

——微信与移动Web开发之

# 第11讲 微信支付与Oauth授权

# 本次课程目录

- 一 . 微信支付分类与流程
- 二 . Oauth协议基础
- 三 . 微信网页授权

# 课程概要

- 本次课程先简单讲解一下微信支付，同时以扫码支付为例进行支付流程的解释说明，由于条件限制，无法进行示例演示。
- 然后是Oauth入门，简单了解Oauth。
- 微信网页授权是Oauth2.0协议的具体实现，实现微信网页授权获取用户基本信息。

# 第一节

一 . 微信支付分类与流程

二 . Oauth协议基础

三 . 微信网页授权

# 前言

- 本次课程讲解微信支付主要是在理论层面，实际的支付场景是无法实现的，为了能让同学们应对实际的场景而做的一些说明。
- 主要涉及到主要支付方式的流程处理，回调处理，需要注意的问题，以及微信支付SDK中存在的问题。

# 微信支付分类



## 刷卡支付

用户打开微信钱包的刷卡的界面，商户扫码后提交完成支付



## 公众号支付

用户在微信内进入商家H5页面，页面内调用JSSDK完成支付



## 扫码支付

用户打开“微信扫一扫”，扫描商户的二维码后完成支付



## APP支付

商户APP中集成微信SDK，用户点击后跳转到微信内完成支付



## H5支付

用户在微信以外的手机浏览器请求微信支付的情景唤起微信支付



## 小程序支付

用户在微信小程序中使用微信支付的场景

# 配置接入微信支付

支付配置 ?

公众号支付

支付授权目录 [添加](#)

JSAPI支付授权目录 (最多可添加5个)

操作

删除

http://wechatpay.muya.co.kr/

删除

http://www.baidu.cOm/

删除

删除

删除

删除

删除

扫码支付

扫码回调链接

扫码回调链接

操作

http://www.sinaxxx.com/

修改

H5支付

H5支付域名 [添加](#)

H5支付域名 (最多可添加5个)

操作

test.alliewechat.com

删除

# PC端扫码支付

- 步骤1：商户根据微信支付的规则，为不同商品生成不同的二维码，展示在各种场景，用于用户扫描购买。
- 步骤2：用户使用微信“扫一扫”扫描二维码后，获取商品支付信息，引导用户完成支付。
- 步骤3：用户确认支付，输入支付密码。
- 步骤4：支付完成后会提示用户支付成功，商户后台得到支付成功的通知，然后进行发货处理。



# 扫码支付流程(模式一)

- (1) 商户后台系统根据微信支付规定格式生成二维码(规则见下文),展示给用户扫码。
- (2) 用户打开微信“扫一扫”扫描二维码,微信客户端将扫码内容发送到微信支付系统。
- (3) 微信支付系统收到客户端请求,发起对商户后台系统支付回调URL的调用。调用请求将带productid和用户的openid等参数,并要求商户系统返回数据包,详细请见“本节3.1回调数据输入参数”
- (4) 商户后台系统收到微信支付系统的回调请求,根据productid生成商户系统的订单。
- (5) 商户系统调用微信支付【统一下单API】请求下单,获取交易会话标识(prepay\_id)
- (6) 微信支付系统根据商户系统的请求生成预支付交易,并返回交易会话标识(prepay\_id)。
- (7) 商户后台系统得到交易会话标识prepay\_id(2小时内有效)。
- (8) 商户后台系统将prepay\_id返回给微信支付系统。返回数据见“本节3.2回调数据输出参数”
- (9) 微信支付系统根据交易会话标识,发起用户端授权支付流程。
- (10) 用户在微信客户端输入密码,确认支付后,微信客户端提交支付授权。
- (11) 微信支付系统验证后扣款,完成支付交易。
- (12) 微信支付系统完成支付交易后给微信客户端返回交易结果,并将交易结果通过短信、微信消息提示用户。微信客户端展示支付交易结果页面。
- (13) 微信支付系统通过发送异步消息通知商户后台系统支付结果。商户后台系统需回复接收情况,通知微信后台系统不再发送该单支付通知。

# 生成二维码需要的参数

名称	变量名	类型	必填	示例值	描述
公众账号ID	appid	String(32)	是	wx8888888888888888	微信分配的公众账号ID
商户号	mch_id	String(32)	是	1900000109	微信支付分配的商户号
时间戳	time_stamp	String(10)	是	1414488825	系统当前时间，定义规则 详见 <a href="#">时间戳</a>
随机字符串	nonce_str	String(32)	是	5K8264ILTKCH16CQ2502SI8Z NMTM67VS	随机字符串，不长于32位。 推荐 <a href="#">随机数生成算法</a>
商品ID	product_id	String(32)	是	88888	商户定义的商品id 或者订 单号
签名	sign	String(32)	是	C380BEC2BFD727A4B68451 33519F3AD6	签名，详见 <a href="#">签名生成算法</a>

# 支付参数签名算法

- 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：
  - 1.参数名ASCII码从小到大排序（字典序）；
  - 2.如果参数的值为空不参与签名；
  - 3.参数名区分大小写；
  - 4.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。
  - 5.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](https://pay.weixin.qq.com))-->账户设置-->API安全-->密钥设置

# 需要注意的问题

- 下载的微信支付SDK代码中，lib/WxApi.php 中的方法 notify 存在问题，在PHP5.6+版本中，需要使用file\_get\_contents('php://input','r');获取POST数据。去掉\$GLOBALS[ 'HTTP\_RAW\_POST\_DATA' ]的方式。

```
public static function notify($callback, &$msg)
{
    // 获取通知的数据
    $xml = $GLOBALS['HTTP_RAW_POST_DATA'];
    // 如果返回成功则验证签名
    try {
        $result = WxPayResults::Init($xml);
    } catch (WxPayException $e){
        $msg = $e->errorMessage();
        return false;
    }

    return call_user_func($callback, $result);
}
```

# 第二讲

一 . 微信支付分类与流程

二 . Oauth协议基础

三 . 微信网页授权

# 什么是OAuth协议

- OAuth ( Open Authorization , 开放授权 ) 是为用户资源的授权定义了一个安全、开放及简单的标准，第三方无需知道用户的账号及密码，就可获取到用户的授权信息，并且这是安全的。
- 目前的版本是2.0。

# 应用场景

- OAuth的出现是为了解决这样一个问题：

场景：

- 1，你有一个Flickr帐号，在上面有很多照片；
- 2，你需要使用某照片打印网站打印Flickr上面的照片，你可以：
  - 2.1，从Flickr下载照片到本地，然后从电脑里上传照片到打印网站；
  - 2.2，在照片打印网站输入Flickr的帐号密码，直接让网站去Flickr读照片；

问题：把Flickr帐号密码透露给了第三方。

- 解决方案：

**OAuth是让第三方应用不需要用户名密码读取用户数据的一个认证过程。**

上面的场景中，Flickr有API支持OAuth，那么打印网站需要根据API注册应用，打印网站可以要求用户授权访问Flickr照片，而不是提供Flickr的帐号密码。这个授权过程就是OAuth的作用。

# Oauth的思路

- OAuth在"客户端"与"服务提供商"之间，设置了一个授权层（ authorization layer ）。"客户端"不能直接登录"服务提供商"，只能登录授权层，以此将用户与客户端区分开来。"客户端"登录授权层所用的令牌（ token ），与用户的密码不同。用户可以在登录的时候，指定授权层令牌的权限范围和有效期。
- "客户端"登录授权层以后，"服务提供商"根据令牌的权限范围和有效期，向"客户端"开放用户储存的资料。



# Oauth优点

- 安全性更高，用户仅需对需要的操作授权，同时不用提供账号密码。
- 不需要针对不同的网站注册多个账号，使用授权就可以实现一个账号通过授权的方式登录不同的网站。

# 基本处理流程

- (A) 用户打开客户端以后，客户端要求用户给予授权。
- (B) 用户同意给予客户端授权。
- (C) 客户端使用上一步获得的授权，向认证服务器申请令牌。
- (D) 认证服务器对客户端进行认证以后，确认无误，同意发放令牌。
- (E) 客户端使用令牌，向资源服务器申请获取资源。
- (F) 资源服务器确认令牌无误，同意向客户端开放资源。

# 第三讲

- 一 . 微信支付分类与流程
- 二 . Oauth协议基础
- 三 . 微信网页授权

# 如何配置

- 1、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。
- 2、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。

# 微信授权类型说明

- 1、以snsapi\_base为scope发起的网页授权，是用来获取进入页面的用户的openid的，并且是静默授权并自动跳转到回调页的。用户感知的就是直接进入了回调页（往往是业务页面）
- 2、以snsapi\_userinfo为scope发起的网页授权，是用来获取用户的基本信息的。但这种授权需要用户手动同意，并且由于用户同意过，所以无须关注，就可在授权后获取该用户的基本信息。
- 3、用户管理类接口中的“获取用户基本信息接口”，是在用户和公众号产生消息交互或关注后事件推送后，才能根据用户OpenID来获取用户基本信息。这个接口，包括其他微信接口，都是需要该用户（即openid）关注了公众号后，才能调用成功的。

# 微信授权处理流程1

第一步：用户同意授权，获取code

`https://open.weixin.qq.com/connect/oauth2/authorize?appid=APPID&redirect_uri=REDIRECT_URI&response_type=code&scope=SCOPE&state=STATE#wechat_redirect`

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， <b>请使用urlEncode对链接进行处理</b>
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

# 微信授权处理流程2

## 第二步：通过code换取网页授权access\_token

`https://api.weixin.qq.com/sns/oauth2/access_token?appid=APPID&secret=SECRET&code=CODE&grant_type=authorization_code`

注意：这里通过code换取的是一个特殊的网页授权access\_token, 与基础支持中的access\_token（该access\_token用于调用其他接口）不同。公众号可通过下述接口来获取网页授权access\_token。如果网页授权的作用域为snsapi\_base，则本步骤中获取到网页授权access\_token的同时，也获取到了openid，snsapi\_base式的网页授权流程即到此为止。

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

# 获取网页access\_token返回值

- 正确:

```
{  
  "access_token":"ACCESS_TOKEN",  
  "expires_in":7200,  
  "refresh_token":"REFRESH_TOKEN",  
  "openid":"OPENID",  
  "scope":"SCOPE"  
}
```
- 错误:

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



# 微信授权处理流程3

第三步：刷新access\_token（如果需要）

`https://api.weixin.qq.com/sns/oauth2/refresh_token?appid=APPID&grant_type=refresh_token&refresh_token=REFRESH_TOKEN`

由于access\_token拥有较短的有效期，当access\_token超时后，可以使用refresh\_token进行刷新，refresh\_token有效期为30天，当refresh\_token失效之后，需要用户重新授权。

参数	是否必须	说明
appid	是	公众号的唯一标识
grant_type	是	填写为refresh_token
refresh_token	是	填写通过access_token获取到的refresh_token参数

# 微信授权处理流程4

## 第四步：拉取用户信息(需scope为 snsapi\_userinfo)

[https://api.weixin.qq.com/sns/userinfo?access\\_token=ACCESS\\_TOKEN&openid=OPENID&lang=zh\\_CN](https://api.weixin.qq.com/sns/userinfo?access_token=ACCESS_TOKEN&openid=OPENID&lang=zh_CN)

- 正确时返回的JSON数据包如下：

```
{  "openid": "OPENID",
  "nickname": NICKNAME,
  "sex": "1",
  "province": "PROVINCE",
  "city": "CITY",
  "country": "COUNTRY",
  "headimgurl": "http://wx.qlogo.cn/mmopen/g3MonUZtNHkdmzicIlibx6iaFqAc56vxLSUfpb6n5WKSYVY0ChQKkiaJSgQ1dZuTOgvLLrhJbERQQ4eMsv84eavHiaiceqxibJxCfHe/46",
  "privilege": [ "PRIVILEGE1" "PRIVILEGE2" ],
  "unionid": "o6_bmasdasdsad6_2sgVt7hMZOPfL"
}
```

- 错误：{"errcode":40003,"errmsg":"invalid openid "}

# 返回信息参数说明

参数	描述
openid	用户的唯一标识
nickname	用户昵称
sex	用户的性别，值为1时是男性，值为2时是女性，值为0时是未知
province	用户个人资料填写的省份
city	普通用户个人资料填写的城市
country	国家，如中国为CN
headimgurl	用户头像，最后一个数值代表正方形头像大小（有0、46、64、96、132数值可选，0代表640*640正方形头像），用户没有头像时该项为空。若用户更换头像，原有头像URL将失效。
privilege	用户特权信息，json 数组，如微信沃卡用户为（chinaunicom）
unionid	只有在用户将公众号绑定到微信开放平台帐号后，才会出现该字段。

# 感谢聆听！

---

THANK YOU FOR YOUR ATTENTION