

# ios接入微信支付（小白都能看懂的微信支付）



作者 lyoniOS (/u/0714484ea84f) ✓ 已关注

2016.06.17 11:06 字数 734 阅读 10073 评论 146 喜欢 148  
(/u/0714484ea84f)

因为近期项目中需要接入微信支付功能，自己也爬了很多的坑，所以做了一下这边文章供大家学习参考，远离爬坑，文章主要讲到以下五部分：

一、 填写商户平台所需资料

二、 具体Demo代码@Github下载地址  
(<https://github.com/18814131164/WxPayDemo>)

本文为本人学习记录笔记，如需转载，请注明出处@iOS\_lyon  
(<http://www.jianshu.com/writer#/notebooks/4592932/notes/4284323/preview>)

## 填写商户平台所需资料

一、填写经营信息

提交资料

填写经营信息

填写商户信息

填写结算账户

确认提交

联系信息

\* 联系人姓名

请填写贵司微信支付业务联系人

\* 手机号码

+86

获取短信验证码

该号码非常重要，将接收与微信支付管理相关的重要信息

\* 短信验证码

请输入6位数字手机短信验证码

\* 常用邮箱

非常重要！用于接收微信支付的账号密码

APP应用信息

☐ APP下载地址

☒ APP页面截图

\* APP页面截图

APP页面截图应该包含：APP首页、APP版权信息页、APP商品展示页、APP支付方式页。[查看截图指引](#)  
截图最多上传5张，文件大小不超过2M，文件格式为bmp、png、jpeg、jpg或gif

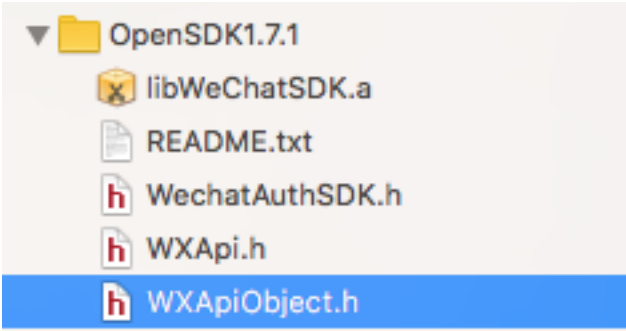
上传文件

@查看截图指引 (<http://kf.qq.com/faq/120911VrYVrA140522FNVZjl.html>)

下图选择不同的类目，所需要上传的资料也是有所不同的，下图拿其它为例子



1.将下载好的SDK导入项目,如下图



Paste\_Image.png

2.添加依赖库



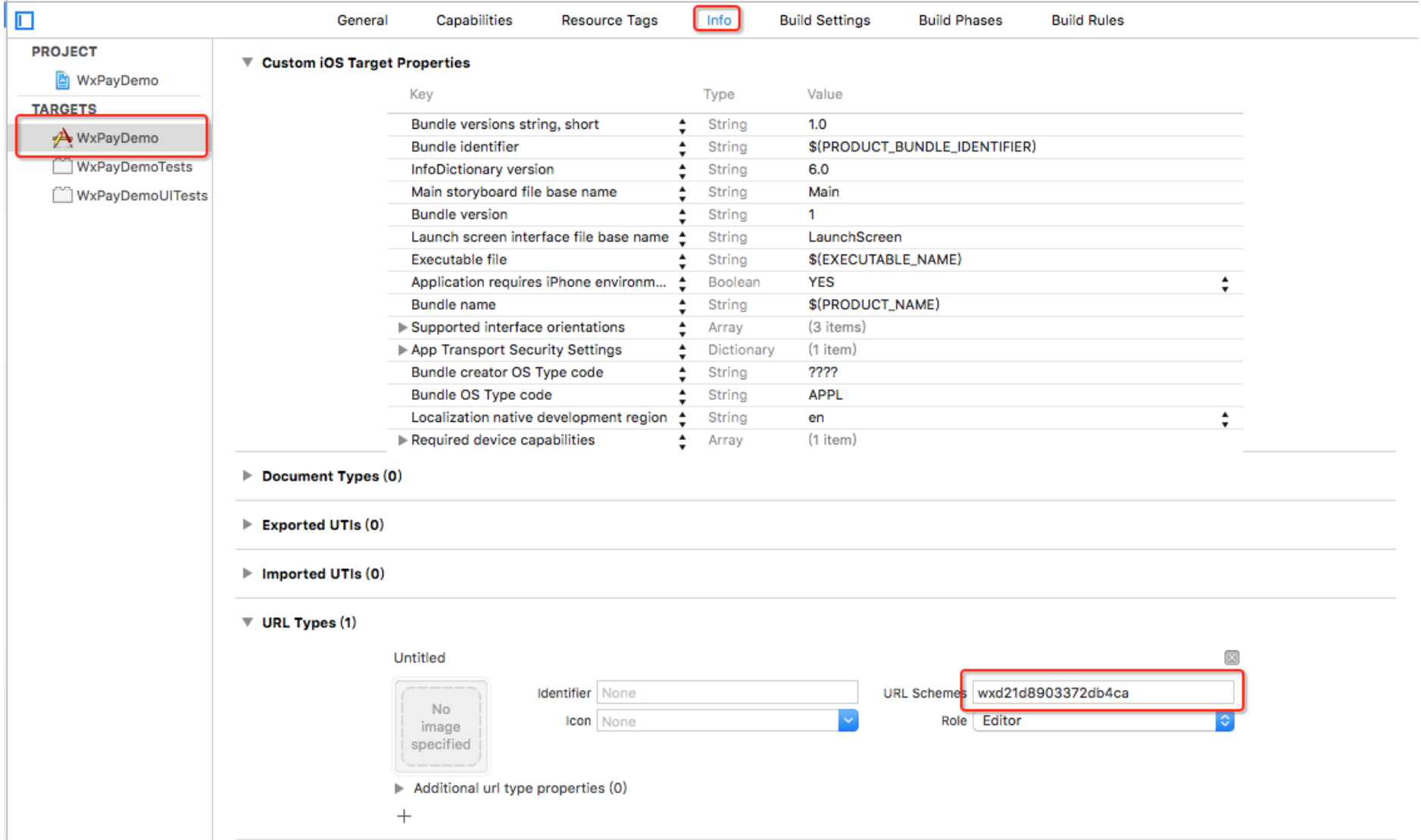
Paste\_Image.png

3.准备所需配置的数据

```
// 开放平台登录https://open.weixin.qq.com的开发者中心获取APPID
#define WX_APPID @"wxd21d890***2db4ca"
// 开放平台登录https://open.weixin.qq.com的开发者中心获取AppSecret。
#define WX_APPSecret @"fc32dfae9*****eb5f77dddd4ea5"
// 微信支付商户号
#define MCH_ID @"13536**702"

// 安全校验码（MD5）密钥，商户平台登录账户和密码登录http://pay.weixin.qq.com 平台设置的“API密钥”，为了安全，请设置为以数字和字母组成的32字符串。
#define WX_PartnerKey @"b5f9c901480*****0f4c6e659be0"
```

4.在Xcode中，选择你的工程设置项，选中“TARGETS”一栏，在“info”标签栏的“URL type”添加“URL scheme”为你所注册AppId（如下图所示）



Paste\_Image.png

配置好上述参数后就可以写代码了，具体可查看@Github下载地址  
(<https://github.com/18814131164/WxPayDemo>)

- PCH头文件: 在该文件配置好以下备注中要配置参数即可运行demo

```
#ifndef PrefixHeader_pch
#define PrefixHeader_pch

#pragma mark -
#pragma mark - 微信支付配置参数

// 开放平台登录https://open.weixin.qq.com的开发者中心获取APPID
#define WX_APPID @"wxd21d89033***b4ca"
// 开放平台登录https://open.weixin.qq.com的开发者中心获取AppSecret。
#define WX_APPSecret @"fc32dfae99bc67e***5f77dddd4ea5"
// 微信支付商户号
#define MCH_ID @"1353***702"
// 安全校验码（MD5）密钥，商户平台登录账户和密码登录http://pay.weixin.qq.com
// 平台设置的“API密钥”，为了安全，请设置为以数字和字母组成的32字符串。
#define WX_PartnerKey @"B6246A6D8***C730EEA0F78D3B461"

#pragma mark -
#pragma mark - 统一下单请求参数键值

// 应用id
#define WXAPPID @"appid"
// 商户号
#define WXMCHID @"mch_id"
// 随机字符串
#define WXNONCESTR @"nonce_str"
// 签名
#define WXSIGN @"sign"
// 商品描述
#define WXBODY @"body"
// 商户订单号
#define WXOUTTRADENO @"out_trade_no"
// 总金额
#define WXTOTALFEE @"total_fee"
// 终端IP
#define WXEQUIPMENTIP @"spbill_create_ip"
// 通知地址
#define WXNOTIFYURL @"notify_url"
// 交易类型
#define WXTRADETYPE @"trade_type"
// 预支付交易会话
#define WXPREPAYID @"prepay_id"

#pragma mark -
#pragma mark - 微信下单接口

// 微信统一下单接口连接
#define WXUNIFIEDORDERURL @"https://api.mch.weixin.qq.com/pay/unifiedorder"

#endif /* PrefixHeader_pch */
```

```
#import "AppDelegate.h"
#import "WXApiManager.h"

@interface AppDelegate ()

@end

@implementation AppDelegate

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // 注册微信
    [WXApi registerApp:WX_APPID withDescription:@"demo 2.0"];

    return YES;
}

- (BOOL)application:(UIApplication *)application handleOpenURL:(NSURL *)url
{
    return [WXApi handleOpenURL:url delegate:[WXApiManager sharedManager]];
}

- (BOOL)application:(UIApplication *)application openURL:(NSURL *)url sourceApplication:(NSString *)sourceApplication annotation:(id)annotation
{
    return [WXApi handleOpenURL:url delegate:[WXApiManager sharedManager]];
}

@end
```

微信支付回调管理类

```
#import "WXApiManager.h"

@implementation WXApiManager

#pragma mark - 单粒

+(instancetype)sharedManager {
    static dispatch_once_t onceToken;
    static WXApiManager *instance;
    dispatch_once(&onceToken, ^{
        instance = [[WXApiManager alloc] init];
    });
    return instance;
}

#pragma mark - WXApiDelegate

- (void)onResp:(BaseResp *)resp
{
    if([resp isKindOfClass:[PayResp class]]){

        //支付返回结果，实际支付结果需要去微信服务器端查询
        NSString *strMsg;

        switch (resp.errCode) {
            case WXSuccess:
                strMsg = @"支付结果：成功! ";
                NSLog(@"支付成功-PaySuccess, retcode = %d", resp.errCode);
                break;

            default:
                strMsg = [NSString stringWithFormat:@"支付结果：失败! retcode = %d, retstr = %@", resp.errCode,resp.errStr];
                NSLog(@"错误, retcode = %d, retstr = %@", resp.errCode,resp.errStr);
                break;
        }
    }
}

@end
```

步骤1：用户在商户APP中选择商品，提交订单，选择微信支付。

```
#import "ViewController.h"
#import "WXApiRequestHandler.h"
@interface ViewController ()

@end

@implementation ViewController

- (void)touchesBegan:(NSSet<UITouch *> *)touches withEvent:(UIEvent *)event
{
    // 发起支付
    [WXApiRequestHandler jumpToBizPay];
}

@end
```

步骤2：商户后台收到用户支付单，调用微信支付统一下单接口。参见【统一下单API ([https://pay.weixin.qq.com/wiki/doc/api/app/app.php?chapter=9\\_1](https://pay.weixin.qq.com/wiki/doc/api/app/app.php?chapter=9_1))】。

```
#import <Foundation/Foundation.h>
#import "WXApiObject.h"

@interface WXApiRequestHandler : NSObject

+ (NSString *)jumpToBizPay;

@end
```

```
#import "WXApi.h"
#import "WXApiRequestHandler.h"
#import "WXApiManager.h"
```

```
#import "DataMD5.h"
#import "XMLDictionary.h"
#import <AFNetworking.h>

#pragma mark - 用于获取设备ip地址

#include <ifaddrs.h>
#include <arpa/inet.h>

@implementation WXApiRequestHandler

#pragma mark - 产生随机字符串

//生成随机数算法 ,随机字符串, 不长于32位
//微信支付API接口协议中包含字段nonce_str, 主要保证签名不可预测。
//我们推荐生成随机数算法如下: 调用随机数函数生成, 将得到的值转换为字符串。

+ (NSString *)generateTradeNO {

    static int kNumber = 15;

    NSString *sourceStr = @"0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ";

    NSMutableString *resultStr = [[NSMutableString alloc] init];

    // srand函数是初始化随机数的种子, 为接下来的rand函数调用做准备。
    // time(0)函数返回某一特定时间的小数值。
    // 这条语句的意思就是初始化随机数种子, time函数是为了提高随机的质量 (也就是减少重复) 而使用的。

    // srand(time(0)) 就是给这个算法一个启动种子, 也就是算法的随机种子数, 有这个数以后才可以产生随机数,用1970.1.1至今的秒数, 初始化随机数种子。
    // Srand是种下随机种子数, 你每回种下的种子不一样, 用Rand得到的随机数就不一样。为了每回种下一个不一样的种子, 所以就选用Time(0), Time(0)是得到当前时时间值 (因为每时每刻时间是不一样的了) 。

    srand(time(0)); // 此行代码有警告:

    for (int i = 0; i < kNumber; i++) {

        unsigned index = rand() % [sourceStr length];

        NSString *oneStr = [sourceStr substringWithRange:NSMakeRange(index, 1)];

        [resultStr appendString:oneStr];
    }
    return resultStr;
}

#pragma mark - 获取设备ip地址

+ (NSString *)fetchIPAddress {
    NSString *address = @"error";
    struct ifaddrs *interfaces = NULL;
    struct ifaddrs *temp_addr = NULL;
    int success = 0;
    // retrieve the current interfaces - returns 0 on success
    success = getifaddrs(&interfaces);
    if (success == 0) {
        // Loop through linked list of interfaces
        temp_addr = interfaces;
        while(temp_addr != NULL) {
            if(temp_addr->ifa_addr->sa_family == AF_INET) {
                // Check if interface is en0 which is the wifi connection on the
                iPhone
                if([NSString stringWithUTF8String:temp_addr->ifa_name] isEqualToString:@"en0"]) {
                    // Get NSString from C String
                    address = [NSString stringWithUTF8String:inet_ntoa(((struct s
                    ockaddr_in *)temp_addr->ifa_addr)->sin_addr)];
                }
            }
            temp_addr = temp_addr->ifa_next;
        }
    }
    // Free memory
    freeifaddrs(interfaces);
    return address;
}

#pragma mark - Public Methods
// 发起微信支付
+ (void)jumpToWxPay
{

#pragma mark 客户端操作时候的代码 \ 但是这些步骤应该放在服务端操作
```



```
//=====
// V3&V4支付流程实现
// 注意:参数配置请查看服务器端Demo
// 更新时间: 2015年11月20日
//=====

// 交易类型
#define TRADE_TYPE @"APP"

// 交易结果通知网站此处用于测试, 随意填写, 正式使用时填写正确网站
#define NOTIFY_URL @"http://wxpay.weixin.qq.com/pub_v2/pay/notify.v2.php"

// 交易价格1表示0.01元, 10表示0.1元
#define PRICE @"1"


// 随机字符串变量 这里最好使用和安卓端一致的生成逻辑
NSString *nonce_str = [self generateTradeNO];


// 设备IP地址, 请再wifi环境下测试, 否则获取的ip地址为error, 正确格式应该是8.8.8.8
NSString *addressIP = [self fetchIPAddress];


// 随机产生订单号用于测试, 正式使用请换成你从自己服务器获取的订单号
NSString *orderno = [NSString stringWithFormat:@"%ld", time(0)];


// 获取SIGN签名
DataMD5 *data = [[DataMD5 alloc] initWithAppid:WX_APPID
                                     mch_id:MCH_ID
                               nonce_str:nonce_str
                             partner_id:WX_PartnerKey
                                   body:@"充值"
                              out_trade_no:orderno
                                total_fee:PRICE
                           spbill_create_ip:addressIP
                            notify_url:NOTIFY_URL
                              trade_type:TRADE_TYPE];


// 转换成xml字符串
NSString *string = [[data dic] XMLString];


AFHTTPSessionManager *session = [AFHTTPSessionManager manager];
//这里传入的xml字符串只是形似xml, 但不是正确是xml格式, 需要使用AF方法进行转义
session.responseSerializer = [[AFHTTPResponseSerializer alloc] init];
[session.requestSerializer setValue:@"text/xml; charset=utf-8" forHTTPHeaderField:@"Content-Type"];
[session.requestSerializer setValue:WXUNIFIEDORDERURL forHTTPHeaderField:@"SOAPAction"];
[session.requestSerializer setQueryStringSerializationWithBlock:^(NSString *(NSURLRequest *request, NSDictionary *parameters, NSError *__autoreleasing *error)) {
    return string;
}];
[session POST:WXUNIFIEDORDERURL parameters:string progress:nil success:^(NSURLSessionDataTask * _Nonnull task, id _Nullable responseObject) {

    // 输出XML数据
    NSString *responseString = [[NSString alloc] initWithData:responseObject
                                                         encoding:NSUTF8StringEncoding] ;
    // 将微信返回的xml数据解析转义成字典
    NSDictionary *dic = [NSDictionary dictionaryWithXMLString:responseString]
;

    // 判断返回的许可
    if ([[[dic objectForKey:@"result_code"] isEqualToString:@"SUCCESS"]
        &&[[dic objectForKey:@"return_code"] isEqualToString:@"SUCCESS"] ] ) {
        // 发起微信支付, 设置参数
        PayReq *request = [[PayReq alloc] init];
        request.openID = [dic objectForKey:WXAPPID];
        request.partnerId = [dic objectForKey:WXMCHID];
        request.prepayId= [dic objectForKey:WXPREPAYID];
        request.package = @"Sign=WXPay";
        request.nonceStr= [dic objectForKey:WXNONCESTR];
        // 将当前时间转化成时间戳
        NSDate *datenow = [NSDate date];
        NSString *timeSp = [NSString stringWithFormat:@"%ld", (long)[datenow
            timeIntervalSince1970]];
        UInt32 timeStamp =[timeSp intValue];
        request.timeStamp= timeStamp;
        // 签名加密
        DataMD5 *md5 = [[DataMD5 alloc] init];
        request.sign = [dic objectForKey:@"sign"];
        request.sign=[md5 createMD5SingForPay:request.openID
            partnerid:request.partnerId
```

```
                                prepayid:request.prepayId
                                package:request.package
                                noncestr:request.nonceStr
                                timestamp:request.timeStamp];

                                // 调用微信
                                [WXApi sendReq:request];
                                }

                                } failure:^(NSURLSessionDataTask * _Nullable task, NSError * _Nonnull error)
{

                                NSLog(@"%@",error);
                                }];

#pragma mark 服务端操作微信支付 / 上述客户端操作可以忽略（仅供参考）没办法，靠后台还不如靠自己，先自己了解客户端实现支付的操作

                                NSMutableDictionary *params = [NSMutableDictionary dictionary];
                                params[WXTOTALFEE] = @"1";
                                params[WXEQUIPMENTIP] = [self fetchIPAddress];

                                AFHTTPSessionManager *session = [AFHTTPSessionManager manager];
                                [session POST:URLSTRING parameters:params progress:nil success:^(NSURLSession
                                DataTask * _Nonnull task, id _Nullable responseObject) {

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

                                // 判断返回的许可
                                if ([[responseObject objectForKey:@"result_code"] isEqualToString:@"SUCCE
                                SS"]
                                &&[[responseObject objectForKey:@"return_code"] isEqualToString:@"SUC
                                CESS"] ) {

                                // 发起微信支付，设置参数
                                PayReq *request = [[PayReq alloc] init];
                                request.openID = [responseObject objectForKey:WXAPPID];
                                request.partnerId = [responseObject objectForKey:WXMCHID];
                                request.prepayId = [responseObject objectForKey:WXPREPAYID];
                                request.package = @"Sign=WXPay";
                                request.nonceStr = [responseObject objectForKey:WXNONCESTR];
                                request.timeStamp = [[responseObject objectForKey:@"timestamp"] int
                                Value];

                                request.sign = [responseObject objectForKey:@"sign"];
                                // 调用微信支付
                                [WXApi sendReq:request];
                                }else{
                                // 显示错误信息
                                [LyonKeyWindow.rootViewController showHint:responseObject[@"err_code_
                                des"]];
                                }

                                } failure:^(NSURLSessionDataTask * _Nullable task, NSError * _Nonnull error)
{

                                NSLog(@"%@",error);
                                }];

                                }

                                @end
```

以下皆是客户端的签名方法类

```
#import "DataMD5.h"
#import <CommonCrypto/CommonDigest.h>
@interface DataMD5()

@property (nonatomic,strong) NSString *appid;
@property (nonatomic,strong) NSString *mch_id;
@property (nonatomic,strong) NSString *nonce_str;
@property (nonatomic,strong) NSString *partnerkey;
@property (nonatomic,strong) NSString *body;
@property (nonatomic,strong) NSString *out_trade_no;
@property (nonatomic,strong) NSString *total_fee;
@property (nonatomic,strong) NSString *spbill_create_ip;
@property (nonatomic,strong) NSString *notify_url;
@property (nonatomic,strong) NSString *trade_type;

@end

@implementation DataMD5

#pragma makr - 懒加载
```



```

- (NSMutableDictionary *)dic
{
    if (!_dic) {
        _dic = [NSMutableDictionary dictionary];
    }
    return _dic;
}

#pragma mark - Config

-(instancetype)initWithAppid:(NSString *)appid_key
                      mch_id:(NSString *)mch_id_key
                      nonce_str:(NSString *)noce_str_key
                      partner_id:(NSString *)partner_id
                      body:(NSString *)body_key
                      out_trade_no : (NSString *)out_trade_no_key
                      total_fee:(NSString *)total_fee_key
                      spbill_create_ip:(NSString *)spbill_create_ip_key
                      notify_url:(NSString *)notify_url_key
                      trade_type:(NSString *)trade_type_key
{
    if (self = [super init]) {

        _appid          = appid_key;
        _mch_id          = mch_id_key;
        _nonce_str       = noce_str_key;
        _partnerkey       = partner_id;
        _body             = body_key;
        _out_trade_no    = out_trade_no_key;
        _total_fee        = total_fee_key;
        _spbill_create_ip = spbill_create_ip_key;
        _notify_url       = notify_url_key;
        _trade_type       = trade_type_key;

        [self.dic setValue:_appid forKey:WXAPPID];
        [self.dic setValue:_mch_id forKey:WXMCHID];
        [self.dic setValue:_nonce_str forKey:WXNONCESTR];
        [self.dic setValue:_body forKey:WXBODY];
        [self.dic setValue:_out_trade_no forKey:WXOUTTRADENO];
        [self.dic setValue:_total_fee forKey:WXTOTALFEE];
        [self.dic setValue:_spbill_create_ip forKey:WXEQUIPMENTIP];
        [self.dic setValue:_notify_url forKey:WXNOTIFYURL];
        [self.dic setValue:_trade_type forKey:WXTRADETYPE];

        [self createMd5Sign:self.dic];
    }
    return self;
}

//创建签名
//签名算法
//签名生成的通用步骤如下：
//第一步， 设所有发送或者接收到的数据为集合M， 将集合M内非空参数值的参数按照参数名ASCII码从小到大排序（字典序）， 使用URL键值对的格式（即key1=value1&key2=value2...）拼接成字符串stringA。
//特别注意以下重要规则：
//◆ 参数名ASCII码从小到大排序（字典序）；
//◆ 如果参数的值为空不参与签名；
//◆ 参数名区分大小写；
//◆ 验证调用返回或微信主动通知签名时， 传送的sign参数不参与签名， 将生成的签名与该sign值作校验。
//◆ 微信接口可能增加字段， 验证签名时必须支持增加的扩展字段
//第二步， 在stringA最后拼接上key得到stringSignTemp字符串， 并对stringSignTemp进行MD5运算， 再将得到的字符串所有字符转换为大写， 得到sign值signValue。
//key设置路径： 微信商户平台(pay.weixin.qq.com)-->账户设置-->API安全-->密钥设置

-(void)createMd5Sign:(NSMutableDictionary*)dict
{
    NSMutableString *contentString  = [NSMutableString string];

    NSArray *keys = [dict allKeys];

    //按字母顺序排序
    NSArray *sortedArray = [keys sortedArrayUsingComparator:^(NSComparisonResult(i
d obj1, id obj2) {
        return [obj1 compare:obj2 options:NSNumericSearch];
    }];

    //拼接字符串
    for (NSString *categoryId in sortedArray) {

        if (    ![dict objectForKey:categoryId] isEqualToString:@""]
            && ![dict objectForKey:categoryId] isEqualToString:@"sign"]
            && ![dict objectForKey:categoryId] isEqualToString:@"key"]
        )
        {
            [contentString appendFormat:@"%s=%s", categoryId, [dict objectForKey
:categoryId]];

```

```

    }
}
//添加商户密钥key字段
[contentString appendFormat:@"%key=%@",_partnerkey];

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

//MD5 获取Sign签名
NSString *md5Sign =[self md5:contentString];

//
[self.dic setValue:md5Sign forKey:@"sign"];

}

//创建发起支付时的sige签名

-(NSString *)createMD5SingForPay:(NSString *)appid_key partnerid:(NSString *)part
nerid_key prepayid:(NSString *)prepayid_key package:(NSString *)package_key nonce
str:(NSString *)noncestr_key timestamp:(UInt32)timestamp_key{
    NSMutableDictionary *signParams = [NSMutableDictionary dictionary];
    [signParams setObject:appid_key forKey:@"appid"];
    [signParams setObject:noncestr_key forKey:@"noncestr"];
    [signParams setObject:package_key forKey:@"package"];
    [signParams setObject:partnerid_key forKey:@"partnerid"];
    [signParams setObject:prepayid_key forKey:@"prepayid"];
    [signParams setObject:[NSString stringWithFormat:@"%u",(unsigned int)timeスタ
mp_key] forKey:@"timestamp"];

    NSMutableString *contentString = [NSMutableString string];
    NSArray *keys = [signParams allKeys];
    //按字母顺序排序
    NSArray *sortedArray = [keys sortedArrayUsingComparator:^NSComparisonResult(i
d obj1, id obj2) {
        return [obj1 compare:obj2 options:NSNumericSearch];
    }];
    //拼接字符串
    for (NSString *categoryId in sortedArray) {
        if (    ![signParams objectForKey:categoryId] isEqualToString:@""]
            && ![signParams objectForKey:categoryId] isEqualToString:@"sign"]
            && ![signParams objectForKey:categoryId] isEqualToString:@"key"]
            )
        {
            [contentString appendFormat:@"% %@=%@", categoryId, [signParams object
ForKey:categoryId]];
        }
    }
    //添加商户密钥key字段
#warning 注意此处一定要添加上商户密钥
    [contentString appendFormat:@"%key=%@", WX_PartnerKey];
    NSString *result = [self md5:contentString];

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

// MD5加密算法
-(NSString *) md5:(NSString *)str
{
    const char *cStr = [str UTF8String];
    //加密规则，因为逗比微信没有出微信支付demo，这里加密规则是参照安卓demo来得
    unsigned char result[16]= "0123456789abcdef";
    CC_MD5(cStr, (CC_LONG)strlen(cStr), result);
    //这里的x是小写则产生的md5也是小写，x是大写则md5是大写，这里只能用大写，逗比微信的大小写验
    证很逗
    return [NSString stringWithFormat:
        @"%02X%02X%02X%02X%02X%02X%02X%02X%02X%02X%02X%02X%02X%02X%02X%02X",
        result[0], result[1], result[2], result[3],
        result[4], result[5], result[6], result[7],
        result[8], result[9], result[10], result[11],
        result[12], result[13], result[14], result[15]
        ];
}

@end

```



如果觉得我的文章对您有用，请随意打赏。您的支持将鼓励我继续创作！

赞赏支持



♡ 喜欢

148



更多分享

(http://cwb.assets.jianshu.io/notes/images/4284260/)



写下你的评论...

146条评论

只看作者

按喜欢排序 按时间正序 按时间倒序



瞳若点星 (/u/d6bee3702033)

2楼 · 2016.06.17 17:52

(/u/d6bee3702033)

刚做到这块，正需要，谢谢分享



赞



回复

lyoniOS (/u/0714484ea84f): @瞳若点星 (/users/d6bee3702033) 共勉！！

2016.06.17 19:09 回复

瞳若点星 (/u/d6bee3702033): @iOS\_lyon (/users/0714484ea84f) 楼主有放到github上吗？想参考一下。

2016.06.23 09:34 回复

lyoniOS (/u/0714484ea84f): @瞳若点星 (/users/d6bee3702033) 这个没有放到github，如果你有需要，可以邮箱发你一份自己测试的demo

2016.06.23 10:12 回复

0263401535b9 (/u/0263401535b9): @iOSlyon (/users/0714484ea84f) 可以给我邮箱 发一份demo吗？1399149518@qq.com

2016.07.12 17:12 回复

代码爱好者 (/u/63fd59f1da1c): @iOSlyon (/users/0714484ea84f) 可以发我一份吗？1290993310@qq.com

2016.07.17 22:32 回复

lyoniOS (/u/0714484ea84f): @代码爱好者 (/users/63fd59f1da1c) 文中有Demo的下载链接，你可以直接下载

2016.07.18 08:47 回复



添加新评论

收起



巴赫海兹 (/u/12b1371bab88)

3楼 · 2016.06.28 17:25

(/u/12b1371bab88)

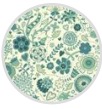
楼主，现在要在支付时添加一些字段以便于微信返回数据到服务器上时便于获取，怎么做？



赞



回复



白金唱片 (/u/177e2f7f8f57)  
4楼 · 2016.07.11 11:20  
(/u/177e2f7f8f57)  
953520170@qq.com 可以发一份demo 借鉴一下嘛 谢谢楼主

👍 赞    💬 回复

lyoniOS (/u/0714484ea84f): 已发  
2016.07.11 11:26    💬 回复

Apples8023 (/u/5c8b6032e786): @iOSlyon (/users/0714484ea84f) 335983653@qq.com  
楼主，求发一份demo  
2016.07.13 11:42    💬 回复

Coke26 (/u/115af0776bec): 求楼主大大的demo 365754921@qq.com谢谢  
2016.11.10 16:16    💬 回复

✍️ 添加新评论



Viterbi (/u/1e9bc41934d4)  
5楼 · 2016.07.11 15:25  
(/u/1e9bc41934d4)  
楼主，照着你的代码 老是只出现一个确定按钮 不知道您出现过没有

👍 赞    💬 回复

lyoniOS (/u/0714484ea84f): @Viterbi (/users/1e9bc41934d4) 你可以使用微信平台上面的签名工具看看你最终的签名跟微信平台上得到的签名是否一致，一般都是签名问题。还有你的url scheme是否有填写  
2016.07.11 16:02    💬 回复

✍️ 添加新评论



lukyy (/u/437734a25841)  
6楼 · 2016.07.12 11:01  
(/u/437734a25841)  
1531073749@qq.com 可以发一份demo 现在需要 谢谢楼主

👍 赞    💬 回复

lyoniOS (/u/0714484ea84f): @lukyy (/users/437734a25841) 已发，请查收  
2016.07.12 11:04    💬 回复

lukyy (/u/437734a25841): @iOSlyon (/users/0714484ea84f) 收到了，非常感谢🙏🙏，写的很详细  
2016.07.12 11:10    💬 回复

✍️ 添加新评论



詹姆斯洪a尼古拉斯利 (/u/53dfe175a39f)  
7楼 · 2016.07.12 17:01  
(/u/53dfe175a39f)  
前几天刚做完微信和支付宝支付的嵌入发现支付宝官方文档指导性的描述很少，前端还好有现成demo，但后台返回签名字段总出问题一直显示ALI系列的问题，最后发现是参数排序有问题，需按照前端demo里的参数排序一样。接下来就是最恶心的微信，文档描述居然是错的，官方文档说所有参数按照驼峰式规则书写，但实际上是需要把所有参数字段都小写才可以

👍 赞    💬 回复



0263401535b9 (/u/0263401535b9)  
8楼 · 2016.07.12 17:10  
(/u/0263401535b9)  
已赞 楼主可以发我一份demo吗？

👍 赞    💬 回复

lyoniOS (/u/0714484ea84f): @0263401535b9 (/users/0263401535b9) 文章里面有demo的下载地址，你可以直接在上面下载

2016.07.12 17:11    💬 回复

0263401535b9 (/u/0263401535b9): 平台设置的“API密钥”在商户平台查不到？请问在哪里可以查到？？

2016.07.12 18:31    💬 回复

lyoniOS (/u/0714484ea84f): @0263401535b9 (/users/0263401535b9) 这个是在商户平台中设置的，设置完之后就拿来用。只能在商户平台

2016.07.12 19:01    💬 回复

✎ 添加新评论



进击的小白 (/u/21593bb673fb)  
9楼 · 2016.07.12 17:32

(/u/21593bb673fb)  
大神给我发一份呗，谢谢你啦，github上下载的总是提示错误 179434316@qq.com

👍 赞    💬 回复

lyoniOS (/u/0714484ea84f): @进击的小白 (/users/21593bb673fb) github上下载的提示错误是指什么意思。我去看看

2016.07.12 17:34    💬 回复

进击的小白 (/u/21593bb673fb): @iOSlyon (/users/0714484ea84f) linker command failed with exit code 1 (use -v to see invocation) 方便加您一个QQ吗

2016.07.12 17:37    💬 回复

lyoniOS (/u/0714484ea84f): @进击的小白 (/users/21593bb673fb) QQ也不怎么上，有什么问题你可以在这反馈给我，我去进行修改

2016.07.12 17:40    💬 回复

✎ 添加新评论    |    还有3条评论， 展开查看



鹰凌天下 (/u/3c2aabaa015f)  
10楼 · 2016.07.12 22:51

(/u/3c2aabaa015f)  
app内置微信支付，不使用apple的内购，能通过审核？

👍 赞    💬 回复

lyoniOS (/u/0714484ea84f): @鹰凌天下 (/users/3c2aabaa015f) 可以的，内购只是在一些情况下才必须使用内购

2016.07.12 23:44    💬 回复

✎ 添加新评论



15356163496 (/u/a0fd05c33e8c)  
11楼 · 2016.07.12 23:18

(/u/a0fd05c33e8c)  
735479094@qq.com 可以发一份demo 借鉴一下嘛 谢谢楼主

👍 赞    💬 回复

lyoniOS (/u/0714484ea84f): Demo可以在文章中的链接中下载

2016.07.12 23:50    💬 回复

✎ 添加新评论





XF1994 (/u/946164958331)  
12楼 · 2016.07.13 11:46  
(/u/946164958331)  
给可以发个Demo, 273559753@qq.com,谢谢

👍 赞    💬 回复

lyoniOS (/u/0714484ea84f): @\_1994\_ (/users/946164958331) 文章中有demo的下载链接, 你可以上去下载  
2016.07.13 11:48    💬 回复

✎ 添加新评论



伊乐兔 (/u/055e1009ae88)  
13楼 · 2016.07.13 14:50  
(/u/055e1009ae88)  
楼主, 请问你有没有遇到这样一个问题 就是支付完成之后 页面右边会出现 按钮可以返回 app  
如果点击 回调正常 不点击 回调不起作用

👍 赞    💬 回复

0263401535b9 (/u/0263401535b9): @lilejianshu (/users/055e1009ae88) 这个问题解决了吗? ?  
2016.07.14 14:51    💬 回复

伊乐兔 (/u/055e1009ae88): @0263401535b9 (/users/0263401535b9)  
没有, 我只是写了一个支付的Demo 写的都是死数据 不知道为什么会出现这个问题  
你有解决的办法吗 或者说 为什么会引起这个问题  
2016.07.14 18:07    💬 回复

lyoniOS (/u/0714484ea84f): @lilejianshu (/users/055e1009ae88) 介不介意把你遇到问题的Demo发给我, 我看下是什么问题  
2016.07.14 18:10    💬 回复

JhonBook (/u/f1e0f94abc3d): @iOSlyon (/users/0714484ea84f) 遇到了和13楼同志同样的问题, 微信支付完成, 竟然不能自动跳转回自己的应用  
2016.08.17 10:32    💬 回复

lyoniOS (/u/0714484ea84f): @f1e0f94abc3d (/users/f1e0f94abc3d) 设置URL scheme了吗  
2016.08.17 10:35    💬 回复

✎ 添加新评论    |    收起



12343954 (/u/20d022cb2cf4)  
14楼 · 2016.07.15 16:02  
(/u/20d022cb2cf4)  
你好, 添加你的代码, 但是PrefixHeader.pch里定义的宏, 在ViewController.m找不到, 什么原因呢? 谢谢

👍 赞    💬 回复


12343954 (/u/20d022cb2cf4): 也没见这个文件在项目中引用  
2016.07.15 16:03    💬 回复

lyoniOS (/u/0714484ea84f): @12343954 (/users/20d022cb2cf4) pch是在build setting那里设置的  
2016.07.15 16:24    💬 回复

12343954 (/u/20d022cb2cf4): @iOSlyon (/users/0714484ea84f) 哦。呵呵。iOS小白了。我去鼓捣鼓捣  
2016.07.15 16:42    💬 回复





12343954 (/u/20d022cb2cf4): @iOSlyon (/users/0714484ea84f) 请问，这些参数为什么要放到预编译里？以后这些参数可能会变的，比如对多个商家交易


2016.07.15 16:44     回复

12343954 (/u/20d022cb2cf4): 你好，我有个app，用的是科技公司的appid，里面有多个商户的商品（微信商户ID），在付款的时候需要付给不同的商家，请问能做到吗？我的调试信息是：“商户号mch\_id与appid不匹配”

谢谢


2016.07.15 17:38     回复

 添加新评论    |    收起


 心海蓝 (/u/7b929ffe1108)  
15楼 · 2016.08.02 09:58  
(/u/7b929ffe1108)  
楼主，求一份儿demo，速度啊，救急


 赞     回复


lyoniOS (/u/0714484ea84f): @心海蓝 (/users/7b929ffe1108) 文中有下载链接，你可以在上面下载

2016.08.02 11:39     回复

心海蓝 (/u/7b929ffe1108): @iOSlyon (/users/0714484ea84f) 已经做好了，谢谢

2016.08.02 11:39     回复

 添加新评论

 白行夜 (/u/c3281e46cac4)  
16楼 · 2016.08.11 11:37  
(/u/c3281e46cac4)  
楼主，app怎么调取微信客户端进行支付？

 赞     回复

被以下专题收入，发现更多相似内容

+

我的专题

iOS

iOS开发实战

首页投稿

程序员

牛叉的demo

iOS 开发成神之路

iOS Dev...

iOS进阶

iOS支付资料

iOS OC ...

iOS

iOS开发文集

iOS & O...

键盘上的鼓手

面试

加载更多...