

# 同城汽配Android 客户端 push\_library 使用说明

## 1.module 详解

### 1.1 libs 下存放友盟和小米推送SDK

### 1.2 Java 文件下为具体push SDK具体封装和调用

#### 1.2.1 miui包MiuiReceiver 自定义miui推送回调等。

#### 1.2.2 model具体数据模型类

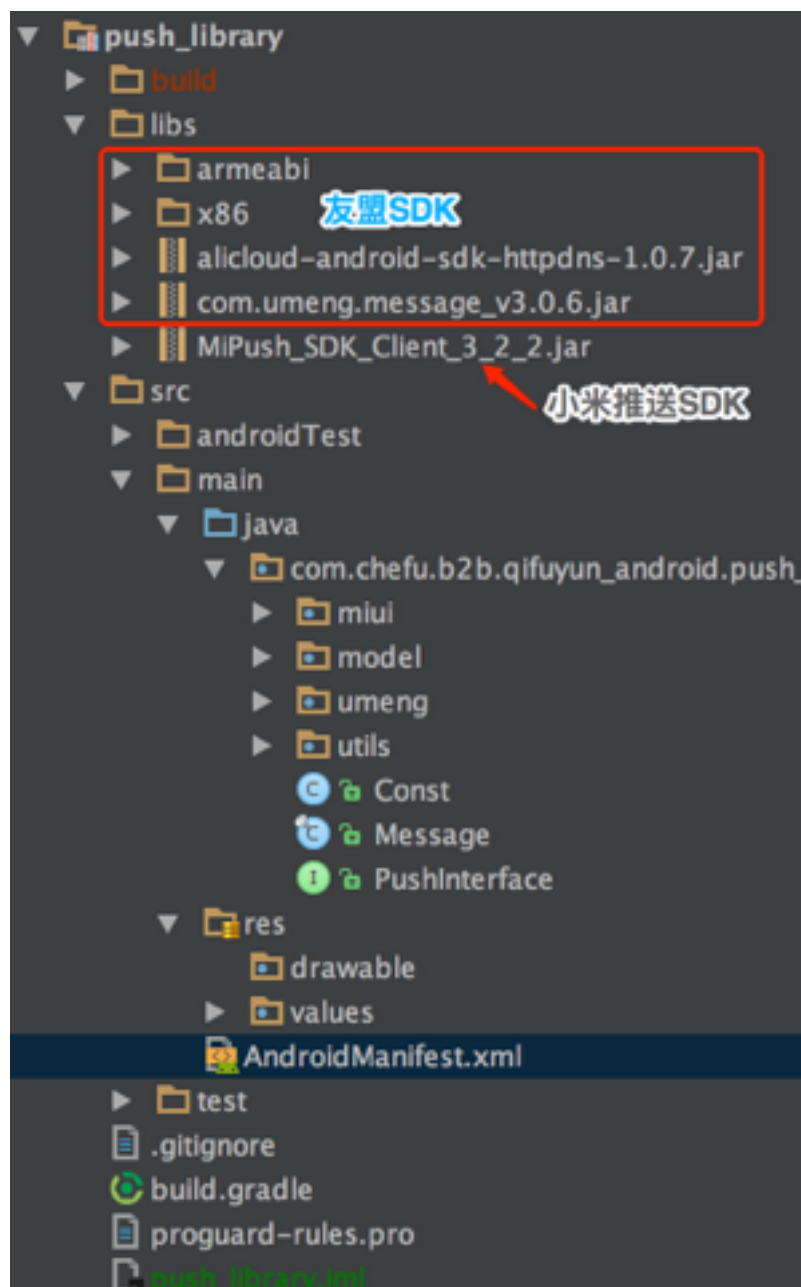
#### 1.2.3 umeng 友盟推送API封装 NotificationClickerHandler 自定义通知栏处理

UmengManger 包含友盟消息推送重要API调用 包括 注册, 添加别名, 订阅等方法

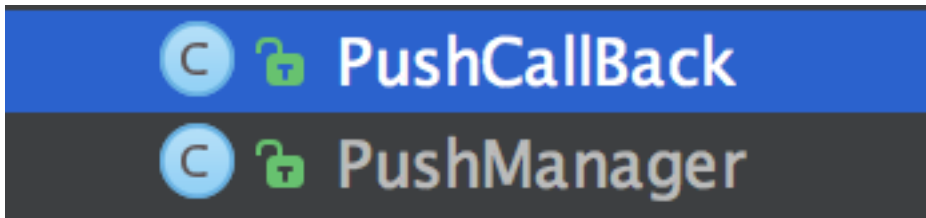
#### 1.2.4 utils 存放module所需工具类

#### 1.2.5 Const 静态变量

#### 1.2.6 PushInterface push 回调接口 包含注册成功, 点击通知栏, 消息到达客户端等回调



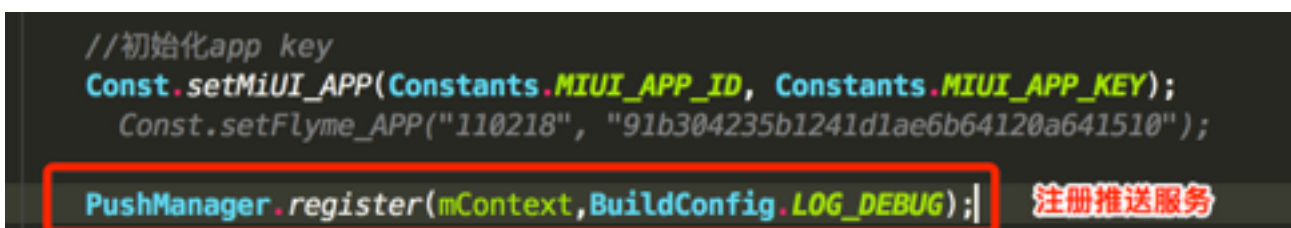
## 2.module 使用



### 2.1 PushCallBack 实现pushinterface 做具体回调操作

2.2 PushManager 是推送管理类包含如：注册推送，取消注册，设置别名，获取device\_token，删除别名，关闭推送，开启推送。

## 3.使用



在自定义application中添加register()方法注册推送服务即可使用推送服务

## 4.push\_library 源码详解

需求分析：目前推送平台包括，百度，华为，小米，友盟，极光，个推等 其原理大多是客户端与服务器保持长连接方式已达到消息推送功能。但是应用个别ROM厂商上运行，被杀死进程之后长连接断开导致消息不能实时到达，so 如果使用与手机系统相同的推送平台即可与手机系统共享系统级长连接，即可保证进程被杀死之后长连接不断裂消息仍能收到。所以只要能在不同手机上开启不同推送即可

### 4.1 判断手机是什么品牌？ ROM？

Android 常见的ROM类型表示 参考 <http://blog.csdn.net/u014608640/article/details/52849072>

可使用一下代码判断具体手机ROM

```
/**
 * rom 标识
 */
```

```

//华为
private static final String KEY_EMUI_VERSION_CODE = "ro.build.version.emui";
//小米
private static final String KEY_MIUI_VERSION_CODE = "ro.miui.ui.version.code";
private static final String KEY_MIUI_VERSION_NAME = "ro.miui.ui.version.name";
private static final String KEY_MIUI_HANDY_MODE_SF = "ro.miui.has_handy_mode_sf";
private static final String KEY_MIUI_REAL_BLUR = "ro.miui.has_real_blur";
//魅族
private static final String KEY_FLYME_ICON = "persist.sys.use.flyme.icon";
private static final String KEY_FLYME_PUBLISHED = "ro.flyme.published";
private static final String KEY_FLYME_FLYME = "ro.meizu.setupwizard.flyme";

/**
 * 华为rom
 *
 * @return
 */
private static boolean isEMUI() {
    try {
        final BuildProperties prop = BuildProperties.newInstance();
        return prop.containsKey(KEY_EMUI_VERSION_CODE);
    } catch (Exception e) {
        e.printStackTrace();
        return false;
    }
}

/**
 * 小米rom
 *
 * @return
 */
private static boolean isMIUI() {
    try {
        final BuildProperties prop = BuildProperties.newInstance();

        return prop.containsKey(KEY_MIUI_VERSION_CODE)
            || prop.containsKey(KEY_MIUI_VERSION_NAME)
            || prop.containsKey(KEY_MIUI_REAL_BLUR)
            || prop.containsKey(KEY_MIUI_HANDY_MODE_SF);
    } catch (Exception e) {
        e.printStackTrace();
        return false;
    }
}

/**
 * 魅族rom
 *
 * @return
 */
private static boolean isFlyme() {

```

```

try {
    final BuildProperties prop = BuildProperties.newInstance();
    return prop.containsKey(KEY_FLYME_ICON)
        || prop.containsKey(KEY_FLYME_PUBLISHED)
        || prop.containsKey(KEY_FLYME_FLYME);
} catch (Exception e) {
    e.printStackTrace();
    return false;
}
}
}

```

```

/**
 * 判断手机rom
 * @return
 */
public static Target rom() {

    Target mTarget;

    if (isEMUI()) {
        mTarget = Target.EMUI;
    } else if (isMIUI()) {
        mTarget = Target.MIUI;
    } else if (isFlyme()) {
        mTarget = Target.FLYME;
    } else {
        mTarget = Target.UMENG;
    }

    Log.d(TAG, "rom: mTarget =" + mTarget);

    return mTarget;
}

```

4.2 根据不同手机ROM启动不同推送SDK  
具体代码请参考module中PushManager.java

4.3 集成过程遇到的相关问题参考：

友盟：

<http://dev.umeng.com/push/android/integration>

<http://bbs.umeng.com/thread-5911-1-1.html>

小米：

<http://dev.xiaomi.com/doc/?p=544>

PS:

文档如有错误请斧正

源码，文档等持续更新维护中欢迎提出建议和意见

史守峰 2017/3/14