

# Client-Server 程序文档



# 同濟大學

项目组编号: 03

成 员: 倪雨婷 1352837 罗晓丹 1352866

王刚 1352873 叶坤宇 1352895 刘旭东 1352918

时 间: 2016-3-26

**文档信息：**

|      |  |    |           |
|------|--|----|-----------|
| 文档名称 | 项目程序文档   |    |           |
| 版本   | 1.0  |    |           |
| 文档编号 | 01   |    |           |
| 文档状态 | <input type="checkbox"/> 草稿 <input type="checkbox"/> 正在修改 <input checked="" type="checkbox"/> 正式发布 |    |           |
| 项目名称 | Client-Server 应用程序   |    |           |
| 撰写   | 王刚   | 日期 | 2016/3/26 |

修订历史记录：

| 日期        | 版本   | 说明   | 作者 |
|-----------|------|------|----|
| 2016/3/26 | V1.0 | 撰写初稿 | 王刚 |
|           |      |      |    |
|           |      |      |    |

|                |           |
|----------------|-----------|
| <b>一. 引言</b>   | <b>5</b>  |
| 1.1 程序编写目标:    | 5         |
| 1.2 项目具体成果要求:  | 5         |
| <b>二. 程序设计</b> | <b>6</b>  |
| (一) Client 部分: | 6         |
| 提供登录注册功能:      | 6         |
| “客户端界面”功能:     | 8         |
| (二) Server 部分: | 10        |
| 监听客户端发送的消息并转发: | 11        |
| (三) 测试代码部分:    | 12        |
| <b>三. 参考文献</b> | <b>12</b> |

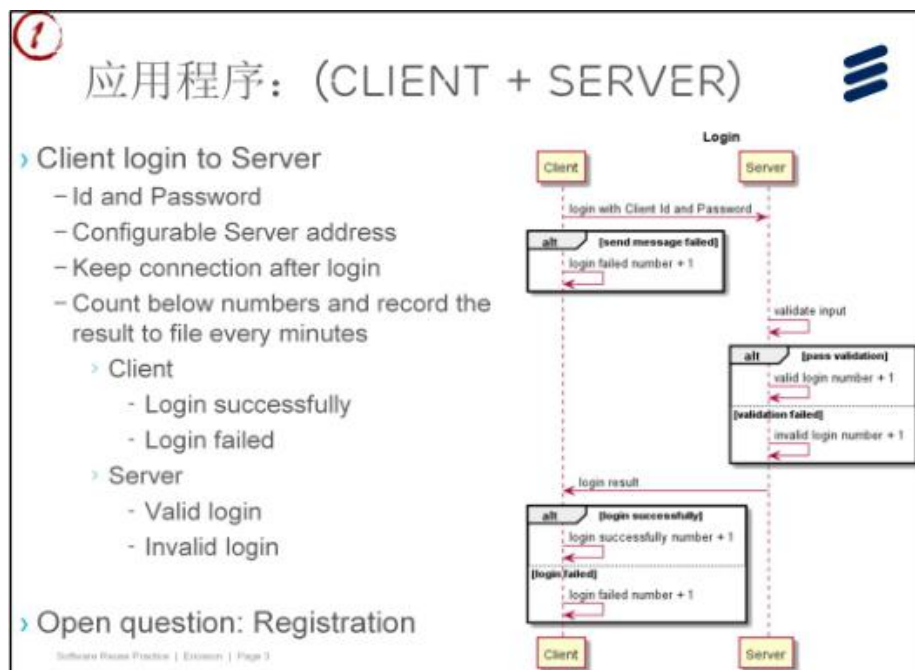
# 一. 引言

## 1.1 程序编写目标:

建立一个信息通信的应用，主要包括 Client 以及 Server 两部分的功能，具体来说将实现以下功能：

- ❖ 实现用户的登录和注册；
- ❖ 保持用户端和服务端的不断的 connection；
- ❖ 记录并显示客户端登录状态以及发送的信息；
- ❖ 服务端将任意客户端的信息转发到其余登录状态下的客户端；
- ❖ 拒绝客户端在一秒钟内发送超过 5 则信息到服务端；
- ❖ 当某一客户端发送 100 则总数的信息后自动退出登录，下次登录发送信息数将从 0 重新计数；

## 1.2 项目具体成果要求:





## 应用程序：(CLIENT + SERVER)



- › Client send messages to Server after login
  - Server validates the received messages and sends responses
    - › Ignore messages before Client login
    - › Do not allow Client to send more than 5(configurable) messages/second
      - Ignore further messages without response
    - › Do not allow Client to send more than 100(configurable) messages/login
      - Response "Redo login" for the 100<sup>th</sup> message
      - Ignore further messages before Client re-login
      - Keep the connection to Client
      - Not count previous ignored messages in this 100
    - › Response OK if validation passed
  - Client should automatically re-login after receiving "Redo login"
  - Count below numbers and record the result to file every minutes
    - › Client: Send messages number
    - › Server: Received/Ignored messages number

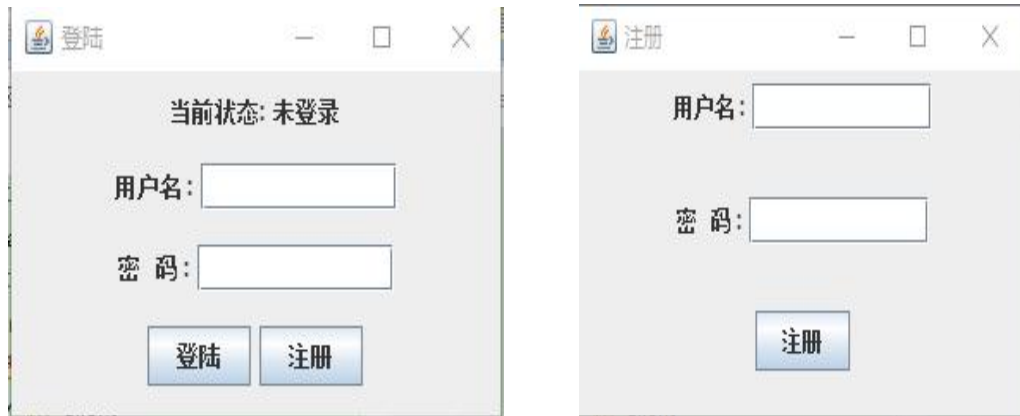
Software Reuse Practice | Ericsson | Page 4

## 二. 程序设计

本项目为基于 Windows 平台，采用 Java 语言编写，使用中间件 ActiveMQ 进行通信的搭建。主要设计有界面有“登录注册界面”和“客户端界面”。总体功能上分为客户端部分和服务端部分。主要功能是客户端向服务端发送信息，服务端接收信息并转发到其余处于成功登录状态的客户端上显示。

### (一) Client 部分：

提供登录注册功能：



首先在 Server 中的类 User 中有关于用户各自信息的定义：

这里有关的是：

```
public String UserName;
```

```
public String Password;
```

通过遍历用户姓名，进行与输入框中的用户名的匹配，根据匹配结果再进行密码校对，从而确定登录的具体情况并在之后显示出来。

用到的组件有：

```
JLabel regUsername;
JLabel regPassword;
JTextField regUsernameInput;
JPasswordField regPasswordInput;
JButton regBtn;
```

此处“登录界面”的布局由下列代码布置：

```
username = new JLabel("用户名 :");
password = new JLabel("密 码 :");
usernameInput = new JTextField(10);
passwordInput = new JPasswordField(10);
login = new JButton("登陆");
register = new JButton("注册");
```

具体说来，当用户点击 Button “登录”的时候，会触发  
loginPanel.add(login); 实现登录功能。

而点击“注册”按钮时候，会跳转到“注册界面”，此页面的布置与前面的登录界面类似。

登录执行之后会根据反馈结果进行相关的状态反馈：

• `boolean checkConnection(String userName)`

| 返回值   | 代表结果 |
|-------|------|
| true  | 在登录  |
| false | 未登录  |



登陆

当前状态: 已登录

用户名:

密 码:

“客户端界面” 功能：

实现界面如下：



客户端

登陆成功次数: 1      反馈结果: 登陆成功

登陆失败次数: 0      已发送消息数目: 0

消息显示框

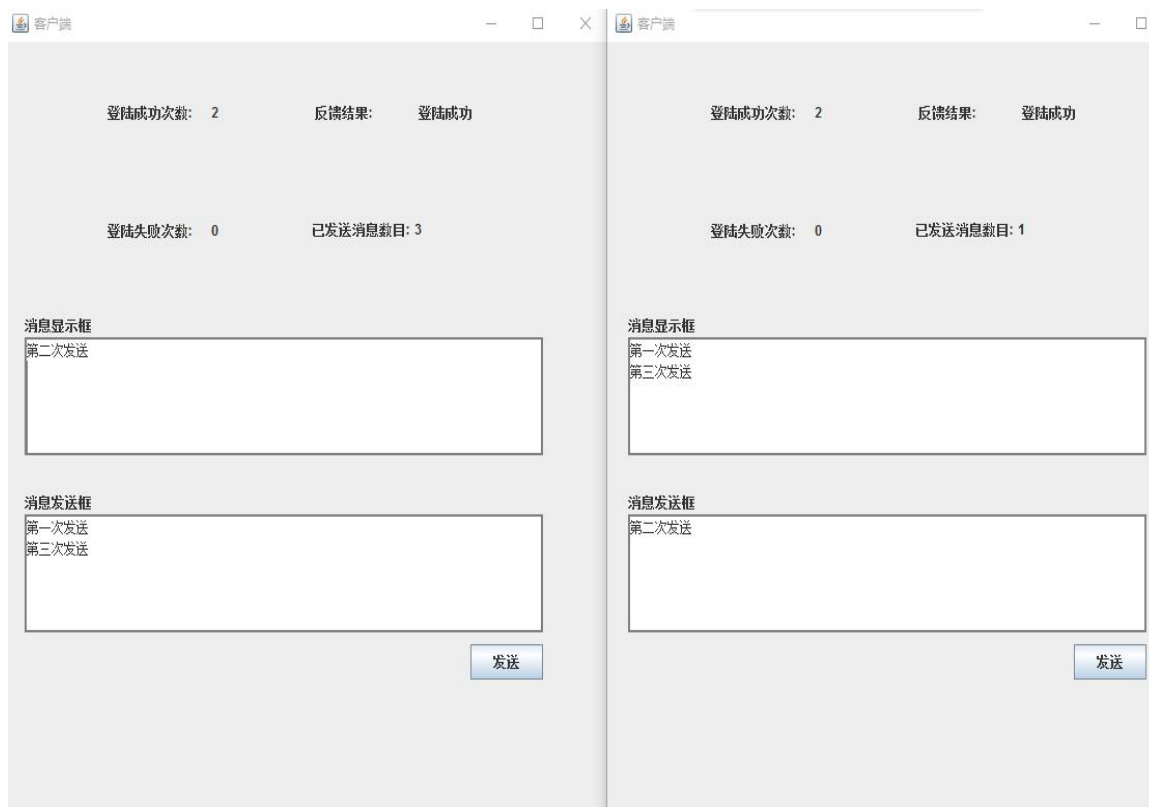
消息发送框



直观上可以看出，这里主要包含了“登录状态”显示、“登录成功次数”显示、“登录失败次数”的记录以及“已发送消息数目”的记录。

下方的两部分对话框类似的部件是用以执行“客户端向服务端发送消息”以及“服务端转发的消息显示”的两个窗口。其中，向服务端执行发送的功能由 Button “发送”控制。

部分功能演示结果如下：



位于“客户端”界面上面部分的四个信息，是由 Client 处代码对 Server 端所记录的数据进行调用，并进行分析解释后显示出来的结果，下方为举例说明：

Server 端: ..... (前面代码省略)

```
validLoginTime++;
theUser.isLogin = true;
return 200;
```

Client 端: `public void ListenMsg() throws JMSEException {.....}`

..... (前面代码省略)

```
if(txtMsg.getText().equals("200")){
    feedbackDisplay.setText("登陆成功");}
```

【注：Server 端返回结果 200 代表的是“登录成功”】

而消息发送功能【转发功能(在之后的 Server 端分析中将说明)】由以下代码实现：

消息发送功能：..... (前面代码省略)

```
sentButton.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent e) {
    // TODO Auto-generated method stub
    String str = msgSent.getText();
    msgNumberCount++;
msgNumberDisplay.setText(String.valueOf(msgNumberCount));
    //msgDisplay.setText(str); //显示到消息显示框
    status=false;
    sendMsg(str,"Ericsson",false);
}
});
.....
// 定义 sendMsg() 功能函数:
public void sendMsg(String msgText,String toipcName,boolean
isLogin){
    .....
}
```

这样便实现了将“消息发送框”中的信息文本读取并发送给到服务端的功能。

## (二) Server 部分:

限制用户不得在一秒时间内发送 5 则以上消息:

```
if (current-user.loginDate < 1000)
{
```

```

if (user.loginRequsetTime<5)
{
    System.out.println(user.loginRequsetTime);
    user.loginRequsetTime++;
    return true;
}}

```

设定在用户发送消息总数小于 100 以及超过 100 时的应对:

```

sender.sendMessageNum++;
System.out.println("Server 0");
if (sender.sendMessageNum < 100) {
    System.out.println("Server 1");
    return true;
}else return false;

```

//超过 100 后退出重登录, 并将发送消息数归 0

..... (前面代码省略)

```

sender.isLogin = false;
sender.sendMessageNum = 0;

```

## 监听客户端发送的消息并转发:

### 监听客户端:

```

public void ListenMsg(){
    ActiveMQConnectionFactory factory = new
ActiveMQConnectionFactory("tcp://localhost:61616");
    Connection connection;
    .....}

```

这里将对客户端传递到服务端的消息进行不断的监听。

### 对客户端的消息进行转发:

```

public void sendMsg(String msgText,String toipcName){
    .....
}

```

这部分代码是对服务端将自己收到的消息按照“根据客户端状态判断”, 将所收到消息转发到对应的其余客户端的功能实现。

(三) 测试代码部分:

.....

### 三. 参考文献

- ❖ 《软件工程导论（第五版）》 张海藩著 清华大学出版社
- ❖ 《实用软件文档写作》 肖钢著 清华大学出版社

[ISBN: 9787302103738]