pay|4coins

Pay4Coins
API Dokumentation

Inhaltsverzeichnis

Einführung in Pay4Coins	2
Integrationsschritte	2
Integration von Pay4Coins	3
Aufruf der API und Authentifizierung	3
Zahlungsdatenübergabe	3
Beispiel für einen Aufruf der Checkout-Page	5
Antwort der API verarbeiten	6
Parameterübergabe von uns an Ihr System	6
Einfaches Beispiel für die Hash-Prüfung (Authentifizierung)	7
Ausführliche Prüfung aller Parameter inkl. Hash-Prüfung	8
Testmodus (Schnittstelle testen)	10
Support & Kontakt	10
Impressum	10
Änderungsprotokoll	11

Einführung in Pay4Coins

Für die Zahlungsabwicklung mit Pay4Coins müssen sich Ihre Kunden nicht bei Pay4Coins registrieren. Sie bezahlen schnell und einfach mit den von Pay4Coins zur Verfügung gestellten Zahlarten. Dank der Echtzeitbestätigung der Zahlungen können Sie als Händler die Ware direkt versenden beziehungsweise die Coins sofort gutgeschrieben.

Eine Transaktion mit Pay4Coins gliedert sich aus Sicht Ihrer Kunden in folgende Schritte:

- Sobald ein Kunde mit Pay4Coins bezahlen möchte, wird er auf eine Seite von Pay4Coins geleitet.
- Unter der dortigen Übersicht aller angebotenen Zahlarten wird der Kunde aufgefordert, eine Zahlart zu wählen, mit der er bezahlen möchte.
- Anschließend muss Ihr Kunde wie gewohnt die Zahlung mit seiner gewählten Zahlungsart abwickeln.
- Sofern die Zahlung erfolgreich war, bekommt der Kunde einen Hinweis.

Integrationsschritte

Um Pay4Coins in Ihr System zu integrieren, sind allgemein folgende Schritte notwendig. Eine detaillierte Beschreibung der einzelnen Schritte folgt anschließend:

1. Registrieren Sie sich als Anbieter auf unserer Seite: https://merchant.pay4coins.com/Partner-werden
2. Integrieren Sie Pay4Coins in Ihren Shop.
3. Sobald ein Kunde mit Pay4Coins bezahlen möchte, rufen Sie unsere Schnittstelle mit den jeweiligen Zahlungsdaten auf. Als Antwort erhalten Sie einen Link auf unsere Checkout-Page, auf den Sie den Kunden zur Zahlung weiterleiten.
4. Der Kunde wickelt die Zahlung auf unserem System ab.
5. Im Fall einer erfolgreichen Transaktion werden Sie automatisch von Pay4Coins über diese Transaktion benachrichtigt. Anschließend können Sie sofort weitere Schritte, wie bspw. Warenversand oder Freischaltung zum Online-Angebot veranlassen oder Coins gutschreiben. Die Benachrichtigung der erfolgreichen Transaktion erfolgt dabei sowohl per E-Mail, als auch via GET-Benachrichtigung, die Ihr System automatisch über eine Statusänderung benachrichtigt.

Hinweis:

Bitte achten Sie bei der Integration darauf, dass eine Weiterleitung des Zahlenden auf das Zahlformular stattfinden muss, so dass die URL von Pay4Coins und das SSL-Zertifikat von CIPA Media sichtbar sind. Das heißt, eine eingeframte Lösung beispielsweise mit <iframe> ist aus rechtlichen Gründen nicht erlaubt.

Integration von Pay4Coins

Damit Ihre Kunden Pay4Coins nutzen können, müssen Sie in einem ersten Schritt Pay4Coins im Checkout Ihres Shops einbinden. Wählt der Kunde Pay4Coins und seine Zahlart und bestätigt seine Bestellung, beginnt die Kommunikation zwischen Ihrem Shop und unserer Schnittstelle (API).

Aufruf der API und Authentifizierung

Um Missbrauch der Schnittstelle zu verhindern, wird für jeden Aufruf der Schnittstelle eine Authentifizierung durchgeführt. Hierzu werden Ihre Shop-ID als Benutzername und der API-Key als Passwort übergeben.

Der Aufruf der Schnittstelle erfolgt über folgende URL:

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

Zahlungsdatenübergabe

Diese Tabelle enthält alle möglichen Parameter, die Sie in einer Anfrage an unser System übergeben können.

Pflichtparameter		
Parameter	**Typ**	**Erklärung**
api_key	String	Ihr Pay4Coins-API-Key
api_url	String	URL zu Ihrer Schnittstelle, an den wir den Zahlungsstatus der Transaktion senden.
api_version	String	Bitte „**v1**“ eintragen.
coins	Integer	Anzahl der Coins die der Kunde erhalten soll
email	String	E-Mail-Adresse Ihres Kunden Ihr Kunde wird von uns per E-Mail über eine erfolgreiche Transaktion informiert.
hash	String	Hash aus allen angegebenen Parametern außer dem Parameter „hash“ selbst.
lang	String (2)	Gibt die Sprache an, in welcher die Checkout-Page geladen werden soll. Bsp: „de“ oder „en“. Wird dieser Parameter nicht angegeben, wird automatisch die eingestellte Browsersprache des Users genutzt.
member_id	Integer	Eindeutige ID Ihres Kunden

name	String	Name des Artikel den der Kunde kauft. zum Beispiel: „2000 Coins“
price	Integer	Betrag des Artikels in Euro-Cent
shop_id	Integer	Ihre Pay4Coins Shop-ID
username	String	Benutzername Ihres Kunden
Optionale Parameter		
error_url	String	URL, auf die Ihr Kunde weitergeleitet werden soll, wenn die Transaktion abgebrochen wird. Wird dies nicht angegeben, wird dem Kunden eine Seite von uns angezeigt.
success_url	String	URL, auf die Ihr Kunde weitergeleitet werden soll, wenn die Transaktion erfolgreich abgeschlossen wurde. Wird dies nicht angegeben, wird dem Kunden eine Erfolgsseite von uns angezeigt.
created_at	Integer	Gibt den aktuellen Zeitpunkt in Unix-Zeitstempeln (in Sekunden) an, an dem die URL erstellt/generiert wurde.
expires	Integer	Gibt die Dauer in Stunden an, wie lange die URL gültig sein soll. (von 1 bis 168)
payment_method	String	Wenn Sie diesen Parameter senden, wird der Kunde direkt zu der von Ihnen übergebenen Zahlungsart geleitet. Ansonsten wird ihm eine Übersicht aller Zahlungsarten angezeigt. **code = Bezahlmethode** cashtocode = CashtoCode creditcard = Kreditkarte crypto = Kryptowährung (Bitcoin, ETH, LTC usw) directdebit = SEPA - Lastschrift ideal = iDeal neosurf = Neosurf paysafecard = paysafecard phone_landline = Telefon Festnetz phone_mobile = Telefon Mobil prepay = Vorkasse / Vorabüberweisung przelewy24 = Przelewy24 sofort = Sofortüberweisung (Klarna) tink = Tink
Mit diesen Parametern können Sie weitere beliebige Werte, über unsere API, an Ihr System zurückgeben.		

user_variable_1	String	
user_variable_2	String	
user_variable_3	String	
user_variable_4	String	
Parameter für „AdKlick“ Diese benötigen Sie nur, wenn Sie mit dem Partnerprogramm AdKlick arbeiten.		
adklick_pid	String	Ihre AdKlick „PID“
adklick_tid	String	Ihre AdKlick „TID“

Beispiel für einen Aufruf der 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;

?>
```

Die generierte URL würde in dem Beispiel wie folgt aussehen:

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

Antwort der API verarbeiten

Nach erfolgreicher Zahlung, benachrichtigt unser System Ihr System per GET-Benachrichtigung über den neuen Status. Sie müssen diese Benachrichtigung verarbeiten. Um Missbrauch der Schnittstelle zu verhindern, wird für jeden Aufruf Ihrer Schnittstelle eine Authentifizierung durchgeführt. Hierzu wird der API-Key als Passwort übergeben.

Parameterübergabe von uns an Ihr System

Nach erfolgreicher Zahlung übermittelt unser System an Ihr System per GET die folgenden Parameter. Für die Übermittlung haben Sie uns Ihre API-URL mitgeteilt.

Wichtig!:

Ihr System muss nach erfolgreicher Verarbeitung aller Parameter mit „**OK**“ an uns antworten!

Beispiel Ihrer API-URL die Sie uns mitgeteilt haben:

```
https://ihre-webseite.com/pay4coins_api/index.php
```

Pflichtparameter		
Parameter	**Typ**	**Erklärung**
amount	Float	Artikelpreis in Euro & Cent (Bsp.: 19.99 für 19,99 EUR)
coins	Integer	Anzahl der Coins die der Kunde erhalten soll
currency	String (3)	Währung (EUR oder USD)
event	String (1)	p => neue Buchung (payed) c => Rückbuchung/Storno (chargeback) a => Gutschrift nach Rückbuchung
hash	Integer	Hash aus allen angegebenen Parametern außer dem Parameter „hash“ selbst.
paytype	String	**ccard** => Kreditkarte

		ideal => iDEAL **neosurf** => Neosurf **paybycall** => PayByCall (Bezahlen per Telefon) **paysafecard** => paysafecard **prepay** => SEPA-Überweisung (Vorkasse) **sepadd** => SEPA Lastschrift **sofort** => SOFORT Überweisung (KLARNA.)
transaction_id	String	Transaktionsnummer der Zahlung
userid	Integer	Eindeutige ID Ihres Kunden
Optionale Parameter Sofern Sie uns diese Parameter übermittelt haben, gibt unser System diese wieder an Ihr System zurück.		
user_variable_1	String	
user_variable_2	String	
user_variable_3	String	
user_variable_4	String	

Einfaches Beispiel für die Hash-Prüfung (Authentifizierung)

```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';

?>
```

Ausführliche Prüfung aller Parameter inkl. Hash-Prüfung

```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',
    'created_at',
    'expires',
    'coins',
    '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";
}

?>
```

Testmodus (Schnittstelle testen)

Bevor wir Ihre Website bei uns live schalten, befindet sich diese im Testmodus.
Der Testmodus erlaubt es Ihnen, mit Hilfe der Zahlart “Vorkasse”, Transaktionen auszulösen die nicht in das Live-System fließen und Ihnen somit keine Kosten entstehen. Es Vorkasse wird nur simuliert. Die Vorkasse wird als erfolgreiche Transaktion, in Echtzeit, an Ihr System weitergeleitet. Siehe dazu auch Abschnitt: “Parameterübergabe von uns an Ihr System”

Support & Kontakt

Sollten Sie Hilfe benötigen, steht Ihnen das Team von Pay4Coins zur Verfügung.

Technische Beratung:
E-Mail: techsupport@pay4coins.com

Allgemeine Fragen:
E-Mail support@pay4coins.com

Impressum

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

Informationen für Kunden:
www.pay4coins.com

Handelsregister Teneriffa: VAT: CIF B76659168

Die Dokumentation einschließlich aller veröffentlichten Inhalte ist urheberrechtlich geschützt. Nachdruck oder Reproduktion jeglicher Art sowie die Verarbeitung, Vervielfältigung und Verbreitung unter Verwendung elektronischer Systeme in irgendeiner Form bedarf der vorherigen schriftlichen Genehmigung der CIPA MEDIA S.L.

Die Verwendung dieser Dokumentation und die Umsetzung der darin enthaltenen Angaben erfolgt ausdrücklich auf eigenes Risiko. Die CIPA MEDIA S.L. übernimmt keine Gewähr für die Funktion einzelner Programme oder von Teilen derselben. Insbesondere übernimmt die CIPA MEDIA keine Haftung für etwaige, aus dem Gebrauch resultierende Folgeschäden.

Änderungsprotokoll

2024-09-03 (Version 1.9)

- Der Parameter *payment_method => "giropay"* wurde entfernt. Giropay wird Ende 2024 eingestellt.
- Der Parameter *payment_method => "bank_transfer"*, wurde durch "sofort" ersetzt.
- Der Parameter *payment_method => "tink"* wurde hinzugefügt.
- Der Parameter *amount* war falsch dokumentiert und Integer in Float korrigiert.