东软睿道

项目实训总结报告

项 目 名 称:局域网聊天（IM）软件

学 校 名 称:北京理工大学

项 目 讲 师:郝庆晋

姓 名:马浩元

学 号:1120172723

实 训 日 期: 2019年8月26日-9月5日

**1 引言**

1.1项目概述

待开发的软件系统的名称：

Linux 系统下的 IM（即时通讯）软件

本项目的任务提出者：

计算机科学与技术综合实践课程郝庆晋老师

项目任务开发者：

小学期计算机科学与技术综合实践教学五班潘雨非小组

用户：

局域网内 Linux 系统用户

1.2 需求分析

开发一款运行于Linux操作系统平台的IM(即时通信)软件，可以实现局域网内任意电脑之间的信息接收与发送，也可作为支持200人中小型规模的企业，企业信息平台中的一个组件及功能存在。

1.3 运行环境

硬件：

CPU：Core i7 2.8Ghz

内存：8GB DDR4

硬盘：256GB SSD + 1TB HDD

软件：

系统：Linux RHEL\_5.2 x86

软件：vsftp，mysql，gcc，gdb，Qt4

**2 项目设计**

2.1 设计思路

依据C/S模式工作原理，服务器对端口进行监听，与客户端连接后，新建一个进程进行接受数据（字符串），根据字符串首位标识符确定客户端的请求类型，并调用对应函数处理数据。

2.2 模块功能介绍

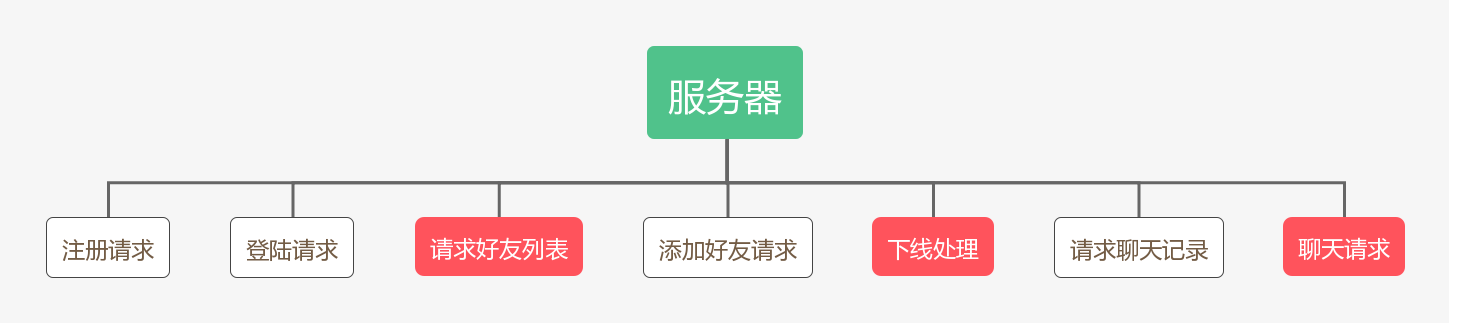
好友列表：向客户端返回请求用户的好友列表。

转发消息：接受并向客户端发送收到的聊天消息。

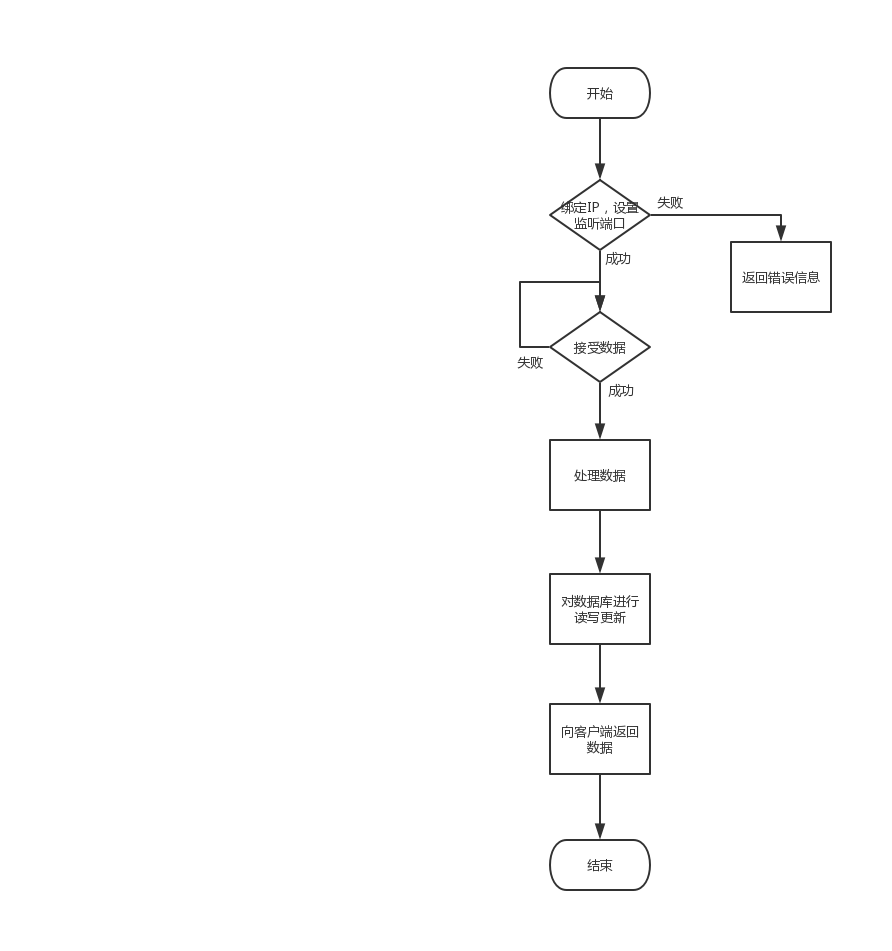
下线处理：当用户下线后处理数据库中储存的在线状态，断开连接。

掉线处理：用户连接意外断开后进行下线处理。

2.3 模块结构图



2.4 程序流程图



2.5 功能设计分工

2019/08/26-2019/08/29：安装虚拟机，配置环境，学习相关知识。

2019/08/30-2019/08/31：编写并调试返回好友列表函数。

2019/09/01-2019/09/02：编写并调试发送消息函数。

2019/09/02-2019/09/03：编写并调试处理用户下线的函数。

2019/09/03-2019/09/04：联合调试，修改bug，完善文档。

**3 详细设计**

3.1 数据传输协议

采用字符串传输数据，如下表：



核心技术：对于收到的数据，先判断第一个字符，确定请求类型，然后调用相应的函数。字符串处理函数 sprintf，sscanf。

细写明项目的设计（系统设计、数据库设计）等，并写明对所涉技术、核心功能等描述；

3.2 数据库设计



3.3 函数设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **函数名** | findFriend | | | |
| **功能概要** | 返回好友列表 | | | |
| **记述形式** | Int findFriend（char \*buf）； | | | |
| **参数** | | | | |
| **类型** | **变量名** | | **I/O** | **说明** |
| char\* | Buf | | I | 缓冲区 |
| **返回值** | **类型** | Int | **说明** | |
| **值** |  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **函数名** | sendMessage | | | |
| **功能概要** | 发送消息 | | | |
| **记述形式** | int sendMseeage（char\* buf） | | | |
| **参数** | | | | |
| **类型** | **变量名** | | **I/O** | **说明** |
| char\* | buf | | I | 缓冲区 |
| **返回值** | **类型** | Int | **说明** | |
| **值** |  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **函数名** | ofline | | | |
| **功能概要** | 下线处理 | | | |
| **记述形式** | int offline (char\* buf) | | | |
| **参数** | | | | |
| **类型** | **变量名** | | **I/O** | **说明** |
| char\* | buf | | I | 缓冲区 |
| **返回值** | **类型** | Int | **说明** | |
| **值** |  |  | |
|  |  | |

**4 实现方案**

4.1 int findFriend（char \*buf）

功能：向客户端发送好友列表

输入：标识符为8的字符串，eg：8|50001；

返回值：无

实现方法：

1.从buf中解析用户id

sscanf(buf+2,"%s",userid);

2.从数据库查询该用户连接的套接字标识符

select confd from user where userId = userid

3.从数据库查询该用户的好友

select friendId, friendName ,Online from friendList, user where friendList.userId = userid and user.userId=friendList.friendId

4.将用户好友发送至客户端

while(row = mysql\_fetch\_row(result))

{

strcat(sendBuf,row[0]);

strcat(sendBuf,"|");

strcat(sendBuf,row[1]);

strcat(sendBuf,"|");

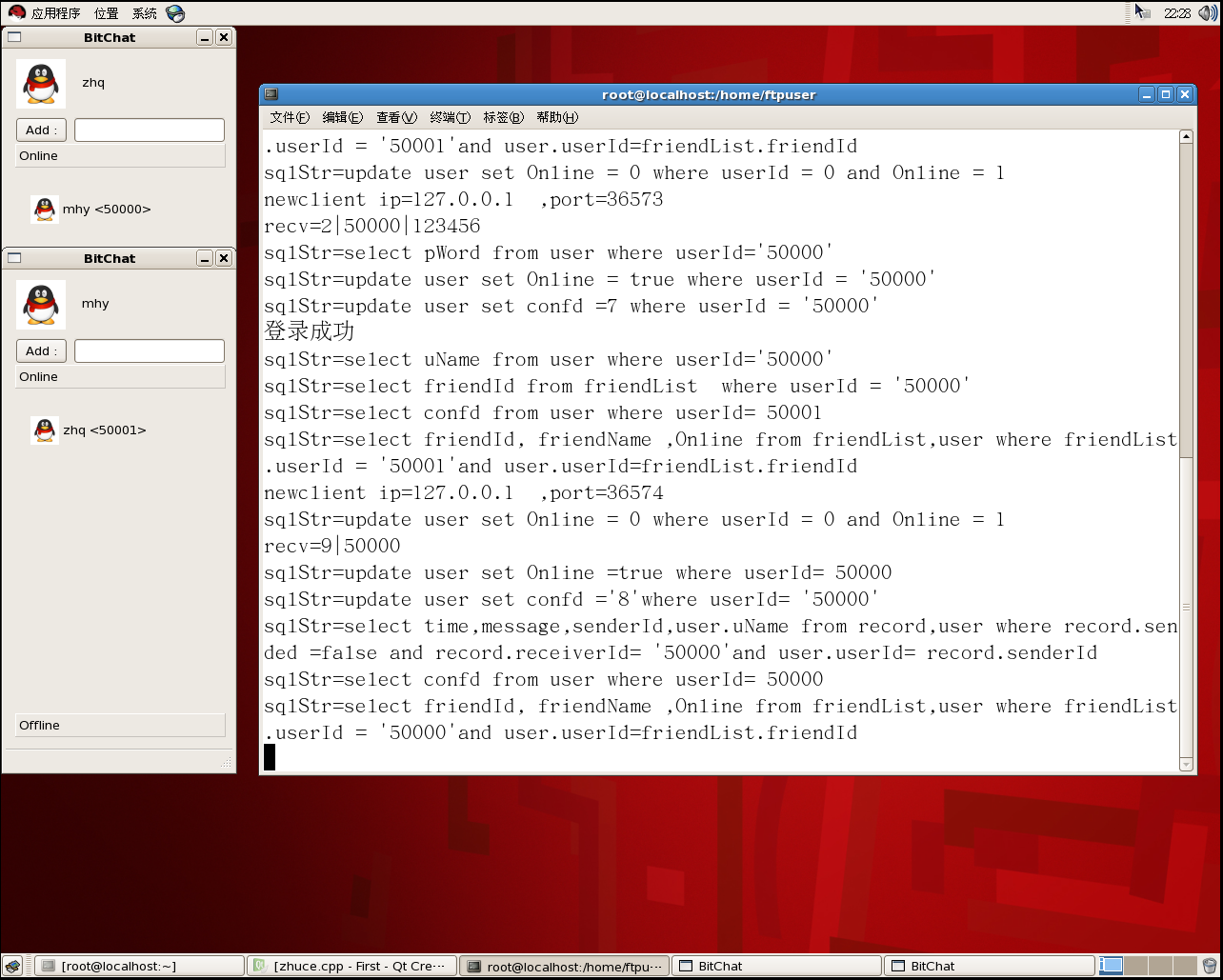
strcat(sendBuf,row[2]);

strcat(sendBuf,"|");

}

send(confd, sendBuf, strlen(sendBuf), 0);

截图：



4.2 int sendMessage（char \*buf）

功能：收到客户端聊天请求，向对应用户发送消息。

输入：标识符为5的字符串，eg：5|2019-09-03 10:26:25|hello|50000|50004

返回值：无

实现方法：

1.从buf中解析出相关数据，发送者，接收者，时间及消息内容。

2.从数据库中查询接收者的在线状态及套接字标识符。

select Online, confd from user where userId = receiver

3.如果接收者在线，将消息发送给接收者。

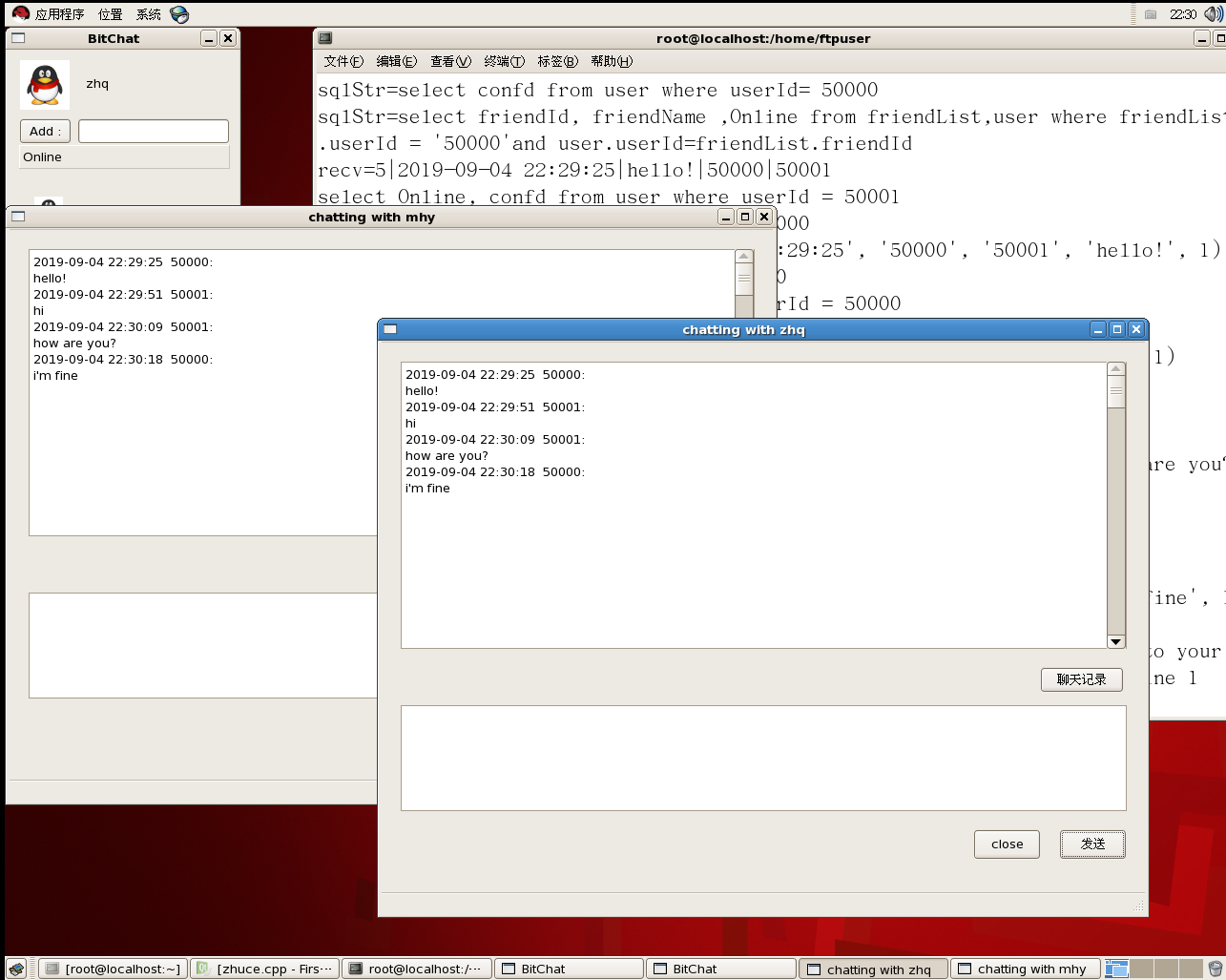
sprintf(sendbuf, "5|%s|%s|%s|%s", time, message, sender, senderName);

send(confd, sendbuf, strlen(sendbuf), 0);

4.将该消息写入聊天记录数据库

sprintf(sqlStr, "insert into record values('%s', '%s', '%s', '%s', %d)", time, sender, receiver, message, isOnline);

截图：



4.3 int offline（char \*buf）

功能：收到客户端下线请求，处理数据库用户在线状态。

输入：标识符为5的字符串，eg：5|2019-09-03 10:26:25|hello|50000|50004

返回值：无

实现方法：

1.从buf中解析出用户id

2.更新数据库中用户在线状态为false

3.向该用户的好友发送该用户已离线，调用findFriend更新他们的好友列表

select friendId from friendList where userId = userId

while(row = mysql\_fetch\_row(result))

{

char buf[1024] = {0};

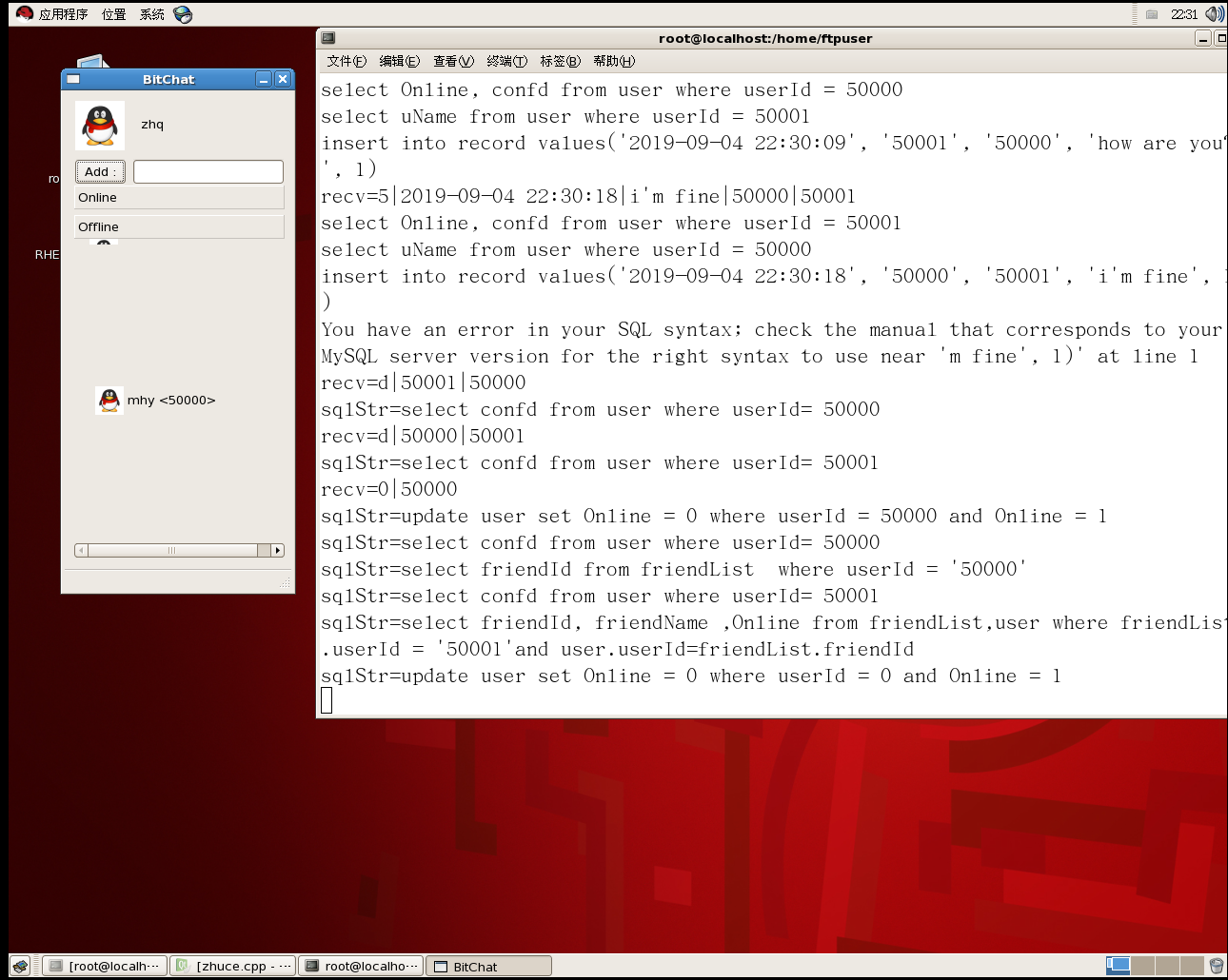
strcpy(buf,"8|");

strcat(buf,row[0]);

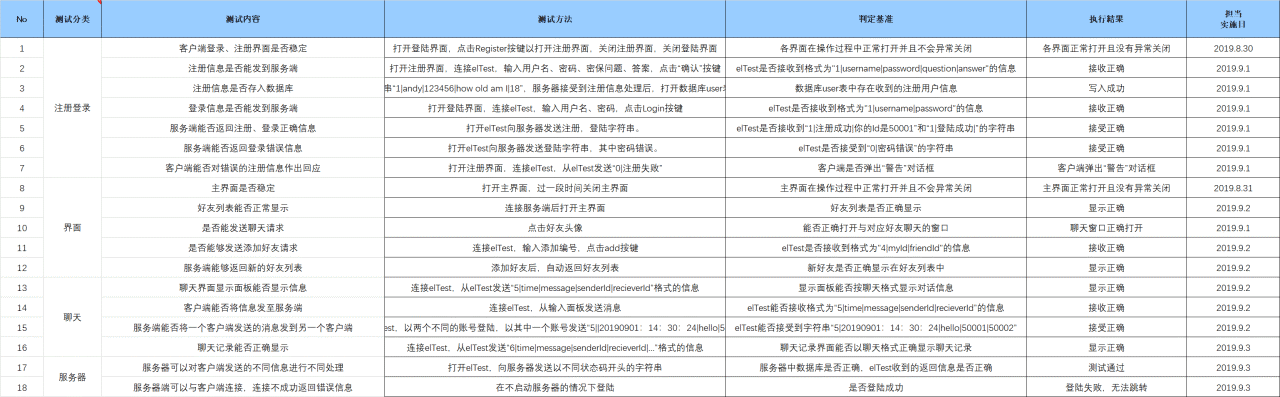
findFriend(buf);

}

截图：



**5 系统测试**

**6 心得体会**

两周的开发实践，让我有了第一次团队开发的经历。上学期的Android开发课程，使我有过一次简单的开发经验，当时自己一个人完成从头到尾的所有工作。但此次开发实践是以团队为单位。不再需要一个人完成所有的事情，这就要求我们在完成好自己的任务的同时也要注意队友进度，注意各个部分的协调，能够顺利对接。团队的协调与合作不是件容易的事情，好在我们组分工明确，沟通及时，尽可能早的发现了问题，并作出相应修改调整。

此次开发实践，我主要负责服务器的编写，学习到了网络程序开发的基本流程与方法。更深入的理解了C/S模式的工作原理。同时，由于本次开发是在Linux环境下进行，我了解了Linux系统的基本操作，以及Linux下的开发工具的使用，比如gcc，mysql，vim等等。

此次实践，除了学习了许多新知识，我还体会到了团队协作的重要性，如此复杂的工程，靠个人是很难在这么短的时间完成，但有了团队协作及分工开发，我们在一周多的时间就实现了软件的基本功能。