung.setEnabled(true);
70         simpan.setEnabled(true);
71         hapus.setEnabled(false);
72         tambah.setEnabled(true);
73         ubah.setEnabled(false);
74         batal.setEnabled(true);
75     }
76
77

```

3. Untuk action batal

```

79 public void batal(){
80     nik.setEnabled(false);
81     nama.setEnabled(false);
82     jk.setEnabled(false);
83     hp.setEnabled(false);
84     alamat.setEnabled(false);
85     jabatan.setEnabled(false);
86     gaji.setEnabled(false);
87     lembur.setEnabled(false);
88     gajil.setEnabled(false);
89     gajit.setEnabled(false);
90
91     hitung.setEnabled(false);
92     simpan.setEnabled(false);
93     hapus.setEnabled(false);
94     tambah.setEnabled(true);
95     ubah.setEnabled(false);
96     batal.setEnabled(false);
97
98     nik.setText(" ");
99     nama.setText(" ");
100    hp.setText(" ");
101    alamat.setText(" ");
102    gaji.setText(" ");
103    gajit.setText(" ");
104    gajil.setText(" ");
105
106 }

```

4. Hitung dan hapus

```

107
108 public void hitung(){
109     int c = Integer.valueOf(gaji.getText());
110     int d = Integer.valueOf(gajil.getText());
111
112     b = c + d;
113     gajit.setText(String.valueOf(b));
114 }
115
116 public void hapus(){
117     int konf = JOptionPane.showConfirmDialog(rootPane, "are you sure want to delete this data?",
118     "delete confirm", JOptionPane.YES_NO_OPTION);
119     if (konf == JOptionPane.YES_OPTION){
120         index = tbl.getSelectedRow();
121         service.delete(index);
122         total -- ;
123         batal();
124         tampil();
125     }
126 }
127

```

5. Untuk menampilkan data pada table

```

127
128 public void tampil(){
129     tbl.setModel(service.tampil());
130     total = tbl.getRowCount();
131     txttotal.setText(String.valueOf(total));
132 }
133

```

6. Untuk menyimpan data

```
gaji.java x Datagaji.java x GajiService.java x GajiServiceImp.java x FormGaji.java x
Source Design History
127
128 public void tampil() {
129     tbl.setModel(service.tampil());
130     total = tbl.getRowCount();
131     txttotal.setText(String.valueOf(total));
132 }
133
134 public void simpan() {
135     gaji ga = new gaji();
136
137     ga.setNik(nik.getText());
138     ga.setNama(nama.getText());
139     ga.setNohp(hp.getText());
140     ga.setAlamat(alamat.getText());
141     ga.setJk(jk.getSelectedItem().toString());
142     ga.setJabatan(jabatan.getSelectedItem().toString());
143     ga.setGaji(Integer.valueOf(gaji.getText()));
144     ga.setLembur(lembur.getSelectedItem().toString());
145     ga.setGaji1(Integer.valueOf(gaji1.getText()));
146     ga.setTot(Double.valueOf(gaji1.getText()));
147
148     if (t == 0) {
149         service.insert(ga);
150         total++;
151     } else if (t == 1) {
152         index = tbl.getSelectedRow();
153         service.update(index, ga);
154     }
155     batal();
156     tampil();
157 }
```

7. Untuk set data kedalam tabel

```
gaji.java x Datagaji.java x GajiService.java x GajiServiceImp.java x FormGaji.java x
Source Design History
158
159 public void lokasi() {
160     int lo = tbl.getSelectedRowCount();
161     if (lo > 0) {
162         ubah.setEnabled(true);
163         hapus.setEnabled(true);
164         index = tbl.getSelectedRow();
165
166         nik.setText(tbl.getValueAt(index, 0).toString());
167
168         nama.setText(tbl.getValueAt(index, 1).toString());
169
170         jk.setSelectedItem(tbl.getValueAt(index, 2).toString());
171
172         jabatan.setSelectedItem(tbl.getValueAt(index, 3).toString());
173
174         hp.setText(tbl.getValueAt(index, 4).toString());
175
176         alamat.setText(tbl.getValueAt(index, 5).toString());
177
178         gaji.setText(tbl.getValueAt(index, 6).toString());
179
180         lembur.setSelectedItem(tbl.getValueAt(index, 7).toString());
181
182         gaji1.setText(tbl.getValueAt(index, 8).toString());
183
184         gaji1t.setText(tbl.getValueAt(index, 9).toString());
185     }
186 }
187 }
```

8. Method constructor untuk button (cancel, change, add, etc)

```

gaji.java x Datagaji.java x GajiService.java x GajiServiceImp.java x FormGaji.java x
Source Design History
194 /**
195  * This method is called from within the constructor to initialize the form.
196  * WARNING: Do NOT modify this code. The content of this method is always
197  * regenerated by the Form Editor.
198  */
199 @SuppressWarnings("unchecked")
200 Generated Code
585
586 private void batalActionPerformed(java.awt.event.ActionEvent evt) {
587     batal();
588 }
589
590 private void ubahActionPerformed(java.awt.event.ActionEvent evt) {
591     ubah();
592 }
593
594 private void tambahActionPerformed(java.awt.event.ActionEvent evt) {
595     tambah();
596 }
597
598 private void jkActionPerformed(java.awt.event.ActionEvent evt) {
599     if (jk.getSelectedItem().equals("PILIH")) {
600
601     } else if (jk.getSelectedItem().equals("Laki-Laki")) {
602
603     } else if (jk.getSelectedItem().equals("Perempuan")) {
604     } else {
605     }
606 }
607
608

```

```

gaji.java x Datagaji.java x GajiService.java x GajiServiceImp.java x FormGaji.java x
Source Design History
606     }
607 }
608
609 private void hapusActionPerformed(java.awt.event.ActionEvent evt) {
610     hapus();
611 }
612
613 private void simpanActionPerformed(java.awt.event.ActionEvent evt) {
614     simpan();
615 }
616
617 private void hitungActionPerformed(java.awt.event.ActionEvent evt) {
618     hitung();
619 }
620

```

9. Action perform untuk memilih item

```

620
621 private void lemburActionPerformed(java.awt.event.ActionEvent evt) {
622     if (lembur.getSelectedItem().equals("PILIH")) {
623
624     } else if (lembur.getSelectedItem().equals("gol 1")) {
625         gaji1.setText("5000");
626     } else if (lembur.getSelectedItem().equals("gol 2")) {
627         gaji1.setText("3000");
628     } else if (lembur.getSelectedItem().equals("gol 3")) {
629         gaji1.setText("2000");
630     } else if (lembur.getSelectedItem().equals("gol 4")) {
631         gaji1.setText("1500");
632     } else if (lembur.getSelectedItem().equals("gol 5")) {
633         gaji1.setText("1000");
634     }
635     //else {}
636 }

```

```

638 private void jabatanActionPerformed(java.awt.event.ActionEvent evt) {
639     if (jabatan.getSelectedItem().equals("PILIH")){
640
641     } else if (jabatan.getSelectedItem().equals("I")){
642         gaji.setText("500000");
643     } else if (jabatan.getSelectedItem().equals("II")){
644         gaji.setText("300000");
645     } else if (jabatan.getSelectedItem().equals("III")){
646         gaji.setText("250000");
647     } else if (jabatan.getSelectedItem().equals("IV")){
648         gaji.setText("100000");
649     } else if (jabatan.getSelectedItem().equals("IV")){
650         gaji.setText("50000");
651     }
652     // else {}
653 }

```

10. Main method

```

gaji.java x Datagaji.java x GajiService.java x GajiServiceImp.java x FormGaji.java x
Source Design History
670
671 private void tblMouseClicked(java.awt.event.MouseEvent evt) {
672     lokasi();
673 }
674
675
676
677 /**
678  * @param args the command line arguments
679  */
680 public static void main(String args[]) {
681     /* Set the Nimbus look and feel */
682     Look and feel setting code (optional)
683
684     /* Create and display the form */
685     java.awt.EventQueue.invokeLater(new Runnable() {
686         public void run() {
687             new FormGaji().setVisible(true);
688         }
689     });
690 }
691

```

```

// Variables declaration - do not modify
private javax.swing.JTextArea alamat;
private javax.swing.JButton batal;
private javax.swing.JTextField gaji;
private javax.swing.JTextField gaji1;
private javax.swing.JTextField gaji2;
private javax.swing.JButton hapus;
private javax.swing.JButton hitung;
private javax.swing.JTextField hp;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel12;
private javax.swing.JLabel jLabel13;
private javax.swing.JLabel jLabel14;
private javax.swing.JLabel jLabel15;
private javax.swing.JLabel jLabel16;
private javax.swing.JLabel jLabel17;
private javax.swing.JLabel jLabel18;
private javax.swing.JLabel jLabel19;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JScrollPane jScrollPane3;
private javax.swing.JComboBox<String> jabatan;
private javax.swing.JComboBox<String> ik;
private javax.swing.JComboBox<String> lembur;

```

BAB 4

PENUTUP

1. Kesimpulan

Efektifitas dan akurasi pengolahan gaji diharapkan bisa efektif dan aplikasi dibandingkan dengan sistem konvensional

2. Saran

Saya harap aplikasi bisa dapat diaplikasikan oleh badan-badan usaha