



# 网络聊天室

---

李佳灏

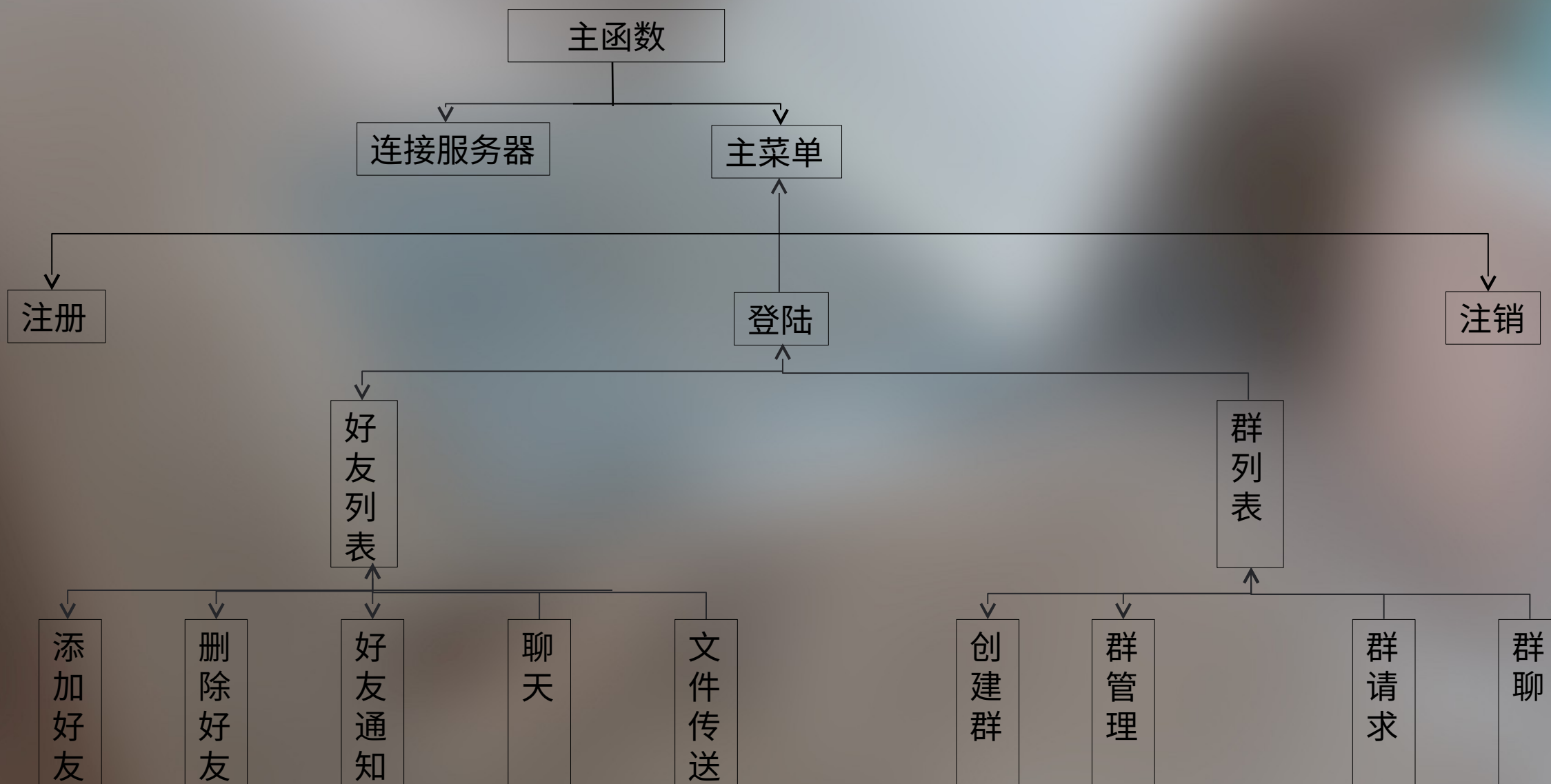
# 01

## 项目简介

本项目是 Linux 操作系统下的聊天室，采用了主从式架构 (Client-server model)，可以实现用户的注册登陆，以及登陆后的一些通信功能。由于最初自己设计的问题导致用户体验不是非常的好。

• • • • •


# 01 客户端流程图





设计思路：Linux C 编程实战

解决问题方法：小组讨论，  
请教学长学姐，网上查阅资料



**01** CPU:P4 1.8GHz 以上

**02** 内存: 256MB 以上 硬盘: 500MB 以上

**03** 显示器分辨率: 1024\*768 以上

**04** 操作系统: 各种版本的 Linux



## 02 特点



01

发送离线文件：客户端将文件上传至服务器，在起初用户注册时，便会为每个用户在服务器创建文件夹。然后接收的用户可以通过在数据库中存储的信息从而从服务器下载文件。下载后服务器会删除该文件

02

客户端多线程刷新列表：这个主要时由于自己起初架构的原因不得已选择的方式，另外自己作死分了群列表和好友列表。导致一些提醒上面的功能不是非常的完善。



### 实现文件描述符的刷新

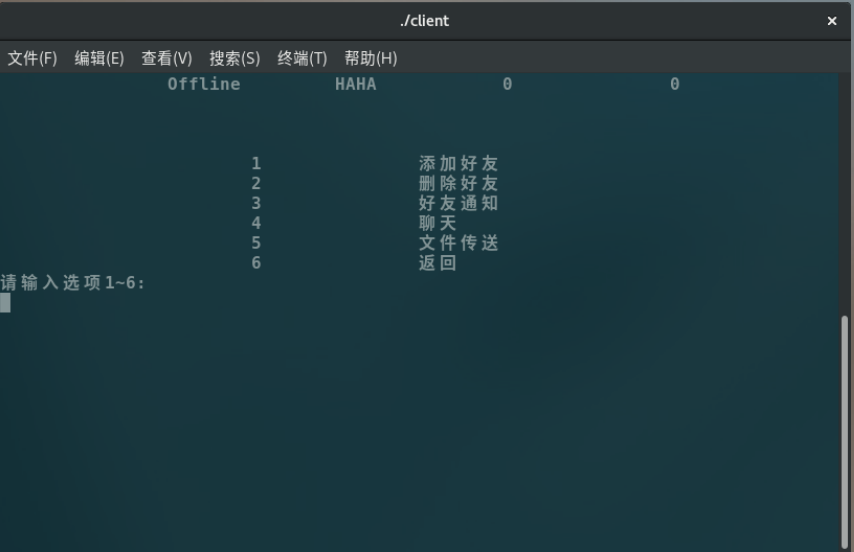
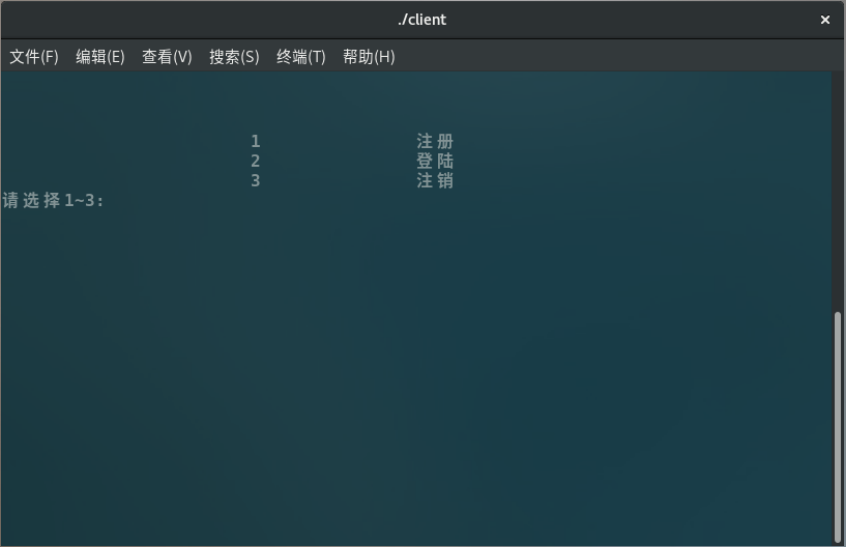
```
void fflush_sock(int sockfd, int size)
{
    char buf[size];
    fd_set readfds;
    int ret;
    struct timeval tv;
    while(1)
    {
        FD_ZERO(&readfds);
        FD_SET(sockfd, &readfds);
        tv.tv_sec = 0;
        tv.tv_usec = 0;
        ret = select(sockfd + 1, &readfds, NULL, NULL, &tv);
        switch(ret)
        {
            case -1:
                exit(-1);
                break;
            case 0:
                return;
            default:
                recv(sockfd, buf, size, 0);
        }
    }
}
```

### 手动锁的实现

```
void* send_frhistory(datapu* pack)
{
    while(pack->flag)
    {
        send_friend_history(pack->clifd, pack->mysql, pack->name, pack->friname);
        sleep(5);
    }
    pack->flag = 1;
    pthread_exit(0);
}
```

```
if(strcmp(mess.news, "return") == 0)
{
    pack->flag = 0;
    free(pack);
    break;
}
```

# 01 测试运行图片







---

谢谢观看

Thanks For Watching

---