|  |  |
| --- | --- |
| 档号 | 序号 |
|  |  |

**软件体系结构与设计课程项目**

**APP聊啥**

**详细设计说明书**

V1.0

软件体系与结构设计A组

2016年5月6日

版本修订历史

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 文件状态：  [ ]草稿  [√]正式发布 | | 当前版本： | | V1.0 | | | | |
| 作 者： | |  | | | | |
| 审 核 人： | |  | | | | |
| 发布日期： | |  | | | | |
| 日期 | 版本编号 | | 简要说明 | | 著者 | 审核者 | 审核日期 |
|  |  | |  | |  |  |  |
|  |  | |  | |  |  |  |
|  |  | |  | |  |  |  |
|  |  | |  | |  |  |  |
|  |  | |  | |  |  |  |
|  |  | |  | |  |  |  |
|  |  | |  | |  |  |  |
|  |  | |  | |  |  |  |
|  |  | |  | |  |  |  |
|  |  | |  | |  |  |  |
|  |  | |  | |  |  |  |
|  |  | |  | |  |  |  |
|  |  | |  | |  |  |  |

目录

[1 引言 1](#_Toc451436115)

[1.1 编写目的 1](#_Toc451436116)

[1.2 参考资料 1](#_Toc451436117)

[2 用户模块程序设计 1](#_Toc451436118)

[2.1 用户登录 1](#_Toc451436119)

[2.1.1 程序描述 1](#_Toc451436120)

[2.1.2 功能 1](#_Toc451436121)

[2.1.3 流程逻辑 1](#_Toc451436122)

[2.1.4 类、接口设计 2](#_Toc451436123)

[2.1.5 数据结构 2](#_Toc451436124)

[2.1.6 时序图 3](#_Toc451436125)

[2.2 用户注册 3](#_Toc451436126)

[2.2.1 程序描述 3](#_Toc451436127)

[2.2.2 功能 3](#_Toc451436128)

[2.2.3 流程逻辑 4](#_Toc451436129)

[2.2.4 类、接口设计 4](#_Toc451436130)

[2.2.5 时序图 5](#_Toc451436131)

[2.3 忘记密码 5](#_Toc451436132)

[2.3.1 程序描述 5](#_Toc451436133)

[2.3.2 功能 6](#_Toc451436134)

[2.3.3 流程逻辑 6](#_Toc451436135)

[2.3.4 类、接口设计 6](#_Toc451436136)

[2.3.5 时序图 7](#_Toc451436137)

[2.4 重置密码 7](#_Toc451436138)

[2.4.1 程序描述 7](#_Toc451436139)

[2.4.2 功能 7](#_Toc451436140)

[2.4.3 流程逻辑 8](#_Toc451436141)

[2.4.4 类、接口设计 8](#_Toc451436142)

[2.4.5 时序图 9](#_Toc451436143)

[3 通讯录模块程序设计 9](#_Toc451436144)

[3.1 程序描述 9](#_Toc451436145)

[3.2 功能 9](#_Toc451436146)

[3.3 性能 9](#_Toc451436147)

[3.4 输入项 9](#_Toc451436148)

[3.5 输出项 10](#_Toc451436149)

[3.6 算法设计 10](#_Toc451436150)

[3.7 流程逻辑 10](#_Toc451436151)

[3.8 类、接口设计 11](#_Toc451436152)

[3.8.1 数据库接口类 11](#_Toc451436153)

[3.8.2 百度推送服务类 12](#_Toc451436154)

[3.8.3 界面显示类 13](#_Toc451436155)

[3.9 数据结构 15](#_Toc451436156)

[3.10 物理存储 15](#_Toc451436157)

[3.11 序列图 15](#_Toc451436158)

[4 微信模块程序设计 16](#_Toc451436159)

[4.1 程序描述 16](#_Toc451436160)

[4.2 功能 16](#_Toc451436161)

[4.3 性能 16](#_Toc451436162)

[4.4 输入项 16](#_Toc451436163)

[4.5 输出项 16](#_Toc451436164)

[4.6 算法设计 16](#_Toc451436165)

[4.7 流程逻辑 16](#_Toc451436166)

[4.7.1 发送消息 16](#_Toc451436167)

[4.7.2 接收消息 17](#_Toc451436168)

[4.8 类、接口设计 18](#_Toc451436169)

[4.9 数据结构 18](#_Toc451436170)

[5 接口设计 18](#_Toc451436171)

[5.1 接口一 18](#_Toc451436172)

[5.2 接口二 19](#_Toc451436173)

[5.3 接口三 19](#_Toc451436174)

[5.4 …….. 19](#_Toc451436175)

# 引言

## 编写目的

阐述APP “聊啥”的功能和界面设计。

## 参考资料

# 用户模块程序设计

## 用户登录

### 程序描述

输入用户登录所需要的信息，提交登录，验证提交的登录信息的有效性，通过则保存进入主界面，没有通过验证则提示出错信息。

### 功能

输入：登录所需的用户手机号和密码

输出：登录是否成功的提示信息

### 流程逻辑



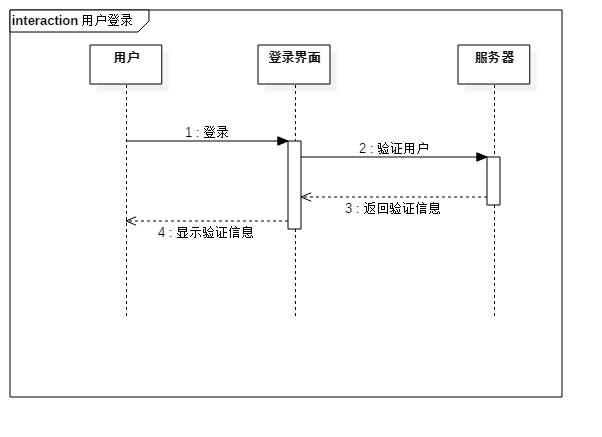
### 类、接口设计

|  |  |
| --- | --- |
| 功能名称 | 登录 |
| 接口名称 | user/login |
| 参数描述 | 1、mobile 手机号，字符串类型，例如：13012345678 |
| 2、pwd 密码，字符串类型 |
| 返回值 | 1、ret “0” 或者 “1” |
| 2、data 如果0，则返回userinfo；如果1，存放错误信息 |
| 备注 | 0表示成功，1表示失败；userinfo包含：userid/name/mobile/ |

### 数据结构

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 字段名 | 类型 | 长度 | 默认值 | 是否为空 | 主键 | 说明 |
| 1 | userid | varchar | 32 |  | No | Yes | 主键 |
| 2 | mobile | varchar | 11 |  | No | No | 手机号 |
| 3 | name | varchar | 50 |  | No | No | 昵称 |
| 4 | password | varchar | 50 |  | No | No | 密码 |
| 5 | register\_time | date |  | (getdate()) | No | No | 注册日期 |

### 时序图



## 用户注册

### 程序描述

输入用户登录所需要的信息，提交注册，验证提交的注册信息的有效性，通过则保存，没有通过验证则提示出错信息。

### 功能

输入：注册所需的用户手机号、密码和验证码

输出：注册是否成功的提示信息

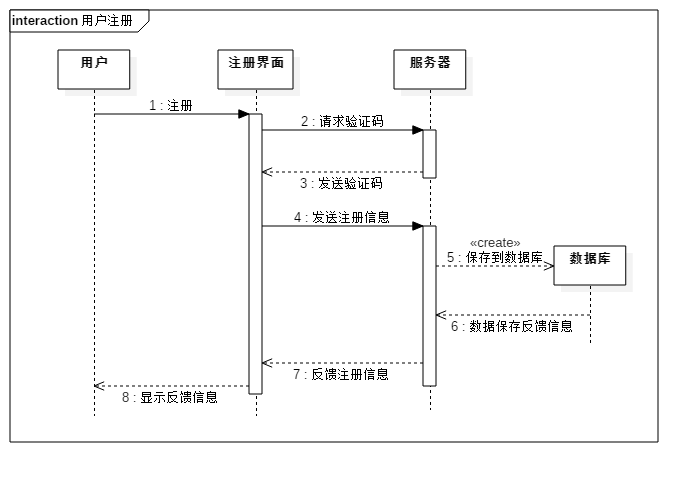
### 流程逻辑



### 类、接口设计

|  |  |
| --- | --- |
| 功能名称 | 注册 |
| 接口名称 | user/register |
| 参数描述 | 1、mobile 手机号，字符串类型，例如：13012345678 |
| 2、pwd 密码，字符串类型 |
| 3、name 昵称，字符串类型 |
| 返回值 | 1、ret “0” 或者 “1” |
| 2、data 如果1，存放错误信息 |
| 备注 | 0表示成功，1表示失败 |

### 时序图



## 忘记密码

### 程序描述

用户忘记密码时，提供用户重置密码的功能。

### 功能

输入：提交所需的用户手机号、验证码、新密码

输出：密码修改是否成功的提示信息

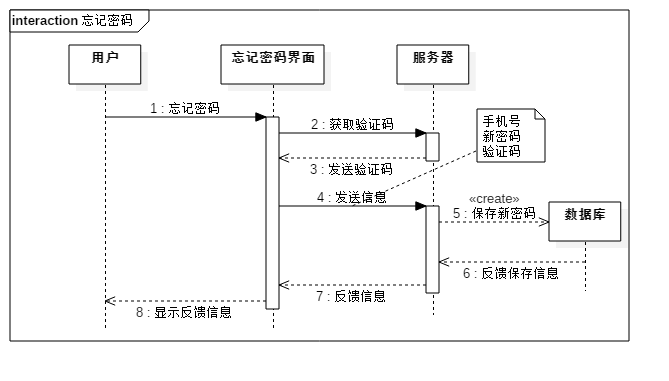
### 流程逻辑



### 类、接口设计

|  |  |
| --- | --- |
| 功能名称 | 忘记密码 |
| 接口名称 | user/ forgot\_password |
| 参数描述 | 1、mobile 手机号，字符串类型，例如：13012345678 |
| 2、pwd 密码，字符串类型 |
| 3、validation 验证码，字符串类型 |
| 返回值 | 1、ret “0” 或者 “1” |
| 2、data 如果1，存放错误信息 |
| 备注 | 0表示成功，1表示失败 |

### 时序图



## 重置密码

### 程序描述

用户重置密码的功能。

### 功能

输入：提交所需的旧密码、新密码

输出：密码修改是否成功的提示信息

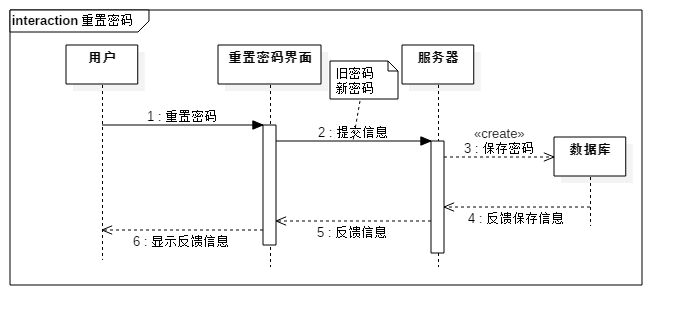
### 流程逻辑



### 类、接口设计

|  |  |
| --- | --- |
| 功能名称 | 重置密码 |
| 接口名称 | user/change\_pwd |
| 参数描述 | 1、old\_pwd 旧密码，字符串类型 |
| 2、new\_pwd 新密码，字符串类型 |
| 返回值 | 1、ret “0” 或者 “1” |
| 2、data 如果1，存放错误信息 |
| 备注 | 0表示成功，1表示失败 |

### 时序图



# 通讯录模块程序设计

## 程序描述

这部分主要以可伸缩的列表展示用户已经加为好友的联系人，并且用户可以管理联系人的分组。当用户点击其中某个联系人的时候，进入聊天界面并且可以开始聊天。

## 功能

显示用户所有联系人

将联系人分组

用户可以下拉刷新联系人列表

## 性能

更新性能：用户登录成功后2秒之内更新所有联系人列表

显示性能：用户点击联系人tab，0.5秒之内显示该分类所有联系人

## 输入项

用户id：用来唯一标识某个用户的序号，在用户注册的时候有服务器发号生成并且唯一。

## 输出项

用户所有的联系人列表。

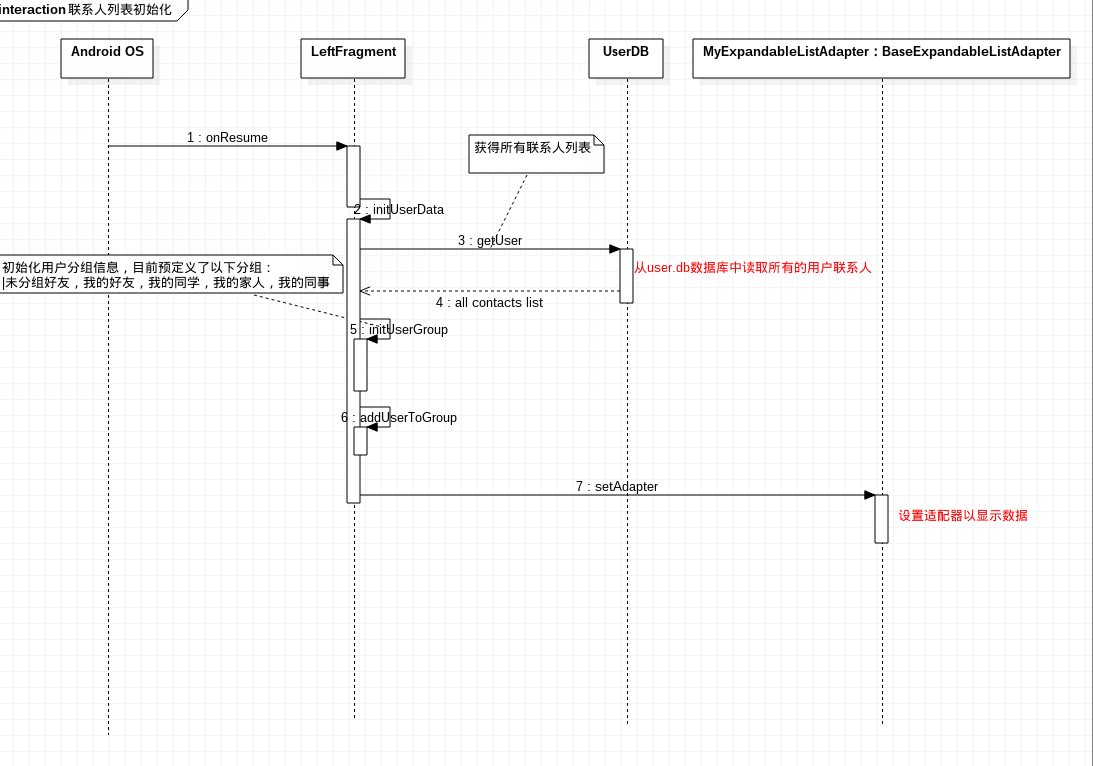
## 算法设计

本功能无特殊算法设计，均使用android标准的接口操作方式进行功能构建开发。

## 流程逻辑

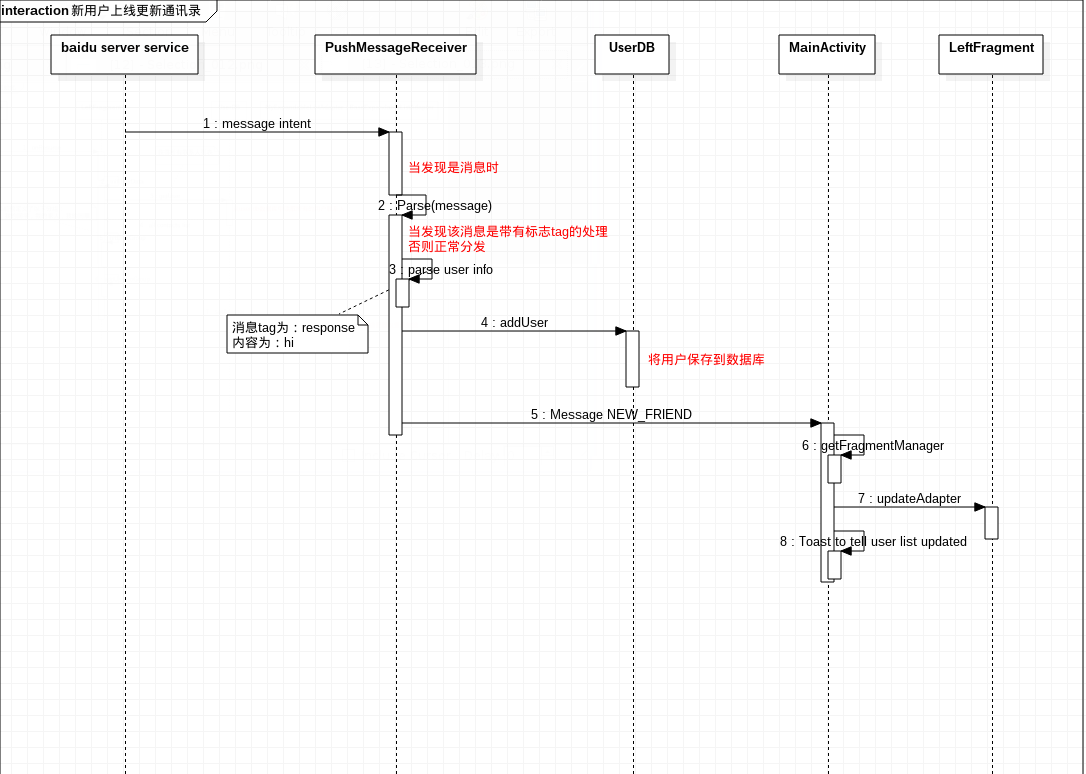
本部分的设计逻辑分为两块：用户登录成功后更新用户列表和别的新用户注册成功后更新联系人列表。

我们先看第一个模块也就是用户登录成功后更新用户列表部分的设计，以下是设计的时序图：



首先我们会被android操作系统回调我们的生命周期方法onResume，然后我们会初始化用户列表主要就是从本地数据库中获得所有联系人的列表，最后我们设置适配器更新界面显示。

接下来我们看一下当有新用户注册的时候联系人更新的逻辑设计：

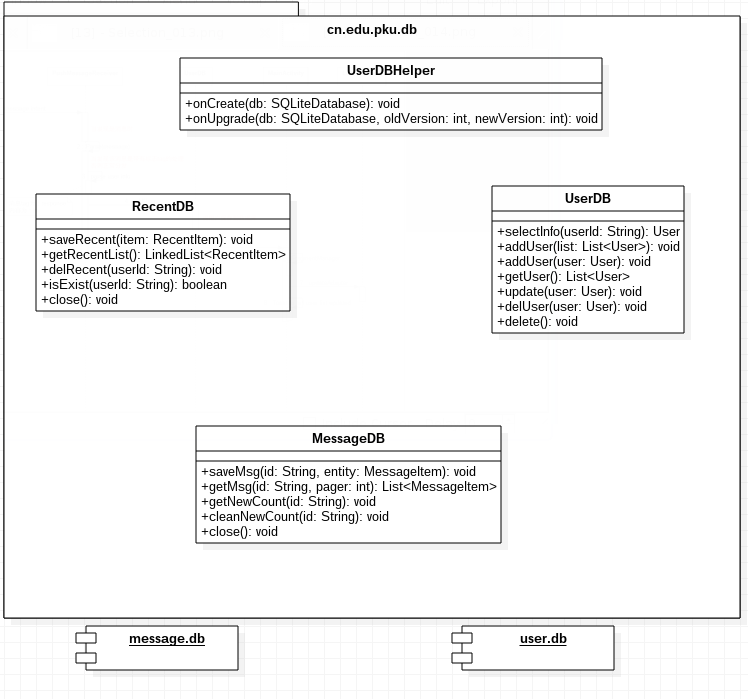


每个用户注册并且登录成功的时候都会向所有的在线和非在线用户发送特定的打招呼信息，然后我们的app在收到这个特定的信息的后会将这个用户存放到本地数据库中，然后更新UI。

## 类、接口设计

### 数据库接口类

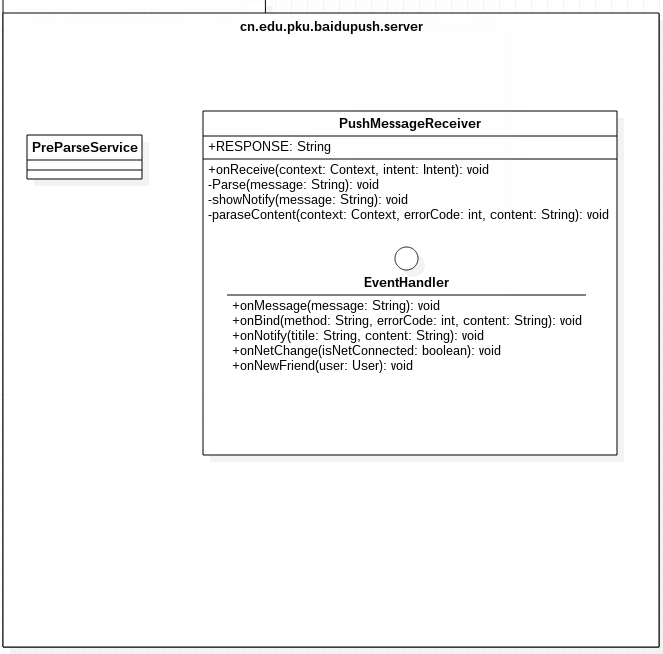
这部分设计的类比较多，首先是数据库这块的接口类设计：



我们一共设计了3个接口类和一个帮助类（帮助类的作用主要就是打开关闭数据库等），其中MessageDB和RecentDB类是和message.db数据库交互的，MessageDB类主要处理用户聊天数据，RecentDB主要处理最近聊天历史数据；UserDB是处理用户身份信息的接口类，和user.db数据库交互。

### 百度推送服务类

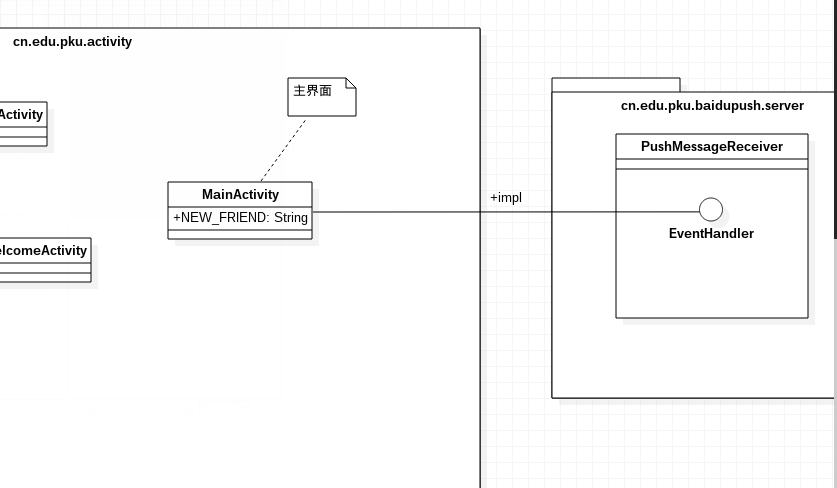
这个类是使用百度云推送必须实现的类，设计如下：



其中我们实现了一个内部接口，这个接口是用来和界面类交互的接口，主要是处理不同的消息。

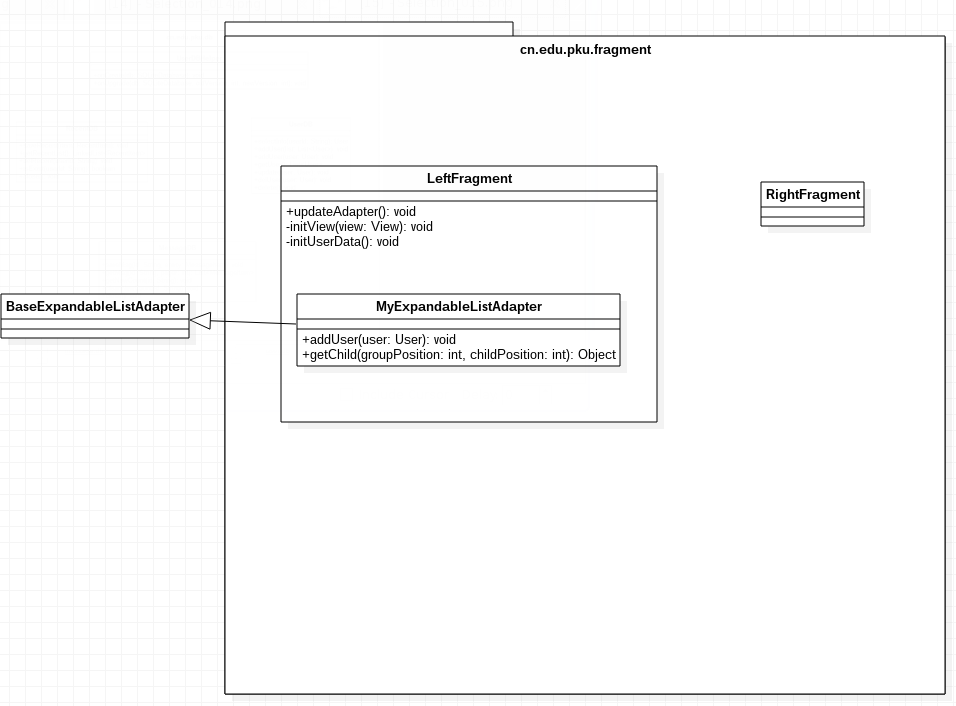
### 界面显示类

首先是activity类设计：



Activity类是界面主要类，是真实显示类的容器类，这里我们在主界面类中实现EventHandler这个接口，以接受来自网络的数据。

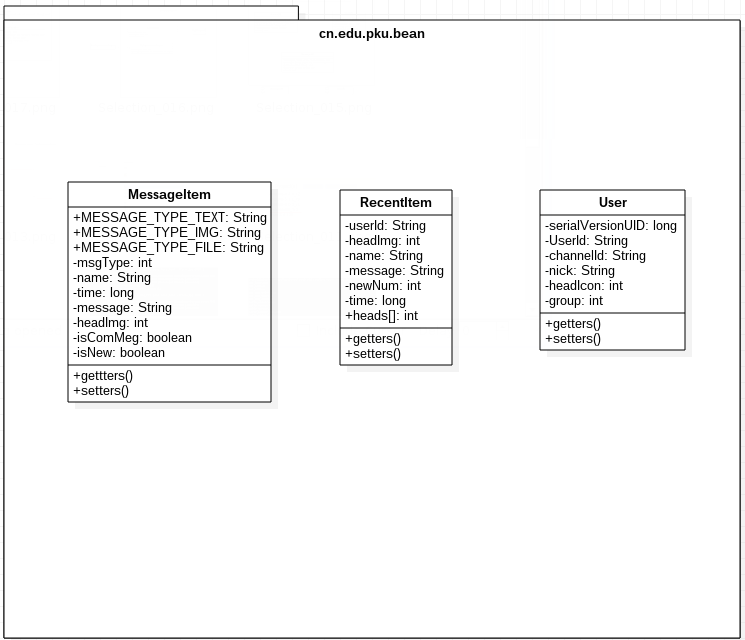
我们使用了android的fragment的界面设计方式，这样设计出的UI更加灵活，fragment的设计如下：



这些fragment类主要就是显示实际数据的类，使用android的标准接口。

## 数据结构

本模块的数据结构基本就是java beans的设计，如下图：



一共设计三个类，第一个是MessageItem表示一个网络消息，RecentItem表示一个最近聊天纪录，User表示用户类。

## 物理存储

## 序列图

# 微信模块程序设计

## 程序描述

这部分是整个项目的核心功能之一，这个模块主要实现用户之间的聊天功能，目前设计的聊天功能有：文字聊天，表情聊天，目前只有单聊一种聊天形式。

## 功能

发送文字聊天

发送表情聊天

## 性能

用户点击发送后，在网络条件正常情况下（100M网络，20M带宽），3~5s内传送给目标用户，相应地对方发送消息本地也要在这个时间内收到消息。

## 输入项

用户的聊天信息

## 输出项

对端用户的消息

## 算法设计

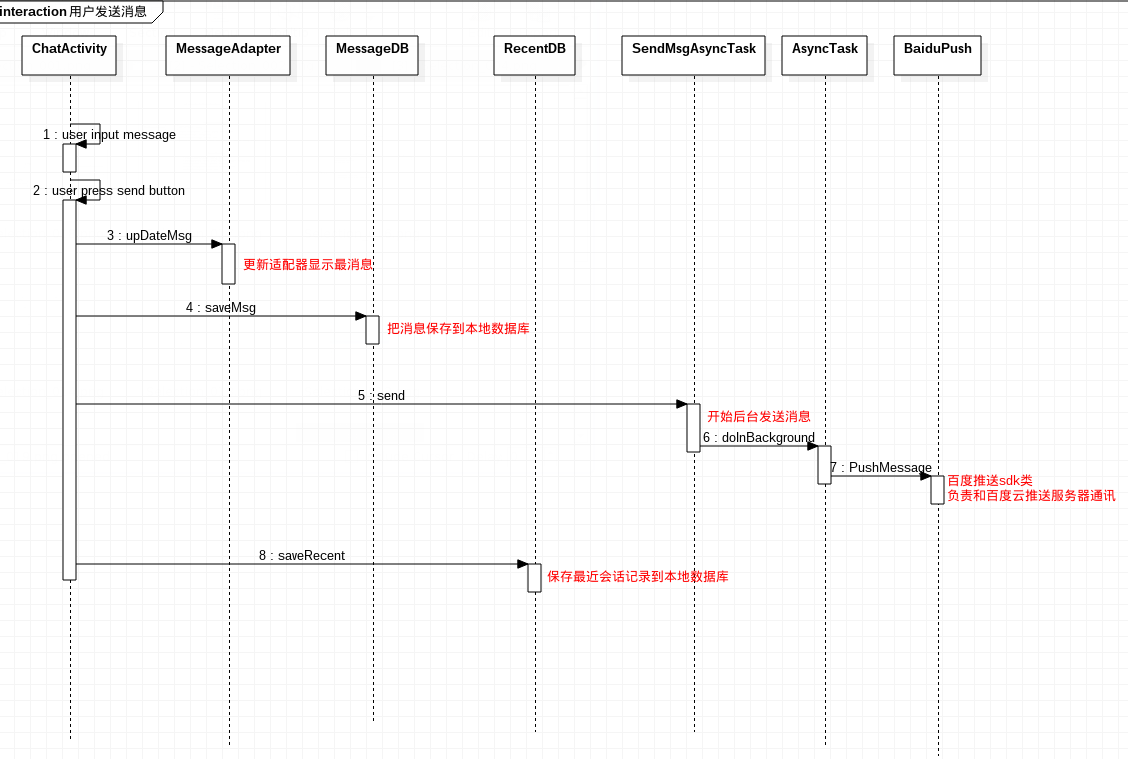
本功能无特殊算法设计，均使用android标准的接口操作方式进行功能构建开发。

## 流程逻辑

本模块一共有两个子模块：发送消息和接收消息。

### 发送消息

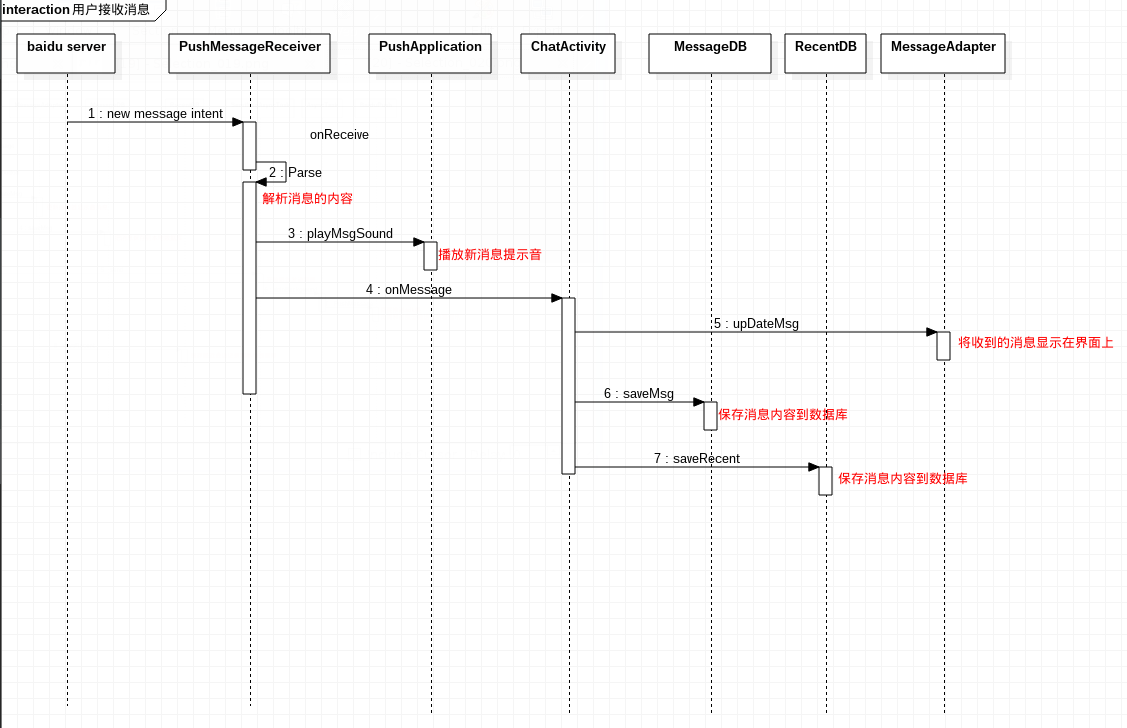
我们首先看下发送消息的部分：



用户发送消息是从用户点击发送按钮开始的。当用户按下发送按钮的时候，如果消息框的内容不为空的话，那么就开始发送过程了。首先，我们会将这个消息对象放到适配器中，在界面上显示这条消息；然后，我们再将消息保存到数据库和最近历史消息数据库；最后我们通过android提供的后台操作接口在后台通过baidu的sdk将数据推送到百度云服务器上。

### 接收消息

接受消息是随时都会发生的，只要对方成功发送一条消息之后本地app会触发以下逻辑流程：



当对方用户成功地发送一条消息的时候，百度云服务器会及时将这条消息发送给我们本地sdk中百度service，然后百度service会发送一个intent，这个intent中包含了受到的消息。然后android操作系统会回调我们注册的接口，在我们定义的回调接口中首先会解析这个消息的内容，如果是一条正常的消息（即不带任何tag的消息，具体参见通讯录部分的设计），那我们首先会播放提示音，然后在回调界面类实现的onMessage接口，将这个消息给界面显示。

## 类、接口设计

这部分的类，接口设计和通讯录部分的是一样的，因为通讯录也是用过特殊的消息同步更新的。

## 数据结构

这部分的数据结构和通讯录模块是一样的，使用了相同的数据结构。

# 接口设计

## 接口一

//接口说明

//接口实现画图说明

//接口实现文字描述

## 接口二

## 接口三

## ……..