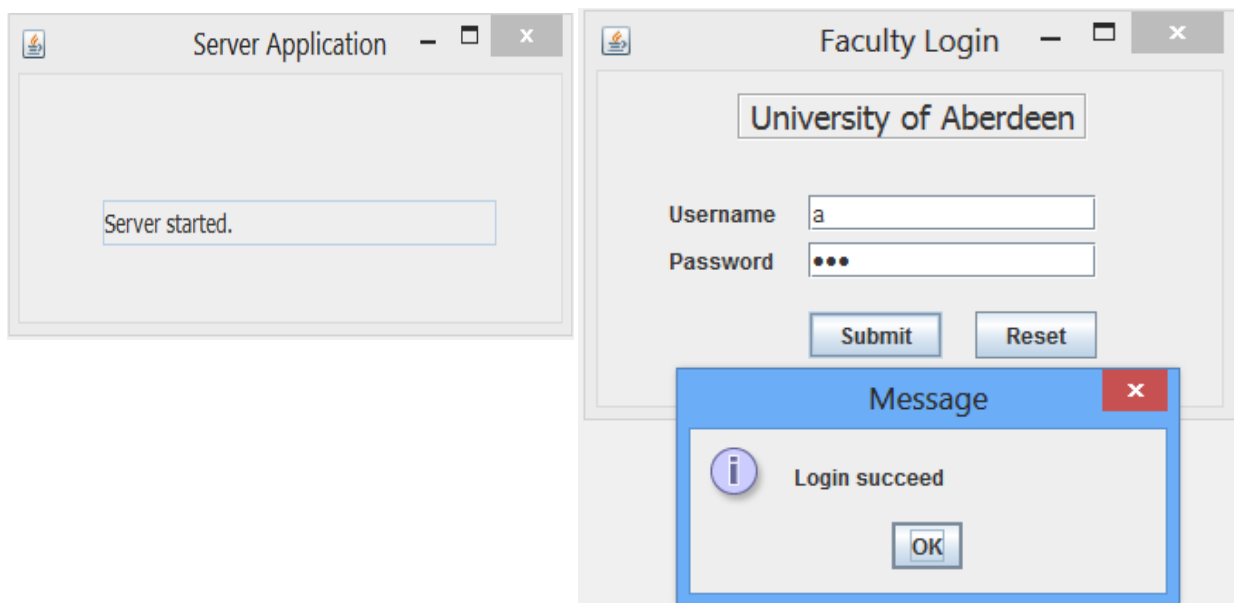


Sau khi thực hành xong bài này, học viên có khả năng :

- Xây dựng được ứng dụng Client – Server sử dụng RMI.
- Trình bày được các bước phát triển ứng dụng RMI trong Java

Xây dựng ứng dụng Client-Server có giao diện sau đây và thực hiện đăng nhập từ Client , phương thức đăng nhập sẽ được gọi đến Server để xử lý và trả về kết quả true/false cho biết thành công hay thất bại. Dữ liệu để kiểm tra đăng nhập được lưu trong tập tin data.txt ở phía Server .



Bước 1 . Mở ứng dụng NetBean , New Project | Java | Java Application , nhấn Next , trên hộp thoại “New Java Application ” điền các thông tin trong hộp thoại như sau :
Project Name : **ASM_Ngay_7_RMI** ; project Location : “nhập đường dẫn lưu ứng dụng”
và **Bỏ chọn** “Create Main Class” , nhấn Finish để thúc .

- 1.1 Nhấp phải chuột trên **<default package>** chọn New | Java Class , đặt tên lớp : **ILogin** , nhấn Finish để kết thúc .
- 1.2 Nhấp phải chuột trên **<default package>** chọn New | Java Class , đặt tên lớp : **LoginRemoteObject**, nhấn Finish để kết thúc

- 1.3 Nhấp phải chuột trên **<default package>** chọn New | JFrame Form , đặt tên lớp : **LoginServer**, nhấp Finish để kết thúc
- 1.4 Nhấp phải chuột trên **<default package>** chọn New | JFrame Form, đặt tên lớp : **LoginClient**, nhấp Finish để kết thúc
- 1.5 Nhấp phải chuột trên **ASM_Ngay_7_RMI**, chọn New | Other, trên hộp thoại New File , mục Categories chọn Other | Empty File , nhấp Next , đặt tên tập tin : **data.txt**, nhấp Finish để kết thúc . Tập tin này lưu tài khoản để kiểm tra đăng nhập.

*Nhập vào nội dung của **data.txt** 3 tài khoản như sau :*

a@123

b@456

c@789

Bước 2. Phát triển các đối tượng để gọi từ xa , interface ILogin dùng cho cả Client và Server

2.1 Phát triển interface **ILogin** , viết mã cho tập tin **ILogin.java** như sau :

```
import java.rmi.Remote;  
import java.rmi.RemoteException;  
public interface ILogin extends Remote {  
  
    public boolean check(String user, String pass) throws RemoteException;  
}
```

2.2 Phát triển đối tượng từ xa , viết mã cho tập tin **LoginRemoteObject.java** như sau :

```
import java.rmi.RemoteException;  
import java.rmi.server.UnicastRemoteObject;  
import java.util.*;  
import java.io.*;
```

```

public class LoginRemoteObject extends UnicastRemoteObject implements ILogin {
    //khai bao hashtable de luu tai khoan nguoi dung
    Hashtable<String,String> ht = new Hashtable<String,String>();
    public LoginRemoteObject() throws RemoteException {
        String str;
        try {
            //doc du lieu tu data.txt
            BufferedReader br = new BufferedReader(new FileReader("data.txt"));
            //doc tung dong va luu vao trong hashtable
            while ((str = br.readLine()) != null){
                //tach chuoi a@123 theo dau @
                String arr[] = str.split("@");
                ht.put(arr[0],arr[1]);
            }
        } catch (IOException ie){
            System.err.println("Exception occured - " + ie.getMessage());
        }
    }
    public boolean check(String user, String pass) throws RemoteException {
        //kiem tra user name
        if (ht.containsKey(user)){
            //kiem tra password.
            if (ht.get(user).equalsIgnoreCase(pass))
                return true;
        }
        return false;
    }
}

```

Bước 3. Phát triển ứng dụng LoginServer , thiết kế giao diện cho LoginServer.java :

STT	Loại Control	Tên thuộc tính/ Tên Sự Kiện	Giá trị thiết lập
1	JLabel	border	Titled Border
		Variable Name	lbMessage
2	JFrame	title	Server Application

Viết mã cho LoginServer.java :

```

import java.rmi.registry.LocateRegistry;
import java.rmi.*;

public class LoginServer extends javax.swing.JFrame {

    public LoginServer() {
        initComponents();
        try {
            LoginRemoteObject db = new LoginRemoteObject();
            LocateRegistry.createRegistry(1099);
            Naming.rebind("rmi://localhost:1099/LoginServer", db);
            lbMessage.setText("Server started.");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    ....

} //end class

```

Bước 4. Phát triển ứng dụng LoginClient , thiết kế giao diện cho LoginClient.java :

STT	Loại Control	Tên thuộc tính/ Tên Sự Kiện	Giá trị thiết lập
1	JLabel	text	University of Aberdeen
2	JLabel	text	User name
3	JLabel	text	Password
4	JPasswordField	text	
		Variable Name	txtPassword
5	JTextField	text	
		Variable Name	txtUserName
6	JButton	text	Submit
		Variable Name	btnSubmit
		Đăng ký sự kiện	ActionPerformed

7	JButton	text	Reset
		Variable Name	btnReset
		Đăng ký sự kiện :	ActionPerformed
8	JFrame	title	Faculty Login

Viết mã cho LoginClient.java :

```
import java.rmi.Naming;
import java.rmi.*;
import javax.swing.*;

public class LoginClient extends javax.swing.JFrame {

    ILogin stub;

    public LoginClient() {
        initComponents();
        try {
            stub = (ILogin) Naming.lookup("rmi://localhost:1099/LoginServer");
        } catch (Exception e) {
            System.out.println("Exception occurred - " + e.getStackTrace());
        }
    }

    private void btnResetActionPerformed(java.awt.event.ActionEvent evt) {
        txtUserName.setText("");
        txtPassword.setText("");
    }

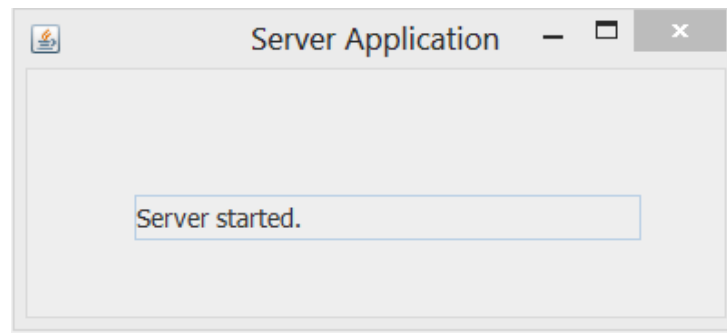
    private void btnSubmitActionPerformed(java.awt.event.ActionEvent evt) {
        try {
            if (stub.check(txtUserName.getText(), txtPassword.getText()))
                JOptionPane.showMessageDialog(this, "Login succeed");
            else
                JOptionPane.showMessageDialog(this, "Login failed");
        } catch (RemoteException e) {
            System.out.println("Exception occurred - " + e.getStackTrace());
        }
    }
}
```

.....

```
} //end class
```

Bước 5: Chạy ứng dụng và kiểm tra kết quả

5.1 . Chạy ứng dụng LoginServer.java



5.2 . Chạy ứng dụng LoginClient.java , nhập vào user name: a, password : 123 , nhấp nút Submit để xem kết quả.

