

一、实习目的及要求

[结合所选择的题目阐述]

Android 开发提高：

提供 6 个基础样例代码，发挥想象力和创造力对其中一个进行改进和提高，比如：增加程序的功能，改进程序的人机交互性，以及提高程序运行的性能等。

二、实习选题的背景及目标

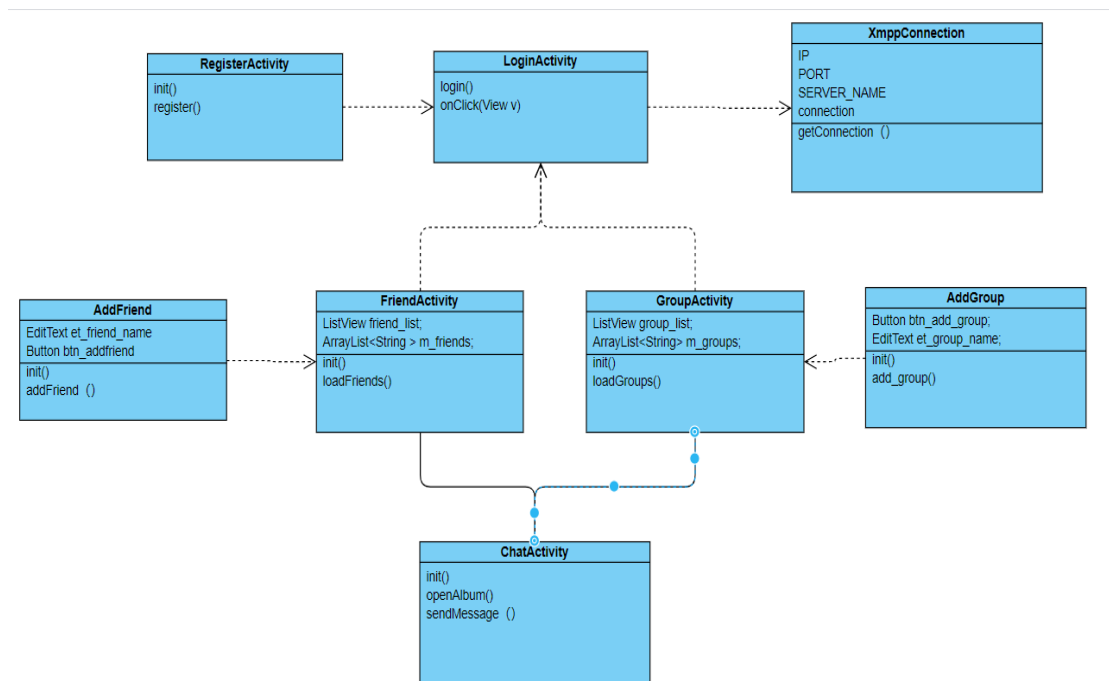
背景：安卓（Android）是一种基于 Linux 的自由及开放源代码的操作系统。主要使用于移动设备，如智能手机和平板电脑，由 Google 公司和开放手机联盟领导及开发。伴随着 Android 版本不断的升级，Android 在全球的市场份额不断提升，已经占用绝对的优势，本次实习，编写安卓小程序，练习安卓开发。

目标：

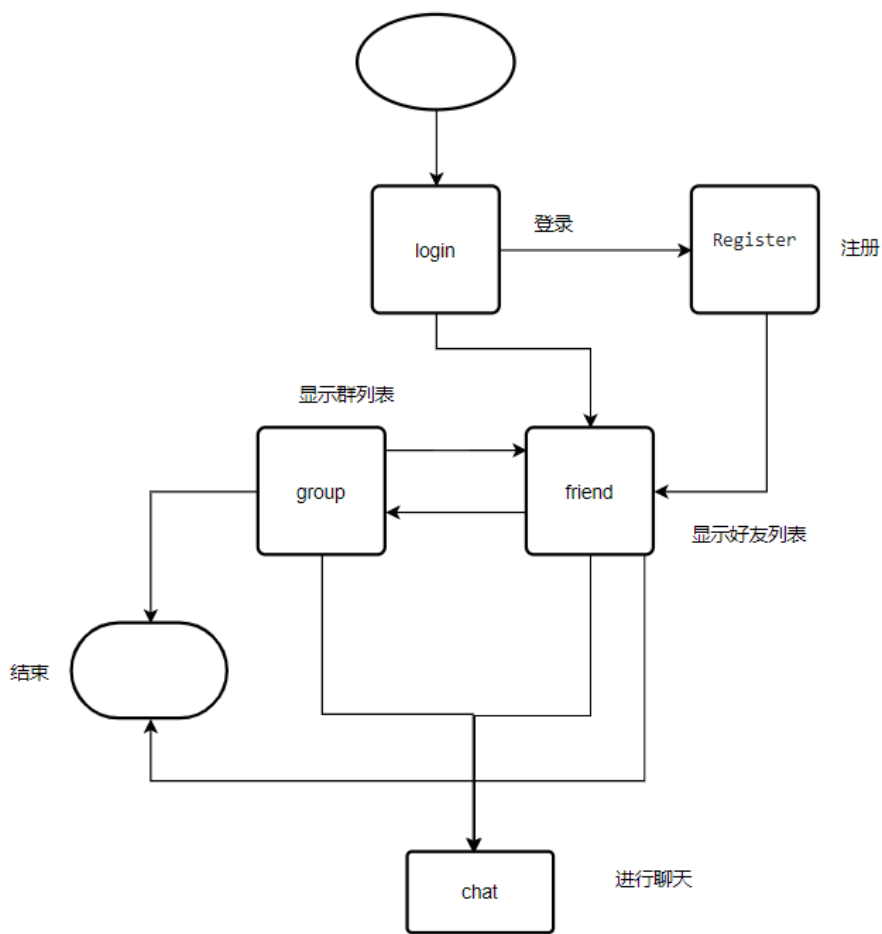
- （1）熟悉 Android 开发环境和基本操作；编写 hello world 程序。
- （2）进一步熟悉 Android 开发环境和基本操作；修改并增加程序的功能。

三、设计方案

类图：



流程图：



对各种类的介绍：

类名	功能
XmppConnection	连接的实现类，采用单例模式，连接的协议是XMPP 协议，在利用 asmack 包的 api
RegisterActivity	注册类，实现了注册功能，注册者需要输入邮箱、账号、用户名、密码，完成注册
LoginActivity	登录类，实现了登录功能，登录者需要输入用户名、密码，完成登录
FriendActivity	好友显示类，在里面显示好友列表，可以点击好友进行聊天
AddFriend	加好友的类，输入账号即可添加
GroupActivity	好友群类，在里面显示群的列表，可以点击进行聊天
AddGroup	加群的类，输入群的名字即可添加
ChatActivity	聊天的类，实现了文字和图片的发送。在好友列表和群的列表都可以进入。

四、实习环境与主要步骤描述

硬件平台：lenovoR720
操作系统：Windos10、Android、Centos7
使用的软件：Android Studio
开发语言：java

主要步骤：
我采取的是使用基于 XMPP 协议的方式进行通讯，服务器使用已经写好的 openfire 软件，并且可以部署到云服务器上，实现跨局域网通讯。客户端使用 asmack 的开源库进行开发，asmack 是一套很好的开源即时通讯 api, 是基于 Xmpp 协议的实现。

①首先写连接类 XmppConnection，采用的是单例模式，全局共享一个连接。下表是 XmppConnection 的一些属性和方法介绍。

属性：

属性名	介绍
XMPPConnection connection	和服务器的连接，实现功能主要考它
FileTransferManager fileManager	用于文件发送，目前只实习了图片发送
MultiUserChat multiUserChat	用户群聊

方法：

方法名	参数	返回值	功能介绍
XMPPConnection getConnection()		XMPPConnection 类型的连接	返回连接
boolean createRoom(String roomName, String password)	roomName: 房间名字 password: 密码	Boolean 类型，返回是否创建成功	这里的房间就是群组，功能是创建群组
FileTransferManager getFileTransferManager()		FileTransferManager 类型，用于文件发送	返回文件控制类，实现文件传输

②编写注册类 RegisterActivity，实现普通的注册功能。下表是对 RegisterActivity 的一些属性和方法介绍：

属性：

属性名	介绍
EditText etEmail	Email 输入框，用于输入 Email
EditText etAccount	账号输入框，用于输入账号
EditText etName	用户名输入框，用于输入用户名
EditText etPassword	密码输入框，用于输入密码
Handler handler	用于控制界面的信息

方法:

方法名	参数	返回值	功能介绍
void init()			初始化控件，以及一些监听器
void register()			用于实现注册功能

③编写登陆类 LoginActivity，实现登陆功能。下表是对 LoginActivity 的一些属性和方法介绍:

属性:

属性名	介绍
EditText etUser	账号输入框，用于输入账号
EditText etPassword	密码输入框，用户输入密码
Button btnLogin	登录按钮
Button btnRegister	注册按钮
Handler handler	用于控制界面的信息

方法:

方法名	参数	返回值	功能介绍
void init()			初始化各种控件，以及设置监听器
void login()			用于实现登录的功能

④编写朋友显示类 FriendActivity，实现好友数量的显示。下表是对 FriendActivity 的一些属性和方法介绍:

属性:

属性名	介绍
ListView friend_list	好友列表显示的控件
ArrayList<String > m_friends	好友的姓名
BottomBar mBottomBar	底部的菜单栏，用于切换到群聊
Handler handler	用于控制界面的信息

方法:

方法名	参数	返回值	功能介绍
void init()			初始化各种控件，以及设置监听器
void loadFriends()			加载好友列表
void addSubscriptionListener()			添加监听器查看是否有人加你

⑤编写 AddFriend 类，实现添加好友的功能。下表是对 AddFriend 的一些属性和方法介绍：

属性：

属性名	介绍
EditText et_friend_name	输入框，用于输入好友姓名
Button btn_addfriend	添加好友

方法：

方法名	参数	返回值	功能介绍
void init()			初始化各种控件，以及设置监听器
void addFriend()			用于加好友功能实现

⑥编写 GroupActivity 类，实现群的显示功能。下表是对 GroupActivity 的一些属性和方法介绍：

属性：

属性名	介绍
ListView group_list;	群显示列表
ArrayList<String> m_groups	群的名字的列表
BottomBar mBottomBar	底部的菜单栏，用于切换到好友聊天
Handler handler	用于控制界面的信息

方法：

方法名	参数	返回值	功能介绍
void init()			初始化各种控件，以及设置监听器
void loadGroups()			用于加载群组的名字

⑦编写 AddGroup 类，实现添加群的功能。下表是对 AddGroup 的一些属性和方法介绍：

属性：

属性名	介绍
Button btn_add_group;	
EditText et_group_name;	

方法:

方法名	参数	返回值	功能介绍
void init()			初始化各种控件，以及设置监听器
add_group() {			用于加群的实现

⑧编写 ChatActivity 类,实现聊天的功能。下表是对 AddGroup 的一些属性和方法介绍:

属性:

属性名	介绍
EditText et_input;	内容的输入框
RecyclerView msgRecyclerView	用于内容的显示
ChatManager chatMan	个人聊天使用的管理
MultiUserChat muc	群聊使用的管理

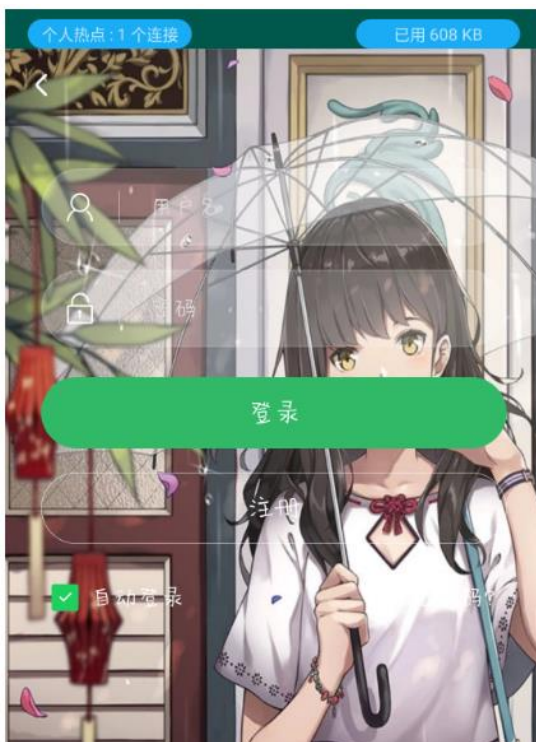
方法:

方法名	参数	返回值	功能介绍
void init()			初始化各种控件，以及设置监听器
void openAlbum()			打开相册
void addChatListener()			单人聊天消息的监听
void addGroupListener()			群聊天消息的监听
void addFileListerer()			文件的监听

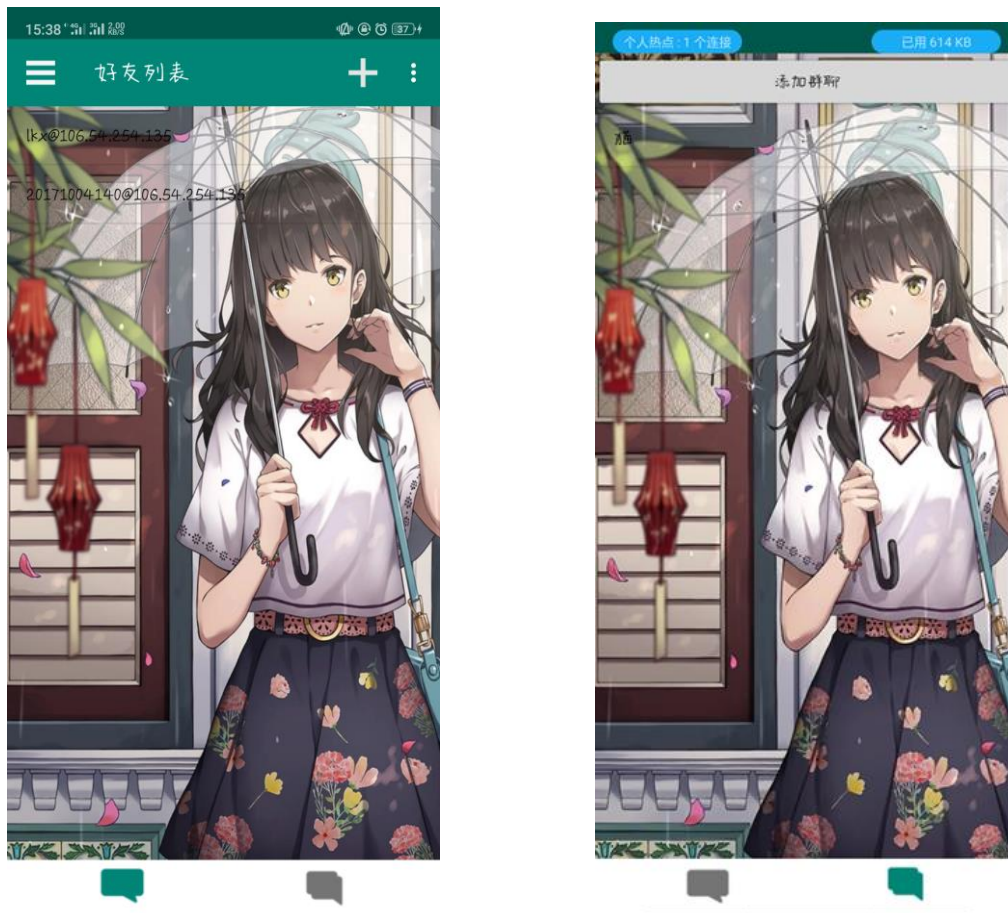
五、实习的结果与讨论

结果测试均是在公网下测试

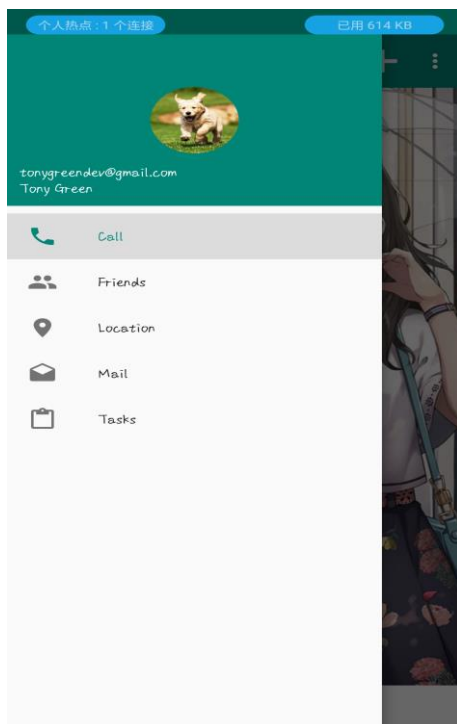
登录界面和注册界面:



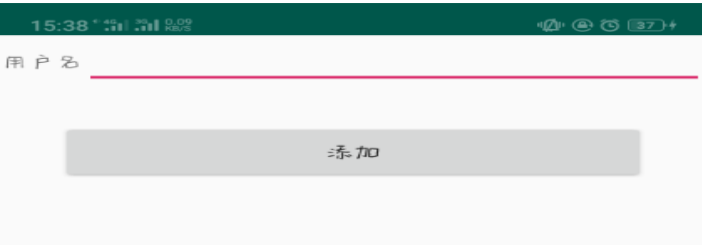
好友列表和群列表显示：（底部的菜单栏可以切换好友列表和群列表）



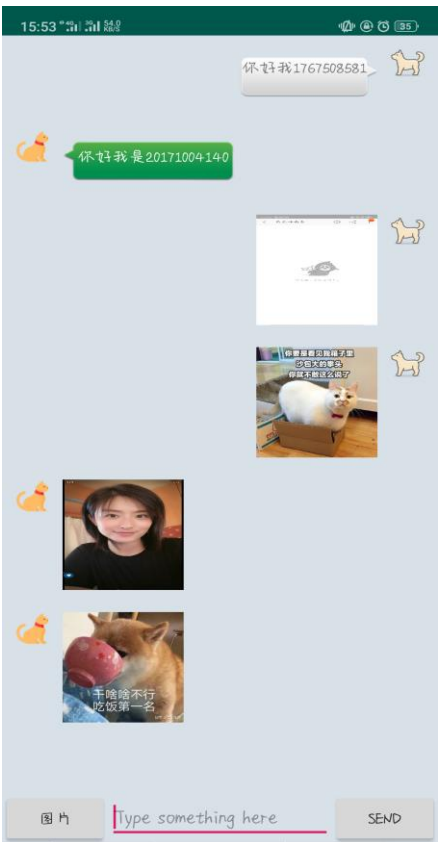
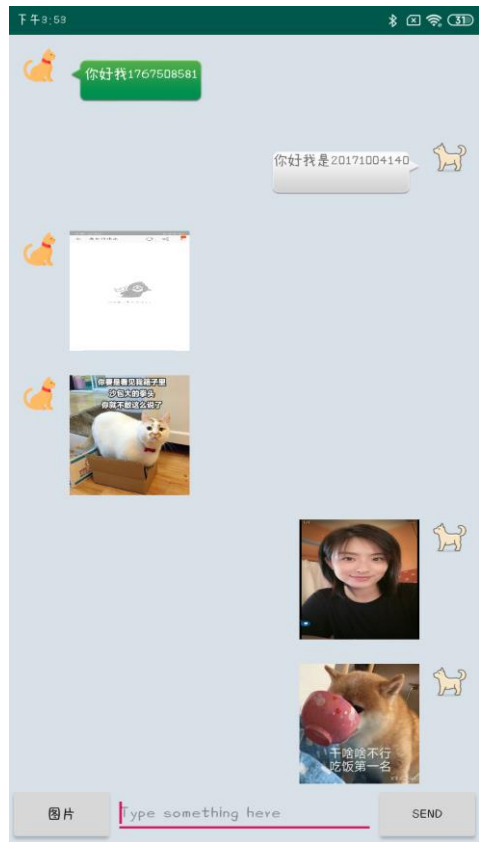
好友列表侧边的滑动菜单，以后可以设置为用户个人信息



添加好友和添加群的展示：



好友聊天测试：



群聊测试（由自己发信息都是猫头像，别人发都是狗头像，所以其实那两条狗是两个人发的信息）：

