金融平台SDK集成

2017-03-02

目录

[一、概述 3](#_Toc476229311)

[二、SDK集成 3](#_Toc476229312)

[1、iOS项目集成SDK 3](#_Toc476229313)

[1.1、引入文件 3](#_Toc476229314)

[1.1.1、引入头文件 3](#_Toc476229315)

[1.1.2、引入库文件 4](#_Toc476229316)

[1.2、项目配置 4](#_Toc476229317)

[1.2.1、 Other Link Flags 4](#_Toc476229318)

[1.2.2、添加支持库 4](#_Toc476229319)

[1.2.3、Header Search Path 4](#_Toc476229320)

[1.2.4、Keychain Sharing 5](#_Toc476229321)

[1.2.5、bitcode 6](#_Toc476229322)

[1.3、支付操作 6](#_Toc476229323)

[1.3.1、注册appID 6](#_Toc476229324)

[1.3.2、设置图片位置 6](#_Toc476229325)

[1.3.3、调起支付 6](#_Toc476229326)

[2、Android项目集成SDK 7](#_Toc476229327)

# 一、概述

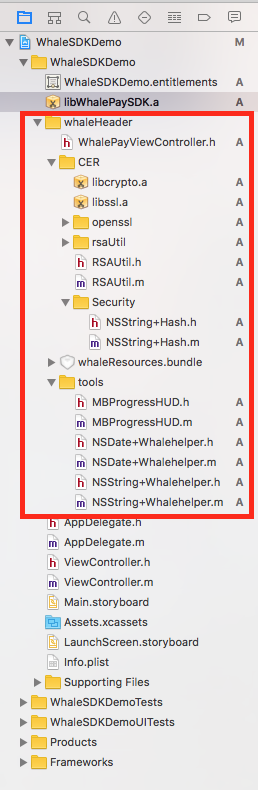
# 二、SDK集成

# 1、iOS项目集成SDK

## 1.1、引入文件

### 1.1.1、引入头文件

在whaleHeader文件夹中，CER文件夹为加密解密相关代码，tools里面为sdk需要工具类，接入者对文件内方法进行修改可能会导致错误，如有需要可添加方法。whaleResources.bundle为资源文件，接入者可替换内部图片，要保持名称不变。



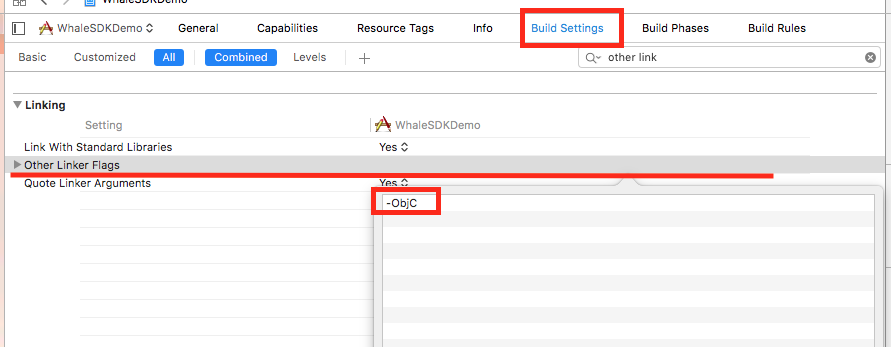
### 1.1.2、引入库文件

在项目中引入libWhalePaySDK.a

## 1.2、项目配置

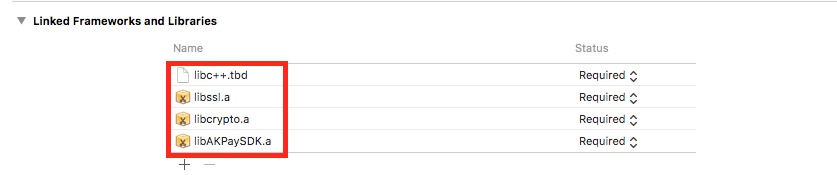
### 1.2.1、 Other Link Flags

在targets->build Setting 中配置Other Link Flags 中的参数设定为-ObjC



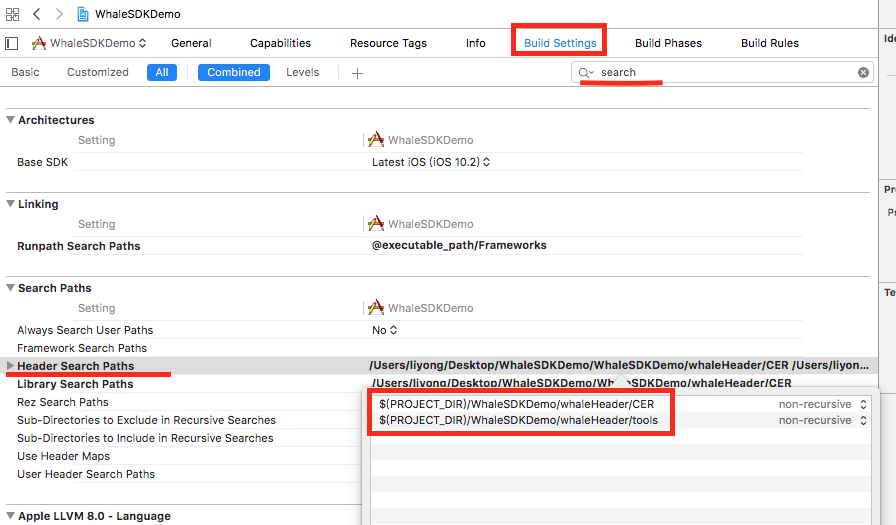
### 1.2.2、添加支持库

在targets->General中配置libc++.tbd文件，将libcrypto.a和libssl.a两个类库拖进工程中，这两个属于openssl库的静态库



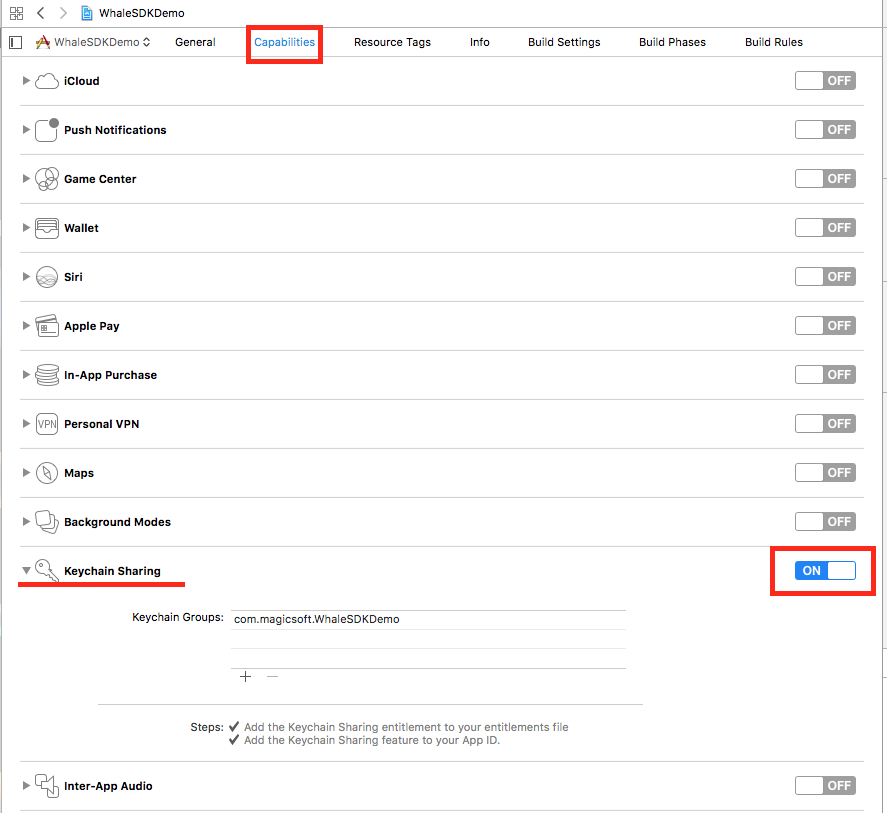
### 1.2.3、Header Search Path

将Demo中的whaleHeader文件夹拖进工程，如果出现头文件找不到的问题，请在targets->build Setting 中配置Header Search Path 配置CER文件夹的路径



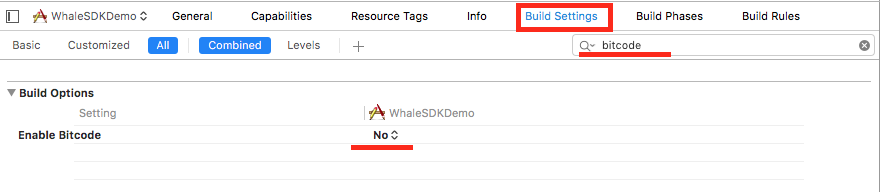
### 1.2.4、Keychain Sharing

xcode8更新后，插件加密方法完全失效，需要打开targets -> Capabilities -> Keychain Sharing



### 1.2.5、bitcode

运行时如果出现bitcode导致的错误可将值设置为no



## 1.3、支付操作

### 1.3.1、注册appID

使用支付工具前需要在AppDelegate.m的 - (BOOL)application:(UIApplication \*)application didFinishLaunchingWithOptions:(NSDictionary \*)launchOptions;方法中向金融平台注册appid，代码如下：

//向金融平台注册appID

/\*\*

\* 打开app时注册 包含：1.注册APPID（微信）2.appkey 3.appsecret

\* key value

\* aliSchemel 支付宝schemel

\* wxAppid 微信支付ID，例如“wx42967af88ec67866”

\* appkey 接入应用新增成功后，系统将自动生成20位的appkey和32位的appsercet

\* appsecret 接入应用新增成功后，系统将自动生成20位的appkey和32位的appsercet

\*/

[[AKPayViewController sharedInstance] setAppId:@{

@"appkey" : @"",

@"appsecret" : @" "

}];

### 1.3.2、设置图片位置

如果需要的话可以在调起支付前通过[AKPayViewController sharedInstance] .bundleName = @"";更改库文件加载图片的位置，要保证更改的图片名称不变。----暂不可用！

### 1.3.3、调起支付

调起支付时需要首先配置好订单信息，下面只填写了必要参数，未填写可选参数。

必要参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 含义 | 备注 |
| pno | string | 平台账号 | 本app相对于金融平台对应的ID号 |
| cno | string | 用户针对金融平台中当前应用的身份 |  |
| destArray | array | 支持的收款通道 |  |
| money | string | 交易金额 |  |
| orderno | string | 接入方生成的订单编号 |  |
| content | string | 交易说明（显示给用户的内容） |  |
| userId | string | 本app里面当前用户对应的ID号 |  |

可选参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 含义 | 备注 |
| username | string | 本app里面当前用户对应的用户名 |  |
| detail | string | 商品描述 |  |
| mobile | string | 本app里面当前用户的手机号 |  |
| province | string | 省 |  |
| city | string | 市 |  |
| block | string | 行政区 |  |
| gbName | string | 小区名称 |  |
| remoteid | string | 调用方ip地址 |  |
| callback | string | 业务系统回调 |  |
| stoptime | string | 交易失效时间 |  |
| starttime | string | 交易生效时间 |  |

AKOrder \*akOrder = [[AKOrder alloc] init];

akOrder.pno = @"123456789";

akOrder.cno = @"987654321";

akOrder.money = @"0.10";

akOrder.orderno = @"123123123";

akOrder.content = @"机械键盘";

akOrder.userId = @"233";

akOrder.destArray = @[@{

@"atid" : @"2",

@"pano" : @"qg\_platform\_1111"

},

@{

@"atid" : @"4",

@"pano" : @"1004b4aac2537a3b40cdaf3bac370ad0"

},

@{

@"atid" : @"31",

@"pano" : @"1031c5a4efa552b24491bda9d212abbb"

},

@{

@"atid" : @"22",

@"pano" : @"1022b894ecacd84948f39a6c659ac7eb"

},

@{

@"atid" : @"24",

@"pano" : @"10248312ea4508534c3dbee977ddd917"

}];

[[AKPayViewController sharedInstance]createPayment:akOrder viewController:self withCompletion:^(NSDictionary \*result) {

if ([[result objectForKey:@"code"] integerValue] == 0){

[MBProgressHUD showMessage:[result objectForKey:@"message"] toView:self.view];

}else{

[MBProgressHUD showMessage:[result objectForKey:@"message"] toView:self.view];

NSLog(@"%@",[result objectForKey:@"message"]);

}

}];

# 2、Android项目集成SDK