

用户中心 SDK 文档

前言

本文档是安智单机游戏 SDK 的接入文档，面向专业的 Android 技术人员。开发者在接入前必须阅读此文档。

文档规则如下：

- 1、必接项目在标题前有说明；
- 2、蓝色字体为与上一版有改动内容；
- 3、红色字体为强调部分；

目录

一、 名词解释.....	1
二、 接入前准备.....	1
三、 安智 SDK 开发环境搭建.....	1
3.1 客户端工程框架搭建.....	1
四、 SDK 客户端接入模块.....	2
4.1 初始化模块（必接）	2
4.1.1 重要方法定义.....	2
4.1.2 参数定义.....	3
4.1.3 代码详解.....	3
4.2 账号登录模块.....	4
4.2.1 重要方法.....	4
4.2.2 重要参数.....	4
4.2.3 代码详解.....	4
4.3 通知接口（必接）	5
4.3.1 重要接口.....	5
4.3.2 重要的方法和参数.....	5
4.3.3 代码详解.....	7
2.4 悬浮球接口（必接）	8
4.4.1 重要方法.....	8
2.5 上传游戏信息（必接）	8
2.5 上传游戏得分接口(如果在 cpinfo 设置了接入排行榜，则必接).....	9
2.5 退出广告位（必接）	9
5.1 初始化调试 log.....	9
六、 混淆配置.....	10

一、 名词解释

Appkey: 接入安智 SDK 所需要的参数之一, 由安智商务提供;

AppSecret: 接入安智 SDK 的参数之一, 由安智商务提供;

SDK: 安智单机游戏 SDK, 本文简称 SDK;

注: 在接入前请确保参数的有效性, 参数未生效同安智商务协调解决。

二、 接入前准备

开发者在接入 SDK 前需要做以下准备:

1、获得可用的 SDK 参数包括 Appkey 和 AppSecret, 并将获得的 SDK 参数放入 demo 进行测试, 测试账号功能和上传得分功能和排行榜功能。

2、获得 SDK 开发包, SDK 开发包含以下内容;

a、客户端、服务端 demo 和说明文档

b、自测用例

c、所需其他的资源包括: 角标、闪屏资源;

注: 角标, 闪屏资源 PSD 原图如下:

<http://bbs.zhiyoo.com/thread-9022938-1-1.html>

开发者在提测前, 需对照安智提供的测试用例。测试用例提供的每一个测试点都必须覆盖。

提测之后, 我们 QA 会对游戏包做验收, 并给开发者提供测试报告, 如果测试未通过, 开发者根据测试报告修复自身 bug。

三、 安智 SDK 开发环境搭建

3.1 客户端工程框架搭建

SDK 接入的开发环境以 win 系统下 eclipse 环境为范例为大家讲解。包括所需引入的工程和资源等。

1、工程 lib: 安智游戏 SDK 工程 lib 名称为 anzhi.user.com, 接入可以直接应用该工程。

2、assets : assets 目录是 Android 开发工程中的一个重要目录, 该目录中的文件必须放于主工程下否则不能打入 apk 包中, 厂商需要负责 demo 中的 assets 目录下的文件至游戏工程中。

3、AndroidManifest.xml: AndroidManifest 文件是 Android 配置的核心文件，配置出错会影响 Activity 生命周期方法，从而导致程序异常。

四、 SDK 客户端接入模块

安智用户中心的客户端接入模块主要包括初始化模块、通知模块、账号模块、悬浮球模块、支付模块、退出广告位模块、上传游戏信息。完整的用户中心接入五个模块缺一不可。

(1)、在初始化模块中，开发者需要设置一些基本的厂商信息和 SDK 基本配置参数，主要包括 appkey 和 appSecret，SDK 页面的横竖屏等，详见：4.1

(2)、通知模块，用户 SDK 和游戏客户端数据的通知，包括登入、登出和页面回退，详见 4.2。

(3)、账号登录模块，安智用户中心的账号登录包括安智账号登录和官方账号登录（游戏方原有的账号）。

(4)、上传得分信息。

(5)、悬浮窗口模块。

(6)、上传游戏信息。

安智用户中心主要对外方法都需要使用 AnzhiUserCenter 的实例去调用，AnzhiUserCenter 使用单例实现，全局只有一个。

4.1 初始化模块（必接）

SDK 的初始化模块需要设置 SDK 基本参数包括 AppKey 和 AppSecret, 以及一些其他的参数信息。

注：初始化代码必须先于 SDK 其他的任何模块执行。

4.1.1 重要方法定义

<pre>void azinitSDK(Activity activity,CPInfo info,InitSDKCallback callback, AzOutGameInter azInter)</pre>	<p>SDK 初始化方法，此方法用于初始化 SDK，第二个参数是用户信息，第三个参数是初始化成功的回调接口。（可以接口中直接实现登录方法）</p> <p>注增加本方法增加一个参数用于实现提出广告。</p>
<pre>void setActivityOrientation(int i)</pre>	<p>SDK 界面横竖屏设置 API，此方法用于设置 SDK 的界面的横竖屏。SDK 的界面要同游戏保持一致。参数 i 为 0 横屏,1 竖屏,4 根据物理感应来选择方向。</p>

4.1.2 参数定义

名称	解释
AppKey	应用的唯一标示，用于标示应用。
AppSecret	与 AppKey 配套使用
AnzhiUserCenter	下文代码中 mAnzhiCenter 实例的类，用户中心的基本操作类。单例实现全局只有一个，用户中心接入的所有操作都通过它的单例实例来实现。
CPIInfo	厂商信息类，保存 app 应用接入厂商的基本信息，包括 AppKey 和 AppSecret 等。

4.1.3 代码详解

开发者在初始化 SDK 前需要实例化 CPIInfo 类和 AnzhiUserCenter 类。在 CPIInfo 类中，厂商设置 Appkey 和 AppSecret 和一些其他的参数，最后将 CPIInfo 的实例 set 到 AnzhiUserCenter 的实例中去。

另外厂商还要通过 setActivityOrientation (int i) 方法来设置 SDK 的横竖屏方向，具体代码如下。

```
final CPIInfo info = new CPIInfo();
info.setAppKey(Appkey); // 设置Appkey
info.setSecret(AppSecret); // 设置AppSecret
info.setChannel("AnZhi"); // 传固定值"AnZhi"
info.setGameName(getResources().getString(R.string.app_name));
info.setGameType(CPIInfo.STANDALONE_GAME); // 设置游戏类型为单机游戏
info.setRank(true); // 设置是否接入排行榜
info.setBindAccount(true); // 如果有官方账号设置是否绑定安智账号
mAnzhiCenter = AnzhiUserCenter.getInstance();
mAnzhiCenter.azinitSDK(this, info, this); // 初始化方法
mAnzhiCenter.setKeybackCall(this); // 设置返回监听按钮
mAnzhiCenter.setCallback(mCallback); // 设置 登录、登出、接口
mAnzhiCenter.setUploadScoreCallback(this); // 设置上传分数接口
mAnzhiCenter.setActivityOrientation(1); // 3.1.1 新增 API 用于设置 SDK 的横竖屏 // 0 横屏, 1 竖屏, 4 根据物理感应来选择方向
注：支付页面默认竖屏，论坛页面默认根据物理感应来选择方向，无法修改
```

4.2 账号登录模块

账号登录模块包括包括安智账号模块和游戏老帐户模块(游戏老帐户默认不接)。安智用户中心的整体的逻辑如下：(1) 在完成初始化模块之后；(2) 调用登录的方法 login；(3) 游戏客户端接受通知，获取必要参数；(4) 游戏客户端将参数透传至游戏服务端验证用户；(5) 游戏服务端向安智服务端验证用户。成功验证后返回游戏客户端完成登录。

注：3.3 增加从登录功能，在游戏过程中，安智账号可能会因为某些原因意外下线，玩家会重新登录，厂商需要判断重新登录账号和当前账号是否为同一个，如果不相同需要刷新对应的数据。

4.2.1 重要方法

<code>void login(Context context,boolean b);</code>	登入方法，第一个参数是 Android 上下文，第二个参数是预留参数，直接传 true；调起该方法后 SDK 自动完成安智账号的初始化。起调该方法可以触发登入通知。
<code>void logout(Context context);</code>	登出方法，游戏厂商调起登出方法之后，安智账号会注销，并清楚客户端信息。调用该方法会触发登出通知。（注：在个人中心的切换账号也是调用该方法，所以触发的通知是相同的）
<code>viewUserInfo(Context context);</code>	个人中心方法，用于调起安智用户中心的个人中心方法。

4.2.2 重要参数

有关安智用户中心的重要参数主要有 sid、uid、loginname 三个，都是通过通知来透传给游戏客户端。（其余参数开发可以通过通知内容自行解析）

sid	sid 是安智用户中心的会话 id，用于同服务器的交互。
uid	uid 是安智用户的唯一标示符，每一个安智用户都有一个唯一的 uid，厂商通过 uid 来区分安智用户。
login_name	登录，没有安智的用户可有多个登录名，同一用户的多个登录名对应一个 uid。

4.2.3 代码详解

此处代码比较简单，如有不解可以参考 demo。

4.3 通知接口（必接）

安智 SDK 的通知只要有 3 类，分别是“登录通知”、“登出通知”、“上传得分通知”以及“退出 SDK 界面的通知”（在安智 SDK 界面，点击 back 键或回退键回到游戏界面的通知）。厂商根据自己的需要实现通知的回调接口。

4.3.1 重要接口

Interface AnzhiCallback()	登入、登出、通知所用的接口。不同的通知采用不同的标示码来区分。
Interface KeybackCall()	退出安智 SDK 界面的通知，不同的界面采用不同标示码。
Interface AzOutGameInter()	退出游戏的接口，开发者在该接口的方法中完成退出游戏的操作
Interface UploadScoreCallback()	上传得分信息后的通知，开发者在该接口的方法中完成上传得分的操作

4.3.2 重要的方法和参数

（1）登入、登出、支付回调

void onCallback(CPInfo cpInfo, String result)	AnzhiCallback()接口所实现的方法，result 是 String 形式的 json。sdk 通过 result 给游戏客户端透传厂商所需要的数据。以下是 result 中包含的重要的参数
callback_key	获取方法：json.optString("callback_key"); 通知标示码，用于区分不同的通知： 1、key_login：登入通知 2、key_logout：登出通知
sid	获取方法 json.optString("sid"); 登入通知特有字段，会话 id，详细请看 4.2.2 账号模块参数。
uid	获取方法 json.optString("uid"); 登入通知特有字段，安智账号的唯一标示符，详细请看 2.2.2 账号模块参数
login_name	获取方法 json.optString("login_name"); 登入通知特有字段，安智账号的登录名，详细请看 2.2.2 账号模块参数
code	code 值为 200 是表示登录成功，其他的标示

	失败。
code_desc	code 的具体描述信息。
注：登出回调不需要解析字段	

(2) 退出 SDK 界面通知

void KeybackCall(String st)	KeybackCall() 所需实现的接口方法
st	<p>st 是 SDK 界面的标示符，每一个不同 SDK 界面都对应一个不同的 st:</p> <p>Login 登录界面</p> <p>UserCenter 个人中心界面</p> <p>GameGift 礼包页面</p> <p>anzhiBbs 论坛页面</p> <p>Message 消息页面</p> <p>Feedback 客服中心页面</p> <p>Login 登录页面</p> <p>GameRank 排行榜界面</p>

4.3.3 代码详解

登录、登出

```
AnzhiCallback mCallback = new AnzhiCallback() {
    @Override
    public void onCallback(CPInfo cpInfo, final String result) {
        Log.e("anzhi", "result " + result); // result 为json内容根据具体回调的类型而确定
        try {
            JSONObject json = new JSONObject(result);
            String key = json.optString("callback_key"); // 通过"callback_key"获得回调类型

            // 支付: key_pay
            // 登出: key_logout
            // 登入: key_login

            if ("key_logout".equals(key)) {
                /*
                 * 安智账号退出以后, 所对应的游戏角色不能继续游戏。 前端流程表现为, 游戏必须回到游戏的登录界面或安智的登录界面。
                 */
                Toast.makeText(MainActivity.this, "退出账号",
                    Toast.LENGTH_SHORT).show();
                mControllLayout.setVisibility(View.GONE);
                mBtnLogin.setVisibility(View.VISIBLE);
            } else if ("key_login".equals(key)) {
                int code = json.optInt("code"); // 登入成功标志位
                String desc = json.optString("code_desc"); // code 的详细描述符

                uid = json.getString("uid"); // uid是安智账号的唯一标示, 用于标记安智用户, 开发必须用uid来对应玩家的游戏角色
                sid = json.getString("sid"); // sid是会话id, 用向安智的服务器验证, 用户是否在线

                if (code == 200) {
                    mHandler.sendMessage(LOGIN_SUCCEEDED);
                } else {
                    mHandler.obtainMessage(LOGIN_FAILED, desc)
                        .sendToTarget();
                }
            }
        } catch (Exception e) {
            Log.e(TAG, e + "");
        }
    }
};
```

2.4 悬浮球接口（必接）

安智悬浮球是 SDK 的交互工具，玩家通过悬浮球快速的进入个人中心、论坛、排行榜等界面。

安智的悬浮球模块必须关联一个 Activity，一般为游戏的主 Activity，且必须常驻（可以在某些特殊的界面隐藏比如游戏副本界面）。悬浮球模块一共提供了 3 个方法，分别是销毁、隐藏、显示，供厂商使用。

悬浮球的实现的方法集成在了初始化方法中，无须开发者额外调用。

注：点击悬浮窗口提示“网络连接失败，请重启应用”。遇到该问题，游戏方首先检查手机网络是否畅通。

4.4.1 重要方法

<code>void dismissFloaticon()</code>	隐藏悬浮球方法，调用该接口后可以隐藏悬浮球。 注：必须在主线程调用。
<code>void showFloaticon</code>	显示悬浮接口，在悬浮球为隐藏状态时，调用本方法，可以重新显示悬浮球。 注：本方法主线程调用。
<code>void gameOver(Context context)</code>	<u>悬浮球销毁方法</u> 调用本方法可以完全销毁该activity所绑定的悬浮球对象。
<code>Void removeFloaticon ()</code>	悬浮球销毁方法

2.5 上传游戏信息（必接）

`submitGameInfo(final Context context, final UserGameInfo gameInfo);`

只要上传信息有：

```
UserGameInfo info = new UserGameInfo();
info.setAppName("游戏名称");//游戏名称
info.setGameArea("1");// 游戏的区
info.setGameLevel("10");// 游戏级别
info.setMemo("");// 备注
info.setNickName("未知");//安智昵称
info.setUid("adfdsafdsaf");// 安智 uid
info.setUserRole("aaaa");// 角色（或角色 Id）
```

2.5 上传游戏得分接口(如果在 cpinfo 设置了接入排行榜, 则必接)

注: 若接, 厂商需在每关结束后调用此接口

```
submitGameScore(final Context context,final long score,  
final String descript, final int rule)
```

参数说明:

score: 上传的分数, 用于排行榜计算的唯一指标

descript: 具体分数详情, 用于显示

rule: 分数排名规则, 1代表升序(分数小的排在前面), 0代表降序(分数大的排在前面)

2.5 退出广告位 (必接)

- 1、实现方法在初始化方法中, 传入退出接口;
- 2、游戏退出按钮调用 azoutGame()方法;
- 3、在退出接口的方法中实现退出游戏 azOutGameInter()。

5.1 初始化调试 log

初始化的调试的 log 的主要有：

1、厂商信息初始化信息检查，完成初始化之后，SDK 的调试 LOG，会将开发者的参数打印出来，开发可以通过打印信息来查看参数是否正确。厂商在 set CpInfo 的时可以查看该 log 信息。

2、assets 资源检查：在起调 Login () 方法会起调。SDK 的正常使用需要依赖 assets 目录下面的资源文件，如果安装包没有正确加入资源。会有如下测试信息输出“[查看 assets 目录下资源是否正确添加](#)”；如果没有信息输出说明资源没有错误。

3、在检查 assets 目录的同时，还会检查厂商是否开启游戏老帐号接口。

4、支付调试 LOG，在起调 pay 方法时，测试 log 会输出参数值，并输出启用的支付类型。

六、混淆配置

如果厂商需要对代码进行混淆许需要配置以下代码。

```
-keep class com.anzhi.usercenter.sdk.**{*;}  
=keep class com.unionpay.upomp.tbow.**{*;}  
-keep class **.R$**{*;}
```