# 平台介绍

椰子小游戏平台是XXXXX

# 接入引导

## 接入说明

### 接入流程



### 获取参数

### 结算与合同

1. 游戏内购：扣除5%的通道费后按照各个流量平台分成策略进行分成。

2、广告变现：CP可获得50%的广告收入分成政策。

### 资质要求

CP需具备以下资质：《计算机软件著作权登记证书》、《游戏自审自查报告》、《广电总局版号批文》、《文化部备案信息》。

## 评测说明

CP将游戏发送给商务进行评测，评测时间预计需要2~3个工作日，结果会以邮件形式回复，通过评测的游戏可继续走接入流程，邮件格式如下：

评测邮件格式：

邮件标题：椰子小游戏评测+《产品名》+公司名+单机/对战/网游+H5/小游戏

邮件发送：xxxx@chukong-inc.com

内容格式：

游戏名称：XXXX

游戏类型：XXXXX（游戏玩法：XXXXX）

评测：H5链接、二维码或APK游戏包

已上线的平台：XXXXXX

## 接入规范

### 游戏规范

待补充

### 素材规范

待补充

# 技术对接

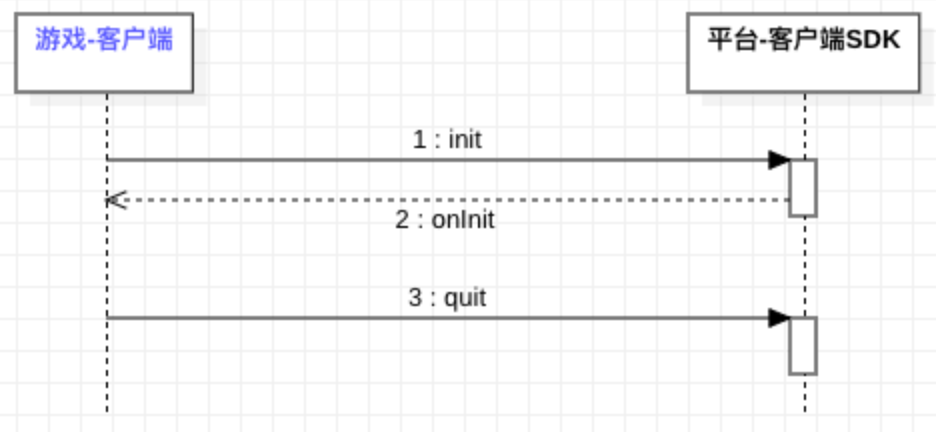
对接流程如下：

1. 通过开放平台获取对接参数gameId、gameKey、gameSecret。[前往](#_获取参数)
2. [下载对接平台SDK。](http://www.cocos.com/)
3. [使用自测工具验证是否完成接入。](#_自测工具)

## 非对战游戏

### 客户端接入

**非对战游戏接口调用时序图如下：**



说明：

1. 游戏客户端调用平台客户端init命令，初始化
2. 游戏客户端收到平台客户端onInit消息，获得init命令的返回结果
3. 游戏客户端调用平台客户端quit命令，结束游戏

#### 导入资源

1. 游戏客户端的启动文件为：index.html文件。
2. 游戏客户端需要在游戏中引用GameSDK\_Cocos.js文件。
3. 如：在index.html中的开始位置加入：
4. <script type = "text/javascript" src="./GameSDK\_Cocos.js" ></script>

#### 2.调用接口

1. **初始化**

调用平台任何其他的函数前，都要先调用该函数请求初始化SDK。初始化结果将会通过onInit消息返回给游戏。

函数：

GameSDK.init( version, gameId, gameKey, gameSecret )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| version | 接口版本号 | int | 非空 | 当前版本为：1 |
| gameId | 游戏Id | int | 非空 | 后台游戏资料中获得 |
| gameKey | 游戏Key | string | 非空 | 后台游戏资料中获得 |
| gameSecret | 游戏签名 | string | 非空 | 后台游戏资料中获得 |

1. **游戏退出**

调用该函数后平台会立即结束游戏并关闭游戏窗口。

函数：

GameSDK.quit( reason )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| reason | 退出原因 | int | 非空 | 1、正常退出；2、异常退出 |

1. **设置屏幕朝向**

调用该函数可以设置游戏时的屏幕显示朝向。

函数：

GameSDK.setOrientation( orientation )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| orientation | 朝向 | int | 非空 | 1. 横屏； 1、竖屏 |

1. **设置声音**

调用该函数可以设置游戏时声音是否开启以及音量大小。

函数：

GameSDK.setAudio( enable, volume )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| enable | 是否开启 | int | 非空 | 0、关闭；1、开启 |
| volume | 声量 | int | 非空 | 1~100 |

1. **设置游戏加载进度（SDK版本 >= 2）**

SDK版本2开始，平台增加了统一的游戏加载进度界面，用于游戏后台加载时显示。

游戏需要在初始化后，通过此函数报告游戏加载进度。加载界面将显示“加载中...”

当进度>=100%时，如果是对战类游戏，加载界面将提醒用户“等待对手进入中...”，并且游戏需要在对手都进入房间后，调用hideLoadProgress函数关闭加载界面。

当进度>=100%时，如果是非对战类游戏，加载界面将提醒用户“加载完成”，游戏需调用hideLoadProgress函数关闭加载界面。

函数：

GameSDK. setLoadProgress (progress)

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| progress | 加载进度 | int | 非空 | 1-100 |

1. **隐藏游戏加载进度（SDK版本 >= 2）**

用于关闭加载进度界面。此后玩家才可以和游戏交互。

函数：

GameSDK. hideLoadProgress ()

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| 无 |  |  |  |  |

1. **支付**

函数：

GameSDK.pay(orderId, goodsName, goodsDesc, orderAmount, extension, notifyURL)

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| orderId | 订单号 | string | 非空 |  |
| goodsName | 商品名称 | string | 非空 |  |
| goodsDesc | 商品描述 | string | 非空 |  |
| orderAmount | 商品价格 | int | 非空 | 以“分”为单位 |
| extension | 透传数据 | string | 非空 | 透传发送到游戏服务器 |
| notifyURL | 支付付款通知地址 | string | 非空 | 支付成功通知游戏服务器 |

#### 3.注册回调

1. **设置游戏音量变化消息回调**

函数：

GameSDK.setOnAudioCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onAudio回调函数 | function | 非空 | 详见onAudio函数说明 |

onAudio函数说明：

function onAudio( param )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| param | 结束通知参数 | object | 非空 | Json对象，见下文 |

Param Json对象:

{

enable, // int 是否开启 0、关，1、开

volume // int 音量 1 - 100

}

1. **设置游戏暂停回调**

函数：

GameSDK.setOnPauseCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onPause回调函数 | function | 非空 | 详见onPause函数说明 |

onPause函数说明：

function onPause()

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| 无 |  |  |  |  |

1. **设置游戏继续回调**

函数：

GameSDK.setOnResumeCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onResume回调函数 | function | 非空 | 详见onResume函数说明 |

onResume函数说明：

function onResume()

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| 无 |  |  |  |  |

1. **设置支付返回回调**

函数：

GameSDK.setOnPayCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onPay回调函数 | function | 非空 | 详见onPay函数说明 |

onPay函数说明：

function onPay( param )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| param | 结束通知参数 | object | 非空 | Json对象，见下文 |

Param Json对象:

{

result, // int 支付结果 0、成功，非0、错误号

message // string 描述

}

1. **设置初始化结果回调**

函数：

GameSDK.setOnInitCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onInit回调函数 | function | 非空 | 详见onInit函数说明 |

onInit函数说明：

function onInit( param )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| param | init返回 | object | 非空 | Json对象，见下文 |

Param Json对象:

{

error**,** // int 错误代码：0、成功（见文末）

userId**,** // int当前玩家id

nickName**,** // string当前玩家名称

headUrl**,** // string 当前玩家头像地址

location**,** //string当前玩家地址

sex**,** // int当前玩家性别

age // int 当前玩家年龄

## 对战游戏

### 客户端接入

#### 导入资源

游戏客户端的启动文件为：index.html文件。

游戏客户端需要在游戏中引用GameSDK\_Cocos.js文件。

如：在index.html中的开始位置加入：

<script type = "text/javascript" src="./GameSDK\_Cocos.js" ></script>

#### 2.调用接口

1. **初始化**

调用平台任何其他的函数前，都要先调用该函数请求初始化SDK。初始化结果将会通过onInit消息返回给游戏。

函数：

GameSDK.init( version, gameId, gameKey, gameSecret )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| version | 接口版本号 | int | 非空 | 当前版本为：1 |
| gameId | 游戏Id | int | 非空 | 后台游戏资料中获得 |
| gameKey | 游戏Key | string | 非空 | 后台游戏资料中获得 |
| gameSecret | 游戏签名 | string | 非空 | 后台游戏资料中获得 |

1. **游戏退出**

调用该函数后平台会立即结束游戏并关闭游戏窗口。

函数：

GameSDK.quit( reason )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| reason | 退出原因 | int | 非空 | 1、正常退出；2、异常退出 |

1. **设置屏幕朝向**

调用该函数可以设置游戏时的屏幕显示朝向。

函数：

GameSDK.setOrientation( orientation )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| orientation | 朝向 | int | 非空 | 1. 横屏； 1、竖屏 |

1. **设置声音**

调用该函数可以设置游戏时声音是否开启以及音量大小。

函数：

GameSDK.setAudio( enable, volume )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| enable | 是否开启 | int | 非空 | 0、关闭；1、开启 |
| volume | 声量 | int | 非空 | 1~100 |

1. **设置游戏加载进度（SDK版本 >= 2）**

SDK版本2开始，平台增加了统一的游戏加载进度界面，用于游戏后台加载时显示。

游戏需要在初始化后，通过此函数报告游戏加载进度。加载界面将显示“加载中...”

当进度>=100%时，如果是对战类游戏，加载界面将提醒用户“等待对手进入中...”，并且游戏需要在对手都进入房间后，调用hideLoadProgress函数关闭加载界面。

当进度>=100%时，如果是非对战类游戏，加载界面将提醒用户“加载完成”，游戏需调用hideLoadProgress函数关闭加载界面。

函数：

GameSDK. setLoadProgress (progress)

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| progress | 加载进度 | int | 非空 | 1-100 |

1. **隐藏游戏加载进度（SDK版本 >= 2）**

用于关闭加载进度界面。此后玩家才可以和游戏交互。

函数：

GameSDK. hideLoadProgress ()

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| 无 |  |  |  |  |

1. **支付**

函数：

GameSDK.pay(orderId, goodsName, goodsDesc, orderAmount, extension, notifyURL)

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| orderId | 订单号 | string | 非空 |  |
| goodsName | 商品名称 | string | 非空 |  |
| goodsDesc | 商品描述 | string | 非空 |  |
| orderAmount | 商品价格 | int | 非空 | 以“分”为单位 |
| extension | 透传数据 | string | 非空 | 透传发送到游戏服务器 |
| notifyURL | 支付付款通知地址 | string | 非空 | 支付成功通知游戏服务器 |

1. **查询房间信息**

调用该函数请求对战房间以及对战双方的信息，结果将通过onRoomInfo消息返回给游戏。

函数：

GameSDK.getRoomInfo()

参数说明：

无参数

1. **游戏终止**

调用该函数后平台会立即结束游戏、关闭游戏窗口，如果游戏为对战类游戏则弹出游戏结算界面。

函数：

GameSDK.finish( result )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| result | 游戏结果 | int | 非空 | 1、胜；2、负；3、平 |

1. **游戏准备（用于支持不含自有服务器游戏）**

调用该函数报告玩家已经准备好开始游戏，游戏中的其他玩家将通过onReady消息获得通知，并且附带着玩家调用函数时的用户数据。只有当所有游戏玩家都准备好之后，游戏才会收到onStart消息，开始游戏。

函数：

GameSDK.ready( userData )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| userData | 用户数据 | string | 非空 | 准备时附带数据，在OnReady消息中传给对手（见后文） |

1. **游戏消息广播（用于支持不含自有服务器游戏）**

调用该函数将游戏消息广播给游戏房间中的所有玩家，或者排除自己的所有玩家。对于对战游戏来说，就是把消息发送给对手。玩家会通过onMessage消息获得玩家发送的消息。

函数：

GameSDK.broadcast( message )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| message | 广播消息 | string | 非空 | 广播的消息内容 |
| includeMe | 是否也广播给自己 | int | 可空 | 0、不包含，1、包含 |

1. **游戏结束（用于支持不含自有服务器游戏）**

调用该函数报告游戏结束，和游戏的结果，因为每个游戏客户端都可能调用该函数，所以平台以第一个获得的报告为准。所有玩家都会收到onFinish消息获得游戏结束的通知和结果。

函数：

GameSDK.gameOver( result )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| result | 游戏结果 | int | 非空 | 1、胜；2、负；3、平 |

1. **设置麦克风**

调用该函数可以设置游戏时麦克风是否开启以及音量大小。

函数：

GameSDK.setMic( enable, volume )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| enable | 是否开启 | int | 非空 | 0、关闭；1、开启 |
| volume | 声量 | int | 非空 | 1~100 |

#### 3.注册回调

1. **设置游戏音量变化消息回调**

函数：

GameSDK.setOnAudioCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onAudio回调函数 | function | 非空 | 详见onAudio函数说明 |

onAudio函数说明：

function onAudio( param )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| param | 结束通知参数 | object | 非空 | Json对象，见下文 |

Param Json对象:

{

enable, // int 是否开启 0、关，1、开

volume // int 音量 1 - 100

}

1. **设置游戏暂停回调**

函数：

GameSDK.setOnPauseCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onPause回调函数 | function | 非空 | 详见onPause函数说明 |

onPause函数说明：

function onPause()

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| 无 |  |  |  |  |

1. **设置游戏继续回调**

函数：

GameSDK.setOnResumeCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onResume回调函数 | function | 非空 | 详见onResume函数说明 |

onResume函数说明：

function onResume()

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| 无 |  |  |  |  |

1. **设置支付返回回调**

函数：

GameSDK.setOnPayCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onPay回调函数 | function | 非空 | 详见onPay函数说明 |

onPay函数说明：

function onPay( param )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| param | 结束通知参数 | object | 非空 | Json对象，见下文 |

Param Json对象:

{

result, // int 支付结果 0、成功，非0、错误号

message // string 描述

}

1. **设置初始化结果回调**

函数：

GameSDK.setOnInitCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onInit回调函数 | function | 非空 | 详见onInit函数说明 |

onInit函数说明：

function onInit( param )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| param | init返回 | object | 非空 | Json对象，见下文 |

Param Json对象:

{

error**,** // int 错误代码：0、成功（见文末）

}

1. **设置查询房间结果回调**

函数：

GameSDK.setOnRoomInfoCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onRoomInfo回调函数 | function | 非空 | 详见onRoomInfo函数说明 |

onRoomInfo函数说明：

function onRoomInfo( param )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| param | getRoomInfo返回 | object | 非空 | Json对象，见下文 |

Param Json对象:

{

error**,** // int 错误代码：0、成功（见文末）

gameId**,** // int 游戏id

roomId**,** // int 房间id

roomKey**,** // int 房间key

users**:** // 房间内玩家数组，第一个玩家总是自己

[{

type**,** // int 玩家类型：1、玩家，2、机器人

userId**,** // int 玩家id

nickName**,** // string 玩家昵称

headUrl**,** // string 玩家头像

sex // string 玩家性别："f" - 女；"m" - 男；"x" – 未知

}]

}

1. **设置游戏准备回调**

函数：

GameSDK.setOnReadyCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onReady回调函数 | function | 非空 | 详见onReady函数说明 |

onReady函数说明：

function onReady( param )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| param | 准备通知参数 | object | 非空 | Json对象，见下文 |

Param Json对象:

{

userId**,** // int 准备的玩家id

userData**,** // string 玩家准备时附带的数据

}

1. **设置游戏开始回调**

函数：

GameSDK.setOnStartCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onStart回调函数 | function | 非空 | 详见onStart函数说明 |

onStart函数说明：

function onStart()

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| 无 |  |  |  |  |

1. **设置游戏接收消息回调**

函数：

GameSDK.setOnMessageCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onMessage回调函数 | function | 非空 | 详见onMessage函数说明 |

onMessage函数说明：

function onMessage( param )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| param | 消息通知参数 | object | 非空 | Json对象，见下文 |

Param Json对象:

{

userId**,** // int 发送消息的玩家id

message**,** // string 消息内容

}

1. **设置游戏结束消息回调**

函数：

GameSDK.setOnFinishCB( func )

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| func | onFinish回调函数 | function | 非空 | 详见onFinish函数说明 |

onFinish函数说明：

function onFinish( param )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| param | 结束通知参数 | object | 非空 | Json对象，见下文 |

Param Json对象:

{

result // int 游戏结果 1、胜，2、负，3、平

}

### 服务器接口

#### 游戏开始接口

##### 接口说明

游戏通知平台服务器游戏对战开始，并透传相关参数。

##### 访问地址

线上环境：http:// www.jeekegame.com:18080/jeekegame/battle/gameapi/gameStart

测试环境：http:// www.jeekegame.com:18080/jeekegame/battle/gameapi/gameStart

##### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| version | 接口版本号 | 整数 | 非空 | 调用接口版本号 |
| gameId | 游戏Id | 整数 | 非空 | 游戏Id |
| roomId | 房间Id | 整数 | 非空 | 房间Id |
| roomKey | 房间key | 整数 | 非空 | 房间key |
| userIdList | 玩家ID列表 | 整数列表 | 非空 | 玩家ID列表 |
| timeStamp | 时间戳，毫秒 | 字符串 | 非空 | 时间戳，毫秒 |
| sign | 签名 | 字符串 | 非空 | 签名，生成方式参考文末 |
| extension | 扩展字段 | 字符串 | 可空 | 扩展字段 |

##### 返回结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| errorCode | 错误码 | 整数 | 非空 | 参考文末 |
| errorMsg | 错误描述 | 字符串 | 非空 | 结果描述 |
| extension | 透传扩展字段 | 字符串 | 可空 | 透传扩展字段 |

#### 游戏结束接口

##### 接口说明

游戏通知平台服务器游戏对战结束，并透传相关参数。

##### 访问地址

线上环境：http:// www.jeekegame.com:18080/jeekegame/battle/gameapi/gameEnd

测试环境：http:// www.jeekegame.com:18080/jeekegame/battle/gameapi/gameEnd

##### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| version | 接口版本号 | 整数 | 非空 | 调用接口版本号 |
| gameId | 游戏Id | 整数 | 非空 | 游戏Id |
| roomId | 对战房间Id | 整数 | 非空 | 对战房间Id |
| roomKey | 房间key | 整数 | 非空 | 房间key |
| resultList | 对战结果resultList列表 | 对象列表 | 非空 | 见下文 |
| timeStamp | 时间戳，毫秒 | 字符串 | 非空 | 时间戳，毫秒 |
| sign | 签名 | 字符串 | 非空 | 签名，生成方式参考文末 |
| extension | 扩展字段 | 字符串 | 可空 | 扩展字段 |

##### 返回结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **含义** | **类型** | **是否为空** | **备注** |
| errorCode | 错误码 | 整数 | 非空 | 参考文末 |
| errorMsg | 错误描述 | 字符串 | 非空 | 结果描述 |
| extension | 透传扩展字段 | 字符串 | 可空 | 透传扩展字段 |

#### 签名规则

##### 1、构成原串

a. 构造源串所需参数

gameId( 游戏Id )

gameKey( 游戏鉴权key )

gameSecret( 游戏鉴权secret )

roomId( 房间ID )

roomKey( 房间Key )

timeStamp( 时间戳 )

b. 将参数按照字典序排序, 使用URL键值对的格式（即key1=value1&key2=value2…）拼接成字符串。

注意:参数名区分大小写

##### 2、生成Sign值

使用 md5 算法对构造的源串进行加密,将得到的字符串所有字符转换为大写示例：

gameId=11&gameKey=11&gameSecret=22&roomId=22&roomKey=33&timeStamp=15281

## 支付服务端接口

#### 订单查询接口

##### 接口说明

商户根据订单的订单号等信息，从我司支付服务器查询指定订单的支付情况。

##### 访问地址

<http://IP:PORT/jeekegame/api>/gameQueryOrder

##### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **含义** | **长度** | **类型** | **是否为空** | **备注** |
| version | 接口版本号 | 10 | **整数** | **非空** | 方便接口升级，当前为：1.0.0 |
| gameId | 游戏Id | 64 | **整数** | **非空** | 游戏Id |
| cpOrderId | 商户订单号 | 64 | **字符串** | **非空** | 商户订单号 |
| sign | 签名 | 10 | **字符串** | **非空** | 签名，生成方式参考文末 |
| extension | 扩展字段 |  | **字符串** | **可空** | 扩展字段 |

**注意:http post请求时传递参数中文要进行url编码**

##### 返回参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **含义** | **长度** | **类型** | **是否为空** | **备注** |
| errorCode | 错误码,参考文末 | 10 | **整数** | **非空** |  |
| errorMsg | 错误描述 | 20 | **字符串** | **非空** | 结果描述 |
| extension | 透传扩展字段 | 20 | **字符串** | **可空** | 透传扩展字段 |
| cpOrderId | 商户自定义订单号 | 64 | **字符串** | **非空** | 商户自定义，最长 64 位字母、数字和下划线组成,**商户订单号必须唯一** |
| orderAmount | 交易金额 | 100 | **整数** | **非空** | 交易金额，单位（**分**） |
| orderId | 我司订单号 | 64 | **字符串** | **非空** | 我司订单号，唯一性 |
| tradeResult | 支付结果 | 10 | **整数** | **非空** | 支付结果，0：未知；1：成功；2：失败 |
| userId | 我司生成的userId | 20 | **长整数** | **非空** | 我司生成的userId，唯一性 |
| payTime | 交易时间 | 30 | **字符串** | **非空** | 交易时间。格式：yyyyMMddHHmmss |
| sign | 签名字符串 | 30 | **字符串** | **非空** | 对关键信息签名后得到的字符串1，用于商户验签签名规则请参考签名生成规则 |

#### 订单异步通知接口

##### 接口说明

支付完成后，我司支付服务器会把交易结果通知到下单时传递的url：notifyUrl，建议商户根据异步通知结果修改订单交易状态。

注意：因为网络或者其它原因，可能会有重复通知的情况，CP需要做好去重处理，避免多次发货

##### 访问地址

接口地址：支付结果通知地址，在下单接口中通过参数notifyUrl传递。

请求方式：POST请求

##### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **含义** | **长度** | **类型** | **是否为空** | **备注** |
| errorCode | 错误码,参考文末 | 10 | **整数** | **非空** |  |
| errorMsg | 错误描述 | 20 | **字符串** | **非空** | 结果描述 |
| gameId | 游戏Id | 10 | **整数** | **非空** | 游戏Id |
| cpOrderId | 商户自定义订单号 | 64 | **字符串** | **非空** | 商户自定义，最长 64 位字母、数字和下划线组成,**商户订单号必须唯一** |
| orderAmount | 交易金额 | 100 | **整数** | **非空** | 交易金额，单位（**分**） |
| orderId | 我司订单号 | 64 | **字符串** | **非空** | 我司订单号，唯一性 |
| tradeResult | 支付结果 | 10 | **整数** | **非空** | 支付结果，0：未知；1：成功；2：失败 |
| payTime | 交易时间 | 30 | **字符串** | **非空** | 交易时间。格式：yyyyMMddHHmmss |
| userId | 我司生成的userId | 20 | **长整数** | **非空** | 我司生成的userId，唯一性 |
| sign | 签名字符串 | 30 | **字符串** | **非空** | 对关键信息签名后得到的字符串1，用于商户验签签名规则请参考签名生成规则 |
| extension | 透传扩展字段 | 20 | **字符串** | **可空** | 透传扩展字段 |

##### 返回结果

返回格式：字符串

我司支付服务器端通过notifyUrl通知CP系统，CP系统处理完相关的业务逻辑后，需要以字符串的形式反馈处理结果，内容如下

success：处理成功，我司系统收到此结果后不再进行后续通知；

fail：处理不成功，我司系统收到此结果或者没有收到任何结果，系统通过补单机制再次通知；

补单机制：该异步通知，最多会发送5次，直到我司支付服务器收到CP服务器返回success为止。如果5次我司服务器都没收到，则不再发送通知，商户后续需要请求订单查询接口来修正订单支付结果

#### 签名生成规则

##### 构成原串

将参数按照字典序排序,使用URL键值对的格式（即key1=value1&key2=value2…）拼接成字符串,在末尾加上& to\_lower\_case(md5(gameSecret))。

注意:

参数名区分大小写

示例

##### 生成sign值

使用 md5 算法对构造的源串进行加密,将得到的字符串所有字符转换为大写

## 服务端错误码对照表

|  |  |
| --- | --- |
| **错误码** | **错误码说明** |
| 0 | 成功 |
| 10001 | 参数错误 |
| 10002 | 签名错误 |
| 10003 | 系统错误 |
| 10008 | 游戏错误 |
| 10009 | 订单错误 |

## 自测工具

为了方便游戏开发者测试游戏与平台的对接，平台提供游戏测试工具App（[点我下载](http://www.cocos.com/)）。测试工具可以拉起游戏并提供游戏匹配功能。游戏开发者可以在多个手机上安装测试工具，拉起游戏，进行匹配，测试游戏。

测试工具可以通过两个方式拉起游戏：

1. 通过URL拉起游戏
2. 通过本地游戏包拉起游戏

测试工具通过读取配置文件，确定拉起的游戏，以及游戏拉起方式。

# 打包说明

游戏客户端需要打包并且签名才能在平台审核及运行。

下面为客户端打包签名流程：

1. 将客户端压缩成zip文件，index.html需要在压缩包的根目录。
2. 使用jarsigner工具进行签名（见下文）。
3. 将签名后的包重命名为以.cpk为后缀的文件名。

**签名流程：**

1. 生成keystore文件，如已有，则跳过该步骤。例如:生成名为test.keystore，alias为test的keystore文件：

keytool -genkey -keystore test.keystore -alias test -keyalg RSA -keysize 2048 -sigalg MD5withRSA -validity 99999

1. 使用jarsigner工具签名。例如：使用名为test.keystore，alias为test的keystore文件对Demo.zip进行签名，签名文件保存为Signed.cpk：

jarsigner -verbose -keystore test.keystore -signedjar Signed.cpk Demo.zip -digestalg SHA1 -sigalg MD5withRSA test

# 资源下载

### SDK下载

### 设计资源下载