

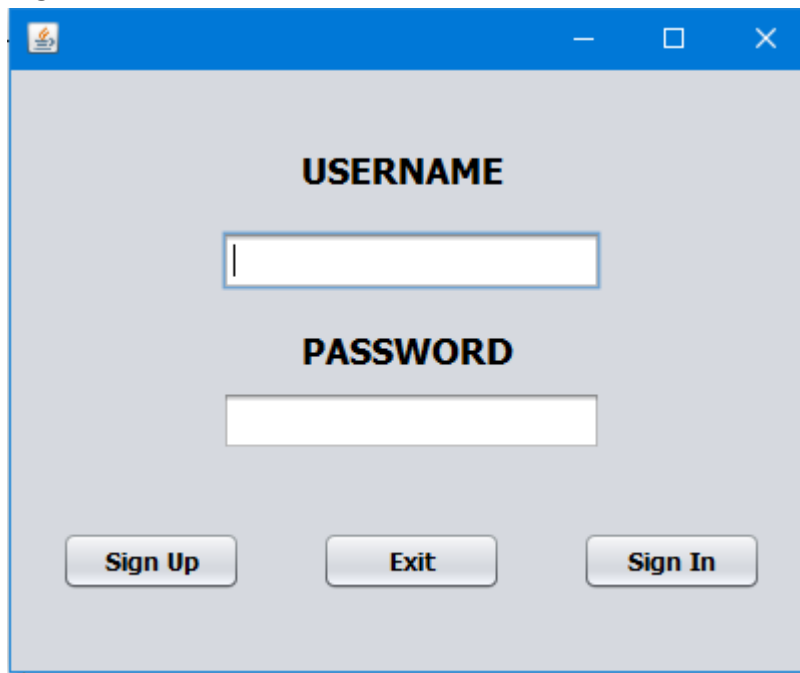


Nama : Muhammad Rizki Amrullah

Kelas : XI RPL 4

Absen : 24

Login View



Pada menu login aplikasi ini, terdapat opsi untuk mendaftar yaitu Sign Up, opsi masuk yaitu Sign in dan keluar yaitu exit. Jika kita memilih opsi register maka apa yang kita inputkan akan dimasukkan ke database. Lalu jika kita memilih opsi sign-in maka akan menjalankan validasi untuk mengecek apakah username dan password tsb sudah terdaftar di database atau belum. Dan jika kita memilih opsi exit maka aplikasi akan keluar. Berikut adalah script yang digunakan.

```
private void btnSignUpActionPerformed(java.awt.event.ActionEvent evt) {  
    String username = txtNama1.getText();  
    String password = txtPass.getText();  
  
    try {  
        try (Statement statement = (Statement) file_koneksi.GetConnection().createStatement()) {  
            statement.executeUpdate("INSERT INTO `tb_akun` (`username`, `password`) VALUES ('"+username+"', '"+password+"')");  
        }  
        JOptionPane.showMessageDialog(null, "Selamat! Anda Berhasil Sign Up!");  
    } catch (Exception e) {  
        JOptionPane.showMessageDialog(null, "Mohon maaf, ulangi lagi prosedur");  
    }  
}  
  
private void btnSignInActionPerformed(java.awt.event.ActionEvent evt) {  
    Connection connection;  
    PreparedStatement ps;  
    try {  
        connection = (Connection) DriverManager.getConnection(  
            "jdbc:mysql://localhost:3306/listrikdb?zeroDateTimeBehavior=convertToNull", "root", "");  
        ps = connection.prepareStatement("SELECT * FROM `tb_akun` WHERE username =? AND password =?");  
        ps.setString(1, txtNama1.getText());  
        ps.setString(2, txtPass.getText());  
        ResultSet result = ps.executeQuery();  
        if (result.next()) {  
            new MenuUtama().show();  
            this.dispose();  
        }  
        else {  
            JOptionPane.showMessageDialog(rootPane, "Salah!");  
            txtPass.setText("");  
            txtNama1.requestFocus();  
        }  
    }  
}
```

```

    } catch (Exception e) {
        JOptionPane.showMessageDialog(rootPane, "gagal");
    }
}

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

```

Menu Utama



Menu ini merupakan halaman yang menampilkan menu apa saja yang disediakan pada pembayaran listrik. Disini ada 5 button untuk menampilkan data. Berikut script yang digunakan.

```
private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    Transaksi obTran = new Transaksi();  
    obTran.show();  
}
```

```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    Petugas obpetgas = new Petugas();  
    obpetgas.show();  
}
```

```
private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    Daya obDaya = new Daya();  
    obDaya.show();  
}
```

```
private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    Pelanggan obPel = new Pelanggan();  
    obPel.show();  
}
```

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    dispose();  
}
```

Menu Daya

—

□

×

Daya

Kode Daya

Simpan

Daya

Hapus

Golongan Tarif

Keluar

kode_daya	daya	golongan_tarif
-----------	------	----------------

Menu Daya disini bertujuan untuk menambahkan Daya yang ingin diberikan. Berikut Script dari masing masing fungsi Button (Simpan, Hapus, Keluar).

```

public class Daya extends javax.swing.JFrame {
    /**
     * Creates new form Daya
     */
    public Connection conn;
    public Statement cn;
    public Daya() {
        initComponents();
        bersih();
        tampildata();
    }
    public void koneksi() {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/listrikdb","root","");
            cn=conn.createStatement();
        } catch (Exception e) {
            JOptionPane.showMessageDialog(null,"koneksi gagal");
            System.out.println(e.getMessage());
        }
    }

    public void bersih() {
        kd_daya.setText("");
        daya.setText("");
        gol_tarif.setText("");
    }

    public void tampildata() {
        DefaultTableModel tablenya = new DefaultTableModel();
        tablenya.addColumn("kode_daya");
        tablenya.addColumn("daya");
        tablenya.addColumn("golongan_tarif");
        try {
            koneksi();
            String sql = "Select*from dayatb";
            ResultSet rs =cn.executeQuery(sql);
            while (rs.next())
            {
                tablenya.addRow(new Object[]{rs.getString(1),rs.getString(2),rs.getString(3)});
            } jTable1.setModel(tablenya);
        }
        catch (Exception e) {
            JOptionPane.showMessageDialog(null,"anda salah");
        }
    }
}

```

```

private void btInputActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try{
        koneksi();
        String sql ="insert into dayatb values('"+kd_daya.getText()+"','"+daya.getText()+"','"+gol_tarif.getText()+"')";
        cn.executeUpdate(sql);
        conn.close();

        tampildata();
        bersih();
        JOptionPane.showMessageDialog(null, "Data Berhasil Disimpan");
    }
    catch (Exception e){
        JOptionPane.showMessageDialog(null, "Data Gagal Disimpan");
        System.out.println(e.getMessage());
    }
}

```

```

private void btDeleteActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try{
        koneksi();
        String sql ="delete from dayatb where kode_daya =('"+kd_daya.getText()+"')";
        cn.executeUpdate(sql);
        conn.close();

        tampildata();
        bersih();
        JOptionPane.showMessageDialog(null, "Data Berhasil Dihapus");
    }
    catch (Exception e){
        JOptionPane.showMessageDialog(null, "Data Gagal Dihapus");
        System.out.println(e.getMessage());
    }
}

```

```

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

```

Menu Pelanggan

Pelanggan

ID Pelanggan

Nama

Alamat

id_pel	nama	alamat
--------	------	--------

Fungsinya sama untuk menambahkan data. Berikut scriptnya.

```
public class Pelanggan extends javax.swing.JFrame {  
  
    /**  
     * Creates new form Pelanggan  
     */  
    public Connection conn;  
    public Statement cn;  
    public Pelanggan() {  
        initComponents();  
        bersih();  
        tampilkan();  
    }  
  
    public void koneksi() {  
        try {  
            Class.forName("com.mysql.jdbc.Driver");  
            conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/listrikdb","root","");  
            cn=conn.createStatement();  
        } catch (Exception e) {  
            JOptionPane.showMessageDialog(null,"koneksi gagal");  
            System.out.println(e.getMessage());  
        }  
    }  
}
```



```

public void bersih(){
    id_pel.setText("");
    nama_pel.setText("");
    alamat_pel.setText("");
}

public void tampildata(){
    DefaultTableModel tablenya = new DefaultTableModel();
    tablenya.addColumn("id_pel");
    tablenya.addColumn("nama");
    tablenya.addColumn("alamat");
    try{
        koneksi();
        String sql = "Select*from pelangganb";
        ResultSet rs =cn.executeQuery(sql);
        while (rs.next())
        {
            tablenya.addRow(new Object[]{rs.getString(1),rs.getString(2),rs.getString(3)});
        }jTable1.setModel(tablenya);

    }
    catch(Exception e){
        JOptionPane.showMessageDialog(null,"anda salah");
    }
}

```

```

private void btInputActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try{
        koneksi();
        String sql ="insert into pelangganb values ('"+id_pel.getText()+"', "
        + "'"+nama_pel.getText()+"', '"+alamat_pel.getText()+"')";
        cn.executeUpdate(sql);
        conn.close();


        tampildata();
        bersih();
        JOptionPane.showMessageDialog(null, "berhasil");
    }
    catch (Exception e){
        JOptionPane.showMessageDialog(null, "koneksi ne gagal");
        System.out.println(e.getMessage());
    }
}

```

```
private void btHapusActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try{  
        koneksi();  
        String sql ="delete from pelanggan where id_pel =('"+id_pel.getText()+"')";  
        cn.executeUpdate(sql);  
        conn.close();  
  
        tampildata();  
        bersih();  
        JOptionPane.showMessageDialog(null, "berhasil");  
    }  
    catch (Exception e){  
        JOptionPane.showMessageDialog(null, "koneksi gagal");  
        System.out.println(e.getMessage());  
    }  
}
```

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

Menu Petugas



—

□

×

Petugas

ID Petugas

Simpan

Nama

Hapus

Alamat

Tutup

id_petugas	nama_petugas	alamat
------------	--------------	--------

Fungsinya sama untuk menambahkan data. Berikut scriptnya.

```

- */
public class Petugas extends javax.swing.JFrame {

-   /**
-    * Creates new form Petugas
-    */
    public Connection conn;
    public Statement cn;
-   public Petugas() {
        initComponents();
        bersih();
        tampildata();
-   }

-   public void koneksi() {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/listrikdb","root","");
            cn=conn.createStatement();
        } catch (Exception e) {
            JOptionPane.showMessageDialog(null,"koneksi gagal");
            System.out.println(e.getMessage());
        }

-   }

    public void bersih() {
        id_petugas.setText("");
        nama_petugas.setText("");
        alamat_petugas.setText("");
    }

    public void tampildata() {
        DefaultTableModel tablenya = new DefaultTableModel();
        tablenya.addColumn("id_petugas");
        tablenya.addColumn("nama_petugas");
        tablenya.addColumn("alamat");
        try {
            koneksi();
            String sql = "Select*from petugastb";
            ResultSet rs =cn.executeQuery(sql);
            while (rs.next())
            {
                tablenya.addRow(new Object[]{rs.getString(1),rs.getString(2),rs.getString(3)});
            } jTable1.setModel(tablenya);

        } catch (Exception e) {
            JOptionPane.showMessageDialog(null,"anda salah");
        }
    }
}

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try{
        koneksi();
        String sql ="insert into petugastb values ('"+id_petugas.getText()+"', '"
        +nama_petugas.getText()+"', '"+alamat_petugas.getText()+"')";
        cn.executeUpdate(sql);
        conn.close();

        tampildata();
        bersih();
        JOptionPane.showMessageDialog(null, "berhasil!!");
    }
    catch (Exception e){
        JOptionPane.showMessageDialog(null, "gagal");
        System.out.println(e.getMessage());
    }
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try{
        koneksi();
        String sql ="delete from petugastb where id_petugas = ('"+id_petugas.getText()+"')";
        cn.executeUpdate(sql);
        conn.close();

        tampildata();
        bersih();
        JOptionPane.showMessageDialog(null, "berhasil!!");
    }
    catch (Exception e){
        JOptionPane.showMessageDialog(null, "koneksi gagal");
        System.out.println(e.getMessage());
    }
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    dispose();
}

```

Menu Transaksi

Transaksi pembayaran

No Pembayaran

ID Petugas

Simpan

Tanggal Pembayaran

Hapus

ID Pelanggan

Tutup

Kode Daya

Tagihan

Administrasi

Print

Total Bayar

Title 1	Title 2	Title 3	Title 4

Menu Transaksi disini fungsinya untuk memasukan data sebelumnya(tambah daya,tambah pelanggan,tambah petugas)menjadi satu.Data yang diinputkan harus sama dengan data sebelumnya dari masing masing(tambah daya,tambah pelanggan,tambah petugas).Berikut Scriptnya.

```
public class Transaksi extends javax.swing.JFrame {

    /**
     * Creates new form Transaksi
     */
    public Connection conn;
    public Statement cn;
    public Transaksi() {
        initComponents();
        bersih();
        tampildata();
    }

    public void koneksi() {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/listrikdb","root","");
            cn=conn.createStatement();
        } catch (Exception e) {
            JOptionPane.showMessageDialog(null, "koneksi gagal");
            System.out.println(e.getMessage());
        }
    }

    public void bersih() {
        jTextField1.setText("");
        jTextField2.setText("");
        jTextField3.setText("");
        jTextField4.setText("");
        jTextField5.setText("");
        jTextField6.setText("");
        jTextField7.setText("");
        jTextField8.setText("");
    }

    public void tampildata() {
        DefaultTableModel tablenya = new DefaultTableModel();
        tablenya.addColumn("no_pembayaran");
        tablenya.addColumn("id_petugas");
        tablenya.addColumn("tanggal_pembayaran");
        tablenya.addColumn("id_pel");
        tablenya.addColumn("kode_daya");
        tablenya.addColumn("tagihan");
        tablenya.addColumn("administrasi");
        tablenya.addColumn("total_bayar");
    }
}
```

```

try{
    koneksi();
    String sql = "Select*from pembayarantb";
    ResultSet rs =cn.executeQuery(sql);
    while (rs.next())
    {
        tablenya.addRow(new Object[]{
            rs.getString(1),rs.getString(2),rs.getString(3),
            rs.getString(4),rs.getString(5),rs.getString(6),rs.getString(7),rs.getString(8)});
    }jTable1.setModel(tablenya);

    }
    catch(Exception e){
        JOptionPane.showMessageDialog(null,"anda salah");
    }
}

```

```

private void jTextField8MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    int text6 = Integer.parseInt(jTextField6.getText());
    int text7 = Integer.parseInt(jTextField7.getText());
    int hasil = text6+text7;
    jTextField8.setText(String.valueOf(hasil));
}

```

```

private void jTextField8ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

```

```

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    int baris = jTable1.getSelectedRow();
    if (baris !=1) {
        jTextField1.setText(jTable1.getValueAt(baris, 0).toString());
        jTextField2.setText(jTable1.getValueAt(baris, 1).toString());
        jTextField3.setText(jTable1.getValueAt(baris, 2).toString());
        jTextField4.setText(jTable1.getValueAt(baris, 3).toString());
        jTextField5.setText(jTable1.getValueAt(baris, 4).toString());
        jTextField6.setText(jTable1.getValueAt(baris, 6).toString());
        jTextField7.setText(jTable1.getValueAt(baris, 6).toString());
        jTextField8.setText(jTable1.getValueAt(baris, 7).toString());
    }
}

```



```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try{
        koneksi();
        String sql ="insert into pembayarantb values('"+jTextField1.getText()+"', "
            + "'" +jTextField2.getText()+"', '"+jTextField3.getText()+"', "
            + "'" +jTextField4.getText()+"', '"+jTextField5.getText()+"', "
            + "'" +jTextField6.getText()+"', '"+jTextField7.getText()+"', '"+jTextField8.getText()+"')";

        cn.executeUpdate(sql);
        conn.close();

        tampildata();
        bersih();
        JOptionPane.showMessageDialog(null, "Data Berhasil Disimpan");
    }
    catch (Exception e){
        JOptionPane.showMessageDialog(null, "Data Gagal Disimpan");
        System.out.println(e.getMessage());
    }
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try{
        koneksi();
        String sql ="delete from pembayarantb where no_pembayaran = ('"+jTextField1.getText()+"')";
        cn.executeUpdate(sql);
        conn.close();

        tampildata();
        bersih();
        JOptionPane.showMessageDialog(null, "Data Berhasil Dihapus");
    }
    catch (Exception e){
        JOptionPane.showMessageDialog(null, "Data Gagal Dihapus");
        System.out.println(e.getMessage());
    }
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    dispose();
}

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    MessageFormat header = new MessageFormat("Data Pembayaran Listrik");
    MessageFormat footer = new MessageFormat("Page (0,number,integer) ");
    try{
        jTable1.print(JTable.PrintMode.FIT_WIDTH, header, footer,true, null, true, null);
    }catch(java.awt.print.PrinterException e){
        System.err.format("Cannot Print %s%n", e.getMessage());
    }
}

```

Ada juga file KoneksiDB dan file koneksi yang digunakan untuk mengkoneksikan aplikasi dengan database.

KoneksiDB

```
public class KoneksiDB {
    public static Connection setKoneksi() {
        String konString = "jdbc:mysql://localhost:3306/listrikdb";
        Connection koneksi = null;

        try{
            Class.forName("com.mysql.jdbc.Driver");
            koneksi = (Connection)
                DriverManager.getConnection(konString,"root","");
            System.out.println("Koneksi Berhasil");
        }catch (ClassNotFoundException | SQLException ex) {
            java.util.logging.Logger.getLogger(KoneksiDB.class.getName()).log(Level.SEVERE, null, ex);
            System.out.println("Koneksi Gagal");
        }
    }
    return koneksi;
}

public static int execute(String SQL) {
    int status = 0;
    Connection koneksi = setKoneksi();
    try{
        Statement st = koneksi.createStatement();
        status = st.executeUpdate(SQL);
    } catch (SQLException ex) {
        java.util.logging.Logger.getLogger(KoneksiDB.class.getName()).log(Level.SEVERE, null, ex);
    }
    return status;
}

public static ResultSet executeQuery(String SQL) {
    ResultSet rs = null;
    Connection koneksi = setKoneksi();
    try{
        Statement st = koneksi.createStatement();
        rs = st.executeQuery(SQL);
    }catch (SQLException ex) {
        java.util.logging.Logger.getLogger(KoneksiDB.class.getName()).log(Level.SEVERE, null, ex);
    }
    return rs;
}

static Object GetConnection() {
    throw new UnsupportedOperationException("Not supported"
        + "yet.");
}
}
```

File_koneksi

```
public class file_koneksi {
    private static Connection koneksi;
    public static Connection GetConnection() throws SQLException
    {
        if (koneksi == null) {
            Driver driver = new Driver();
            koneksi = (Connection) DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/listrikdb?zeroDateTimeBehavior=convertToNull", "root", "");
        }
        return koneksi;
    }
}
```