**天宇游 iOS SDK接入参考手册**

V1.0.9

目录

[1 获得接入参数及资源包 4](#_Toc525031245)

[1.1 SDK使用注意 4](#_Toc525031246)

[1.2 禁止使用UIAlertView 4](#_Toc525031247)

[2 开发环境兼容性 4](#_Toc525031248)

[2.1 兼容性处理 4](#_Toc525031249)

[Xcode版本低于9.4，没有Swift4.1.2从而导致无法编译运行。 4](#_Toc525031250)

[2.1.1 下载Swift4.1.2插件 4](#_Toc525031251)

[2.1.2 切换Swift版本 5](#_Toc525031252)

[3 TYYSDK构成 5](#_Toc525031253)

[3.1 简介 5](#_Toc525031254)

[3.2 第三方依赖库 5](#_Toc525031255)

[4 资源引入和环境配置 6](#_Toc525031256)

[4.1 引入资源 6](#_Toc525031257)

[4.2 项目配置 6](#_Toc525031258)

[4.2.1 (必须)General配置 6](#_Toc525031259)

[4.2.2 (必须)Info配置 7](#_Toc525031260)

[4.2.3 (必须)Build Setting设置 8](#_Toc525031261)

[4.2.4 OC工程额外配置 10](#_Toc525031262)

[4.2.5 全屏配置 11](#_Toc525031263)

[4.3 闪屏配置 12](#_Toc525031264)

[4.3.1 拷贝文件 12](#_Toc525031265)

[4.3.2 修改配置 13](#_Toc525031266)

[5 参数配置 13](#_Toc525031267)

[6 SDK类及方法介绍 13](#_Toc525031268)

[6.1 TYYSDK 13](#_Toc525031269)

[6.2 TYYViewController 13](#_Toc525031270)

[6.3 TYYPayUploadParams 14](#_Toc525031271)

[6.4 TYYPayData 14](#_Toc525031272)

[6.5 TYYUserInfoUploadParams 14](#_Toc525031273)

[6.6 TYYChatUploadParams 14](#_Toc525031274)

[6.7 TYYLoginResultData 14](#_Toc525031275)

[7 OC调用SDK方法 14](#_Toc525031276)

[7.1 头文件引入 14](#_Toc525031277)

[7.2 (必接)继承TYYViewController 15](#_Toc525031278)

[7.3 (必接)初始化SDK 15](#_Toc525031279)

[7.4 (必接)登录 15](#_Toc525031280)

[7.5 注销 15](#_Toc525031281)

[7.6 (必接)支付充值 15](#_Toc525031282)

[7.7 (必接)上传角色信息 15](#_Toc525031283)

[7.8 上传聊天记录 15](#_Toc525031284)

[7.9 显示主界面 15](#_Toc525031285)

[7.10 (必接)Application回调 16](#_Toc525031286)

[7.11 输出Log 16](#_Toc525031287)

[7.12 是否汇报崩溃日志 16](#_Toc525031288)

[7.13 提示弹窗 16](#_Toc525031289)

[8 Swift调用SDK方法 16](#_Toc525031290)

[8.1 头文件引入 16](#_Toc525031291)

[8.2 (必接)继承TYYViewController 17](#_Toc525031292)

[8.3 (必接)初始化SDK 17](#_Toc525031293)

[8.4 (必接)登录 17](#_Toc525031294)

[8.5 注销 17](#_Toc525031295)

[8.6 (必接)支付充值 17](#_Toc525031296)

[8.7 (必接)上传角色信息 17](#_Toc525031297)

[8.8 上传聊天记录 17](#_Toc525031298)

[8.9 显示主界面 17](#_Toc525031299)

[8.10 (必接)Application回调 18](#_Toc525031300)

[8.11 输出Log 18](#_Toc525031301)

[8.12 是否汇报崩溃日志 18](#_Toc525031302)

[8.13 提示弹窗 18](#_Toc525031303)

1. 获得接入参数及资源包

请联系客服索取

* 1. SDK使用注意

请勿修改SDK中文件名称

* 1. 禁止使用UIAlertView

由于UIAlertView会导致，SDK界面异常。

如需弹窗请使用TYYSDK中showError或showToast

1. 开发环境兼容性

* Xcode9.0以上
* Swift4.1.2
* iOS9.0及以上版本
* 构架支持armv7 arm64
  1. 兼容性处理

Xcode版本低于9.4，没有Swift4.1.2从而导致无法编译运行。

* + 1. 下载Swift4.1.2插件

https://swift.org/builds/swift-4.1.2-release/xcode/swift-4.1.2-RELEASE/swift-4.1.2-RELEASE-osx.pkg

* + 1. 切换Swift版本

****

1. TYYSDK构成
   1. 简介

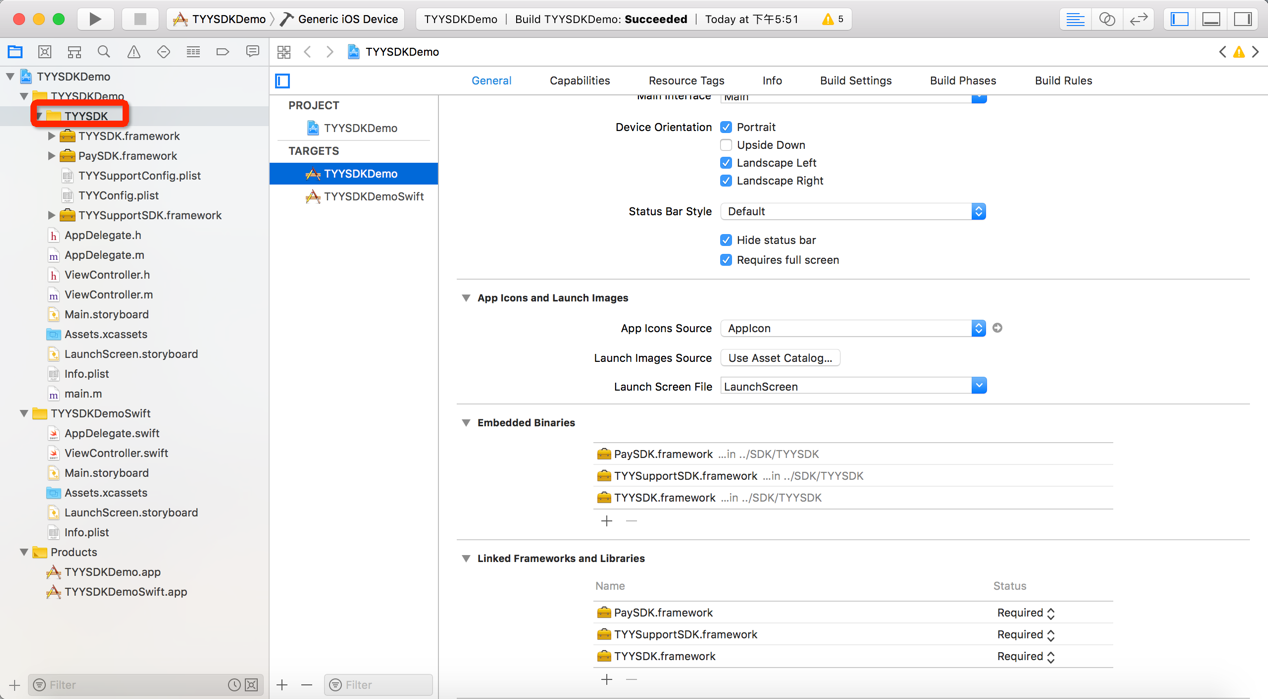
* TYYSDK.framework：游戏代码中使用的核心库
* TYYSupportSDK.framework：平台SDK支持库，游戏中不需要使用
* PaySDK.framework：支付支持库，游戏中不需要使用
* TYYConfig.plist：发行配置参数
* TYYSupportConfig.plist：SDK配置参数
  1. 第三方依赖库
* Alamofire.framework：版本号4.7
* AlamofireImage.framework：版本号3.3
* AlamofireNetworkActivityIndicator.framework：版本号2.2
* AlamofireObjectMapper.framework：版本号5.1
* ObjectMapper.framework：版本号3.3
* SQLite.framework：版本号0.11.5

如第三库依赖库冲突，请联系客服。

1. 资源引入和环境配置
   1. 引入资源

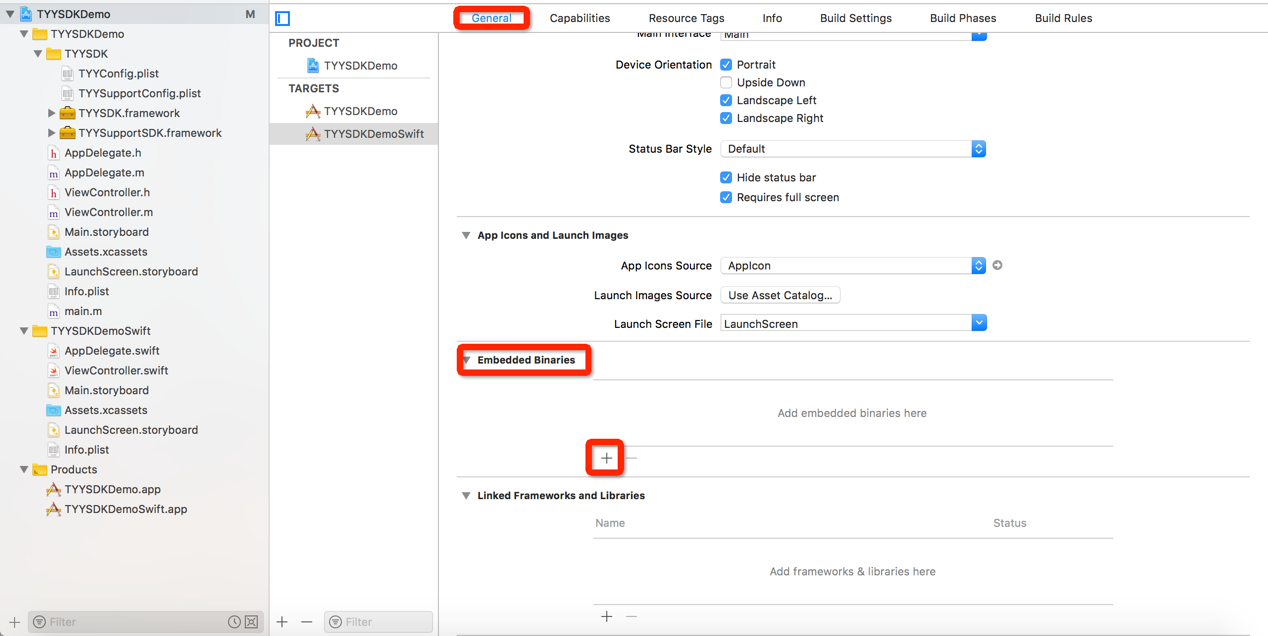
添加SDK包，将SDK文件夹直接拖拽到项目工程里,并勾选copy items。

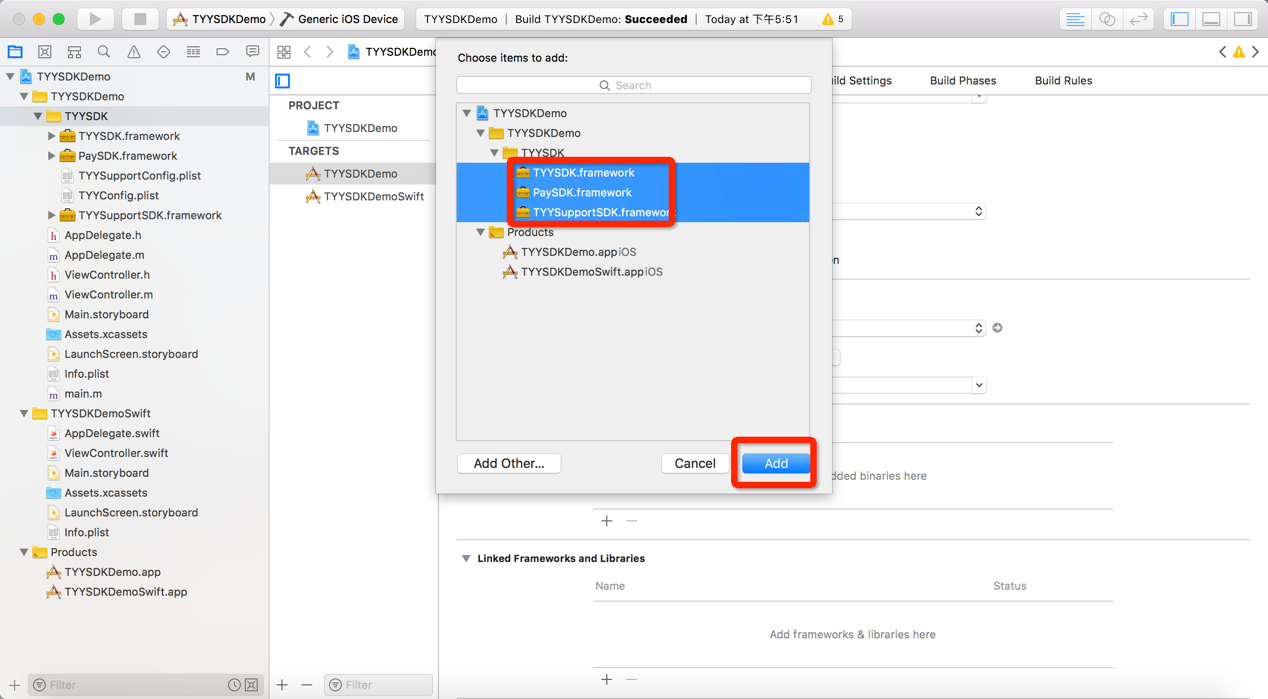
如果觉得SDK比较大，发布时使用Device目录下的包。

****

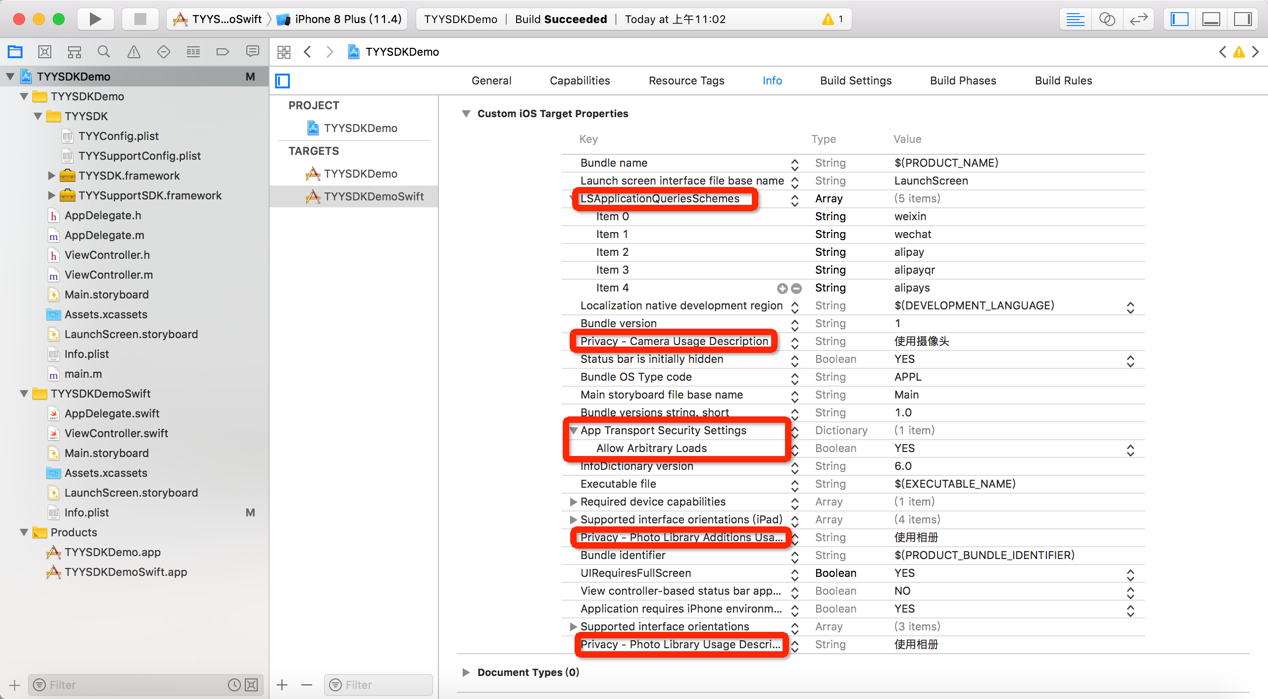
* 1. 项目配置
     1. (必须)General配置

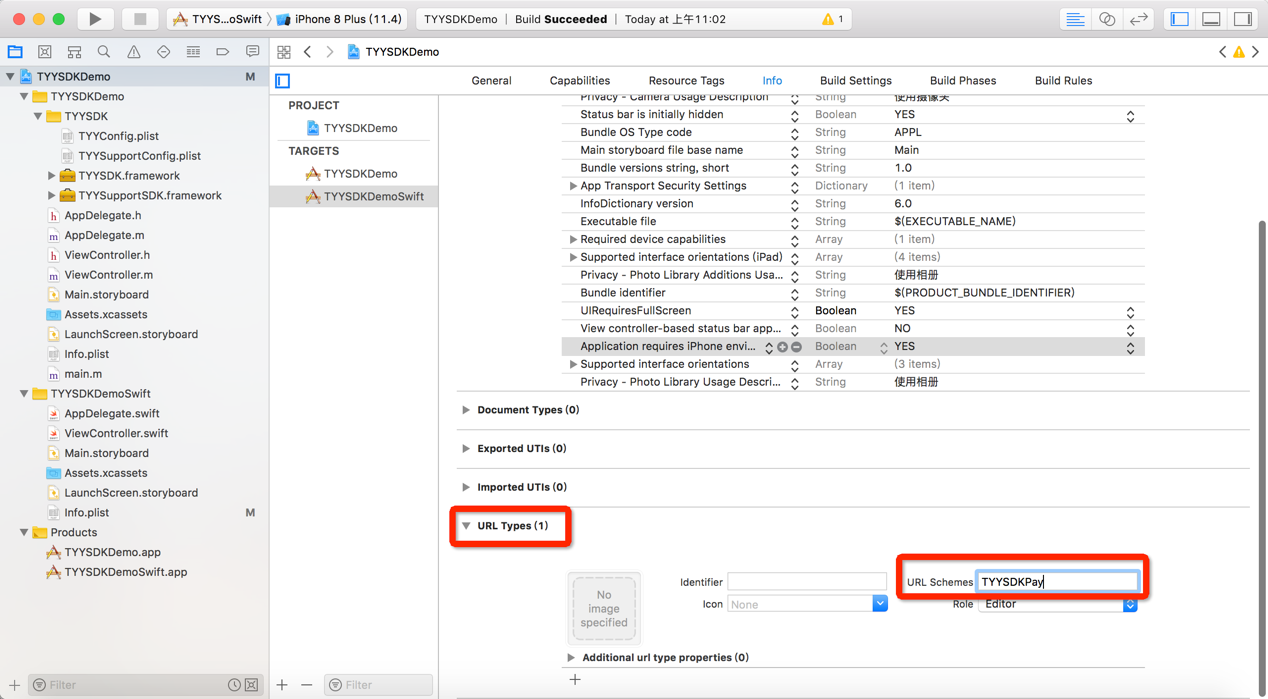
添加TYYSDK.framework，TYYSupportSDK.framework和PaySDK.framework到Embedded Binaries



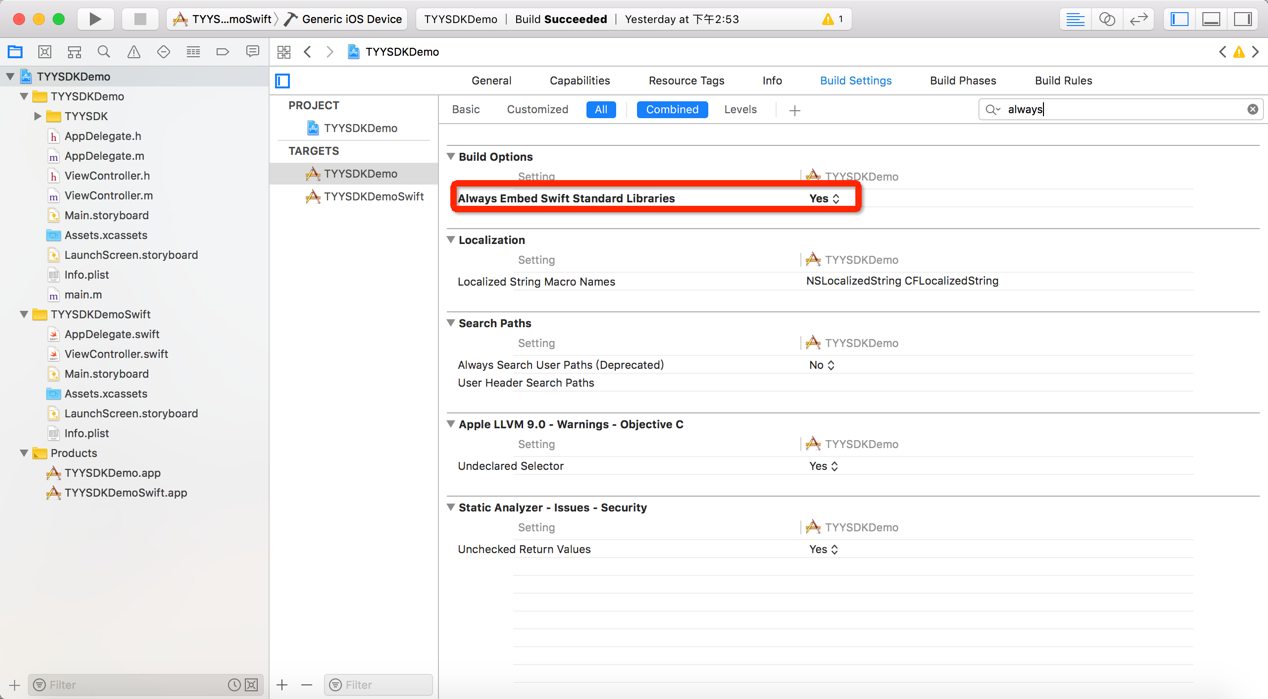
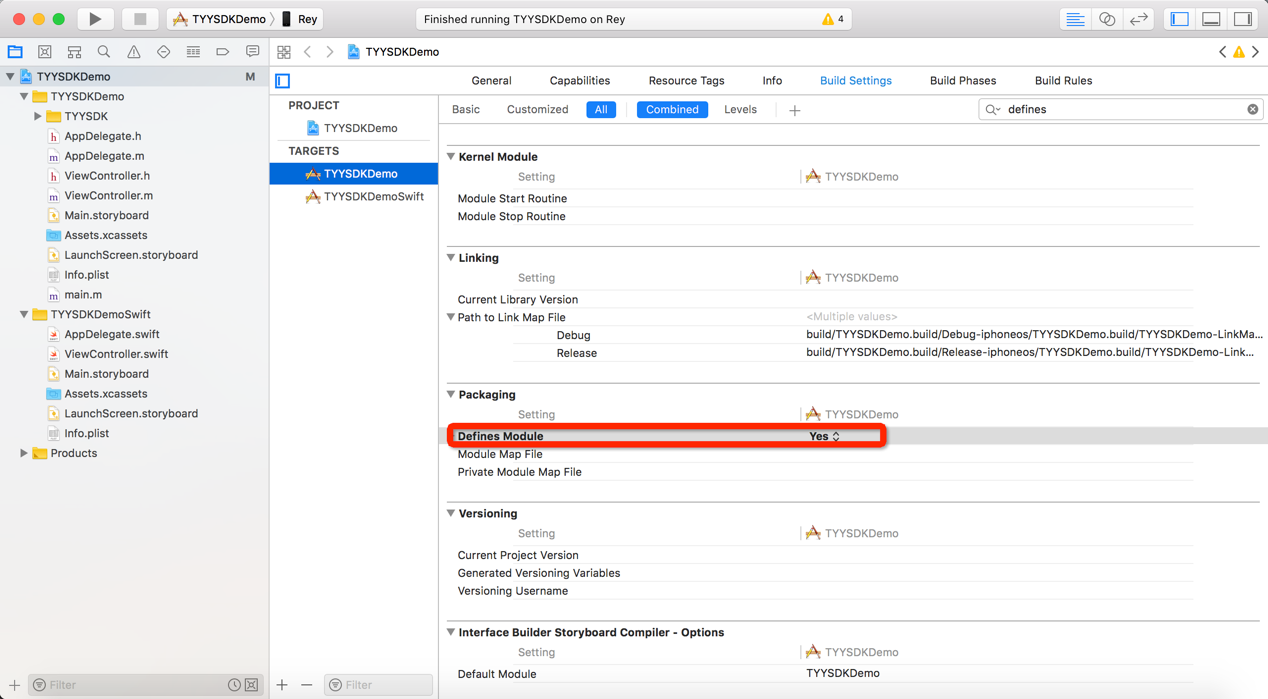


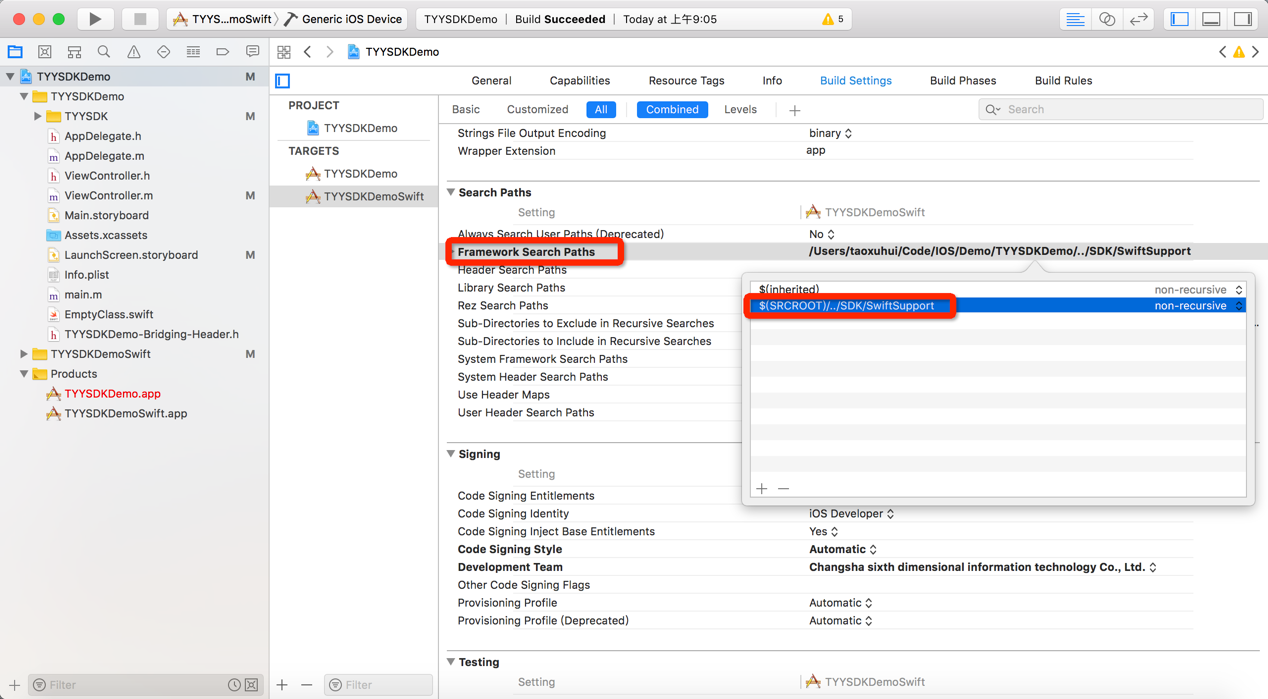
* + 1. (必须)Info配置
* 添加＂App Transport Security Settings＂字段，并设置＂Allow Arbitrary Loads＂为YES；
* 添加＂Privacy - Camera Usage Description＂字段，并设置提示语句＂使用摄像头＂，该内容可随意填写；
* 添加＂Privacy - Photo Library Additions Usage Description＂字段，并设置提示语句＂使用相册＂，该内容可随意填写；
* 添加＂Privacy - Photo Library Usage Description＂字段，并设置提示语句＂使用相册＂，该内容可随意填写；
* 添加＂LSApplicationQueriesSchemes＂字段，并设置数组内容为＂weixin＂，＂wechat＂，＂alipay＂，＂alipayqr＂，＂alipays＂，＂mqqapi＂，＂mqq＂，＂mqqwpa＂，＂mqzone＂；
* 添加URL Types，设置＂URL Schemes＂为＂TYYSDKPay%%%%%＂，其中%%%%%为申请的appid。appid可以在TYYConfig.plist文件中找到。
* 添加URL Types，设置＂URL Schemes＂为＂wx2a5538052969956e＂。





* + 1. (必须)Build Setting设置
* 设置＂Enable Bitcode＂为＂NO＂；
* (OC工程需额外设置)＂Defines Module＂为＂YES＂
* (OC工程需额外设置)＂Always Embed Swift Standard Libraries＂为＂YES＂
* (Swift工程需额外设置)由于swift不支持静态库，所以需要设置Framework Search Paths 指定到SDK中＂SwiftSupport＂文件夹的路径，不需要将该文件夹下的库引入到工程，因为TYYSDK已经包含。

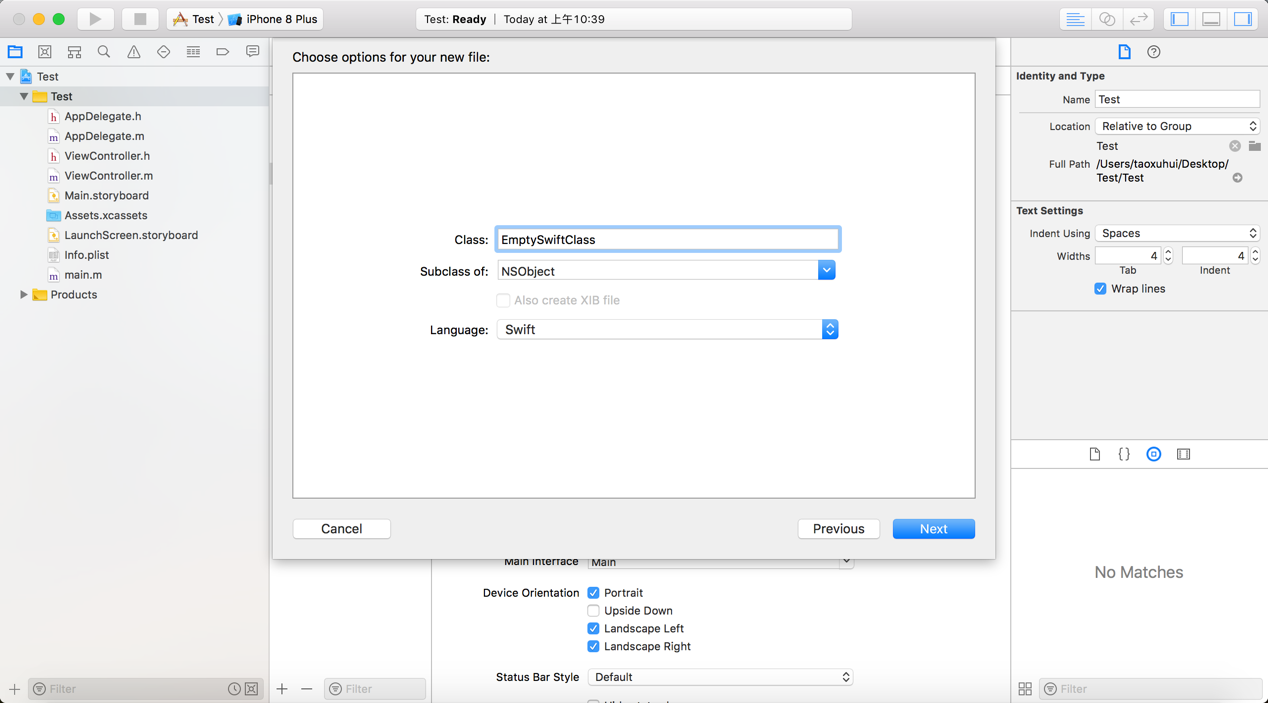


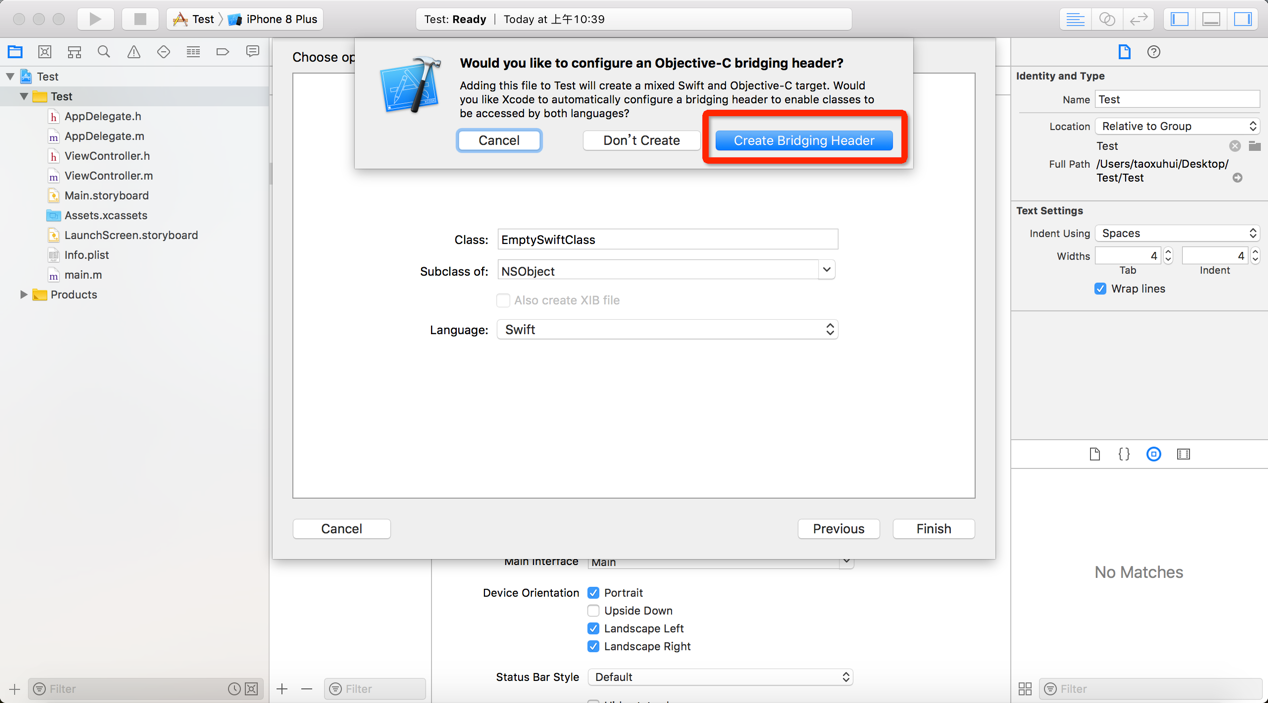


* + 1. OC工程额外配置

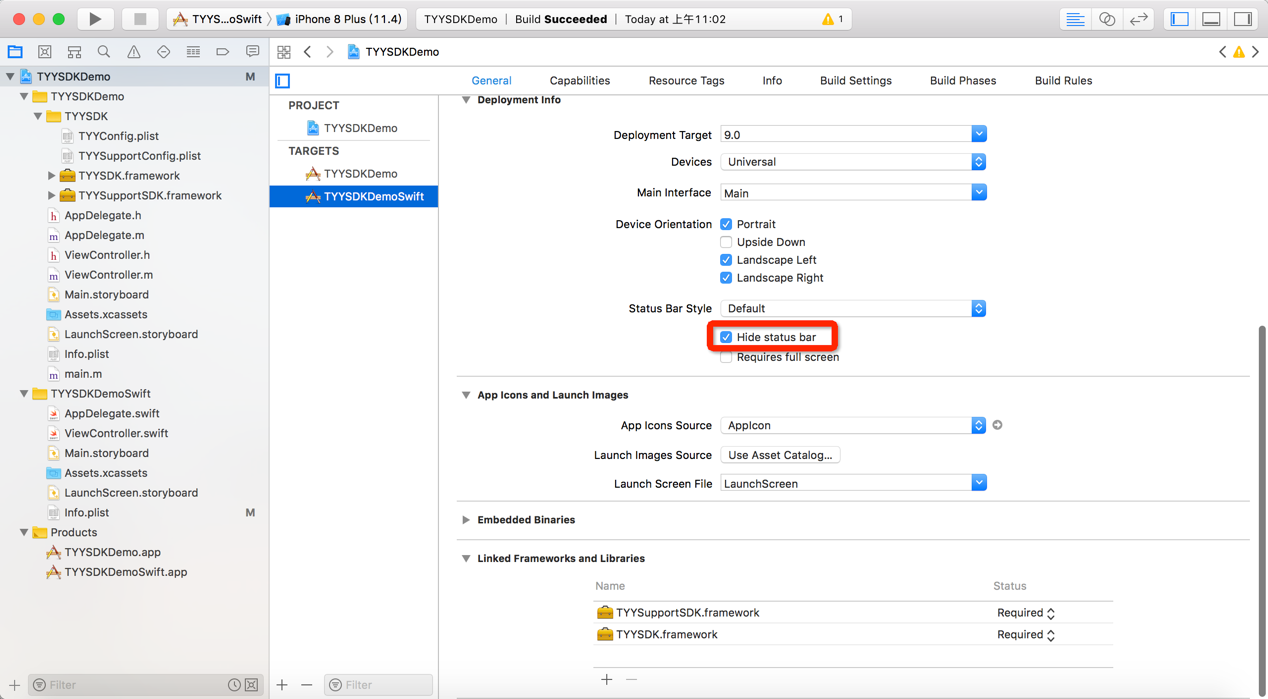
由于TYYSDK是用Swift写的，所以OC工程需要添加一个空的Swift类，让工程会自动引入Swift标准库。

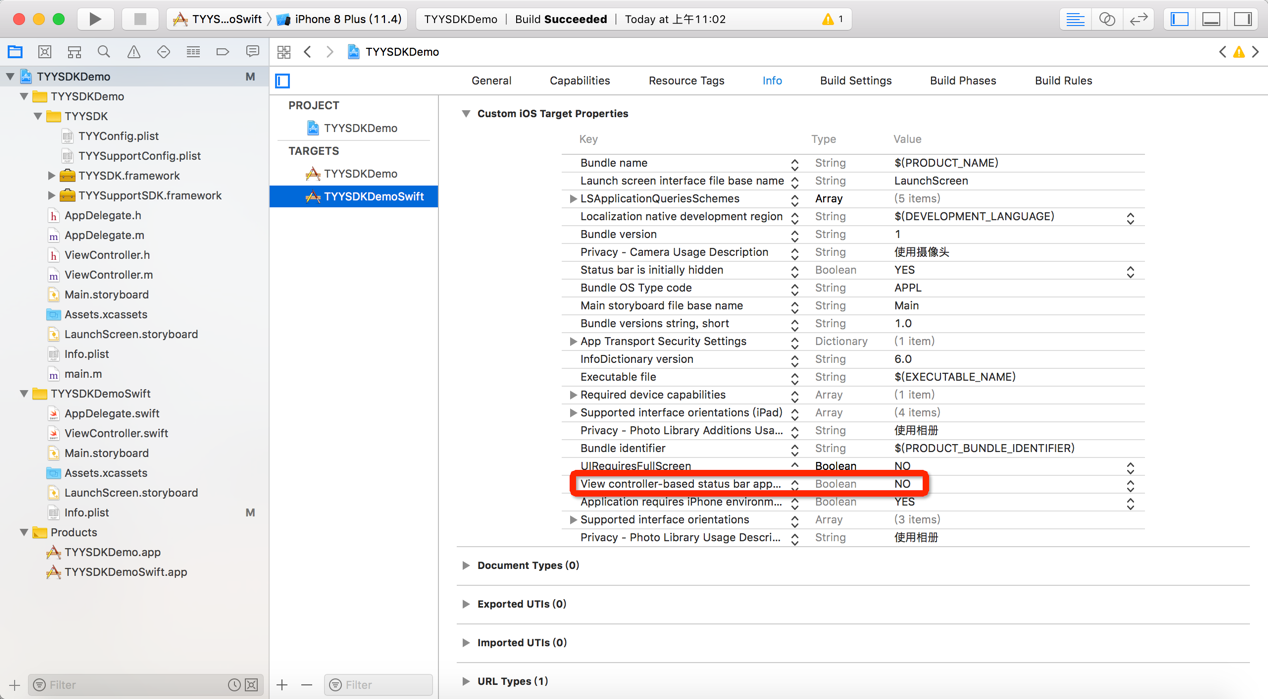
创建的Swift类，在文件中必须添加＂import UIKit＂，不能删除否则无法自动引入Swift标准库。





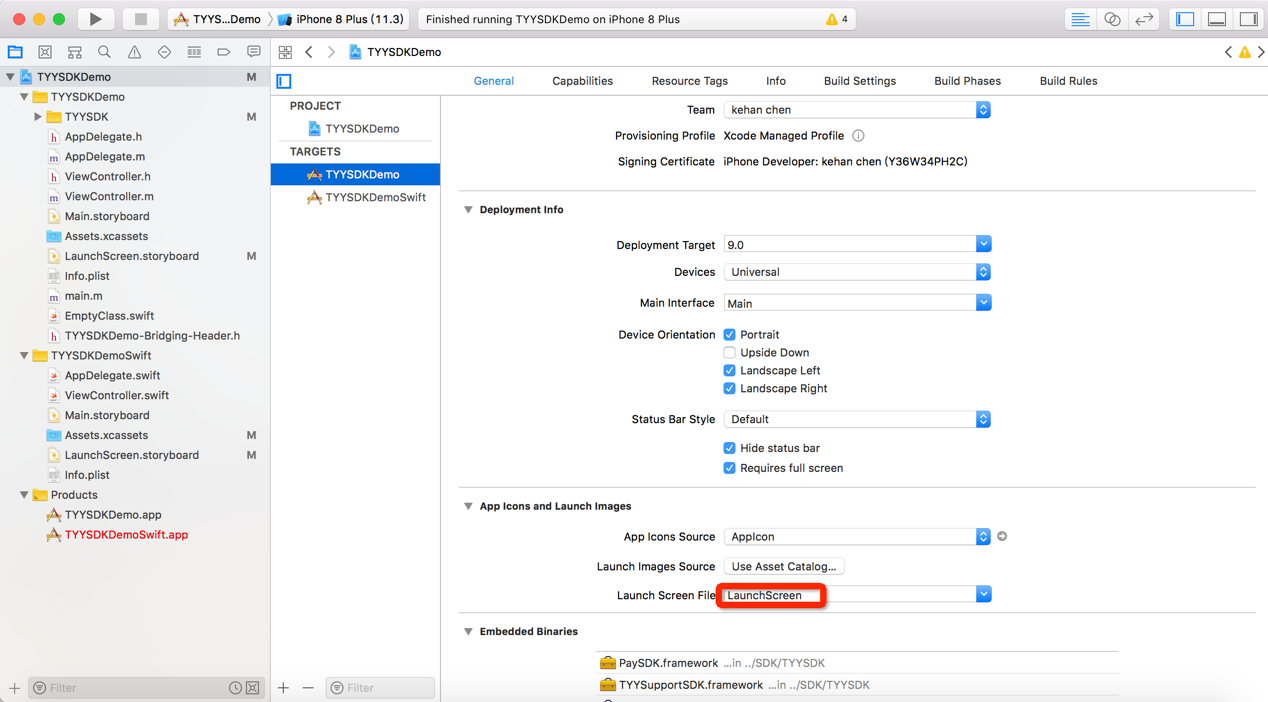
* + 1. 全屏配置
* General配置页中，勾选＂Hide status bar＂；
* Info配置页中，添加＂View controller-based status bar appearance＂，并设置为＂NO＂；





* 1. 闪屏配置
     1. 拷贝文件
* 拷贝Demo中的LaunchScreen.storyboard替换工程中的
* 拷贝Aessets.xcassets中＂launchbottom.imageset＂，＂launchlogo-land.imageset＂，＂launchlogo.imageset＂文件夹到自己工程Aessets.xcassets下
  + 1. 修改配置

修改Launch Screen File为＂LaunchScreen＂



1. 参数配置

替换客服提供的TYYConfig.plist和TYYSupportConfig.plist 2个文件

1. SDK类及方法介绍
   1. TYYSDK

核心功能类，SDK功能调用都需要使用该类。

* 1. TYYViewController

UIViewController基类，只是为了实现＂摇一摇＂显示隐藏悬浮框，没有额外获取。

* 1. TYYPayUploadParams

支付参数类。

* 1. TYYPayData

支付返回数据类。

* 1. TYYUserInfoUploadParams

用户数据上传参数类。

* 1. TYYChatUploadParams

聊天信息上传参数类。

* 1. TYYLoginResultData

登录返回数据类。

1. OC调用SDK方法

只进行简单概述,详细介绍请查看Demo，代码中已做详细注释。

* 1. 头文件引入

#import <TYYSDK/TYYSDK-Swift.h>

* 1. (必接)继承TYYViewController
  2. (必接)初始化SDK

[[TYYSDK share] initSdkWithInitCallback: loginCallback: payCallback: logoutCallback:]

* 1. (必接)登录

[[TYYSDK share] login]

* 1. 注销

[[TYYSDK share] logout]

* 1. (必接)支付充值

[[TYYSDK share] pay:uploadData]

* 1. (必接)上传角色信息

[[TYYSDK share] uploadUserInfo: callback:]

* 1. 上传聊天记录

[[TYYSDK share] uploadChat: callback:]

* 1. 显示主界面

[[TYYSDK share] showMainView]

* 1. (必接)Application回调

由于以下接口是支付需要调用,所以必接接入

[[TYYSDK share] application:application didFinishLaunchingWithOptions:launchOptions]

[[TYYSDK share] applicationWillEnterForeground:application]

[[TYYSDK share] application:app open:url options:options]

[[TYYSDK share] application:application open:url sourceApplication:sourceApplication annotation:annotation]

[[TYYSDK share] application:application handleOpen:url]

* 1. 输出Log

[TYYSDK setDebugWithDebug:true]

* 1. 是否汇报崩溃日志

[TYYSDK setReportWithreport:true]

* 1. 提示弹窗

[TYYSDK showError]　样式和UIAlertView和UIAlertViewController一样

[TYYSDK showToast] 自动隐藏的提示框

1. Swift调用SDK方法

只进行简单概述,详细介绍请查看Demo，代码中已做详细注释。

* 1. 头文件引入

import TYYSDK

* 1. (必接)继承TYYViewController
  2. (必接)初始化SDK

TYYSDK.share().initSdk(InitCallback: loginCallback: payCallback: logoutCallback:)

* 1. (必接)登录

TYYSDK.share().login()

* 1. 注销

TYYSDK.share().logout()

* 1. (必接)支付充值

TYYSDK.share().pay(uploadData)

* 1. (必接)上传角色信息

TYYSDK.share().uploadUserInfo(,callback:)

* 1. 上传聊天记录

TYYSDK.share().uploadChat(,callback:)

* 1. 显示主界面

TYYSDK.share().showMainView()

* 1. (必接)Application回调

由于以下接口是支付需要调用,所以必接接入

TYYSDK.share().application(application,didFinishLaunchingWithOptions:launchOptions)

TYYSDK.share().application(app,open:url,options:options)

TYYSDK.share().application(application,open:url,sourceApplication:sourceApplication,annotation:annotation)

TYYSDK.share().application(application,handleOpen:url)

TYYSDK.share().applicationWillEnterForeground(application)

* 1. 输出Log

TYYSDK.setDebug(debug: true)

* 1. 是否汇报崩溃日志

TYYSDK.setReport(report:true]

* 1. 提示弹窗

[TYYSDK showError]　样式和UIAlertView和UIAlertViewController一样

[TYYSDK showToast] 自动隐藏的提示框