pay|4coins

Pay4Coins API documentation

Table of contents

Introduction to Pay4Coins

Your customers do not need to register with Pay4Coins to make payments with Pay4Coins. They can pay quickly and easily with the payment methods provided by Pay4Coins. Thanks to the real-time confirmation of payments, you as a merchant can send the goods directly or credit the coins immediately.

A transaction with Pay4Coins is structured into the following steps from the customer's perspective:

- As soon as a customer wants to pay with Pay4Coins, he will be directed to a Pay4Coins page.
- A selection of payment methods is offered, below, the customer is asked to choose his preferred payment method.
- Afterwards, your customer must process the payment as usual with his chosen payment method.
- If the payment was successful, the customer will be notified.

Integration steps

To integrate Pay4Coins into your system, the following steps are generally necessary. The individual steps are described in detail below:

1. Register as a provider on our website: https://merchant.pay4coins.com/Partner-werden
2. Integrate Pay4Coins into your shop.
3. As soon as a customer wants to pay with Pay4Coins, access our interface with the respective payment data. In response you will receive a link to our checkout page, where you can forward the customer to make the payment.
4. The customer processes the payment in our system.
5. If the transaction was successful, you will be automatically notified by Pay4Coins. You can then immediately take further steps, such as shipping the goods or activating the online offer or crediting coins. The notification of a successful transaction takes place via e-mail as well as via GET notification, which automatically informs your system about any status changes.

Please note:

Please note that during the integration process the customer must be redirected to the payment form so that the URL of Pay4Coins and the SSL certificate of CIPA Media are visible. This means that a framed solution, for example with <iframe> is not allowed for legal reasons

Integration of Pay4Coins

To enable your customers to use Pay4Coins, you must first integrate Pay4Coins in the checkout of your shop. If the customer chooses Pay4Coins and his payment method, then confirms his order, the communication between your shop and our interface (API) begins.

Requesting the API and authentication

To prevent misuse of the interface, an authentication is performed for each interface request. For this purpose, your shop ID is transferred as username and the API key as password.

The interface is called via the following URL:

```
https://api.pay4coins.com/checkout_init.php
```

Transfer of payment data

This table contains all possible parameters that you can pass to our system in a request.

Mandatory parameters		
Parameters	**Type**	**Declaration**
api_key	string	Your Pay4Coins API Key
api_url	string	URL to your interface, to which we send the payment status of the transaction
api_version	string	Please enter "**v1**".
coins	integer	Number of coins the customer should receive
created_at	integer	Specifies the current time in Unix timestamps (in seconds) at which the URL was created/generated.
expires	integer	Specifies the duration in hours how long the URL should be valid. (from 1 to 168)
email	string	E-mail address of your customer Your customer will be informed about a successful transaction by email.
hash	string	Hash from all specified parameters except the parameter "hash" itself.
lang	string (2)	Specifies the language in which the checkout page should be loaded. Example: "de" or "en". If this parameter is not specified, the set browser language of the user is used automatically.

member_id	integer	Unique ID of your customer
name	string	Name of the article the customer buys. For example: "2000 Coins"
price	integer	Price of the article in Euro-Cent
shop_id	integer	Your Pay4Coins Shop-ID
username	string	Username of your customer
Optional parameters		
error_url	string	URL to which your customer should be redirected if the transaction is aborted. If this is not specified, the customer will be shown a page from us.
success_url	string	URL to which your customer should be redirected when the transaction has been successfully completed. If this is not specified, we will show the customer a success page.
created_at	integer	Specifies the current time in Unix timestamps (in seconds) at which the URL was created/generated.
expires	integer	Specifies the duration in hours how long the URL should be valid. (from 1 to 168)
payment_method	String	If you send this parameter, the customer will be taken directly to the payment method you have specified. Otherwise, an overview of all payment methods will be displayed. **code = payment method** cashtocode = CashtoCode creditcard = credit card crypto = Cryptocurrency (Bitcoin, ETH, LTC usw) directdebit = SEPA - direct debit ideal = iDeal neosurf = Neosurf paysafecard = paysafecard phone_landline = Phone landline phone_mobile = Phone mobil prepay = Pre pay przelewy24 = Przelewy24 sofort = Sofortüberweisung (Klarna) tink = Tink
With these parameters you can pass any other values back to your system via our API.		

user_variable_1	string	
user_variable_2	string	
user_variable_3	string	
user_variable_4	string	
Parameter for „AdKlick“ You only need these if you work with the AdKlick partner program.		
adklick_pid	string	Your AdKlick „PID“
adklick_tid	string	Your AdKlick „TID“

Example for requesting the Checkout-Page

```php
<?php

$parameter = array(
    "api_url"       => 'http://sdk.pay4coins.com/receive_data.php',
    "api_version"   => 'v1',
    'created_at'    => time(),
    'expires'       => '24',
    "coins"         => 2000,
    "email"         => 'johndoe@example.com',
    "member_id"     => 1,
    "name"          => 'My nice Product',
    "price"         => 1999,
    "shop_id"       => 123456,
    "username"      => 'JohnDoe85'
);

// Your Pay4Coins-API-Key
$p4c_api_key = 'XXXXXXX';
// remove empty GET parameters
$parameter = array_filter($parameter, "strlen");
//sorting alphabetical
ksort($parameter);
// create http-query
$parameter = http_build_query($parameter);
// create hash from query
$hash = hash("sha512", $parameter.$p4c_api_key);
// create URL
$url = 'https://api.pay4coins.com/checkout_init.php?'.$parameter.'&hash='.$hash;

header('Location: '.$url);
exit;

?>
```

The generated URL would look like the following in the example:

```
https://api.pay4coins.com/checkout_init.php?api_version=v1&coins=2000&created_at=1707301246&expires=24&email=max123%40example.com&member_id=1&name=2000+Coins&price=1999&shop_id=50001&username=Max123&hash=a7ccd4213b9f751bdd7082eaef7a61227cd24cfa2beeada5df62d5dc0b9a15add38b202642e9b7e36851836ff4c7af7fc78d30abfd03e7bc097f7ca9d10bb052
```

Processing the API response

After successful payment, our system will notify your system via GET-notification about the new status. You have to process this notification. To prevent misuse of the interface, an authentication is performed for each call of your interface. For this purpose, the API key is passed as the password.

Parameter transfer from our system to yours

After successful payment, our system transmits the following parameters to your system via GET. You have provided us with your API-URL for the transmission.

IMPORTANT!

Your system must respond to us with "**OK**" after successfully processing all parameters!

Example of the API-URL that we received from you:

```
https://example.com/pay4coins_api/index.php
```

Mandatory parameters		
Parameters	**Type**	**Declaration**
amount	float	Item price in Euro & Cent (e.g.: 19.99 for 19.99 EUR)
coins	integer	Number of coins the customer should receive
currency	string (3)	Currency (EUR or USD)
event	string (1)	p => payed (new transaction) c => chargeback a => credit after chargeback
hash	integer	Hash from all specified parameters except the parameter "hash" itself.
paytype	string	**ccard** => credit card

		ideal => iDEAL **neosurf** => Neosurf **paybycall** => PayByCall (pay by phone) **paysafecard** => paysafecard **prepay** => SEPA credit transfer (advance payment) **sepadd** => SEPA Lastschrift **sofort** => IMMEDIATE bank transfer (KLARNA.)
transaction_id	string	Transaction number of the payment
userid	integer	Unique ID of your customer
Optional parameters If you have sent us these parameters, our system will return them to your system.		
user_variable_1	string	
user_variable_2	string	
user_variable_3	string	
user_variable_4	string	

Simple example of hash checking (authentication)

```php
<?php

$var['hash'] = trim($_GET["hash"]);
unset($_GET['hash']);

// Pay4Coins-API-Key
$api_key = 'xxxxxxxxx';

// remove empty GET parameters
$parameter = array_filter($_GET, "strlen");
//sorting alphabetical
ksort($parameter);
// create http-query
$query = http_build_query($parameter, '&amp;');
// create hash from query
$q_hash = hash("sha512", $query.$api_key);
// check GET-hash is equal to query-hash
if ($var['hash'] != $q_hash) {
    die('Hash incorrect!');
}

echo 'hash is correct';

?>
```

Detailed check of all parameters including hash checking

```php
<?php

// Your Pay4Coins-API-Key
$p4c_api_key = 'XXXXXXX';


/** Allow access only from an IP of Pay4Coins **/
/** ======================================== **/
// Pay4Coins IP's
$ips = array ('91.184.46.111', '91.184.50.9');

// If no remote adress esists - than exit
if (!isset($_SERVER["REMOTE_ADDR"])) {die('empty remote_addr');}

$p4cIP['remote'] = $_SERVER["REMOTE_ADDR"];
$p4cIP['local'] = $ips;

// If remote adress is none from Pay4Coins - then exit
if (!in_array($p4cIP['remote'], $p4cIP['local'])) {die('fals ip');}
/** === IP-check - end ===================== **/

// Check if the parameter are allowed
// Specify here the parameters that you allow
$get_aray = array(
    'amount',
    'coins',
    'created_at',
    'expires',
    'currency',
    'event',
    'paytype',
    'transaction_id',
    'user_variable_1',
    'user_variable_2',
    'user_variable_3',
    'user_variable_4',
    'userid',
    'hash'
);

foreach ($_GET as $key => $value) {
    if (!in_array($key, $get_aray)) {
        echo 'The parameters are incorrect!';
        exit;
    }
}

// Clean the parameters
$var['status']          = preg_replace('/[^a-z]/i', '', $_GET['event']); // (p =>
payed (new transaction), c => transaction chargeback, a => credit after chargeback)
$var['transaction_id']  = preg_replace('/[^a-z0-9-]/i', '', $_GET['transaction_id']);
// Transaction ID
```

```php
$var['uid']             = abs($_GET['userid']); // Unique customer-ID from your
system
$var['price_coins']     = abs($_GET['coins']); // Number of coins the customer should
receive
$var['price']           = floatval($_GET['amount'])*100; // Amount in Euro-Cent
$var['currency']        = preg_replace('/[^A-Z]/i', '', $_GET['currency']);//
Currency
$var['paytype']         = preg_replace('/[^a-z0-9-]/i', '', $_GET['paytype']);//
Payment method (short name)
$var['hash']            = trim($_GET["hash"]);


/** Check if hash is correct **/
/** ======================== **/
unset($_GET['hash']);
$parameter = array_filter($_GET, "strlen"); // remove empty GET parameters
ksort($parameter); //sorting alphabetical
$query = http_build_query($parameter, '&amp;'); // create http-query
$q_hash = hash("sha512", $query.$p4c_api_key); // create hash from query

// check GET-hash is equal to query-hash
if ($var['hash'] != $q_hash) {die('Hash incorrect!');}
/** === check hash - end === **/

/**
 * Here you can insert your code.
 * For example, you can credit the user's coins
 * **/
/*
echo 'Parameters:<br />';
echo '<pre>';
print_r($var);
echo '</pre>';
*/

/**
 * If all parameters have been processed correctly, you must issue an "OK"
 * **/

// p => payed (new transaction
if ($var['status'] == 'p') {
    echo "OK";

// c => transaction chargeback
} else if ($var['status'] == 'c') {
    echo "OK";

// a => credit after transaction chargeback
} else if ($var['status'] == 'c') {
    echo "OK";
}

?>
```

Test mode (API test)

Before we go live with your website, it is in test mode. The test mode allows you to use the payment method “prepayment” to resolve transactions that do not flow into the live system and therefore do not incur any costs. Prepayment is only simulated. The transaction is forwarded to your system as successful in real time. See also section: “Parameter transfer from us to your system”

Support & contact

If you need help, the Pay4Coins team is at your disposal.

Technical advice:
E-Mail: techsupport@pay4coins.com
,
General questions:
E-Mail support@pay4coins.com

Imprint

CIPA Media S.L.
Av. Emilio Luque Moreno 19
Oficina 6
38300 La Orotava (S/C de Tenerife)

Information for customers:
www.pay4coins.com

Handelsregister Teneriffa: VAT: CIF B76659168

Commercial Register Tenerife: VAT: CIF B76659168

The documentation including all published contents is protected by copyright. Reprinting or reproduction of any kind as well as processing, duplication and distribution using electronic systems in any form requires the prior written consent of CIPA MEDIA S.L.

The use of this documentation and the implementation of the information contained therein is expressly at your own risk. CIPA MEDIA S.L. does not assume any guarantee for the function of individual programs or parts thereof. In particular, CIPA MEDIA S.L. assumes no liability for any consequential damage resulting from use.

Changelog

2024-09-03 (Version 1.9)

- The parameter payment_method => “giropay” has been removed. Giropay will be discontinued at the end of 2024.
- The parameter payment_method => “bank_transfer” has been replaced by “sofort”.
- The parameter payment_method => “tink” has been added.
- The parameter amount was incorrectly documented and Integer corrected to Float.