

# QChat——基于 Qt 的聊天交友软件

逸一时误一世队

张嵩仁 杨锐

## 一、程序功能介绍

### 1.1 概述

本项目开发了一个可进行实时通信的聊天交友软件 QChat，其由服务器（QChatServer）与客户端（QChatClient）两个软件组成，实现了常见聊天软件中的大部分基本功能，并在此基础上加入了许多便捷、实用的原创功能，以提升用户的使用体验。

### 1.2 注册—登录体系

该体系由客户端的注册界面、登录界面，以及邮箱验证码登录界面共三个界面组成，用户可通过以上途径经过验证后直接进入主界面。

#### 1.2.1 注册界面

注册界面（Register Window）（图 1-2-1）中，用户可自行输入昵称、密码、确认密码、个人邮箱，并接收服务器所发送的验证码。服务器会对昵称与邮箱是否已被注册、密码强度是否达标、两次输入的密码是否一致、邮箱格式是否正确、验证码是否正确等信息进行验证，通过后即可以该身份进入主界面。

图 1-2-1 注册界面

点击“显示”按钮可切换密码的可见性。

界面会对两次输入密码是否一致，以及密码强度大小进行提示。

### 1.2.2 登录界面

登录界面（Login Window）（图 1-2-2）中，用户可输入以前注册过的昵称/邮箱+密码进行登录。服务器将自动验证昵称/邮箱是否存在，以及密码是否正确。



图 1-2-2 登录界面

勾选左下角“记住我”选项，可记忆本次输入的所有内容，下次运行时将自动填充。

### 1.2.3 邮箱验证码登录界面

邮箱验证码登录界面（Email Login Window）（图 1-2-3）中，用户可输入以前注册过的邮箱，填写所接收到的验证码以登录。服务器将自动识别邮箱格式是否正确，邮箱是否已注册，以及验证码是否正确。



图 1-2-3 邮箱验证码登录界面

## 1.3 个人信息修改

客户端用户进入主界面后，可在左侧窗口栏点击设置图标进入设置界面（Setting Page），用户可在其中修改头像、昵称、密码、聊天背景、个人兴趣。服务器会实时检测昵称是否已存在，并对客户端的有效修改进行回应。

未自行设置头像的用户将使用默认头像。

修改聊天背景时，将打开聊天背景修改界面（Background Setting Dialog）（图 1-3），

其中可自行于取色板中取色，或拾取屏幕中的颜色，也可选择图片作为聊天背景。已选择的聊天背景将在该界面展示预览图。



图 1-3 聊天背景设置界面

### 1.4 好友—匹配体系

该体系由客户端的好友界面、好友匹配界面两个界面组成，用于对用户好友进行个性化添加、删除等操作。

#### 1.4.1 好友界面

好友界面（Friend List Page）（图 1-4）中展示了用户所有已添加的好友（包括服务器），用户可直接看到好友的在线情况。

1.4.1.1 点击上方“添加好友”可输入好友昵称进行添加，服务器会反馈该用户是否存在，该用户是否已添加过，好友是否成功添加。

1.4.1.2 右键好友可对好友进行删除。

1.4.1.3 将鼠标悬停在好友头像上，可查看放大后的清晰头像。

1.4.1.4 双击好友，可直接切换到与该好友的聊天界面。



图 1-4 好友界面

1.4.2 匹配界面

用户可在该界面进行好友匹配，服务器将自动推荐与该用户所填写兴趣相似的好友，并显示其昵称与兴趣，用户可直接将其添加为好友。

无论好友是否添加，用户可点击“重新匹配”再次匹配好友。

匹配过程中，用户可随时点击“取消匹配”结束匹配。

1.5 聊天界面

聊天界面（Chat Page）（图 1-5）内集成了大量聊天软件基本功能。

1.5.1 聊天主界面

1.5.1.1 聊天主界面左上角为菜单栏，点击后会出现许多原创特色工具，详情见 1.5.2。

1.5.1.2 聊天主界面左上方显示聊天对象昵称。

1.5.1.3 聊天主界面右上角为查找按钮，点击后其下方新增查找框（Message Search Widget），可输入关键词对相关聊天记录进行查找并快速跳跃至当前位置。点击“下一个”按钮或点击 Enter 快捷键，可跳跃到下一个含有关键词的聊天记录；点击“上一个”按钮或点击 Shift+Enter 快捷键，可跳跃到上一个含有关键词的聊天记录。

1.5.1.4 聊天主界面中央为聊天记录显示区，我方发送的消息用绿色气泡显示，对方发送的消息用蓝色气泡显示。气泡中包含消息具体接收时间及消息内容。文件消息可点击“打开”按钮直接打开。关闭程序后再次打开，聊天记录依旧会留存。

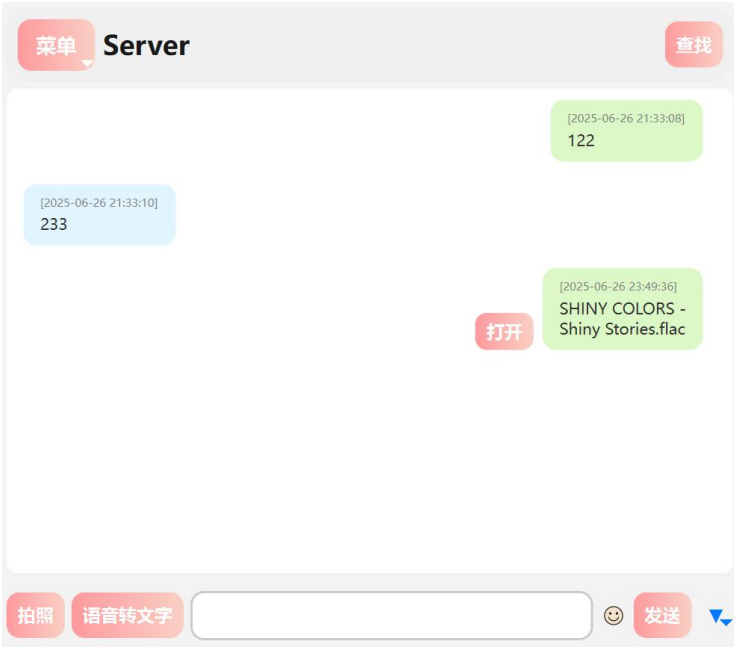


图 1-5 聊天主界面

1.5.1.5 聊天主界面左下角为拍照按钮，点击后可打开拍照界面（Camera），再点击“拍照并发送”按钮，可直接使用电脑内置摄像头拍摄图片并发送，同时图片会保存到本地，屏幕上会显示图片保存位置及分辨率。

1.5.1.6 拍照按钮右侧为语音转文字按钮，点击后可打开语音转文字界面（Record Dialog），点击“开始录制”可录制音频，点击“结束录制”后，语音转文字结果将直接显示于输入框中。

1.5.1.7 聊天主界面正下方为输入框，可输入消息并点击右侧“发送”或 Enter 快捷键进行发送，若发送/接收内容包含某些关键词，屏幕中将展示相应的动画。

1.5.1.8 输入框右侧为表情包界面（Emoji Picker），打开后展示预置的 50 个 emoji，点击可直接将 emoji 添加于输入框中。

1.5.1.9 聊天界面右下角为下拉菜单，其中包括“发送文件”按钮和“延迟发送”按钮，点击前者可发送文件，点击后者将打开延迟发送界面（Delay Send Dialog），其中可设置延迟发送的绝对时间（年月日 时分秒）或相对时间（x 时 x 分 x 秒后）进行延迟发送。

1.5.1.10 将文件直接拖拽到聊天主界面可触发发送文件确认窗口（File Confirm Dialog），其可直接发送文件。

## 1.5.2 菜单界面

点击左上角“菜单”后将展示菜单栏（Feature Menu Widget），其中显示以下五种原创特色功能。

1.5.2.1 高频词统计功能，其将根据聊天记录生成词云图，词云图中的词语文字大小对应其在聊天记录中的出现频率。

1.5.2.2 用户关系分析功能，其将根据聊天记录推测二者的可能关系。

1.5.2.3 导出为 PDF 功能，其可将聊天记录导出为 PDF。

1.5.2.4 聊天时间轴功能，其将显示二者各时间段的聊天频率。

1.5.2.5 反馈与建议功能，点击后打开反馈界面（Feedback Dialog），用户可输入反馈或建议，直接发送至开发者邮箱。

## 1.5.3 消息右键菜单

对某则消息右键将显示消息右键菜单（Message Context Menu Handler），其中包含以下五种功能。

1.5.3.1 复制功能，可复制该条消息的文本内容。

1.5.3.2 朗读功能，可朗读该条消息。

1.5.3.3 翻译功能，可对该条消息进行翻译（中英翻译）。

1.5.3.4 搜索功能，可打开默认浏览器搜索该消息内容。

1.5.3.5 提醒功能，可设置绝对时间，达到该时间后将弹出含该消息的弹窗进行提醒。

## 1.6 关闭窗口

点击右上角关闭按键后将打开退出确认窗口（Close Confirm Dialog）（图 1-6）。点击“最小化”可将软件最小化至托盘；点击“不再提示”可记住此次选择（最小化或退出），下次将不再展示关闭窗口。



图 1-6 退出确认窗口

## 1.7 托盘显示

两应用程序图标均将通过托盘管理器（Tray Manager）于托盘中显示（图 1-7），鼠标悬停将显示“xxx 正在后台运行”，右键后可选择打开或退出程序。

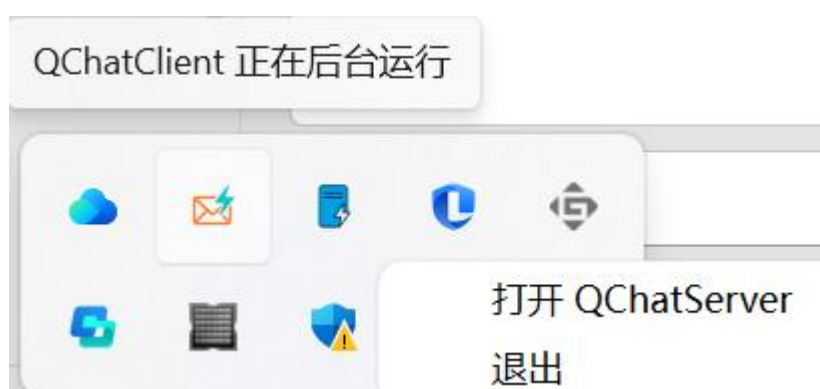


图 1-7 托盘显示

# 二、项目细节

## 2.1 欢迎界面

使用 Qt 自带的 Quick3D 模块编写了 welcome.qml 作为欢迎界面（图 2-1），其将在程序运行时即展示旋转的软件图标，下方显示“欢迎使用 QChat”字样，且背景区均为透明。

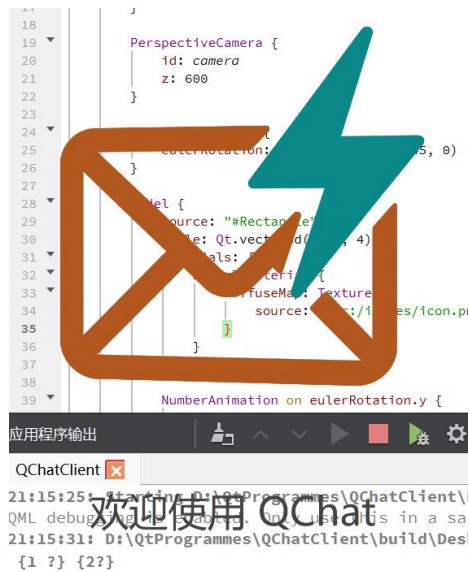


图 2-1 欢迎界面

## 2.2 关于账号

### 2.2.1 关于邮箱与验证码

2.2.1.1 邮箱格式是否正确的判断：使用正则表达式，判断邮箱名、主机名、域名、顶级域格式是否正确。

2.2.1.2 验证码的发送：客户端发送邮件至服务器，服务器侧使用 python 的 smtplib 库和 email.mime.text 库将随机生成的验证码发送至用户邮箱，为保证兼容性已将.py 文件转换为.exe 文件。

### 2.2.2 关于用户

所有已注册用户的头像与昵称均将显示于服务器的好友列表中，便于快速查找。

同时，所有已注册用户的头像、昵称、密码、邮箱、兴趣等信息均将通过服务器的 User Auth Database Manager 类存储于服务器侧的 user 数据库（使用 SQLite 实现）的 users 表格中。服务器将按注册顺序对每个用户进行唯一的编号。

### 2.2.3 关于好友

用户的好友添加或删除信息将传输至服务器，服务器使用 User Auth Database Manager 类判断操作是否成功，并在 user 数据库的 friends 表格中存储好友关系数据。

### 2.2.4 关于头像

用户从电脑文件中选择图片，通过自行编写的 Avatar Cropper 类截取其中正方形部分作

为头像。

## 2.3 关于通信

消息通信为本项目的核心内容，服务器和客户端分别使用 `Server` 类和 `Client` 类实现。

### 2.3.1 关于联网

2.3.1.1 局域网部署情形：由于校园网等内网可能由于设置了禁止端与端之间的通信，于多台电脑运行的服务器与客户端间无法正常通信。经测试，使用个人热点可实现两端间的有效通信。

2.3.1.2 非局域网部署情形：开发者已租用具备近一年有效期的专属隧道，通过运行 `Natapp` 客户端并输入授权密钥，即可在服务器端快速建立一条公网可访问的安全隧道。客户端无需处于与服务器相同的局域网中，只需将主函数中地址修改为连接该公网隧道所分配的地址，即可完成与服务器的有效通信。

### 2.3.2 关于普通文本消息传输

文本消息的传输及处理经历以下几个步骤：

发送端：（1）调用 `write` 函数将文本消息传输至接收端；（2）调用 `flush` 函数保证立即发送；（3）`chatPage` 调用 `updateMessage` 函数在聊天记录显示区更新消息；（4）`dbManager` 调用 `insertMessage` 函数将消息存储于数据库中。

接收端：（1）触发 `onReadyRead` 槽函数，根据消息前缀判断所接收消息是否为文件；（2）由于仅为文本消息，直接进入 `handleTextMessage` 函数，根据消息前缀判断所接收消息为通信信息或普通文本信息；（3）由于消息仅为普通文本消息，触发 `updateMessage` 函数与 `insertMessage` 函数将消息呈现于屏幕，同时存储于数据库。

### 2.3.3 关于文件传输

发送端在文件开头加入“`File:`”前缀，并在其后添加文件名，“`:`”，文件大小，“`\n`”，文件内容，接收端将对其进行解析分割，并反复调用 `onReadyRead` 函数与 `tryFinishFile` 函数进行接收。

由于文件可能较大，接收端将临时建立 `FileInfo` 结构体，其中存储文件名、文件内容、文件大小、文件是否正在接收、文件头是否已接收，用于 `onReadyRead` 函数与 `tryFinishFile` 函数中的判断语句，以正确接收文件内容。

### 2.3.4 关于通信消息传输

发送方仅需在文本消息开头加入特定前缀，不同单位信息（如注册信息中的昵称、密码、



邮箱) 间以“|”隔开, 接收方接收时先通过前缀判断消息类型, 再通过分隔符对消息进行解析, 即可有效接收通信消息。

### 2.3.5 关于 TCP 粘包问题

由于 Qt 自带的 QTcp 相关类无法彻底解决 TCP 粘包问题, 仅可人为进行优化。发送方在特定大文件 (如用户头像、好友头像) 发送后, 再发送一则消息即“PASS”标志, 接收方对其进行解析并忽略, 可在一定程度上减少 TCP 粘包问题对消息传输的影响。

## 2.4 语音转文字

通过使用 QMediaRecord 类的 record 函数录制音频后, 调用 speech\_to\_text.py 文件, 将音频转换为适配模型的 16 kHz, 单声道, 16 位 wav 文件, 并调用离线的 Vosk 模型即可实现语音转文字。

## 2.5 前端细节

### 2.5.1 匹配界面背景

通过使用 QMovie 类, 在开始匹配时背景循环播放 gif 图像, 匹配成功、停止匹配或切换界面时停止播放, 提升好友匹配体验。

### 2.5.2 鼠标悬停

2.5.2.1 鼠标悬停于按钮时, 按钮样式由“红-浅红渐变”变为“浅紫-蓝渐变”(图 2-5-1)。

2.5.2.2 鼠标悬停于好友时, 该好友对应行也将变色; 鼠标悬停于好友头像时, 通过调用临时存储的头像文件, 将显示清晰放大后的原头像 (图 2-5-2)。



图 2-5-1 鼠标悬停于按钮, 正在输入的输入框



图 2-5-2 鼠标悬停于好友及好友头像

### 2.5.3 输入可视化

正在输入时，输入框外边缘将显示为红色，其他输入框外边缘显示为灰色。

## 三、小组成员分工

张嵩仁同学负责大部分后端内容的实现（如通信、数据库操作等），少量前端内容的美化（如布局等），以及部分图标绘制；杨锐同学负责绝大部分前端内容的实现（如控件样式等），部分后端内容的实现（如匹配界面实现逻辑等），以及部分图标绘制。

小组报告、演示视频录制由两人共同完成，视频剪辑及路演演讲由张嵩仁同学完成。

## 四、总结与反思

### 4.1 项目总结

项目实现了实时通信的核心功能，并在此基础上加入了许多原创的便捷功能，显著提升了用户体验。以下为项目的主要亮点：

（1）完善的注册与登录体系：支持昵称密码注册、昵称/邮箱+密码登录，以及邮箱验证码登录等多种方式，确保了用户身份的安全性和便捷性。

（2）个性化设置：用户可以自由修改头像、昵称、密码、聊天背景等，增强了软件的个性化体验。

（3）好友匹配系统：通过兴趣匹配推荐好友，为用户提供了更智能的交友方式。

（4）丰富的聊天功能：支持文本、文件、即时拍照、语音转文字、延迟发送等多种消息类型，并集成了高频词统计、用户关系分析等独特原创功能。

（5）优化的用户体验：包括鼠标悬停效果、输入可视化效果、右键工具、托盘显示等细节设计，进一步提升了软件的易用性与美观性。

（6）技术实现：项目采用了 Qt 的网络模块、通信模块、Quick3D 模块、SQLite 数据库、Vosk 语音识别模型、中英翻译模型、内网穿透、邮件验证码发送、FTP 模拟、消息增强等复杂技术，实现了许多便捷且高效的功能，展现了较强的技术能力。

### 4.2 反思与展望

由于 TCP 粘包问题的不可避免性，两端之间在 1 秒内快速传输多个文件较为困难，即对于好友头像的快速更新过程仍有缺陷；且多用户同时运行时，可能会出现不可避免的延迟等问题。我们设想可通过开设更多端口进行并行传输，或选择其他有效的方法解决该问题。

软件发布后我们也将继续精进代码并修改可能的缺陷，以不断提升我们的程序设计能力。