

有道云笔记 Android 端 SDK 文档

概述

本文主要描述 YNOTE SDK 的开放的接口设计，目标读者为开发人员，测试人员和架构师。

主要功能

从第三方发送笔记到有道云笔记

添加第三方应用的快捷启动到有道云笔记

从有道笔记启动注册的第三方应用

开发者注册

1. 注册账号
2. 添加 APP 从而获取 APP ID

应用名称	应用描述	应用下载市场连接
应用官方网站 (可选)	水印图标	高清图标
应用签名	应用报名	

3. 等待审核 - 可以开发调试，但是受限
4. 审核通过

API 使用指南

1. 下载有道云笔记 Android 端开发工具包

- 1) Ynote_sdk_sample 工程

样例工程，导出工程签名时需要使用工程中的签名文件 (debug.keystore)，
密码为 android

2) Libynotesdk.jar

开发 sdk 用到的包

3) YNoteSigFetcher.apk

取应用指纹工具

4) sdk 文档

2.申请应用程序 AppID

使用签名抽取工具 (sigFetcher) 取得应用指纹信息 (请确保应用的签名信息正确)，并到开发者应用程序登记页面进行 App 注册取得 AppId，为了笔记的跨平台处理，目前需要先注册 OpenApi

3.使用 sdk 提供的 api

[1]利用申请到的 AppId 生成 api 接口类

```
final IYNoteAPI api = YNoteAPIFactory.getYNoteAPI(this, AppId);
```

[2]注册 App 到有道云笔记

```
if(!api.isRegistered()){  
    api.registerApp();  
}
```

注册成功的应用可以在有道云笔记更多笔记中展示

[3]保存笔记到有道云笔记

1) 生成正文容器

```
YNoteContent mContent = new YNoteContent();
```

2) 为正文添加标题

```
mContent.setTitle("from sdk sample");
```

3) 为正文添加文本、图片、及附件，保持添加顺序

```
mContent.addObject(new YNotePlainTextContent(txt));  
mContent.addObject(new YNoteHtmlTextContent(txt));
```

使用图片的绝对路径或者 uri 来定位图片

```
mContent.addObject(new YNoteImageContent(uri));  
or  
mContent.addObject(new YNoteImageContent(absPath));  
or  
mContent.addObject(new YNoteImageContent(uri,absPath));
```

使用附件的绝对路径或者 uri 来定位附件

```
mContent.addObject(new YNoteAttachment (uri));  
or  
mContent.addObject(new YNoteAttachment (absPath));  
or  
mContent.addObject(new YNoteAttachment (uri,absPath));  
or
```

4) 生成发送笔记请求后设置笔记内容并发送

```
SendNoteRequest request = new SendNoteRequest();  
request.setYNoteContent(mContent);  
api.sendRequest(request);
```

[4]从有道云笔记启动第三方应用

1)只要成功注册到有道云笔记，就可以在更多笔记类型中展示应用的图标；
2)如果第三方实现了 packageName.ynoteapi.YNoteEntryActivity 并在 manifest 中设 activity 的 export=true，点击第三方应用图标时，笔记将调用这个页面，否则将直接启动应用程序；

3)如果工程使用代码混淆并使用 YNoteEntryActivity，则需要保持

YNoteEntryActivity 不被混淆，在 proguard.cfg 加上
-keep class packageName.ynoteapi.YNoteEntryActivity {*;}

[5]监听笔记打开事件

1) 使用权限

```
<uses-permission android:name="com.youdao.note.openapi.permission.LISTEN_OPENAPP" />
```

2) 注册 receiver

```
<receiver  
    android:name=".ynoteapi.YNoteOpenRegister"  
    android:permission="com.youdao.note.openapi.permission.LISTEN_OPENAPP" >  
    <intent-filter>  
        <action android:name="com.youdao.note.openapi.Intent.ACTION_OPEN_YNOTEAPP" />  
    </intent-filter>  
</receiver>
```

3) 简单的 receiver 只是处理注册事件

```
public class YNoteOpenRegister extends BroadcastReceiver {  
  
    @Override  
    public void onReceive(Context context, Intent intent) {  
        final IYNoteAPI api = YNoteAPIFactory.getYNoteAPI(context,  
            MainActivity.sAppId);  
        if (!api.isRegistered()) {  
            api.registerApp();  
        }  
    }  
}
```

主要类

YNoteAPIFactory

API 类工厂，取得 api 类

方法名	描述
IYNoteAPI getYNoteAPI(Context context, String appId)	取得有道云笔记 API 类

IYNoteAPI

Api 类接口，用于注册应用、发送笔记等

方法名	描述
void registerApp()	注册应用到有道云笔记 ,从而有道云笔记的插件栏里可以显示该应用

<code>void unregisterApp()</code>	取消注册
<code>void isRegistered()</code>	是否注册过
<code>boolean openYNoteApp()</code>	启动有道云笔记
<code>boolean isYNoteAppInstalled()</code>	检查有道云笔记是否安装
<code>boolean isYNoteAppSupportAPI()</code>	检查有道云笔记是否支持 API
<code>boolean sendRequest(BaseMessage request)</code>	发送请求到有道云笔记
<code>boolean sendResponse(BaseMessage response)</code>	发送反馈到有道云笔记 ()
<code>int getSupportAPIVersion()</code>	取得有道云笔记支持的 API 最低版本
<code>boolean handleIntent(Intent pIntent, IYNoteEventHandler pHandler)</code>	响应有道云笔记的请求和反馈

IYNoteEventHandler

事件响应类，响应有道云笔记的操作

方法名	描述
<code>void handleRequest(Intent requestIntent)</code>	当接收到有道云笔记发来的请求时调用
<code>void handleResponse(Intent reponseIntent)</code>	当接收到有道云笔记发来的响应时调用

BaseMessage

所有传递消息的父类

方法名	描述
<code>int getMsgType()</code>	消息的类型

<code>void setTransaction(String trans)</code>	设置消息回话 ID，默认设置为当前系统时间
<code>string getTransaction()</code>	取得消息回话 ID

`CreateNoteRequest` extends `BaseMessage`

从有道云笔记启动第三方应用时传递的信息类，用于标识传递的信息类型及扩展

`SendNoteRequest` extends `BaseMessage`

从第三方应用发送到有道云笔记的请求

方法名	描述
<code>int getType()</code>	请求的类型
<code>void setYNoteContent(IYNoteContent content)</code>	设置笔记正文
<code>void toBundle(Bundle bundle)</code>	将正文请求放入 bundle 中
<code>void fromBundle(Bundle bundle)</code>	从 bundle 中取得正文请求

`YNoteContent`

笔记正文容器类，可以添加正文内容对象

方法名	描述
<code>boolean addObject(YNoteContentObject contentObject)</code>	添加内容数据对象
<code>void setTitle(String title)</code>	设置标题
<code>String getTitle()</code>	取得标题
<code>void toBundle(Bundle bundle)</code>	保存内容到 bundle 中

<code>void fromBundle(Bundle bundle)</code>	从 bundle 中取得内容
---	----------------

`IYNoteContentObject`

笔记内容对象

方法名	描述
<code>int getType()</code>	返回数据对象
<code>void toBundle(Bundle bundle)</code>	把数据属性存到 bundle 里
<code>void fromBundle(Bundle bundle)</code>	从 bundle 里还原出数据属性

`YNotePlainTextContent` `extends` `YNoteContentObject`

平文本对象

方法名	描述
<code>void setPlainText(String txt)</code>	设置平文本内容
<code>String getPlainText()</code>	取平文本内容

`YNoteHtmlTextContent` `extends` `YNoteContentObject`

html 文本对象

方法名	描述
<code>void setHtmlText(String txt)</code>	设置带 html 格式的文本内容，html 标签可以包含在<body>中
<code>String getHtmlText()</code>	取带 html 格式的文本内容

`YNoteImageContent` `extends` `YNoteContentObject`

图片对象

方法名	描述
<code>void setPath(String path)</code>	设置图片绝对路径
<code>void setUri(Uri uri)</code>	设置图片 uri
<code>String getPath()</code>	取图片绝对路径
<code>Uri getUri()</code>	取图片 uri

`YNoteAttachment` **extends** `YNoteContentObject`

附件对象

方法名	描述
<code>void setPath(String path)</code>	设置附件绝对路径
<code>void setUri(Uri uri)</code>	设置附件 uri
<code>String getPath()</code>	取附件绝对路径
<code>Uri getUri()</code>	取附件 uri

`YNoteAPIConstants`

常量类