

一.完成的内容

必做：全部

选做：群聊，P2P 文件分发

二.设计思路

1.先写和服务器通信的函数在 CS.py 里，调试好

2.再写和其他用户收发信息文件的函数和收信息文件的类在 P2P.py 里，调试好

这样轮子就造好了。后面调这些写起来就不再管网络的事儿了。收文件要写类是因为要设置一个信号，这样才能子控件向父控件传，pyqt 的信号不写在类里用不了。

3.写登陆窗口

4.写主界面：就按电脑版微信那样设计的界面。本来只想写必做的，写的时候先没管群的问题。输入框和聊天记录显示都是小事儿。好友列表是继承了 QListWidget 写出来的，要开个线程不停地遍历好友，看是否在线，然后直接改 item.text 显示出来，然后在线的好友双击就能跟他聊天，不在线的好友的就会提示本程序仅支持在线聊天。然后好友信息和聊天记录在主窗口打开时读文件来初始化，关文件时存文件。运行期间，加好友，收发消息等都要做对应记录，这样关的时候存起来方便。为代码利用率高文件存的形式都是用字典存。然后为方便并没有加好友的同意与否，可以直接加好友，A 加了 B，就可以给 B 发信息，然后 B 收到 A 的信息，A 就自动出现在 B 的好友列表里了。发文件时会在聊天记录窗显示一下，然后文件是直接存到/rcv_file 里，没有让接收方选择存哪。加好友写了个小框，很简单就。加好友时有让选头像，头像其实主要是看着好看和提升逼格，程序不维护每个好友和头像对应关系，每次登陆时读文件知道有那么多好友，然后头像是随机分配的。显示消息记录是，根据当前双击选择的好友，然后遍历那个存所有聊天记录的 list，对 list 的所有元素（是字典类型）进行遍历，选出该显示的显示。

4.然后是选做部分。本来不打算写的，但是群聊和群发文件 3 分很诱人，我最后还是写了。本来想和微信一样把群也放在好友列表里平权的，但是发现要改的地方太多了，放一起思路容易糊，最终决定群聊新开一个界面，长相和主界面几乎完全一样。然后就是把主界面的那些代码的基础上进行修改。主要改变有：要维护每个群都有哪些人，发信息和文件的数据设计有变化，P2P.py 里发信息文件也要写新的（比如 A 开了个群，里面有 BCD，然后 A 在群聊发消息，BCD 就要自动增加这个群，所以消息里不仅要有发送的消息，目标，来源，还要包含这个群的信息，群名和群有哪些成员，这样 BCD 才能自动建一个跟 A 开的一样的群）。

三.总结

1.学会了套接字网络传输，对课上所学有了更深的理解

2.学会了手写 pyqt5，虽然学了很一会儿，但是写熟了真的爽，又快，用 QGridLayout 布局又很整齐，而且不用 qt designer 点来点去就很有逼格。而且自己写相比于 qt designer 生成.py 再继承写思路更清楚，文件更少，总而言之就很爽！

3.收消息的线程一直在运行，怎么关很有点东西：设一个 settimeout 它才会 accept 没收到东西也过一段时间把 while 里想下再回到上面跑一圈，这样 while 的判断才会被执行到，才能关。

4.发文件也有点东西，设计发文件的 ack 确认是为了保证第一次发的就只有规定好的字典数据，这样接收方收到才能直接处理这个字典。没有 ack 机制则会发送字典加整个文件，接收方就不好从中提取出数据类型是'text'还是'file'了。

5.pyqt5 的信号机制这次彻底想清楚了。C++小学期用信号与槽用的晕晕乎乎的。父控件有什么事件当然可以直接 connect 一个函数去让子控件干啥，但是子控件有什么事件想惊动父控件，就要创一个 pyqtSignal 了该子控件后，再写一句子控件的信号 connect 到一个父控件里的函数然后父控件里声明而且 pyqtSignal 还可以传参数，str/object 的，然后父控件的那个函数的参数里就要有上这个信号传出来的参数。

6.第一次接触多线程，感觉写和 debug 过程中对多线程理解不断加深。

四.存在的问题

1.自己运行通过改端口号上两个号互相发大小文件都可以。自己运行一个号加自己为好友发大小文件都没问题。和室友一人运行一个号发文件可能部分缺失，发大文件对方接收到一定大小就不收了，然后发文件进程就会卡在那儿，这时候强行直接关程序会导致写的关程序时要执行的关线程代码没有正常执行，导致线程没有关掉，收消息的端口被线程占用着，再打开程序就打不开了。不知道这个是不是防火墙或者自己电脑什么特殊设置有关。我们寝室不同电脑收大文件收到多少就不收了的那个阈值是不同的。

2.发文件没有显示发的进度，发大文件就不知道到底发到哪了，直接跑代码可以在命令窗口中还能通过 print 发的东西来看到，直接运行.exe 文件就看不到了。