东软睿道

项目实训总结报告

项 目 名 称:LCS

学 校 名 称:北京理工大学

项 目 讲 师:崔浩

姓 名:刘宇捷

学 号:1820201052

实 训 日 期: 2022年8月15日-8月25日

**1 引言**

1.1项目概述

本项目来自于北京理工大学学生在计算机基础上所实践并提升能力的训练计划，目的是在Linux系统下制作一个局域网内的聊天软件。由于Linux操作系统是C语言所编译的，所以对不同架构的CPU都有着良好的兼容性。另外，本项目是建立在QT软件上，因为QT是个简单易用的开发框架的软件。所以，在Linux操作系统下开发QT项目能减小解决所要面对的接口问题。

本项目所涉及的关键技术有TCP/IP协议通讯、数据库Sqlite、QT图形开发工具、Socket“套接字”、MD5等。本报告将详细介绍我在LCS项目中所负责的开发制作过程。

1.2 需求分析

QT是一个跨平台的C++应用程序开发框架。广泛用于开发GUI程序，这可称为部件工具箱。它也可用于开发非GUI程序，例如控制台工具和服务器。我会选择尝试做这个项目是因为这个项目相似于我们日常生活中说使用的聊天软件，如：QQ，微信等。在制作这个项目的过程中可以让我了更解聊天软件其中的功能，如聊天系统、系统登入、如何与服务器进行连接等。

1.3 运行环境

软件环境：QT Creator 4.11.2 + Windows + Linux

硬件环境：Intel 3以上，2GB RAM

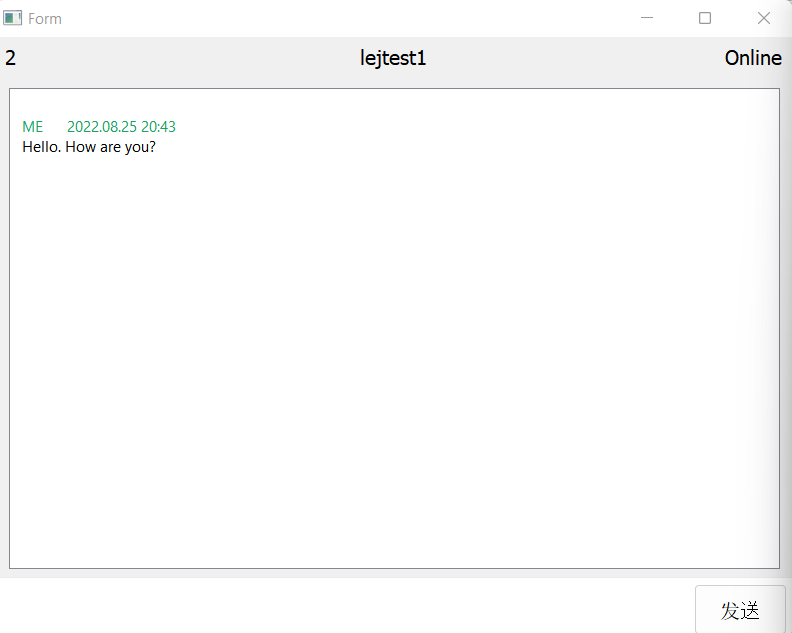
使用GitHub进行成员的程序同步

**2 项目设计**（小三黑体，居中）（以个人主要负责内容为主）

2.1 设计思路

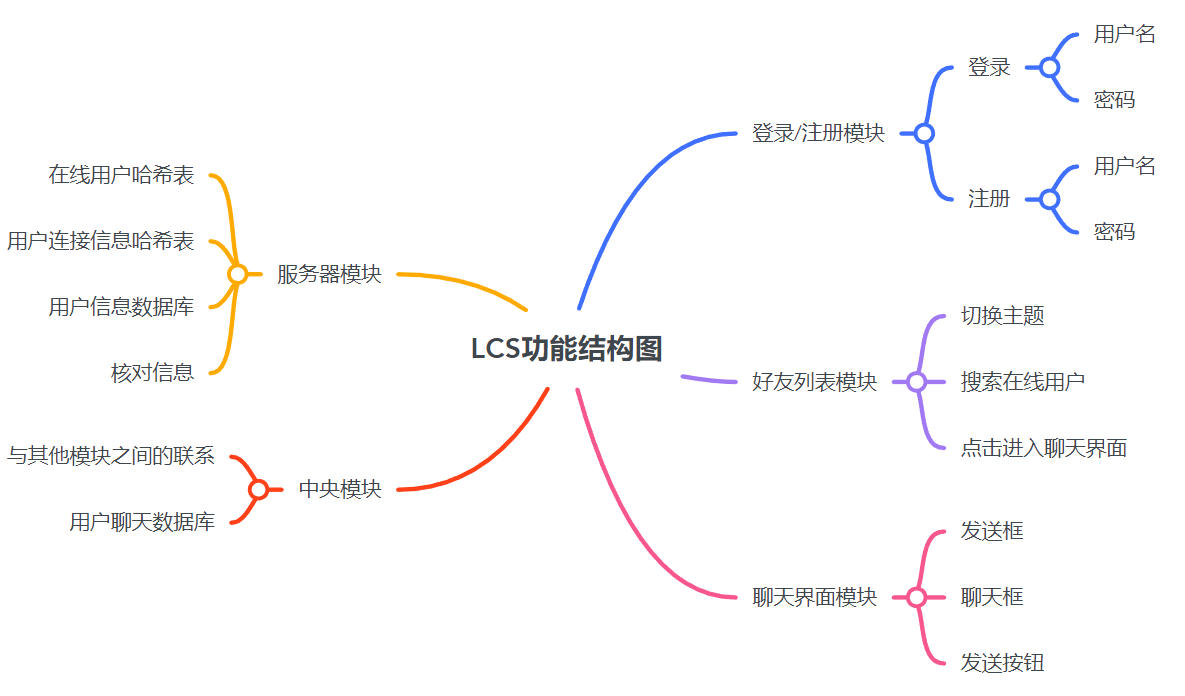
我主要负责聊天界面的功能函数以及聊天界面的ui修改。聊天界面最主要有聊天显示区、文字发送区、发送按钮、文件发送等功能组成。UI设计好后，将按钮、textEdit还有textLabel等模块写上相应的功能函数代码。

2.2 模块功能介绍

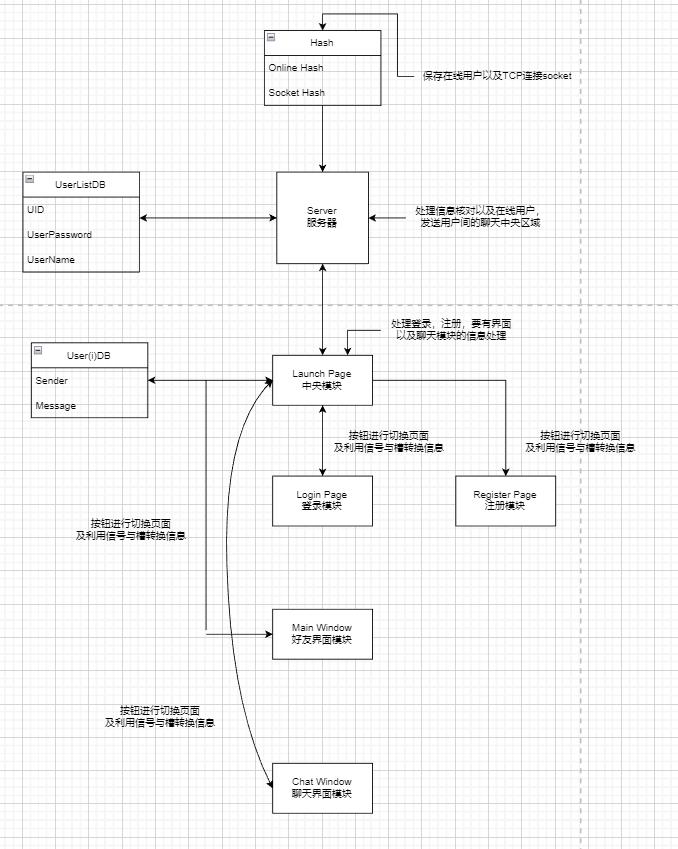


这是聊天界面。红色框是聊天显示区、蓝色框是文字发送区（打字区）、黄色框是发送按钮、绿色框是对方用户ID、黑色框是对方用户名称。

2.3 模块结构图



2.4 程序流程图



2.5 功能设计分工

根据本项目的程序流程图，我们组将分工分成五份。分工安排如下：

一、李沄澺——负责开发UI设计。

二、刘乐也——负责添加登录与注册界面的控件函数

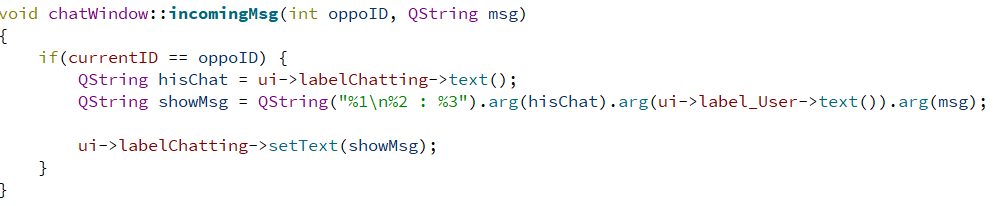
三、陈誉航——负责添加主页面控件函数、数据库

四、刘宇捷——负责聊天界面控件函数

五、李思净——负责设计服务器、调试整体代码以及页面连接

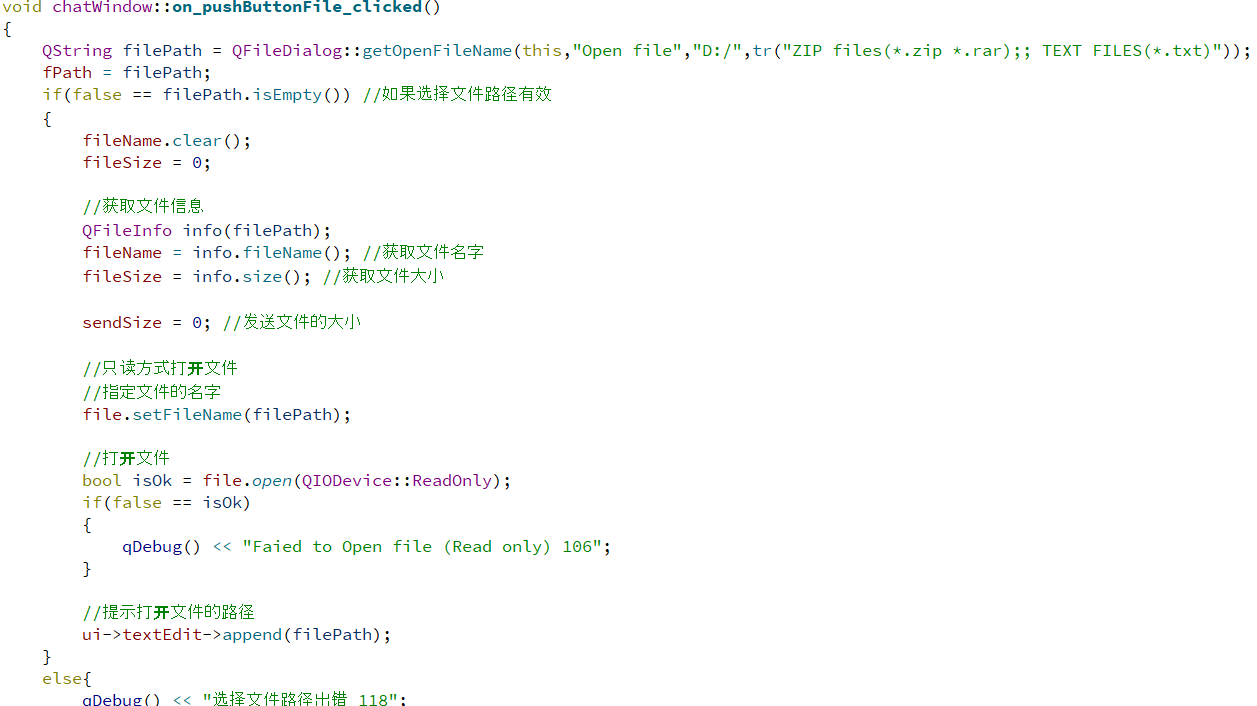
**3 详细设计**（小三黑体，居中）

文字发送&发送按钮：在打字区的文字再点击发送后会传输到服务器，在从服务器传输给另一个客户端。在点击发送按钮后，我们会把用户ID、信息与信息种类当作参数传递给服务器。另一个客户端（用户二）接受信息时会以 “（用户1的名字）：信息 ”的方式接受。



发送信息代码

接收信息代码

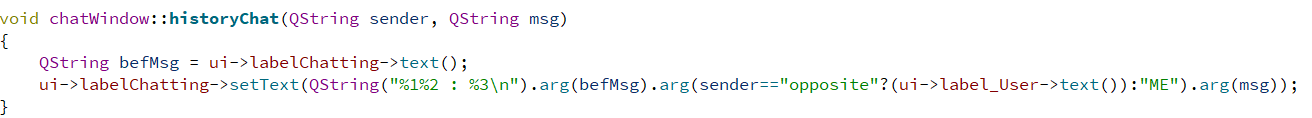


文件选择代码：

现设一个QString，再用QFileDialog来选择文件路径赋值给QString。过后再用QFileInfo以只读方式读取文件信息。传发功能就和文字传发一样给服务器再传给另一个客户端。

聊天显示历史聊天记录：

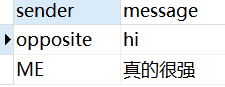
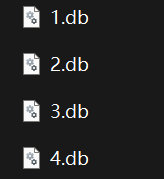
之前所发送的聊天内容在上传到服务器的同时也会保存在数据库中，这样一来便可以在关闭窗口之后看见以往的对话记录。



**4 实现方案**（小三黑体，居中）

1.登录后的操作

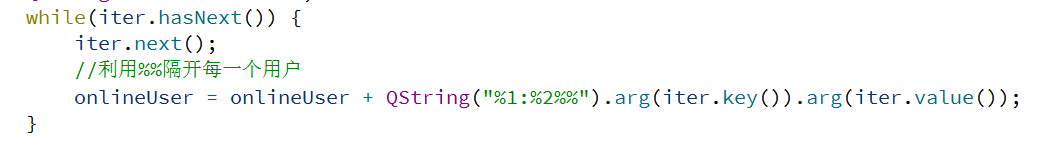
成功登陆后触发函数 void LaunchPage::**responseVertify**(QString data)，其功能包含创建该登录用户自己的数据库用来保存会话记录(该部分将在聊天窗口打开时使用)。db.setDatabaseName(QString("../%1.db").arg(UID));。



并且开启好友窗口（上图），以及向server端发出信息来获取在线用户信息QString onlineList=QString("LCS|%1|%2|").arg(UID).arg(todoAction::ONLINELIST) clientSocket->write(onlineList.toUtf8());

2.获取在线用户显示在表上

当server端接收到信号时，server将从Hash里读取在线用户数据并发送



QString returnOnline = QString("LCS|%1|%2").arg(serverAction::ONLINEUSER).arg(onlineUser);

socketHash.find(UID).value()->write(returnOnline.toUtf8());

回给发送端，并且触发信号case serverAction::ONLINEUSER:emit responseOnlineList(serverMsg, UID);其槽函数为connect(this, &LaunchPage::responseOnlineList, &mainPage, &MainWindow::responseOnlineList);最后主窗口读取数据（serverMsg）并更新在线列表

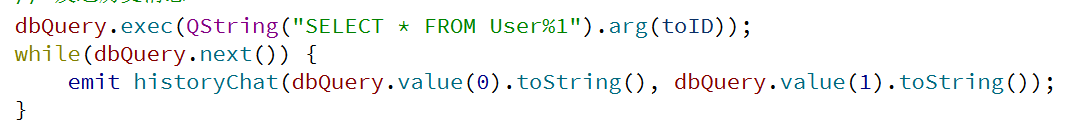


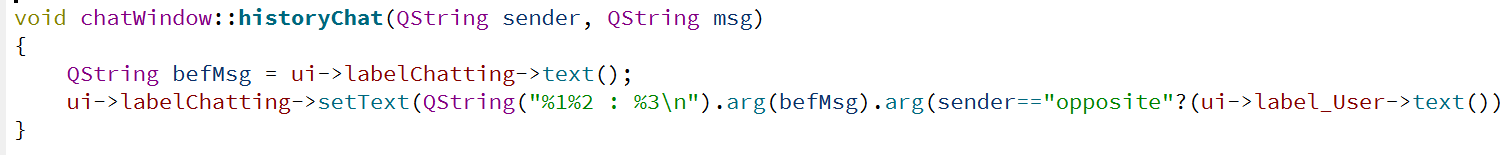
3.双击用户进入聊天页面

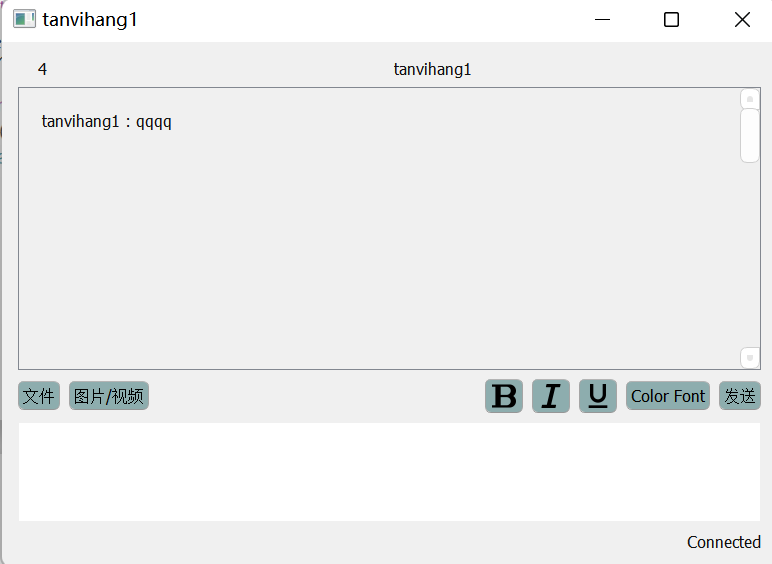
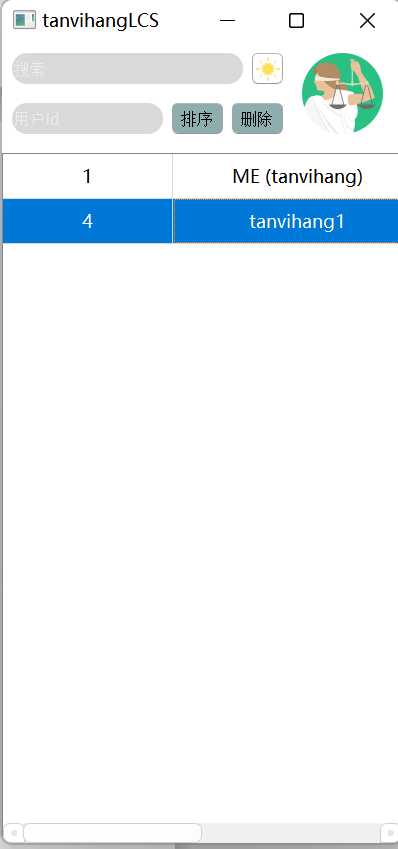
双击在线用户时发送信号emit requestChat(toId, toUsr);向launch page请求打开聊天窗口connect(&mainPage, &MainWindow::requestChat, this, &LaunchPage::requestChat);打开聊天窗口chatPage.show();后

并且发出信号emit responseChat(toID, toUsr);让聊天窗口获取用户以及对方信息。



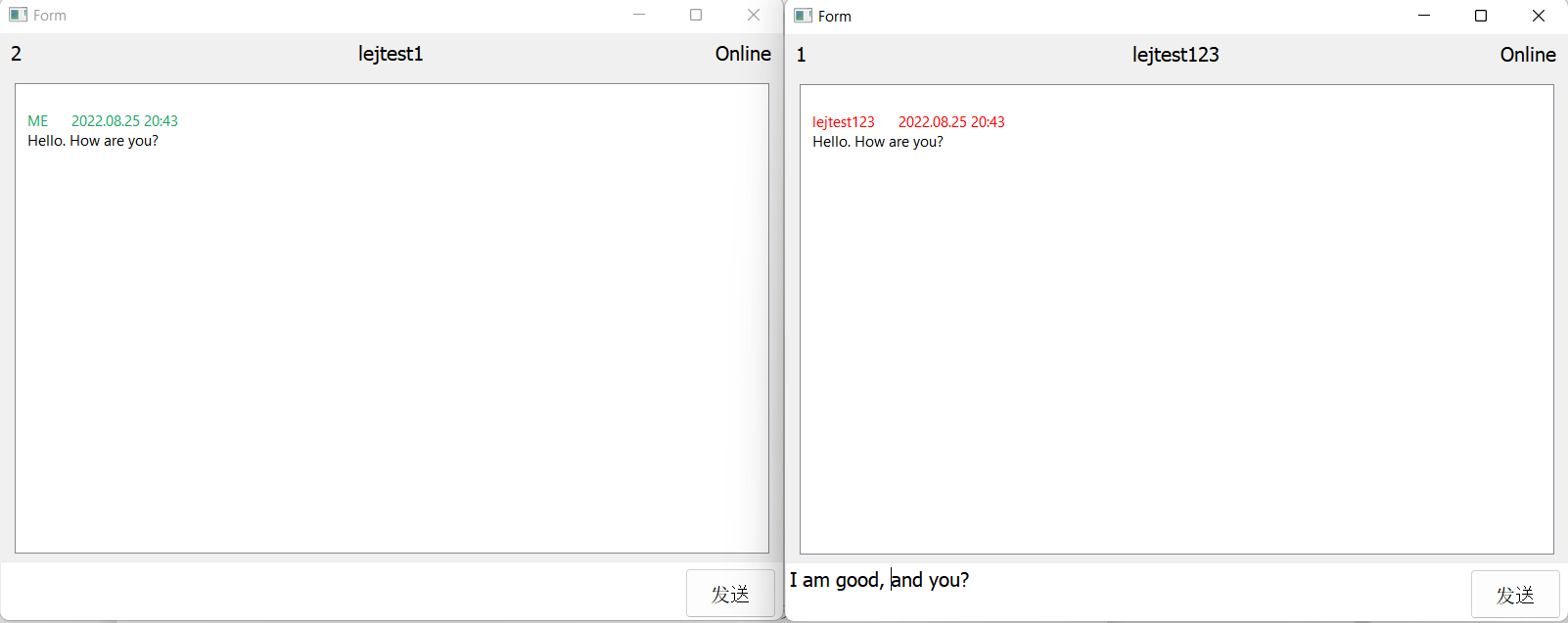
最后读取历史聊天记录





**5 系统测试**（小三黑体，居中）

从用户1发送信息，用户2接收信息。同理用户2 ->用户1



**6 心得体会**（小三黑体，居中）

在这次的项目中我分配到了设计聊天界面及聊天界面有关的工作。我对聊天界面的功能有很多想法，当然也在当中也遇到了种种问题，导致有些想法无法实现，但还是找到了另一些方法来替代，所以还是很满意的。

这次是第一次开发局域网的程序，时间也相当紧凑。在开发过程中也使用了数据库以及C++类等技术，实际应用了大一及大二所学习的知识点。虽然我是负责聊天界面这块，但是我也是有去了解如何与数据库进行连接，去研究组员们写的代码，这让我在这次的开发过程中学习了不少。