

Java 程序设计

(大作业实验报告)

项目名称：

基于 TCP、UDP 通讯协议的聊天软件

项目组长：牛迪

项目成员：姜庭宇、边中阳

所在班级：

16 软工 A3 (牛迪、姜庭宇)

16 软工 A2 (边中阳)

完成日期： 2018 年 6 月 9 日

项目简介

(用 300 字左右简要介绍下：本系统实现了哪些功能、覆盖了哪些知识点、小组分工情况)

基本功能介绍：

整个项目分为服务器和客户端，服务器与客户端之间是 TCP 通讯，客户端与客户端是 UDP 协议

客户端功能：注册账户、账户登录、修改密码、密码保护、好友列表图形化显示、好友聊天

服务器功能：账户信息验证、账户信息修改、消息转发、消息处理、通信请求处理、用户请求处理

知识点：

- 1、 Java 对 XML 文档的增删改查功能**
- 2、 Java 的 GUI 编程**
- 3、 Java 网络编程**
- 4、 程序多线程开发与应用**
- 5、 继承、接口的应用**

分工：

牛迪：项目策划、需求分析、Java 网络编程、XML 文档操作编写

登录界面、好友列表、聊天窗口设计与实现、服务器设计、服务器编写、数据分析处理、服务器整体实现

姜廷宇、边中阳：Java 的 GUI 编程、登录界面、好友列表、聊天窗口设计与实现

系统设计与实现

（结合代码和运行时截图介绍系统的各个功能模块）

1、多线程

很多应用程序都存在对并发性的需求。例如，一个图形界面的应用程序希望在处理长时间任务的同时能快速响应用户的鼠标和键盘事件，一个在因特网上运行的服务程序希望能同时处理大量的客户请求。多线程（**multithreading**）为处理程序中的并发性提供了强有力的工具。

本程序主要通过两种方法创建线程：实现**Runnable**接口，并重写里面的**run()**方法和直接继承**Thread**类，并重写里面的**run()**，实现多线程，其中实现**Runnable**接口时，亦主要是通过两种方法创建线程：直接创建类实现**Runnable**接口和直接在程序中创建一个继承式的匿名内部类，创建一个线程。

值得注意的是，在**JVM**环境中，用户程序以线程的形式来运行。当所有的非守护线程结束时，程序也就停止了。

2、图形用户界面

Java语言图形用户界面（**Graphics User Interface, GUI**）是实现用户与程序之间交互的一个控制面板，包含菜单、组件（或控件）。通过这种直观的图形界面接收用户的输入，显示程序运行结果。**Java GUI**从最初的抽象窗口工具包**AWT**（**Abstract Window Toolkit**），到后来的**Swing**组件，现在还有**MyEclipse**的**SWT**（**Standard widget Toolkit**）及构件基于**SWT**的**JFace**，满足不同时期、不同应用市场的需求。

3、网络编程

网络应用是**Java**语言取得成功的领域之一。**Java**提供了**java.net**包，其中包括一系列网络类，支持**Java**程序直接发送与接收**TCP**或**UDP**数据报。程序员使用这些类可轻松实现基于**TCP**和**UDP**的编程。

UDP的特性：它不属于连接型协议，因而具有资源消耗小，处理速度快的特点，所以通常音频、视频和普通数据在传输中使用**UDP**较多，因为它们偶尔丢失一两个数据包，也不会对接收结果产生太大影响，所以本程序客户端与客户端的之间使用**UDP**传输。

TCP协议能为应用程序提供可靠的通信连接使一台计算机发出的字节流无差错的发送到其他电脑，对于可靠性要求高的数据通信往往使用**TCP**通信，所以本程序客户端与服务器端之间使用**TCP**传输。

【客户端】

(1) 登录界面



客户端基本功能:

- 1、 自动识别账号，当输入已注册好的账号时，显示该用户的头像
原理：在注册账号时，后台在本地写好了一个 XML (ClientUserInformation.xml) 文档，用于记录该用户的某些信息，在进行窗体加载时，对账号文本框进行持续监听，监听的时候会主动的去读取本地的 xml 文档，一旦读到该账户信息，则自动加载头像



- 2、 注册账号：
填写如下信息

昵 称:

密 码:

确认密码:

密保问题:

密保答案:

个性签名:

头 像:

00

提 交

取 消



A registration form window titled 'EasyChat' with a pink background. The form contains the following fields:

- 昵称 (Nickname): Main工作室-牛迪
- 密码 (Password): 1234567890
- 确认密码 (Confirm Password): 1234567890
- 密保问题 (Security Question): 最喜欢的歌手是谁?
- 密保答案 (Security Answer): 周杰伦
- 个性签名 (Signature): Stay hungry, Stay foolish
- 头像 (Avatar): 00

Below the fields is a square avatar placeholder with a blue background. It features a cartoon character with glasses and a blue shirt, surrounded by text: 'Script', 'Java', 'C++', 'PHP', 'Linux', and 'mysql'. At the bottom of the placeholder, it says '看编程, 学编程, 爱学习, 程序员的日常, 程序员的世界'.

At the bottom of the form are two buttons: '提交' (Submit) and '取消' (Cancel).

原理:

绘制窗体时, 对提交按钮进行监听, 当提交按钮单击事件被触发的时候, 会获得窗体上所有的内容, 比如昵称密码之类, 然后在后台会自动生成一个 4 开头的 9 位数字为用户 ID,

生成 ID 后做两件事:

第一件事: 记录账号、昵称、头像编号、密保问题、问题答案、个性签名, 将以上 6 项记录并写到名叫 ClientUserInformation.xml 的 xml 文档中,

该文档路径为: EasyCHat/system/Document/ ClientUserInformation.xml

第二件事: 发送账号、密码、昵称、头像编号、密保问题、问题答案、个性签名到服务器, 由服务器进行数据记录, 添加到名叫 ServerUserInformation.xml 的 xml 文档中

该文档路径为: EasyCHat/system/Document/ServerUserInformation.xml

3、找回密码

原理：自动读取本地的 **ClientUserInformation.xml** 文档，获取密保问题和密保答案，正确后进行密码更改并且修改本地文档信息，还需要向服务器发送本 ID 的新密码，由服务器读取文档修改密码记录。



找回密码界面，背景为红色渐变。包含以下元素：

- 账号输入框：显示“账 号：”后跟一个空文本框。
- 密保问题输入框：显示“密 保 问 题：”后跟一个空文本框。
- 提交按钮：显示“提 交”。
- 取消按钮：显示“取 消”。



找回密码界面，背景为红色渐变。包含以下元素：

- 账号输入框：显示“账 号：”后跟文本框，内容为“473034128”。
- 密保问题输入框：显示“密 保 问 题：”后跟文本框，内容为“最喜欢的歌手是谁？”。
- 提交按钮：显示“提 交”。
- 取消按钮：显示“取 消”。



找回密码界面，背景为红色渐变。包含以下元素：

- 新密码输入框：显示“新 密 码：”后跟一个空文本框。
- 确认密码输入框：显示“确 认 密 码：”后跟一个空文本框。
- 提交按钮：显示“提 交”。
- 取消按钮：显示“取 消”。



找回密码界面，背景为红色渐变。包含以下元素：

- 新密码输入框：显示“新 密 码：”后跟一个空文本框。
- 确认密码输入框：显示“确 认 密 码：”后跟一个空文本框。
- 提交按钮：显示“提 交”。
- 取消按钮：显示“取 消”。

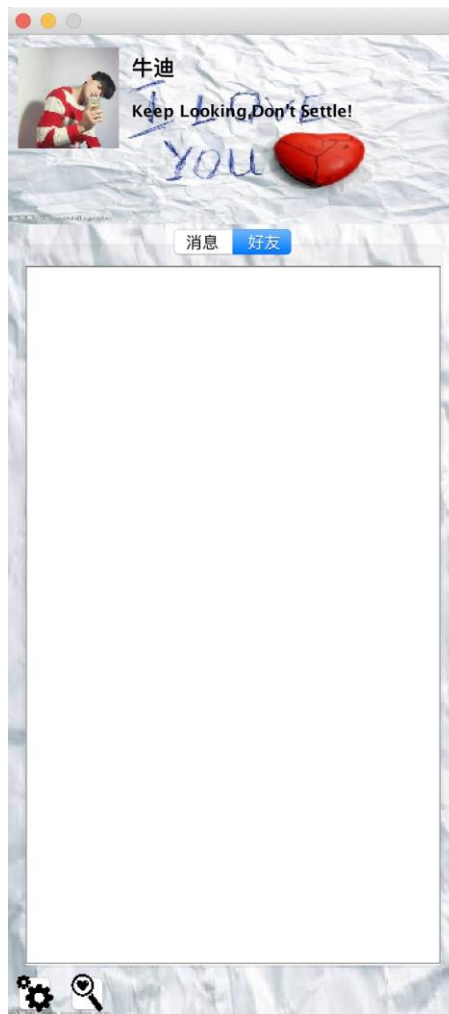
4、 记住账号

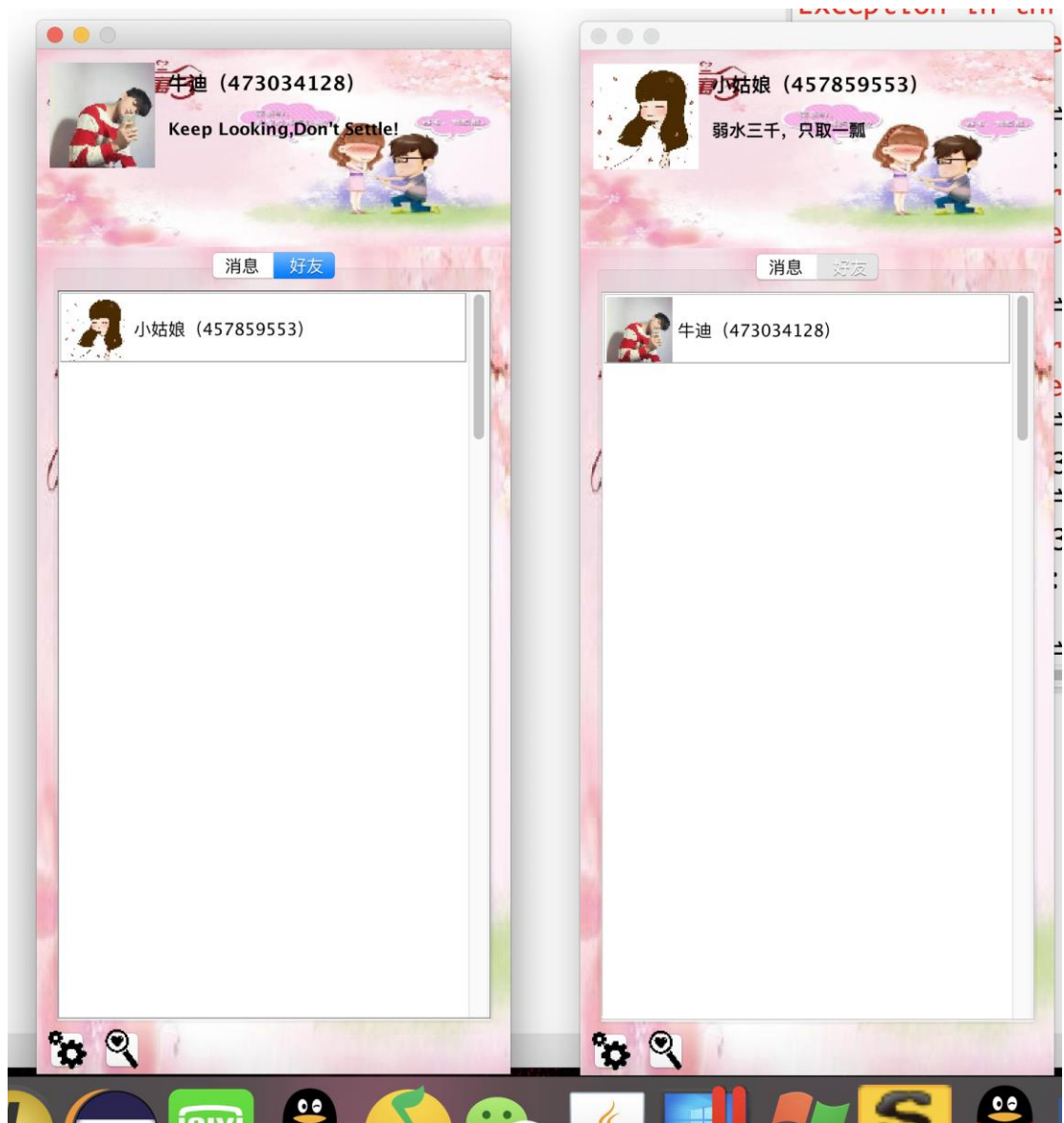
原理：给记住账号选择框设置监听，当记住账号选项被选中的时候，触发监听事件，监听事件中调用一个方法，该方法作用是在本地生成一个 **Remember.xml** 文档，文档中只有一个账号，以及判断是否自动登录，即上次触发记住账号事件的账号。一旦记住账号选择框被选中，则先清空该 **xml** 文档，再写入本次输入的账号对话框中的内容（该功能还未做出）

5、 自动登录

在最初生成登录界面之前，首先读取 **remember.xml** 文档，读取的时候先判断是否是记住账号，再判断是否是自动登录。文档中默认的值是一个初始值为 **false** 的布尔变量，当自动登录被选中的时候，文档中的布尔变量的值变为 **true**，由此每次在打开客户端的时候，就已经读出了状态。

（2） 好友列表





1、好友列表的信息获取

原理：登陆成功之后，服务器自动为所有在线的客户端发送一份在线好友列表，由此列表获得在线用户信息。用户信息包含：账号、名字、头像、个性签名。

好友列表设计：中间部分是一个滚动条，在此基础之上添加一个 `JPanel`，该 `Jpane` 的布局为 `GridLayout()`；每一个用户都会集中在一个 `JLabel` 中，该 `JLabel` 动态生成，自动添加到 `panel` 上。

2、选择好友聊天

原理：在服务器发送的好友列表信息中（该信息为一个 `Map`，不会有重复账号），包含有用户的端口号，在进行点对点通信时，通过该端口选择聊天用户。好友列表中，每一个 `JLabel` 的 `click` 监听事件中设置了，当获取到点击次数为 2 的时候，自动获取该 `JLabel` 包含的信息，从中得到端口号，由此向服务器发起聊天请求。服务器去进行端口判断，由端口号选择将信息内容传送给该端口号对应的客户端

这样方便简单的实现了点对点的即时通讯

3、设置按钮功能

单击设置按钮会有如下显示：



所有功能都是对数据的读取和修改。当涉及到密保问题和密码的操作的时候会发送请求给服务器，查询用户信息，查询之后将数据返还给客户端。由此实现了数据的设置和修改

4、聊天界面：





客户端与客户端聊天原理：

在登录开始的时候，会生成一个 **UDP** 端口号，保存并将这个端口号与 **IP** 地址发送给服务器。当用户想要与其他用户聊天时，会联系服务器，服务器将该端口号发送给目标用户，然后用户与用户进行 **UDP** 通讯

【一】服务器

服务器共分为三个界面，第一个是状态信息。主要是为了确认好友列表是否已发送到所有在线用户，第二个是在线用户列表，以图形化的形式显示所有在线用户信息。第三个用来显示用户登录信息

服务器数据处理原理：

每次的客户端的请求都会有一个标识符用来区分客户端想要服务器实现的

动作。比如：当客户端请求登录的时候，会以字符串的形式发送给服务器一条信息，这条信息以“%LOGIN%”开始，其后面紧接着的是请求登录的用户信息，当服务器收到该条信息的时候，会自动的去数据库读取用户的信息验证用户密码是否正确。





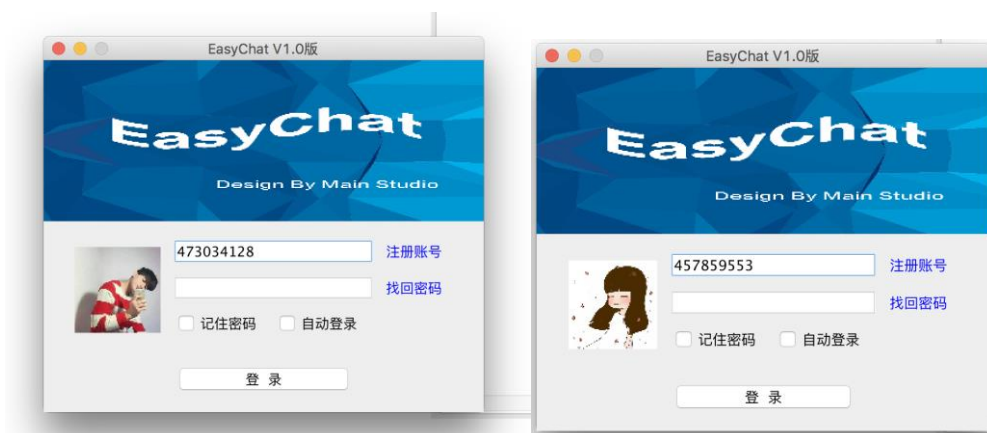


项目总结

（总结本系统的难点和亮点、以及在系统开发中的收获和体会）

1、 登录界面：

随着用户的输入，自动读取本地存取的用户信息，自动识别出用户信息并且显示用户头像



2、 注册用户

当用户进行申请账户时，会进行密码保护设置。

3、 好友列表



好友列表单独开了一个线程 ,做到了实时刷新在线信息。每次有新的用户上线 ,
服务器都会发送一次在线好友列表给所以在线用户。每次收到用户列表 ,客户端
都会动态的在 JPanel 上添加 JLabel