**中山大学移动信息工程学院本科生项目报告**

**（2017年秋季学期）**

课程名称：移动应用开发 任课教师： 郑贵锋

|  |  |  |  |
| --- | --- | --- | --- |
| 年级 | 2015 | 专业（方向） | 移动互联网 |
| 学号 | 15352019 | 姓名 | 曾何萌 |
| 电话 | 15525866978 | Email | 2313086554@qq.com |
| 开始日期 | 2017.12.28 | 完成日期 | 201.1.20 |

# 项目名称

树洞——基于Bmob后端云的多功能日记本

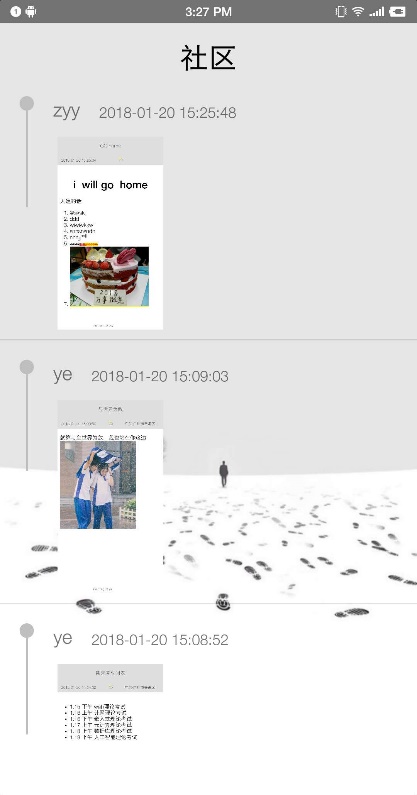
# 小组个人分工

* + - 1. 富文本编辑
      2. 新添日记
      3. 日记修改
      4. 图片上传与加载

# 实现内容与结果展示

1. **展示截图**

1)富文本编辑与新添日记 2）保存成功再次查看日记 3)下载图片显示

1. **实现步骤以及关键代码**
   * + 1. 富文本编辑——利用RichEditor实现.

首先添加依赖，并进行布局：





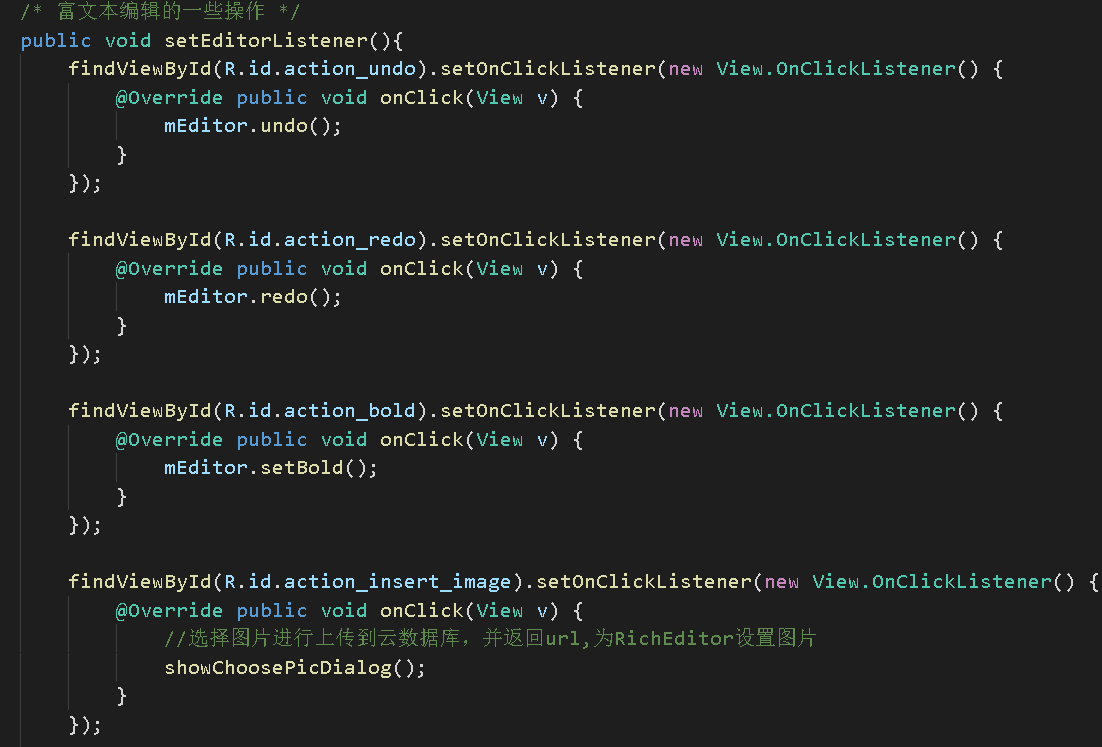
查看对应头文件，可以看到RichEditor的基于WebView实现的：



RichEditor之所以具有富文本编辑的功能，是因为它利用WebView可以显示html网页的功能，并为其设置了一系列事件处理函数（用javaScript实现），编辑的时候就是在修改html源码。常用的事件处理示例如下：



在java文件中，可以为不同的功能按钮（位于页面最上方的一排编辑按钮）实现对应的富文本编辑功能示例如下（更多功能需要产看项目源码）：



* + - 1. 新添日记

首先请求地理位置，天气等信息：



编辑成功后，保存提交。关于日记内容，提交的是html源码，方便之后查看日记的时候能保留原来的日记格式：

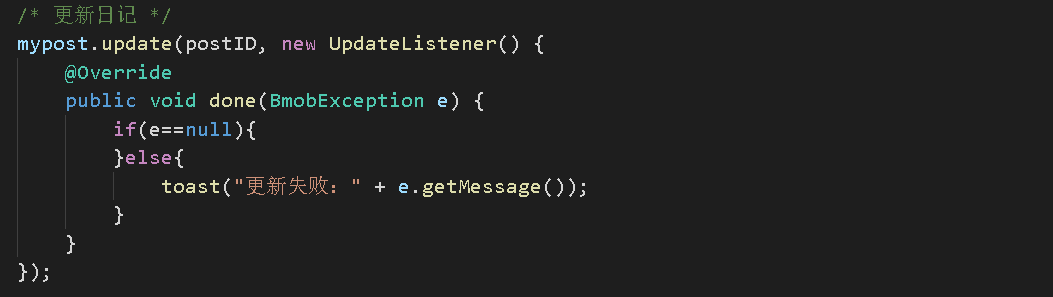


* + - 1. 修改日记

同新添日记类似，不过要首先从根据postID数据库查询该日记并显示出来：



提交更新：



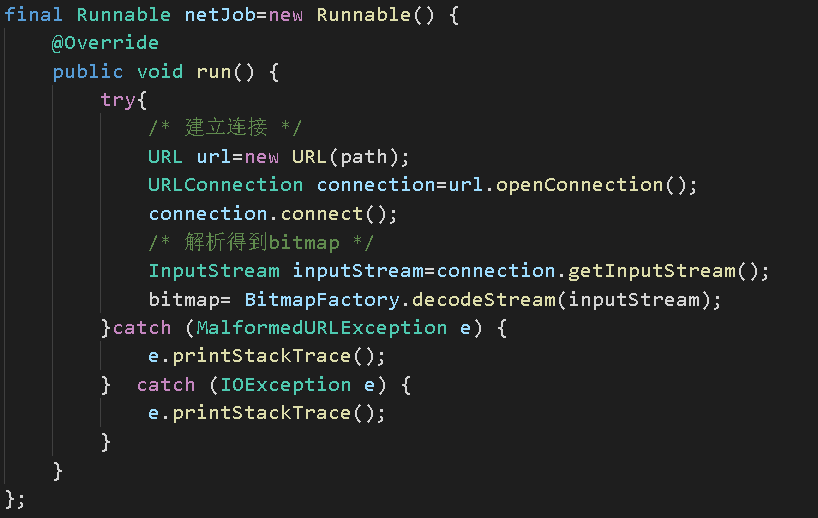
* + - 1. 图片上传与加载

RichEditor的图片上传与显示：首先从本地图库或拍照得到图片后并进行裁剪后。

将图片上传到云数据库的素材文件夹中，并返回该图片的url，之后利用该url为RichEditor设置图片：



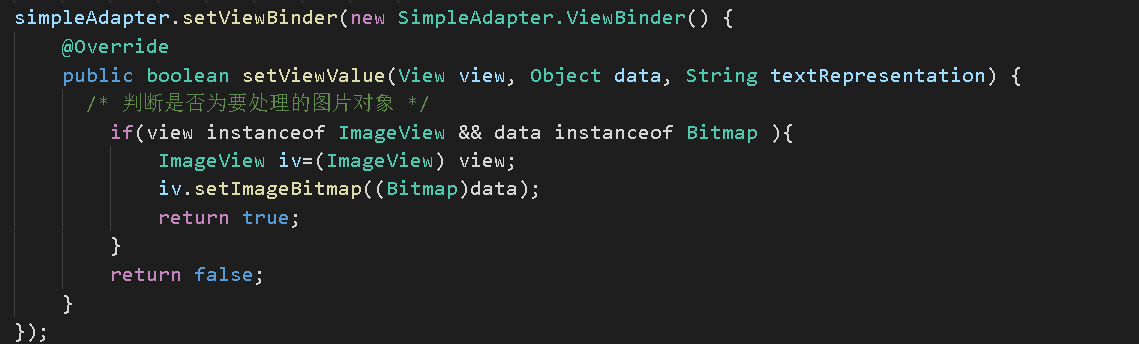
社区中，日记图片的加载：首先实现runnable接口



利用listView显示图片，首先要从数据库中查询分享列表，接着利用url请求图片：



为bitmap对象单独设置显示,需要重载适配器的setViewBinder函数：



1. **遇到困难以及解决思路**

1.社区中加载bitmap图像导致内存溢出,需要定期释放内存，使用bitmap.recycle()和system.gc()。

2.关于日记RichEditor中图片的处理，因为html源码中保存的是图片的地址，为了可以在不同手机上查看日记，需要将使用网络地址，首先将图片在本地处理完后进行上传，并返回链接，写入html源码中。

## 感想与总结

这次的期末项目使我认识到了app性能的重要性：之前的实验课项目都是关注UI设计和基本功能的实现，但一个完整的app需要同时兼顾UI、功能和性能。“树洞“涉及到多处网络上传与加载图片，出现了内存溢出和加载超时等各种问题，所以app运行起来有点卡顿，用户体验也不是很理想，在debug的时候，通过多线程网络请求和内存的及时回收在一定程度上解决了这些问题，但还是存在一些不必要的多次网络请求，这些问题同样可以通过Activity的通信机制和本地缓存来解决。

最近上网查了以下关于app性能优化的几个方面：比如数据结构的合理性、多线程、缓存数据、图片压缩、延迟加载、内存管理等，觉得自己对android的认识还欠缺很多，对项目开发的很多方面考虑不够，需要继续学习android开发，另外要重点关注原理，这样在做开发的出来的app才能更达标。

另外github上有很多开源的好用也很炫酷的控件，在开发时使用它们可以大大节省时间和精力，比如这次的RichEditor,在仔细阅读的说明文档以及查看源码后，理解了它的原理，使用起来很方便。

期中和期末的两次项目，都是小组成员努力的结果，大家分工明确，及时沟通，遇到问题互相帮忙，感谢队友，也感谢老师和助教们一个学期的帮助。