# Weilog开发文档

App Key：172318056

App Secret：be6c63432fd65482ad9576fa4980e324

## 概述

Weilog是作者开发的一款第三方微博App，是一次试手之作，如果发现bug，请及时反馈。初次设想中，APP的界面类似微博国际版。主要区域分为用户关注内容、热门内容、通知、关于个人用户设置四部分。个人设置放在侧滑栏，用户关注内容、热门内容、通知放在主页面的三个TAB中。

## 二、个人设置侧滑菜单

1. 页面实现总述：由于在整个app中，Tab的所有页面均能打开侧滑按钮，则此侧滑菜单重要性不言而喻。本app用谷歌的NavigationLayout布局。此布局包含两个子布局，一个是侧滑布局，设置其layout\_gravity为start，另一个则是页面布局。页面布局分其他三个tab子布局的父布局，分为三部分：顶上栏，底部栏和中间的显示区域。中间显示区域根据每个Tab的不同，分别编写，此处不再赘述。

2、侧滑页面的实现：侧滑界面用NavigationView控件实现。其中包括两个重要设置，headerLayout，即为个人信息头像显示区域；menu，即个人页面中各项按钮，分为group和item，group中以item来写每个按钮。侧滑按钮的点击进入：用到actionBarDrawerToggle，用法：**val** toggle=ActionBarDrawerToggle(  
 **this**, drawer, toolbar, R.string.*navigation\_drawer\_open*, R.string.*navigation\_drawer\_close*)  
drawer.setDrawerListener(toggle)  
toggle.syncState()

3、侧滑栏的点击监听方法：用获取Item id的方式

4、点击事件：

1）我的：跳转新的Activity，分为：我的个人信息，我的微博。我的个人信息作为listview的一个头item。

2）草稿：新的Activity，分为状态栏和草稿，草稿部分用listview显示，点击进入之后，跳转到编辑页面。

3）扫一扫：调用相机进行二维码扫面（相机页面怎么编写）

4）浏览记录：可以调用网页，比如直接把网页版的浏览记录的URL进行解析显示

4）设置：A）账号管理：添加新账号

B）WiFi环境自动播放视频及GIF图片（检测是否是WiFi，需要用service来实现

C）声音开关

D）通知开关

E）显示设置：根据一个样板来调整字体大小

F）语言：弹出对话框，用alert实现

G）关于：版本信息，开发者信息

H）注销账户

1. 夜间模式：怎么设置全局为暗黑色的夜间模式
2. 清楚缓存：如何得到缓存数据
3. 点击头像：直接进入我的页面

## 三、用户关注内容页面

1. 标题栏：分为三部分，左边打开侧滑栏按钮，然后是一个点击下拉菜单，用Spinner控件，此处仅有两个选项：全部和我点赞过。两个选项分别对应打开不同的数据源，将对应的内容显示在内容区。最右边是搜索，点击进入搜索页面，搜索栏下方显示历史搜索。搜索输入文字，自动匹配相关内容，并在下方显示
2. 内容区域：主要内容实现使用listview，由于微博内容不固定，所以需要自定义的WeilogAdapter来进行控件的各种加载。整个listview上滑，隐藏标题栏和底部栏。下滑，标题栏和底部栏出现，下滑到第一个item时，自动刷新。状态栏隐去之后，怎样全屏显示？如何检测上下滑动？
3. 右下角悬浮按钮：点击弹出三个按钮，分别是拍照，相册，写微博。

拍照调用相机，拍照完成，进入写微博页面，并自动加载刚才拍摄照片。

相册进入手机相册页面，读取所有图片，右下角完成，进入写微博页面，自动加载刚才选择图片。

写微博页面，

标题栏：左边推出页面，退出时检查文本框是否有文字，如果有，询问是否保存。如果保存，将文字存入草稿箱。中间显示账户头像，右边按键发送微博

文本编辑：右下角选择公开，朋友可见，仅自己可见

底部栏：分别是图片入口，跳转至相册页面，话题入口，跳转话题搜索页面，@页面，进入用户搜索页面，表情入口以及最右的字数统计。发送微博，将微博内容转化为json数据，传送至服务器端

1. 底部栏，公用，单独叙述

## 四、探索页面（重复内容不再详细叙述）

1. 状态栏（参见四）
2. 内容界面加入布局viewPager，分为热门和视频两部分

热门：调用WeilogAdapter的结构，显示热门的数据

视频：使用普通的Adapter，用videoview来显示视频

1. 底部栏同上
2. 上滑viewPager栏上滑至顶部，停滞（实现？）

## 五、消息提醒

1. 状态栏：将右上角搜索修改为私信，点击进入搜索用户页面，选择用户之后，进入与用户的对话页面
2. Viewpager页面，分为@我，评论，点赞，私信四个页面

@我：显示@过我的微博

评论：选择收到的评论以及发出的评论（内容复杂）

点赞：单一Adapter，显示id ，头像，点赞内容，微博图片，感谢按钮：对话框（自定义）

私信：显示私信用户名单，点击进入私信界面（对话界面的实现）按照私信时间排序

1. 底部栏同上

## 六、底部栏

点击不同的tab显示不同的页面，图片来自于微博国际版的解压文件

# 开发日志

## 项目重启概述：

## 分为三个阶段：

## 在原来开发的基础上，增加微博基本功能，包括微博流的显示，微博详情页的查看，微博的发送，微博的评论及其查看。

## 在完成基本功能之后，进行其余功能的实现，包括搜索，设置，夜间模式，清除缓存等等。

## 第三，对UI更改外观，加入动画，调整位置，包括登录界面修改，按键动画实现等等。对项目进行必要优化，包括抽象接口，使用框架。

2018年1月12日

问题首要：微博主页流的显示，以及加入volley网络模块

Volley模块添加成功！对获取的Json进行解析并填入时间线

定义Bean类，用Gson。首先，需知微博时间线的数据种类：

1. 用户头像
2. 用户名
3. 微博发布时间
4. 微博正文
5. 微博图片（仅显示三张）
6. 微博视频
7. 微博转发内容：1）用户名2）分享内容3）分享图片4）分享视频
8. 用户手机标识
9. 赞数，评论数，转发数

根据开发者文档获得范例如下

**{**

**"statuses": [**

**{**

**"created\_at": "Tue May 31 17:46:55 +0800 2011",**

**"id": 11488058246,**

**"text": "求关注。"，**

**"source": "<a href="http://weibo.com" rel="nofollow">新浪微博</a>",**

**"favorited": false,**

**"truncated": false,**

**"in\_reply\_to\_status\_id": "",**

**"in\_reply\_to\_user\_id": "",**

**"in\_reply\_to\_screen\_name": "",**

**"geo": null,**

**"mid": "5612814510546515491",**

**"reposts\_count": 8,**

**"comments\_count": 9,**

**"annotations": [],**

**"user": {**

**"id": 1404376560,**

**"screen\_name": "zaku",**

**"name": "zaku",**

**"province": "11",**

**"city": "5",**

**"location": "北京 朝阳区",**

**"description": "人生五十年，乃如梦如幻；有生斯有死，壮士复何憾。",**

**"url": "<http://blog.sina.com.cn/zaku>",**

**"profile\_image\_url": "<http://tp1.sinaimg.cn/1404376560/50/0/1>",**

**"domain": "zaku",**

**"gender": "m",**

**"followers\_count": 1204,**

**"friends\_count": 447,**

**"statuses\_count": 2908,**

**"favourites\_count": 0,**

**"created\_at": "Fri Aug 28 00:00:00 +0800 2009",**

**"following": false,**

**"allow\_all\_act\_msg": false,**

**"remark": "",**

**"geo\_enabled": true,**

**"verified": false,**

**"allow\_all\_comment": true,**

**"avatar\_large": "<http://tp1.sinaimg.cn/1404376560/180/0/1>",**

**"verified\_reason": "",**

**"follow\_me": false,**

**"online\_status": 0,**

**"bi\_followers\_count": 215**

**}**

**},**

**...**

**],**

**"ad": [**

**{**

**"id": 3366614911586452,**

**"mark": "AB21321XDFJJK"**

**},**

**...**

**],**

**"previous\_cursor": 0,                   // 暂未支持**

**"next\_cursor": 11488013766,    // 暂未支持**

**"total\_number": 81655**

**}**

获取的Json数据可以得出

1. 微博创建时间 created\_at
2. 微博内容 text
3. 微博来源 source
4. 是否收藏（以后考虑） favorite
5. 缩略图片地址 thumbnail\_pic
6. 中等尺寸图片地址 bmiddle\_pic
7. 原始图片地址 original\_pic
8. 微博用户名 user
9. 转发微博的信息 retweeted\_status(与微博所含内容相同）
10. 转发数，评论数，赞数 reposts\_count,comments\_count,attitudes\_count

思考：自己发微博将所有可能的情况全部凑齐，然后进行该条的Json获取，分析，建立相关的Bean类，最后将解析的数据导入

导入Gson。