# 摘要

如今是一个移动互联网的时代，每个人都有他/她的手机。在日常生活中手机的比重越来越大，人们每天基本上都会打开QQ、微信跟朋友、陌生人、同事、家人进行交流和沟通，所以即时通讯越来越重要，而且即时通讯在很多方面都有应用，如电商、教育、企业、论坛等方面都有应用。

即时通讯（Instant messaging，简称IM）是一个终端服务，允许两人或多人使用网络进行传递文本、图片、文件、音视频等进行交流和沟通。如今手机市场中操作系统繁多，而Android系统是一种基于[Linux](https://baike.baidu.com/item/Linux)的自由及开放源代码的[操作系统](https://baike.baidu.com/item/%E6%93%8D%E4%BD%9C%E7%B3%BB%E7%BB%9F" \t "_blank)，主要使用于[移动设备](https://baike.baidu.com/item/%E7%A7%BB%E5%8A%A8%E8%AE%BE%E5%A4%87)。而且Android操作系统在手机行业的比重也相当巨大，基本上10人中有8人使用Android系统，所以一款基于Android的软件也会有广大的用户群体。

本文首先介绍课题的研究背景,对于Android系统、即时通讯软件和即时通讯协议做一个简单的介绍。然后详细的介绍Android的底层架构及其特征以及Android

关键词：IM，即时通讯

# Abstract

Today is the era of mobile Internet, everyone has his/her mobile phone.The proportion of cell phones in their daily lives more and more big, people will open the QQ, micro letter basically every day with friends and strangers, colleagues, family communication and communication, so im more and more important, and are used in many ways, such as electricity, education, enterprise, BBS etc are used.

Instant messaging (IM) is a terminal service that allows two or more people to communicate and communicate text, pictures, files, audio and video using the network.Today, the mobile phone market is full of operating systems, while Android is a free and open source operating system based on Linux, mainly used in mobile devices.And the Android operating system is a huge part of the mobile phone industry, with 8 out of 10 people using Android, so an android-based software will also have a large user base.

Key words: IM, instant messaging.

# 绪论

本章节主要内容为课题的研究背景，国内外发展现状，研究的内容和意义以及本轮文的组织结构。

## 课题研究背景

如今是移动互联网的时代，手机遍地都是，不仅价格优惠，性能还挺好所以基本上每个人都拥有自己的手机，上至六、七十岁的大妈，下至五、六岁的小孩，且每天玩手机的时间平均在四到五小时作用，如果是那种没工作的或许整天都在玩手机，而且人们打开即时通讯软件的频率也相当高，像国内的QQ和微信，在人们日常生活中已经离不开了，人们每天都会打开QQ或微信看是否有好友、同事、亲人、陌生人给自己发消息，自己也给他们发消息，进行每天的交流和沟通。

而在手机操作系统中，主要有三大操作系统为主流，如Android、IOS、Windows Phone，而Android系统在手机操作系统中占据极大的比重，主要原因如下：

一、Android是基于Linux的开源的操作系统，开放的平台吸引越来越多的开发者，开源的代码库、免费的开发软件、社区、第三方开源共享，在带来巨大的竞争的同时也使得android在开放的平台中显得日益成熟；

二、android在终端天生就有网络特色，也使用户离互联网更近，主要也是同时网络的不断发展，从2G、3G到如今的4G和以后更加强大的网络，也因此带来更多强大的用户体验；

三、由于Android的开放性，很多厂商为了达到更加吸引用户的目的，会对android的基础上加以改造，推出功能特色各具的各种产品，而同时不会影响到数据同步、甚至软件的兼容（例如：三星、华为、小米等具有自己不一样的功能特色），用户体验也不断丰富；

四、由于android的开放性使得第三方开发商可以自由的开发需要的软件，众多的开源代码库也使得开发变得更加简单方便，软件的功能也是不断的推陈出新，功能不断的强大，也使得如今android的软件占据着最大的应用市场。

五、android手机大部分都价格低廉，普通人也可以承受的起这低廉的消费，所以用户群体庞大，且Google提供了开发者很好的软件发布平台google play。

开发一款基于android的即时通迅软件会有广大的用户群体，且即时通讯在人们日常生活中的地位也相当的高，可见即时通讯软件有很好的前景。

## 国内外研究现状

自1998年面世以来，即时通迅已经发展了20年。如今即时通讯技术以相对成熟，用来开发即时通讯的协议也有三、四种，如XMPP、IMPP、scoket等协议。而国内专门做即时通讯技术的厂商也有不少，如腾讯、网易、环信、融云等大厂商提供即时通讯的API服务。

经过这20年的发展，即时通讯已经不在仅仅是在交流这一方面运用，如今已经集成了电子邮件、电商、游戏、教育、企业办公、娱乐等为一体化的综合信息平台。

而如今国内电商领域，如淘宝、京东等，主要用于买家和卖家对商品的讨价还价以及商品各属性的了解。在游戏领域，如王者荣耀、阴阳师等，主要用于玩家与玩家进行游戏的心得和攻略的交流，以及叫好友一起开黑。在娱乐领域，如新浪微博，主要用于发布一些新的资讯以及用户发布自己的一些文章或动态。在企业办公主要用于企业内部员工之间信息传递，会议沟通、交流协作等。而在教育方面主要用于老师在线上给学生进行知识的传授，学生有疑惑的地方可以在线上与老师进行沟通交流，使人们可以非常方便的学到来自全世界各地的知识。在交流方面，如QQ、微信是国内用于人们日常交流最常用的两款APP，主要给人们提供文字聊天、发送图片、发送文件、群聊、空间、朋友圈等功能来丰富人们的日常生活，大大方便了人与人之间的交流。

国外比较火的即时通讯为MSN、facebook、youtube、twitter等占据了国外即时通讯的大部分市场，而且国内也有几百甚至几千万用户使用VPN去访问国外的这些网站。

## 研究的内容及意义

即时通讯是利用互联网进行数据的传输，能够传输文本、图片、音视频等格式的消息来进行交流，这大大节省了通讯双方的经济成本和时间成本。

如今是移动互联网时代，各大PC端的软件都有相应的移动端的产品，开发移动端产品使人与人之间的交流更加方便、快捷。只要有一台手机，在手机上安装了即时通讯的APP，连接网络后便可随时随地与人进行交流、互动。对人们日常生活的信息传递更加便利、也可以远距离增加人与人之间的感情。

如今即时通讯技术也非常成熟，专一做即时通讯的各大厂商都提供了相应的API服务，只要选择一个厂商提供的API服务便可快速开发出一款即时通讯APP。本文选择使用网易云信提供的API服务开发即时通讯APP，或许会有人问为什么要选择厂商提供的API来进行开发，而不使用即时通讯的通讯协议来进行开发。原因如下：

一、使用通信协议开发不是不行，主要太耗费时间、耗费精力。主要本系统是由一个人来开发完成的如果使用通信协议需要耗费太大的时间和精力，而且在企业中也不会自己去使用协议去进行开发，哪样企业需要发费太多的人力和物力，没法带来太多的收益，且一个APP开发周期本来就短，所以使用厂商提供的API来开发本系统。

二、使用厂商提供的API，不仅能大大剪短开发周期，也能提高开发效率，对企业带来的利润是巨大的，节省了大量的人力和物力。开发也起来非常简单、高效、便捷。

## 论文组织结构

# 相关理论及技术介绍

本章节主要介绍即时通讯的相关理论基础、以及特点和常用的通讯协议，本文使用的网易云信的API主要使用UDP和TCP协议进行传输，重点讲解UDP和TCP协议。然后介绍Android相关的理论基础、系统架构、四大组件等。

## 即时通讯概述

即时通讯（Instant messaging，简称IM）是一种基于网络的实时沟通的服务。1996年，第一款即时通讯软件由三个以色列人开发。1998年，即时通讯面世。经过20多年的发展，IM已经相当成熟了，且其应用领域也非常广泛。即时通讯丰富了各个软件，如果一个软件没有即时通讯，那么它的使用频率必定不高，一旦加入即时通讯，该软件就可推送很多用户感兴趣的许多消息，也可以实现人与人之间的交流，这样更能提高用户打开APP的欲望。即时通讯不仅可以用来吸引人流量，也方便了人与人之间的交流，所以即时通讯在APP中占据了一定的地位。

### 即时通讯的特点

即时通讯具有以下特点：

（1）实时性。只要用户安装即时通讯APP，且添加了好友，便可以在任何时间、任何地点与好友进行通讯，只要发送端一发出消息，接收端便会立马接收到消息，即时通讯可以实时的传递消息。

（2）通讯录管理。用户可以在即时通讯APP中的通讯录的活动中看见所有的好友，可以查看好友的资料信息，也可以删除好友，还能查看好友的在线状态，如果好友在线就可以立马跟好友进行交流沟通；如果好友不在线，可以给好友留言，当好友上线后便可立马看到发送端发送的消息，并对发送端进行回复。

（3）多线程处理机制。如用户在进行一些基本设置时，也可以接收到好友发送的消息，也可以进行一些其它的操作。

（4）简单易用。在即时通讯APP中，只要我们进入好友互相发送消息的活动页面中，用户便可以发送文本消息、图片消息、还可以使用手机进行拍照，然后发送拍摄的图片、还可以发送文件，满足了用户多种消息的发送需求，而且如今即时通讯中发送表情包成为了一大乐趣，所以图片消息更满足了人们的休闲娱乐，可以给用户带来更多欢乐。文件消息满足了人们之间传递大文本的需求，如果自己输入一端超长的文字，发送者需要大量的时间去打字，这样太耗费时间，给人的体验也不怎么好，所以发送文件便可轻松解决上面的问题。以上消息发送起来也非常简单、快捷、易用。

### 即时通讯常用协议

即时通讯发展至今，已是非常重要的互联网应用形态之一，尤其移动互联网时代，它正以无与论比的优势降低了沟通成本和交流门槛，对各种应用形态产生了深远影响。

即时通讯协议,最基础的IM协议为TCP(Transmission Control Protocol)[传输控制协议](https://baike.baidu.com/item/%E4%BC%A0%E8%BE%93%E6%8E%A7%E5%88%B6%E5%8D%8F%E8%AE%AE)和UDP(User Datagram Protocol)用户数据报协议；主流的IM协议有4种,IMPP(Instant Messaging And Presence Protocol)即时消息和存在协议, PRIM(Presence and Instant Messageing Protocol)存在和即时消息协议,SIMPLE(SIP for Instant Messaging and Presence Leveraging Extensio ns)即时消息和存在扩展协议,XMPP(Extensible Messageing and Presence Pr otocol) 可扩展消息与存在协议。

#### 2.1.2.1 UDP协议

UDP是一个简单的面向数据报的传输层协议：进程的每个输出操作都正好产生一个UDP数据报，并组装成一份待发送的IP数据报。

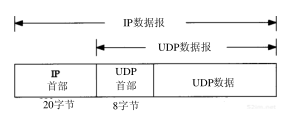
UDP数据报封装成一份IP数据报的格式如图2-1所示。

图2-1 UDP数据封装格式图

RFC 768[Postel 1980]是UDP的正式规规范。UDP不提供可靠性：它把应用程序传给IP层的数据发送出去，但是并不保证它们能到达目的地。

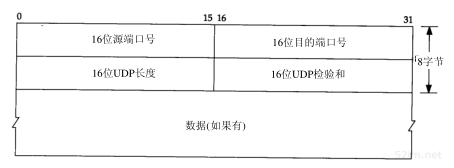
 UDP首部的各字段如图2-2所示。

图2-2 UDP首部字段图

UDP端口号用来表示发送进程和接收进程。

UDP长度字段指的是UDP首部和UDP数据的字节长度。该字段的最小值为8字节（发送一份0字节的UDP数据报是OK）。这个UDP长度是有冗余的。IP数据报长度指的是数据报全长，因此UDP数据报长度是全长减去IP首部的长度（该值在首部长度字段中指定）。

UDP检验和是一个端到端的检验和。它由发送端计算，然后由接收端验证。其目的是为了发现UDP首部和数据在发送端到接收端之间发生的任何改动。

UDP的检验和是可选的。尽管UDP检验和是可选的，但是它们应该总是在用。在80年代，一些计算机产商在默认条件下关闭UDP检验和的功能，以提高使用UDP协议的NFS（Network File System）的速度。在单个局域网中这可能是可以接受的，但是在数据报通过路由器时，通过对链路层数据帧进行循环冗余检验（如以太网或令牌环数据帧）可以检测到大多数的差错，导致传输失败。不管相信与否，路由器中也存在软件和硬件差错，以致于修改数据报中的数据。如果关闭端到端的UDP检验和功能，那么这些差错在UDP数据报中就不能被检测出来。另外，一些数据链路层协议（如SLIP）没有任何形式的数据链路检验和。

#### 2.1.2.2 TCP协议

尽管TCP和UDP都使用相同的网络层（IP），TCP却向应用层提供与UDP完全不同的服务。TCP提供一种面向连接的、可靠的字节流服务。

面向连接意味着两个使用TCP的应用（通常是一个客户和一个服务器）在彼此交换数据之前必须先建立一个TCP连接。这一过程与打电话很相似，先拨号振铃，等待对方摘机说“喂”，然后才说明是谁。

TCP通过下列方式来提供可靠性：

应用数据被分割成TCP认为最适合发送的数据块。这和UDP完全不同，应用程序产生的数据报长度将保持不变。由TCP传递给IP的信息单位称为报文段或段（segment）。

当TCP发出一个段后，它启动一个定时器，等待目的端确认收到这个报文段。如果不能及时收到一个确认，将重发这个报文段。在第21章我们将了解TCP协议中自适应的超时及重传策略。

当TCP收到发自TCP连接另一端的数据，它将发送一个确认。这个确认不是立即发送，通常将推迟几分之一秒。

TCP将保持它首部和数据的检验和。这是一个端到端的检验和，目的是检测数据在传输过程中的任何变化。如果收到段的检验和有差错，TCP将丢弃这个报文段和不确认收到此报文段（希望发端超时并重发）。

既然TCP报文段作为IP数据报来传输，而IP数据报的到达可能会失序，因此TCP报文段的到达也可能会失序。如果必要，TCP将对收到的数据进行重新排序，将收到的数据以正确的顺序交给应用层。

既然IP数据报会发生重复，TCP的接收端必须丢弃重复的数据。

TCP还能提供流量控制。TCP连接的每一方都有固定大小的缓冲空间。TCP的接收端只允许另一端发送接收端缓冲区所能接纳的数据。这将防止较快主机致使较慢主机的缓冲区溢出。

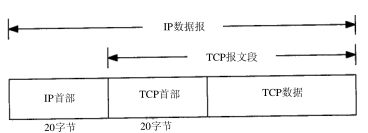
TCP数据被封装在一个IP数据报中，如图2-3所示。

图2-3 TCP数据封装格式图

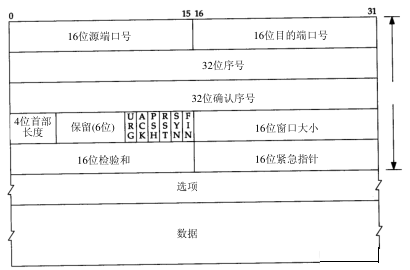
图2-4显示TCP首部的数据格式。如果不计任选字段，它通常是20个字节。

图2-4 TCP首部图

每个TCP段都包含源端和目的端的端口号，用于寻找发端和收端应用进程。这两个值加上IP首部中的源端IP地址和目的端IP地址唯一确定一个TCP连接。

序号用来标识从TCP发端向TCP收端发送的数据字节流，它表示在这个报文段中的的第一个数据字节。如果将字节流看作在两个应用程序间的单向流动，则TCP用序号对每个字节进行计数。序号是32 bit的无符号数，序号到达232－1后又从0开始。

## Android概述

### Android系统简介

Android是一种基于[Linux](https://baike.baidu.com/item/Linux" \t "_blank)的自由及开放源代码的[操作系统](https://baike.baidu.com/item/%E6%93%8D%E4%BD%9C%E7%B3%BB%E7%BB%9F" \t "_blank)，主要使用于[移动设备](https://baike.baidu.com/item/%E7%A7%BB%E5%8A%A8%E8%AE%BE%E5%A4%87)，如[智能手机](https://baike.baidu.com/item/%E6%99%BA%E8%83%BD%E6%89%8B%E6%9C%BA)和[平板电脑](https://baike.baidu.com/item/%E5%B9%B3%E6%9D%BF%E7%94%B5%E8%84%91)，由[Google](https://baike.baidu.com/item/Google)公司和[开放手机联盟](https://baike.baidu.com/item/%E5%BC%80%E6%94%BE%E6%89%8B%E6%9C%BA%E8%81%94%E7%9B%9F" \t "_blank)领导及开发。如今Android已经发展了15年，发布了28个版本了。在这15年中，Google为Android建立了完整的生态系统。厂商、开发者、用户共同推进着Android的蓬勃发展。

### Android系统架构

Android系统架构大致分为四层：Linux内核层、系统运行库层、应用框架层和应用层。

1. Linux内核层

Android的核心系统服务依赖于Linux2.6内核，如安全性，内存管理，进程管理，网络协议栈和驱动模型。为Android设备提供底层驱动，如wifi驱动、蓝牙驱动、网络驱动等。

1. 系统运行库层

Android 包含一些C/C++库，这些库能被Android系统中不同的组件使用。它们通过Android应用程序框架为开发者提供服务。

Android包括了一个核心库，该核心库提供了JAVA编程语言核心库的大多数功能。每一个Android应用程序都在它自己的进程中运行，都拥有一个独立的Dalvik虚拟机实例。Dalvik被设计成一个设备可以同时高效地运行多个虚拟系统。Dalvik虚拟机执行（.dex）的Dalvik可执行文件，该格式文件针对小内存使用做了优化。同时虚拟机是基于寄存器的，所有的类都经由JAVA编译器编译，然后通过SDK中的"dx" 工具转化成.dex格式由虚拟机执行。

1. 应用框架层

开发人员也可以完全访问核心应用程序所使用的API 框架。该应用程序的架构设计简化了组件的重用；任何一个应用程序都可以发布它的功能块并且任何其 它的应用程序都可以使用其所发布的功能块（不过得遵循框架的安全性限制）。同样，该应用程序重用机制也使用户可以方便的替换程序组件。

1. 应用层

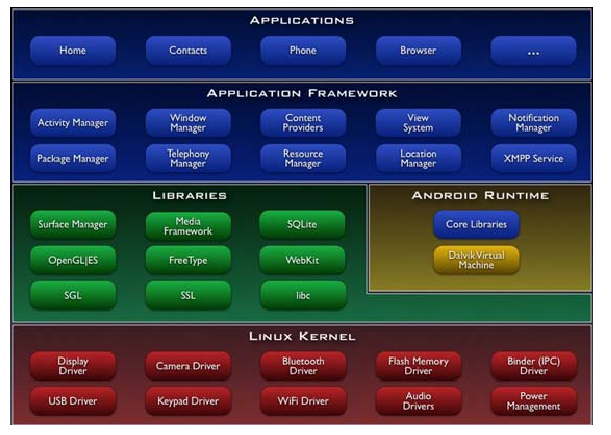
所有安装在手机上的APP都属于这一层，如QQ、微信等

图2-5 Android系统架构图

### Android四大组件

Android四大组件为：活动(Activity)、服务(Service)、广播接收器(Broadcast Receiver)、内容提供器(Content Provider)。

1. 活动

Android中，Activity是所有程序的根本，所有程序的流程都运行在Activ ity之中，Activity可以算是开发者遇到的最频繁，也是Android当中最基本的模块之一。在Android的程序当中，Activity一般代表手机屏幕的一屏。如果把手机比作一个浏览器，那么Activity就相当于一个网页。在Activity 当中可以添加一些Button、Checkbox等控件。可以看到Activity 概念和网页的概念相当类似。一般一个Android应用是由多个Activity组成的。这多个Act ivity之间可以进行相互跳转，例如，按下一个Button按钮后，可能会跳转到其他的Activity。和网页跳转稍微有些不一样的是，Activity之间的跳转有可能返回值，例如，从ActivityA跳转到ActivityB，那么当ActivityB运行结束的时候，有可能会给ActivityA一个返回值。这样做在很多时候是相当方便的。

当打开一个新的屏幕时，之前一个屏幕会被置为暂停状态，并且压入历史堆栈中。用户可以通过回退操作返回到以前打开过的屏幕。可以选择性的移除一些没有必要保留的屏幕，因为Android会把每个应用的开始到当前的每个屏幕保存在堆栈中。

1. 服务

Service是android系统中的一种组件，它跟Activity 的级别差不多，但是他不能自己运行，只能后台运行，并且可以和其他组件进行交互。Service 是没有界面的长生命周期的代码。Service是一种程序，它可以运行很长时间，但是它却没有用户界面。当我们打开一个音乐播放器的程序，这个时候若想上网了，那么，打开Android浏览器，这个时候虽然已经进入了浏览器这个程序，但是，歌曲播放并没有停止，而是在后台继续一首接着一首的播放。其实这个播放就是由播放音乐的Service进行控制。当然这个播放音乐的Servi ce也可以停止，例如，当播放列表里边的歌曲都结束，或者用户按下了停止音乐播放的快捷键等。Service可以在和多场合的应用中使用，比如播放多媒体的时候用户启动了其他Activity这个时候程序要在后台继续播放，比如检测SD卡上文件的变化，再或者在后台记录地理信息位置的改变等等，总之服务嘛，总是藏在后头默默工作，默默付出，给我们的APP带来极佳的用户体验。

1. 广播接收器

在Android 中，Broadcast是一种广泛运用的在应用程序之间传输信息的机制。而BroadcastReceiver 是对发送出来的Broadcast进行过滤接受并响应的一类组件。可以使用BroadcastReceiver 来让应用对一个外部的事件做出响应。这是非常有意思的，例如，当电话呼入这个外部事件到来的时候，可以利用BroadcastReceiver进行处理。例如，当下载一个程序成功完成的时候，仍然可以利用BroadcastReceiver进行处理。BroadcastReceiver不能生成UI，也就是说对于用户来说不是透明的，用户是看不到的。BroadcastReceiver通过NotificationManager来通知用户这些事情发生了。BroadcastReceiver既可以在AndroidManifest.xml中注册，也可以在运行时的代码中使用Contex t.registerReceiver()进行注册。只要是注册了，当事件来临的时候，即使程序没有启动，系统也在需要的时候启动程序。各种应用还可以通过使用Contex t.sendBroadcast()将它们自己的Intent Broadcasts广播给其他应用程序。

1. 内容提供器

Content Provider 是Android提供的第三方应用数据的访问方案。

在Android中，对数据的保护是很严密的，除了放在SD卡中的数据，一个应用所持有的数据库、文件等内容，都是不允许其他直接访问的。Andorid当然不会真的把每个应用都做成一座孤岛，它为所有应用都准备了一扇窗，这就是Content Provider。应用想对外提供的数据，可以通过派生Content Provi der类，封装成一枚Content Provider，每个Content Provider都用一个uri作为独立的标识，形如：content://com.xxxxx。所有东西看着像REST的样子，但实际上，它比REST更为灵活。和REST类似，uri也可以有两种类型，一种是带id的，另一种是列表的，但实现者不需要按照这个模式来做，给id的uri也可以返回列表类型的数据，只要调用者明白，就无妨，不用苛求所谓的REST。

# 系统分析和实现

本章节首先对系统进行一个简单的概述，然后对系统的功能需求进行分析，最后对客户端如何实现给出了相应模块的流程图及文字说明。

## 系统概述

如今是一个移动互联网的时代，基本上每个人在任何时间、任何地点都能使用网络。所以即时通讯便可以得到很好的发展，且即时通讯可以即时的帮助人们传递消息，对人们的日常沟通和交流带来了极大的便利，而且大大的节省了人与人沟通的成本。

而作为一款即时通讯APP，必须拥有一些常规的即时通讯功能来满足人们的基本日常交流需求，且必须简单易操作，用户可以快速上手，还要拥有友好、高颜值的UI界面来满足人们的审美需求，毕竟这是一个看脸的时代，只有拥有高颜值才能使用户有使用APP的欲望，没有高颜值用户会不会下载都是问题。

本系统的主要任务是使用网易云信提供的API服务，来实现基于Android系统的即时通讯APP。本系统主要实现了常规的即时通讯的功能，如登入、通讯录管理、添加好友、删除好友、单聊、群聊、发送文本消息、发送图片消息、发送文件、资料设置等功能。

根据软件工程的理念，系统必须进行层次的明确划分，分为界面层、逻辑层、数据持久层。严格采用MVC模式来进行系统的开发和设计，使用各种设计模式，来降低个模块之间的耦合度，提高代码复用率。

## 系统功能分析

即时通讯系统因其的及时性和便利性方便了人与人之间的沟通，给人们的日常交流带来了极大便利，深受广大用户的喜爱。如今是移动互联网的时代，移动APP应用及其丰富。而且越来越多的APP需要继承即时通讯的功能，这样做，一方面，可以为APP中用户提供社交功能，方便用户交流；另一方面，还能给用户推送一些用户感兴趣的消息，吸引用户量。所以即时通讯的利出也是非常明显的。

本文基于Android的即时通讯APP满足即时通讯的基本的功能需求。本系统的基本功能需求如下：

1. 基础的即时通讯功能

传统的即时通讯功能，最基础的功能为文字聊天，这是即时通讯中最基础也最容易实现的。而如今发送表情包是即时通讯中非常火的一个功能，所以还必须拥有发送图片的功能。因此本系统发送消息模块，首先实现发送文本消息和发送图片消息的功能。

1. 发送文件功能

文件发送在即时通讯中的地位也越来越重要。文件传输更加方便人们的工作和传递大文本信息的需求。如果没有文件发送功能，人们在工作中交流和日常要发送大文本或进行协作的时候，就必须一个一个字码出来，这样太耗费时间了。

1. 个人资料设置

如今，在即时通讯中，要认识其它不同区域的陌生人，首先都是通过看用户的资料，然后在考虑是否添加好友，所以设置个人资料方便陌生人可以了解你的一些基本信息，方便对方的互相了解。

1. 好友管理

在不是非常了解一个用户时，我们通过用户的一些基本资料，添加了好友，但在后面的聊天过程中，发现双方不是特别合的来，或对方做出了一些不友好

的事情，我们需要对好友进行删除等其它操作。

1. 群组管理

如今，群聊是一起讨论交流的一个非常重要的功能。在一个亲人群中，大家可以交流最近大家开心的事情，谈一谈家常。在同事群中，方便大家一起讨论工作中遇到的问题，一起讨论工作进度，一起进行会议决策。在技术交流群中，大家可以分享最新的技术知识，帮助他人解决一些技术问题等。群聊对多人进行沟通带来极大的便利。

## 客户端实现

### 登入、注销模块

当用户启动APP后，首先会判断用户是否登入，用户没有登入，会看到一个开机动画，开机动画结束后会跳转到用户登入界面；如果用户已经登入过，就会直接进行到主界面。

在登入界面用户需要输入账号和密码，然后点击登入按钮，系统便会拿到用户输入的账号和密码，然后对账号和密码进行查找和判断，如果账号和密码正确，就将账号和密码通过SharedPreferences来保存在本机中，当用户下次在进入本系统时，用户就不需要登入，可以直接进入到系统的主界面；如果账号和密码错误，则使用Toast在手机屏幕中显示账号或密码错误，来提醒用户

重新输入账号和密码。登入模块流程如图3-1所示。

启动APP

判断用户是否登入

进入主界面

进入启动动画页面

进入登入界面

判断账号和密码是否正确

否

是

否，返回登入界面，并显示错误信息

是

图3-1 登入模块流程图

用户登入系统后，打开左侧的侧滑栏可以看到一个注销按钮，当用户点击注销按钮，用户将跳转到登入模块，并且将用户在SharedPreferences保存的信息清除，用户下次进入APP时将无法直接进入主界面，会先进入登入界面要求用户进行登入操作。用户注销流程如图3-2所示。

主界面

打开侧滑栏

点击注销按钮

登入页面

清除SharedPreferences中保存的用户信息

图3-2 用户注销流程图

### 聊天相关模块

该部分包括聊天功能的具体实现，文本消息的发送和接收，图片消息的发送和接收，文件消息的发送和接收。

#### 3.2.2.1 发送文本消息

当用户在输入框输入消息后，然后点击发送按钮，就会将输入框中的内容封装成一个文本类型的消息对象，消息对象中包括聊天类型有单聊和群聊，有发送的消息，有接收人的账号，然后发送给接收端。当消息发送之后还需在聊天界面将发送的消息显示在手机上。在显示过程中本系统根据不同的消息类型来判断调用那种显示的方法，当为文本消息是调用显示文本的方法，当为图片消息是调用显示图片的方法，当为文件消息是调用显示文件的方法。发送聊天消息。发送文本消息流程如图3-3所示。

在输入框输入消息

点击发送按钮

将文本消息封装成一个消息对象进行发送

在界面上显示文本消息

判断消息的类型

调用显示文本消息的方法

文本消息

图3-3 发送文本消息流程图

#### 3.2.2.2 发送图片消息

（1）发送相册图片

当用户点击加号图片按钮会显示隐藏的发送图片，拍摄照片，发送文件的Layout，点击里面的图片按钮，会打开手机自带的相册，然后选择相册中的一张图片，便会把所选择图片的绝对路径返回，然后根据返回的路径，创建一个文件对象，然后创建消息对象，消息对象的构造方法需要传入聊天类型，接收者账号、文件对象、文件名字四个参数，创建消息对象后发送给接收方接收方便可接收到文件，发送方需要将发送的图片显示在手机屏幕上，如果直接将发送的图片显示在屏幕上太占地方，所以先将图片进行压缩，然后生成缩略图，最后将缩略图显示在屏幕上。发送相册图片流程如图3-4所示。

点击+号按钮，显示隐藏域

点击相册按钮

进入相册，选择图片

返回图片的绝对路径

创建文件对象

创建图片消息对象，并发送

判断消息类型

将图片进行压缩生成缩略图

显示缩略图

图片消息

图3-4发送相册图片流程图

（2）发送拍摄图片

其实发送拍摄图片，前一部分跟发送相册图片有所不同，但后面显示图片、发送图片消息是相同的。当用户点击加号图片按钮会显示隐藏的发送图片，拍摄照片，发送文件的Layout，点击拍摄按钮，会打开手机的摄像头，点击拍摄按钮，拍摄完毕后，点击勾号，就会将拍摄好的图片保存在手机中，并返回拍摄按钮的路径，然后将路径进行拼装，得到拍摄图片的绝对路径，然后就创建文件对象，创建图片消息对象，发送图片消息，在手机上显示图片消息。发送拍摄图片流程如图3-5所示。

点击+号按钮，显示隐藏域

点击拍摄按钮

打开手机摄像头

点击拍摄按钮

拍摄完成

将拍摄照片保存在手机中，并返回路径

将路径进行剪切拼装

创建文件对象、创建图片消息对象

显示缩略图

图3-5 发送拍摄图片流程图

#### 3.2.2.3 发送文件消息

先点击加号按钮显示隐藏域，点击文件按钮，打开文件选择器，在打开文件选择器时，可以在代码限制可以发送的文件类型，在打开的文件选择器中，选择要发送的文件，点击确定按钮，然后返回选择文件的绝对路径，再然后生成文件对象，最后创建文件消息对象并发送，在将发送的文件消息显示在手机屏幕上。发送文件消息流程如图3-6所示。

点击+号按钮，显示隐藏域

点击文件按钮

打开文件选择器

选择文件并确定

返回所选择文件的路径

创建文件对象，创建文件消息对象并发送

显示发送的文件消息

图3-6 发送文件消息流程图

#### 3.2.2.4 接收消息

接收消息是在发送消息的活动中的onCreate()方法中注册了监听事件，当有消息发送过来的时候便会触发监听事件。将接收到的消息根据发送的消息类型显示在手机屏幕上。在显示文件和图片消息的时候，需要先将发送过来的消息中携带的附件先下载下来，然后才能拿到消息中附件的路径，才能在界面上显示出来，不然消息是显示不出来的。开始的时候这个bug困扰了我很久，最后还是解决了。接收消息流程如图3-7所示。

在发送消息的活动中注册监听

当有消息触发监听时

得到接收到的消息

判断消息的类型

显示文字消息

先下载消息中的附件然后显示消息

文字消息

带附件的消息

图3-7 接收消息流程图

### 好友管理模块

#### 3.3.3.1 构建好友通讯录

当用户登入系统后，在通讯录Fragment中的FriendsFragment中进行初始化，通过访问网易云信提供的API服务，FriendService的getFriendAccounts()方法获取到当前登入用户的所有好友的账号，然后通过UserService的getUser

InfoList(List<String> accounts)方法来获取所有用户的资料，将的得到的用户的集合添加到FriendsListAdaper的中，在Adapter中将每个用户的数据取到并显示出来。构建通讯录流程如图3-8所示。

初始化FriendsFragment

得到用户的所有好友的账号

根据账号得到用户的个人资料的集合

将得到的用户资料的集合添加到FriendsListAdaper中

图3-8 构建通讯录流程图

#### 3.3.3.2 添加好友

用户在添加好友页面，通过输入账号精确查找或通过名字进行模糊查找，将找到的用户添加到ListView的Adapter中并将信息显示在界面上，当用户点击ListView的Item时就会触发点击事件，将点击的用户的账号传递给显示个人资料的页面，在个人资料页面用户可以看到该用户的一些基本资料，如昵称、年龄、性别等，用户可以点击界面中的添加好友按钮，便可以成功添加好友。添加好友调用的是FriendService的addFriend(AddFriendData data)方法来添加好友。添加好友流程如图3-9所示。

进入添加好友界面

输入要查找的账号或名字

点击查找到的人

进入到用户个人资料页面

点击个人资料页面中的添加好友

图3-9 添加好友流程图

#### 3.3.3.3 删除好友

用户在发送消息页面点，点击标题栏的人图标按钮就能进入到聊天设置也面，在聊天设置页面用户可以点击下方的删除按钮，对好友进行删除。在用户的个人资料页面点击标题栏的更多按钮，可以进入到更多的页面，在该页面用户也可以点击下方的删除好友按钮，对好友进行删除。删除好友调用的是Frien

dService的deleteFriend(String account)方法来删除好友。删除好友流程如图3-10所示。

发送消息页面

进入聊天设置页面

点击删除好友按钮

好友资料页面

点击更多按钮

进入更多页面

点击删除好友按钮

图3-10 删除好友流程图

#### 3.3.3.4 查看好友详细资料

查看用户资料，当我们点击一个好友时，就会触发FriendsListAdapter的监听事件，在监听事件中，我们设置了页面的跳转，在跳转到FriendDetailsActi

vity活动中时，我们传入了该好友的账号，通过该账号我们就可以获取到，该好友的所有资料信息，在该页面大致显示了好友的一些常用信息，如果要查看好友的所有信息，需要在跳转到详细资料页面，才可以看到好友的所有详细信息。查看好友详细资料流程如图3-11所示。

图3-11 查看好友详细资料流程图

进入好友个人资料页面

点击显示用户常用信息按钮

跳转到好友详细资料页面

### 用户资料设置模块

用户进入主界面后，可以通过左侧的侧滑栏，点击侧滑栏中的头部，可以

进入到用户的个人资料页面，在个人资料页面用户可以修改个人资料页面的背景、还可以修改头像，而修改背景和头像都可以从相册中选择图片，或是直接使用手机进行拍照，然后使用拍摄的图片。用户可以点击自己的个性签名，然后进入个性签名修改页面修改自己的签名。在用户个人资料页面还显示了用户的账号、年龄、性别等常用信息。当用户点击修改资料按钮时，用户会跳转到修改资料页面，在该页面用户可以修改自己的昵称、性别以及选择自己的出生日期，当用户点击选择出生日期时，会跳转到选择出生日期页面，在该页面有一个DatePicker供用户来选择自己的出生日期，选择出生日期后，还会动态的显示用户的年龄。用户资料设置流程如图3-12所示。

图3-12 用户资料设置流程图

主界面

打开侧滑栏

点击头部

进入到用户资料界面

点击编辑资料按钮

点击显示常用资料区域

编辑资料页面

### 群组管理模块

#### 添加群

用户进入到添加页面，在页面输入要添加的群的名字或ID，然后点击查找群，如果找到了群就会将群显示在界面上用户可以点击一个群进入到群资料页

面，点击下方的加入群按钮，即可加入到群组中。添加群流程如图3-13所示。

进入添加群界面

查找要加入的群

点击在界面上显示的查找到的群

进入到群资料页面

点击加入群按钮

结束

图3-13 添加群组流程图

#### 创建群

用户在主界面的通讯录模块点击创建群聊按钮，就会进入到选择要创建群聊的初始成员界面，用户可以选择自己的好友，作为要创建群中的初始成员，选择好成员后，点击立即创建就将群创建好了。创建群流程如图3-14所示。

点击创建群聊按钮

跳转到选择初始成员界面

选择初始成员

点击创建按钮

结束

图3-14 创建群流程图

#### 退出、解散群

用户在群聊发送消息页面，点击群聊设置按钮，进入到群聊设置页面，在群聊设置页面下方有个按钮，如果你是普通成员，下方按钮显示为退出群聊，如果你是群的创建者，下方则显示解散群聊。当普通成员点击退出群聊按钮后，该普通成员便退出了群聊；如果管理员点击解散群聊按钮后，群聊就解散了用户将再也看不到自己加入的该群了。退出、解散群流程如图3-15所示。

图3-15 退出、解散群流程图

进入群聊设置界面

是否为群的创建者

点击退出群聊按钮

点击解散群聊按钮

结束

# 系统功能测试

## 登入、注销测试

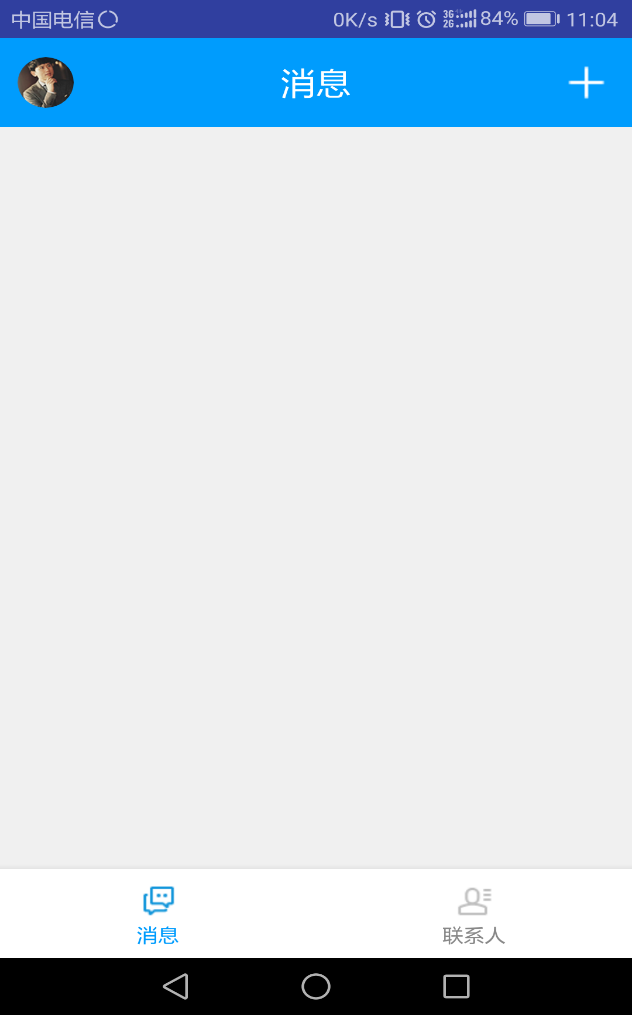
用户需要输入正确的账号和密码才能够登入到主界面，其中账号为用户的手机号，密码为用户设置的密码。若用户输入账号和密码后，账户或密码其中一项不正确，就会在手机屏幕上显示用户名或密码不正确，为什么不告诉用户哪项不正确，是为了防止某些用户恶意测试使用他人账号来登入。登入成功测试效果如图4-1所示。

图4-1 登入成功效果测试图

## 聊天模块测试

使用两台Android手机进行测试，两台手机分别登入不同的账号，该两个账号互相为好友，然后点击给好友发送消息，首先测试发送文本消息，看发送方与接收方是否可以收到消息并显示；然后测试发送相册中图片，看发送方是否发送成功并显示，接收方是否接收到并显示接收到的图片；再然后测试发送拍摄的图片，看发送方能否成功发送图片并显示图片，接收方是否成功接收并

显示接收的图片；最后测试文件传输，发送方发送文件，看发送方发送文件是否成功，接收方是否接收到文件。

## 好友管理模块测试

登入成功后进入通讯录页面，看是否能够构建好友通讯录。

在添加界面中查找用户，然后将找到的用户进行添加，看添加好友功能是否成功，然后到通讯录查看添加的好友。

点击好友的个人资料，在更多页面中对好友进行删除，然后查看通讯录中好友是否已经删除。

在好友的个人资料中点击显示常用信息的区域，然后进入到详细资料页面，看能否查看到好友的详细资料。

## 用户资料设置模块测试

## 群组管理测试

# 总结和展望

## 总结

在如今这个高速发展的时代，网络给人们的日常生活带来了极大的便利，而且在不久移动网络将更新到5G时代，如今移动端网速已经非常快了，到时候，网速将更加快，而即时通讯也会更方便人们认识和了解世界各地的陌生人，在与世界各地人的交流过程中了解其国家的文化，促进文化的传播和交流。

本文研究的基于Android的即时通迅系统，基本功能都已经实现。而在开发过程中也遇到了许多问题，经过自己的测试也将APP的bug进行修改和完善，尽量做到没有bug。在写发送图片消息的时候，发送端发送后，发送的图片会立即显示在界面上，而接收端接收到消息后，总是没有将接收到的图片显示。这个bug一直困扰了我很久，我有一段时间不断的修改那部分的代码，不断的测试，就是一直显示不出来，到后来我都不想在去修改了，我便先将其放置了一段时间，去开发其它的功能。当其它功能都开发差不多的时候，我回来看这个bug，突然就有了灵感，我先将其进行下载，在下载哪设置一个回调函数，如果下载成功，我在去拿到图片进行显示，这样困扰我很久的bug突然就解决了，我感到非常开心，甚至有点小激动，bug终于解决了。在开发过程中也还遇到过其它一些bug，就不一一列举了，至少最终都解决了。

在开发过程不管写了那个功能都需要进行测试，哪怕你很有自信的认为这绝对没有bug，也需要进行测试，如果不进行测试，如果在多人协作开发过程中，就可能给你的团队中的成员带来一些麻烦，你还要在去修改你写的那段代码，所有自测在软件开发过程中是非常重要的。然后在开发过程中，就算你遇到了bug，不要灰心，你要一步一步去调试，发现问题的根本原因，然后在根据原因进行逻辑的处理和修改，在遇到超级烦人的bug时，不妨先放一放，休息放松一会，说不定等过会来看，就会发现原来这么好解决。

## 展望

本文在使用AndroidStudio中开发发费了大量的时间和精力，而且因为时间和精力有限，也由于刚接触Android没多久，以前使使用JAVA开发网页的，所以还有许多地方需要进行完善和解决。所以对未来可以进行的工作进行展望。

（1）由于刚学android没多久，没有学到android中特别高深的知识，比如性能优化、内存优化等，对底层架构如何实现没有清晰的认识，所以等以后有时间，学了更多android知识后，有可能会进行二次开发和升级。

（2）功能还不太完善，本系统，现在只提供了即时通讯的一些基本功能，向QQ中的空间、微信中的朋友圈都没有，这样就没法发布用户一天的心情和发布自己一些平常有趣的事情分享给自己的朋友。所以进行二次开发的时候会加入动态这一个功能，还会加入如今常用的新技术，扫码加好友、面对面建群，还会对文件选择进行优化，因为现在的文件选择要找文件还比较麻烦需要自己一个文件夹一个文件夹去找，给用户的体验不是很好，也需要完善。而且如今只写了android客户端，没有写服务器端，很多数据就没法保存，所以为了扩展和保存会写一个服务器端。

（3）由于使用的是API，对信息的安全和加密也不太了解，不了解信息是否安全，在进行完善的时候可能会遗弃掉API，自己实现如何发送和接收消息，虽然这样会发费大量的时间和精力，但对底层的实现将会更加清楚，如果有团队一起开发那就更好了。

随着研究的深入，这些问题我会尽自己的能力，尽量去解决这些问题，我不敢说能全部解决，因为即时通讯是一个开发中的一个领域，或许需要发费一生的精力去研究和了解，或许可以使即时通讯更加的繁荣，未来发展的将更好。

# 参考文献

# 致谢