# 微信相关接口

1. 微信登录

\webapp\avision\Lib\Action\WeixinCallAction.class.php

这文件会同时部署到两个服务器上（live.av365.cn、www.av365.cn）由于类里面划分有两种角色（Server、Client）目前live.av365.cn作为Server、[www.av365.cn](http://www.av365.cn)作为Client。[www.av365.cn](http://www.av365.cn)的请求都需要经过live.av365.cn。

* 1. 目前申请授权登录流程
* 页面跳转到微信登录页：Home/wxLogin，此页面可传入两个参数：chnId或backUrl。在方法内部最终转换为backUrl。
* 调用WeixinCall/Apply，传入成功后的回访地址backUrl (A)。  
  填写发起端message表，task="WeixinCall"，action="PlatformWx", attr={ "backUrl":"'.$backUrl.'"}, step=0, 获取查询此记录的关键字keystr  
  引导客户端浏览器打开微信登录地址。在WxBase.php中定义WX\_SERVER\_QRY。目前是远端live.av365.cn中的http://live.av365.cn/home.php/WeixinCall/RemoteQuery?msgStr=, 其中msgStr包含keystr
* 访问RemoteQuery后，Live.av365.cn会调用GoWx()方法，调用WxOauth2类的AskMobile或AskPC向微信获取用户信息。微信会在后台回访define('WX\_Oauth2PCBack', U('WeixinCall/PcBack', NULL, NULL, NULL, true));或define('WX\_Oauth2MoBack', U('WeixinCall/MoBack', NULL, NULL, NULL, true));
* 当微信后台回访PcBack或MoBack时，会调用WeixinCall/GetAndSave获取用户信息并调用WxOauth2/SaveWxLog保存到wxlog表中。
* GetAndSave内部：  
  $ouath->GetAccessToken获取微信通讯令牌  
  $ouath->GetUserInfo2 取微信用户信息，中间调用CommonFun. GetUrlJson,过滤昵称中的表情符。  
  $ouath->SaveWxLog 把用户信息记录在wxlog表中
* 授权结束调用WeixinCall/ApplyEnd在这里若是远端调用，则从message表的attr属性中读取backUrl并引导浏览器跳转，这个URL是调用远端前生成的，当前是近端的WeixinCall/RemoteBack。
* 在近端的WeixinCall/ RemoteBack先调用远端的WeixinCall/RemoteGetInfo获取记录在原端message表中的微信用户信息，通过json格式返回。
* 接着调用HandleUserInfo登录，若需要则创建本地用户账号。
* 读近端的message表获取需要返回的业务页面，并再引用用户浏览器访问[www.av365.cn](http://www.av365.cn)的回访地址(A)。
  1. 微信登录开发

文档

<https://developers.weixin.qq.com/doc/offiaccount/Getting_Started/Overview.html>

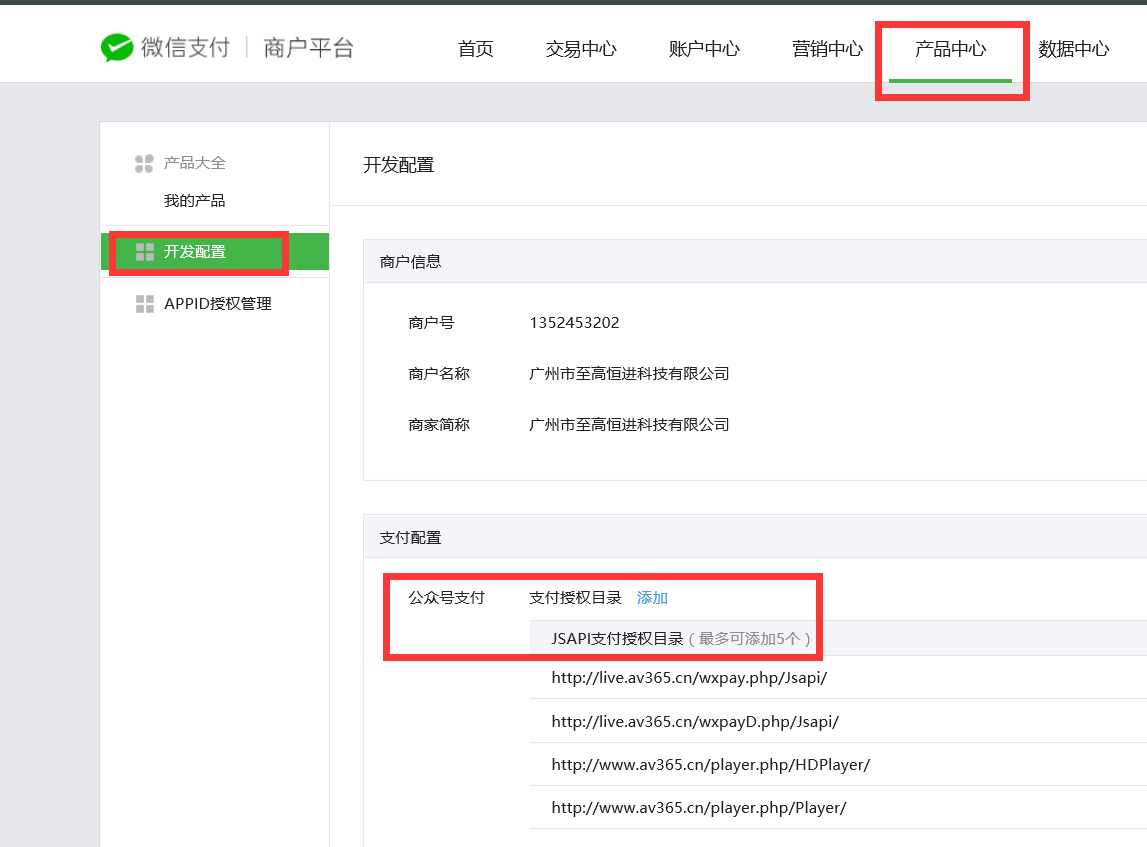
1. 微信支付商户平台

\webapp\wxpay\Lib\Action\JsapiAction.class.php

说明：封装了微信支付接口（主要是调用微信的JsApiPay类），向外定义了支付参数及支付成功后的回调。

* 1. 支付授权目录

●微信支付商户平台（pay.weixin.qq.com）中需要设置支付授权目录，设置位置如下图



**目前设置了四个支付授权目录**

公共支付目录 http://live.av365.cn/wxpay.php/Jsapi/

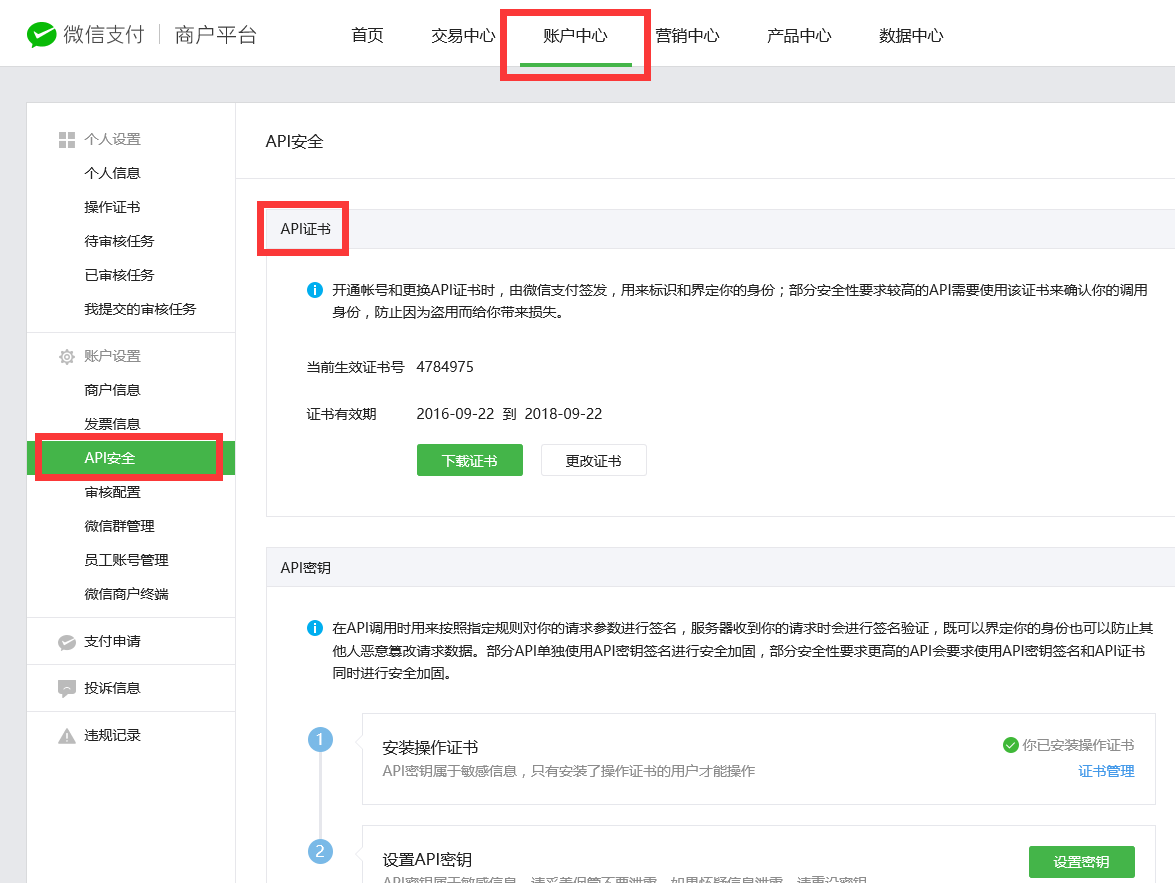
公共支付测试 http://live.av365.cn/wxpayD.php/Jsapi/

播放器目录直接支付，为了减少跳转http://www.av365.cn/player.php/HDPlayer/

播放器直接支付测试

http://www.av365.cn/player.php/Player/

* 1. API证书的安装说明



目前存放的位置有

公共支付

/var/www/ywz/webapp/public/wxpay/cert (live.av365.cn)

播放端直接支付

/home/www/ywz/webapp/public/wxpay/cert (www.av365.cn)

在\webapp\public\wxpay\WxPay.Config.php代码中设置了路径常量

define('SSLCERT\_PATH', dirname(\_\_FILE\_\_).'/cert/apiclient\_cert.pem');

define('SSLKEY\_PATH', dirname(\_\_FILE\_\_).'/cert/apiclient\_key.pem');

\webapp\public\wxpay\WxPay.Api.php会引用到。

**API证书近期需要更新（5月29日前），留意下图：**



* 1. API密钥

向服务端申请的秘钥，用于通信安全。



秘钥是一个字串存放在：

webapp/public/wxpay/WxPay.Config.php 中的**const** KEY常量中。

* 1. 业务逻辑

JsapiAction类内部逻辑如下：

该文件会同时部署到两个服务器上（live.av365.cn、www.av365.cn）由于类里面划分有两种角色（Server、Client）目前live.av365.cn作为Server、[www.av365.cn](http://www.av365.cn)作为Client。[www.av365.cn](http://www.av365.cn)的请求都需要经过live.av365.cn。

**!警告! 两个服务器上wxpay/JsapiAction.class.php虽然文件名相同，但gotoPay()与gotoPayJs()的参数不同，live是json字串，www中是php数组！**

[www.av365.cn后端向live.av365.cn调用/wxpay.php/Jsapi/RemoteApply.html](http://www.av365.cn后端向live.av365.cn调用/wxpay.php/Jsapi/RemoteApply.html)请求支付，获取临时字串token。再引导浏览器访问live.av365.cn的/wxpay.php/Jsapi/h5js?t=token。来到支付页面。

调用RemoteApply可POST一个p变量，其中包括了订单信息，相当于gotoPay的$para参数。

支付完成后，浏览器回跳转到当初设置的回调页面。而微信后端会回访live.av365.cn的JsapiAction的notify方法，live.av365.cn处理好本地的业务后会回访[www.av365.cn的JsapiAction的notifyB](http://www.av365.cn的JsapiAction的notifyB)方法。[www.av365.cn](http://www.av365.cn)会处理好本地业务，然后会回访当初设置的业务回访地址（相当于业务结束事件响应）。业务回访地址主要是处理业务层级的逻辑处理（如：购买套餐后如果赠送。）

JsapiAction的业务入口函数有两个，分别是gotoPay和gotoPayJs。传入的参数是一样的，区别是使用场景的不同。gotoPay是引导用户浏览器跳转到live.av365.cn的支付页面上。gotoPayJs不需要引导用户浏览器跳转到live.av365.cn页面，而是通过ajax访问live.av365.cn获取支付信息后，在当前页面局部加载页面，简单说就是不跳转。

JsapiAction的业务调用代码片，可参照：

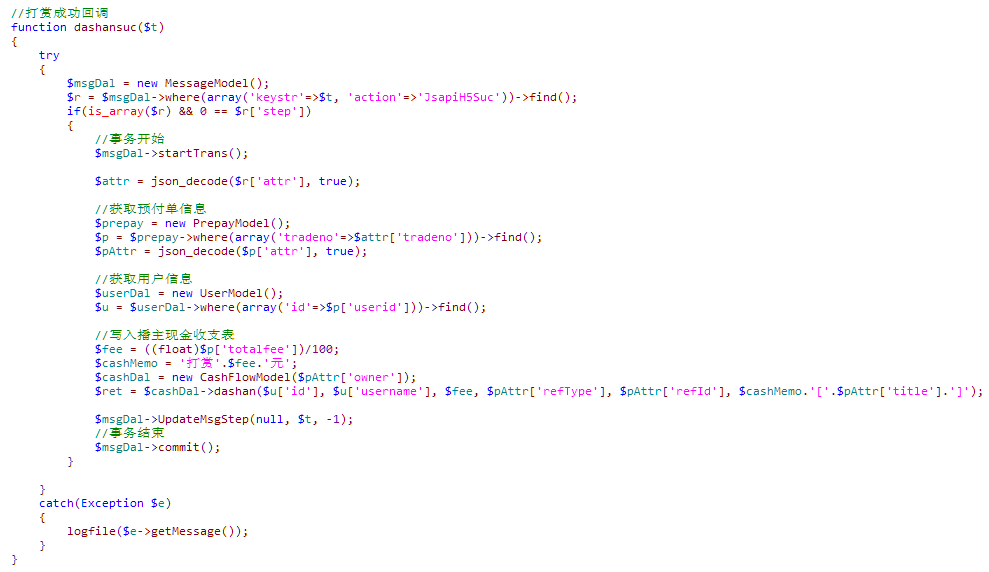
/webapp/avision/Lib/Action/HDPlayerAction.class.php的dashan方法，下图。



支付成功的回调处理，可参照

/webapp/avision/Lib/Action/HDPlayerAction.class.php的dashansuc方法，

下图。



* + 1. 扫二维码支付过程-账号充值
* 入口/home.php/Home/recharge 点击“确定充值”后调用业内JS函数pay()
* pay()内：  
  onCodeRecv是获取微信支付二维码参数后，进行下一步处理的JS函数名。  
  codeUrl=/home.php/Home/rechargePayCode，通过ajax调用后台获取支付二维码相关参数。
* /home.php/Home/rechargePayCode  
  相关支付信息填入内部message表中以跟踪支付返回过程。二维码包括以下关键信息：  
  二维码验证通过后发起支付的地址：payurl=rechargepay  
  message跟踪表里唯一标识此交易的字串：t  
  交易金额：fee
* 前端收到二维码参数后，调用$(“#qrcode”).qrcode生成二维码，同时启动JS函数payCheck()定时向后端查询支付是否已经完成。  
  后端支持函数为：rechargePayCheck(), 若支付完成返回：has=true
* 前端扫描二维码成功后，在手机端调用payurl即rechargepay，后者发起支付过程。$payApi->gotoPay($para);   
  gotoPay位于/webapp/wxpay/Lib/Action/JsapiAction.class.php中
* gotoPay 生成预付订单记录prepay；  
  GetUrlContent($url, $p, false); 调用支付服务端，准备支付，获取服务端支付token：  
  http://live.av365.cn/wxpay.php/Jsapi/RemoteApply.html 这里设置支付完成后，服务端回调请求端的地址，此例是http://www.av365.cn/wxpay.php/Jsapi/notifyB  
  跳转到：http://live.av365.cn/wxpay.php/Jsapi/h5?t=，显示支付页面
* 微信支付成功后 回调服务端的接口http://live.av365.cn/wxpay.php/Jsapi/notify  
  若由客户端发起支付，此接口继续调用，客户端的callback，就是上一步的notifyB
  + 1. 微信浏览器内支付过程

1. 微信jssdk(分享、图片传输、不跳页支付)

微信jssdk提供了分享、图片传输、不跳页支付等等的js层接口。

jssdk有大量的参数和签名信息需要动态生成。

webapp/public/WxJs.Class.php提供了动态生成JSSDK的支持。

通过JS接口调用微信功能的页面需要用到JSSDK。

HDPlayerAction.class.php中

**public function** js()

提供了动态嵌入JSSDK的示例。

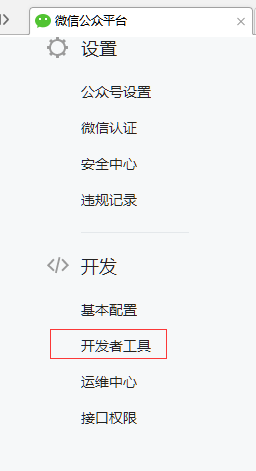
1. 公众号菜单

/webapp/WeiXinHaoAction.php里调用到了/public/WxMenu.class.php

http://www.av365.cn/home.php/WeiXinHao/menu做菜单管理

1. 微信web开发者工具

登录mp.weixin.qq.com。左则菜单最下面



1. 微信服务网关live.av365.cn

设置此服务器作为与微信的统一接口，目的是避免应用服务器的变化要过多设置与微信的通讯参数，同时使得一个接口可支持的服务器得到扩展。

但这些目标目前似乎实现得不好。

* 1. 使用到的数据表

Av2\_epaylog 流水记录，记录微信支付成功的信息，微信支付成功回调时回写入。

av2\_message 业务消息表。与[www.av365.cn](http://www.av365.cn)上的用途一样。

av2\_wxlog 微信用户授权登录的log，当调用getUserInfo成功是回写入。可用于判断用户之前是否曾经授权登录过，流水记录。

av2\_wxmsg 微信公众号回调，当有回调时会写入。类型有，文本消息，菜单事件等。流水记录。

* 1. Message表说明

delimiter $$

CREATE TABLE `av2\_message` (

`id` int(11) NOT NULL auto\_increment,

`task` varchar(50) default NULL COMMENT '业务',

`action` varchar(50) default NULL COMMENT '细分业务',

`createtime` int(14) default NULL COMMENT '创建时间',

`updatetime` int(14) default NULL COMMENT '更新时间',

`createstr` varchar(45) default NULL COMMENT '创建时间字串',

`updatestr` varchar(45) default NULL COMMENT '更新时间字串',

`attr` varchar(1000) default NULL COMMENT '属性内容',

`keystr` varchar(50) default NULL COMMENT '关键字串',

`step` int(3) default NULL COMMENT '消息状态，全局定义0表示新增消息未经处理，-1表示已处理消息，不再需要处理。其它数值由不同的业务去处理。',

PRIMARY KEY (`id`),

KEY `IND\_keystr` (`keystr`)

) ENGINE=InnoDB AUTO\_INCREMENT=1233552 DEFAULT CHARSET=utf8 ROW\_FORMAT=COMPACT COMMENT='消息队列'$$

字段说明

●Task 业务名称，也可以说是模块名称，一般是由调用模块自身决定，例如，微信支付模块，会用WxPay字串，微信授权登录模块会用WeixinCall字串。

●Action 细分业务名称，指创建的记录针对什么具体的业务。由模块自己定义。例如，微信支付模块用到的’JsapiH5Suc‘，指支付成功后，回调业务，keystr字段记录这次回调的随机字串。‘‘payCode’’指PC端生成支付二维码业务，keystr字段记录这次支付的随机字串。

●createtime 记录创建时间，可用于判断是否超时

●updatetime 记录更新时间。

●attr 扩展属性内容，创建记录时附带的业务参数，或临时变量。具体内容由业务自身决定。像微信支付业务，会记录支付信息，支付成功的回调地址，如下：{"userId":"135548","openid":"o3NkBwB9Qy-TLW030DpFEPhezGA0","total":100,"body":"\u6253\u8d4f","callback":"http:\/\/www.av365.cn\/home.php\/HDPlayer\/dashansuc","list":[{"detail":"\u6253\u8d4f1\u5143","fee":100,"img":"http:\/\/www.av365.cn\/wxpay\/default\/images\/gift.png"}],"extpara":{"playType":4,"refId":"1232","refType":"live","title":"\u8001\u8d56\u4e0d\u8d56 2018\u7528\u6237\u9881\u5956\u76db\u5178\u54c1\u724c\u65b0\u5f62\u8c61\u53d1\u5e03\u4f1a","owner":"112058"}}

●keystr 临时字串，用于业务间传递，web回调，常作为select关键字。

●step 记录的状态标记，一般定义：0表示新增后未处理，或没有发生变化。－1：表示已处理完毕，不可再使用。业务模块自身也可以定义其它值用作自身逻辑处理。

备注：此表数据量较多，建议定时删除已完成或超时的记录。