

PHP代码篇（七） – ThinkPHP5.1使用支付宝

作者：有故事的人 来源：范文网 www.wtabcd.cn/fanwen/

本文原地址：<https://www.wtabcd.cn/fanwen/zuowen/c4f2ca432dff19020ed673c3946176cd.html>

范文网，为你加油喝彩！

一、说明

1、整体流程

：之前没有做过支付，所以对于支付，还是有点迷糊，这次主要是记录下“支付宝电脑网站支付”的一个完整开发流程。说明（本身支付宝支付，微信支付其实有很多类型，比如支付宝有，当面付，app支付，手机网站支付，微信jsapi支付，付款码支付，native支付，h5支付，我们根据公司业务需求，选择对应的官网开发文档，进行引入即可）。本次开发使用的是thinkphp5.1版本。

2、涉及步骤

：支付宝支付流程说明

：支付宝官方damo的使用

：支付宝回调notify_url异步通知的书写

：集成thinkphp框架下引入支付宝damo的流程

二、支付业务及支付宝流程说明

1、首先我们的业务需求及支付宝流程如下：

2、实际中大多页面场景如下：

三、开发准备步骤：

1、查看支付宝“支付宝支付开放平台”地址/d/file/titlepic/pp
2、文档里面有非常详细的介绍，这边我们不做具体展开说明，实际开发中开创长吉体诗歌的是我们需要了解下，注意接口说明。这里我们点击下载官方的php demo

3、下载后应该是一个压缩包 alipay.trade.page.pay-php-utf-8.zip，我们对其解压，然后重命名为 alipaypc。

4、上面的准备工作做好了后，我们接下来进入实际的开发步骤。

四、实际开发流程，在框架中使用demo

1、涉及步骤

：将解压后的文件夹alipaypc复制到框架中

：对官方写的支付，回调，业务代码进行分离，分离出一个自定义的alipaypc()类

：配置config.php支付相关参数，

：在控制器中调用alipaypc()类，进行支付

：新增回调文件，书写回调代码

2、解压后我们可以看到，demo文件如下，文件具体我这边就不做说明，只说下本次需要参考的文件，具体自己可以看下官网手册。

3、上面的return_url.php 为官方写的回调文件，主要是进行验证

```
$arr=$_post;$alipaysevice = new alipaytradeservice($config); $alipaysev淹没湮没ice->writelog(var_export($_post,true))  
;$result = $alipaysevice->check($arr); //验签方法
```

4、然后是pagepay文件夹里面，有进行支付，查询交易，关闭交易，退款，退款查询，的官方参看代码，待会我们书写自己的alipaypc()类，基本是参考这样写。

5、这里说明下，解压的文件我们是放在thinkphp的extend 目录，有些人可能习惯放到vendor 目录，thinkphp官网有说过，vendor 作为 使用composer安装的扩展文件目录，而extend是手动引入的第三方文件目录，所以我们开发，还是需要准守规范比较好。

6、下面我们在extend下的lib目录，没有新建一个lib文件夹，新建一个刚才说的自定义支付类 alipaypc.php 文件。

7、代码如下，这里面我们进行分离出刚才说的官方支付代码

```
<?php/** * created by phpstorm. * user: wuyan * date: 2020/4/28 * time: 20:00 * 支付宝电脑网  
站支付 */namespace lib\class alipaypc{ /* * 支付宝收款 */ public static  
function setailpay() { }}
```

8、直接复制和修改pagepay目录下的pagepay.php文件，主要是修改下文件引入路径，和参数传入的方式

```
<?php/** * created by phpstorm. * user: wuyan * date: 2020/4/28 * time: 20:00 * 支付宝电脑网站
```

```
支付 */namespace lib\class alipaypc{ /** * 支付宝收款 * @param [type] $out_trade_no 商户订单号 * @param [type] $subject 订单名称 * @param [type] $total_amount 付款金额 * @param [type] $body 商品描述，可空 * @param [type] $config_data 支付参数,数组类型;为空则取默认参数 * @return void */ public static function setAlipay($out_trade_no,$subject,$total_amount,$body,$config_data = []) { require_once env('extend_path').'/alipaypc/config.php';//引入路径参考自己的实际存放路径 require_once env('extend_path').'/alipaypc/pagepay/service/alipaytradeservice.php'; require_once env('extend_path').'/alipaypc/pagepay/buildermodel/alipaytradepagepaycontentbuilder.php'; //构造参数 $payrequestbuilder = new \alipaytradepagepaycontentbuilder(); //类名前面加\调用外部类，需要在类名前加\$ $payrequestbuilder->setbody($body); $payrequestbuilder->setsubject($subject); $payrequestbuilder->settotalamount($total_amount); $payrequestbuilder->setouttradeNo($out_trade_no); $aop = new \alipaytradeservice($config); /** * pagepay 电脑网站支付请求 * @param $builder 业务参数，使用buildmodel中的对象生成。 * @param $return_url 同步跳转地址，公网可以访问 * @param $notify_url 异步通知地址，公网可以访问 * @return $response 支付宝返回的信息 */ $response = $aop->pagepay($payrequestbuilder,$config['return_url'],$config['notify_url']); //输出表单 var_dump($response); }}
```

9、如上，我们支付的类就创建好了，现在我们配置下config.php文件。

10、这里主要是配置，应用id，商户私钥，支付宝公钥，支付宝网关（注意下是正式账号，还是沙箱测试账号）。

补充说明，支付宝如何开通支付和如何申请沙箱账号，这里不做说明，不会的话看下期讲解，然后生成密钥这里也不做介绍，下期。

11、写好支付类alipaypc()，和配置好config.php后，我们在控制器controller中开始调用。

```
<?php namespace app\admin\controller; use lib\alipaypc;//引入支付类 class user extends base{ public function alipay(request $request) { $out_trade_no = "19982019121202021000"; //订单号 $subject = "测试商品000"; //商品名称 $total_amount = 0.1; //商品金额 元 $body = "这是用于测试"; //商品描述 $config_data = [ 'return_url' => $request->domain().'/index.php/admin/user/return_url', //同步跳转待写 'notify_url' => $request->domain().'/index.php/admin/callback/notify_url', //异步跳转待写 ]; //----- 书写业务代码，比如生成订单记录 //----- //因为声明为静态类，所以直接调用 alipaypc::setalipay($out_trade_no,$subject,$total_amount,$body,$config_data = []); }}
```

12、点击你的域名/index.php/admin/user/useralipay发现，如果没有问题，就可以直接掉转到支付宝付款页面，如果是可以直接扫码支付，也可以登录你的账户进行支付，如果支付成功，页面就会跳转到，你写的同步地址index.php/admin/login/return_url处。

13、当然上面的支付成功后，还有一个最重要的异步回调，如下是支付宝官方说明：

：程序执行完后必须打印输出“success”（不包含引号）。如果商户反馈给支付宝的字符不是success这7个字符，支付宝服务器会不断重发通知，直到超过24小时22分钟。一般情况下，25小时以内完成8次通知（通知的间隔频率一般是：4m,10m,10m,1h,2h,6h,15h）；：程序执行完成后，该页面不能执行页面跳转。如果执行页面跳转，支付宝会收不到success字符，会被支付宝服务器判定为该页面程序运行出现异常，而重发处理结果通知；

14、回调的意思就是，通过你设置的notify_url回调地址，支付宝在支付成功后，会带参（比如业务参数类：交易号，交易金额，公共参数：授权appid，签名）来请求你这个地址，告诉你，这一单，我们支付宝收到款了，怎么，怎么的，然后这里肯定有可能会有第三方的恶意请求（伪造请求）。所以你对这些进行验证，然后识别出是支付宝的回调，然后进行业务处理。比如修改订单支付状态。

15、最后，我们书写回调，当然还是先写一个回调的类，在之前的alipaypc()类里面新增setnotify_url()//这个主要是进行验签

```
<?php/** * created by phpstorm. * user: wuyan * date: 2020/4/20 * time: 10:59 * 支付宝 */names
pace lib;class alipay{ /** * 支付宝收款 * @param [type] $out_trade_
no 商户订单号 * @param [type] $subject 订单名称 * @param [type] $total_amount
付款金额 * @param [type] $body 商品描述，可空 * @param [type] $config_data 支
付参数,数组类型;为空则取默认参数 * * @return void */ public static fu
nction setailpay($out_trade_no,$subject,$total_amount,$body,$config_data = []){ require_o
nce env('extend_path'). '/alipaypc/config.php'; require_once env('extend_path'). '/alipaypc/pagep
ay/service/alipaytradeservice.php'; require_once env('extend_path'). '/alipaypc/pagepay/buildermodel/ali
paytradepagepaycontentbuilder.php'; $config = array_merge($config, $config_data);
//构造参数 $payrequestbuilder = new \alipaytradepagepaycontentbuilder(); if(!e
mpty($body)){ $payrequestbuilder->setbody($body); } $p
ayrequestbuilder->setsubject($subject); $payrequestbuilder->settotalamount($total_amount);
$payrequestbuilder->setouttradeno($out_trade_no); $aop = new \alipaytradeservice($config);
/** * pagepay 电脑网站支付请求 * @param $builder
业务参数，使用buildmodel中的对象生成。 * @param $return_url 同步跳转地址，公网可以
访问 * @param $notify_url 异步通知地址，公网可以访问 * @return $r
esponse 支付宝返回的信息 */ $response = $aop->pagepay($payrequestbuilder,
$config['return_url'],$config['notify_url']); //输出表单 // var_dump($response);
} /最基本的英语** * 支付宝服务器异步通知页面 * * @param [type]
] $arr 支付成功返回信息post * * @return void */ public static functio
n setnotify_url($arr) { require_once env('extend_path'). '/alipaypc/config.php';
require_once env('extend_path'). '/alipaypc/pagepay/service/alipaytradeservice.php'; $alipayse
rvice = new \alipaytradeservice($config); $alipaysevice->writelog(var_export($arr,true));
$result = $alipaysevice->check($arr); //进行验签 if($result) { //验证成功
//交易状态 if($arr['trade_status'] == 'trade_finished' || $arr['trade_status'] ==
'trade_success') { return true; } else {
return false; } }else {
return false; } }}
```

16、然后在控制器里面新增回调方法

```
<?php/** * 第三方请求，不需要验证 */namespace app\api\controller;use think\controller;use lib\alipaypc;class
callback extends controller{ /** * 支付宝商户收款异步回调 */ public functi
on notify_url() { $arr = $_post; $result = alipaypc::setnotify_url($arr);
//进行验签 //验签成功后 //判断订单号是否存在数据库 //进行
业务处理----- }}
```

五、结束语

1、支付宝电脑网页支付，整体来说，由于支付宝官方文档和本身集成，还是相对容易的，主要是在实际中，对于

支付日志，回调日志，防止重复提交，防止伪造请求，和重复回调这些代码业务端的进行合理书写。

2、如果大家还是有疑问，可以留言，或是学术研讨会方案页面左上角二维码咨询。

更多作文请访问 https://www.wtabcd.cn/fanwen/list/92_0.html

文章生成doc功能，由[范文网](http://www.wtabcd.cn/fanwen/)开发