iOS工行支付SDK使用文档

最后修改日期：2019/5/15

1、场景介绍

场景1 接入工银e支付： 商家APP调用工行软件开发中心提供的支付SDK。如果用户安装了工行App，商家APP会跳转到App中完成支付，支付完后跳回到商家APP内，最后展示支付结果。如果用户没有安装工行App，会在SDK内启动支付页面进行支付，支付完成后，跳转回到商户界面。

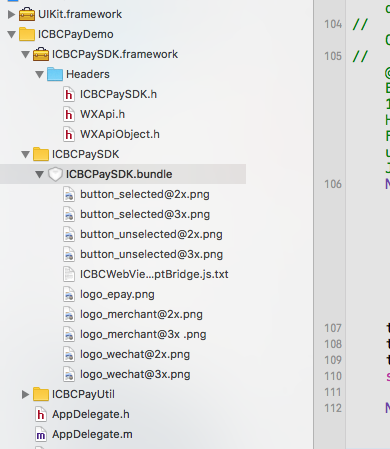
场景2 接入微信支付： 商家APP调用工行软件开发中心提供的支付SDK。如果用户安装了微信，并且微信版本支持微信支付，商家APP会唤起微信完成支付。支付完成后回到商家APP内，最后展示支付结果。

场景3 接入支付宝支付： 商家APP调用工行软件开发中心提供的支付SDK。如果用户安装了支付宝，商家APP会唤起支付宝完成支付,如果用户未安装支付宝客户端,则sdk会呼起html5页面进行支付,支付完成后回到商家APP内，最后展示支付结果。

场景4 调起sdk内的商户列表： 商家APP调用工行软件开发中心提供的支付SDK。商家不自行制作支付列表，直接调起工行sdk中的商户列表，该列表中最多展示工银e支付、微信支付和支付宝支付，三种支付方式。用户在该商户列表中选择支付方式，如果选择工银e支付则进入场景1，如果选择微信支付则进入场景2，如果选择支付宝支付则进入场景3。

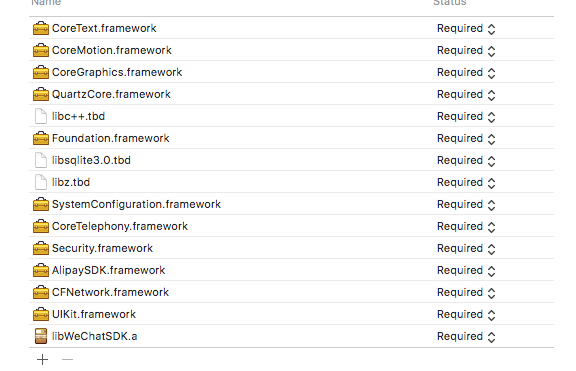
2、快速接入

2.1 接入ICBCPaySDK的静态库ICBCPaySDK.framework及资源包ICBCPaySDK.bundle，如下图所示。

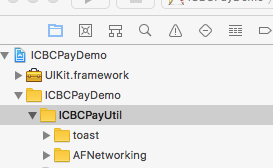


如工程有多个target，请检查build Phases相关文件是否成功引入。

另外，由于导入了微信项目与支付宝项目需要链接如下库否则报错。



2.2、接入工程所需第三方组件。主要为网络请求框架、toast组件。具体如下图所示。



具体地址如下可参照下表。

|  |
| --- |
| 网络请求框架 AFNetWorkworking 3.0版本  地址：https://github.com/AFNetworking/AFNetworking/  toast组件  uiview+toast.view：地址：<https://github.com/scalessec/Toast> |

2.3. 如有报错根据微信要求，在工程配置中的”Other Linker Flags”中加入”-Objc -all\_load”

\*不论选择下面哪种支付方式，请不要忘记阅读最后一章，注意事项。

3.接入工银e支付 代码示例

3.1 引入 ICBCPaySDK.h文件,创建应用单例。

|  |
| --- |
| #import < ICBCPaySDK/ ICBCPaySDK.h>  ICBCPaySDK \*shareSDK = [ICBCPaySDK sharedSdk];  shareSDK.sdkDelegate = self;  **shareSDK.urlListMain** **=** **@https://b2c.icbc.com.cn 请务必配置此生产地址**  **shareSDK.urlPortal = @****[https://mywap2.icbc.com.cn](https://mywap2.icbc.com.cn/) 请务必配置此生产地址** |

\*上面两个地址，在测试环境请配置模测地址(模测地址见第7章)，以便于在出现问题的时候查找日志。具体模测环境的搭建请联系数据中心的同事。

3.2传入支项目设置APPID

在Xcode中打开项目，设置项目属性中的URL Schemes 为子商户的Schemes。如图所示。

付页面

1. interfaceName

接口名

1. interfaceVersion

接口版本号

1. tranData

交易信息

1. merSignMsg

交易信息签名

（5） merCert

商户公钥文件信息

（6） urlScheme

应用包名，必须传入，否则跳转至工行App无法返回。

|  |
| --- |
| NSMutableDictionary \*testDic = [[NSMutableDictionary alloc] init];  testDic[@"urlScheme"] = urlScheme;  testDic[@"interfaceName"] =interfaceName;  testDic[@"interfaceVersion"] = interfaceVersion;  testDic[@"tranData"] = tranData;  testDic[@"merSignMsg"] = merSignMsg;  testDic[@"merCert"] = merCert;  [shareSDK presentICBCPaySDKInViewController:self andTraderInfo:testDic]; |

传入的self，为弹出支付页面的viewcontroller。SDK会对traderInfo的字段进行检查，发现字段为空，导致交易无法进行会进行toast提示。

3.3**代理**结果回调

支付完成或者失败时，会回调此代理方法，商户端，可根据返回的信息自行处理。

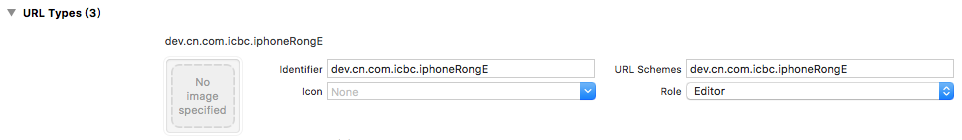
|  |
| --- |
| - (void)paymentEndwithResultDic:(NSDictionary\*)dic｛  NSLog(@"pass the pkpaymentAuthorizationstatus to the page");  [self.view makeToast:@“支付成功，商户端自行处理"];  ｝  字典如下格式如下：  tranCode:交易结果代码  tranMsg:交易结果信息  orderNo：订单号 |

3.4 如果跳转至工行app支付，需要在 AppDelegate.m中添加如下实现。(注意：只接入工银e支付实现如下代码，同时需要接入微信支付，这部分代码请参照接入支付列表中的5.4)

|  |
| --- |
| iOS9.0以下版本，请实现如下方法  - (BOOL)application:(UIApplication \*)application openURL:(NSURL \*)url sourceApplication:  (nullable NSString \*)sourceApplication annotation:(id)annotation {  if ([[url absoluteString] hasSuffix:@"ICBCB2CPAY"]) {  [[ICBCPaySDK sharedSdk] ICBCResultBackWithUrl:url];  }    return YES;  }  iOS9.0以上版本请实现如下方法。  - (BOOL)application:(UIApplication \*)application openURL:(NSURL \*)url options: (NSDictionary<NSString\*, id> \*)options{    if ([[url absoluteString] hasSuffix:@"ICBCB2CPAY"]) {  [[ICBCPaySDK sharedSdk] ICBCResultBackWithUrl:url];  }    return YES;  } |

3.5项目设置urlscheme

在Xcode中打开项目，设置项目属性中的URL Schemes 为子商户的应用包名。如图所示。



3.6 tranCode 对应关系

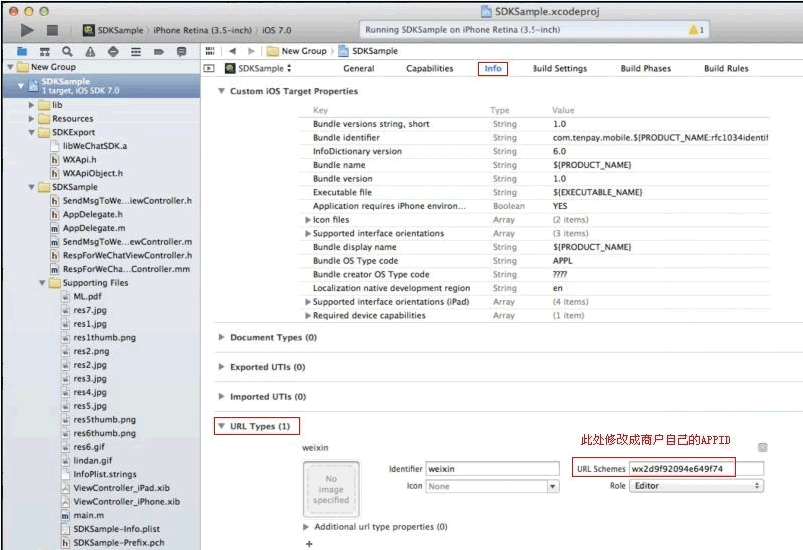
\*此处，由于系统原因在极少的情况下，可能出现trancode为空的情况，交易结果主要通过tranMsg体现，所以，请主要判断tranMsg，判断tranMsg中是否包含成功失败字样等等。

|  |
| --- |
| **tranCode与tranMsg的对应关系：**  **1--交易成功，已清算**  **2--交易失败**  **3--交易可疑**  **4--用户中止交易** |

4.接入微信支付 代码示例

4.1 项目设置APPID

商户在微信开发平台申请开发APP应用后，微信开放平台会生成APP的唯一标识APPID。在Xcode中打开项目，设置项目属性重的URL Schemes 为子商户的APPID。如图标红位置所示。



4.2 引入 ICBCPaySDK.h文件,创建应用单例。

|  |
| --- |
| #import < ICBCPaySDK/ ICBCPaySDK.h>  ICBCPaySDK \*shareSDK = [ICBCPaySDK sharedSdk];  shareSDK.sdkDelegate = self;  **shareSDK.urlListMain** **= @https://b2c.icbc.com.cn 请务必配置此生产地址** |

\*上面的地址，在测试环境请配置模测地址(模测地址见第7章)，以便于在出现问题的时候查找日志。具体模测环境的搭建请联系数据中心的同事。

4.3传入支持参数，弹出支付页面

1. ranData

交易信息

1. merSignMsg

交易信息签名

（3） merCert

商户公钥文件信息

（4）appId

受理商户的APPID

|  |
| --- |
| NSMutableDictionary \*testDic = [[NSMutableDictionary alloc]init];  testDic[@"tranData"] = tranData;  testDic[@"merSignMsg"] = merSignMsg;  testDic[@"merCert"] = merCert;  testDic[@"appId"] = appId;  [shareSDK presentWeChatInViewController:self andTraderInfo:testDic]; |

传入的self，为弹出支付页面的viewcontroller。SDK会对traderInfo的字段进行检查，发现字段为空，导致交易无法进行会进行toast提示。

唤起微信支付之前会判断是否安装了微信以及微信版本是否支持支付功能。无法唤起微信会进行toast提示。

4.4**代理**结果回调

支付完成或者失败时，会回调此方法，商户端，可根据返回的信息自行处理。

|  |
| --- |
| -(void)wxPayEndwithResultDic:(NSDictionary \*)dic{  NSString \*strMsg,\*strTitle = [NSString stringWithFormat:@"支付结果"];  int errorCode = (int)[dic objectForKey:@"errCode"];  NSString \*errStr = [dic objectForKey:@"errStr"];  switch (errorCode) {//?这个可能应该写成枚举  case 0:  strMsg = @"支付结果：成功！";  NSLog(@"支付成功－PaySuccess，retcode = %d", errorCode);  NSLog(@"商户自行处理");  break;    default:  strMsg = [NSString stringWithFormat:@"支付结果：失败！retcode = %d, retstr = %@", errorCode,errStr];  NSLog(@"错误，retcode = %d, retstr = %@", errorCode,errStr);  NSLog(@"商户自行处理");  break;  }  } |

4.5 errorCode 对应关系

|  |
| --- |
| **errorCode与tranMsg的对应关系：**  **1—成功**  **-1—错误**  **-2—用户取消** |

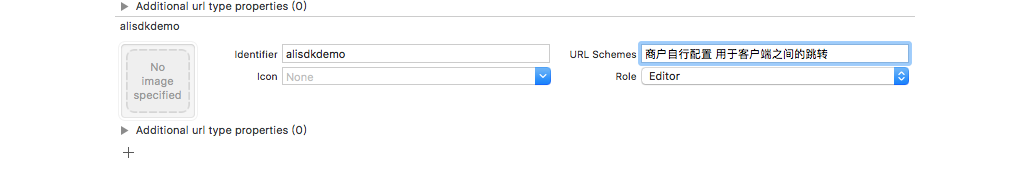
4.6 因为需要启动微信APP，需要在 AppDelegate.m中添加如下实现。(注意：只接入微信支付实现如下代码，同时需要接入微信支付，这部分代码请参照接入支付列表中的5.4)

|  |
| --- |
| iOS9.0以下版本，请实现如下方法    - (BOOL)application:(UIApplication \*)application openURL:(NSURL \*)url sourceApplication:(NSString \*)sourceApplication annotation:(id)annotation {  return [WXApi handleOpenURL:url delegate:[WXApiManager sharedManager]];  }  iOS9.0以上版本请实现如下方法。  - (BOOL)application:(UIApplication \*)application openURL:(NSURL \*)url options: (NSDictionary<NSString\*, id> \*)options{  return [WXApi handleOpenURL:url delegate:[WXApiManager sharedManager]];  } |

5.接入支付宝支付 代码示例

5.1 项目设置APPID

在Xcode中打开项目，设置项目属性中的URL Schemes 为子商户的Schemes。如图所示。这里为了统一，统一传入包名bundleid。



5.2 引入 ICBCPaySDK.h文件,创建应用单例。

|  |
| --- |
| #import < ICBCPaySDK/ ICBCPaySDK.h>  ICBCPaySDK \*shareSDK = [ICBCPaySDK sharedSdk];  shareSDK.sdkDelegate = self;  **shareSDK.urlListMain** **= @https://b2c.icbc.com.cn 请务必配置此生产地址** |

\*上面的地址，在测试环境请配置模测地址(模测地址见第7章)，以便于在出现问题的时候查找日志。具体模测环境的搭建请联系数据中心的同事。

5.3传入支持参数，弹出支付页面

1. tranData

交易信息

1. merSignMsg

交易信息签名

（3） merCert

商户公钥文件信息

（4） urlSchemes

受理商户的设置的scheme（这里为了统一，统一传入包名bundleid）

|  |
| --- |
| NSMutableDictionary \*testDic = [[NSMutableDictionary alloc]init];  testDic[@"tranData"] = tranData;  testDic[@"merSignMsg"] = merSignMsg;  testDic[@"merCert"] = merCert;  testDic[@"urlSchemes"] = schemes; 即商户的URL Schemes  [shareSDK presentAlipaySDKViewController:self andTraderInfo:testDic]; |

传入的self，为弹出支付页面的viewcontroller。SDK会对traderInfo的字段进行检查，发现字段为空，导致交易无法进行会进行toast提示。

5.4**代理**结果回调

支付完成或者失败时，会回调此方法，商户端，可根据返回的信息自行处理。

|  |
| --- |
| -(void) aliPayEndwithResultDic:(NSDictionary \*)dic{  NSLog(@"支付宝支付返回结果 :%@",dic);  // resultStatus，状态码，SDK里没对应信息，第一个文档里有提到：  // 9000 订单支付成功  // 8000 正在处理中  // 4000 订单支付失败  // 6001 用户中途取消  // 6002 网络连接出错  memo， 提示信息，比如状态码为6001时，memo就是“用户中途取消”。但是此提示语并不是太准确 最好自己定义成功或者失败的提示  result，订单信息，以及签名验证信息。  } |

5.5 因为需要启动支付宝APP，需要在 AppDelegate.m中添加如下实现。(注意：只接入支付宝支付实现如下代码，同时需要接入微信支付或者e支付，这部分代码请参照接入支付列表中的6.4)

|  |
| --- |
| ///9.0之前使用  - (BOOL)application:(UIApplication \*)application openURL:(NSURL \*)url sourceApplication:(nullable NSString \*)sourceApplication annotation:(id)annotation {  if ([url.host isEqualToString:@"safepay"]) {  ///支付宝结果处理  [[ICBCPaySDK sharedSdk] aliPayResultBackWithUrl:url];  }  return YES;  }  ///9.0以后使用  - (BOOL)application:(UIApplication \*)application openURL:(NSURL \*)url options:(NSDictionary<NSString\*, id> \*)options{    if ([url.host isEqualToString:@"safepay"]) {  ///支付宝结果处理  [[ICBCPaySDK sharedSdk] aliPayResultBackWithUrl:url];  }  return YES;  } |

7、注意事项

7.1 SDK付款有两种模式：如果外部存在手机银行客户端，则直接跳转到工行手机银行付款；不存在的场景下，在SDK内部进行H5支付。由于需要启动工行客户端，SDK在构件内调用了 canOpne方法，请在plist文件中，增加工行相应app的白名单。

com.icbc.iphoneclient 融e行

com.icbc.iphoneEChannel  融e联

com.icbc.iphone.emall 融e购

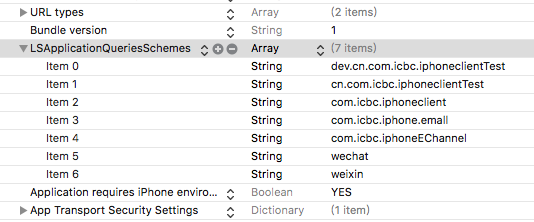
cn.com.icbc.eMallMobileClient 融e购

com.icbc.elife    工银e生活

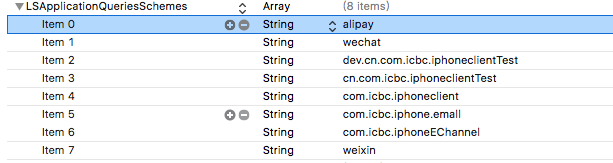
weixin 微信

alipay 支付宝

微信支付在付款时需要启动微信客户端，SDK在构件内调用了 canOpne方法，请在plist文件中，增加微信相应app的白名单。



支付宝支付在付款时需要启动客户端，SDK在构件内调用了 canOpne方法，请在plist文件中，增加支付宝相应app的白名单。



7.2 该版本SDK支持iOS 7.0以上版本。其他未尽事宜，可参照demo工程。

7.3 模测环境地址配置：

shareSDK.urlListMain = @https://b2c3.dccnet.com.cn

shareSDK.urlPortal = @[https://mywap2.dccnet.com.cn:447](https://mywap2.dccnet.com.cn:447/)