

# perfect\_mqtt SDK 使用说明

## 一.本文内容

本文面向 **Android** 开发者，提供简单、快速的 perfect\_mqtt SDK 集成指导

## 二.SDK 包包含内容

- 1、 doc---说明文档
- 2、 demo---简单调试用例
- 3、 libs 包----包括 perfect\_mqtt.jar 包、mqttv3.jar 包

## 三.SDK 集成步骤

- 1、 把 libs 包下的两个 jar 包拷贝至自己的应用程序目录并 Build Path(新版本 eclipse 会自动 build)

- 2、 在 Manifest.xml 文件文件里加上

```
//-----推送服务-----
<service android:name="com.perfect.mqtt.PushService"/>
//-----系统启动检测-----
<receiver android:name="com.perfect.mqtt.CompletedBroadCast">
    <intent-filter>
        <action android:name="android.intent.action.BOOT_COMPLETED"/>
    </intent-filter>
</receiver>
//-----接收服务器推送的消息检测-----
<receiver android:name="你自定义的接收服务器消息的广播">
    <intent-filter>
        <action android:name="android.intent.action.perfect.push"/>
    </intent-filter>
</receiver>
//-----你的应用信息(切忌写在application内)-----
<meta-data android:name="appKey" android:value="你的应用程序名,且不超过
12位字符串,比如dotaApp"/>
```

并且加上以下权限

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.VIBRATE" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_
STATE" />
<uses-permission
android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
<uses-permission
android:name="android.permission.READ_PHONE_STATE"/>
```

- 3、 在你的工程里新建一个 BroadcastReceiver 类来接收服务器推送过来的消息(可以参照 demo 代码)

```
String topic = intent.getStringExtra("topic");//主题
String message = intent.getStringExtra("message");//消息内容
```

- 4、 在自己的应用启动入口处加上以下代码:

#### A. 开始连接服务器并订阅相关主题

String initTopic = “订阅的主题” ;//例如” WANMEI/MESSAGE1” ,

服务器端对应的主题是:appKey + “/” + initTopic

//String[] initTopic = {多个订阅的主题};例如

{” WANMEI/MESSAGE1” , ” WANMEI/MESSAGE2” }

String username = “用户名” ;

String password =” 密码” ;

MqttConnection.getInstance().init(context, initTopic,  
username, password);//单个主题

#### B. 订阅主题

String initTopic = “订阅的主题” ;

//String[] initTopic = {多个订阅的主题};

MqttConnection.getInstance().subscribeTopic (context,

```
initTopic);
```

### **C.取消相关的订阅主题**

```
String initTopic = “订阅的主题”;
```

```
//String[] initTopic = {多个订阅的主题};
```

```
MqttConnection.getInstance().unsubscribe(context,
```

```
initTopic);
```

### **D.停止连接**

```
MqttConnection.getInstance().stopConnect(context);
```