

一、功能介绍

登陆界面：

可以注册一个用户名唯一的账号，并依据已注册账号进行登录

主界面：

展示头像、用户名、性别、年龄、背景图片。

功能键：

修改头像

根据内置的 16 张图片挑选头像修改

修改背景

根据内置的 16 张图片挑选背景修改

发布征求：

编辑征求搭子的细节，以及 tag 类型挑选。

征求大厅：

在大厅中，下滑查看所有正在发布的征求。用户可以对意愿征求发起“参加”。

点击参加后，会向发布者发送申请信息，等待发布者的回复。

我的征求：

在该界面下，用户可以对自己发布的征求进行集中管理。

“结束”：代表不再征求搭子，即不会在大厅中展示。

“over”：代表整个搭子活动的结束，伴随着对应群聊的解散

“删除”：删除该征求，并停止一切相关活动。

朋友列表：

下滑查看所有添加的好友。

群聊列表：

下滑查看所在的所有群聊。

一个“群”与一条搭子征求绑定。征求的结束，也会伴随着群的解散。这样更加符合找搭子“一次性”的特点，倘若想继续结识，可以通过添加好友的方式增加联系。

添加朋友：

根据用户名搜索用户进行添加，对方会接受到好友申请。

待办消息：

下滑可以查看所有待办消息。

待办消息目前包含两部分：1. 好友申请 2. 搭子申请

好友申请的通过，双方形成好友关系。搭子申请的通过，申请人进入对应征求的群聊中，以便后续活动的开展。

二、项目各模块与类设计细节

整体分为 2 个模块，分别为服务器端和客户端。

服务器端：

server 类：

该类实现是服务器端的主体，监听来自客户端的消息，同时根据消息的“类别记号”分类对数据库进行增删改查的内容。在这一部分完成对于整个程序内在

运行逻辑的构建，让数据流通起来。具体包含：注册登录时，对用户名和密码进行存储或校验；更新用户基本信息（头像、性别等）；发送消息，将消息转送给对方账号；申请好友\搭子，将该待办消息转送给对方；同意好友\搭子，完成双方 id 好友关系的绑定\将搭子对象拉入对应群聊中；发布征求，将征求相关的全部信息存储到数据库中；结束\over\删除征求；获取群聊列表\好友列表\待办消息列表，将与该用户 id 关联的群聊、好友、待办消息信息整合发送；用户个人全部征求\在线全部征求，用户可见的征求信息征求和发送；待办消息的同意与拒绝；用户基本信息，根据对方用户 id 发送对应的用户基本信息。

Database 类：

设计数据库框架，以及对于数据库各种操作函数的编写，总体上是和服务端类对于数据库的操作一一对应。

用户基本信息表：包含用户 id，用户名、密码、性别、年龄、头像、背景信息。

群聊表：群聊 id、群名称

征求表：征求 id、发布者 id、征求类型、内容文本、时间、发布状态

好友表：将两个是好友关系的用户 id 联结

群成员表：将用户 id 和群 id 联结

待办事项表：待办信息 id、消息办理用户 id、消息内容。

客户端：

客户端 tcpsocket (my_tcpsocket)：服务器发给客户端的消息的“中转站”

用以连接服务器、监听服务器发来的消息。并根据发来消息的类别，产生不同的信号。不同的界面绑定不同的信号，使得数据精准地传达到对应界面。例如：登录的信号，与主界面信息处理槽绑定。接收到“登陆成功”，登陆界面关闭，加载用户主界面，若是“登陆失败”，则提示“账号或密码错误”。

登录界面类 (widget)：

在该界面能够完成注册，注册成功会有相应提示。

根据正确账号能够完成登录，账号错误会有提示。

列表类 (pendingmessage)：

展示好友列表、群列表、待办消息列表。该类能够相应 my_tcpsocket 信号，根据信号里面的内容，具体化相应列表。

好友、群聊、待办消息列表单元 (frienditem\groupitem\todoitem)：

是作为列表中的一条信息。点击好友头像可以查看好友基本信息。

聊天框类 (chat_window)：

上方正中显示好友用户名/群聊名，在中间具体显示对话内容，下发有文本编辑框。

好友、群聊消息单元 (messageitem)：

显示用户名、头像、时间、消息内容
修改头像按钮（change_avatar）和修改背景按钮（change_back）：
通过与服务器连接实现实时从数据库里修改头像和背景的功能

修改性别和修改年龄编辑框：
通过 override QLineEdit 类，来实现双击激活输入框进行输入的功能，再通过回车键发送信号实现数据库的修改

发布征求、征求大厅、我的征求、朋友列表、群聊列表、添加朋友、代办消息等按钮：
点击通过信号与槽实现与子界面的串联以及后续功能的实现

动态页面类（dynamic）：
通过客户端发送信号，从服务器获取所有正在进行的征集，显示所有存在的征集

个人页面动态类（person_dynamic）：
显示自己发布的所有征集，并可以通过 post_item 在该页面上进行相应操作

动态列表单元（post_item）：
作为动态列表的一个单元，在该单元上用户可以根据访问的权限进行相关的操作，如参加，删除，结束等操作

个人信息页面类（person_message）：
在该页面上显示个人的基本信息，如年龄，性别，头像等

发布征集页面类（release_post）：
在该页面上可以进行发布征集，利用 textEdit 进行输入要发布的内容，有两个按钮，点击发布按钮后会返回上一个页面，并将内容传到服务器

三、小组成员分工

陈李章：服务器类、数据库类、部分的客户端（客户端 tcpsocket、登录界面类、列表类、聊天框类）

祝渐祥：客户端登录之后的主界面以及该界面与之后各个子界面的衔接（主界面中的各个按钮以及其背后对应的各项功能的衔接）

马凯：客户端中的（动态页面类，个人页面动态类，动态列表单元，个人信息页面类，发布征集页面类）

四、总结与反思

目前我们总体实现了初期对于项目的构想。但是仍然存在着一些缺憾。比如：数据库中没有存储聊天记录，这意味着用户间只能在同时打开聊天框的情况下才能通信，但要补足这一点并不困难；对于用户名和昵称暂且混为一谈且没有修改用户名和密码的功能，但是后续在数据库中稍作修改也即可实现；对于搭子好友的评价体系没有构建，初期设想对搭子好友进行分数评价，以规范大家的行为；征求大厅内，没有根据“征求类型”进行分类以及缺少关键词搜索功能，不便于立刻查找到满足自身需求的征求。要实现后面两点，还需要一定的工作量，精力有限，我们小组止步当前。