

**TUGAS BESAR MATA KULIAH PEMROGRAMAN
BERORIENTASI OBJEK**

SocialLife

Oleh :

Diva Annisa Febecca - 1301204302
Johannes Raphael Nandaputra - 1301204243
Muhammad Mufid Utomo - 1301204441
Muhammad Naufal Abdillah - 1301201586
Ricardo Hamonangan - 1301204201

Kelompok 1
IF-44-01



PROGRAM STUDI INFORMATIKA

FAKULTAS INFORMATIKA

2023

Daftar Isi

Daftar Isi	1
BAB 1	3
PENDAHULUAN	3
1.1. Deskripsi Tugas Besar	3
1.2. Tools yang dipakai	3
BAB 2	4
PERANCANGAN	4
2.1. Fitur-fitur	4
2.1.1 Menambahkan teman	4
2.1.2 Menghapus teman	4
2.1.3 Mengupdate data	4
2.1.4 Mengirim pesan	4
2.1.5 Menerima pesan	4
2.2. Class Diagram	5
2.3. Rancangan Basis Data	6
2.4. Rancangan Antarmuka (UI)	7
2.4.1. Landing Page	7
2.4.2. Login	7
2.4.3. Register	8
2.4.4. Chat	8
2.4.4.1. List Chat	8
2.4.4.1.1. Add Friend	9
2.4.4.2. Halaman Chat	9
2.4.5. Customization	10
BAB 3	11
IMPLEMENTASI	11
3.1. Kode Program	11
3.1.1 main.java	11
3.1.2. Model	12
3.1.2.1. Chat Model	12
3.1.2.2. Message Model	13
3.1.2.3. User Model	14
3.1.3. Controller	15
3.1.3.1. Message Controller	15

3.1.3.2. User Controller	16
3.1.4. View	19
3.1.4.1. AddFriendDialog.java	19
3.1.4.2. chatList.java	20
3.1.4.3. chatWindow.java	24
3.1.4.4. CustomizationDialog.java	27
3.1.4.5. LandingView.java	30
3.1.4.6. LoginView.java	33
3.1.4.7. RegisterView.java	37
3.2. Screenshot Aplikasi	41
3.2.1. Landing page	41
3.2.2. Register	41
3.2.3. Login	42
3.2.4. Chat List	43
3.2.5. Add Friend	44
3.2.6. Delete Friend	45
3.2.7. Chat Window	46
3.2.8. Customization	49
3.2.9. Logout	51
Daftar Pustaka	51

BAB 1

PENDAHULUAN

1.1. Deskripsi Tugas Besar

Topik Tugas Besar Pemrograman Berorientasi Objek (PBO) yang kami pilih adalah mengenai aplikasi *Chatting*. Dimana aplikasi *chatting* ini kami beri nama dengan SociaLife. SociaLife ini merupakan aplikasi *chatting* yang dimana diimplementasikan menggunakan bahasa pemrograman Java.

1.2. Tools yang dipakai

Dalam pengerjaan Tugas Besar Pemrograman Berorientasi Objek ini, ada beberapa tools yang digunakan untuk pembuatan aplikasi, baik tools yang secara langsung digunakan untuk membuat aplikasinya, maupun tools - tools yang membantu dalam pembuatan laporan dan perancangan. Berikut merupakan list dari tools - tools yang digunakan.

- Netbeans : IDE yang digunakan selama pembuatan aplikasi.
- XAMPP : Alat bantuan untuk pembuatan database dalam localhost.
- Diagrams.io : Alat bantu untuk pembuatan class diagram pada saat perancangan.

BAB 2

PERANCANGAN

2.1. Fitur-fitur

2.1.1 Menambahkan teman

Didalam fitur ini kita dapat menambahkan teman kita dan akan masuk kedalam list teman dengan catatan teman kita tersebut sudah ada data didalam databasenya jika tidak ada maka akan ada peringatan bahwa tidak ada nama teman kita di databasenya.

2.1.2 Menghapus teman

Didalam fitur ini kita dapat menghapus teman kita dari list teman kita.

2.1.3 Mengupdate data

Didalam fitur ini kita dapat mengubah username dan password kita sehingga nama username kita berubah dan juga data kita akan di perbaharui didalam database yang kita buat.

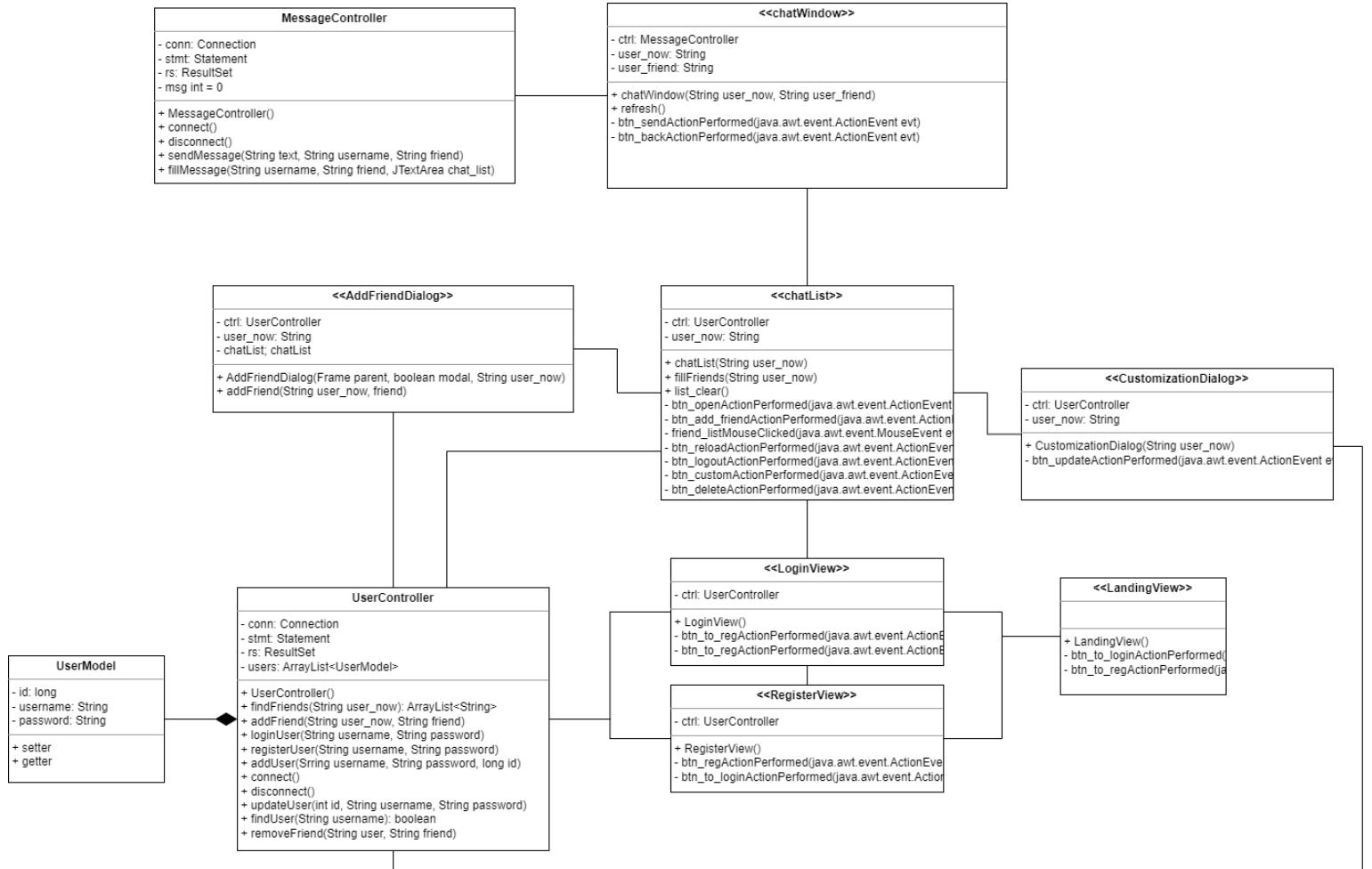
2.1.4 Mengirim pesan

Didalam fitur ini kita dapat mengirim pesan terhadap teman kita yang sudah kita tambahkan tadi.

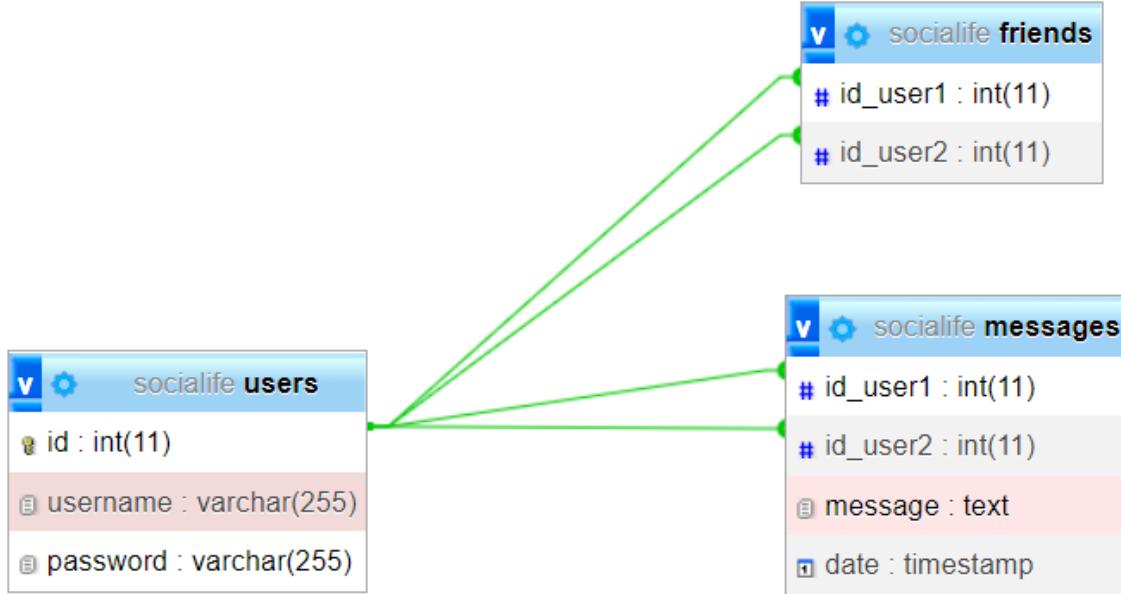
2.1.5 Menerima pesan

Didalam fitur ini kita dapat menerima pesan dari teman kita yang telah mengirimkan pesan kepada kita.

2.2. Class Diagram



2.3. Rancangan Basis Data

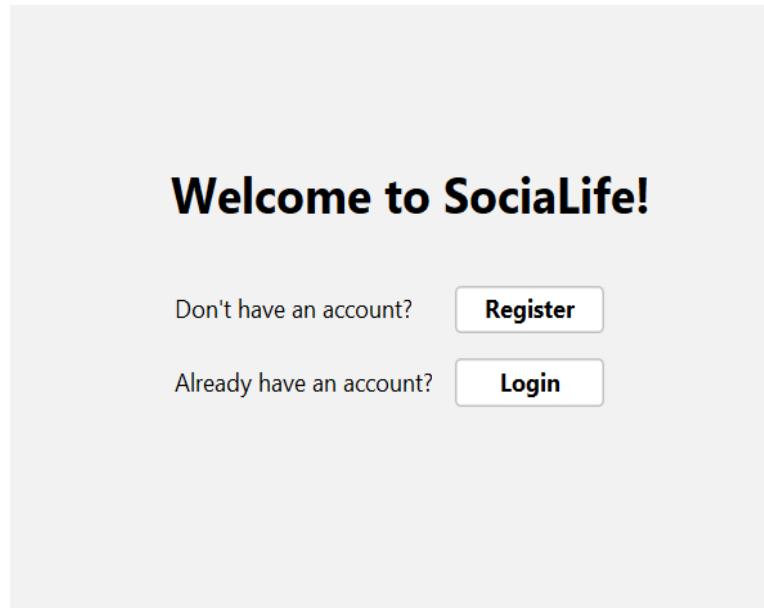


Pada Basis data, terdapat 3 table untuk menyimpan data users, message, dan data friends.

- Table users akan menyimpan ID, dimana sifatnya auto increment. ID ini akan menjadi basis untuk memanggil data - data yang ada di database. Table users juga akan menyimpan data - data username dan password.
- Table messages akan menyimpan `id_user1`(sender) dan `id_user2`(receiver), serta message yang berisi text dari pesan, dan date(kapan pesan dikirim). Messages merupakan table yang akan menyimpan semua data - data message yang dikirim oleh semua user, program akan memanggil message dari table messages.
- Table friends menyimpan data user apa saja yang sudah meng add user lainnya. Jika `id_user1` mengadd `id_user2`, belum tentu `id_user2` telah mengadd `id_user1`.

2.4. Rancangan Antarmuka (UI)

2.4.1. Landing Page



2.4.2. Login

The screenshot shows the login page for the "SocialLife!" application. The page has a light gray background. At the top center, the text "LOGIN PAGE" is displayed in a large, bold, black font. Below this, there are two input fields: one labeled "Name" and another labeled "Password", each accompanied by a rectangular input box. Below these fields is a large rectangular button containing the word "Login" in bold black capital letters. At the bottom of the page, there is a link "Don't have an account?" followed by a rectangular button with the word "Register" in bold black capital letters.

2.4.3. Register

REGISTER PAGE

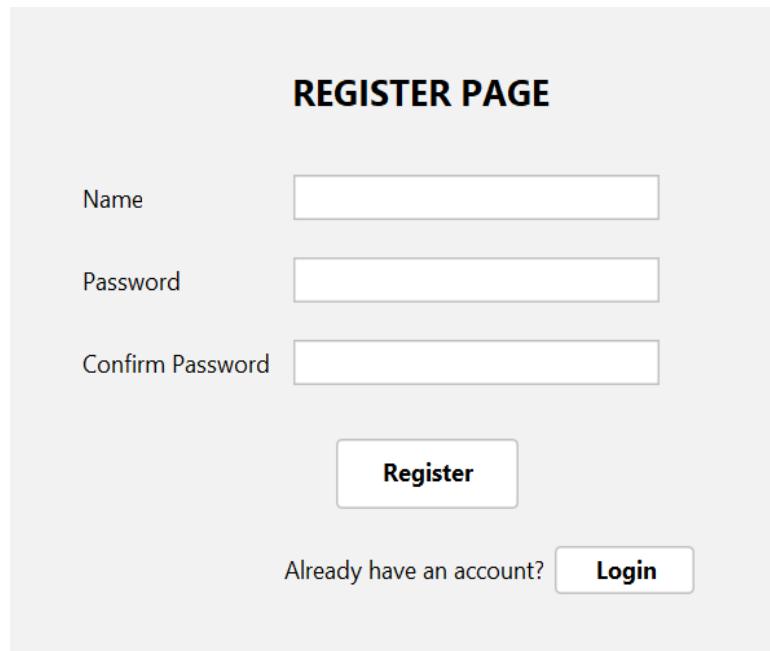
Name

Password

Confirm Password

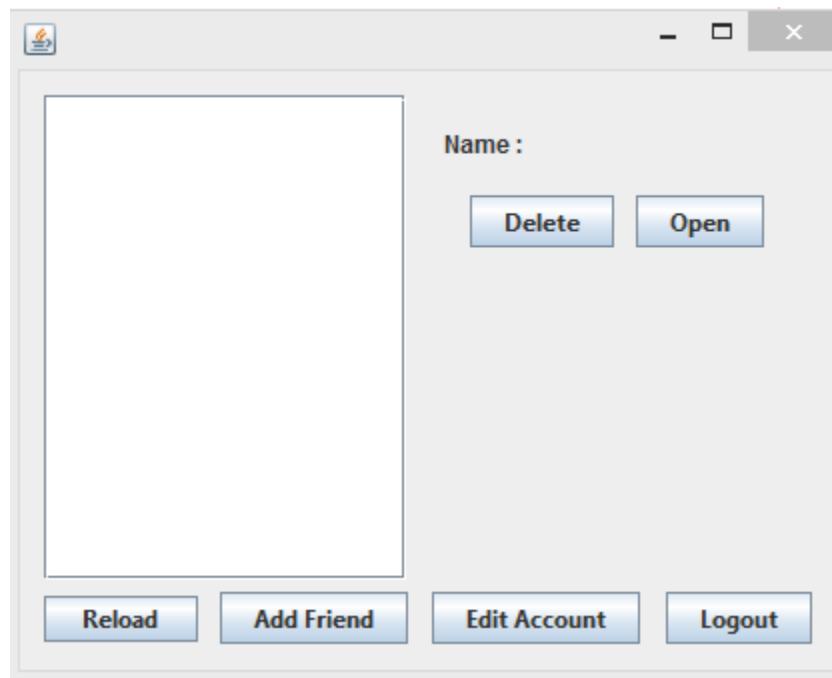
Register

Already have an account? [Login](#)



2.4.4. Chat

2.4.4.1. List Chat

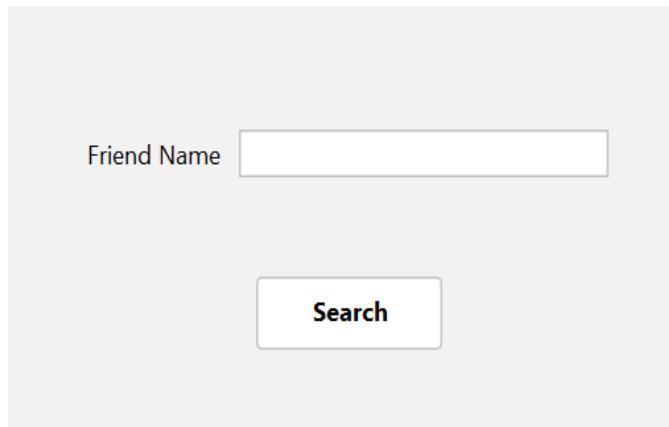


Name :

Delete **Open**

Reload **Add Friend** **Edit Account** **Logout**

2.4.4.1.1. Add Friend

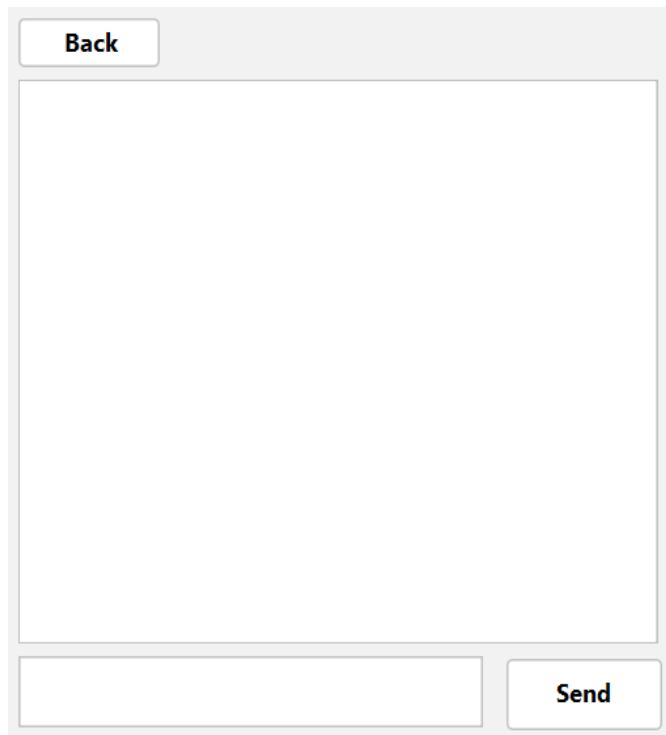


A user interface for adding a friend. It features a text input field labeled "Friend Name" and a button labeled "Search".

Friend Name

Search

2.4.4.2. Halaman Chat



A user interface for a chat application. It includes a "Back" button at the top left, a large central area for messages, and a bottom row with a text input field and a "Send" button.

Back

Send

2.4.5. Customization

Customization

Name

Password

Confirm Password

BAB 3

IMPLEMENTASI

3.1. Kode Program

3.1.1 main.java

```
● ● ●
1 import Controllers.UserController;
2 import Controllers.ChatController;
3 import Controllers.MessageController;
4 import Models.UserModel;
5 import Models.ChatModel;
6 import Models.MessageModel;
7 import java.util.Scanner;
8 import java.awt.image.BufferedImage;
9 import java.awt.Graphics;
10 import java.awt.Graphics2D;
11 import java.awt.RenderingHints;
12 import java.awt.Font;
13 import javax.imageio.ImageIO;
14 import java.io.File;
15 import java.io.IOException;
16 import java.lang.Thread;
17
18 import Views.landingView;
19
20 /**
21 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
22 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
23 */
24
25 /**
26 *
27 * @author johan
28 */
29 public class Main {
30
31
32     public static void main(String[] args) throws IOException, InterruptedException {
33         landingView main = new landingView();
34         main.show();
35
36     }
37 }
38 }
```

3.1.2. Model

3.1.2.1. Chat Model

```
● ● ●
1 package Models;
2
3 /**
4 *
5 * @author johan
6 */
7 public class ChatModel {
8     private long id;
9     private long member1;
10    private long member2;
11
12    public ChatModel(long chat_id, long member1, long member2) {
13        this.id = id;
14        this.member1 = member1;
15        this.member2 = member2;
16    }
17
18    public ChatModel(long member1, long member2){
19        id = Math.round(Math.random());
20        this.member1 = member1;
21        this.member2 = member2;
22    }
23
24    public double getId() {
25        return id;
26    }
27
28    public void setId(long id) {
29        this.id = id;
30    }
31
32    public long getMember1() {
33        return member1;
34    }
35
36    public void setMember1(long member1) {
37        this.member1 = member1;
38    }
39
40    public long getMember2() {
41        return member1;
42    }
43
44    public void setMember2(long member2) {
45        this.member2 = member2;
46    }
47 }
```

3.1.2.2. Message Model

```
● ○ ●
1 package Models;
2 import java.util.Date;
3
4 /**
5  *
6  * @author johan
7  */
8 public class MessageModel {
9     private long id_user1;
10    private long id_user2;
11    private String message;
12    private Date date;
13
14    public MessageModel(long id_user1, long id_user2, String message, Date date) {
15        this.id_user1 = id_user1;
16        this.id_user2 = id_user2;
17        this.message = message;
18        this.date = date;
19    }
20
21    public long getId_user1() {
22        return id_user1;
23    }
24
25    public void setId_user1(long id_user1) {
26        this.id_user1 = id_user1;
27    }
28
29    public long getId_user2() {
30        return id_user2;
31    }
32
33    public void setId_user2(long id_user2) {
34        this.id_user2 = id_user2;
35    }
36
37    public String getMessage() {
38        return message;
39    }
40
41    public void setMessage(String message) {
42        this.message = message;
43    }
44
45    public Date getDate() {
46        return date;
47    }
48
49    public void setDate(Date date) {
50        this.date = date;
51    }
52 }
```

3.1.2.3. User Model

```
● ● ●
1 package Models;
2 import java.lang.Math;
3 import java.util.*;
4 /**
5 *
6 * @author johan
7 */
8 public class UserModel {
9     private long id;
10    private String username;
11    private String password;
12
13    public UserModel(String username, String password, long id) {
14        this.id = id;
15        this.username = username;
16        this.password = password;
17    }
18
19    public void setUsername(String username) {
20        this.username = username;
21    }
22
23    public void setPassword(String password) {
24        this.password = password;
25    }
26
27    public String getUsername() {
28        return username;
29    }
30
31    public String getPassword() {
32        return password;
33    }
34
35    public long getId() {
36        return id;
37    }
38
39    public void setId(long id) {
40        this.id = id;
41    }
42 }
```

3.1.3. Controller

3.1.3.1. Message Controller

```
● ● ●
1 package controllers;
2
3 import Models.MessageModel;
4 import Models.UserModel;
5 import java.lang.Math;
6 import java.sql.*;
7 import java.util.ArrayList;
8 import javax.swing.JTextArea;
9 /**
10 */
11 * @author jahan
12 */
13 public class MessageController {
14     private Connection conn;
15     private Statement stmt;
16     private ResultSet rs;
17     private int msg = 0;
18
19     public MessageController() {
20         try{
21             connect();
22             String sql = "SELECT * FROM users";
23             rs = stmt.executeQuery(sql);
24             disconnect();
25         } catch(Exception e){
26             e.printStackTrace();
27         }
28     }
29
30     public void connect(){
31         try {
32             this.conn = DriverManager.getConnection(
33                 "jdbc:mysql://localhost:3306/socialife",
34                 "root",
35                 "");
36             this.stmt = conn.createStatement();
37         } catch (Exception e) {
38             e.printStackTrace();
39         }
40     }
41
42     public void disconnect(){
43         try{
44             conn.close();
45             stmt.close();
46         } catch(Exception e){
47             e.printStackTrace();
48         }
49     }
50
51     public void sendMessage(String text, String username, String usernameFriend) {
52         System.out.println("send message " + username + usernameFriend);
53         String sql = "INSERT INTO messages (`id_user1`, `id_user2`, `message`, `date`) VALUES ((SELECT id FROM users WHERE username='" + username + "'"), (SELECT id FROM users WHERE username='" + usernameFriend + "'"), '" + text + "', CURRENT_TIMESTAMP)";
54
55         try {
56             connect();
57             stmt.executeUpdate(sql);
58             disconnect();
59         } catch (SQLException e) {
60             e.printStackTrace();
61         }
62         msg = -5;
63     }
64
65     public void fillMessage(String username, String usernameFriend, JTextArea chat_window) {
66         //chat_window.setText(null);
67         String sql = "SELECT users.username, messages.date, messages.message FROM messages INNER JOIN users ON users.id = messages.id_user1 WHERE (messages.id_user1 = (SELECT id FROM users WHERE username = '" +
68             + username +
69             + "') AND messages.id_user2 = (SELECT id FROM users WHERE username = '" +
70             + usernameFriend +
71             + "') ) OR (messages.id_user1 = (SELECT id FROM users WHERE username = '" +
72             + usernameFriend +
73             + "') AND messages.id_user2 = (SELECT id FROM users WHERE username = '" +
74             + username +
75             + "') ) ORDER BY MESSAGES.DATE ASC";
76
77         try {
78             connect();
79             rs = stmt.executeQuery(sql);
80             rs.next();
81             rs.last();
82             if (msg < rs.getRow()){
83                 rs.first();
84                 do {
85                     chat_window.append(rs.getObject(1).toString() + " " + rs.getObject(2).toString() + ": \n" + rs.getObject(3).toString() + "\n \n");
86                 } while (rs.next());
87                 rs.last();
88                 msg = rs.getRow();
89             }
90             disconnect();
91         } catch (SQLException e) {
92             e.printStackTrace();
93         }
94     }
95 }
```

3.1.3.2. User Controller

```
1 package Controllers;
2 import java.util.Scanner;
3 import Models.UserModel;
4 import Views.chatList;
5 import java.sql.*;
6 import java.util.*;
7 import javax.swing.JOptionPane;
8 /**
9  * @author johan
10 */
11 public class UserController {
12     private Connection conn;
13     private Statement stmt;
14     private ResultSet rs;
15     private ArrayList<UserModel> users;
16
17     public UserController() {
18         users = new ArrayList<>();
19         try{
20             connect();
21             String sql = "SELECT * FROM users";
22             rs = stmt.executeQuery(sql);
23             while(rs.next()){
24                 addUser(rs.getString("username"), rs.getString("password"),
25                         rs.getInt("id"));
26             }
27             disconnect();
28         } catch(Exception e){
29             e.printStackTrace();
30         }
31     }
32 }
33
34 public ArrayList<String> findFriends(String user_now){
35     ArrayList<String> friends = new ArrayList<>();
36     try{
37         connect();
38         String sql = "SELECT users.username FROM friends INNER JOIN users ON friends.id_user2=users.id WHERE friends.id_user1 = (SELECT id FROM users WHERE username = '" + user_now + "')";
39         rs = stmt.executeQuery(sql);
40         while(rs.next()){
41             friends.add(rs.getString("username"));
42         }
43         disconnect();
44     } catch(Exception e){
45         e.printStackTrace();
46     }
47
48     return friends;
49 }
```

```

1  public void addFriend(String user_now, String friend){
2      boolean friendSwitch = false;
3      String sql;
4
5      //cek friend exists
6      try{
7          connect();
8          sql = "SELECT * FROM users WHERE username='"+friend+"'";
9          rs = stmt.executeQuery(sql);
10         rs.next();
11         if(friend.equals(rs.getObject(2).toString())) friendSwitch = true; System.out.println("friendSwitch TRUE!");
12         disconnect();
13     } catch (Exception e){
14         JOptionPane.showMessageDialog(null, "No such user in database!");
15     }
16
17     //cek if already friends
18     try{
19         sql = "SELECT * FROM friends WHERE id_user1 = (SELECT id FROM users WHERE username='"+ user_now +"') AND id_user2 = (SELECT id FROM users WHERE username='"+ friend +"')";
20         connect();
21         rs = stmt.executeQuery(sql);
22         int i = 0;
23         while(rs.next()) i++;
24         disconnect();
25         if(i>0){
26             friendSwitch = false;
27         }
28
29         if(friendSwitch){
30             try{
31                 sql = "INSERT INTO friends VALUES ((SELECT id FROM users WHERE username='"+ user_now +"'), (SELECT id FROM users WHERE username='"+ friend +"'))";
32                 connect();
33                 stmt.executeUpdate(sql);
34                 disconnect();
35                 JOptionPane.showMessageDialog(null, "friend added succesfully!!!");
36             } catch(Exception e){
37                 JOptionPane.showMessageDialog(null, "Error adding friend!");
38             }
39         }
40
41     } catch(Exception e){
42         e.printStackTrace();
43     }
44 }
45
46 public void loginUser(String username, String password) {
47     try {
48         connect();
49         String sql = "SELECT * FROM users where username='"+username+"' and password='"+password+"'";
50         rs = stmt.executeQuery(sql);
51         if (rs.next()) {
52             chatlist landview = new chatlist(username);
53             landview.show();
54         } else {
55             JOptionPane.showMessageDialog(null, "username atau password salah");
56         }
57         disconnect();
58     } catch (SQLException e) {
59         e.printStackTrace();
60     }
61 }
62
63 public void registerUser(String username, String password) {
64     UserController x = new UserController();
65     try {
66         connect();
67         if(x.findUser(username) == false){
68             stmt.executeUpdate("INSERT INTO users (username, password) VALUES('"+username+"', '"+password+"');");
69         } else {
70             //placeholder. change later
71             System.out.println("UDAH ADA USERNAME TERSIMPAN");
72         }
73         disconnect();
74     } catch (SQLException e){
75         e.printStackTrace();
76     }
77 }

```

```
1 public void addUser(String username, String password, long id){
2     UserModel x = new UserModel(username, password, id);
3     users.add(x);
4 }
5
6 public void connect(){
7     try {
8         this.conn = DriverManager.getConnection(
9             "jdbc:mysql://localhost:3306/socialife",
10            "root",
11            "");
12         this.stmt = conn.createStatement();
13     } catch (Exception e) {
14         e.printStackTrace();
15     }
16 }
17
18 public void disconnect(){
19     try{
20         conn.close();
21         stmt.close();
22     } catch(Exception e){
23         e.printStackTrace();
24     }
25 }
26
27 public void updateUser(int id, String username, String password) {
28     try {
29         connect();
30         String sql = "UPDATE users SET username='"+username+"', password='"+password+"' WHERE id='"+id+"'";
31         stmt.executeUpdate(sql);
32         disconnect();
33     } catch (SQLException e){
34         e.printStackTrace();
35     }
36 }
37
38 public boolean findUser(String username){
39
40     boolean cek = false;
41     try{
42         connect();
43         rs = stmt.executeQuery("SELECT * FROM users WHERE username='"+username+"'");
44         while(rs.next()){
45             cek = true;
46         }
47         disconnect();
48     } catch (Exception e){
49         e.printStackTrace();
50     }
51
52
53     return cek;
54 }
```

3.1.4. View

3.1.4.1. AddFriendDialog.java

```
1 package Views;
2
3 import Controllers.UserController;
4
5 /**
6 * 
7 * @author naufalabdillah
8 */
9 public class AddFriendDialog extends javax.swing.JDialog {
10     UserController ctr = new UserController();
11     chatList chatlist;
12     String user_now;
13
14     /**
15      * Creates new form AddFriendDialog
16     */
17     public AddFriendDialog(java.awt.Frame parent, boolean modal, String user_now) {
18         super(parent, modal);
19         initComponents();
20         this.user_now = user_now;
21     }
22
23     /**
24      * This method is called from within the constructor to initialize the form.
25      * WARNING: Do NOT modify this code. The content of this method is always
26      * regenerated by the Form Editor.
27      */
28     @SuppressWarnings("unchecked")
29 // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
30     private void initComponents() {
31
32         jLabel1 = new javax.swing.JLabel();
33         input_friend = new javax.swing.JTextField();
34         btn_friend = new javax.swing.JButton();
35
36         setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
37
38         jLabel1.setText("Friend name: ");
39
40         btn_friend.setText("Search");
41         btn_friend.addActionListener(new java.awt.event.ActionListener() {
42             public void actionPerformed(java.awt.event.ActionEvent evt) {
43                 btn_friendActionPerformed(evt);
44             }
45         });
46
47         javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
48         getContentPane().setLayout(layout);
49         layout.setHorizontalGroup(
50             layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
51                 .addGroup(layout.createSequentialGroup()
52                     .addGap(60, 60, 60)
53                     .addComponent(jLabel1)
54                     .addGap(18, 18, 18)
55                     .addComponent(input_friend, javax.swing.GroupLayout.PREFERRED_SIZE, 154, javax.swing.GroupLayout.PREFERRED_SIZE)
56                     .addGap(10, 10, 10)
57                     .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
58                         .addGroup(layout.createSequentialGroup()
59                             .addGap(40, 40, 40)
60                             .addComponent(btn_friend)
61                             .addGap(125, 125, 125))
62                         .addGroup(layout.createSequentialGroup()
63                             .addGap(58, 58, 58)
64                             .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
65                                 .addComponent(jLabel1)
66                                 .addComponent(input_friend, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
67                                 .addComponent(btn_friend)
68                                 .addGap(40, 40, 40)
69                                 .addComponent(btn_friend)
70                                 .addGap(55, 55, 55)
71                             )
72                         )
73                     )
74                 );
75
76         pack();
77     }// </editor-fold>//GEN-END:initComponents
78
79     private void btn_friendActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_friendActionPerformed
80         // TODO add your handling code here:
81         String friend = input_friend.getText();
82         ctr.addFriend(this.user_now, friend);
83         chatList.fillFriends(user_now);
84     }//GEN-LAST:event_btn_friendActionPerformed
85
86     /**
87      * @param args the command line arguments
88     */
89
90     // Variables declaration - do not modify//GEN-BEGIN:variables
91     private javax.swing.JButton btn_friend;
92     private javax.swing.JTextField input_friend;
93     private javax.swing.JLabel jLabel1;
94     // End of Variables declaration//GEN-END:variables
95 }
```

3.1.4.2. chatList.java

```
1 package Views;
2
3 import Controllers.UserController;
4 import java.util.ArrayList;
5 import javax.swing.DefaultListModel;
6
7 /**
8 *
9 * @author naufalabdillah
10 */
11 public class chatList extends javax.swing.JFrame {
12     UserController ctr = new UserController();
13     String user_now;
14     /**
15      * Creates new form chatList
16     */
17
18     public void fillFriends(String user_now){
19         this.user_now = user_now;
20         DefaultListModel listModel = new DefaultListModel();
21         ArrayList<String> friends = new ArrayList<>();
22         friends = ctr.findFriends(user_now);
23         for(String f : friends){
24             listModel.addElement(f);
25         }
26         friend_list.setModel(listModel);
27     }
28
29     public void list_clear(){
30         DefaultListModel listModel = new DefaultListModel();
31         listModel.clear();
32         friend_list.setModel(listModel);
33     }
34
35     public chatList(String user_now) {
36         initComponents();
37         fillFriends(user_now);
38     }
39
40     /**
41      * This method is called from within the constructor to initialize the form.
42      * WARNING: Do NOT modify this code. The content of this method is always
43      * regenerated by the Form Editor.
44      */
45     @SuppressWarnings("unchecked")
46     // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
47 }
```

```
● ● ●  
1 private void initComponents() {  
2  
3     jScrollPane1 = new javax.swing.JScrollPane();  
4     friend_list = new javax.swing.JList();  
5     jLabel1 = new javax.swing.JLabel();  
6     jLabel2 = new javax.swing.JLabel();  
7     name_disp = new javax.swing.JLabel();  
8     time_disp = new javax.swing.JLabel();  
9     btn_open = new javax.swing.JButton();  
10    btn_add_friend = new javax.swing.JButton();  
11    btn_reload = new javax.swing.JButton();  
12  
13    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);  
14  
15    friend_list.addMouseListener(new java.awt.event.MouseAdapter() {  
16        public void mouseClicked(java.awt.event.MouseEvent evt) {  
17            friend_listMouseClicked(evt);  
18        }  
19    });  
20    jScrollPane1.setViewportView(friend_list);  
21  
22    jLabel1.setText("Name: ");  
23  
24    jLabel2.setText("Last Online: ");  
25  
26    time_disp.setText("WIP");  
27  
28    btn_open.setText("Open");  
29    btn_open.addActionListener(new java.awt.event.ActionListener() {  
30        public void actionPerformed(java.awt.event.ActionEvent evt) {  
31            btn_openActionPerformed(evt);  
32        }  
33    });  
34  
35    btn_add_friend.setText("Add Friend");  
36    btn_add_friend.addActionListener(new java.awt.event.ActionListener() {  
37        public void actionPerformed(java.awt.event.ActionEvent evt) {  
38            btn_add_friendActionPerformed(evt);  
39        }  
40    });  
41  
42    btn_reload.setText("Reload");  
43    btn_reload.addActionListener(new java.awt.event.ActionListener() {  
44        public void actionPerformed(java.awt.event.ActionEvent evt) {  
45            btn_reloadActionPerformed(evt);  
46        }  
47    });
```

```

1  javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
2      getContentPane().setLayout(layout);
3      layout.setHorizontalGroup(
4          layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
5              .addGroup(layout.createSequentialGroup()
6                  .addContainerGap()
7                  .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
8                      .addGroup(layout.createSequentialGroup()
9                          .addComponent(btn_reload)
10                         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
11                         .addComponent(btn_add_friend)
12                         .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 181, javax.swing.GroupLayout.PREFERRED_SIZE))
13                     .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
14                         .addGroup(layout.createSequentialGroup()
15                             .addGap(69, 69, 69)
16                             .addComponent(btn_open)
17                         .addGroup(layout.createSequentialGroup()
18                             .addGap(18, 18, 18)
19                             .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
20                                 .addGroup(layout.createSequentialGroup()
21                                     .addComponent(jLabel1)
22                                     .addGap(54, 54, 54)
23                                     .addComponent(name_disp))
24                                 .addGroup(layout.createSequentialGroup()
25                                     .addComponent(jLabel2)
26                                     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
27                                     .addComponent(time_disp))))
28                         .addGapContainerGap(75, Short.MAX_VALUE)))
29     );
30     layout.setVerticalGroup(
31         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
32             .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
33                 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
34                     .addGroup(layout.createSequentialGroup()
35                         .addGapGap(28, 28, 28)
36                         .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
37                             .addComponent(jLabel1)
38                             .addComponent(name_disp))
39                         .addGapGap(18, 18, 18)
40                         .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
41                             .addComponent(jLabel2)
42                             .addComponent(time_disp))
43                         .addGapGap(32, 32, 32)
44                         .addComponent(btn_open)
45                         .addGapGap(0, 0, Short.MAX_VALUE))
46                     .addGroup(layout.createSequentialGroup()
47                         .addGapContainerGap()
48                         .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT_SIZE, 239, Short.MAX_VALUE)))
49                 .addGapPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
50                 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
51                     .addComponent(btn_add_friend)
52                     .addComponent(btn_reload))
53                 .addGapGap(10, 10, 10)))
54     );
55
56     pack();
57 } //</editor-fold>//GEN-END:initComponents
58
59 private void btn_openActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_openActionPerformed
60     // TODO add your handling code here:
61     String selectedFriend = friend_list.getSelectedItem();
62
63     chatWindow chatWindow = new chatWindow(user_now, selectedFriend);
64     chatWindow.show();
65     this.dispose();
66 } //GEN-LAST:event_btn_openActionPerformed
67
68 private void btn_add_friendActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_add_friendActionPerformed
69     // TODO add your handling code here:
70     AddFriendDialog d = new AddFriendDialog(this, true, user_now);
71     d.show();
72 } //GEN-LAST:event_btn_add_friendActionPerformed

```

```
● ● ●
1 private void friend_listMouseClicked(java.awt.event.MouseEvent evt) { //GEN-FIRST:event_friend_listMouseClicked
2     // TODO add your handling code here:
3     int index = friend_list.getSelectedIndex();
4     name_disp.setText(friend_list.getSelectedValue());
5 } //GEN-LAST:event_friend_listMouseClicked
6
7 private void btn_reloadActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_reloadActionPerformed
8     // TODO add your handling code here:
9     fillFriends(user_now);
10 } //GEN-LAST:event_btn_reloadActionPerformed
11
12 /**
13 * @param args the command line arguments
14 */
15
16 // Variables declaration - do not modify//GEN-BEGIN:variables
17 private javax.swing.JButton btn_add_friend;
18 private javax.swing.JButton btn_open;
19 private javax.swing.JButton btn_reload;
20 private javax.swing.JList<String> friend_list;
21 private javax.swing.JLabel jLabel1;
22 private javax.swing.JLabel jLabel2;
23 private javax.swing.JScrollPane jScrollPane1;
24 private javax.swing.JLabel name_disp;
25 private javax.swing.JLabel time_disp;
26 // End of variables declaration//GEN-END:variables
27 }
```

3.1.4.3. chatWindow.java

```
1 package views;
2
3 import Controllers.MessageController;
4 import static java.lang.Thread.sleep;
5 import java.sql.*;
6 import java.util.concurrent.Executors;
7 import java.util.concurrent.ScheduledExecutorService;
8 import java.util.concurrent.TimeUnit;
9 import javax.swing.JTextArea;
10
11 /**
12  *
13  * @author naufalabdillah
14  */
15 public final class chatWindow extends javax.swing.JFrame {
16     MessageController ctr = new MessageController();
17     String user_now;
18     String user_friend;
19     /**
20      * Creates new form chatWindow
21     */
22
23     public void refresh(){
24         ctr.fillMessage(user_now, user_friend, chat_window);
25     }
26
27
28     public chatWindow(String user_now, String user_friend) {
29         this.setTitle(user_friend);
30         this.user_now = user_now;
31         this.user_friend = user_friend;
32         initComponents();
33         ctr.fillMessage(user_now, user_friend, chat_window);
34         chat_window.setLineWrap(true);
35
36         Runnable fillMsgsec = new Runnable() {
37             public void run() {
38                 refresh();
39             }
40         };
41
42         ScheduledExecutorService executor = Executors.newScheduledThreadPool(1);
43         executor.scheduleAtFixedRate(fillMsgsec, 0, 300, TimeUnit.MILLISECONDS);
44     }
45 }
```

```
1  @SuppressWarnings("unchecked")
2  // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
3  private void initComponents() {
4
5      jScrollPane1 = new javax.swing.JScrollPane();
6      chat_window = new javax.swing.JTextArea();
7      input_text = new javax.swing.JTextField();
8      btn_send = new javax.swing.JButton();
9      btn_back = new javax.swing.JButton();
10
11     setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
12
13     chat_window.setColumns(20);
14     chat_window.setRows(5);
15     jScrollPane1.setViewportView(chat_window);
16
17     btn_send.setText("Send");
18     btn_send.addActionListener(new java.awt.event.ActionListener() {
19         public void actionPerformed(java.awt.event.ActionEvent evt) {
20             btn_sendActionPerformed(evt);
21         }
22     });
23
24     btn_back.setText("Back");
25     btn_back.addActionListener(new java.awt.event.ActionListener() {
26         public void actionPerformed(java.awt.event.ActionEvent evt) {
27             btn_backActionPerformed(evt);
28         }
29     });
30
31     javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
32     getContentPane().setLayout(layout);
33     layout.setHorizontalGroup(
34         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
35             .addGroup(layout.createSequentialGroup()
36                 .addGap(10, 10, 10)
37                 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
38                     .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
39                     .addGroup(layout.createSequentialGroup()
40                         .addComponent(input_text, javax.swing.GroupLayout.PREFERRED_SIZE, 237, javax.swing.GroupLayout.PREFERRED_SIZE)
41                         .addGap(10, 10, 10)
42                         .addComponent(btn_send, javax.swing.GroupLayout.PREFERRED_SIZE, 79, javax.swing.GroupLayout.PREFERRED_SIZE)
43                         .addGap(10, 10, 10)
44                         .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 326, javax.swing.GroupLayout.PREFERRED_SIZE))
45                 .addGap(10, 10, 10)
46                 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
47                     .addComponent(btn_back, javax.swing.GroupLayout.PREFERRED_SIZE, 269, javax.swing.GroupLayout.PREFERRED_SIZE)
48                     .addGroup(layout.createSequentialGroup()
49                         .addGap(10, 10, 10)
50                         .addComponent(btn_back, javax.swing.GroupLayout.PREFERRED_SIZE, 269, javax.swing.GroupLayout.PREFERRED_SIZE)
51                         .addGap(10, 10, 10)
52                         .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 269, javax.swing.GroupLayout.PREFERRED_SIZE)
53                         .addGap(10, 10, 10)
54                         .addComponent(btn_send, javax.swing.GroupLayout.PREFERRED_SIZE, 34, javax.swing.GroupLayout.PREFERRED_SIZE)
55                         .addGap(10, 10, 10)
56                         .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 34, javax.swing.GroupLayout.PREFERRED_SIZE))
57                 .addGap(10, 10, 10)
58             )
59         )
60     );
61
62     pack();
63 }// </editor-fold>//GEN-END:initComponents
64
65 private void btn_sendActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_sendActionPerformed
66     // TODO add your handling code here:
67     String msg = input_text.getText();
68     ctr.sendMessage(msg, user_now, user_friend);
69     input_text.setText("");
70     //ctr.fillMessage(user_now, user_friend, chat_window);
71 }//GEN-LAST:event_btn_sendActionPerformed
72
73 private void btn_backActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_backActionPerformed
74     // TODO add your handling code here:
75     chatList landView = new chatList(user_now);
76     landView.show();
77     this.dispose();
78 }
```

```
1 private javax.swing.JButton btn_back;
2     private javax.swing.JButton btn_send;
3     private javax.swing.JTextArea chat_window;
4     private javax.swing.JTextField input_text;
5     private javax.swing.JScrollPane jScrollPane1;
6     // End of variables declaration//GEN-END:variables
7 }
```

3.1.4.4. CustomizationDialog.java

```
 1 package Views;
 2
 3 import Controllers.UserController;
 4 import javax.swing.*;
 5
 6 /**
 7 *
 8 * @author johan
 9 */
10 public class CustomizationDialog extends javax.swing.JFrame {
11
12     /**
13      * Creates new form CustomizationDialog
14      */
15     public CustomizationDialog() {
16         initComponents();
17     }
18
19     /**
20      * This method is called from within the constructor to initialize the form.
21      * WARNING: Do NOT modify this code. The content of this method is always
22      * regenerated by the Form Editor.
23      */
24     @SuppressWarnings("unchecked")
25     // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
26     private void initComponents() {
27
28         jLabel1 = new javax.swing.JLabel();
29         jLabel2 = new javax.swing.JLabel();
30         jLabel3 = new javax.swing.JLabel();
31         jLabel4 = new javax.swing.JLabel();
32         input_user_update = new javax.swing.JTextField();
33         input_pass_update = new javax.swing.JPasswordField();
34         input_confirmpass_update = new javax.swing.JPasswordField();
35         btn_update = new javax.swing.JButton();
36
37         setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
38
39         jLabel1.setText("Customization");
40
41         jLabel2.setText("username");
42
43         jLabel3.setText("password");
44
45         jLabel4.setText("confirm password");
46
47         input_user_update.addActionListener(new java.awt.event.ActionListener() {
48             public void actionPerformed(java.awt.event.ActionEvent evt) {
49                 input_user_updateActionPerformed(evt);
50             }
51         });
52
53         btn_update.setText("Update");
54         btn_update.addActionListener(new java.awt.event.ActionListener() {
55             public void actionPerformed(java.awt.event.ActionEvent evt) {
56                 btn_updateActionPerformed(evt);
57             }
58         });
59     }
60
61     // Variables declaration - do not modify//GEN-END:variables
62 }
```

```

1  javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
2  getContentPane().setLayout(layout);
3  layout.setHorizontalGroup(
4      layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
5          .addGroup(layout.createSequentialGroup()
6              .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
7                  .addGroup(layout.createSequentialGroup()
8                      .addGap(171, 171, 171)
9                      .addComponent(jLabel1))
10                 .addGroup(layout.createSequentialGroup()
11                     .addGap(86, 86, 86)
12                     .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
13                         .addComponent(jLabel3)
14                         .addComponent(jLabel2)
15                         .addComponent(jLabel4))
16                     .addGap(38, 38, 38)
17                     .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
18                         .addComponent(input_user_update, javax.swing.GroupLayout.DEFAULT_SIZE, 115, Short.MAX_VALUE)
19                         .addComponent(input_pass_update)
20                         .addComponent(input_confirmpass_update)))
21                 .addGroup(layout.createSequentialGroup()
22                     .addGap(142, 142, 142)
23                     .addComponent(btn_update)))
24             .addGapContainerGap(77, Short.MAX_VALUE)
25         );
26     layout.setVerticalGroup(
27         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
28             .addGroup(layout.createSequentialGroup()
29                 .addGapGap(27, 27, 27)
30                 .addComponent(jLabel1)
31                 .addGapGap(30, 30, 30)
32                 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
33                     .addComponent(jLabel2)
34                     .addComponent(input_user_update, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
35                 .addGapGap(24, 24, 24)
36                 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
37                     .addComponent(jLabel3)
38                     .addComponent(input_pass_update, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
39                 .addGapGap(26, 26, 26)
40                 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
41                     .addComponent(jLabel4)
42                     .addComponent(input_confirmpass_update, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
43                 .addGapPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 53, Short.MAX_VALUE)
44                 .addComponent(btn_update)
45                 .addGapGap(43, 43, 43))
46     );
47
48     pack();
49 }

```

```
 1 private void input_user_updateActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_input_user_updateActionPerformed
 2     // TODO add your handling code here:
 3 } //GEN-LAST:event_input_user_updateActionPerformed
 4
 5 private void btn_updateActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_updateActionPerformed
 6     // TODO add your handling code here:
 7     try {
 8         String username = input_user_update.getText();
 9         String password = String.valueOf(input_pass_update.getPassword());
10         String confirmPassword = String.valueOf(input_confirmpass_update.getPassword());
11
12         if (username.isEmpty()) {
13             JOptionPane.showMessageDialog(null, "Username tidak boleh kosong");
14             this.requestFocus();
15         }
16         else if (!password.equals(confirmPassword)) {
17             JOptionPane.showMessageDialog(null, "Password tidak sesuai");
18         } else {
19             ctrl.updateUser(1, username, password);
20             this.dispose();
21             LoginView logView = new LoginView();
22             logView.show();
23         }
24     } catch (Exception e) {
25         System.out.println("Register Gagal");
26     }
27 } //GEN-LAST:event_btn_updateActionPerformed
28
29 /**
30 * @param args the command line arguments
31 */
32 public static void main(String args[]) {
33     /* Set the Nimbus look and feel */
34     // <editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
35     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
36      * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
37     */
38     try {
39         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
40             if ("Nimbus".equals(info.getName())) {
41                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
42                 break;
43             }
44         }
45     } catch (ClassNotFoundException ex) {
46         java.util.logging.Logger.getLogger(CustomizationDialog.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
47     } catch (InstantiationException ex) {
48         java.util.logging.Logger.getLogger(CustomizationDialog.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
49     } catch (IllegalAccessException ex) {
50         java.util.logging.Logger.getLogger(CustomizationDialog.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
51     } catch (javax.swing.UnsupportedLookAndFeelException ex) {
52         java.util.logging.Logger.getLogger(CustomizationDialog.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
53     }
54     // </editor-fold>
55
56     /* Create and display the form */
57     java.awt.EventQueue.invokeLater(new Runnable() {
58         public void run() {
59             new CustomizationDialog().setVisible(true);
60         }
61     });
62 }
63 private UserController ctrl = new UserController();
64 // Variables declaration - do not modify//GEN-BEGIN:variables
65 private javax.swing.JButton btn_update;
66 private javax.swing.JPasswordField input_confirmpass_update;
67 private javax.swing.JPasswordField input_pass_update;
68 private javax.swing.JTextField input_user_update;
69 private javax.swing.JLabel jLabel1;
70 private javax.swing.JLabel jLabel2;
71 private javax.swing.JLabel jLabel3;
72 private javax.swing.JLabel jLabel4;
73 // End of variables declaration//GEN-END:variables
74 }
```

3.1.4.5. LandingView.java

```
1 package Views;
2
3 /**
4 *
5 * @author naufalabdillah
6 */
7 public class landingView extends javax.swing.JFrame {
8
9     /**
10      * Creates new form landingView
11      */
12     public landingView() {
13         initComponents();
14     }
15
16     /**
17      * This method is called from within the constructor to initialize the form.
18      * WARNING: Do NOT modify this code. The content of this method is always
19      * regenerated by the Form Editor.
20      */
21     @SuppressWarnings("unchecked")
22 // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
23     private void initComponents() {
24
25         jLabel1 = new javax.swing.JLabel();
26         jLabel2 = new javax.swing.JLabel();
27         jLabel3 = new javax.swing.JLabel();
28         btn_to_reg = new javax.swing.JButton();
29         btn_to_login = new javax.swing.JButton();
30
31         setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
32
33         jLabel1.setFont(new java.awt.Font("Tahoma", 0, 24)); // NOI18N
34         jLabel1.setText("Welcome to SocialLife!");
35
36         jLabel2.setText("Don't have an account?");
37
38         jLabel3.setText("Already have an account?");
39
40         btn_to_reg.setText("Register");
41         btn_to_reg.addActionListener(new java.awt.event.ActionListener() {
42             public void actionPerformed(java.awt.event.ActionEvent evt) {
43                 btn_to_regActionPerformed(evt);
44             }
45         });
46
47         btn_to_login.setText("Login");
48         btn_to_login.addActionListener(new java.awt.event.ActionListener() {
49             public void actionPerformed(java.awt.event.ActionEvent evt) {
50                 btn_to_loginActionPerformed(evt);
51             }
52         });
53     }
54
55 }
```

```
1 javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
2 getContentPane().setLayout(layout);
3 layout.setHorizontalGroup(
4     layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
5         .addGroup(layout.createSequentialGroup()
6             .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
7                 .addGroup(layout.createSequentialGroup()
8                     .addGap(83, 83, 83)
9                     .addComponent(jLabel1))
10                .addGroup(layout.createSequentialGroup()
11                    .addGap(88, 88, 88)
12                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
13                        .addGroup(layout.createSequentialGroup()
14                            .addComponent(jLabel3)
15                            .addGap(18, 18, 18)
16                            .addComponent(btn_to_login))
17                        .addGroup(layout.createSequentialGroup()
18                            .addComponent(jLabel2)
19                            .addGap(18, 18, 18)
20                            .addComponent(btn_to_reg))))))
21            .addGap(82, Short.MAX_VALUE))
22        );
23    layout.setVerticalGroup(
24        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
25            .addGroup(layout.createSequentialGroup()
26                .addGap(77, 77, 77)
27                .addComponent(jLabel1)
28                .addGap(28, 28, 28)
29                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
30                    .addComponent(jLabel2)
31                    .addComponent(btn_to_reg))
32                .addGap(18, 18, 18)
33                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
34                    .addComponent(jLabel3)
35                    .addComponent(btn_to_login))
36            .addGap(103, Short.MAX_VALUE))
37        );
38    )
39    pack();
40 }
```

```

1  private void btn_to_regActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_to_regActionPerformed
2      // TODO add your handling code here:
3      RegisterView regView = new RegisterView();
4      regView.show();
5      dispose();
6  }//GEN-LAST:event_btn_to_regActionPerformed
7
8  private void btn_to_loginActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_to_loginActionPerformed
9      // TODO add your handling code here:
10     LoginView logView = new LoginView();
11     logView.show();
12     dispose();
13 };//GEN-LAST:event_btn_to_loginActionPerformed
14
15 /**
16  * @param args the command line arguments
17 */
18 public static void main(String args[]) {
19     /* Set the Nimbus look and feel */
20     //editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
21     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
22      * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
23     */
24     try {
25         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
26             if ("Nimbus".equals(info.getName())) {
27                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
28                 break;
29             }
30         }
31     } catch (ClassNotFoundException ex) {
32         java.util.logging.Logger.getLogger(landingView.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
33     } catch (InstantiationException ex) {
34         java.util.logging.Logger.getLogger(landingView.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
35     } catch (IllegalAccessException ex) {
36         java.util.logging.Logger.getLogger(landingView.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
37     } catch (javax.swing.UnsupportedLookAndFeelException ex) {
38         java.util.logging.Logger.getLogger(landingView.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
39     }
40     //
41
42     /* Create and display the form */
43     java.awt.EventQueue.invokeLater(new Runnable() {
44         public void run() {
45             new landingView().setVisible(true);
46         }
47     });
48 }
49
50 // Variables declaration - do not modify//GEN-BEGIN:variables
51 private javax.swing.JButton btn_to_login;
52 private javax.swing.JButton btn_to_reg;
53 private javax.swing.JLabel jLabel1;
54 private javax.swing.JLabel jLabel2;
55 private javax.swing.JLabel jLabel3;
56 // End of variables declaration//GEN-END:variables
57 }

```

3.1.4.6. LoginView.java

```
● ○ ●
1 package Views;
2
3 import Controllers.UserController;
4 import javax.swing.*;
5 import java.sql.*;
6
7 /**
8 *
9 * @author naufalabdillah
10 */
11 public class LoginView extends javax.swing.JFrame {
12
13     /**
14      * Creates new form LoginView
15      */
16     public LoginView() {
17         initComponents();
18     }
19
20     /**
21      * This method is called from within the constructor to initialize the form.
22      * WARNING: Do NOT modify this code. The content of this method is always
23      * regenerated by the Form Editor.
24      */
25     @SuppressWarnings("unchecked")
26     // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
27     private void initComponents() {
28
29         jLabel1 = new javax.swing.JLabel();
30         jLabel2 = new javax.swing.JLabel();
31         jLabel3 = new javax.swing.JLabel();
32         input_user_login = new javax.swing.JTextField();
33         input_pass_login = new javax.swing.JPasswordField();
34         btn_login = new javax.swing.JButton();
35         jLabel4 = new javax.swing.JLabel();
36         btn_to_reg = new javax.swing.JButton();
37
38         setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
39
40         jLabel1.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
41         jLabel1.setText("LOGIN PAGE");
42
43         jLabel2.setText("nama");
44
45         jLabel3.setText("password");
46
47         btn_login.setText("Login");
48         btn_login.addActionListener(new java.awt.event.ActionListener() {
49             public void actionPerformed(java.awt.event.ActionEvent evt) {
50                 btn_loginActionPerformed(evt);
51             }
52         });
53     }
54 }
```

```
1 jLabel4.setText("Don't have an account?");
2
3     btn_to_reg.setText("Register");
4     btn_to_reg.addActionListener(new java.awt.event.ActionListener() {
5         public void actionPerformed(java.awt.event.ActionEvent evt) {
6             btn_to_regActionPerformed(evt);
7         }
8     });
9
10    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
11    getContentPane().setLayout(layout);
12    layout.setHorizontalGroup(
13        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
14            .addGroup(layout.createSequentialGroup()
15                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
16                    .addGroup(layout.createSequentialGroup()
17                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
18                            .addGroup(layout.createSequentialGroup()
19                                .addGap(143, 143, 143)
20                                .addComponent(jLabel1))
21                            .addGroup(layout.createSequentialGroup()
22                                .addGap(43, 43, 43)
23                                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
24                                    .addComponent(jLabel2)
25                                    .addComponent(jLabel3)))
26                            .addGap(22, 22, 22)
27                            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
28                                .addComponent(input_user_login)
29                                .addComponent(input_pass_login, javax.swing.GroupLayout.DEFAULT_SIZE, 214, Short.MAX_VALUE)))
30                            .addGap(0, 54, Short.MAX_VALUE))
31                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
32                            .addGroup(layout.createSequentialGroup()
33                                .addGap(0, 0, Short.MAX_VALUE)
34                                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
35                                    .addComponent(btn_login)
36                                    .addGroup(layout.createSequentialGroup()
37                                        .addComponent(jLabel4)
38                                        .addGap(18, 18, 18)
39                                        .addComponent(btn_to_reg))))))
40                    .addGapContainerGap()
41                );
42            );
43        );
44    )
45 }
```

```
1  layout.setVerticalGroup(
2      layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
3          .addGroup(layout.createSequentialGroup()
4              .addContainerGap()
5              .addComponent(jLabel1)
6              .addGap(32, 32, 32)
7              .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
8                  .addComponent(jLabel2)
9                  .addComponent(input_user_login, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
10             .addGap(18, 18, 18)
11             .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
12                 .addComponent(jLabel3)
13                 .addComponent(input_pass_login, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
14             .addGap(41, 41, 41)
15             .addComponent(btn_login)
16             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 67, Short.MAX_VALUE)
17             .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
18                 .addComponent(btn_to_reg)
19                 .addComponent(jLabel4))
20             .addContainerGap())
21     );
22 
23     pack();
24 } // </editor-fold> //GEN-END:initComponents
25 
26 private void btn_loginActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_loginActionPerformed
27     // TODO add your handling code here:
28     try {
29         String username = input_user_login.getText();
30         String password = input_pass_login.getText();
31 
32         ctrl.loginUser(username, password);
33         dispose();
34     } catch (Exception e) {
35         e.printStackTrace();
36     }
37 } //GEN-LAST:event_btn_loginActionPerformed
38 
39 private void btn_to_regActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_to_regActionPerformed
40     // TODO add your handling code here:
41     dispose();
42     RegisterView regView = new RegisterView();
43     regView.show();
44 }
```

```
1 public static void main(String args[]) {
2     /* Set the Nimbus look and feel */
3     //editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
4     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
5      * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
6      */
7     try {
8         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
9             if ("Nimbus".equals(info.getName())) {
10                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
11                 break;
12             }
13         }
14     } catch (ClassNotFoundException ex) {
15         java.util.logging.Logger.getLogger(LoginView.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
16     } catch (InstantiationException ex) {
17         java.util.logging.Logger.getLogger(LoginView.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
18     } catch (IllegalAccessException ex) {
19         java.util.logging.Logger.getLogger(LoginView.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
20     } catch (javax.swing.UnsupportedLookAndFeelException ex) {
21         java.util.logging.Logger.getLogger(LoginView.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
22     }
23 //
```

/* Create and display the form */

```
24     java.awt.EventQueue.invokeLater(new Runnable() {
25         public void run() {
26             new LoginView().setVisible(true);
27         }
28     });
29 }
30
31
32 private UserController ctrl = new UserController();
33 // Variables declaration - do not modify//GEN-BEGIN:variables
34 private javax.swing.JButton btn_login;
35 private javax.swing.JButton btn_to_reg;
36 private javax.swing.JPasswordField input_pass_login;
37 private javax.swing.JTextField input_user_login;
38 private javax.swing.JLabel jLabel1;
39 private javax.swing.JLabel jLabel2;
40 private javax.swing.JLabel jLabel3;
41 private javax.swing.JLabel jLabel4;
42
43 // End of variables declaration//GEN-END:variables
44 }
```

3.1.4.7. RegisterView.java

```
1 package Views;
2 import Controllers.UserController;
3 import javax.swing.JOptionPane;
4 /**
5  *
6  * @author johan
7  */
8 public class RegisterView extends javax.swing.JFrame {
9
10    /**
11     * Creates new form RegisterView
12     */
13    public RegisterView() {
14        this.ctrl = new UserController();
15        initComponents();
16    }
17
18    /**
19     * This method is called from within the constructor to initialize the form.
20     * WARNING: Do NOT modify this code. The content of this method is always
21     * regenerated by the Form Editor.
22     */
23    @SuppressWarnings("unchecked")
24    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
25    private void initComponents() {
26
27        jLabel1 = new javax.swing.JLabel();
28        jLabel2 = new javax.swing.JLabel();
29        jLabel3 = new javax.swing.JLabel();
30        jLabel4 = new javax.swing.JLabel();
31        input_user_reg = new javax.swing.JTextField();
32        input_pass_reg = new javax.swing.JPasswordField();
33        input_conf_reg = new javax.swing.JPasswordField();
34        btn_reg = new javax.swing.JButton();
35        jLabel5 = new javax.swing.JLabel();
36        btn_to_login = new javax.swing.JButton();
37
38        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
39
40        jLabel1.setText("REGISTER PAGE");
41
42        jLabel2.setText("Nama");
43
44        jLabel3.setText("Password");
45
46        jLabel4.setText("Confirm Password");
47
48        btn_reg.setText("Register");
49        btn_reg.addActionListener(new java.awt.event.ActionListener() {
50            public void actionPerformed(java.awt.event.ActionEvent evt) {
51                btn_regActionPerformed(evt);
52            }
53        });
54
55        jLabel5.setText("Already have an account?");
56
57        btn_to_login.setText("Login");
58        btn_to_login.addActionListener(new java.awt.event.ActionListener() {
59            public void actionPerformed(java.awt.event.ActionEvent evt) {
60                btn_to_loginActionPerformed(evt);
61            }
62        });
63    }
64
65    // Variables declaration - do not modify//GEN-END:initComponents
66    private UserController ctrl;
67    private javax.swing.JButton btn_to_login;
68    private javax.swing.JButton btn_reg;
69    private javax.swing.JLabel jLabel1;
70    private javax.swing.JLabel jLabel2;
71    private javax.swing.JLabel jLabel3;
72    private javax.swing.JLabel jLabel4;
73    private javax.swing.JLabel jLabel5;
74    private javax.swing.JPasswordField input_conf_reg;
75    private javax.swing.JPasswordField input_pass_reg;
76    private javax.swing.JTextField input_user_reg;
```

```
● ● ●
1 javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
2 getContentPane().setLayout(layout);
3 layout.setHorizontalGroup(
4     layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
5         .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
6             .addGap(152, Short.MAX_VALUE)
7             .addComponent(jLabel15)
8             .addGap(18, 18, 18)
9             .addComponent(btn_to_login)
10            .addGap(21, 21, 21))
11         .addGroup(layout.createSequentialGroup()
12             .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
13                 .addGroup(layout.createSequentialGroup()
14                     .addGap(145, 145, 145)
15                     .addComponent(jLabel1))
16                 .addGroup(layout.createSequentialGroup()
17                     .addGap(37, 37, 37)
18                     .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
19                         .addGroup(layout.createSequentialGroup()
20                             .addGap(58, 58, 58)
21                             .addComponent(jLabel3)
22                             .addComponent(jLabel2))
23                         .addGroup(layout.createSequentialGroup()
24                             .addGap(189, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
25                             .addComponent(input_user_reg)
26                             .addComponent(input_pass_reg, javax.swing.GroupLayout.PREFERRED_SIZE, 94, javax.swing.GroupLayout.PREFERRED_SIZE)
27                             .addComponent(input_conf_reg)))))))
28             .addGroup(layout.createSequentialGroup()
29                 .addComponent(jLabel4)
30                 .addGapPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
31                 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
32                     .addComponent(btn_reg, javax.swing.GroupLayout.PREFERRED_SIZE, 94, javax.swing.GroupLayout.PREFERRED_SIZE)
33                     .addComponent(input_conf_reg))))))
34         .addGapContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
);
});
```

```

1 layout.setVerticalGroup(
2     layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
3     .addGroup(layout.createSequentialGroup()
4         .addGap(29, 29, 29)
5         .addComponent(jLabel1)
6         .addGap(28, 28, 28)
7         .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
8             .addComponent(jLabel2)
9             .addComponent(input_user_reg, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
10        .addGap(18, 18, 18)
11        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
12            .addComponent(jLabel3)
13            .addComponent(input_pass_reg, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
14        .addGap(18, 18, 18)
15        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
16            .addComponent(jLabel4)
17            .addComponent(input_conf_reg, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
18        .addGap(26, 26, 26)
19        .addComponent(btn_reg, javax.swing.GroupLayout.PREFERRED_SIZE, 34, javax.swing.GroupLayout.PREFERRED_SIZE)
20        .addGap(44, 44, 44)
21        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
22            .addComponent(jLabel5)
23            .addComponent(btn_to_login)))
24        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
25    );
26
27     pack();
28 } // </editor-fold>//GEN-END:initComponents
29
30 private void btn_regActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_regActionPerformed
31     // TODO add your handling code here:
32     try {
33         String username = input_user_reg.getText();
34         String password = String.valueOf(input_pass_reg.getPassword());
35         String confirmPassword = String.valueOf(input_conf_reg.getPassword());
36
37         if (username.isEmpty()) {
38             JOptionPane.showMessageDialog(null, "Username tidak boleh kosong");
39             this.requestFocus();
40         }
41         else if (!password.equals(confirmPassword)) {
42             JOptionPane.showMessageDialog(null, "Password tidak sesuai");
43         } else {
44             ctrl.registerUser(username, password);
45             this.dispose();
46             LoginView logView = new LoginView();
47             logView.show();
48         }
49     } catch (Exception e) {
50         System.out.println("Register Gagal");
51     }
52 }

```

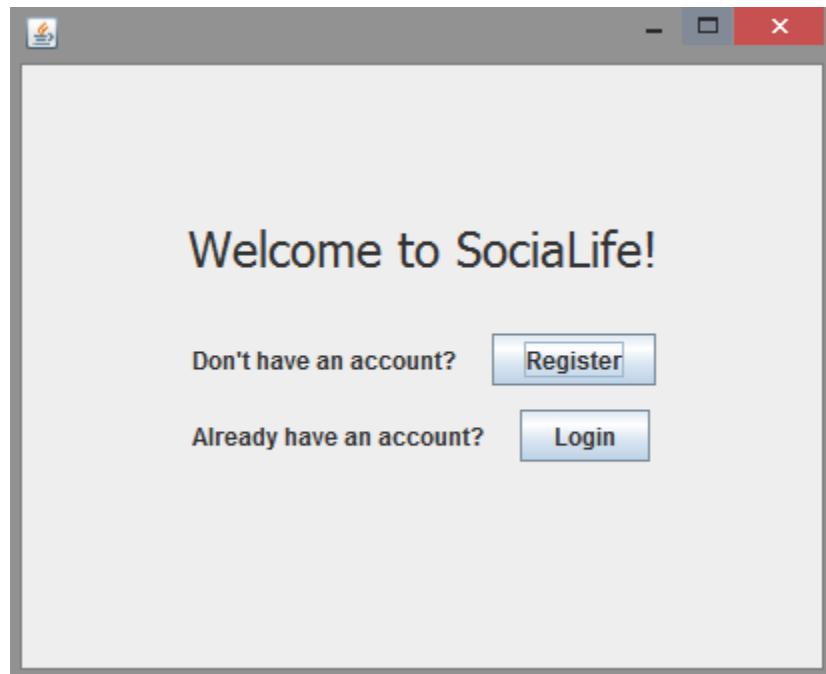
```

1 private void btn_to_loginActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_to_loginActionPerformed
2     // TODO add your handling code here:
3     dispose();
4     LoginView logView = new LoginView();
5     logView.show();
6 } //GEN-LAST:event_btn_to_loginActionPerformed
7
8 /**
9  * @param args the command line arguments
10 */
11 public static void main(String args[]) {
12     /* Set the Nimbus look and feel */
13     // <editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) " >
14     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
15      * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
16      */
17     try {
18         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
19             if ("Nimbus".equals(info.getName())) {
20                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
21                 break;
22             }
23         }
24     } catch (ClassNotFoundException ex) {
25         java.util.logging.Logger.getLogger(RegisterView.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
26     } catch (InstantiationException ex) {
27         java.util.logging.Logger.getLogger(RegisterView.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
28     } catch (IllegalAccessException ex) {
29         java.util.logging.Logger.getLogger(RegisterView.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
30     } catch (javax.swing.UnsupportedLookAndFeelException ex) {
31         java.util.logging.Logger.getLogger(RegisterView.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
32     }
33     // </editor-fold>
34
35     /* Create and display the form */
36     java.awt.EventQueue.invokeLater(new Runnable() {
37         public void run() {
38             new RegisterView().setVisible(true);
39         }
40     });
41 }
42 private UserController ctrl = new UserController();
43 // Variables declaration - do not modify//GEN-BEGIN:variables
44 private javax.swing.JButton btn_reg;
45 private javax.swing.JButton btn_to_login;
46 private javax.swing.JPasswordField input_conf_reg;
47 private javax.swing.JPasswordField input_pass_reg;
48 private javax.swing.JTextField input_user_reg;
49 private javax.swing.JLabel jLabel1;
50 private javax.swing.JLabel jLabel2;
51 private javax.swing.JLabel jLabel3;
52 private javax.swing.JLabel jLabel4;
53 private javax.swing.JLabel jLabel5;
54 // End of variables declaration//GEN-END:variables
55 }

```

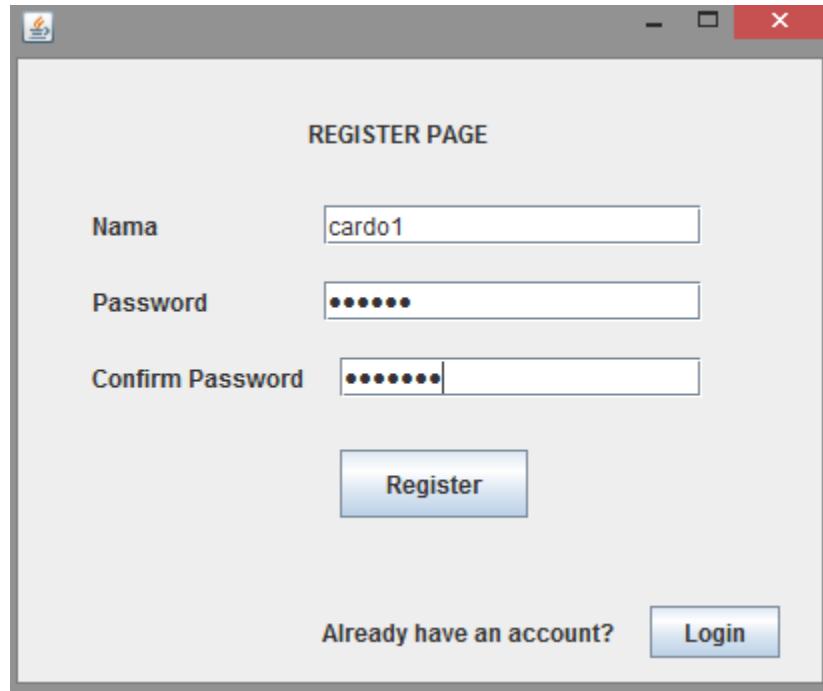
3.2. Screenshot Aplikasi

3.2.1. Landing page



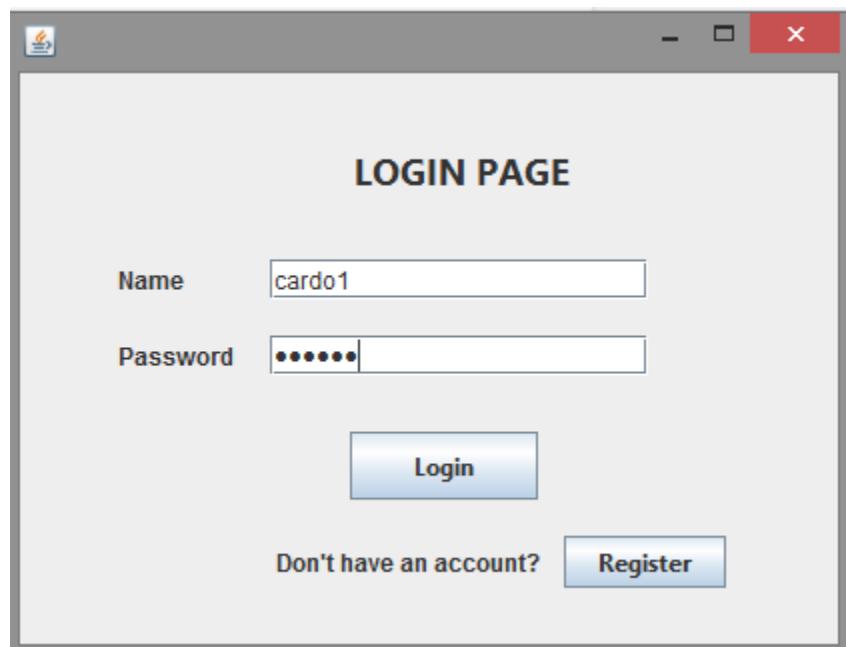
Keterangan : Halaman ini adalah halaman paling awal yang akan ditampilkan kepada user. User disini dapat memilih register dan login.

3.2.2. Register



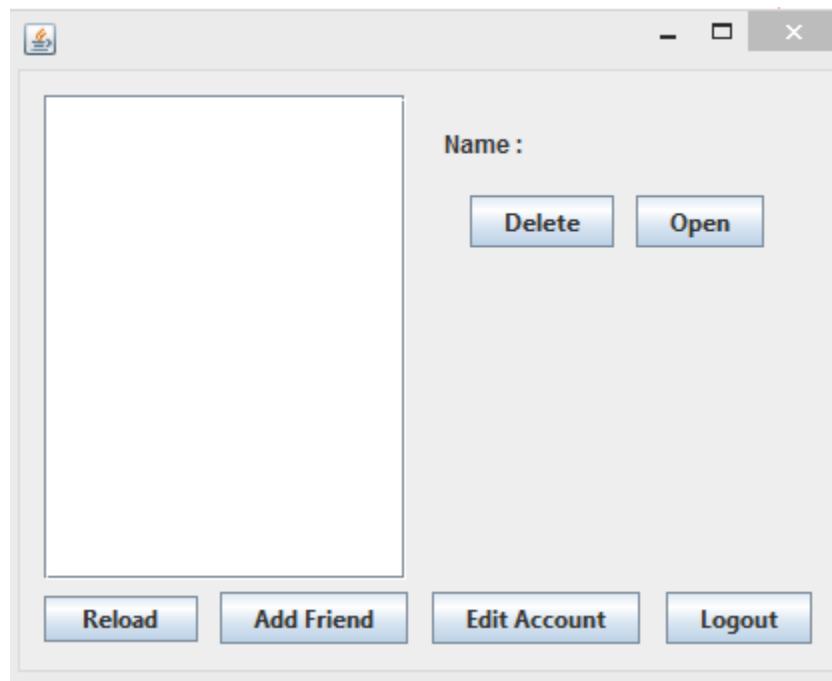
Keterangan : Di Halaman register user diminta untuk menginput nama, password dan confirm passwordnya. Ketika sudah, user akan mengklik tombol register dan akan masuk kedalam halaman login.

3.2.3. Login



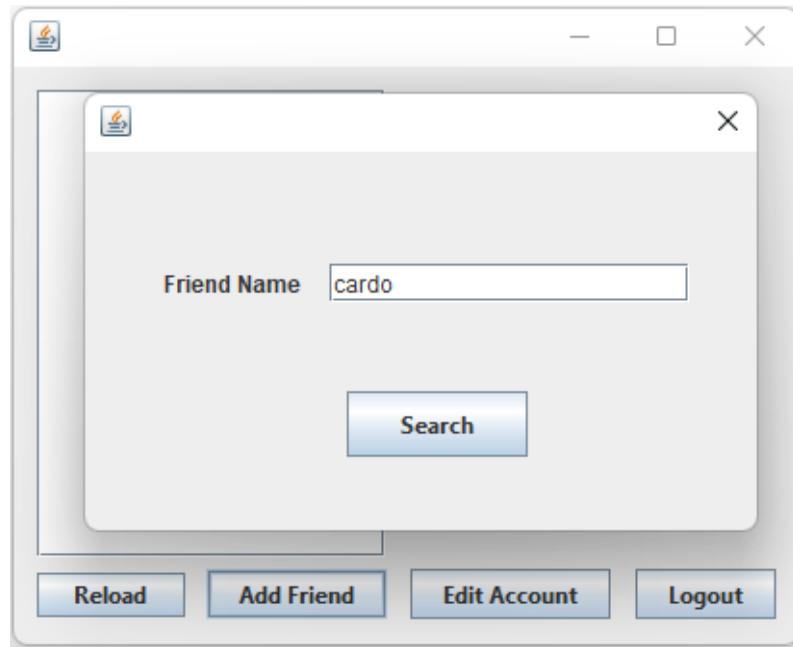
Keterangan : Di Halaman login user menginputkan nama dan password yang sudah diregistrasi sebelumnya. Sesudah user mengklik tombol login maka user akan masuk kedalam halaman chat.

3.2.4. Chat List

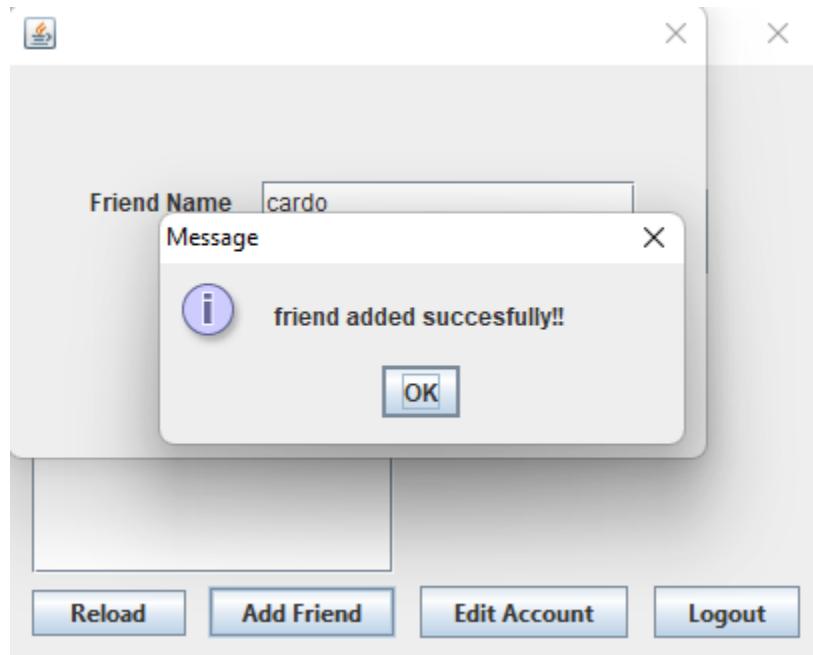


Keterangan : Di Halaman chat List, user belum memiliki teman, maka user dapat menambahkan teman dengan cara menggunakan tombol addfriend.

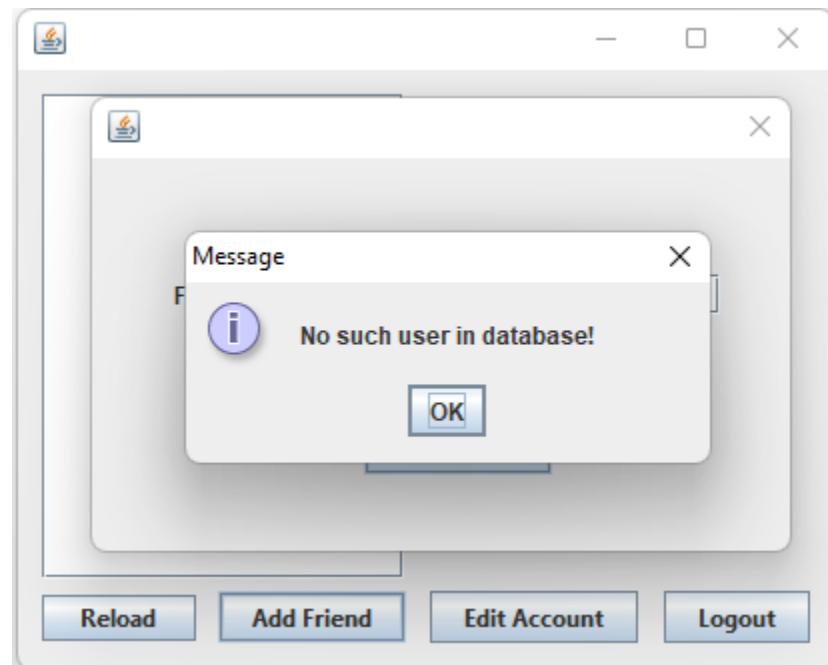
3.2.5. Add Friend



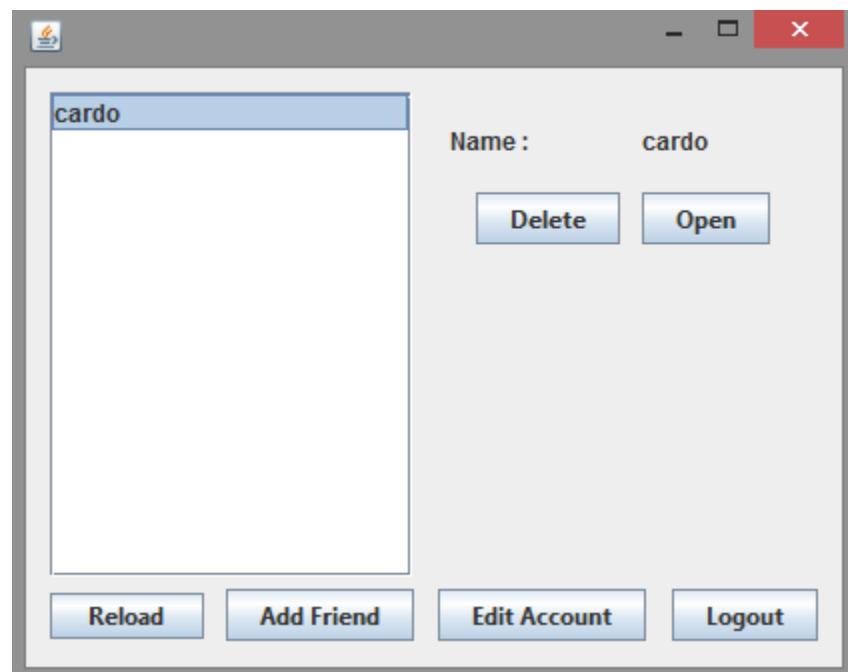
Keterangan : Kemudian ketika di dalam addfriends, user diminta untuk menginputkan nama teman kita yang sudah ada datanya di dalam database aplikasinya dan menekan tombol search. Maka akan muncul pemberitahuan “friend added successfully!!”.



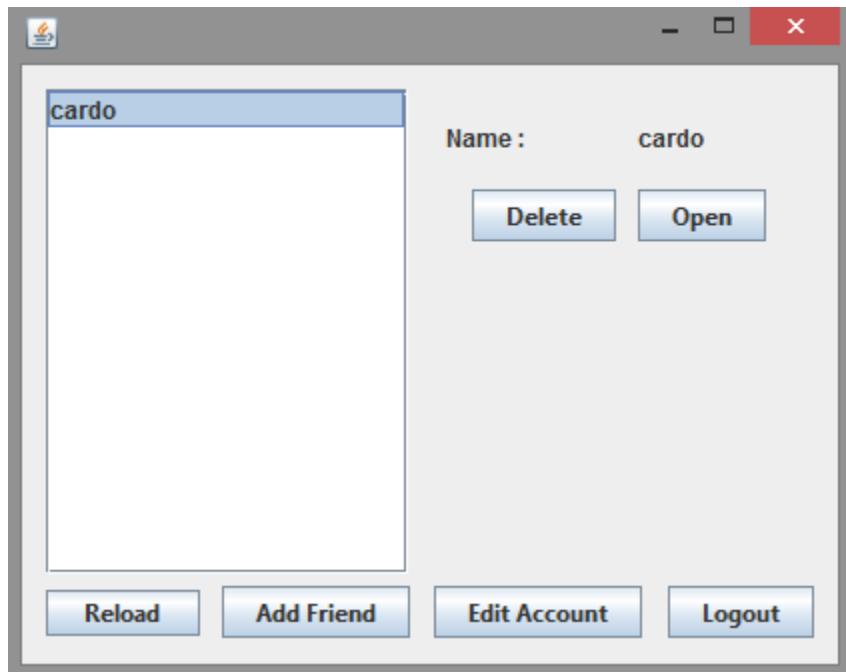
Keterangan : Jika kita mencari nama teman yang tidak ada didalam database aplikasinya maka akan muncul pemberitahuan no such user in database.



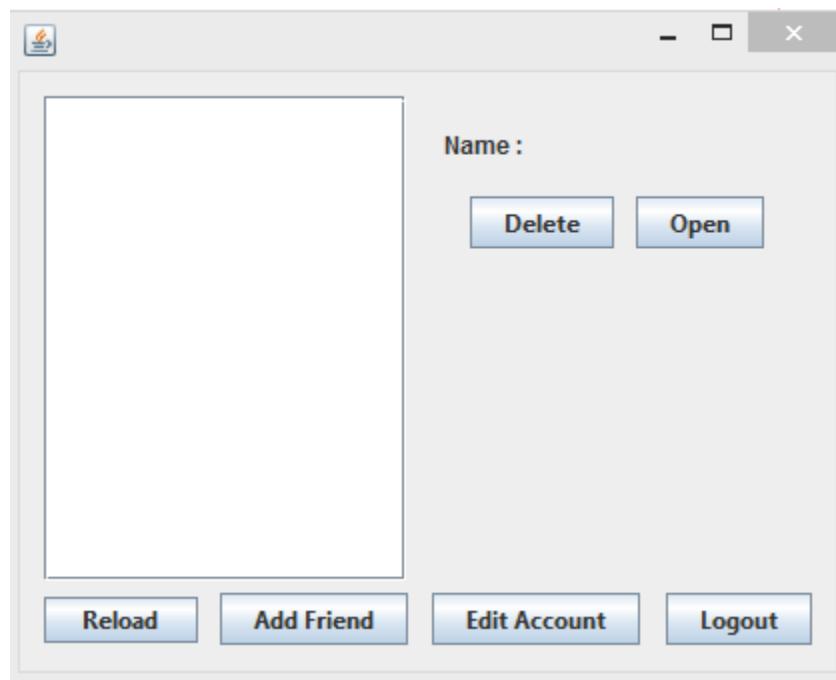
Keterangan : Setelah kita berhasil menambahkan teman maka nama teman tersebut akan masuk kedalam list yang ada di dalam halaman chat tersebut. Jika tidak ada, maka dapat mengklik tombol reload.



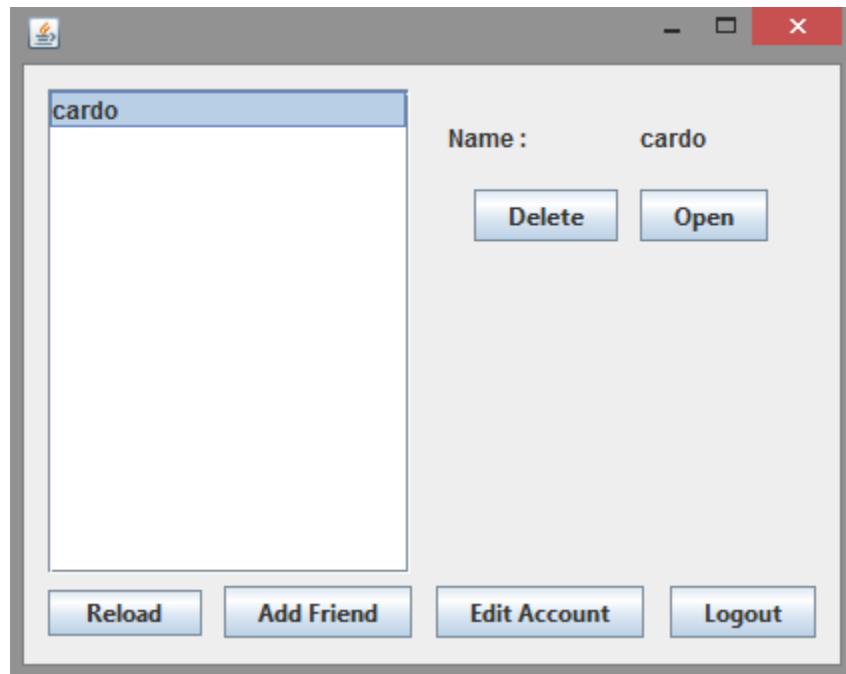
3.2.6. Delete Friend



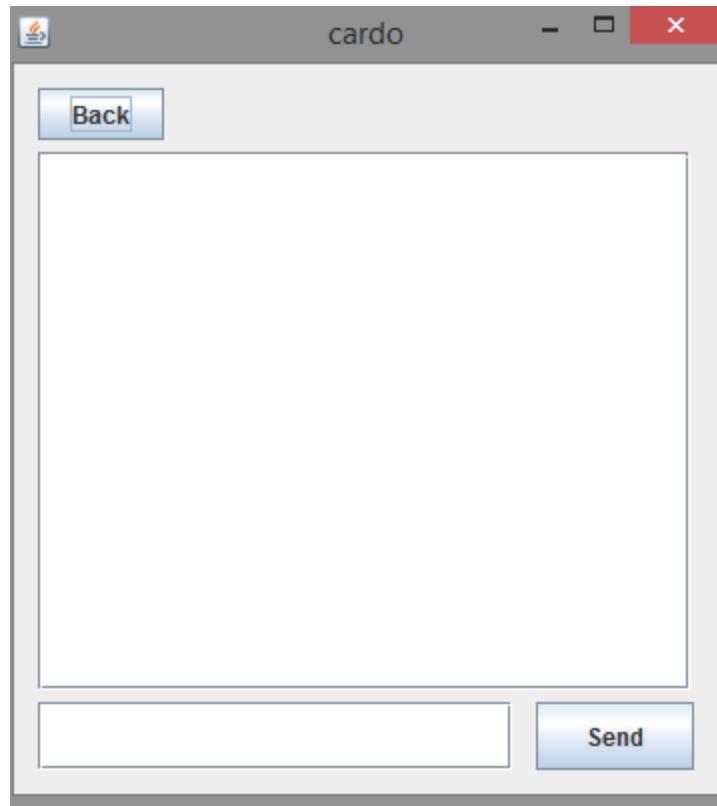
Keterangan : Jika kita ingin menghapus list teman kita dapat menekan tombol delete.



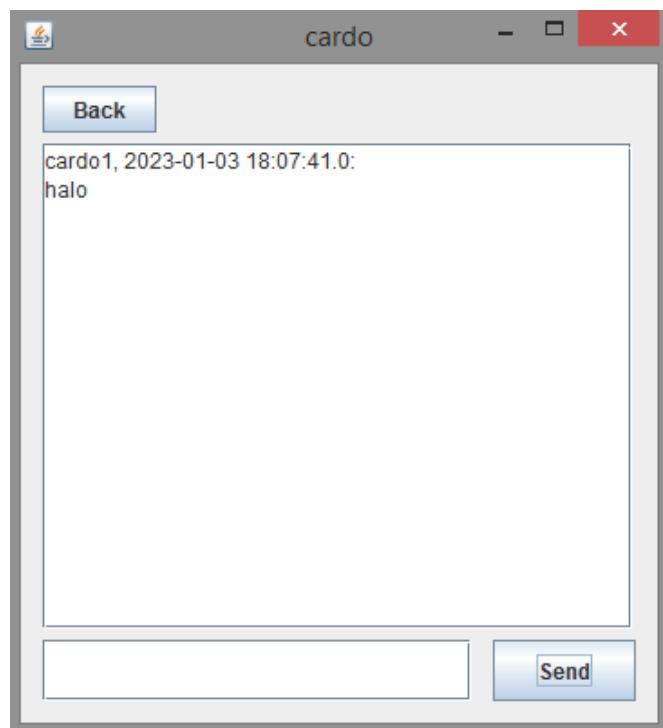
3.2.7. Chat Window



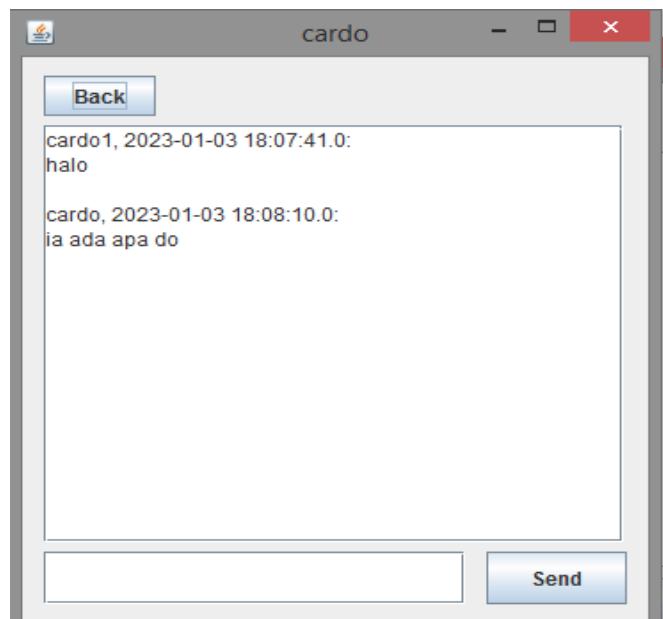
Keterangan : Kemudian jika kita ingin melakukan chat maka kita akan mengklik nama teman yang ada di list dan mengklik tombol open, maka kita akan masuk kedalam halaman ruangan untuk chatnya.



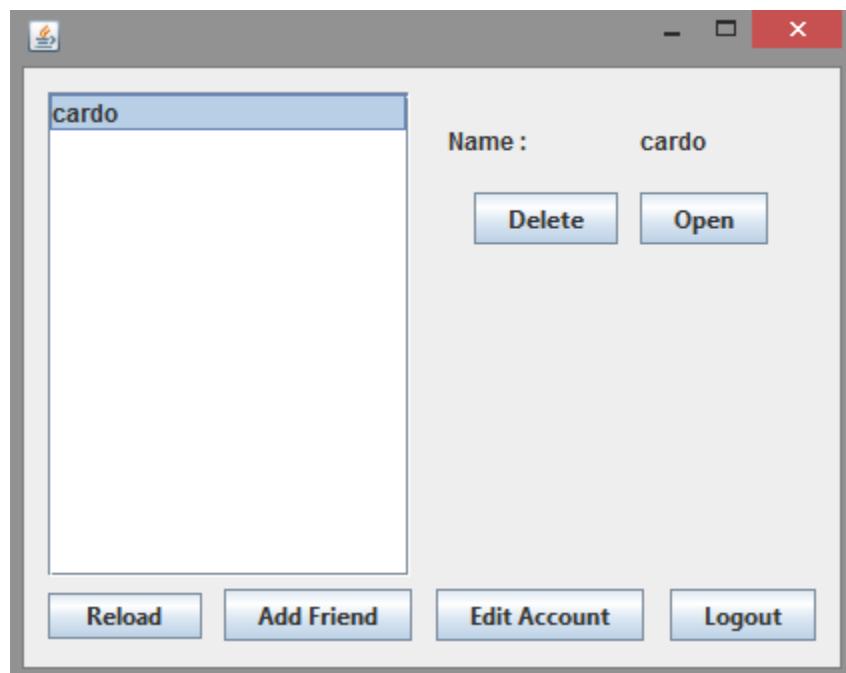
Keterangan : Di Dalam ruangan chat ini kita dapat mengirim pesan dengan teman yang sudah kita tambahkan tadi dan kita mengirim chat tersebut dengan menginputkan kata kata kedalam text dibawah dan menekan tombol send.



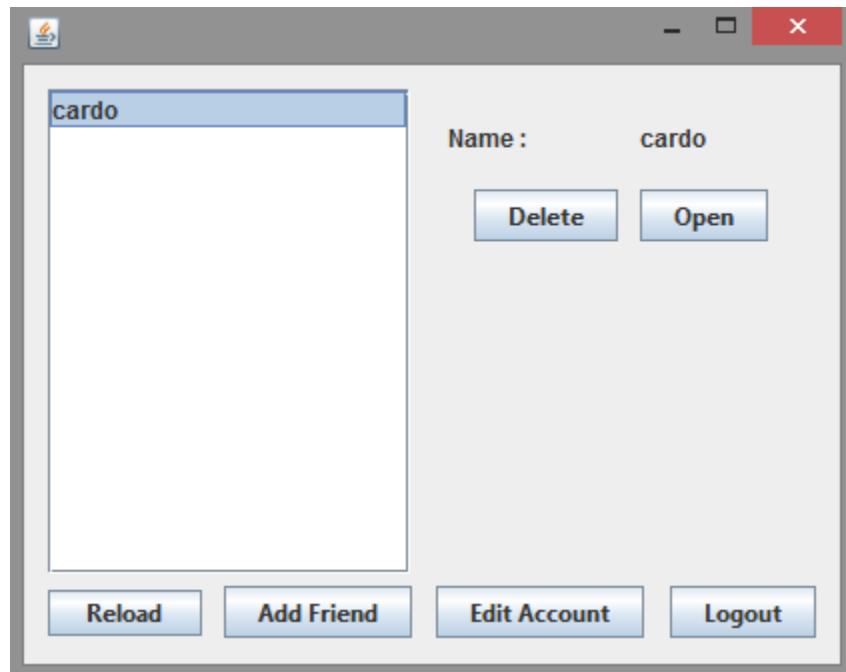
Keterangan : Dan jika teman kita membalas pesan tersebut maka akan muncul juga balasan nya di dalam ruangan chat tersebut.



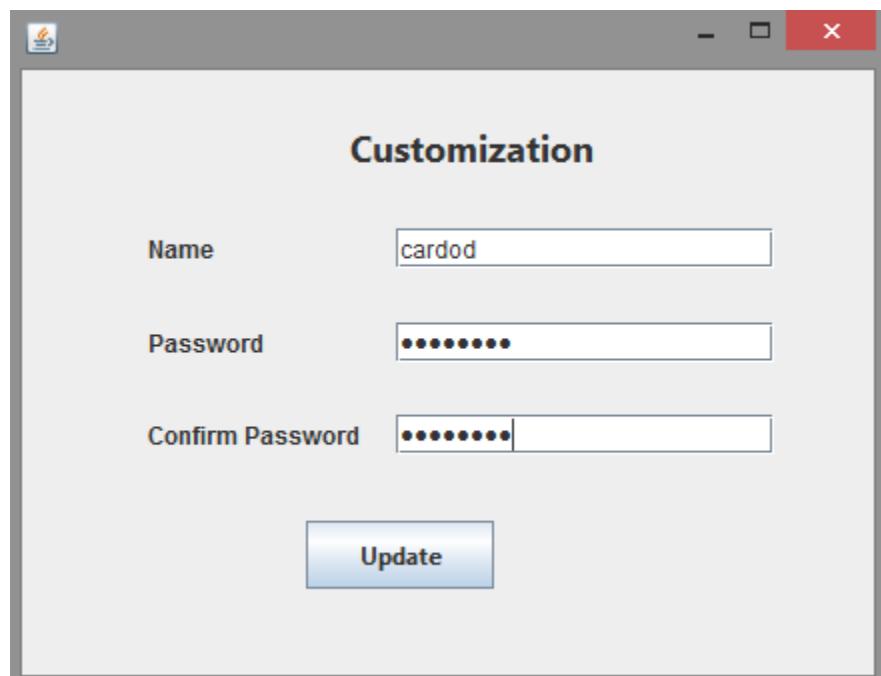
Keterangan : Selanjutnya ketika kita ingin kembali ke halaman chat kita hanya perlu menekan tombol back yang berada diatas.



3.2.8. Customization

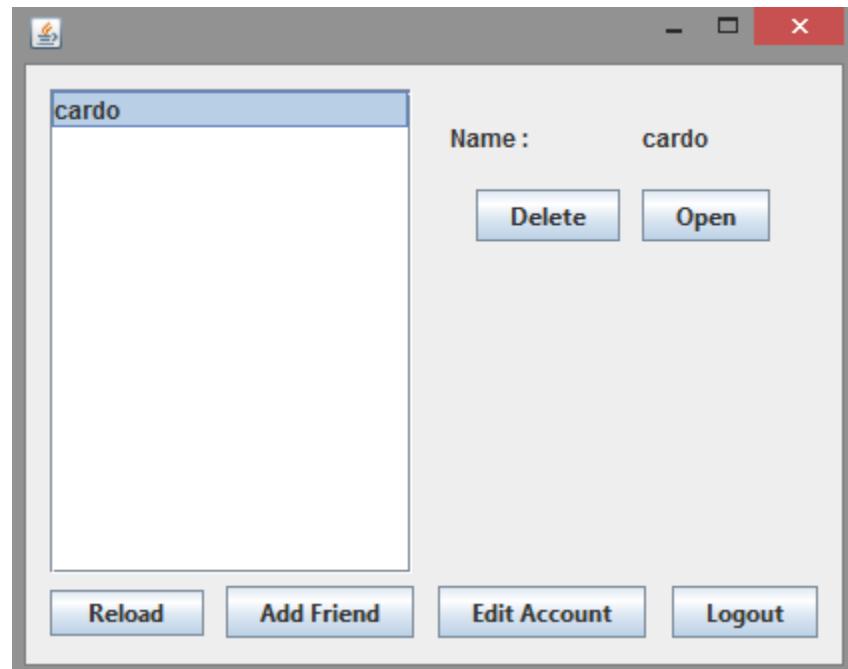


Keterangan : Jika ingin mengubah data user, maka dapat mengklik tombol Edit Account di halaman Chat List.



Keterangan : Ketika kita sudah mengklik tombol Edit Account maka kita akan langsung masuk ke dalam halaman customization, lalu kita disuruh menginput nama, password dan confirm password baru dan setelah semua di input kita mengklik tombol update. Setelah kita mengklik tombol update data tersebut akan masuk kedalam database dan juga kita akan kembali ke dalam halaman login.

3.2.9. Logout



Keterangan : Ketika kita ingin keluar dari aplikasi maka kita hanya perlu mengklik tombol Logout, maka kita akan keluar dari aplikasi dan kembali ke halaman login.

Daftar Pustaka

Tim Dosen Pemrograman Berorientasi Objek, MVC and GUI. Diambil pada tanggal 2 Januari 2023.