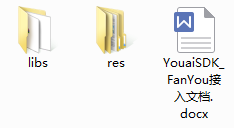
## **自主切换用户中间件接入说明**

注，本SDK应用可以正常使用基于 返回第三方SDK openId 游戏可以找回对应的角色，需要游戏服务器做相应的处理。开启切换功能后游戏需要将用户中心，切换帐号等功能做相应处理

SDK文件清单

解压后文件夹中包括



## **SDK资源接入步骤**

1. 在AndroidManifest.xml 中添加以下权限

<uses-permission android:name=*"android.permission.INTERNET"* />

<uses-permission android:name=*"android.permission.WRITE\_EXTERNAL\_STORAGE"* />

<uses-permission android:name=*"android.permission.MOUNT\_UNMOUNT\_FILESYSTEMS"* />

1. 在AndroidManifest.xml 中添加以下activity

<!-- alipay sdk begin -->

<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>

<!-- alipay sdk end -->

<!-- fanyouSDK start -->

<activity

android:name=*"com.owner.sdk.OwnerSDK"*

android:label=*"owner\_sdk"*

android:theme=*"@android:style/Theme.Dialog"* >

</activity>

<activity android:name=*"com.owner.middleware.utils.SelectRolesActivity"*

android:theme=*"@style/fanyou\_DialogParent"* >

</activity>

<activity

android:name=*"com.owner.middleware.utils.BindActivity"*

android:theme=*"@style/fanyou\_DialogParent"* >

</activity>

<activity

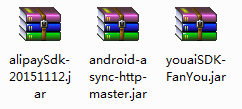
android:name=*"com.owner.middleware.utils.ResultActivity"*

android:theme=*"@style/fanyou\_DialogParent"* >

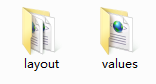
</activity>

<!-- fanyouSDK end -->

1. 添加libs目录下文件到你的工程中



1. 添加资源文件夹res直接拷贝覆盖到你工程中的res文件夹



## **SDK代码接入部分**

1. 初始化

//在接口onCreate时调用

OwnerSdkHelper.*init*(activity, appid, appkey, channel, baseUrl, finishListener, initListener);

示例代码：

OwnerSdkHelper.*init*(**this**, //Context上下文

"Appid",//自建9133平台申请的appid

"AppSecret",//自建9133平台申请的Appsecret

"channel",//自建9133平台申请的子渠道值

"http://1.1.1.1:8080",//后台baseUrl

**new** OwnerSDKEventsListener() {//登录和充值回调listener

@Override

**public** **void** onEventDispatch(**int** i, Intent data) {

Log.*i*("youaisdk","code:" + i + "data:" + data.toString());

**if** (i == OwnerSDK.*LOGIN\_ACTION\_CODE*) {

//登录成功后返回的信息

data.getStringExtra("open\_id") data.getStringExtra("token") data.getStringExtra("timestamp")

} **else** **if** (i == OwnerSDK.*PAY\_ACTION\_CODE*) {

}

}

},**new** OwnerSDKInitListener() {//初始化回调接口

@Override

**public** **void** onSuccess(String content) {

//Log.e("初始化成功回调", content);

}

@Override

**public** **void** onFail(String content) {

//Log.e("初始化失败回调", content);

}

});

2、void record接口

//需在进入游戏时调用，确保可以保存和获取到相应的数据

OwnerSdkHelper.record(

String openId, //渠道SDK登录成功后返回的openid

String userType, //原渠道SDK用户类型

String serverId, //用户服务器id

String userName, //用户名称

String userId, //用户id

String level, //用户等级

String loginType, //登录类型（可传""）

String Custom); //自定义参数（可传""）

1. boolean isShowChangeUser接口

//提供给游戏判断是否开启切换用户功能

OwnerSdkHelper.isShowChangeUser();

4、void showChangeUser接口

//当判断双账号为开启时调用showChangeUser,否则调用游戏正常调用对应SDK逻辑

OwnerSdkHelper.showChangeUser();

1. void pay接口

//支付接口

OwnerSdkHelper.pay(

String serverId, //服务器ID

String roleName, //playerID

**int** amount,//游戏币，支付金额兑换比例IMG_256

到后台的运营资料设置汇率

String callBackInfo);//自定义参数

更新日志 2016-8-24

1. 添加初始化回调
2. 开启双账号如无记录调用游爱账号登录
3. 用游爱账号登录Record接口不重复记录