39\_IOS平台SDK对接文档

版本号：3.1

更新日期：2018/04/18

|  |  |  |
| --- | --- | --- |
| 版本号 | 时间 | 变更内容 |
| 3.1 | 2018.4.18 | 每个游戏的sdk单独区分，接口变动较大 |
| 3.0 | 2018.3.30 | 3.1.1修改，AppleSdk修改为OrzOSDK，请详细阅读本文档和快速更新文档 |
| 2.4 | 2018.3.6 | 更新app与h5切换的功能 |
| 2.2 | 2018.1.8 | 新加渠道用户中心界面关闭打开回调 |
| 2.0 | 2017-12-15 | 3.1.1修改，接口名称将juhe改为apple，改动较大，请详细阅读本文档和快速更新文档 |
| 1.3 | 2017-12-11 | 第三方支付修改，支付接口改动 |
| 1.2 | 2017-11-28 | 内置39内购 |
| 1.1 | 2017-11-08 | 内置39sdk |
| 1.0 | 2017-09-22 | 创建 |

目录

1. 接入说明 4

1.1. 谁来阅读此文档 4

1.2. 注意事项 4

1.3. 名词定义 4

1.4. 提审服切换接口 4

1.5. 闪屏图、loading图、登录页等替换 4

1.6. 测试及打包 5

2. SDK导入（三步） 5

3. SDK功能(具体代码可以参考sdkdemo) 7

3.1. 简述 7

3.2. 接入前设置 8

3.3. 设置参数 8

3.4. 添加AppDelegate协议(必须接入) 9

3.5. 添加代理协议(必须接入) 10

3.6. 初始化(必须调用) 10

初始化接口在进游戏后立即调用！收到初始化回调后再继续游戏逻辑 10

3.7. 登录接口（必须调用） 10

3.8. 登出接口(必须接入) 10

3.9. 悬浮窗接口(必须接入) 11

1、 显示悬浮窗 11

2、 隐藏悬浮窗 11

3.10. 支付(必须接入) 11

3.11. SDK数据统计接口(必须接入) 12

3.12. SDK回调信息：XRetoolLegendDelegate (必须接入) 14

3.13. 关于APP与H5之间切换的对接（运营无需求，可不对接） 16

4. 常见问题 18

4.1. Framework报错 18

1. 接入说明
   1. 谁来阅读此文档

接入厂商的产品开发技术人员。

* 1. 注意事项
* 接口参数大小写敏感。
* 支持ipod touch，iphone，ipad。要求armv7、armv7s或arm64架构，操作系统要求iOS7.0+
* sdk文件包请在MAC电脑下解压。(重要！否则可能出现头文件损坏)
* 请在buildsettings中的other linkerflags中加入-ObjC。
* 如果使用Xcode9打包，且游戏内使用了相册功能，则需要加以下权限，否则会导致app闪退

C:\Users\Administrator\Documents\Tencent Files\514713374\Image\C2C\{116764CB-4391-A7B0-1A92-B65C4A0D9138}.jpg

C:\Users\Administrator\Documents\Tencent Files\514713374\Image\C2C\{905DA747-74B5-3E51-3112-7E40896C5B5B}.jpg

C:\Users\Administrator\Documents\Tencent Files\514713374\Image\C2C\{F7A73367-073F-49CC-2D64-0ECEF7087AE3}.jpg

* 1. 名词定义

|  |  |
| --- | --- |
| **名词** | **说明** |
| XRetoolLegend\_GamePackageId | 接入时由okwan提供，游戏包id，客户端使用 |
| channel\_okwan\_gid | 接入时由okwan提供，对接服务端使用 |
| channel\_okwan\_apikey: | 接入时由okwan提供，对接服务端使用 |
| channel\_okwan\_gamename | 接入时由okwan提供，对接服务端使用 |
| channel\_okwan\_secretkey | 接入时由okwan提供，对接服务端使用 |

* 1. 提审服切换接口

CP需要预留切换提审服和正式服的接口。在sdk初始化完成后，初始化回调信息里面会有serverid参数，CP根据serverid设置当前显示提审服还是正式服，serverid的值可由双方商定。

此接口主要方便以后提审服切换，不需要每次联系CP切换服务器。

* 1. 闪屏图、loading图、登录页等替换

为了防止提审是出现重复应用，CP需要预留出替换闪屏图、loading图、登录页、logo的接口，并告知我方替换这些图片的路径。

* 1. 测试及打包

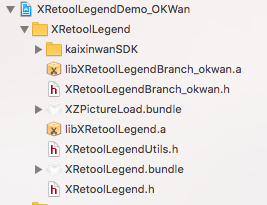
CP接入完，自行测试，登录、支付测试没问题后，将对接好的工程交于我方。我方开始打渠道包。

1. SDK导入（三步）

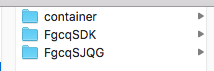
目前IOS SDK的提供方式是示例源码，添加步骤如下：

添加sdk到工程时不要拖动文件到工程，要右击工程add

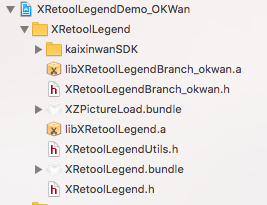
1. 39sdk,有如下部分内容

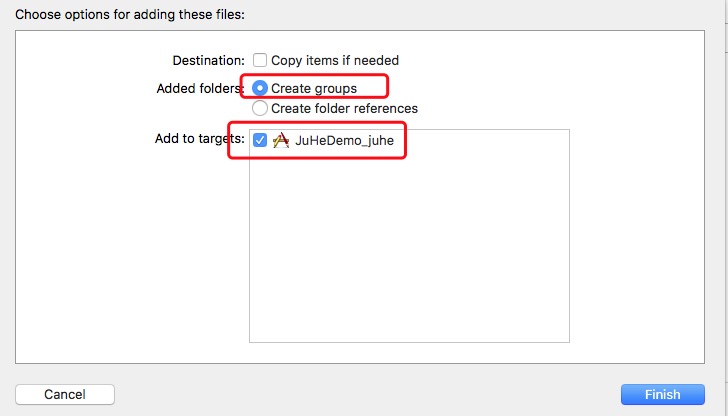


1. 删除重装传奇原有的SDK目录

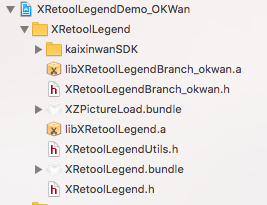


1. 加入39sdk,如下图所示：直接将以上部分sdk文件夹拖入工程

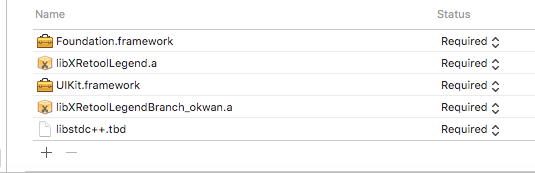




导入后的工程结构如图所示：



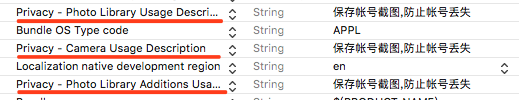
其他Framework:



1. info.plist中加入相册权限

Xcode9以下只需要Privacy - Photo Library Usage Description

Xcode9需要设置以下3个权限



1. SDK功能(具体代码可以参考sdkdemo)
   1. 简述

39IOS平台提供的所有功能都是由类XRetoolLegend来提供，主要功能包括登录，注册，浮标，支付，数据接口等。

* 1. 接入前设置

1、要使用SDK，首先需要导入头文件。如下所示：

#import "XRetoolLegend.h"

#import "XRetoolLegendUtils.h"

2、设置网络权限

C:\Users\Administrator\Documents\Tencent Files\514713374\Image\C2C\{23357E88-DBE5-6EB8-DCD2-05C752B3D405}.jpg

3、添加-ObjC

Other Link Flags中添加-ObjC

* 1. 设置参数

在使用39SDK提供的所有功能前，需要先设置参数，在Info.plist中填入参数；参数有39平台提供

渠道参数：

XRetoolLegendParams: 39游戏包id，由39提供

XRetoolLegend\_landscape : 横竖屏参数，如果是横屏游戏，则传YES；如果是竖屏游戏，则传NO

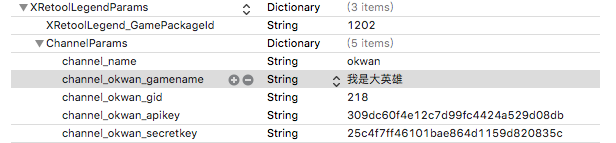
channel\_name :okwan

channel\_okwan\_gid: //okwan提供

channel\_okwan\_apikey: //okwan提供

channel\_okwan\_gamename: //okwan提供

channel\_okwan\_secretkey: //okwan提供其他参数：



* 1. 添加AppDelegate协议(必须接入)

#import "XZsdkManagercdcn.h"

在游戏的主AppDelegate文件中，必须在以下方法中调用相应的SDK方法

- (BOOL)application:(UIApplication \*)application didFinishLaunchingWithOptions:(NSDictionary \*)launchOptions {

[XZsdkManagercdcn XZstartLoginWithGid:@"173" apiKey:@"1095bb9da7a730e3e61b59a53ceeba6b" secretKey:@"08ba476ab94c67f783e49421b026b075" version:@"1.0.0"];

// [KXWsdkManager startLoginWithGid:@"85" apiKey:@"9d8efccf311c58a33cfd7cb62e9ef855" secretKey:@"750357705a95d414902b90be1129ed58" version:@"1.4.5"];

[XZsdkManagercdcn XZReVerifyToService];//向服务器再次发起验证凭证

return YES;

}

- (void)applicationWillResignActive:(UIApplication \*)application {

// Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.

// Use this method to pause ongoing tasks, disable timers, and invalidate graphics rendering callbacks. Games should use this method to pause the game.

}

- (void)applicationDidEnterBackground:(UIApplication \*)application {

// Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.

// If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.

}

- (void)applicationWillEnterForeground:(UIApplication \*)application {

// Called as part of the transition from the background to the active state; here you can undo many of the changes made on entering the background.

[XZsdkManagercdcn XZWillEnterForeground:application];

}

- (void)applicationDidBecomeActive:(UIApplication \*)application {

// Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.

}

- (void)applicationWillTerminate:(UIApplication \*)application {

// Called when the application is about to terminate. Save data if appropriate. See also applicationDidEnterBackground:.

}

//iOS9 之后使用这个回调方法。

- (BOOL)application:(UIApplication \*)app openURL:(NSURL \*)url options:(NSDictionary<NSString \*,id> \*)options

{

return [XZsdkManagercdcn application:app openURL:url options:options];

}

//iOS9 之前使用这个回调方法。

- (BOOL)application:(UIApplication \*)application openURL:(NSURL \*)url sourceApplication:(nullable NSString \*)sourceApplication annotation:(id)annotation

{

return [XZsdkManagercdcn application:application openURL:url sourceApplication:sourceApplication annotation:annotation];

}

* 1. 添加代理协议(必须接入)

#import " XRetoolLegend.h"

#import " XRetoolLegendUtils.h"

@interface ViewController()<XRetoolLegendDelegate>

添加identifier：okwan

URLSchemes：详情请见bundleID(填写为Bundle id)

* 1. 初始化(必须调用)

初始化接口在进游戏后立即调用！收到初始化回调后再继续游戏逻辑

进入游戏首先调用初始化接口。需要在初始化成功之后，再进行登录调用

[XRetoolLegend getSharedInstance]. XRetoolLegendDelegate = self;

[[XRetoolLegend getSharedInstance] initXRetoolLegend];

* 1. 登录接口（必须调用）

[[XRetoolLegend getSharedInstance] start XRetoolLegendLogin];

* 1. 登出接口(必须接入)

[[XRetoolLegend getSharedInstance] start XRetoolLegendLogout];

* 1. 悬浮窗接口(必须接入)

1. 显示悬浮窗

[[XRetoolLegend getSharedInstance] showXRetoolLegendFloatView];

1. 隐藏悬浮窗

[[XRetoolLegend getSharedInstance] hideXRetoolLegendFloatView];

* 1. 支付(必须接入)

填入的参数，数据类型都是NSString

NSDate\* dat = [NSDate dateWithTimeIntervalSinceNow:0];

NSTimeInterval a=[dat timeIntervalSince1970]\*1000;

NSString\* orderId = [NSString stringWithFormat:@"demo%f", a];

NSMutableDictionary \* data = [NSMutableDictionary dictionary];

[data setValue:@"60yuanbao" forKey: XRetoolLegend \_IOS\_PRODUCT\_NAME];//必传，商品名称

[data setValue:@"6" forKey: XRetoolLegend\_IOS\_PRODUCT\_ID];//必传，商品id，由运营提供

[data setValue:orderId forKey:XRetoolLegend\_IOS\_CP\_ORDERID];//必传，CP订单号，若无，则传时间戳

[data setValue:@"XRetoolLegend\_IOS\_PRODUCT\_DESC" forKey: XRetoolLegend\_IOS\_PRODUCT\_DESC];//必传，若无，则传商品名称

[data setValue:@"6" forKey: XRetoolLegend\_IOS\_PRODUCT\_PRICE];//必传，商品价格，单位：元,不带小数点

[data setValue:@"1" forKey: XRetoolLegend\_IOS\_GOODS\_NUM];//必传，商品购买数量；一般传1

[data setValue:@"XRetoolLegend\_IOS\_EXTRA" forKey:XRetoolLegend\_IOS\_EXTRA];//可选，扩展信息，若无，则传空

[data setValue:@"XRetoolLegend\_IOS\_ROLE\_ID" forKey:XRetoolLegend\_IOS\_ROLE\_ID];//可选，角色id，若无，则传空

[data setValue:@"XRetoolLegend\_IOS\_ROLE\_NAME" forKey:XRetoolLegend\_IOS\_ROLE\_NAME];//可选，角色名称，若无，则传空

[data setValue:@"XRetoolLegend\_IOS\_ROLE\_LEVEL" forKey:XRetoolLegend\_IOS\_ROLE\_LEVEL];//可选，角色等级，若无，则传空

[data setValue:@"XRetoolLegend\_IOS\_SERVER\_ID" forKey:XRetoolLegend\_IOS\_SERVER\_ID];//可选，服务器id，若无，则传空

[data setValue:@"XRetoolLegend\_IOS\_SERVER\_NAME" forKey:XRetoolLegend\_IOS\_SERVER\_NAME];//可选，服务器名称，若无，则传空

[[XRetoolLegend getSharedInstance] startXRetoolLegendRegcharg:data];

|  |  |
| --- | --- |
| **名词** | **说明** |
| XRETOOLLEGEND \_IOS\_PRODUCT\_NAME | 商品名称 |
| XRETOOLLEGEND\_ IOS \_PRODUCT\_ID | 商品id，由39提供 |
| XRETOOLLEGEND\_ IOS \_CP\_ORDERID | CP订单号 |
| XRETOOLLEGEND\_ IOS \_PRODUCT\_DESC | 商品描述 |
| XRETOOLLEGEND\_ IOS \_PRODUCT\_PRICE | 商品价格，nsstring类型，不带小数点 |
| XRETOOLLEGEND\_ IOS \_GOODS\_NUM | 购买数量 |
| XRETOOLLEGEND\_ IOS \_EXTRA | 扩展信息，不推荐使用json格式 |
| XRETOOLLEGEND\_ IOS \_ROLE\_ID | 角色id |
| XRETOOLLEGEND\_ IOS \_ROLE\_NAME | 角色名字 |
| XRETOOLLEGEND\_ IOS \_ROLE\_LEVEL | 角色等级 |
| XRETOOLLEGEND\_ IOS \_SERVER\_ID | 服务器id |
| XRETOOLLEGEND\_ IOS \_SERVER\_NAME | 服务器名称 |

* 1. SDK数据统计接口(必须接入)

**数据接口传入参数，均为NSString类型**

1. 创建角色统计

统计节点：完成角色创建后调用

NSMutableDictionary \* data = [NSMutableDictionary dictionary];

[data setValue:@"1" forKey:XRetoolLegend\_ROLE\_ID];//必传，角色ID

[data setValue:@"1" forKey:XRetoolLegend\_ROLE\_LEVEL];//必传，角色等级

[data setValue:@"roleName" forKey:XRetoolLegend\_ROLE\_NAME];//必传，角色名称

[data setValue:@"roleCreateTime" forKey:XRetoolLegend\_ROLE\_CREATE\_TIME];//必传，角色创建时间，时间戳，精确到秒

[data setValue:@"1" forKey:XRetoolLegend\_SERVER\_ID];//必传，服务器id

[data setValue:@"serverName" forKey:XRetoolLegend\_SERVER\_NAME];//必传，服务器名称

[[XRetoolLegend getSharedInstance] sendXRetoolLegendData:XRetoolLegend\_SUBMIT\_ROLE\_CREATE data:data];

1. 角色登陆统计

统计节点：游戏角色成功登陆游戏时调用

NSMutableDictionary \* data = [NSMutableDictionary dictionary];

[data setValue:@"1" forKey:XRetoolLegend\_ROLE\_ID];//必传，角色ID

[data setValue:@"1" forKey:XRetoolLegend\_ROLE\_LEVEL];//必传，角色等级

[data setValue:@"roleName" forKey:XRetoolLegend\_ROLE\_NAME];//必传，角色名称

[data setValue:@"roleCreateTime" forKey:XRetoolLegend\_ROLE\_CREATE\_TIME];//必传，角色创建时间，时间戳，精确到秒

[data setValue:@"1" forKey:XRetoolLegend\_SERVER\_ID];//必传，服务器id

[data setValue:@"serverName" forKey:XRetoolLegend\_SERVER\_NAME];//必传，服务器名称

[[XRetoolLegend getSharedInstance] sendXRetoolLegendData:XRetoolLegend\_SUBMIT\_ROLE\_ENTERSERVER data:data];

1. 等级统计

统计节点：角色升级时记录

NSMutableDictionary \* data = [NSMutableDictionary dictionary];

[data setValue:@"1" forKey:XRetoolLegend\_ROLE\_ID];//必传，角色ID

[data setValue:@"1" forKey:XRetoolLegend\_ROLE\_LEVEL];//必传，角色等级

[data setValue:@"roleName" forKey:XRetoolLegend\_ROLE\_NAME];//必传，角色名称

[data setValue:@"roleCreateTime" forKey:XRetoolLegend\_ROLE\_CREATE\_TIME];//必传，角色创建时间，时间戳，精确到秒

[data setValue:@"1" forKey:XRetoolLegend\_SERVER\_ID];//必传，服务器id

[data setValue:@"serverName" forKey:XRetoolLegend\_SERVER\_NAME];//必传，服务器名称

[[XRetoolLegend getSharedInstance] sendXRetoolLegendData:XRetoolLegend\_SUBMIT\_ROLE\_LEVELUP data:data];

* 1. SDK回调信息：XRetoolLegendDelegate (必须接入)

1. **Sdk初始化成功/失败**

//---------------XRetoolLegendDelegate---------------

-(void)initXRetoolLegendSuccess:(NSDictionary \*)result{

NSLog(@"initSuccess : %@", result);

/\*

result内容：{

code = 0;

msg = "init success!";

serverid = "xxxx";

}

\*/

//初始化成功回调中有serverid参数，此参数是39后台配置的服务器状态

//游戏研发可以用此参数来切换提审服和正式服

//提审服和正式服的serverid可以双方商定

NSString \*serverid = [result objectForKey:@"serverid"];

}

-(void)initXRetoolLegendFail:(NSDictionary \*)result{

NSLog(@"initFail : %@", result);

}

**初始化成功，会回调服务器状态serverid，研发可根据此状态切换提审服和正式服。这样之后切换提审服和正式服就不需要再联系研发修改了。**

1. **Sdk登录成功/失败**

-(void)loginXRetoolLegendSuccess:(NSDictionary \*)result{

NSLog(@"loginSuccess : %@", result);

[[XRetoolLegend getSharedInstance] showXRetoolLegendFloatView];

}

-(void)loginXRetoolLegendFail:(NSDictionary \*)result{

NSLog(@"loginFail , result = %@", result);

}

登陆成功回调：字典类型result中的内容说明如下：

channel\_id:渠道id；用来告知游戏，当前接入的是哪个渠道

channel\_name:渠道名称；用来告知游戏，当前接入的是哪个渠道

channel\_userid:渠道的用户id，例如：渠道为itools，则此字段为itools的用户id

code:返回码，0表示登陆成功

juhe\_nickname:39用户昵称

juhe\_token : 39登录验证使用的token

juhe\_userid:39用户id，即channel\_userid在39平台所对应的用户id

juhe\_username:39用户名

msg:登录信息

**关于用户唯一标识的说明：**

**如果游戏在渠道A上已经上线，则用39sdk打渠道A的包时，使用channel\_userid作为用户唯一标识**

**如果游戏在渠道A上从未上线过，则用39sdk打渠道A的包时，使用juhe\_userid作为用户唯一标识**

**以上主要为了防止出现角色丢失的情况**

1. **Sdk登出成功/失败**

-(void)logoutXRetoolLegendSuccess:(NSDictionary \*)result{

NSLog(@"logoutSuccess , result = %@", result);

}

-(void)logoutXRetoolLegendFail:(NSDictionary \*)result{

NSLog(@"logoutFail , result = %@", result);

}

1. **Sdk支付成功/失败**

-(void)RegchargXRetoolLegendSuccess:(NSDictionary \*)result{

NSLog(@" result = %@", result);

}

-(void)RegchargXRetoolLegendFail:(NSDictionary \*)result{

NSLog(@"result = %@", result);

}

1. **sdk用户中心关闭打开回调**

-(void)openXRetoolLegendUserCenter{

NSLog(@"openUserCenter");

}

-(void)closeXRetoolLegendUserCenter{

NSLog(@"closeUserCenter");

}