Login.java:

package aggregation;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JTextField;

import javax.swing.JButton;

import javax.swing.JPanel;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.UIManager;

import java.awt.BorderLayout;

import java.awt.Dimension;

import java.awt.Color;

import java.awt.Font;

import javax.swing.JPasswordField;

import javax.swing.JOptionPane;

import java.io.BufferedWriter;

import java.io.BufferedReader;

import java.net.URL;

import java.io.OutputStreamWriter;

import java.io.InputStreamReader;

import java.net.URLConnection;

import java.net.Proxy;

import java.net.InetSocketAddress;

public class Login extends JFrame

{

CustomPanel p1;

JLabel l1,l2;

JTextField tf1,tf2;

JButton b1,b2,b3;

Font f1;

public Login(){

super("Login Screen");

p1 = new CustomPanel();

p1.setTitle(" Login Screen");

p1.setLayout(null);

JPanel main = new JPanel();

main.setLayout(new BorderLayout());

f1 = new Font("Microsoft Sanserif",Font.BOLD,11);

JPanel pan1 = new JPanel();

l1 = new JLabel("Username");

l1.setForeground(Color.white);

l1.setFont(f1);

pan1.add(l1);

tf1 = new JTextField(15);

tf1.setFont(f1);

pan1.add(tf1);

JPanel pan2 = new JPanel();

l2 = new JLabel("Password");

l2.setForeground(Color.white);

l2.setFont(f1);

pan2.add(l2);

tf2 = new JPasswordField(15);

tf2.setFont(f1);

pan2.add(tf2);

JPanel pan3 = new JPanel();

b1 = new JButton("Login");

b1.setFont(f1);

pan3.add(b1);

b1.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent ae){

login();

}

});

b2 = new JButton("Register");

b2.setFont(f1);

pan3.add(b2);

b2.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent ae){

register();

}

});

b3 = new JButton("Reset");

b3.setFont(f1);

pan3.add(b3);

b3.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent ae){

tf1.setText("");

tf2.setText("");

}

});

main.setBackground(new Color(128, 128, 128));

pan1.setBackground(new Color(128, 128, 128));

pan2.setBackground(new Color(128, 128, 128));

pan3.setBackground(new Color(128, 128, 128));

main.add(pan1,BorderLayout.NORTH);

main.add(pan2,BorderLayout.CENTER);

main.add(pan3,BorderLayout.SOUTH);

main.setBounds(50,80,300,100);

p1.add(main);

getContentPane().add(p1,BorderLayout.CENTER);

}

public void register(){

Register reg = new Register(this);

reg.setVisible(true);

reg.setLocationRelativeTo(null);

reg.setSize(400,400);

}

public static void main(String a[])throws Exception{

UIManager.setLookAndFeel("com.sun.java.swing.plaf.nimbus.NimbusLookAndFeel");

Login login = new Login();

login.setVisible(true);

login.setSize(400,300);

login.setLocationRelativeTo(null);

login.setResizable(false);

}

public void clear(){

tf1.setText("");

tf2.setText("");

}

public void login(){

String user = tf1.getText();

String pass = tf2.getText();

if(user == null || user.trim().length() <= 0){

JOptionPane.showMessageDialog(this,"Username must be enter");

tf1.requestFocus();

return;

}

if(pass == null || pass.trim().length() <= 0){

JOptionPane.showMessageDialog(this,"Password must be enter");

tf2.requestFocus();

return;

}

try{

Proxy proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress("10.0.0.10",3128));

URL url = new URL("https://keyaggregatecryptoapp.appspot.com/Login");

URLConnection con = url.openConnection(proxy);

con.setDoOutput(true);

BufferedWriter bw = new BufferedWriter(new OutputStreamWriter(con.getOutputStream()));

bw.write("t1="+user+"&t2="+pass);

bw.flush();

BufferedReader br = new BufferedReader(new InputStreamReader(con.getInputStream()));

String response = br.readLine();

if(response.equals("success")){

setVisible(false);

UserScreen us = new UserScreen(user,this);

us.setVisible(true);

us.setLocationRelativeTo(null);

us.setSize(600,200);

}else{

JOptionPane.showMessageDialog(this,"Error in login");

}

}catch(Exception e){

e.printStackTrace();

}

}

}

ShareFile.java:

package aggregation;

import javax.swing.JFrame;

import javax.swing.JButton;

import javax.swing.JPanel;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.UIManager;

import java.awt.BorderLayout;

import java.awt.Dimension;

import java.awt.Color;

import java.awt.Font;

import javax.swing.JOptionPane;

import javax.swing.JFileChooser;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.BufferedWriter;

import java.io.BufferedReader;

import java.net.URL;

import java.io.OutputStreamWriter;

import java.io.InputStreamReader;

import java.net.URLConnection;

import java.net.Proxy;

import java.net.InetSocketAddress;

import javax.swing.JLabel;

import javax.swing.JComboBox;

import java.util.ArrayList;

public class ShareFile extends JFrame

{

JButton b1;

JLabel l1,l2;

JComboBox c1,c2;

JPanel p2;

CustomPanel p1;

Font f1;

String owner;

public ShareFile(String own){

super("Share File Screen");

owner = own;

p1 = new CustomPanel();

p1.setTitle(" Share File Screen");

p1.setLayout(null);

f1 = new Font("Microsoft Sanserif",Font.BOLD,11);

p2 = new JPanel();

p2.setLayout(null);

l1 = new JLabel("User Name");

l1.setBounds(100,20,120,30);

l1.setFont(f1);

p2.add(l1);

c1 = new JComboBox();

c1.setBounds(220,20,120,30);

c1.setFont(f1);

p2.add(c1);

l2 = new JLabel("File Name");

l2.setBounds(100,70,120,30);

l2.setFont(f1);

p2.add(l2);

c2 = new JComboBox();

c2.setBounds(220,70,120,30);

c2.setFont(f1);

p2.add(c2);

b1 = new JButton("Share File");

b1.setFont(f1);

b1.setBounds(150,120,120,30);

p2.add(b1);

b1.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent ae){

share();

}

});

p2.setBackground(Color.white);

p1.add(p2);

p2.setBounds(0,45,600,400);

getContentPane().add(p1,BorderLayout.CENTER);

}

public void getFiles(){

try{

Proxy proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress("10.0.0.10",3128));

URL url = new URL("https://keyaggregatecryptoapp.appspot.com/GetFiles");

URLConnection con = url.openConnection(proxy);

con.setDoOutput(true);

BufferedWriter bw = new BufferedWriter(new OutputStreamWriter(con.getOutputStream()));

bw.write("t1="+owner);

bw.flush();

BufferedReader br = new BufferedReader(new InputStreamReader(con.getInputStream()));

String response = null;

while((response = br.readLine())!=null){

String res[] = response.split(",");

for(String str : res){

String file[] = str.split("#");

c2.addItem(file[0]);

}

}

}catch(Exception e){

e.printStackTrace();

}

}

public void getUser(){

try{

Proxy proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress("10.0.0.10",3128));

URL url = new URL("https://keyaggregatecryptoapp.appspot.com/GetUsers");

URLConnection con = url.openConnection(proxy);

con.setDoOutput(true);

BufferedWriter bw = new BufferedWriter(new OutputStreamWriter(con.getOutputStream()));

bw.write("t1="+owner);

bw.flush();

BufferedReader br = new BufferedReader(new InputStreamReader(con.getInputStream()));

String response = null;

while((response = br.readLine())!=null){

String res[] = response.split(",");

for(String str : res){

c1.addItem(str);

}

}

}catch(Exception e){

e.printStackTrace();

}

}

public void share(){

try{

String user = c1.getSelectedItem().toString().trim();

String file = c2.getSelectedItem().toString().trim();

Proxy proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress("10.0.0.10",3128));

URL url = new URL("https://keyaggregatecryptoapp.appspot.com/AccessFile");

URLConnection con = url.openConnection(proxy);

con.setDoOutput(true);

BufferedWriter bw = new BufferedWriter(new OutputStreamWriter(con.getOutputStream()));

bw.write("t1="+owner+"&t2="+user+"&t3="+file);

bw.flush();

BufferedReader br = new BufferedReader(new InputStreamReader(con.getInputStream()));

String response = br.readLine();

if(response.equals("success")){

JOptionPane.showMessageDialog(this,"Aggregation keys sent to sharing user");

}else{

JOptionPane.showMessageDialog(this,"Error in aggregating key");

}

}catch(Exception e){

e.printStackTrace();

}

}

}