

游戏猫SDK接入

- [开始的准备工作](#)
- [引入游戏猫sdk库](#)
- [配置AndroidManifest.xml](#)
- [在代码中接入SDK](#)
- [环境切换](#)
- [账号登陆](#)
- [支付订单](#)
- [用户中心](#)
- [用户退出登录](#)

开始的准备工作

引入游戏猫sdk库

//Eclipse 引入：

1.引入gamecatsdk库：File -> Import -> Android -> Existing Android Code Into WorkspaceNext -> Browser -> ->选择gamecatsdk工程目录，finish即可；

2.依赖gamecatsdk库： 选择需要依赖gamecatsdk库的项目 -> 右键单击 -> Properties -> Android -> 下方的Library栏，单击Add，添加之前gamecatsdk库项目即可。

3.请把linkeaarug.data复制到assets目录下(联调和测试的文件不同，请记住在上线前把秘钥替换掉)（打开我们提供的demo可直接open指向GameCatSDKDemo，即可关联导入进行编译）

配置AndroidManifest.xml

配置权限

```

<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.CHANGE_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
<uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW" />
<uses-permission android:name="android.permission.DOWNLOAD_WITHOUT_NOTIFICATION" />

<uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS"/>
    <uses-permission android:name="android.permission.READ_SETTINGS"/>
<uses-permission android:name="android.permission.WRITE_SETTINGS"/>
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>

```

配置Activity

<!--游戏的activity声明-->

```

<activity
    android:name="com.gamecat.pay.action.activity.GameCatActivity"
    android:configChanges="orientation|keyboardHidden|navigation"
    android:exported="false"
    android:theme="@android:style/Theme.Translucent"
    android:screenOrientation="behind" >
</activity>

<activity
    android:name="com.gamecat.sdk.PayActivity"
    android:configChanges="orientation|keyboardHidden|navigation"
    android:exported="false"
    android:theme="@android:style/Theme.Translucent"
    android:screenOrientation="behind" >
</activity>

<activity
    android:name="com.alipay.sdk.app.H5PayActivity"
    android:configChanges="orientation|keyboardHidden|navigation"
    android:exported="false"
    android:screenOrientation="behind" >
</activity>

<activity
    android:name="com.alipay.sdk.auth.AuthActivity"
    android:configChanges="orientation|keyboardHidden|navigation"

```

```
        android:exported="false"
        android:screenOrientation="behind" >
</activity>
```

配置平台的GameCatAppKey（应用ID）和GameCatChlId（渠道号）（我们会提供）

```
<meta-data
    android:name="GameCatAppKey"
    android:value="我们提供" />
```

```
<meta-data
    android:name="GameCatChlId"
    android:value="我们提供" />
```

配置平台的数据统计（我们会提供）

```
<meta-data
    android:name="com.alibaba.app.appkey"
    android:value="我们提供" />

<meta-data
    android:name="com.alibaba.app.appsecret"
    android:value="我们提供" />
<meta-data
```

在代码中接入SDK

环境切换

方法：GameCatSDK.setEnvironment(Activity activity,int environment,String aesKey) 初始化时必须调用方法

参数	类型	说明	备注
activity	Activity	游戏的活动中的activity	
environment	int	需要设置的环境	（0代表联调环境，1代表正式环境）
aesKey	String	数据传输加密的密钥	需要我们提供

例子

GameCatSDK.setEnvironment(activity,0,"ljh2OU4XH9Ydq7ap4x1Pow==");

账号登陆

需要登陆的时候调用这个方法（参数全部必填）

方法：GameCatSDK.Login(Activity activity ,String gameId ,final GameCatSDKListener listener);

参数	类型	说明	备注
activity	Activity	游戏的活动中的activity	
gameId	String	游戏的id	我们平台分配的游戏id
listener	GameCatSDKListener	监听	成功会回调

返回值

登录成功返回值	说明	备注
openId	成功时的用户的唯一标识	
token	鉴别用户是否成功登陆的密钥	
userName	用户的昵称	

登录失败返回值	说明	备注
string	失败时返回错误原因	

例子

```
GameCatSDK.Login(activity,gameId,new GameCatSDKListener() {
    @Override
    public void onSuccess(JSONObject message) {
        try {
            //取得用户的唯一标识进行游戏的登陆
            String openId= message.getString("openId");
            //秘钥
            String token=message.getString("token");
            //用户的昵称
            String userName=message.getString("userName");

        } catch (JSONException e) {
            e.printStackTrace();
        }
        listener.setText(message+"");
    }

    @Override
    public void onFail(String message) {
        listener.setText(message+"");
    }
});
```

支付订单

方法：支付的时候调用这个方法（参数全部必填）

```
GameCatSDK.Order(Activity activity,double amount, String cpDiscript, String codeNo
, String notifyUrl,String extend,final GameCatSDKListener listener)
```

参数	类型	说明	备注
activity	Activity	当前活动的activity	
amount	double	金额	最高限额为99万，最低为1，必须为整数。（测试时可输入0.1,如果在支付环节使用喵点支付，可以提供手机号让我们充值）
cpDiscript	String	产品介绍	
codeNo	String	订单编号	用于对账跟发货凭证（必须唯一）
notifyUrl	String	发货回调地址	通知游戏端发货的地址
extend	String	游戏方的透传参数	传进什么我们都会按原数据返回，（建议用jsonString）
listener	GameCatSDKListener	订单回调监听	成功失败都会回调codeNo，成功支付会服务器通知

注意：游戏和我们平台的对单是通过codeNo，所以请确保订单号唯一性。

例子

```
GameCatSDK.Order(activity,0.1, "倚天剑", "123456", "http://                                mock.yo
uximao.cn/mockjsdata/11/sdk/notify","这是1区张无忌的订单",new GameCatSDKListener() {
    @Override
    public void onSuccess(JSONObject message) {
        json形式回调codeNo
    }

    @Override
    public void onFail(String message) {
        jsonString形式回调codeNo
    }
});
```

用户中心

用户中心的时候调用这个方法（参数全部必填）

```
GameCatSDK.UserCenter(Activity activity,final GameCatSDKListener listener);
```

参数	类型	说明	备注
activity	Activity	游戏的活动中的activity	
listener	GameCatSDKListener	监听	成功会回调

例子

```
GameCatSDK.UserCenter(activity,new GameCatSDKListener() {  
  
    @Override  
    public void onSuccess(JSONObject message) {  
  
    }  
  
    @Override  
    public void onFail(String message) {  
  
    }  
});
```

用户退出登录

用户因在操作过程中token失效或被系统拉黑名单操作回调，此方法游戏需要作退出到登录界面（参数全部必填，建议在初始化后就作出监听）

```
GameCatSDK.sdkCancelListener(final GameCatSDKListener listener);
```

参数	说明	备注	
listener	GameCatSDKListener	监听	成功会回调

例子

```
GameCatSDK.sdkCancelListener(new GameCatSDKListener() {  
    @Override  
    public void onSuccess(final JSONObject message) {  
  
        //请在这里做退出到登录界面的操作  
    }  
  
    @Override  
    public void onFail(String message) {  
  
    }  
});
```

更多数据请看demo或咨询技术支持