

**API Dokumentation Stand 01.2013**

Die API nutzt zur einfachen, sicheren Abwicklung die Weiterleitung auf beim PSP gehostete Eingabeformulare.

Die Übergabe der Formulardaten muss mittels **HTTP POST** erfolgen.

**Pflichtfelder**

Feldname	Beschreibung, mögliche Werte und Formatangaben	„Sale“ Transaktion	Vorautorisierung (nur Kreditkarte)	Buchung einer Vorautorisierung (nur Kreditkarte)	Storno
txntype	'sale', 'preauth', 'postauth' oder 'void' (Der Transaktionstyp - bitte <i>Die Möglichkeit ‚void‘ zu senden ist eingeschränkt. Bitte wenden Sie sich an das Support Team, wenn Sie diese Option nutzen wollen.</i> )	X (sale)	X (preauth)	X (postauth)	X (void)
timezone	GMT, CET oder EET (Die Zeitzone der Transaktion)	X	X	X	X
txndatetime	YYYY:MM:DD-hh:mm:ss (Der genaue Zeitpunkt der Transaktion)	X	X	X	X
hash	Dies ist ein SHA1 Hash der folgenden Felder: storename + txndatetime + chargetotal + currency + sharedsecret. Beachten Sie, dass der Hash in genau dieser Reihenfolge zusammengestellt wird. Weiter unten finden Sie ein Beispiel für die Generierung eines SHA1 Hash.	X	X	X	X
storename	Ihre Shop-ID, (wird vom PSP mitgeteilt.)	X	X	X	X
mode	'fullpay', 'payonly' oder 'payplus' <b>(Hier ist payonly zu verwenden)</b>	X	X		

chargetotal	Der Gesamtbetrag der Transaktion. Als Dezimal-Trennzeichen muss ein Komma oder Punkt verwendet werden (z. B. 12,34 für 12 Euro und 34 Cent). Tausender-Trennzeichen (z. B. 1.000 EUR) sind nicht zugelassen.	X	X	X	X
currency	Die Währung der Transaktion als dreistelliger numerischer ISO 4217 Wert (siehe Tabelle Währung unten).	X	X	X	
oid	Bestellnummer (Order ID) des Vorgangs zu dem eine Buchung der Vorauforderung oder ein Storno erfolgen soll			X	X
tdate	Genauere Kennzeichnung der Transaktion eines Vorgangs, die storniert werden soll. Diesen Wert erhalten Sie als Antwortparameter 'tdate' zur entsprechenden Transaktion				X
paymentMethod	MasterCard 'M' Visa 'V' AmEx 'A' Diners 'C' JCB 'J' Lastschrift 'debitDE' giropay 'giropay' Maestro 'MA' ClickandBuy 'clickAndBuy'	X	X	X	
responseSuccessURL	Vom Shop übergebener Wert wohin der Kunde nach erfolgreicher Transaktion zurückgeleitet wird	X	X	X	
responseFailURL	Vom Shop übergebener Wert wohin der Kunde bei fehlgeschlagener Transaktion umgeleitet wird (kann identisch zur responseSuccess URL sein)	X	X	X	

### Hash Wert Berechnug

Zur Absicherung der Transaktion und des Transaktionsergebnisses wird ein Hash-Wert berechnet. Der Hashwert setzt sich aus verschiedenen Parametern zusammen. Die Parameter zuerst als String verkettet. Hierbei ist zu beachten, dass auch der Betrag hier als String verarbeitet wird. Dieser String wird anschließend stellenweise in den entsprechenden Hexadezimalwert gewandelt und dann mit dem SHA-1 in den Hash gewandelt.

Beispiel einer HASH-Berechnung:

storename: 12066666666

txndatetime: 2010:05:06-12:37:43

chargetotal: 13.00

currency: 978

sharedsecret: ganzGeheim

wird zu folgendem String zusammengesetzt:

120666666662010:05:06-12:37:4313.00978ganzGeheim

Dann findet eine Umwandlung in einen Hexadezimal-String statt:

3130313233343536373839323031303a30353a30362d31323a33373a343331332e30

3039373867616e7a47656865696d

Nun wird der SHA-1 HASH gebildet:

5f243d63a4f96391a31ec8a1943be1e36f12aa9d

### Ablauf der Zahlungsabwicklung:

Der Kunde bestätigt, nachdem er vorher die für ihn entsprechende Zahlungsmethode ausgewählt hat, im Shop seinen Einkauf durch Klick auf den „Kaufen/verbindlich/kostenpflichtig Bestellen“ Button und wird dann wahlweise per iFrame oder per neuer Seite auf das sichere Zahlungsformular des PSP weitergeleitet, hierzu benötigt der PSP die ausgewählte Zahlungsart um eine entsprechende Seite anzeigen zu können.

hat der Kunde seine Zahlungsinformationen eingegeben erfolgt die Prüfung durch den PSP, je nach Ergebnis erfolgt die entsprechende Weiterleitung (responseSuccessURL responseFailURL)

Der Shop muss in diesen Fällen entsprechend reagieren, (Success = Auftragsstatus setzen, Kunde informieren das bezahlt wurde / Fail = Kunde informieren das er es nochmal versuchen soll, bzw. eine andere Zahlungsart wählen soll.

### Optionale Formularfelder

**oid** (max. 100 Zeichen)

Dieses Feld ermöglicht Ihnen, eine eindeutige Kennung für eine Bestellung zu vergeben. Mit der Erweiterungsfunktion „Debit Text“ kann diese Angabe bei Lastschrifttransaktionen an die Bank des Kunden weitergeleitet werden, sodass sie auf dem Kontoauszug Ihres Kunden ausgewiesen werden kann. Die ersten 30 Zeichen der Bestellnummer werden bei Kreditkartenzahlungen mit an Ihren Acquirer übertragen und können eventuell mit auf der Händlerabrechnung ausgewiesen werden.

Wenn Sie selbst keine Bestellnummer vergeben möchten, generiert das System automatisch eine für Sie.

**customerid** (max. 32 Zeichen)

Dieses Feld ermöglicht Ihnen einen beliebigen Wert, z. B. eine Kennung für den Kunden zu übergeben. Mit der Erweiterungsfunktion „Debit Text“ kann diese Angabe bei Lastschriftransaktionen an die Bank des Kunden weitergeleitet werden, sodass sie auf dem Kontoauszug Ihres Kunden ausgewiesen werden kann. Die ersten 30 Zeichen der Kundennummer werden bei Kreditkartenzahlungen mit an Ihren Acquirer und die Bank des Karteninhabers übertragen und können eventuell mit auf der Händlerabrechnung, bzw. Karteninhaberabrechnung ausgewiesen werden.

**invoicenumber** (max. 48 Zeichen)

Dieses Feld ermöglicht Ihnen einen beliebigen Wert, z. B. eine Rechnungsnummer oder Warengruppe zu übergeben.

**refer** (max. 2000 Zeichen)

Dieses Feld gibt an, wer dem Kunden Ihren Shop empfohlen hat.

**comments** (max. 2000 Zeichen)

Hier tragen Sie mögliche Anmerkungen zur Transaktion ein.

**transactionNotificationURL**

Die URL, an die das Transaktionsergebnis direkt mitgeteilt werden soll

**language**

Mit der Angabe einer Sprache für die Zahlseite können Sie ausländischen Kunden die Zahlseite in deren Sprache anzeigen. Die für Ihren Shop hinterlegte Standardsprache wird dabei überschrieben. Aktuell sind folgende Sprachen wählbar:

Sprache	language
Deutsch	de_DE
Englisch (USA)	en_US
Englisch (Großbritannien)	en_GB
Französisch	fr_FR
Italienisch	it_IT

Optionale Erweiterung Data Storage

Mit der Erweiterungsfunktion Data Storage können Kreditkartendaten und Bankverbindungen der Kunden in einer verschlüsselten Datenbank beim PSP gespeichert werden, um auf diese für spätere Zahlungen zurückgreifen zu können.

Wenn Sie diese Funktion vom Kunden beauftragt wurde, bietet Ihnen die Schnittstelle die folgenden Funktionen:

- Speicherung oder Aktualisierung der Zahlungsinformationen bei einer Transaktion  
Übermitteln Sie bei einer Zahlungstransaktion zusätzlich den Parameter

**hosteddataid** (max. 128 Zeichen) mit einer eindeutigen Kennung für die zu hinterlegenden

Zahlungsinformationen. Je nach Zahlart werden bei einer erfolgreichern Zahlungstransaktion unter dieser Kennung Kartenummer und Verfalldatum bzw. Kontonummer und Bankleitzahl hinterlegt. Ist die ‚hosteddataid‘ bereits vorhanden, werden die gespeicherten Informationen überschrieben.

- Zahlungstransaktionen mit hinterlegten Daten

Wurden zu einem Kunden Zahlungsinformationen hinterlegt, können Sie unter Angaben der ‚hosteddataid‘ Zahlungstransaktionen durchführen ohne dass eine erneute Angabe von Kartenummer und Verfalldatum bzw. Kontonummer und Bankleitzahl notwendig sind. Bitte beachten Sie, dass die Kartenprüfnummer (meistens auf der Rückseite der Kreditkarte) nicht gespeichert werden darf und daher bei Kreditkartenzahlungen zusätzlich vom Kunden angegeben werden muss.

Zur Vermeidung, dass Kunden sich bei Ihnen mehrmals mit denselben Zahlungsinformationen registrieren können, besteht die Möglichkeit, bei der Speicherung den zusätzlichen Parameter **declineHostedDataDuplicates** zu übermitteln. Gültige Werte für diesen Parameter sind ‚true‘ und ‚false‘. Wird der Wert ‚true‘ übergeben und die übermittelten Zahlungsinformationen liegen bereits unter einer anderen ‚hosteddataid‘ vor, wird der Vorgang abgelehnt.

### Transaktionsergebnis

Nach der Verarbeitung einer Transaktion werden die Transaktionsdaten als versteckte Felder an die von Ihnen definierte responseSuccessURL oder responseFailURL geschickt:

Feldname	Beschreibung
approval_code	Genehmigungscode der Transaktion Das erste Zeichen kennzeichnet das Transaktionsergebnis: ,Y‘ - Transaktion erfolgreich ,N‘ - Transaktion nicht erfolgreich “?” - Transaktion wurde erfolgreich initialisiert, und noch nicht finalisiert. Transaktion wird zum späteren Zeitpunkt aktualisiert.
oid	Bestellnummer (Order ID)
refnumber	Referenznummer
status	Transaktionsstatus

txndate_processed	Zeitpunkt der Transaktionsverarbeitung		
tdate	Transaktionskennung mit der die Transaktion ggf. später storniert werden kann		
fail_reason	Grund für ein etwaiges Scheitern der Transaktion		
response_hash	Hash-Wert zur Absicherung der Kommunikation (siehe Hinweise unten)		
processor_response_code	Der Antwortcode der Transaktionsverarbeitung  Bitte beachten Sie, dass diese Antwortcodes je nach Zahlungsart und entsprechendem Hintergrundsystem unterschiedlich sein können. Während bei Kreditkartenzahlungen z. B. der Antwortcode ‚00‘ auf eine erfolgreiche Genehmigung hinweist, ist der vom Betreibersystem übermittelte Antwortcode für erfolgreiche giro-pay-Transaktionen ‚4000‘.		
fail_rc	Internes Verarbeitungsmerkmal bei fehlgeschlagenen Transaktionen		
terminal_id	Die Terminal ID, die für die Transaktion verwendet wurde		
ccbin	6-stellige Kennung der kartenherausgebenden Bank		
cccountry	3-stelliger ISO Code für das Land des Karteninhabers (z.B. USA, DEU, ITA, etc.) "N/A", sofern das Land nicht ermittelt werden kann oder keine Kreditkarte genutzt wird.		
ccbrand	Brand of the credit or debit card: <table border="1" data-bbox="683 1928 1050 2011"> <tr> <td>MC</td> </tr> <tr> <td>VISA</td> </tr> </table>	MC	VISA
MC			
VISA			

	AMEX
	DINERS/DISCOVER
	JCB
	UNIONPAY
	MAESTRO
	LASER
	"N/A" wenn keine Kreditkarte

Zusätzlich werden an die spezifische URL Ihre selbst definierten Felder und die Felder mit Liefer- und Rechnungsdaten gesendet.

Der Parameter **response\_hash** ermöglicht Ihnen zu überprüfen, ob das empfangene Transaktionsergebnis auch tatsächlich vom PSP stammt und kann Sie somit vor betrügerischen Manipulationen schützen.

Erzeugen Sie dazu einen SHA 1 Hash aus dem Parameter-String

sharedsecret + approval\_code + chargetotal + currency + txndatetime + storename

**Bei der Hash-Berechnung ist der Parameter chargetotal in Formtierung Sprachabhängig.**

**Bitte verwenden Sie für die HASH-Berechnung, die Formatierung, wie sie in der Antwort zurück kommt.**

Beispiel Transaktionen:

#### 1. Erfolgreiche Transaktion

##### SHOP POST TO PAYMENT GATEWAY

```

chargetotal = 2,00
currency = 978
oid = 12345-0001-2013:04:10-14:12:42
responseSuccessURL = http://127.0.0.1/minishopv2/ipg-response.php
responseFailURL = http://127.0.0.1/minishopv2/ipg-response.php
timezone = CET
storename = 10123456789
txntype = sale
mode = payonly
txndatetime = 2013:04:10-14:12:42
hash = c7658686a0e81a8727afb1bca33dc7ad9d07b313
paymentMethod =
customerid = 0001
invoicenumber = 12345
authenticateTransaction = true
language = de_DE
    
```

**RESPONSE FROM PAYMENT GATEWAY**

```
response_code_3dsecure = 7
expmonth = 05
terminal_id = 54000015
txndatetime = 2013:04:10-14:39:05
status = GENEHMIGT
cardnumber = (VISA) ... 1111
invoicenummer = 12345
currency = 978
approval_code = Y:SM8250:0013399515:PPXM:3882500273
processor_response_code = 00
expyear = 2014
refnumber = 888250
ccbrand = VISA
cccountrry = N/A
timezone = CET
chargetotal = 2,00
customerid = 0001
txntype = sale
oid = 12345-0001-2013:04:10-14:39:05
ccbin = 411111
tdate = 1365597556
txndate_processed = 10.04.13 14:39:16
response_hash = 1500edc32fc2e9d61cb13cc29f582fee4ba4e767
paymentMethod = V
```



## 2.ZURÜCKGEWIESENE TRANSAKTION

### POST TO PAYMENT GATEWAY

```
chargetotal = 2,00
currency = 978
oid = 12345-0001-2013:04:10-14:40:49
responseSuccessURL = http://127.0.0.1/minishopv2/ipg-response.php
responseFailURL = http://127.0.0.1/minishopv2/ipg-response.php
timezone = CET
storename = 10123456789
txntype = sale
mode = payonly
txndatetime = 2013:04:10-14:40:49
hash = 6f5f56a7c307b0d2d2882faa12d10b4e65562fc2
paymentMethod =
customerid = 0001
invoicenumber = 12345
authenticateTransaction = true
language = de_DE
```

### RESPONSE

```
response_code_3dsecure = 7
expmonth = 11
terminal_id = 54000015
invoicenumber = 12345
fail_reason = Ungültige Bankverbindung
currency = 978
approval_code = N:01:Unknown Processor Response Code
chargetotal = 2,00
timezone = CET
cccountry = N/A
oid = 12345-0001-2013:04:10-14:42:45
ccbin = 411111
tdate = 1365597762
fail_rc = 1
response_hash = e74c4fc9d4c787163b2b252e79d56cdf207fcb3e
txndatetime = 2013:04:10-14:42:45
status = ABGELEHNT
cardnumber = (VISA) ... 1111
processor_response_code = 01
expyear = 2014
ccbrand = VISA
customerid = 0001
txntype = sale
txndate_processed = 10.04.13 14:42:42
paymentMethod = V
```

### 3. ERFOLGREICHE TRANSAKTION LASTSCHRIFT

#### POST TO PAYMENT GATEWAY

```
chargetotal = 2,00
currency = 978
oid = 12345-0001-2013:04:10-15:10:06
responseSuccessURL = http://127.0.0.1/minishopv2/ipg-response.php
responseFailURL = http://127.0.0.1/minishopv2/ipg-response.php
timezone = CET
storename = 10123456789
txntype = sale
mode = payonly
txndatetime = 2013:04:10-15:10:06
hash = da1573f87ac0ae2d87038f70fd8ed5c2aec3dc66
paymentMethod = debitDE
customerid = 0001
invoicenumber = 12345
authenticateTransaction = true
language = de_DE
```

#### RESPONSE

```
terminal_id = 54000015
txndatetime = 2013:04:10-15:10:57
status = GENEHMIGT
invoicenumber = 12345
currency = 978
processor_response_code = 00
approval_code = Y:000000:0013399673:PPXM:3882620283
refnumber = 000000
ccbrand = N/A
cccountry = N/A
timezone = CET
chargetotal = 2,00
customerid = 0001
bankcode = 12070024
txntype = sale
oid = 12345-0001-2013:04:10-15:10:57
tdate = 1365599499
accountnumber = 5884
txndate_processed = 10.04.13 15:11:39
response_hash = fd7cec3a75cbdaad8d1bcba921d980035e38fdc1
paymentMethod = debitDE
```

### 4. FEHLGESCHLAGENE TRANSAKTION LASTSCHRIFT

#### RESPONSE FROM PAYMENT GATEWAY

```
txndatetime = 2013:04:10-15:13:49
status = FEHLGESCHLAGEN
invoicenummer = 12345
fail_reason = Ungültige Bankverbindung
currency = 978
approval_code = N:-1:Invalid account data
ccbrand = N/A
timezone = CET
chargetotal = 2,00
cccountry = N/A
customerid = 0001
bankcode = 50010060
txntype = sale
oid = 12345-0001-2013:04:10-15:13:49
tdate = 1365599644
accountnumber = 5676
fail_rc = 1
txndate_processed = 10.04.13 15:14:04
response_hash = d9311ea06c04ef11d2501315945a1c959718b0a2
paymentMethod = debitDE
```

### Beispiel Script für den Datenaustausch in PHP

Das folgende PHP Script überibt den notwendigen Mindestbestand an Daten an die pay4one Connect Schnittstelle weiter.

```
<? include("ipg-util.php"); ?>

<html>
<head><title>IPG Connect Sample for PHP</title></head>
<body>
<p><h1>Order Form</h1>

<form method="post"
action="https://test.paymentserverURL.com"> <input
type="hidden" name="txntype" value="sale">
  <input type="hidden" name="timezone" value="CET"/>
<input type="hidden" name="txndatetime" value="<?php
echo getDateTime() ?>"/>
<input type="hidden" name="hash" value="<?php echo
createHash( "13.00", "978" ) ?>"/>
<input type="hidden" name="storename"
value="12066666666"/> <input type="hidden" name="mode"
value="fullpay"/>
<input type="text" name="chargetotal" value="13.00"/>
<input type="hidden" name="currency" value="978"/>

  <input type="submit" value="Submit">
</form>
</body>
</html>
```

Auf der nächsten Seite finden Sie den Quellcode für die im Beispiel verwendete ipg-util.php, hier wurden die Funktionen ausgelagert.

```
<?php
    $dateTime = date("Y:m:d-H:i:s");

    function getDateTime() {
        global $dateTime;
        return $dateTime;
    }

    function createHash($chargetotal, $currency) {
        $storename = "10123456789";
        $sharedSecret = "sharedsecret";

        $stringToHash = $storename . getDateTime() . $chargetotal .
            $currency . $sharedSecret;
```