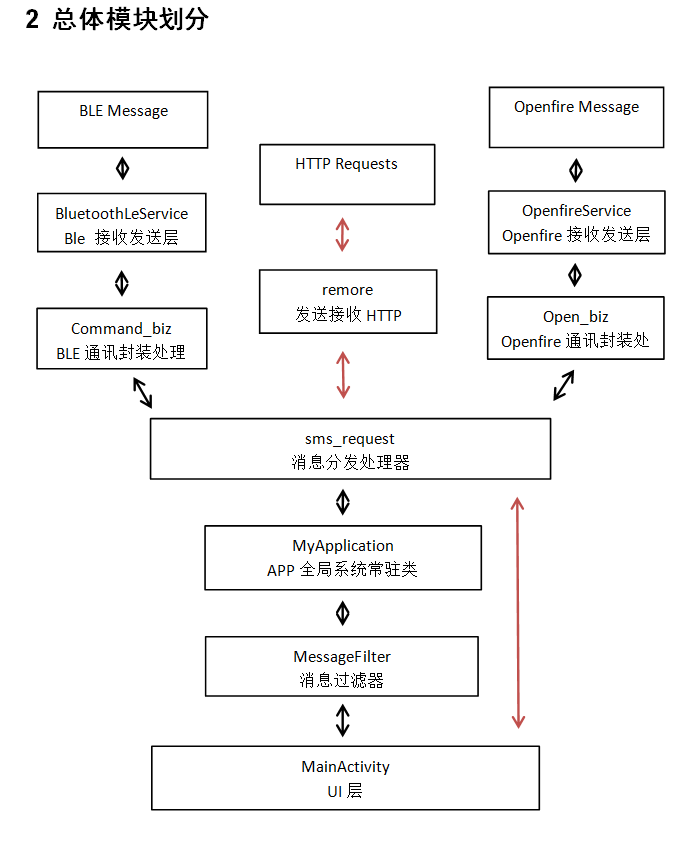
应用数据流程：

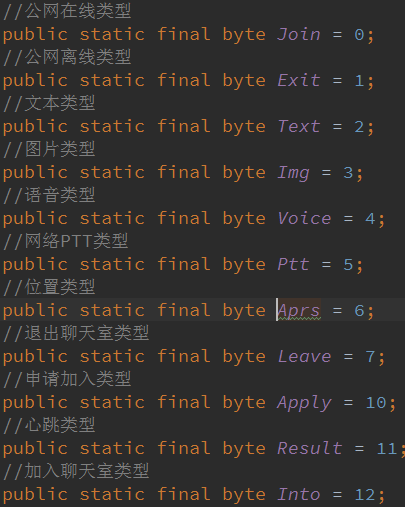


User模块：

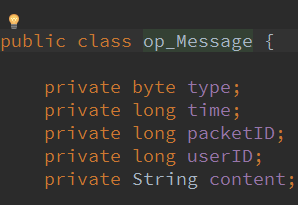
1. 实现用户登录，注册，找回密码，修改头像。
2. 说明书，常见问题，广告图及设备固件的下载存储，每次加载本地，发现新版本就下载更新。
3. 实现APP检查更新功能。

OfenFire 模块：

1. 利用sms服务器获取的用户IM账号，密码，实现OpenFire登录，注册，新建及使用二维码加入，分享聊天室。
2. 聊天室通讯数据类型：



1. 聊天室通讯消息体body 封装成JSON 数据类型传输



1.Time 类型必须要先获取服务器的时间，无法获取再用本地时间。

2.语音类型将数据包携带至 content属性，以序号开头，后面拼接Base64加密语音数据包，（必须加密，否则会出现数据丢失情况）每90个byte组成一个语音包。序号1个byte（0开始递增，最大100，循环。结束包序号为120）。

例如：content = 序号 + 语音数据包。

1. 新建聊天室时房间名字统一格式为 系统时间戳，例如：1550222779593

聊天室描述统一格式为 群主ID 例如：76

1. 二维码字符串拼接格式为
2. TxCode+ “，”+RxCode + “，”+freq++ “，”+fromId+ “，”+fromName+ “，”+ImCode.getId +“，”+ ImCode.getUUID 直接转String 类型拼接
3. ImCode.getName 长度为32的字符串

{ tx (1byte), rx (1byte) , freq (4byte) , id (8byte) , name (String)}

1. ImCode.getId 随机数（0-9999）
2. ImCode.getUUID 长度为16的随机字符串
3. 网络PTT抢占功能，同一聊天室按下及松开PTT都需发送命令至聊天室。同一聊天室只能一人按下PTT，其他成员等待。如果按下PTT 的500毫秒之内接收到了忙状态，以Message 对象的Time属性判断谁先按下。如果是自己慢，则放弃该次录音，进入忙状态。
4. Message 内容 = 1 = 忙。
5. Message 内容 = 2 = 解除忙。
6. 加入聊天室提示延迟20秒，
7. 若20秒之内退出，此次加入聊天室判定为失败，不做提示。
8. 若20秒之内接收到用户消息，立即判定加入成功，做出提示。
9. 公网离线提示不做延迟，只有判定加入成功的成员才做离线提示。
10. 判定为公网在线的成员，保存至数组。用于RF开关于自动状态下的判断。
11. 将加入成功的成员写入本地保存，退出群聊的成员本地删除，维持本地群成员集合。用于RF开关于自动状态下的判断。
12. 网络状态切换需要后台重新恢复openfire的登录及聊天室的在线状态。  
     1.例如：用户从无网络恢复到有网络，后台进行恢复在线状态。
13. 每次加入聊天室之前先判断服务器中是否存在，如果不存在则新建，存在则加入。网络重新恢复后台也要加入聊天室，用户登录逻辑同理。

10. 私有群功能方案，私有群和公开群以openfire中的聊天室时间戳ID做为区分符，用户A加入聊天室后发一个申请加入的消息马上退出。群主收到将处理结果私发给用户A，用户A收到后作出处理。其中涉及到离线消息处理，所以需要在群描述中获取群主ID。

1. Asmack 开发文档：  
     
   <http://download.igniterealtime.org/smack/docs/4.0.1/javadoc/org/jivesoftware/smackx/muc/MultiUserChat.html>

BLE 模块：

1. RF开关功能相应说明：
2. 自动模式：频率，TxCode RxCode 跟随群信息，不允许用户更改

根据聊天室群成员全部在线，RF开关设置关模式，反之有人为离线状态 RF开关设为开模式

1. 开模式：不跟随群信息，消息走双通道
2. 关模式：不跟随群信息，消息只走网络
3. 基于BLE通讯协议正常读写对讲机设备信息。
4. 对讲机通讯区分可靠传输与不可靠传输，可靠传输需成功进行握手协议，否则重发。

例如：写入设备信息的参数

不可靠传输 例如：语音数据包，允许一定程度的丢包。

4. 所有BLE通信都基于单线程轮循发送，不可错乱发送。

5. 新建聊天室设置聊天室Name数据拼接如下：

Name = {(TxCode (1个字节) + RxCode(1个字节) + Freq(4个字节) + id(时间戳Long类型转byte 8个字节)) + name(16个字节以内的输入昵称)};  
  
6. 与对讲机之间的语音数据包则不需要Base64加密。

消息过滤 模块：

1. 转发网络语音数据时判断对讲机状态是否为忙，及是否收到中断命令。
2. 接收到对讲机UserID为0的语音数据，只做保存，不转发。接收到ID 为自己的语音数据转发至openfire聊天室。长度同录音发送一样，90个byte 为一个数据包。
3. 接收到openfire聊天室的语音数据包，判断对讲机状态，决定是否转发至对讲机。  
    1.同时接收两段语音及以上情况，只转发先到的那段语音。以packetID判定每段语音数据。  
    2.转发语音数据的过程中，收到中断命令，对讲机状态变忙，及RF开关状态变为关。都应停止转发语音数据包，以packetID为单位。
4. 将接收到的Message 放进过滤集合，判断userID packedID 是否存在。不存在就写入数据库，推送至UI层，反之舍弃。Message 对象维持30秒的生命周期，超时剔除过滤集合。