

掌阅游戏

接入帮助文档v1.0



2015-7-8

北京掌阅科技有限公司

幸鑫/编写

目录

[引言 3](#_Toc424140723)

[编写目的 3](#_Toc424140724)

[适用对象及范围 3](#_Toc424140725)

[概括 3](#_Toc424140726)

[专业术语说明（请勿直接使用） 3](#_Toc424140727)

[1. 接入前准备 4](#_Toc424140728)

[2.登录接入 4](#_Toc424140729)

[2.1 测试DEMO参数 4](#_Toc424140730)

[2.2 权限设置 5](#_Toc424140731)

[2.3 修改AndroidManifest.xml 5](#_Toc424140732)

[AndroidManifest.xml在使用iReader授权界面的activity里面加上属性: 5](#_Toc424140733)

[android:taskAffinity="com.chaozh.iReader.taskauthor" 5](#_Toc424140734)

[2.4 启动页要求 6](#_Toc424140735)

[使用我们的启动引导页作为启动Activity，不要在该Activity实现任何掌阅登陆授权代码。确保该Acitivity 的android:taskAffinity为默认（不填即可），不同于授权页的com.chaozh.iReader.taskauthor。（要求有独立的启动页与授权页，保证不仅为一个activity，启动页不允许放taskaffinxx） 6](#_Toc424140736)

[2.5 闪屏、icon与包名要求 6](#_Toc424140737)

[2.6 获取账号 6](#_Toc424140738)

[2.7 获取账号返回结果 6](#_Toc424140739)

[public interface IAccountProviderListener { 6](#_Toc424140740)

[void onError(int code) 6](#_Toc424140741)

[void onResult(boolean success,String uid,String token); 6](#_Toc424140742)

[2.8 开放API接口详解 7](#_Toc424140743)

[附加信息： 8](#_Toc424140744)

[状态码 8](#_Toc424140745)

[签名算法 9](#_Toc424140746)

[Api 版本: 1.0 9](#_Toc424140747)

[参数需urlencode 9](#_Toc424140748)

[部署环境 9](#_Toc424140749)

[3支付接入 9](#_Toc424140750)

[3.1 资源包 9](#_Toc424140751)

[3.2导入资源 9](#_Toc424140752)

[1. 引入项目工程 ZhangyuePay 9](#_Toc424140753)

[2. 拷贝 ZhangyuePay\res\values 下中 ids.xml 和 public.xml 到项目中 res\values 下 9](#_Toc424140754)

[3.3 修改Manifest 10](#_Toc424140755)

[1. 申明 SDK 版本 10](#_Toc424140756)

[<uses-sdk android:minSdkVersion="8" android:targetSdkVersion="14" /> 10](#_Toc424140757)

[3.4 支付接口调用 11](#_Toc424140758)

[3.5 订单通知接口 12](#_Toc424140759)

# 引言

## 编写目的

为第三方应用提供便捷的掌阅登录、支付体系。

## 适用对象及范围

适用于适用于策划人员、系统设计人员、开发工程师和测试工程师。

## 概括

北京掌阅科技SDK 开发包提供给任何可信任的第三方合作方伙伴，可信任的第三方合作伙伴通过掌阅 SDK 的开发规范实现在 Android 应用中使用掌阅账号一键登录、支付等功能。 SDK开发包已经包含开发规范所涉及的北京掌阅计SDK 开发标准要求的实现， 合作伙伴只需将 SDK 集成到应用中，并调用相应的接口函数即可完成对开发规范所涉及的应用标准要求的实现。

## 专业术语说明（请勿直接使用）

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 描述 | 样例 |
| 1. | PARTNER | 商户ID,掌阅分配用户区分合作方的标识. | 12BD135B7DB2 |
| 2 | APP\_ID | 应用ID,掌阅分配用于区分应用的标识. | d70e38e25996fec27584 |
| 3 | ORDER\_ID | 掌阅订单,格式为年月日时分秒+6位随机数 | A15070221112885 |
| 4 | SUBJECT | 商品名.用于识别商品名称与数量 | 100元宝 |
| 5 | TOTAL\_FEE | . 商品价格(元为单位).浮点形式 | 233.00 |
| 6 | INFO\_MORE | 扩展字段，用于透传扩展参数 | {"open\_uid":"xxx"} |
| 7 | OPEN\_UID | Ireader账号标识,扩展中透传JSON格式 | 54d67be54084ec84  359cd893fd53062c |
| 8 | Token | 访问令牌，用于判断用户登录是否过期 | 无 |
| 9 | pkcs | 私钥，用于签名使用 | 无 |
| 10 | md5key | MD5加密key，用于支付回调加密 | 200ba8ac6 |

# 接入前准备

接入前请提供接入负责需要的合作方、应用信息，并按要求建立包名；

接入负责在收到资料后会提供第三方参数，在接下来的接入环节必不可少。

参数基本包含了商户ID、APPID、PHP私钥(pkcs1)、JAVA私钥(pkcs8)、回调地址加密key。

接入时环境等信息请看登录SDK[附加信息](#_附一_状态码)。

# 2.登录接入

## 2.1 测试DEMO参数

APP\_ID: d70e38e25996fec27584

PHP私钥(pkcs1):

-----BEGIN RSA PRIVATE KEY-----

MIICXAIBAAKBgQCtkB1GnAnw4qSqTtmkDKZdTr/Mx6XVO1FtcSOlCLtLhWtNhafB

c0FwCp4MMXVDOIhM42Hsfoz/KK+90zbVFMWB0PbbWizgvZFnjOnSvA23h1nXPTcA

VLHERoL2fnfEQsnVFfVbKmMRFtEzK4g+1Fksu2C4eHMfzDtf/gGiBp1FlQIDAQAB

AoGAVEkSDoIkSbimopYcFxCVkeZ2OioSHFTsHCu/VZvB6NDn5QOtDcK82n4tdlpm

SbigaEZb/Y8PRbIqdhKmJCAsMSna9NVtBiD+diH5qoneZMD0unZmWnYQ+2OVZ9X7

4UhKVA9O71u7wPQXMrOuj8XeTUj4kO/0jO9vxQ1W4EIj40ECQQC3LFf5rSir9dA0

4DSK0xop7S+qCL7jPy8pzIZ7me99uAWKExp6C3qqbuJHOus95az2eLBAB58vkv6r

TiqWankZAkEA8pGfUVyaiDMspfNpwDkY9EPH7xDXPxDrdxXv/fIwDHwphs7pAbhA

b+PQVGfPliUYdhQ1hL43jw7iuMbSscDz3QJAHJyrnU+EA+BgxnGjI3A4LQZkpJ27

G4oaFuBgvVvdQfkFvHyhbSDv50FszQ2SA1gl/VX6yfWm0uqWJzINivNUuQJBALyg

dwaa/2RYfyjRX3PmMryIFOqX6voU2zYwrf6JHconp1c4gUMgF6OqwS9RxsBYOmxg

DLob/TJW/gF4QxGNHVUCQAYsvxZWh6SdkSsgkARssNcL1M3U1NfEQStAwhbI0Q/g

dTZWG1/i8G/5DRZ0FetEGPFe5t2NTkXlT+CYUIzSo1Q=

-----END RSA PRIVATE KEY-----

JAVA私钥(pkcs8):

-----BEGIN PRIVATE KEY-----

MIICdgIBADANBgkqhkiG9w0BAQEFAASCAmAwggJcAgEAAoGBAK2QHUacCfDipKpO

2aQMpl1Ov8zHpdU7UW1xI6UIu0uFa02Fp8FzQXAKngwxdUM4iEzjYex+jP8or73T

NtUUxYHQ9ttaLOC9kWeM6dK8DbeHWdc9NwBUscRGgvZ+d8RCydUV9VsqYxEW0TMr

iD7UWSy7YLh4cx/MO1/+AaIGnUWVAgMBAAECgYBUSRIOgiRJuKailhwXEJWR5nY6

KhIcVOwcK79Vm8Ho0OflA60Nwrzafi12WmZJuKBoRlv9jw9Fsip2EqYkICwxKdr0

1W0GIP52Ifmqid5kwPS6dmZadhD7Y5Vn1fvhSEpUD07vW7vA9Bcys66Pxd5NSPiQ

7/SM72/FDVbgQiPjQQJBALcsV/mtKKv10DTgNIrTGintL6oIvuM/LynMhnuZ7324

BYoTGnoLeqpu4kc66z3lrPZ4sEAHny+S/qtOKpZqeRkCQQDykZ9RXJqIMyyl82nA

ORj0Q8fvENc/EOt3Fe/98jAMfCmGzukBuEBv49BUZ8+WJRh2FDWEvjePDuK4xtKx

wPPdAkAcnKudT4QD4GDGcaMjcDgtBmSknbsbihoW4GC9W91B+QW8fKFtIO/nQWzN

DZIDWCX9VfrJ9abS6pYnMg2K81S5AkEAvKB3Bpr/ZFh/KNFfc+YyvIgU6pfq+hTb

NjCt/okdyienVziBQyAXo6rBL1HGwFg6bGAMuhv9Mlb+AXhDEY0dVQJABiy/FlaH

pJ2RKyCQBGyw1wvUzdTU18RBK0DCFsjRD+B1NlYbX+Lwb/kNFnQV60QY8V7m3Y1O

ReVP4JhQjNKjVA==

-----END PRIVATE KEY-----

回调地址(MD5)加密Key: 200ba8ac6

## 2.2修改AndroidManifest.xml

## AndroidManifest.xml在使用iReader授权界面的activity里面加上属性:

## android:taskAffinity="com.chaozh.iReader.taskauthor"

android:allowTaskReparenting="false"

禁止将该属性应用于启动Activity

## 2.3启动页要求

## 使用我们的启动引导页作为启动Activity，不要在该Activity实现任何掌阅登陆授权代码。确保该Acitivity 的android:taskAffinity为默认（不填即可），不同于授权页的com.chaozh.iReader.taskauthor。（要求有独立的启动页与授权页，保证不仅为一个activity，启动页不允许放taskaffinxx）

## 2.4闪屏、icon与包名要求

启动画面需要添加掌阅闪屏（启动图），icon需要添加掌阅角标,包名应使用“.zy”为后缀，禁止使用包含”zhangyue”的包名。 所需文件与素材均在接入负责提供的“掌阅素材”内。

## 2.5获取账号

初始化：helper，传入在掌阅方申请到的appid

AccountHelper helper = new AccountHelper("xxxxxxxx", new IAccountProviderListener() )

public final void getAccountWithDlg(final Context ctx, boolean refreshToken, String privateKey)

refreshToken：true，表示重新拿令牌（主要用于与服务器验证，发现令牌无效时使用）

refreshToken：false 表示拿缓存令牌

|  |  |  |  |
| --- | --- | --- | --- |
| 调用接口 | getAccount | | |
| 输入参数 | 参数名称 | 参数类型 | 描述信息 |
| Context ctx | activity | 登陆activity |
| refreshToken | boolean | 是否刷新token令牌  （如为true则会重新让用户授权获取token） |
| privateKey | String | 掌阅分配的JAVA私钥(pkcs8) |

## 2.6获取账号返回结果

## public interface IAccountProviderListener {

## void onError(int code)

## void onResult(boolean success,String uid,String token);

}

|  |  |  |  |
| --- | --- | --- | --- |
| 回调 | IAccountProviderListener | | |
| 返回结果成功时 | 参数名称 | 参数类型 | 描述信息 |
| success | boolean | 是否登录成功 |
| uid | string | 登录ID |
| Token | String | 登录令牌 |
| 返回结果失败时 | code | int | 状态码  -1; //--网络io异常  -10; //--用户主动取消  -11; //--没有账号  -12; //--用户登录失败  -13; //--用户web登陆失败  -100; //--没有iReader  -101; //--iReader版本过久  -102; //--接口访问限制  -103; //--接口访问失败  -104; //--APP非法  -500; //--未知错误 |

## 2.7 开放API接口详解（每次登录时服务端必须调用该接口判断）

|  |  |  |
| --- | --- | --- |
| 调用接口 | /open/token/check | |
| 输入参数 | 参数名称 | 描述信息 |
| app\_id | 应用ID |
| open\_uid | 掌阅用户UID |
| access\_token | 访问令牌 |
| timestamp | 时间戳 |
| sign\_type | RSA |
| sign | 签名 |
| version | API版本 |
| 返回参数 | code | 状态码  0:'成功'  30200:'签名无效',  30202:'应用非法',  30201:'授权码无效', |
| msg | 错误信息 |
| 返回实例 | {  "code":状态码  0:'成功',  30200:'签名无效',  30202:'应用非法',  30201:'授权码无效',  "msg": "错误信息",  "body": None  } | |

## 附加信息：openuid为“gameyk\_”开头的不需要调用此接口。

## 状态码

1、系统状态码

0: 'OK',

1: '参数错误',

2: '程序内部错误',

3: '第三方接口超时',

4: '接口不存在',

2、业务状态码

30000:'当前操作被锁定',

30200:'签名非法',

30201:'动态授权码非法',

30202:'非法的应用',

30210:'用户没有授权',

30211:'应用授权被取消',

30212:'访问令牌非法',

30213:'访问令牌过期',

30214:'刷新令牌非法',

30215:'刷新令牌过期'

## 签名算法

1、应用把RSA公钥传给掌阅

2、将参数值不为空的参数按参数名字母序排序后以&连接起来进行RSA签名

python表示:

def sign\_by\_rsa(private\_key, param):

sign\_scheme = PKCS115\_SigScheme(private\_key)

data = '&'.join(['%s=%s' % (k, v) for k, v in sorted(param.items()) if v])

h = SHA.new(data)

return base64.b64encode(sign\_scheme.sign(h))

## Api 版本: 1.0

参数需urlencode。

## 部署环境

<https://uc.ireader.com>

# 3支付接入

## 3.1资源包

包含ZhangyueSDK\_common项目与zysdks.jar。

## 3.2导入资源

引入项目工程 ZhangyueSDK\_common项目

## 3.3 修改Manifest

## 1. 申明 SDK 版本

## <uses-sdk android:minSdkVersion="8" android:targetSdkVersion="14" />

1. 申明 Activity，支付强制要求竖屏。

<!-- 掌阅支付主Activity -->

<activity android:name="com.zhangyue.pay.ui.ActivityPay"

android:theme="@style/zypay\_activity\_fee"

android:screenOrientation="portrait"

android:configChanges="orientation|keyboardHidden|screenSize"></activity>

<!-- alipay sdk begin -->

<activity

android:name="com.alipay.sdk.app.H5PayActivity"

android:configChanges="orientation|keyboardHidden|navigation|screenSize"

android:exported="false"

android:screenOrientation="behind"

android:windowSoftInputMode="adjustResize|stateHidden" >

</activity>

//聚合支付

<activity

android:name="com.ipaynow.wechatpay.plugin.inner\_plugin.wechat\_plugin.activity.WeChatNotifyActivity"

android:theme="@android:style/Theme.NoDisplay"

android:windowSoftInputMode="stateHidden"

android:configChanges="orientation|screenSize">

</activity>  
<!-- sdk end -->

1. 申明权限

<!-- 权限管理 -->  
<uses-permission android:name="android.permission.READ\_PHONE\_STATE" />  
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />  
<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />  
<uses-permission android:name="android.permission.CHANGE\_NETWORK\_STATE"/>  
<uses-permission android:name="android.permission.SEND\_SMS" />  
<uses-permission android:name="android.permission.VIBRATE"/>  
<uses-permission android:name="android.permission.WRITE\_SETTINGS"/>  
<uses-permission android:name="com.android.launcher.permission.READ\_SETTINGS"/>  
<uses-permission android:name="com.android.launcher.permission.INSTALL\_SHORTCUT"/>  
<uses-permission android:name="android.permission.ACCESS\_WIFI\_STATE"/>  
<uses-permission android:name="android.permission.CHANGE\_WIFI\_STATE"/>  
<uses-permission android:name="android.permission.RECEIVE\_BOOT\_COMPLETED"/>  
<uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION"/>  
<uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION"/>

## 支付接口调用

参数解释

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 参数类型 | 必填 | 参数描述 | 参数结构（或范例） |
| KEY\_PAY\_INFO\_APP\_ID | String | 必填 | 应用ID | 字符串 |
| KEY\_PAY\_INFO\_BODY | String | 必填 | 计费描述 | 元宝可用于商城道具购买. |
| KEY\_PAY\_INFO\_ORDER\_ID | String | 必填 | 商户订单 | 年月日时分秒+6位随机数 |
| KEY\_PAY\_INFO\_TOTAL\_FEE | String | 必填 | 商品价格(单位:元) | 浮点，例:5.00 |
| KEY\_PAY\_INFO\_PARTNER | String | 必填 | 商户ID | 字符串 |
| KEY\_PAY\_INFO\_SUBJECT | String | 必填 | 商品名 | 商品名称，例: |
| KEY\_PAY\_INFO\_MORE | String | 必填 | 扩展字段 | 透传扩展参数，open\_uid |
| open\_uid |  | 必填 | 用户识别 | {"open\_uid":"参数"} |

代码判断（附件中 DEMO 有此代码）：

1. 唤起支付插件

JSONObject json = new JSONObject();  
try{  
json.put(CONSTANT.KEY\_PAY\_INFO\_APP\_ID, "d70e38e25996fec27584");  
json.put(CONSTANT.KEY\_PAY\_INFO\_BODY, "计费");  
json.put(CONSTANT.KEY\_PAY\_INFO\_ORDER\_ID, "20140813122345123456");  
json.put(CONSTANT.KEY\_PAY\_INFO\_TOTAL\_FEE, "5");  
json.put(CONSTANT.KEY\_PAY\_INFO\_PARTNER, "570A8409C21A ");  
json.put(CONSTANT.KEY\_PAY\_INFO\_SUBJECT, "20 钻石");  
json.put(CONSTANT.KEY\_PAY\_INFO\_MODE, "{\"open\_uid\":\"54d67be54084ec84359cd893fd53062c\"}");  
}catch(Exception ex){  
}  
PayTask payTask = new PayTask(this);  
payTask.pay(json.toString(),new PayListener() {  
@Override  
public void onPayError(int code, String msg) {  
//支付 SDK 异常通知  
Log.e("LOG", msg);  
}  
});

1. 支付通知代码片段

@Override  
protected void onActivityResult(int requestCode, int resultCode, Intent data) {  
switch(requestCode){  
case CODE.CODE\_REQUEST\_PAY:  
Log.e("LOG", "FeeResult:"+resultCode);  
if(resultCode == Activity.RESULT\_OK){  
//--支付成功  
}else{  
//--支付失败  
}  
break;  
}  
}

## 订单通知接口

1. 通知机制

此订单通知接口为 HTTP-POSE 通知机制，通知失败了，每隔一分钟通知一次，一共是 10 次；如果返回的结果不是 success 或者网络不通则一个小时后继续通知。

1. 通知参数如下

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数类型 | 参数描述 | 参数样例 |
| merId | String | 商户ID，合作方标识。 | 12BD135B7DB2 |
| appid | String | 应用id，应用标识 | d70e38e25996fec27584 |
| orderId | String | 掌阅订单号，用于记录订单 | A15070221112885 |
| merOrderId | String | 商户订单 | 123456789 |
| payAmt | String | 订单金额，为浮点 | 5.00 |
| transTime | String | 日期,格式:yyyyMMddHHmmss | 20140709144722 |
| orderStatus | String | 订单状态，1为成功，其他失败 | 1 |
| errorCode | String | 错误码 | 1 |
| errorMsg | String | 错误信息 | Success |
| rechargeType | String | 充值类型 | 11 |
| md5SignValue | String | MD5 签名：签名方式中的 key 由掌阅分配，签名方式为MD5(merId|appId|orderId|payAmt|  md5Key) | 分配 key 的签名：  26d441fead90809b3cadba0985d4a08  1 |

1. 通知范例

merId=ZY1406101100  
appId=B10221  
transData={"merId":"ZY1406101100","appId":"B10221","orderId":"A14070921111114","merOrderId":"123456789","p  
ayAmt":"5","transTime":"20140709144722","orderStatus":"1","errorCode":"1","errorMsg":"success","rechargeType":"1  
1","md5SignValue":"91f9bc99f1235166b70166d5d8c8db71"}  
transData: JSON 字符串

1. 通知返回信息说明

以orderStatus状态为准，订单状态，1为成功，其他失败

errorMsg成功返回： success 失败返回： 任意值

# 4启动和退出弹窗

1：初始化(demo 中代码)

**private** **void** initShowGameDlg(){

ZyGameManager.getInstance().setGameParams(

CONSTANT.KEY\_GAME\_PARAM\_APP\_ID, //appid (key) "a2e761ffa322049992ae");//appid(value)

ZyGameManager.getInstance().setGameParams(

CONSTANT.KEY\_GAME\_PARAM\_OPEN\_UID, //openuid(key) "3d190341efa9429894a44d40735fc28c");//openuid(value)

ZyGameManager.getInstance().init(**this**);//context

}

2：设置启动弹窗

**private** **void** startShowDlg(){

ZyGameManager.*getInstance*().showEvent(MainActivity.**this**, ZyGameEventLocation.*ByStart*, **new** IZyGameEventsListener() {

@Override

**public** **void** onJoin() {//确定

Log.*e*("LOG", "join");

//finish();

}

@Override

**public** **void** onExit(**int** code) {//取消

Log.*e*("LOG", "exit:"+code);

}

});

}

3：设置退出弹窗

**private** **void** exitShowDlg(){

ZyGameManager.*getInstance*().showEvent(**this**, ZyGameEventLocation.*ByExit*, **new** IZyGameEventsListener() {

@Override

**public** **void** onJoin() {

Log.*e*("LOG", "join");

finish();

}

@Override

**public** **void** onExit(**int** code) {

Log.*e*("LOG", "exit:"+code);

finish();

}

});

}