**iFox游戏SDK接入文档**

**版本历史：**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 修订历史说明 | 时间 | 作者 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. 简介

iFox游戏SDK,是提供游戏进行嵌入的服务套件。提供给游戏账号系统管理、支付代收系统。便于游戏的运营代收。

1. 准备工作
   1. **接入前的准备**
2. 获取AppKey与AppSecrect，请咨询技术支持人员提供；
   1. **修改AndroidManifest.xml**

修改Manifest下配置文件,把manifest.txt 相应内容拷贝到AndroidManifest.xml,

注意: AndroidManifest application节点,不能添加主题 android:Theme选项!

2.3 接入项目库

把iFoxSDK\Resource 目录下的文件拷贝到,游戏工程目录下

注:其中alipay\_lib是支付宝资源项目库,需要导入当前资源项目,

将alipay\_lib拷贝到Eclipse workspace，通过Eclipse import该工程，并在此工 程的Properties->Android中选中为library工程,

在商户应用工程的Properties->Android中引用资源工程

2.4 调用接口

调用IFox.init()接口初始化

1. API列表说明
   1. **初始化**

|  |  |
| --- | --- |
| **public** **static** **void** init(**final** Activity act, String appKey, String appSecrect) | |
| 说明 |  |
| activity | 游戏的的主Activity |
| appKey | 游戏的在平台中的app key |
| appSecrect | 游戏的在平台中的app secrect |
| InitCallBack | 初始化回调,必须回调onSuccess才能进行其他动作,支付,登录等,如果失败,需重新调用init方法,  **public** **static** **interface** InitCallBack {  /\*\*  \* 初始化成功  \*/  **public** **void** onSuccess();  /\*\*  \* 初始化失败,必须重新调用初始化  \*/  **public** **void** onFail(String msg);  } |

* 1. **显示登录界面**

|  |  |
| --- | --- |
| **public** **static** **void** loginPage(**final** Activity activity,**final** Bundle bundle, **final** LoginListener listener) | |
| 说明 |  |
| activity | 游戏的的主Activity |
| bundle | 需要额外传入的参数，如果没有，留空 |
| listener | 登录结果回调的处理，回调接口是：  **public** **static** **interface** LoginListener{  /\*\*  \* 登录成功的回调  \*  \* **@param** bundle  \* 回调参数  \*/  **public** **void** onSuccess(Bundle bundle);  /\*\*  \* 登录过程取消  \*/  **public** **void** onCancel();    }  onSuccess -> Bundler 回调参数会回调登录后的token,可用以下方法读取, key : token  String token = bundle.getString(KeyConstants.INTENT\_DATA\_KEY\_TOKEN);  String uid = bundle.getString(KeyConstants.INTENT\_DATA\_KEY\_UID); |
|  |  |

* 1. **显示支付界面**

|  |  |
| --- | --- |
| **public** **static** **void** chargePage(**final** Activity activity,**final** Bundle bundle, **final** ChargeListener listener) | |
| 说明 |  |
| activity | 游戏的的主Activity |
| bundle | 需要额外传入的参数，  道具号(int类型):购买的道具号码, (必传 – 不能为0)  附加信息(String类型):需要保存到服务器的订单附加信息  固定支付金额 (float 类型) 需要固定支付的金额  bundle.putString(KeyConstants.INTENT\_DATA\_KEY\_EXTRA, "需要保存的字符串");  bundle.putInt(KeyConstants.INTENT\_DATA\_KEY\_PID, 333);  bundle.putFloat(KeyConstants.INTENT\_DATA\_KEY\_AMOUNT, 10F);  附加信息key: extra  道具号key: pid  固定支付金额: key:amount |
| listener | 支付结果回调的处理，回调接口是：  **public** **static** **interface** ChargeListener{  /\*\*  \* 支付成功的回调  \*  \* **@param** bundle  \* 回调参数  \*/  **public** **void** onSuccess(Bundle bundle);    /\*\*  \* 支付失败  \*/  **public** **void** onFail();    /\*\*  \* 支付完成  \*/  **public** **void** onFinish();  /\*\*  \* 支付过程取消  \*/  **public** **void** onCancel();    }  //商品id  int pid = bundle.getInt(KeyConstants.INTENT\_DATA\_KEY\_PID);  //订单号  String orderId=bundle.getString(KeyConstants.INTENT\_DATA\_KEY\_ORDERID);  //支付的价钱  float amount = bundle.getFloat(KeyConstants.INTENT\_DATA\_KEY\_AMOUNT);  当支付成功后bundle会回调以上信息  道具号 key : pid  订单号 key: ordered  价钱 key: amount |
|  |  |

1. 显示排行榜页面

|  |  |
| --- | --- |
| **public** **static** **void** leaderboardPage (**final** Activity act) | |
| 说明 |  |
| activity | 游戏的的主Activity |

1. 提交分数

|  |  |
| --- | --- |
| **public** **static** **void** submitScore(**final** Activity activity, **final** **long** score, **final** **int** lid, **final** OnCommitScoreCallBack cb) { | |
| 说明 |  |
| activity | 游戏的的主Activity |
| score | 需要提交的分数 |
| lid | 排行榜id, |
| cb | 提交分数回调的处理，回调接口是：  **public** **interface** OnCommitScoreCallBack {  **public** **void** onSuccess();  **public** **void** onFail(String msg);  } |

1. 分享

|  |  |
| --- | --- |
| **public** **static** **void** share(**final** Activity activity) | |
| 说明 | 分享接口,直接调用,会自动获取下载模版分享 |
| activity | 游戏的的主Activity |

关于Bundle 传值用到的key 可直接调用 KeyConstants里的变量

/\*\*

\* intent 传递数据key ->登录后的token

\*/

**public** **static** String *INTENT\_DATA\_KEY\_TOKEN* = "token";

/\*\*

\* intent 传递数据key -> 支付游戏需要传递记录的数据;如：区服信息，\* 角色信息等 "sn=133242434&role=张三"

\*/

**public** **static** String *INTENT\_DATA\_KEY\_EXTRA* = "extra";

/\*\*

\* 支付key 传递过来的道具编号

\*/

**public** **static** String *INTENT\_DATA\_KEY\_PID* = "pid";

/\*\*

\* 支付金额

\*/

**public** **static** String *INTENT\_DATA\_KEY\_AMOUNT* = "amount";

/\*\*

\* 订单号

\*/

**public** **static** String *INTENT\_DATA\_KEY\_ORDERID* = "orderid";

/\*\*

\* 用户id

\*/

**public** **static** String *INTENT\_DATA\_KEY\_UID* = "uid";