

中手游SDK接入文档

版本号	修改说明	作者	时间
v2.0.3	文档创建	HooRang	2015-08-31

引入sdk

以library的方式引入sdk库项目(lib_CMGE_SDK_v2.0.3)

Unity的项目请注意: lib_CMGE_SDK_v.jar中包含有assets下面的资源, 由于Unity项目打包时无法自动
把这里的资源拷贝到项目中, 所以需要把jar解压出来手动拷贝到Assets-Plugins-Android-assets目录中

配置AndroidManifest.xml

一、权限配置

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW" />
<uses-permission android:name="android.permission.READ_LOGS"/>
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
```

二、Application配置

SDK默认Application为:com.cmge.sdkkit.framework.mw.app.SDKBaseApplication;
如有自定义的Application:需要继承
com.cmge.sdkkit.framework.mw.app.SDKBaseApplication;
将默认的或者是自定义的Application配置到AndroidManifest.xml中Applicataion节点的
android:name当中;

三、修改SDKKIT_APP_GAMEKEY

```
<meta-data android:name="SDKKIT_APP_GAMEKEY" android:value="填入申请的
gamekey" />
```

客户端接口接入

业务功能接口

一、初始化

```
SDKKitPlatformCore sdkObj =
SDKKitPlatformCore.initPlatformFramework(Activity
activity,SDKKitPlatformCallBack callBack);
```

参数	参数说明
activity	游戏当前activity对象
callBack	初始化全局回调对象，所有sdk接口均通过该对象中的接口方法向外进行回调

二、登录

```
sdkObj.User.login(Activity activity);
```

参数	参数说明
activity	游戏当前activity对象

三、支付

```
ParamsContainer pc = new ParamsContainer();
pc.putInt(ParamsKey.KEY_PAY_AMOUNT, );
pc.putInt(ParamsKey.KEY_PAY_PRODUCT_NUM, );
pc.putString(ParamsKey.KEY_PAY_ORDER_ID, "");
pc.putInt(ParamsKey.KEY_PAY_PRODUCT_ID, );
pc.putString(ParamsKey.KEY_PAY_PRODUCT_NAME, "");
pc.putString(ParamsKey.KEY_PAY_SERVER_ID, "");
pc.putString(ParamsKey.KEY_PAY_SERVER_NAME, "");
pc.putString(ParamsKey.KEY_PAY_ROLE_ID, "");
pc.putString(ParamsKey.KEY_PAY_ROLE_NAME, "");
pc.putString(ParamsKey.KEY_PAY_ROLE_LEVEL, "");
pc.putString(ParamsKey.KEY_EXTINFO, "");
sdkObj.Pay.pay(pc);
```

参数	参数说明
ParamsKey.KEY_PAY_AMOUNT	支付金额，整形，以元为单位
ParamsKey.KEY_PAY_PRODUCT_NUM	购买数量，整形，通常为1
ParamsKey.KEY_PAY_ORDER_ID	游戏方自定义订单号，字符串，会通过服务器回传给cp服务器
ParamsKey.KEY_PAY_PRODUCT_ID	商品id，整形，多个充值档位不允许重复
ParamsKey.KEY_PAY_PRODUCT_NAME	当前购买商品的商品名称，字符串
ParamsKey.KEY_PAY_SERVER_ID	当前玩家所在区服id，字符串
ParamsKey.KEY_PAY_SERVER_NAME	当前玩家所在区服名称，字符串
ParamsKey.KEY_PAY_ROLE_ID	当前玩家角色id，字符串
ParamsKey.KEY_PAY_ROLE_NAME	当前玩家角色名，字符串
ParamsKey.KEY_PAY_ROLE_LEVEL	当前玩家角色等级，字符串
ParamsKey.KEY_EXTINFO	扩展参数，字符串，透传给cp服务器，可以为""

四、注销

```
sdkObj.User.logout();
```

五、用户中心

```
sdkObj.User.userCenter();
```

六、退出游戏

```
sdkObj.Base.exitGame(Activity activity);
```

参数	参数说明
activity	游戏当前activity对象

七、重写Activity生命周期函数

```
@Override
protected void onResume() {
    super.onResume();
}
```

```
        if (sdkObj != null) {
            sdkObj.Lifecycle.onResume();
        }
    }

    @Override
    protected void onPause() {
        super.onPause();
        if (sdkObj != null) {
            sdkObj.Lifecycle.onPause();
        }
    }

    @Override
    protected void onStop() {
        super.onStop();
        if (sdkObj != null) {
            sdkObj.Lifecycle.onStop();
        }
    }

    @Override
    protected void onDestroy() {
        super.onDestroy();
        if (sdkObj != null) {
            sdkObj.Lifecycle.onDestroy();
        }
    }

    @Override
    public void onConfigurationChanged(Configuration newConfig) {
        super.onConfigurationChanged(newConfig);
        if (sdkObj != null) {
            sdkObj.Lifecycle.onConfigurationChanged(newConfig);
        }
    }

    @Override
    protected void onSaveInstanceState(Bundle outState) {
        super.onSaveInstanceState(outState);
        if (sdkObj != null) {
            sdkObj.Lifecycle.onSaveInstanceState(outState);
        }
    }

    @Override
    protected void onNewIntent(Intent intent) {
        super.onNewIntent(intent);
        if (sdkObj != null) {
```

```

        sdkObj.LifeCycle.onNewIntent(intent);
    }
}

```

八、全局回调对象SDKKitPlatformCallBack

- initCallBack

调用SDKKitPlatform.initPlatformFramework()后回调		
参数	参数说明	是否可能为空
retStatus	接口调用返回状态 初始化成功： SDKKitPlatformCallBack.STATUS_SUCCESS 初始化失败： SDKKitPlatformCallBack.STATUS_FAIL	否
retMessage	接口调用返回说明信息	是

- loginCallBack

调用sdkObj.User.login()后回调		
参数	参数说明	是否可能为空
retStatus	接口调用返回状态 登录成功： SDKKitPlatformCallBack.STATUS_SUCCESS 登录失败： SDKKitPlatformCallBack.STATUS_FAIL	否
retMessage	接口调用返回说明信息	是
loginUserId	某个渠道用户唯一标识	否
loginUserName	某个渠道用户名	是
loginAuthToken	用户验证token	否
loginOpenId	全平台用户唯一标识	否

- payCallBack

调用sdkObj.Pay.pay()后回调		
参数	参数说明	是否可能为空
retStatus	接口调用返回状态 支付成功： SDKKitPlatformCallBack.STATUS_SUCCESS 支付失败：SDKKitPlatformCallBack.STATUS_FAIL	否
retMessage	接口调用返回说明信息	是
payKitOrderId	平台支付订单号	否

- logoutCallBack

调用sdkObj.User.logout()后回调		
参数	参数说明	是否可能为空
retStatus	接口调用返回状态 注销成功：SDKKitPlatformCallBack.STATUS_SUCCESS 注销失败：SDKKitPlatformCallBack.STATUS_FAIL	否
retMessage	接口调用返回说明信息	是

- exitGameCallBack

调用sdkObj.Base.exitGame()后回调		
参数	参数说明	是否可能为空
retStatus	接口调用返回状态 点击确定：SDKKitPlatformCallBack.STATUS_SUCCESS 点击取消：SDKKitPlatformCallBack.STATUS_FAIL	否
retMessage	接口调用返回说明信息	是

- getOrderResultCallBack

中手游定制sdk暂时没有使用该回调接口，留空即可		
参数	参数说明	是否可能为空

分享接口

一、打开推送

```
ParamsContainer pc = new ParamsContainer();
pc.putString(ParamsKey.KEY_SHARE_TITLE, "");
pc.putString(ParamsKey.KEY_SHARE_DESCRIPTION, "");
pc.putString(ParamsKey.KEY_SHARE_TEXT, "");
pc.putString(ParamsKey.KEY_SHARE_TARGETURL, "");
pc.putString(ParamsKey.KEY_SHARE_IMAGEURL, "");
sdkObj.Share.share(pc);
```

参数	参数说明
ParamsKey.KEY_SHARE_TITLE	分享标题
ParamsKey.KEY_SHARE_DESCRIPTION	分享描述
ParamsKey.KEY_SHARE_TEXT	游戏方自定义订单号，字符串，会通过服务器回传给cp服务器
ParamsKey.KEY_SHARE_TARGETURL	分享目标路径url
ParamsKey.KEY_SHARE_IMAGEURL	分享带图片，图片url

推送接口

```
sdkObj.Push.startWork(Activity activity);
```

参数	参数说明
activity	游戏当前activity对象

统计接口

一、进入游戏

```
ParamsContainer pc = new ParamsContainer();
pc.putString(ParamsKey.KEY_ROLE_ID, "");
pc.putString(ParamsKey.KEY_ROLE_NAME, "");
pc.putInt(ParamsKey.KEY_ROLE_LEVEL, );
pc.putString(ParamsKey.KEY_SERVER_ID, "");
pc.putString(ParamsKey.KEY_SERVER_NAME, "");
sdkObj.Collections.onDatas(DataTypes.DATA_ENTER_GAME, pc);
```

参数	参数说明	是否可以为空
ParamsKey.KEY_ROLE_ID	当前登录用户角色id	否
ParamsKey.KEY_ROLE_NAME	当前登录用户角色名	否
ParamsKey.KEY_ROLE_LEVEL	当前登录用户角色等级	否
ParamsKey.KEY_SERVER_ID	当前登录区服id	否
ParamsKey.KEY_SERVER_NAME	当前登录区服名	否

二、创建角色

```
ParamsContainer pc = new ParamsContainer();
pc.putString(ParamsKey.KEY_ROLE_ID, "");
pc.putString(ParamsKey.KEY_ROLE_NAME, "");
pc.putString(ParamsKey.KEY_SERVER_ID, "");
pc.putString(ParamsKey.KEY_SERVER_NAME, "");
sdkObj.Collections.onDatas(DataTypes.DATA_CREATE_ROLE, pc);
```

参数	参数说明	是否可以为空
ParamsKey.KEY_ROLE_ID	创建的角色id	否
ParamsKey.KEY_ROLE_NAME	创建的角色名	否
ParamsKey.KEY_SERVER_ID	当前登录用户所在区服id	否
ParamsKey.KEY_SERVER_NAME	当前登录用户所在区服名	否

三、角色升级

```
ParamsContainer pc = new ParamsContainer();
pc.putString(ParamsKey.KEY_SERVER_ID, "");
pc.putString(ParamsKey.KEY_ROLE_LEVEL, "");
pc.putString(ParamsKey.KEY_ROLE_ID, "");
pc.putString(ParamsKey.KEY_ROLE_NAME, "");
pc.putString(ParamsKey.KEY_SERVER_NAME, "");
sdkObj.Collections.onDatas(DataTypes.DATA_ROLE_UPGRADE, pc);
```


参数	参数说明	是否可以为空
ParamsKey.KEY_SERVER_ID	所在区服id	否
ParamsKey.KEY_ROLE_LEVEL	角色升级后的等级	否
ParamsKey.KEY_ROLE_ID	当前登录用户角色id	否
ParamsKey.KEY_ROLE_NAME	当前登录用户角色名	否
ParamsKey.KEY_SERVER_NAME	当前登录用户区服名	否

四、支付统计

```
ParamsContainer pc = new ParamsContainer();
pc.putInt(ParamsKey.KEY_AMOUNT, );
pc.putString(ParamsKey.KEY_SERVER_ID, "");
pc.putString(ParamsKey.KEY_SERVER_NAME, "");
pc.putString(ParamsKey.KEY_ROLE_ID, "");
pc.putString(ParamsKey.KEY_ORDERNUMBER, "");
pc.putString(ParamsKey.KEY_ROLE_GRADE, "");
pc.putString(ParamsKey.KEY_ROLE_NAME, "");
pc.putString(ParamsKey.KEY_PRODUCT_DESC, "");
sdkObj.Collections.onDatas(DataTypes.DATA_PAY, pc);
```

参数	参数说明	是否可以为空
ParamsKey.KEY_AMOUNT	支付金额，以实际付款金额为准	否
ParamsKey.KEY_SERVER_ID	当前登录用户所在区服id	否
ParamsKey.KEY_SERVER_NAME	当前登录用户所在区服名	否
ParamsKey.KEY_ROLE_ID	当前登录用户角色id	否
ParamsKey.KEY_ROLE_NAME	当前登录用户区服名	否
ParamsKey.KEY_ROLE_GRADE	当前登录用户角色等级	否
ParamsKey.KEY_ORDERNUMBER	订单号，可以使用平台支付返回订单号/游戏自己的订单号	否
ParamsKey.KEY_PRODUCT_DESC	支付订单描述	是