

# 基于 QT 和 UDP 的聊天室设计与实现

杜 聪 白俊鸽

成都锦城学院计算机与软件学院 四川 成都 611731

**【摘要】**在当前网络迅速发展的时代下诞生了许多聊天通信软件,并广泛应用于人们的日常生活与工作中,为人们彼此之间沟通带来了极大的方便。通过结合 Qt Creator 图形界面开发环境和无连接的 UDP 传输协议,介绍基于 QT 和 UDP 的聊天室设计与实现,开发出一款简单、高效、传输速度快、方便大众使用的产品。

**【关键词】**聊天室; QT; UDP

## 1 引言

随着聊天软件遍及到人们生活的方方面面,为大众的的日常交际与学习带来了极大的方便,并扮演着不可或缺的角色。就现在的聊天通信软件来说,如 QQ,微信等通讯软件早已家喻户晓。但其实对于聊天来讲,功能太过复杂,很多人打开微信、QQ 可能不是为了聊天,是被其它功能所干扰,甚至失去了聊天的本质,所以开发一款简洁、高效、专注于聊天通信方面的软件势在必行。本项目基于 Qt 开发环境和无连接的 UDP 传输协议制作一个局域网内的一对一聊天室,可实现一对一的聊天通信。

## 2 开发环境介绍

QT 是一个能够跨平台开发的 C++ 图形用户界面应用程序框架。<sup>[1]</sup>利用 Qt 开发的软件能够在目前的大部分的平台上编译和运行,并且不需要额外的更改源代码。运用 QT 开发的软件会根据其使用开发平台的不同,从而可以表现出该平台所特别持有的图形化界面风格。除此之外,多平台桌面应用程序的开发,移动端的开发和嵌入式端的开发 QT 都能够进行,几乎包含了目前市面上存在的大部分常用平台。QT 可以实现通信功能,如果要实现网络编程功能,可以引入内部提供的 QTNetwork 模块。除此之外,还支持各种网络协议,比如 FTP、DNS、HTTP,和 TCP/IP 与无连接 UDP 传输协议。强大的 QT 开发平台内部含有如 QUdpSocket 类、QTcpSocket 类和 QTcpServer 等一系列提供给开发人员的类。

## 3 技术介绍

UDP 是 OSI 七层参考模型中一个无需建立连接的传输层协议。<sup>[4]</sup>UDP 是一种简单快捷,不需要进行确认接收和流量控制的传输协议。访问网络中的程序需要绑定一个端口号,端口号是 UDP 中一个最重要的信息,其实就是一个数字就可以代表一个端口号<sup>[3]</sup>。相比于较为安全的 TCP 协议,无连接 UDP 在使用时不用去建立相互之间的连接,拥有数据传输速度快,系统资源开销相对较小等优点。因为不需要利用额外的系统资源花在建立连接、保护连接和差错管制上,这样就可以防止时间延迟的发生,同时比较于 TCP 协议,UDP 协议还可以支持网络广播的功能,这就非常适合于日常生活中网络视频的通讯。<sup>[2]</sup>所以对于本项目来说,采用 UDP 连接,是由于不需要去判断对方是否实时存在,也就是说即便接收端不存在或

无法接纳数据,发送端也能将数据成功发送到接收端,不进行接收确认,传输速度快,简单,高效。

## 4 聊天室的设计

### 4.1 界面设计

本项目采用 QT 中自带的图形化界面开发功能来完成界面设计,分为三个主要界面:主界面,设置界面,显示消息界面。其中主界面用来完成消息收发和显示,设置界面用来设置聊天对象的 IP 地址和端口号,显示消息界面其实就是一个很简单的 textEdit 文本编辑框和 label 组成,用来在主界面中显示消息收发内容,和聊天对象的主要信息(IP 地址和端口号)。如图 1 所示为本项目界面实现。



图 1 界面实现图

该界面设计虽然比较简洁,但易于实现与操作,可以实现本项目基本功能。

### 4.2 功能流程设计

项目的主要功能就是进行一对一的信息传输,在主界面中点击“设置”按钮进入设置界面窗口。经过输入框输入待发送信息的对方端口号和 IP 地址,点击“确认”按钮,即可开始发送消息。随后进入主界面,将会看到发送方和接收方 IP 地址和端口号信息,在主界面中的输入框中输入待发送的信息,点击“发送”按钮。若发送成功,发送的所有内容会显示到右上方的文本控件中,

同时若发送失败,则会通过调用 QMessageBox 类中函数在界面中弹出提示框,提示用户发送失败。

## 5 聊天室功能实现

第一步是先导入头文件,注意需要添加 network 模块,导入头文件之后,就可以开始本项目的开发。利用 QUdpSocket 类中的构造函数创建一个套接字,创建成功之后就会返回一个 QUdpSocket 类型的套接字。如果要进行无连接 UDP 数据接收,要实现通信功能,首先需要绑定一个活动端口号,即调用 QUdpSocket 类中的 bind() 函数为在构造函数创建的 QUdpSocket 套接字绑定一个本地端口号。一般可以在构造函数中调用 QUdpSocket 类中的 bind() 函数。假如不需要接收消息,只需要发送消息,则无需调用 QUdpSocket 类中的 bind() 函数。本项目是基于无连接 UDP 协议,按照之前的项目功能设计要求,编写了以下两个重要的函数,分别负责发送消息和接收消息。除此之外, getLocalIP() 函数用来获取本地 IP 地址, toIPv4() 函数用来将 IP 地址固定为 IPv4 格式, slotCreateObject() 函数用来创建聊天对象, getInfoIP() 函数用来保存对方 IP 地址和端口号。通过完成这些函数,从而实现该项目基本功能。

### 5.1 发送消息

发送消息功能由 toSendMessage() 函数来完成。

```
char buf[512]={0};

QHostAddress clientAddress;
quint16 port;
quint64 len=m_udpSocket->readDatagram(buf,sizeof(buf),&clientAddress,&port);
QString IP=toIPv4(clientAddress);

if(len>0)
{
    QString content = QString("[%1:%2 \n%3").arg(IP).arg(port).arg(buf);
    QString Port=QString("%1").arg(port);

    QListWidgetItem* receiveItem=new QListWidgetItem();
    Bubble* receiver=new Bubble();

    receiver->setText(content);
    receiveItem->setSizeHint(QSize(200,120));

    ui->chatWindow->addItem(receiveItem);
    ui->chatWindow->setItemWidget(receiveItem,receiver);
}
```

图2 接收消息核心代码图

### 5.3 其它功能实现

#### 5.3.1 获取本地 IP 地址

项目中 getLocalIP() 函数用来获取本地 IP 地址,如果是在电脑命令行界面中,可输入“ipconfig”命令查看本地 IP 地址,但是在本项目中需要在 QT 的编程环境下编写一个函数用来自动获取本地 IP 地址并能够保存下来。在这个函数中,通过 QT 中自带的 QProcess 类运行命令行界面,调用“ipconfig”命令,然后通过正则表达式来判断是否具有 IP 地址的格式,最后需要使用一个 if 筛选出本机实际 IP 地址,即除去子网掩码。

#### 5.3.2 IP 地址固定为 IPv4 格式

IPv4 协议是互联网的核心内容,也是目前使用最广泛的网际协议版本,所以本项目都采用 IPv4 协议。通过 QT 中的 toIPv4Address() 函数就可以将 IP 地址转化成 IPv4 格式,然后通过函数返回值返回一个 IPv4 格式的字符串。

在 toSendMessage() 函数中,首先需要获取接收方的 IP 地址和端口号,然后再通过 toPlainText() 函数获取在输入框待发送的消息内容。在发送消息这一过程完成中,为了增加代码的健壮性,需要对输入内容添加限制条件。第一,如果没有在输入框输入消息内容,则不能发送消息(无法点击“发送”按钮)。第二,如果没有设置接收方 IP 和端口号不能发送消息。若通过上述判断后,使用 QUdpSocket 类中的 writeDatagram() 函数,就会发送消息到接收方中。调用完成后,将发送的消息显示到软件界面右上方的文本对话框中。发送消息完成后,调用 clear() 函数清空输入框,并调用 setFocus() 函数使光标聚焦到输入框中,以便在下次发送新的消息。

### 5.2 接收消息

接收消息功能由 receMessage() 函数来完成。在 receMessage() 函数中,首先需要获取发送方的 IP 地址和端口号,然后通过调用 QUdpSocket 类中的 readDatagram() 函数接收发送方发来的消息并保存下来。在接收消息过程当中,为了增加代码的健壮性,需判断发送方发来的消息是否为空,若为空则不显示到界面上。若不为空,则将接收到的消息显示到界面右上方的文本消息对话框中。如图 2 所示,是接收消息函数的核心代码。

#### 5.3.3 创建聊天对象

通过创建聊天对象,获取当前通信对方的信息,主要信息包括聊天对象的 IP 地址和端口号,进行每一次不同的通信就会建立一个聊天对象。在该函数中设置两个 QString 字符串分别用来保存在设置界面中输入的聊天对象的 IP 地址和端口号。通过字符串拼接该 IP 地址和端口号,然后把最后拼接完成的字符串添加到显示消息界面中,最后显示到主界面左上方区域,完成聊天对象的创建。

#### 5.3.4 保存对方 IP 地址和端口号

在项目头文件中创建了一个 QMap 容器,用来保存对方 IP 地址和端口号。函数 getInfoIP() 中有两个参数,分别为 IP 地址和端口号,通过 QMap 容器中的 insert 方法为其插入 IP 地址和端口号,完成对 QMap 容器的初始化,同时保存了保存对方 IP 地址和端口号,方便项目中其它

函数使用该信息。

## 6 测试及结论

项目完成之后,对本聊天室软件采用黑盒测试的方式进行了功能性测试。最后经过测试结果表明,发送消息和接收消息作为本项目主要的功能均可正常使用,在软件主界面中,收发消息内容均可正常显示。即该聊天室软件基础功能均能正常使用。

但相对于目前市面上其他更加完善的聊天通信软件,本聊天室软件还存在许多的不足。后期还需做进一步深入和完善,计划增加文件传输功能,使用户在使用本软件时既可以收发消息,又可以完成文件的传输。计划增加多对多通讯功能,使用户在使用本软件时可以进行多对多的通信交流。通过增加以上两个功能,可以完善软件功能,并进一步优化用户体验。

## 7 结束语

本文介绍了基于QT和UDP的聊天室设计与实现,结合UDP无连接协议传输速度快,实现简单,高效的特点,打造出一款可以用于人们日常交际与工作的通讯软件。

当然目前在产品可用度上,该软件还需做进一步深入和完善,如实现文件传输,多对多通讯功能等。总体来说,该软件实现简单,代码逻辑清晰简单,功能较为完善,开发难度低,比较适合QT初学者练手和使用。

## 【参考文献】

- [1] 宋冲亚,李爱菊. QT局域网聊天项目制作 [J]. 现代物业 (上旬刊), 2012, 11 (01): 25-27.
- [2] 迟浩. 基于UDP的多链路协同通信算法的研究 [D]. 上海师范大学, 2018.
- [3]. 简析IP、UDP和TCP三种协议的关系 [J]. 电脑知识与技术 (经验技巧), 2020 (04): 90-91.
- [4] 陈杰华. 基于状态检测的硬件防火墙的设计 [D]. 福州大学, 2011.