

薪起程聚合支付收银台

IOS SDK 接口文档

文档历史

[illegible]

```

1 # Uncomment the next line to define a global platform for your project
2 platform :ios, '9.0'
3
4 target 'paySDKdemo' do
5   # Uncomment the next line if you're using Swift or would like to use dynamic frameworks
6   use_frameworks!
7   pod 'XQCPaySDK_Objc'
8   |
9   # Pods for paySDKdemo
10
11 end
12

```

第一步配置 Podfile

第二部步配置白名单 LSAApplicationQueriesSchemes

▼ LSAApplicationQueriesSchemes	⇅	Array	(4 items)
Item 0		String	wechat
Item 1		String	weixin
Item 2		String	alipay
Item 3		String	alipays

第三步配置微信 URL Types

▼ URL Types (1)

wechat			
No image specified	Identifier	wechat	URL Schemes
	Icon	None	wx88d30b27f1df7354
		Role	Editor
▶ Additional url type properties (0)			
+			

第四步配置依赖

AVFoundation.framework	Required ⇅
Accelerate.framework	Required ⇅
CoreVideo.framework	Required ⇅
CoreMedia.framework	Required ⇅
CoreLocation.framework	Required ⇅
AudioToolbox.framework	Required ⇅
Security.framework	Required ⇅
AddressBookUI.framework	Required ⇅
AddressBook.framework	Required ⇅
MessageUI.framework	Required ⇅
MobileCoreServices.framework	Required ⇅
CFNetwork.framework	Required ⇅
ImageIO.framework	Required ⇅
libz.1.2.5.tbd	Required ⇅
libc++.tbd	Required ⇅
CoreMotion.framework	Required ⇅
libsqlite3.tbd	Required ⇅
CoreTelephony.framework	Required ⇅
SystemConfiguration.framework	Required ⇅
WebKit.framework	Required ⇅
libstdc++.6.0.9.tbd	Required ⇅
CoreGraphics.framework	Required ⇅

第五步配置证书

XQC_payforapp.pfx

第六步导入

```
10 #import <XQCPaySDK_ObjC-umbrella.h>
11 @implementation XQCAppDelegate
12
13 - (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
14 {
15     // Override point for customization after application launch.
16     [[XQCPayManager defaultManager] setWechatKey:@"wx88d30b27f1df7354"];
17     [[XQCPayManager defaultManager] setConfig:@"https://openfulipayapi.bndxqc.com"];
18     [[XQCPayManager defaultManager] setMacSignKey:@"IZJ018XZCLJQBIO5GVZL4FNHGL8X74PZ" AgentKey:@"CEB9011C0946E23D4959C13EE5E1B785" MerchantId:@"111401100000212" PartnerId:@"XQC_payforapp" Password:@"xQcMy20Pwd19"
19         Notify:@"http://abc.com" AgentNo:@"1000001114" CompanyOpenId:@"123456789" UserOpenId:@"321"];
20     return YES;
21 }
```

第七步设置回调方法

```
1 // iOS9 之前用此方法
2 - (BOOL)application:(UIApplication *)application
3     openURL:(NSURL *)url
4     sourceApplication:(NSString *)sourceApplication
5     annotation:(id)annotation {
6     if (![XQCPayManager handler:url]) {
7         /* 处理其他非微信APP回调过来的URL */
8     }
9     return YES;
10 }
11
12 // iOS9 及以上用此方法
13 - (BOOL)application:(UIApplication *)app
14     openURL:(NSURL *)url
15     options:(NSDictionary<NSString *,id> *)options{
16     if (![XQCPayManager handler:url]) {
17         /* 处理其他非微信APP回调过来的URL */
18     }
19     return YES;
20 }
21
22 - (void)applicationWillEnterForeground:(UIApplication *)application
23 {
24     [XQCPayManager applicationWillEnterForeground:application];
25 }
26
27 R
```

第八步调用

```
// 订单名 订单id
XQCPayViewController *pay = [[XQCPayViewController alloc] initWithOrderTitle:@"薪起程测试" OrderId:[self getOrderId]];
[pay sendPrice:0.1]; // 设置金额
[XQCPayManager defaultManager].result = ^(ResponseModel * _Nonnull model) {
// 支付结果回调
    NSLog(@"model==>%@",model.message);
};
// pay.back = ^{
//// 这里是手动返回操作，如果需要则不需要实现
// };
//
```

唤起密码输入框，校验成功直接回调

```
[XQCPayManager showPasswordViewControllerResult:^(
    });
```