# 准备工作

## 微信公众平台用户授权配置说明：

在微信公众号请求用户网页授权之前，开发者需要先到公众平台官网中的“开发 - 接口权限 - 网页服务 - 网页帐号 - 网页授权获取用户基本信息”的配置选项中，修改授权回调域名。请注意，这里填写的是域名（是一个字符串），而不是URL，因此请勿加 http:// 等协议头；

授权回调域名配置规范为全域名，比如需要网页授权的域名为：www.qq.com，配置以后此域名下面的页面http://www.qq.com/music.html 、 http://www.qq.com/login.html 都可以进行OAuth2.0鉴权。但http://pay.qq.com 、 http://music.qq.com 、 http://qq.com 无法进行OAuth2.0鉴权





如果授权地址配置不对的话，用户授权回调时会报如下错误：



## 微信公众平台支付授权目录配置：

上线前，商户需要提供唤起微信支付地址给到汇付，由汇付工作人员配置支付授权目录

例如唤起微信支付地址为：<https://test.xxx.com/api/hfpayweb/wxindex>

则配置支付授权目录：https://test.xxx.com/api/hfpayweb/

如果支付授权目录配置不对话，支付唤起时会报如下错误：



# 开发步骤

**流程：用户授权->获取openId->调汇付支付->唤起微信支付**

## 拼装生成二维码的URL地址

根据如下地址生成二维码：

https://open.weixin.qq.com/connect/oauth2/authorize?appid=APPID&redirect\_uri=REDIRECT\_URI&response\_type=code&scope=SCOPE&state=STATE#wechat\_redirect

该URL地址实际上是微信授权地址，用户扫码后直接调起微信授权，用户同意授权后，页面直接跳转至redirect\_uri。

URL地址示例： <https://open.weixin.qq.com/connect/oauth2/authorize?appid=wx41c0xxxcxxxddxxx&redirect_uri=https%3A%2F%2Fhfpayts.cloudpnr.com%2Fapi%2Fhfpayweb%2FauthCallBack%2FappPay?response_type=code&scope=snsapi_base&state=f5dec61d-3b35-4b6b-a972-5acdf42f751b#wechat_redirect>

注: scope=snsapi\_base时用户默认授权，不会出现授权页面，用户授权时是无感的。

微信授权参数说明：

|  |  |  |
| --- | --- | --- |
| **参数** | **是否必须** | **说明** |
| appid | 是 | 公众号的唯一标识 |
| redirect\_uri | 是 | 授权后重定向的回调链接地址， 请使用 urlEncode 对链接进行处理 |
| response\_type | 是 | 返回类型，请填写code |
| scope | 是 | 应用授权作用域，snsapi\_base （不弹出授权页面，直接跳转，只能获取用户openid），snsapi\_userinfo （弹出授权页面，可通过openid拿到昵称、性别、所在地。并且， 即使在未关注的情况下，只要用户授权，也能获取其信息 ） |
| state | 否 | 重定向后会带上state参数，开发者可以填写a-zA-Z0-9的参数值，最多128字节 |
| #wechat\_redirect | 是 | 无论直接打开还是做页面302重定向时候，必须带此参数 |

## 通过微信回调返回的code换取openid

后台请求以下链接获取openid：

https://api.weixin.qq.com/sns/oauth2/access\_token?appid=APPID&secret=SECRET&code=CODE&grant\_type=authorization\_code

请求参数说明：

|  |  |  |
| --- | --- | --- |
| **参数** | **是否必须** | **说明** |
| appid | 是 | 公众号的唯一标识 |
| secret | 是 | 公众号的appsecret |
| code | 是 | 填写第一步获取的code参数 |
| grant\_type | 是 | 填写为authorization\_code |

返回参数说明：

|  |  |
| --- | --- |
| **参数** | **描述** |
| access\_token | 网页授权接口调用凭证,注意：此access\_token与基础支持的access\_token不同 |
| expires\_in | access\_token接口调用凭证超时时间，单位（秒） |
| refresh\_token | 用户刷新access\_token |
| openid | 用户唯一标识，请注意，在未关注公众号时，用户访问公众号的网页，也会产生一个用户和公众号唯一的OpenID |
| scope | 用户授权的作用域，使用逗号（,）分隔 |

正确时返回的JSON数据包如下：

{

 "access\_token":"ACCESS\_TOKEN",

 "expires\_in":7200,

 "refresh\_token":"REFRESH\_TOKEN",

 "openid":"OPENID",

 "scope":"SCOPE"

}

错误时微信会返回JSON数据包如下（示例为Code无效错误）:

{"errcode":40029,"errmsg":"invalid code"}

## 调支付管家APP支付接口获取pay\_info

接口详情参考api文档： <https://api.cloudpnr.com/hfpay/wxAliPay.html#app>

主要参数说明：

app\_id = 公众号唯一标识

buyer\_id = 上一步骤取得的openid

app\_pay\_type = 05（微信公众号）

返回参数pay\_info示例：

{"appId":"wx3c9719ebab09d0da","timeStamp":"1593509842","nonceStr":"9b9f2f3261f140f697a9c613046f7bc3","package":"prepay\_id=wx30173722025461c067c6700b1777789400","signType":"RSA","paySign":"QFC6/R3/hz5CTwY5J3mlrKJ/bHihQTz183ikIMFv0NrbUgjgal6FYYD7gnvJTbDoO3dWeBY6Y2LrAmd/xWdpS9KJTLHTdUSPeC5kw6ry44EUB28YprjRlL8Znbx9KSdOKoatVJ1PhwK5nuEFMC/qAJWb0WEW61NziohsyxJIkAclorOAyhznXzsRO7bjDwFun2t/fI3friEGnQ34fbiNMPHWB2TPkrLYutI7+7JTg42Lf3KF8jF3prHYi/fu3A3JYNRaHYyvFgkbTddY7i8snBR+XPcVAejsKTgqczEXzJW8DF6pjbSK7LmrI9Wg+lVWfkXqQPF/Y0Yu5uF9qrFHcw=="}

## 微信支付界面唤起

##### 微信公众号唤起支付

可参照微信官网：<https://pay.weixin.qq.com/wiki/doc/api/external/jsapi.php?chapter=7_7&index=6>

js初始化微信浏览器的WeixinJSBridge对象，唤起微信支付界面，并定义不同返回码的事件，示例：

|  |
| --- |
| /\*\* \* 确保微信JS加载完成 \* **@param** callback \*/**function** wechatReady(callback) { **if** (**typeof** WeixinJSBridge == "undefined") { **if** (document.addEventListener) { document.addEventListener('WeixinJSBridgeReady', callback, **false**); } **else** **if** (document.attachEvent) { document.attachEvent('WeixinJSBridgeReady', callback); document.attachEvent('onWeixinJSBridgeReady', callback); } } **else** { callback && callback(); }}/\*\* \* JS 调起微信支付 \* **@param** payInfo \*/**function** wechatPay(payInfo) {// alert('wechatPay'); wechatReady(**function** () { **var** jsonMsg = payInfo.replace(/\"/g, "\""); **var** payInfoJson = JSON.parse(jsonMsg); WeixinJSBridge.invoke('getBrandWCPayRequest', { "appId": payInfoJson['appId'], "nonceStr": payInfoJson['nonceStr'], "package": payInfoJson['package'], "paySign": payInfoJson['paySign'], "signType": payInfoJson['signType'], "timeStamp": payInfoJson['timeStamp'] }, **function** (res) {// alert('wechatPay callback:' + res.err\_msg); **switch** (res.err\_msg) { **case** "get\_brand\_wcpay\_request:ok": // 支付成功 paySuccess(); // 支付成功函数 **break**; **case** "get\_brand\_wcpay\_request:cancel": // 支付过程中用户取消 **break**; **case** "get\_brand\_wcpay\_request:fail": // 支付失败 **default**: payFail();// 支付失败函数 **break**; } }); });}  |

##### 微信小程序唤起支付

小程序支付利用wx.requestPayment发起支付，示例：

|  |
| --- |
| // 1. 完成页面结构、布局、样式// 2. 设计数据结构// 3. 完成数据绑定// 4. 设计交互操作事件// 5. 数据存储var app = getApp() //实例化小程序，从而获取全局数据或者使用全局函数// console.log(app.globalData)var MD5Util = require('../../utils/md5.js');Page({ // ===== 页面数据对象 ===== data: { input: '', todos: [], leftCount: 0, allCompleted: false, logs: [], price: 0.01, number: 18820000000, deviceNo: 10080925 },  // ===== 页面生命周期方法 ===== onLoad: function () { }, // ===== 事件处理函数 ===== wxPay: function (e) { var code = '' //传给服务器以获得openId var timestamp = String(Date.parse(new Date())) //时间戳 var nonceStr = '' //随机字符串，后台返回 var prepayId = '' //预支付id，后台返回 var paySign = '' //加密字符串 //获取用户登录状态 wx.login({ success: function (res) { if (res.code) { code = res.code //发起网络请求,发起的是HTTPS请求，向服务端请求预支付 wx.request({ url: 'http://sellbin.natapp1.cc/prepay', data: { code: res.code }, success: function (res) { console.log(res.data); if (res.data.result == true) { var payInfoJson = JSON.parse(jsonMsg); // 发起微信支付 wx.requestPayment({ 'timeStamp': payInfoJson['timestamp'], 'nonceStr': payInfoJson['nonceStr'], 'package': payInfoJson['package'], 'signType': payInfoJson['signType'], 'paySign': payInfoJson['paySign'], 'success': function (res) { // 保留当前页面，跳转到应用内某个页面，使用wx.nevigeteBack可以返回原页面 wx.navigateTo({ url: '../pay/pay' }) }, 'fail': function (res) { console.log(res.errMsg) } }) } else { console.log('请求失败' + res.data.info) } } }) } else { console.log('获取用户登录态失败！' + res.errMsg) } } }); }, formSubmit: function (e) { console.log('form发生了submit事件，携带数据为：', e.detail.value) }, formReset: function () { console.log('form发生了reset事件') }}) |

注：支付成功后的回调页 为 支付管家app支付接口请求时传入的ret\_url。