

# Java 课程设计报告

学院： 信息学院

组员： 卢家伟（计科 1804，2018317210406）

罗天舒（计科 1803，2018317210329）

伏文学（计科 1803，2018317210328）

指导教师：余文君

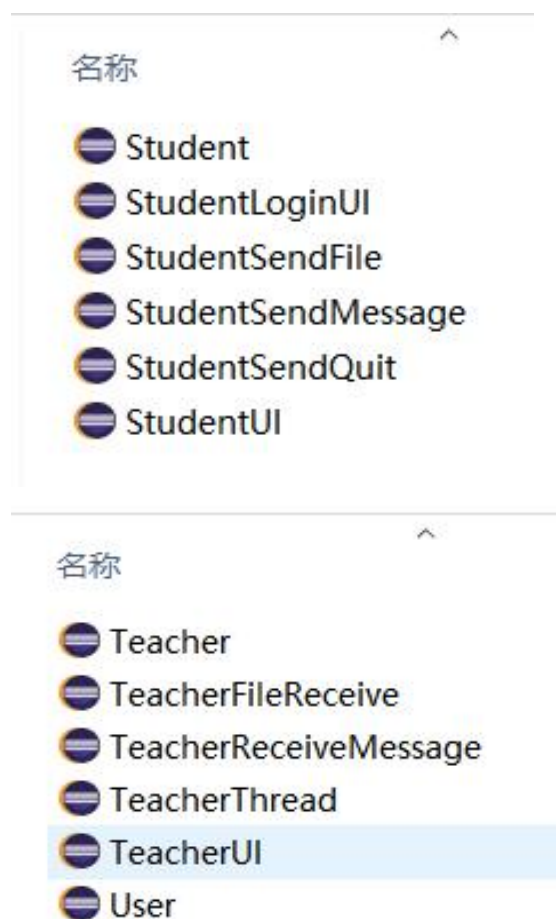
报告上交时间：2019 年 12 月 19 日

# 目录

一、概述	1
二、功能介绍	
1. 学生登录	2
2. 教师登录	2
3. 学生交流	2
4. 提交作业	2
5. 接收作业	2
6. 学生退出	2
三、具体实现方法	
1. 学生端（卢家伟）	3
2. 教师端（罗天舒）	5
3. 数据库（伏文学）	7
四、整体效果展示	
1. 教师端	8
2. 学生端	9
3. 学生交流区	10
4. 提交作业	11
五、个人总结	12

## 一、概述

本次课程设计完成了一个学生考勤系统。运用了 Threads 线程，socket 通信，文件输入输出流，以及 GUI 和数据库等相关知识。此系统实现了通信，文件传输等功能。老师首先登陆教师端，然后学生输入学号密码完成登录（学生信息存储在数据库中，通过与 mysql 数据库的连接实现验证），再将学生端与教师端进行连接。此时学生端为一个线程，而在教师端增加一个线程。此工程下 Student 和 Teacher 两个包下的 class 文件如下：



## 二、功能介绍

### 1. 学生登录:

学生登录时需已在弹出的窗口中输入学号和密码,并通过已连接的数据库验证是否存在该学生以及密码是否正确。若登录成功,则跳转到登录后界面。此时建立新的线程,并通过 Socket 与教师端建立联系,并记录学生姓名、登录时间,发送到教师端。

### 2. 教师登录:

单击 run,即可进入教师登录后界面,等待学生的连接。

### 3. 学生交流:

学生可通过界面中的文本框发送信息到教师端,教师端同时可以查看学生之间的交流。

### 4. 提交作业:

学生可通过 choose file 按钮浏览电脑内的文件,并通过 send 按钮提交作业。

### 5. 接收作业:

因为教师端已经写好文件的储存路径,因此教师无需另外接收作业,文件自动存入指定路径,并按照教师要求的方式命名作业名。

### 6. 学生退出:

学生点击 quiz 即可退出,此时结束本线程,终止与教师端的通信,并向教师端发送退出时间。

### 三、具体实现方法

#### 1. 学生端（卢家伟）

界面分为学生登录时与登录后，使用 Swing 类写出基本界面，学生登录成功后新建一个线程，应用 Socket 与老师端建立通信，实现登录与发送信息，学生登录后在教师端显示学生姓名和登录时间，因此要在登录时将姓名和登录时间等信息发送到教师端。

交作业时，应用了文件输入输出流，再通过 Socket 将文件传给教师端。聊天时，通过 Thread 新建一个线程，然后通过 Socket 实现通信。

以下为代码展示：

##### （1）学生登录后的界面代码：

```

29      //构造函数
30      public StudentUI(Socket s)
31      {
32          Font font = new Font("Times",Font.BOLD,16);
33          Font font1 = new Font("Times",Font.PLAIN,14);
34          this.s = s;
35          f = new JFrame("Student");
36          f.setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
37          f.setSize(500,500);
38          f.setLocation(700,100);
39          f.setLayout(new BorderLayout());
40
41          p1 = new Panel();
42          textArea = new JTextArea(15,30);
43          textArea.setFont(font);
44          textArea.setEditable(false);
45          scrollPane = new JScrollPane();
46          scrollPane.setViewportView(textArea);
47          scrollPane.setBounds(300,300,300,300);
48          scrollPane.setVerticalScrollBarPolicy(JScrollPane.VERTICAL_SCROLLBAR_ALWAYS);
49          p1.add(scrollPane);
50
51          centerPanel = new Panel();
52          sendMessageTextField = new JTextField(20);
53          sendMessageTextField.setFont(font);
54          sendMessageTextField.addKeyListener(new KeyAdapter()
55          {
56              @Override
57              public void keyPressed(KeyEvent e) {
58                  super.keyPressed(e);
59                  if (e.getKeyCode()==KeyEvent.VK_ENTER){
60                      System.out.println("Enter");
61                      Calendar calendar=Calendar.getInstance();
62                      SimpleDateFormat simpleDateFormat=new SimpleDateFormat("hh:mm:ss");
63                      textArea.append(simpleDateFormat.format(calendar.getTime())+"||"+sendMessageTextField.getText()+"\n");
64                      new StudentSendMessage(s,sendMessageTextField.getText()+"\n").start();
65                      sendMessageTextField.setText("");
66                  }
67              }
68          });

```

## (2) 学生登录界面代码:

```

26 private Socket student,
27 //构造函数
28 public StudentLoginUI(Socket s)
29 {
30     student = s;
31     Font font = new Font("Times",Font.BOLD,16);
32     //Frame
33     f = new JFrame("Student");
34     f.setLayout(new BorderLayout());
35     f.setSize(300, 250);
36     f.setLocation(600, 200);
37     f.setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
38     //Panel
39     p1 = new JPanel();
40     p1.setLayout(new GridLayout(2,2,30,15));
41     Dimension d = new Dimension(300,150);
42     p1.setPreferredSize(d);
43     //Label
44     username_l = new JLabel("username:");
45     password_l = new JLabel("password:");
46     username_l.setFont(font);
47     password_l.setFont(font);
48     //TextField
49     username_tf = new JTextField();
50     password_tf = new JTextField();
51     username_tf.setFont(font);
52     password_tf.setFont(font);
53     //username_tf监听器
54     username_tf.addKeyListener(new KeyAdapter()
55     {
56         @Override //检查
57         public void keyPressed(KeyEvent e)
58         {
59             super.keyPressed(e);
60             if (e.getKeyCode()==KeyEvent.VK_ENTER){
61                 username = username_tf.getText();
62                 password = password_tf.getText();
63                 System.out.println(username);
64                 System.out.println(password);
65                 new StudentSendMessage(student,username+":"+password+" ").start();
66             }
67         }
68     });

```

## (3) 学生提交作业:

```

11 public StudentSendFile(Socket socket, File file)
12 {
13     this.socket = socket;
14     this.file = file;
15 }
16 @Override
17 public void run()
18 {
19     try {
20         PrintStream printStream = new PrintStream(socket.getOutputStream());
21         FileInputStream fileInputStream = new FileInputStream(file);
22         printStream.println("@receive_file");
23         printStream.println(file.getName());
24         byte []bytes = new byte[10240];
25         int len = 0;
26         while ((len = fileInputStream.read(bytes)) != -1)
27         {
28             printStream.write(bytes);
29         }
30         fileInputStream.close();
31         printStream.close();
32         System.out.println("file send");
33         System.exit(0);
34     }
35     catch (FileNotFoundException e)
36     {
37         e.printStackTrace();
38     }
39     catch (IOException e)
40     {
41         e.printStackTrace();
42     }
43 }

```

#### (4) 聊天功能代码:

```
9 public class StudentSendMessage extends Thread
10 {
11     Socket student;
12     String message;
13
14     public StudentSendMessage(Socket socket,String message)
15     {
16         student = socket;
17         this.message = message;
18     }
19
20     @Override
21     public void run()
22     {
23         super.run();
24         OutputStream outputStream = null;
25         try
26         {
27             outputStream = student.getOutputStream();
28             BufferedWriter bufferedWriter = new BufferedWriter(new OutputStreamWriter(outputStream));
29             bufferedWriter.write("@receive_message\n");
30             bufferedWriter.flush();
31             bufferedWriter.write(message);
32             bufferedWriter.flush();
33             System.out.println("send message");
34         }
35         catch (IOException e)
36         {
37             e.printStackTrace();
38         }
39     }
40 }
```

## 2. 教师端（罗天舒）

教师端的界面不需要登录，所以用 Swing 写一个有两个文本框的界面。当教师登录时，即新建一个 Socket 通信，等待学生来连接。教师端可有多个线程，同时接收登录的学生信息（姓名，登录时间，退出时间）。Current number 即是在线的学生数。

接收消息时，教师端先新建一个线程，然后通过 Socket 与学生端的线程建立通信联系。

接收文件时，教师端新建一个线程与学生端通过 Socket 通信传输文件。教师端成功接收文件后按照指定路径存入。

以下为代码展示：

(1) 教师端界面代码：

```

10
17 public TeacherUI()
18 {
19     frame = new JFrame("Teacher");
20     frame.setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
21     frame.setSize(900,600);
22     frame.setLocation(100,100);
23     frame.setLayout(new BorderLayout());
24
25     Font f = new Font("Times",Font.PLAIN,16);
26
27     northPanel = new Panel();
28     northPanel.setLayout(new FlowLayout());
29     userTextArea = new JTextArea(20,20);
30     userTextArea.setFont(f);
31     userScrollPane = new JScrollPane();
32     userScrollPane.setViewportView(userTextArea);
33     userScrollPane.setBounds(300,300,300,300);
34     userScrollPane.setVerticalScrollBarPolicy(JScrollPane.VERTICAL_SCROLLBAR_ALWAYS);
35
36     communicationTextArea = new JTextArea(20,30);
37     communicationTextArea.setFont(f);
38     communicationScrollPane = new JScrollPane();
39     communicationScrollPane.setViewportView(communicationTextArea);
40     communicationScrollPane.setVerticalScrollBarPolicy(JScrollPane.VERTICAL_SCROLLBAR_ALWAYS);
41
42     northPanel.add(userScrollPane);
43     northPanel.add(communicationScrollPane);
44
45     southPanel = new Panel();
46     southPanel.setLayout(new FlowLayout());
47     currentNumberLabel = new JLabel("currentNumber:0");
48     Dimension d = new Dimension(200,100);
49     currentNumberLabel.setPreferredSize(d);
50     currentNumberLabel.setFont(f);
51     southPanel.add(currentNumberLabel);
52
53     frame.add(northPanel,BorderLayout.NORTH);
54     frame.add(southPanel,BorderLayout.SOUTH);
55     frame.setVisible(true);
56 }

```

(2) 接收文件：

```

6 public class TeacherFileReceive extends Thread
7 {
8     private Socket socket;
9     private DataInputStream dataInputStream;
10    private FileOutputStream fileOutputStream;
11
12    public TeacherFileReceive(Socket socket) throws IOException
13    {
14        this.socket = socket;
15        dataInputStream = new DataInputStream(socket.getInputStream());
16    }
17
18    @Override
19    public void run()
20    {
21        try
22        {
23            System.out.println("file save start");
24            String fileName = dataInputStream.readUTF();
25            System.out.println(fileName);
26            File file=new File("C:\\Users\\User\\Desktop\\homework\\作业"+fileName);
27            System.out.println(file.getAbsolutePath());
28            if (!file.exists())
29            {
30                file.createNewFile();
31                System.out.println("file create");
32            }
33            fileOutputStream = new FileOutputStream(file);
34            byte []bytes = new byte[10240];
35            int length = 0;
36            while ((length=dataInputStream.read(bytes,0,bytes.length))!=-1)
37            {
38                fileOutputStream.write(bytes,0,length);
39                fileOutputStream.flush();
40            }
41            fileOutputStream.close();
42            System.out.println("file saved");
43        }
44        catch (IOException e)
45        {
46
47        }
48    }
49 }

```



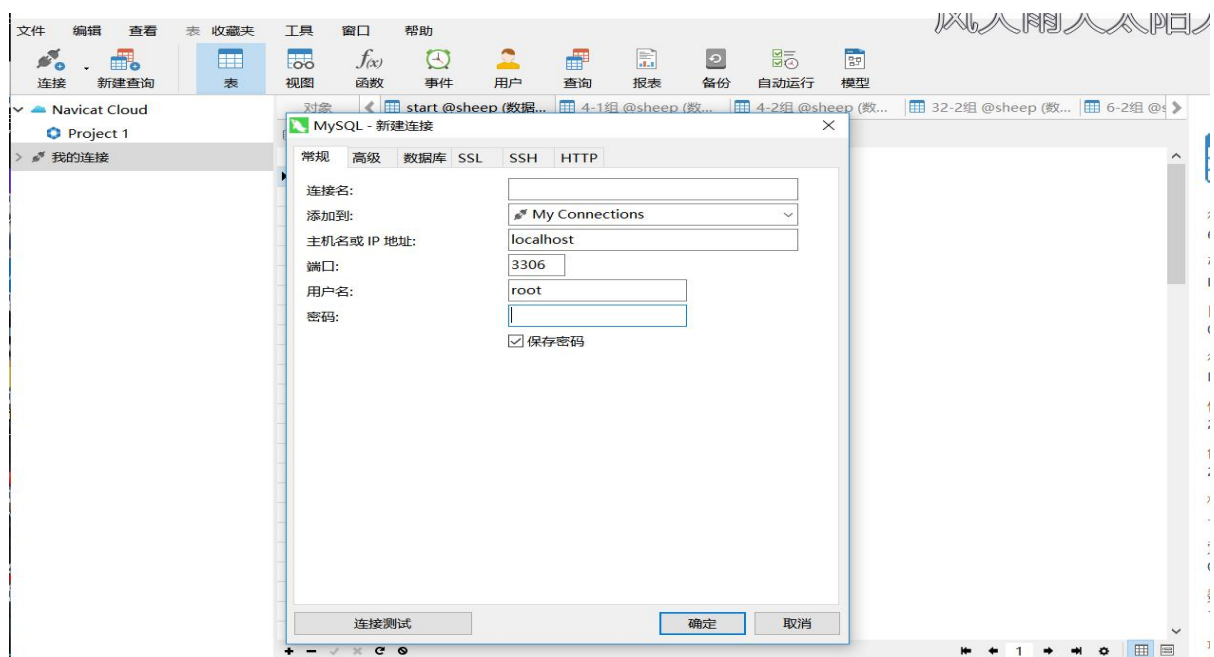
### (3) 接收消息

```
10 public class TeacherReceiveMessage extends Thread
11 {
12     private Socket socket;
13     private User user;
14     private JTextArea communicationTextArea;
15     private JTextArea userTextArea;
16     private JLabel currentNumberLabel;
17     private static String filePath="C:\\Users\\User\\Desktop\\homework\\作业";
18
19     public TeacherReceiveMessage(Socket socket, User user, JTextArea communicationTextArea, JTextArea userTextArea, JLabel label){
20         this.socket = socket;
21         this.user = user;
22         this.communicationTextArea = communicationTextArea;
23         this.userTextArea = userTextArea;
24         currentNumberLabel = label;
25     }
26     @Override
27     public void run()
28     {
29         try
30         {
31             InputStream inputStream = socket.getInputStream();
32             BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(inputStream));
33             String message=bufferedReader.readLine();
34             while (!message.equals("@quit"))
35             {
36                 if (message.equals("@receive_file"))
37                 {
38                     String fileName=bufferedReader.readLine();
39                     File file=new File(filePath+fileName);
40                     System.out.println(file.getAbsolutePath());
41                     if (!file.exists())
42                     {
43                         file.createNewFile();
44                     }
45                     FileOutputStream fileOutputStream=new FileOutputStream(file,true);
46                     int len = 0;
47                     byte []bytes=new byte[10240];
```

### 3. 连接数据库（伏文学）

通过导入已有的数据库，实现学生端与老师端的连接，方便老师检查班级同学出勤情况。

以下为代码展示：

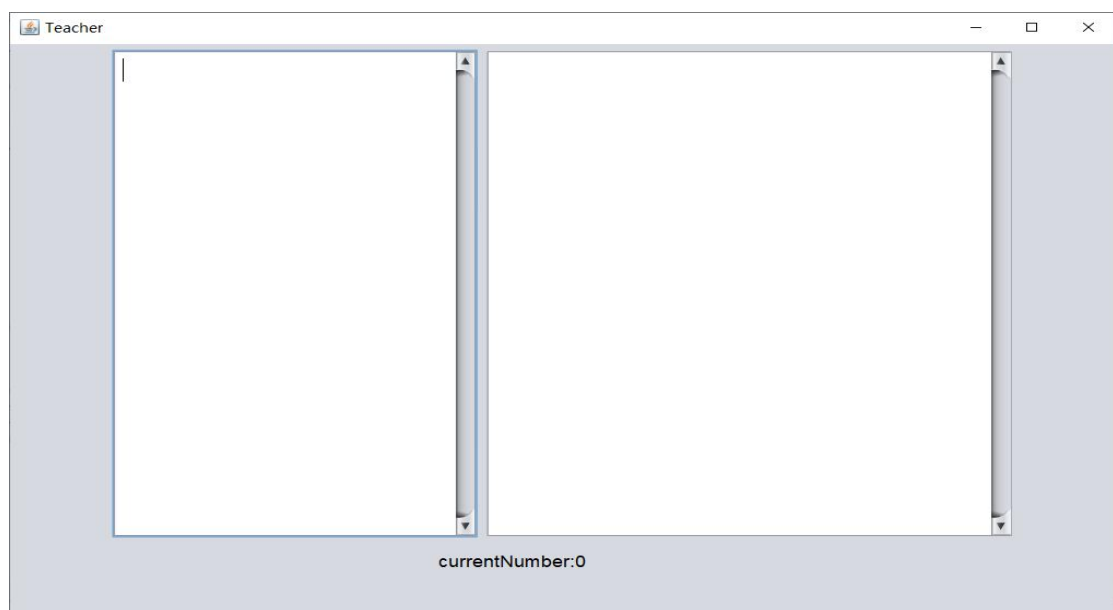


```
2
3 import com.sun.org.apache.xpath.internal.WhitespaceStrippingElementMatcher;
4 import java.sql.*;
5 import java.util.List;
6 import java.util.Map;
7 import java.util.Set;
8
9 public class Test {
10
11
12 public static void main(String[] args) throws ClassNotFoundException, SQLException {
13     String username="root";
14     String password="fwx6666";
15
16     Class.forName("com.mysql.jdbc.Driver");
17     Connection connection= DriverManager.getConnection("jdbc:mysql://localhost:3306/library?ServerTimezone=Asia/Shanghai",username,password);
18
19     Statement statement=connection.createStatement();
20     ResultSet resultSet =statement.executeQuery("select * from user");
21
22     while (resultSet.next()){
23         String username1=resultSet.getString("username");
24         String password1=resultSet.getString("password");
25         String email1=resultSet.getString("email");
26         System.out.println(username1+" "+password1+" "+email1);
27     }
28
29 }
```

## 四、整体效果展示

### 1. 教师端

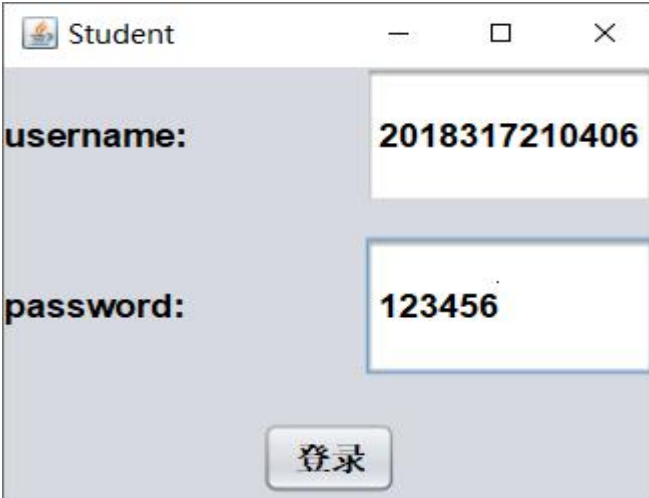
老师启动教师端，弹出界面。（currentNumber: 0 意为此时在线学生数为 0）



## 2. 学生端

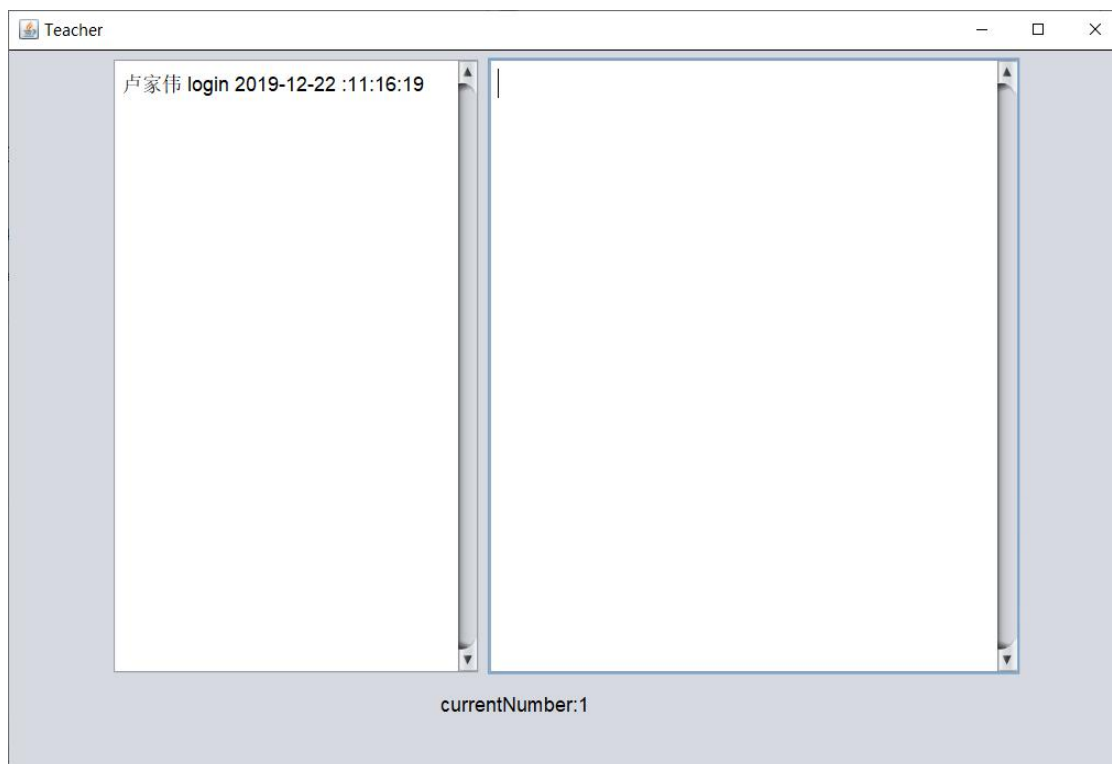
学生在界面中输入学号和密码，验证通过即登陆成功。

登录界面如下：



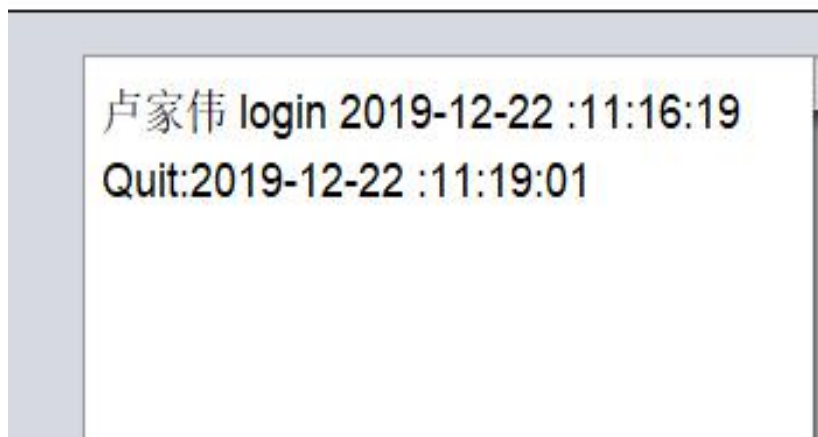
A screenshot of a Java Swing window titled "Student". The window has a light blue background. On the left, there are labels "username:" and "password:" in bold black text. To the right of "username:" is a text input field containing the number "2018317210406". To the right of "password:" is a text input field containing the number "123456". Below these fields is a button with the text "登录" (Login) in Chinese.

登录后开始一个新线程而教师端增加一个线程。教师端显示如下：



（此时 currentNumber:1 代表在线人数为 1）

学生可单击 quiz 选择退出界面，退出后教师端显示如下：

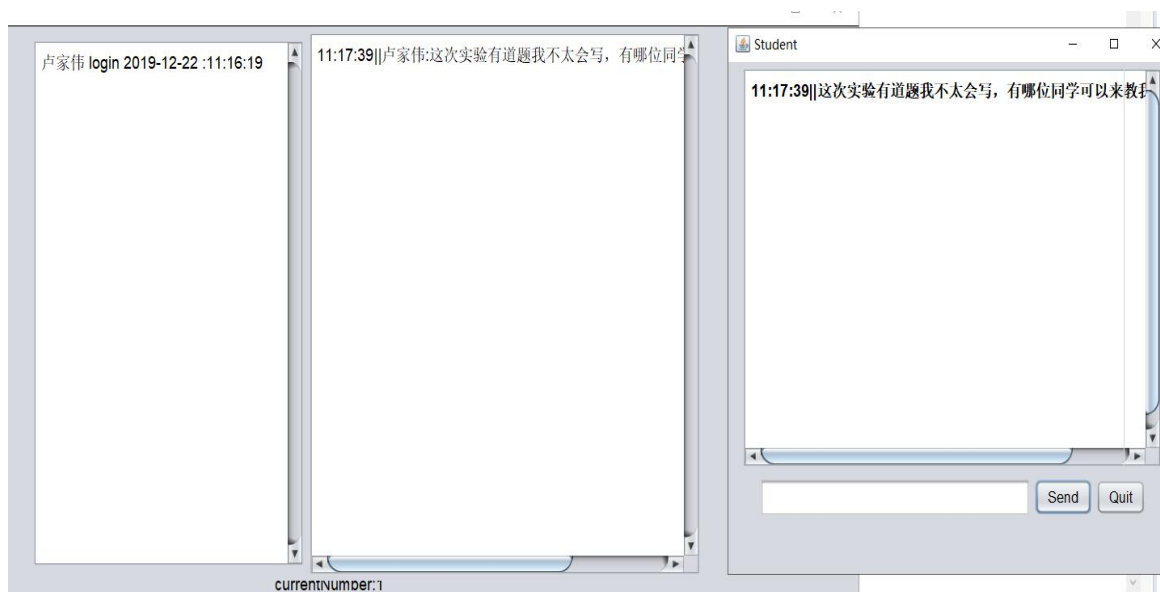


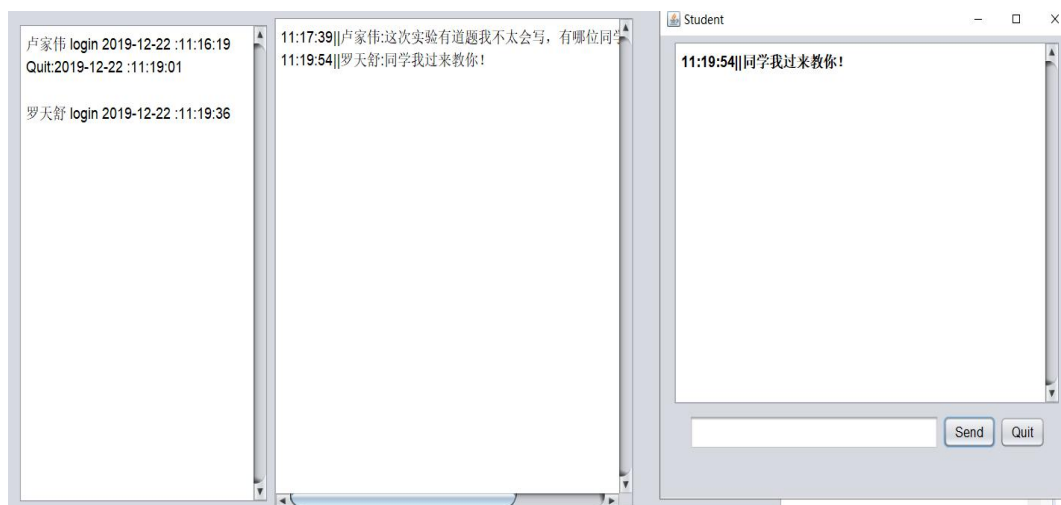
此时为未提交作业时的退出情况，教师可以看到登陆时间与退出时间。

### 3. 学生交流区

学生可通过界面中的文本框发送信息。信息将发送到教师端。教师可查看学生发来的信息。

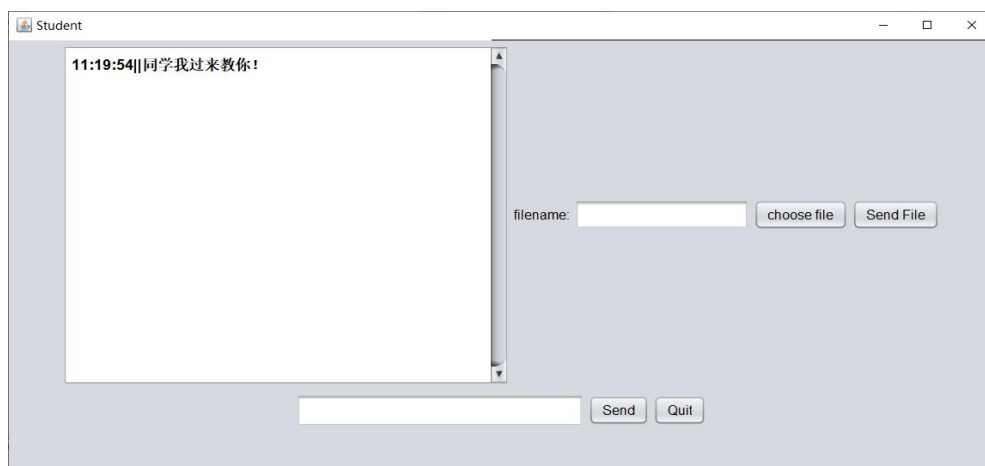
效果展示如下：



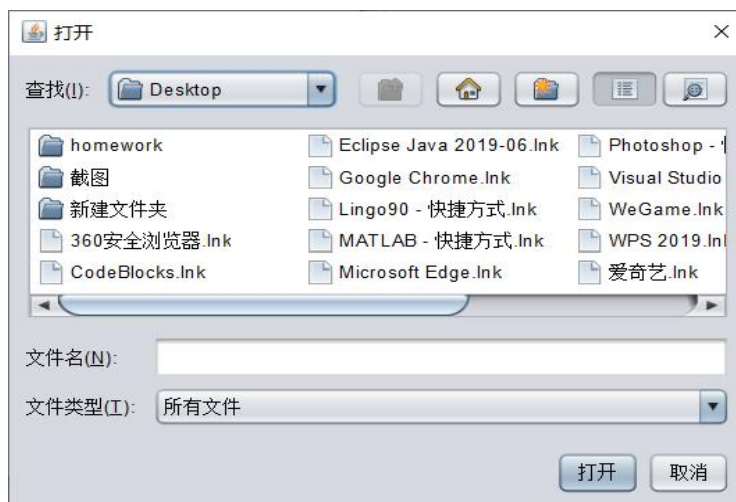


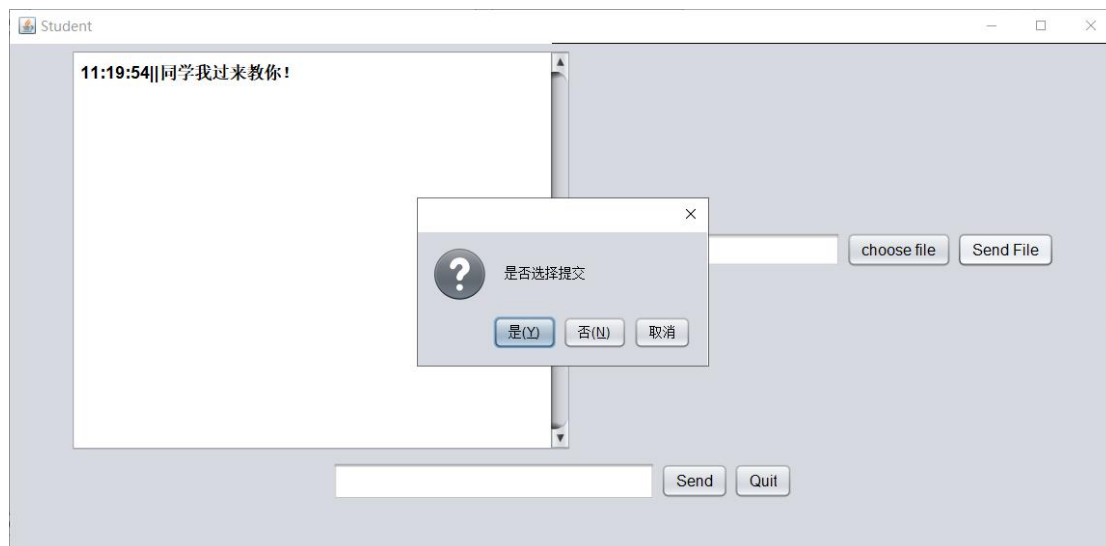
#### 4. 提交作业

学生在界面内可选择提交作业

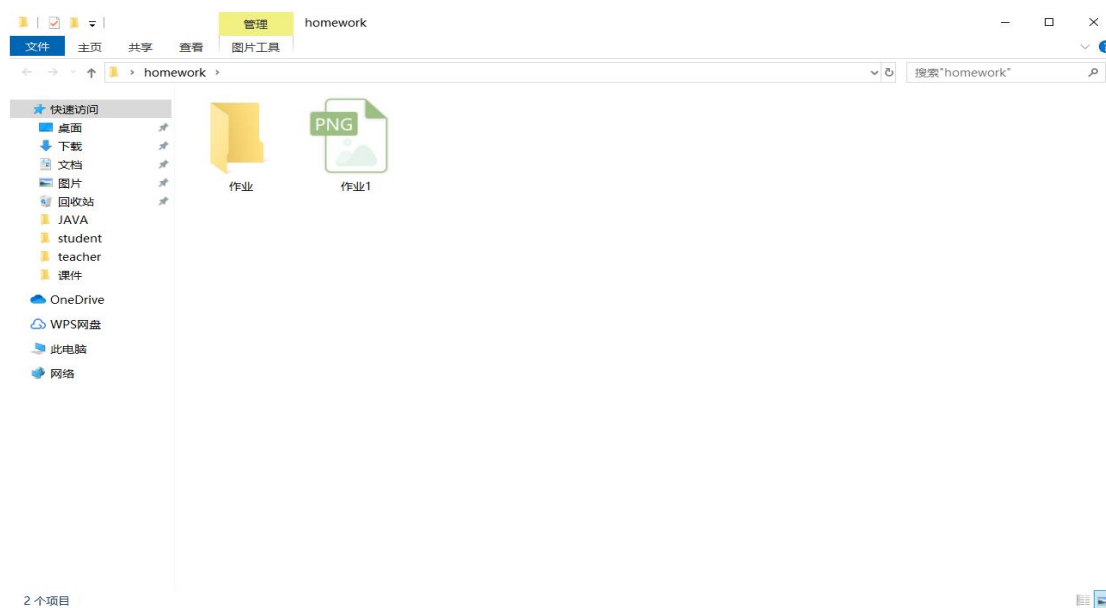


可在电脑内选择文件:





提交成功后，通过教师端存入预设好的路径中。老师不必逐一保存作业。



## 五、个人总结

卢家伟：通过这次课程设计，加深了对 Thread 以及 socket 的理解。本次课程设计从小组成员确定到完成全部设计，共历时三周。期间遇到过无数个 bug，通过百度和询问同学解决。我认为通过本次课程设计，增强了我的心理承受能力和抗压能力，能更好的应对以后的各种课程。

罗天舒：本次课程设计一开始时很不自信，认为这是一个很有挑战性的任务。在详细查阅了各种资料，阅读了各种博客后，虽然对本实验所涉及的知识点有了大概的了解，但仍然不知道如何着手去做。后来经过同学的指导，发现了更为简洁的实现本次实验的方法。所以要感谢队友们。

伏文学：本次课程设计的实施前期大量的寻找资料，翻书。虽然有时候走了一些弯路，但终归学到了很多，在改代码的时候也注意到了很多以前看书没有注意到的细节。通过本次实验，我对 java 中多线程和 Socket 通信加深了了解。更锻炼了寻找信息的能力。