

# Classic Payment API

## SOPG (Service Oriented Prepaid Gateway – xml-basiertes Protokoll) Dokumentation

### Versionshistorie

Version	Datum	Beschreibung	Autor
0.1	10.03.2013	Erster Entwurf	Paul Kneidinger
0.2	20.03.2013	Details hinzugefügt und umstrukturiert	Matthias Vilsecker
0.3	21.03.2013	In App Payment hinzugefügt	Matthias Vilsecker
0.4	24.04.2013	Prüfung und weitere Ergänzung	Natasa Jeremic
1.0	02.10.2013	Finale Version	Natasa Jeremic
1.1	27.01.2014	Kleinere Änderungen	Natasa Jeremic
1.2	31.10.2017	Aktualisiert	Eugen Nemecek
1.3	22.01.2020	MCID Anforderungen ergänzt	Lucas Hannel

Mit technischen Fragen zur Implementierung wenden Sie sich bitte an [integration@paysafecard.com](mailto:integration@paysafecard.com)

## Inhaltsverzeichnis

<b>1. Einführung</b>	<b>3</b>
1.1 Produkteinführung für Kunden	3
1.1.1 paysafecard	3
1.1.2 my paysafecard	3
<b>2. Funktionsübersicht Zahlung</b>	<b>4</b>
2.1 Ablauf von Zahlungen	4
<b>3. Über SOPG</b>	<b>4</b>
3.1 Voraussetzungen	4
3.2 Umgang mit Fehlern	5
3.2.1 Beschreibung Fehlermeldungen	5
3.3 Inhaltstyp und Zeichensatz	5
<b>4. Definition von paysafecard Systemen</b>	<b>5</b>
4.1 API Service Umgebung	5
<b>5. Technische Übersicht</b>	<b>6</b>
<b>6. Funktionsdetails und WSDL-Struktur</b>	<b>7</b>
6.1 Funktionen createDisposition	7
6.2 getSerialNumbers	7
6.3 executeDebit	8
6.4 Zahlungsbenachrichtigung (payment notification)	8
6.4.1 Wiederholung der Zahlungsbenachrichtigung	8
<b>7. Zahlungsablauf</b>	<b>9</b>
<b>8. Parameter</b>	<b>10</b>
8.1. Parameter-Beschreibung	10
8.2. Einschränkungen	12
8.3. Disposition Status	13
8.3.1 Dispositionszeitfenster (disposition time frame)	13
8.4. Transaktionszeitfenster (payment time window)	13
8.5. Details Zahlungsfenster	13
8.5.1 Zahlungsfenster Desktop	13
8.5.2 Zahlungsfenster Mobilgeräte	13
8.6. Gebietsschema- und Spracheinstellungen	14
<b>9. Beispielzahlung</b>	<b>14</b>
9.1 createDisposition	14
9.2 getCustomerPanel	15
9.3 pnUrl request	15
9.3.1 In Zahlungsbenachrichtigungen unterstützte Ländercodes	16
9.4 getSerialNumbers	16
9.5 executeDebit	17
<b>10. Fehlercodes</b>	<b>18</b>

## 1. Einführung

Dieses Dokument bietet einen detaillierten Überblick über Verwendung und Parameter des Service Oriented Prepaid Gateway (SOPG) von paysafecard. Das Gateway ist ein SOAP-basierter XML-Webservice, der Zugriff auf das API-Clientfunktionen bietet, die mit jedem SOAP-kompatiblen Clientsystem genutzt werden können. Im ersten Kapitel finden Sie eine Funktionsübersicht des Zahlungsprozesses. Im Anschluss wird paysafecard SOPG im Detail vorgestellt. Es folgt eine Darstellung der Systeme von paysafecard sowie der technischen Details einschließlich einer vollständigen Beschreibung der Funktionen und Parameter.

### 1.1 Produkteinführung für Kunden

Dieser Abschnitt enthält eine Übersicht der Kundenprodukte von paysafecard, die im Zusammenhang mit der paysafecard SOPG Schnittstelle relevant sind.

#### 1.1.1 paysafecard

Paysafecard ermöglicht Kunden, ohne Bankkonto oder Kreditkarte sicher im Internet zu bezahlen. Der Kunde erwirbt paysafecard an einer Verkaufsstelle in Form eines Ausdrucks mit einem 16-stelligen PIN-Code und zahlt, indem der den PIN-Code in das paysafecard Zahlungsfenster in Ihrem Onlineshop eingibt.

#### 1.1.2 my paysafecard

Neben der Möglichkeit, mit einer PIN zu bezahlen, bietet paysafecard seinen Nutzern außerdem my paysafecard an, ein Konto für die PINs des Kunden. Der Kunde registriert sich für my paysafecard und lädt das Konto mit gekauften paysafecard PINs auf. Das selbe gilt für das my paysafecard administration tool\*. Der Kunde kann dann über die Summe der Werte aller paysafecard PINs auf dem my paysafecard verfügen. Gezahlt wird einfach und sicher durch die Eingabe von Benutzername und Passwort.

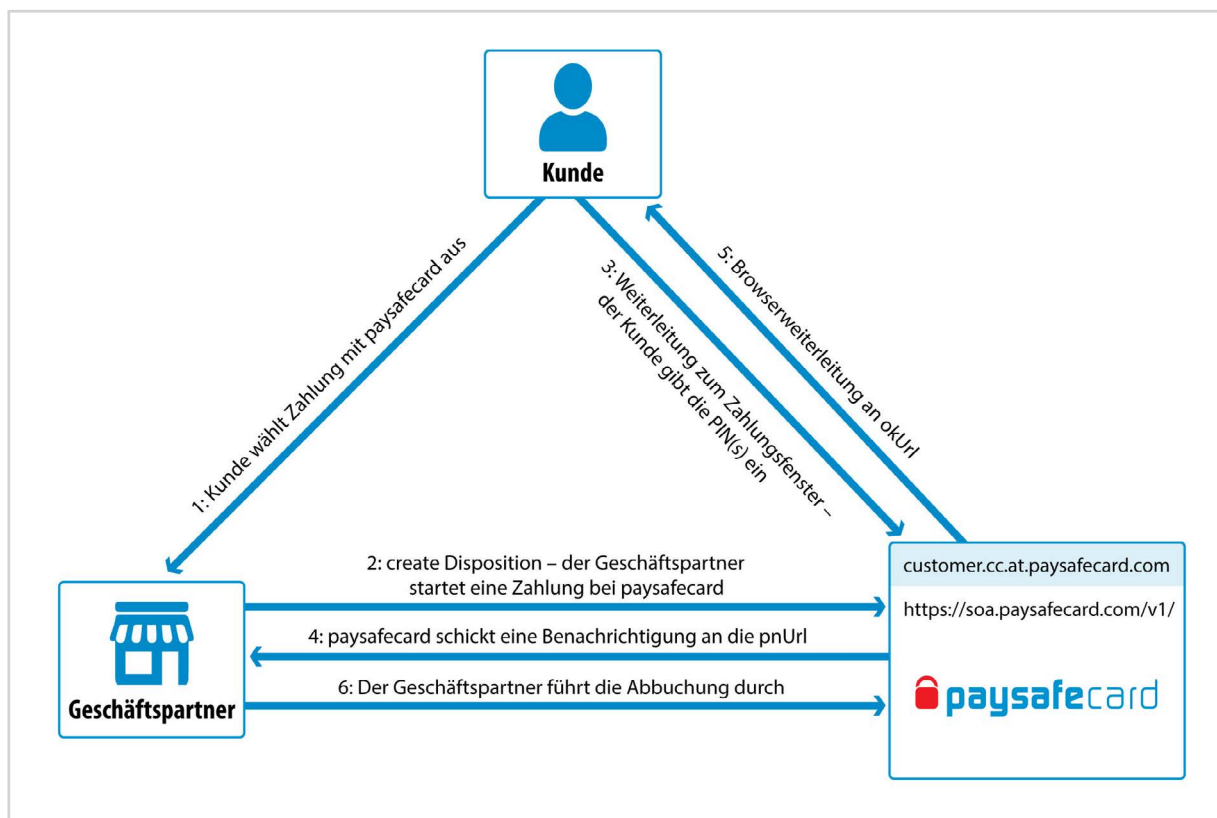
- Das my paysafecard administrations tool ist die Alternative für die klassische my paysafecard account und ausschließlich in den Ländern Australien, Canada, Litauen, Mexico, Neuseeland, Uruguay nutzbar. Das my paysafecard administration tool kann nicht für die Produkte payout oder refund verwendet werden.



## 2. Funktionsübersicht Zahlung

In jede Zahlung sind drei Parteien involviert: Ein Kunde, ein Geschäftspartner und das paysafecard Unternehmen („PSC“). Zahlungen erfolgen als „Zahlungstransaktionen“ oder „Dispositionen“, die durch eine „Merchant Transaction ID“ () eindeutig gekennzeichnet sind und einen Wert enthalten, der als „Betrag“ bezeichnet wird und üblicherweise der Geldsumme entspricht, mit der ein Kunde etwas erwirbt.

### 2.1 Ablauf von Zahlungen



## 3. Über SOPG

Über das Service Oriented Prepaid Gateway (SOPG) werden die Zahlungsfunktionalitäten von paysafecard als Webservice angeboten. Das Webservice basiert auf dem SOAP-Protokoll und kann, unabhängig von der Programmiersprache, von jedem SOAP-Client genutzt werden. Der komplette Zahlungsprozess wird zwischen dem System des Geschäftspartners und dem paysafecard SOPG abgewickelt.

### 3.1 Voraussetzungen

Ein Geschäftspartner kann sich nur dann mit den Systemen von paysafecard verbinden, wenn folgende Voraussetzungen erfüllt sind:

- Er besitzt von paysafecard vergebene SOPG User Daten (Benutzername/Passwort)
- Die IP-Adresse des Zahlungsservers wurde autorisiert (wird beim Versuch, auf den Dienst zuzugreifen, ein „Fehler 403“ ausgegeben, ist anzunehmen, dass die IP-Adresse noch nicht freigeschaltet wurde).

### 3.2 Umgang mit Fehlern

Zu allen SOPG-Funktionen werden „ErrorCode“ und „ResultCode“ ausgegeben. Der „ResultCode“ kann folgende Werte annehmen: „0“ (erfolgreich), „1“ (logisches Problem) oder „2“ (technisches Problem).

Generell gelten folgende Regeln:

- „1“ deutet darauf hin, dass ein Problem mit den übermittelten Daten vorliegt (z. B. falsche Login Daten, Transaktionszeit abgelaufen etc.). Ein erneuter Versuch mit denselben Daten wird nicht erfolgreich sein.
- „2“ (technisches Problem) bedeutet, dass der Service vorübergehend nicht erreichbar ist – der Request kann erneut übermittelt werden.

#### Nachstehend ein Beispiel für einen fehlerhaften Request:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="urn:psscservice">
  <soap:Body>
    <ns1:createDispositionResponse>
      <ns1:createDispositionReturn>
        <ns1:transactionID1234</ns1:transactionID>
        <ns1:subId>Merchant1234</ns1:subId>
        <ns1:mid xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>
        <ns1:resultCode>1</ns1:resultCode>
        <ns1:errorCode>10008</ns1:errorCode>
      </ns1:createDispositionReturn>
    </ns1:createDispositionResponse>
  </soap:Body>
</soap:Envelope>
```

#### 3.2.1 Beschreibung Fehlermeldungen

Bitte bedenken Sie, dass Fehlermeldungen für Kunden nur im paysafecard Zahlungsfenster angezeigt werden. In allen anderen Meldungen wird nur der Fehlercode angezeigt. %1, %2, ... sind Platzhalter für verschiedene Werte, z.B. , MID. Die gängigsten Fehlercodes finden Sie in Kapitel 10.

### 3.3 Inhaltstyp und Zeichensatz

Bitte achten Sie beim Übermitteln von Requests darauf, dass im HTTP-Header folgende Werte gesetzt sind:  
Content-Type: text/xml; charset=UTF-8

## 4. Definition von paysafecard Systemen

### 4.1 API Service Umgebung

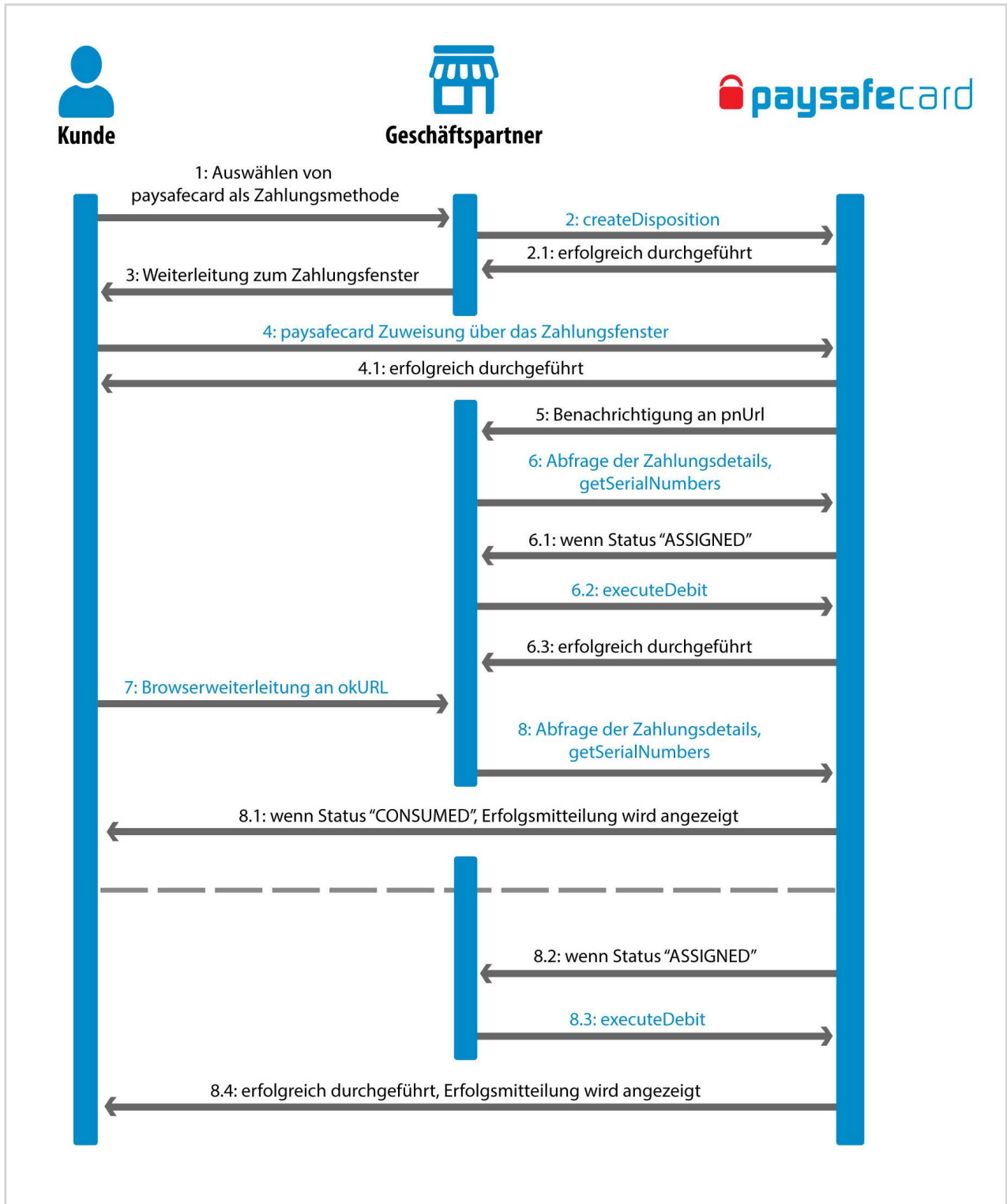
paysafecard stellt das „paysafecard Testsystem“ (SOATEST), eine Testumgebung für die Integration neuer Geschäftspartner zur Verfügung. Die Integration neuer Geschäftspartner erfolgt zunächst in diesem Testsystem, wofür keine IP Adressen auf eine Whitelist hinzugefügt werden. Im Merchant Service Center wird jeder neuer Onboarding Prozess mit einem automatisierten Integration Test begleitet, worin der Code im Backend der Webseite nach dem SOPG-Zahlungsprozess geprüft wird. Nach dem Bestehen des Integrationstest kann die Umstellung auf die Produktivsysteme erfolgen, die für den Geschäftspartner in folgenden Schritten abläuft:

- Umstellung auf Produktiv Login Daten (alle von paysafecard vergeben)
- Ersetzen der Service Endpoint URL
- Ersetzen der WSDL URL

Alle Daten werden vom Merchant Service Center zur Verfügung gestellt.

- Der Endpoint für die Testumgebung ist: <https://soatest.paysafecard.com>
- Der Endpoint für die Produktivumgebung ist: <https://soa.paysafecard.com>

## 5. Technische Übersicht



## 6. Funktionsdetails und WSDL-Struktur

In diesem Dokument werden alle für den grundlegenden Zahlungsprozess erforderlichen Funktionen der SOPG WSDL-Struktur beschrieben. Bitte beachten Sie, dass der Vertrag viele weitere Funktionen umfasst. Alle erforderlichen Zahlungsparameter müssen angegeben und eine Übermittlung vorgenommen werden, auch wenn der Wert NULL bleibt. Benötigt das Webservice Framework eine WDSL in Echtzeit, laden Sie die SOPG WSDL von der WSDL URL herunter und stellen Sie diese in Ihrer lokalen Umgebung zur Verfügung.

**HINWEIS: Ein Abruf der WSDL von den paysafecard Servern in Echtzeit ist nicht möglich.**

### 6.1 Funktionen createDisposition

Der Geschäftspartner initiiert den Zahlungsprozess durch Übermittlung eines „createDisposition“ Request an paysafecard, durch das eine Disposition auf dem Server angelegt wird. Der zulässige Maximalbetrag liegt bei 1.000,00 EUR (bzw. einem gleichwertigen Betrag in einer anderen Währung).

Request - Elemente	Response - Elemente
SOPG-Username	
SOPG-Password	
MerchantTransactionID	
MerchantclientID	
SubID	MID
Amount	ResultCode
Currency	ErrorCode
okUrl	SubID
nokUrl	
pnUrl	
ClientIp	
Dispositionrestrictions	
ShopId	
Shoplabel	

### 6.2 Funktion getSerialNumbers

Ruft den Disposition Status ab und prüft, ob dieser dem erwarteten Status entspricht, bevor die nächste Funktion aufgerufen wird.

Request - Elemente	Response - Elemente
	MTID
	SubID
SOPG-Username	ResultCode
SOPG-Password	ErrorCode
MerchantTransactionID	Amount
SubID	Currency
Currency	DispositionState
	SerialNumbers

### 6.3 Funktion executeDebit

Bucht das Geld von der paysafecard des Kunden ab. In diesem Schritt wird die Zahlung abgeschlossen, wenn die „close“ Flag auf „1“ gesetzt ist. Der Geschäftspartner kann die Transaktion bis zum Ende der Dispositionszeit offenhalten (der Betrag wird in voller Höhe reserviert). Möchte der Geschäftspartner die Disposition schließen, ohne die Zahlung tatsächlich durchzuführen muss die Funktion mit „amount=0.00“ aufgerufen werden.

Request - Elemente	Response - Elemente
SOPG-Username	MTID
SOPG-Password	SubID
MerchantTransactionID	ResultCode
SubID	ErrorCode
Currency	
Close	

### 6.4 Zahlungsbenachrichtigung (payment notification)

Die Zahlungsbenachrichtigung wird verwendet, um den Geschäftspartner nach der PIN-Zuweisung im Zahlungsfenster, unabhängig vom Kundenverhalten, zu informieren. Dieser Service stellt sicher, dass Dispositionen vor dem Aufladen Kundenkontos abgeschlossen werden können und ist deshalb ausgesprochen empfehlenswert, um das Ablaufen von Transaktionen zu verhindern. Wenn ein Kunde im Zahlungsfenster einen PIN eingibt, sendet unsere API einen POST-Request an die, vom Geschäftspartner im createDisposition request definierten, pn url. Um die erfolgreiche Zustellung zu bestätigen erwartet unser Payment Notification Server HTTP-200 als Antwort. Für die Übertragung an eine gesicherten pn url (https) sind die gängigsten Zertifizierungsaussteller (eng. certificate authorities) in unserem Speicher der vertrauenswürdigen Zertifizierungsaussteller (eng. certificate root store) hinterlegt. Eine pn url mit eingetragenem Port (:40) funktioniert nicht.

Eine Zahlungsbenachrichtigung wird nur auch erfolgreicher paysafecard Zuweisung übermittelt. Diese Benachrichtigung wird zusätzlihc zur Weiterleitung zur „okUrl“ versandt. Im Falle technischer Anwendungsfehler auf Seiten des Geschäftspartners wird die Zahlungsbenachrichtigung erneut übermittelt.

Ausgabe-Parameter	Merchant Response-Elemente
Mtid	
eventType (ASSIGN_CARDS is returned)	
serialNumbers	HTTP 200
currency	
disposition amount	
cardTypeld	

#### 6.4.1 Wiederholung der Zahlungsbenachrichtigung

Im Falle technischer Fehler (z. B. Socket Timeout) oder Anwendungsfehler (z. B. HTTP Statuscode 500) wird die Zahlungsbenachrichtigung in regelmäßigen Abständen erneut übermittelt, bis eines der folgenden Kriterien erfüllt ist:

- Die Zahlungsbenachrichtigung wurde erfolgreich zugestellt (d. h. Zahlungsserver antwortet mit HTTP-Statuscode 200).
- Die maximale Anzahl von Wiederholungsversuchen wurde erreicht (aktuelle Konfiguration: 5 Wiederholungsversuche).



### Die Zahlungsbenachrichtigung wird im folgenden Intervall wiederholt:

- 1. Zustellversuch unmittelbar nach Pin Eingabe
- 2. Zustellversuch 1 Sekunde nach Pin Eingabe
- 3. Zustellversuch 60 Sekunden nach Pin Eingabe
- 4. Zustellversuch 2 Minuten nach Pin Eingabe
- 5. Zustellversuch 3 Minuten nach Pin Eingabe

Der Zeitraum zwischen PIN Zuweisung und PIN Entwertung ist nach den AGBs zwischen 1 und 10 Minuten konfiguriert. Je nach Konfiguration des Dispositionszeitfenster (Kapitel 8.3.1) können die letzten drei Zustellversuche nicht ausgeführt werden, weil die Transaktion schon abgelaufen ist.

## 7. Zahlungsablauf

### 1. Zahlung einleiten mit der createDisposition Funktion.

- 1.1 falls die Antwort richtig ist (errorcode & resultcode = 0) -> Leiten Sie den Kunden auf unser Zahlungsfenster weiter.
- 1.2 falls die Antwort falsch ist (errorcode & resultcode nicht 0) -> Zeigen Sie dem Kunden folgende Fehlermeldung: **“Die Transaktion konnte auf Grund von technischen Problemen nicht erstellt werden. Falls dieses Problem weiterhin besteht wenden Sie sich bitten an den (Businesspartner) Support.“**

### 2. Der Kunde ist auf das Zahlungsfenster weitergeleitet worden.

### 3. Der Kunde gibt auf dem Zahlungsfenster einen gültigen paysafecard PIN-Code ein und klickt auf Bezahlen.

- 3.1. Auf dem paysafecard Zahlungsfenster wird der PIN-Code auf Korrektheit validiert. Falls der Kunde am Zahlungsfenster die Transaktion abbricht, bitte folgende Fehlermeldung anzeigen: **„Zahlungsabbruch durch Endkunde“**.

### 4. Da die Karte der Transaktion zugewiesen ist (Zahlungsstatus S für „Disposed“), senden wir die Zahlungsbenachrichtigung an Ihre pnURL

### 5. Nach dem Erhalten der Zahlungsbenachrichtigung antworten Sie mit http Code 200 und prüfen Sie den Status der Transaktion in dem Sie die Funktion getSerialNumbers anfragen.

- 5.1 Falls der Dispositions Status Wert vom der getSerialNumbers Funktion Status O zurückliefert, wurde die Transaktion bereits abgeschlossen.
- 5.2. Falls der Dispositions Status Wert vom der getSerialNumbers Funktion Status S zurückliefert, soll die executeDebit Funktion ausgeführt werden.
  - 5.2.1. Falls die Antwort richtig ist (errorcode & resultcode = 0) -> Bitte das Endkundenkonto aufstocken.
  - 5.2.2. Falls die Antwort falsch ist (errorcode & resultcode nicht 0) -> Keine Aktion erforderlich.

### 6. Der Endkunde wird auf die okUrl weitergeleitet.

### 7. Sie prüfen den Status der Transaktion in dem Sie die Funktion getSerialNumbers anfragen.

- 7.1 Falls die Dispositions Status Werte von der getSerialNumbers Funktion Status “R” oder “X” zurückliefern, zeigen Sie dem Endkunden folgende Fehlermeldung: **“Die Zahlung konnte auf Grund eines temporären Verbindungsproblems nicht abgeschlossen werden. Falls dieses Problem weiterhin besteht wenden Sie sich bitte an den Support.“**
- 7.2. Falls der Dispositions Status Wert vom der getSerialNumbers Funktion Status O zurückliefert, wurde die

Transaktion bereits abgeschlossen. Bitte zeigen Sie dem Endkunden eine Erfolgsmeldung.

7.3 Falls der Dispositions Status Wert von der getSerialNumbers Funktion Status S zurückliefert, soll die executeDebit Funktion ausgeführt werden.

7.3.1 Falls die Antwort richtig ist (errorcode & resultcode = 0) -> Bitte das Endkundenkonto aufstocken und dem Endkunden eine Erfolgsmeldung anzeigen.

7.3.2 Falls die Antwort falsch ist (errorcode & resultcode nicht 0) -> Bitte dem Endkunden eine Fehlermeldung zeigen. Die Fehlermeldung lautet: „**Die Zahlung konnte auf Grund eines temporären Verbindungsproblems nicht abgeschlossen werden. Falls dieses Problem weiterhin besteht wenden Sie sich bitte an den Support.**“

## 8. Parameter

### 8.1. Parameter-Beschreibung

**username** – individueller Konto-Benutzername

- von paysafecard zur Authentifizierung zur Verfügung gestellt

**password** – individuelles Konto-Passwort

- von paysafecard zur Authentifizierung bereitgestellt

**mtid** – (eindeutige) Transaktions-ID, eindeutiger Identifikator für jede Disposition.

- max. Länge: 60 Zeichen
- empfohlen: bis zu 20 Zeichen
- durch Geschäftspartner bereitgestellt
- Nur folgende Werte zulässig: A-Z, a-z, 0-9 sowie - (Bindestrich) und \_ (Unterstrich)
- Beispiel: 3516-6s4dfsad41

**subld** – zwingend erforderlicher Parameter, sofern nichts anderes vereinbart wurde, ist dieser Wert leer zu lassen.

- Sogenannte „reporting criteria“ (Reportingkriterien) bieten die Möglichkeit, Transaktionen zu klassifizieren
- max. Länge: 8 Zeichen (Groß-/Kleinschreibung)
- Vereinbarung mit paysafecard erforderlich
- Beispiel: shop1

**amount** – Dispositionsbetrag

- Angefragter Betrag darf 1.000,00 EUR (bzw. einen gleichwertigen Betrag in einer anderen Transaktionswährung) nicht überschreiten
- max. 11 Zeichen vor - exakt 2 Zeichen nach dem Dezimalpunkt
- Verwenden Sie einen Punkt als Dezimaltrennzeichen
- Beispiel: 100.00

**currency** – Dispositionswährung

- max. Länge: 3 Zeichen, alles Großbuchstaben
- ISO-Währungscode
- Beispiel: EUR

**pnUrl** – Zahlungsbenachrichtigungs-URL, über die paysafecard den Geschäftspartner informiert, sobald eine

- paysafecard erfolgreich einer Disposition zugewiesen wurde (weitere Details in Kapitel 6.2).
- URL muss absolut und URL-kodiert sein, weil sie als Parameter übermittelt wird
- URL muss vom Geschäftspartner definiert werden
- max. Länge: 765 Zeichen

**okUrl** – die URL, zu der Kunden von paysafecard weitergeleitet werden, sobald sie ihre paysafecard PINs

- erfolgreich zugewiesen haben. Der Geschäftspartner kann Informationen in die URL einbinden.
- URL muss absolut und URL-kodiert sein, weil sie als Parameter übermittelt wird
- max. Länge: 765 Zeichen

**nokUrl** – Die URL, zu der Kunden von paysafecard weitergeleitet werden, wenn Sie im paysafecard

- Zahlungsfenster auf „Abbrechen“ klicken.
- URL muss absolut und URL-kodiert sein, weil sie als Parameter übermittelt wird
- max. Länge: 765 Zeichen

Die „okUrl“, „nokUrl“ und „pnUrl“ sind unbedingt URL-kodiert (man spricht auch von prozentkodiert) zu übermitteln. Geschieht dies nicht, kann es zu einer falschen Weiterleitung des Kunden zur Bestätigungsseite sowie möglicherweise zu einem Fehlschlagen der Zahlung kommen.

**merchantClientID (MCID)** – Auch bekannt als „MCID“. Die merchantClientID ist ein wichtiger Parameter für die Integration von paysafecard. Die merchantClientID identifiziert den Kunden auf der Seite unserer Business Partner. Die optimale merchantClientID ist ein komplett zufälliger Wert. Dieser Wert identifiziert den Kunden eindeutig und einzigartig, und beinhaltet keine persönliche Information.

Diese merchantClientID sollte für alle Transaktionen eines Kunden gleich bleiben.

#### Richtlinien für mögliche Merchant Client IDs:

##### Gültige Werte:

###### Wert

- 2c3be0b50c7a5f1964a63d78f38a6ffc41c027e9
- 742f2b1a55cd5d606ea44b4fcb54646a
- 3a5b0d0777dead9df93d502df85c8180e53804eb
- 3192481752123
- CustomerID1

###### Typ

- SHA1 - test@123.com
- MD5 - test@123.com
- SHA1 - UsernameValue1
- Zufälliger Customer Identifier (Customer ID)
- Customer Identifier ohne persönliche Information

##### Ungültige Werte:

- test@123.com
- Username\_1
- Vorname123
- Nachname123
- Zeitmarke
- IP Adresse

**Bitte beachten:** ungültige merchantClientIDs werden nicht akzeptiert.

Wenn Sie paysafecard Transaktionen auf mehreren Brands verarbeiten wollen, erkundigen Sie sich bitte bei uns über die Möglichkeiten, mehrere Instanzen auf Ihrem Konto zu trennen.

**clientIp** – Die IP-Adresse des paysafecard Kunden.

**shopId** – Kennung des Shops, von dem der Request ausgeht. Wird meist von Payment Service Providern verwendet, die auch als Proxy für andere Zahlungsmethoden agieren.

- max. Länge: 60 Zeichen
- Empfohlen: bis zu 20 Zeichen
- durch Geschäftspartner bereitgestellt
- Zulässig sind nur: A-Z, a-z, 0-9 sowie - (Bindestrich) und \_ (Unterstrich)
- Beispiel: 2568-B415rh\_785

**shopLabel** – Label oder URL des Webshops, von dem der Request ausgeht, steht in Zusammenhang mit der „shopId“. Wird am wahrscheinlichsten von Payment Service Providern verwendet, die auch als Proxy für andere Zahlungsmethoden agieren.

- Max. Länge: 60 Zeichen
- Beispiel: www.foodstore.com

**mid** – Merchant ID, eindeutige ID des Geschäftspartners/Währungs-Kombination.

- 10 Zeichen lang
- von paysafecard zur Verfügung gestellt
- Beispiel: 1000001234

**dispositionState** – aktueller Status der Disposition (weitere Details in Kapitel 8.3).

**serialNumbers** – Seriennummer(n) vom Kunden zugewiesener paysafecard PINs, nach Eingabe der PINs im paysafecard Zahlungsfenster (Werte durch Semikolon getrennt).

- currency: ISO-Währungscode
- disposition amount: zu dieser Disposition reservierter Betrag auf der paysafecard des Kunden
- cardTypeld: paysafecards werden in verschiedene Kartentypen eingeteilt, z. B. junior\_paysafecard;
- adult\_paysafecard; inhouse\_paysafecard
- Beispiele:  
 0000000001200000;EUR;7.50;00002;  
 0000000001300000;EUR;5.50;00002;

**close** – Die Close Flag einer Disposition kann auf „0“ oder „1“ gesetzt werden und zeigt an, ob weitere Aktivitäten durchgeführt werden oder nicht.

- „0“ [Transaktion nicht schließen]
- „1“ [Transaktion schließen, dies ist die letzte Abbuchung]

**resultCode** – Ergebniscode der Funktion (Details im Kapitel Ergebniscode).

**errorCode** – Fehlercode der Funktion (Details im Kapitel Fehlercodes).

**dispositionRestrictions** – Dispositionseinschränkungen können von Geschäftspartnern definiert werden, um Zahlungstransaktionen im Rahmen ihrer individuellen Anforderungen zu begrenzen. Details in Kapitel 8.2.

- Mehrere Wiederholungen möglich
- Jede Einschränkung besteht aus einem „key“ und einem „value“:
- „key“ - der Schlüssel der Einschränkung
- „value“ - der Wert der Einschränkung

## 8.2. Einschränkungen

Bitte übermitteln Sie mit dem API-Request „create Disposition“ einen Ländercode (ISO 3166-1), um die Zahlung auf das gewünschte Land zu beschränken. Beim Integrationstest im Merchant Service Center bitte nicht eine Länderbeschränkung im „create Disposition“ API Request (COUNTRY) definieren. Im Integrationstest wird ausschließlich ein Kartentyp verwendet (AT\_Classic).

Schlüssel	Beispielwert	mögliche Werte	Beschreibung
COUNTRY	DE	alle Länder, in denen Paysafecard erhältlich ist (Beispiel: FR, ES, ...)	Beschränkt die Durchführung einer Zahlung exklusiv auf Deutschland. Als Wert können Ländercodes gemäß ISO 3166-1 angegeben werden.

Folgende Beschränkungen sind für paysafecard Zahlungen über ein paysafecard Konto (my paysafecard) verfügbar:

Schlüssel	Beispielwert	mögliche Werte	Beschreibung
MIN_AGE	18	muss einen positiven Zahlenwert enthalten	Beschränkung auf my paysafecard Kontoinhaber einem Mindestalter von 18 Jahren.
MIN_KYC_LEVEL	FULL	SIMPLE oder FULL	Beschränkung auf my paysafecard Kontoinhaber mindestens angegebenem Status, hier FULL.

### 8.3. Disposition Status

One letter code	Meaning	Description
R	Created	Die Disposition wurde erfolgreich angelegt. Geschieht in den kommenden 30 Minuten nichts, wird der Disposition Status durch paysafecard auf „X“ gesetzt.
S	Disposed	Die paysafecard des Kunden wurde der Disposition erfolgreich zugewiesen. Der Geschäftspartner kann „executeDebit“ ausführen; es wurde noch keine Abbuchungen vorgenommen.
O	Consumed	Die finale Abbuchung mit close=1 wurde ausgeführt; keine weiteren Abbuchungen möglich.
L	Cancelled	Die Disposition wurde vom Kunden aktiv abgebrochen.
X	Expired	Das Zeitfenster für die Disposition ist abgelaufen (entweder bevor eine paysafecard zugewiesen oder bevor „executeDebit“ aufgerufen wurde).

#### 8.3.1 Dispositionszeitfenster (disposition time frame)

Sobald eine Disposition in Status „S“ („DISPOSED“) vorliegt, müssen die Geschäftspartner je nach Einstellung Ihre Abbuchungen vornehmen. Grundlegend ist das Dispositionszeitfenster 60 Sekunden. Nach Absprache mit dem Integrationsteam von paysafecard kann man laut AGBs diese Einstellung auf bis zu 10 Minuten ändern. Mit Ende der Dispositionszeit läuft die Disposition automatisch ab, der auf der paysafecard des Kunden reservierte Betrag wird wieder verfügbar. Außerdem werden alle angelegten, jedoch nicht erfolgreich abgebuchten Dispositionen auf „EXPIRED“ gesetzt.

**HINWEIS: Diese Jobs sind nur auf dem paysafecard Produktivserver aktiv. Auf dem Testsystem sind Transaktionen im Status „S“ („DISPOSED“) und der reservierte Betrag so lange gebunden, bis die Transaktion mit executeDebit abgeschlossen wurde.**

### 8.4. Transaktionszeitfenster (payment expiration window)

Jede initiierte Transaktion bleibt für 30 Minuten im Status „R“ („CREATED“). Wenn die Transaktion keine Eingabe eines validen paysafecard PIN erfährt, wird die Transaktion auf Status „X“ („EXPIRED“) gesetzt und ist daher abgelaufen.

### 8.5. Details Zahlungsfenster

#### 8.5.1 Zahlungsfenster Desktop

Das paysafecard Zahlungsfenster kann in einem Popup-Fenster oder alternativ in einem iFrame angezeigt werden.

Um sicherzustellen, dass das gesamte Zahlungsfenster für den Benutzer sichtbar ist, bieten Sie bitte stets die Möglichkeit zum vertikalen Scrollen oder für dynamisches Skalieren.

- Die Breite ist auf **600 px** fixiert
- Die Höhe ist auf **840 px** fixiert

#### 8.5.2 Zahlungsfenster Mobilgeräte

Das paysafecard Zahlungsfenster ist für Mobilgeräte optimiert. Verwendet ein Kunde ein Gerät mit einer Auflösung unter 600 px wird ein optimiertes Zahlungsfenster angeboten. Dasselbe gilt, wenn der eingebettete iFrame schmaler als 600 px ist.

**HINWEIS: Der iFrame zur Einbettung des Desktop-Zahlungsfenster muss mindestens 600 px breit sein. Anderenfalls wird die mobile Version des Zahlungsfensters angezeigt.**

## 8.6 Gebietsschema- und Spracheinstellungen

Zur Gewährleistung der Rückwärtskompatibilität geben alle Sprachparameter weiterhin dieselben Ergebnisse aus wie in Vorgängerversionen der API, alle Sprachen werden jedoch automatisch in Gebietsschemata umgewandelt.

**Sprache und Gebietsschema des Zahlungsfensters werden grundsätzlich durch folgende Regel bestimmt:**

1. Hat der Kunde das Zahlungsfenster schon einmal besucht? Abrufen des Gebietsschemas aus dem gesetzten Cookie.
2. Ableiten des Gebietsschemas aus der IP-Adresse des Kunden<sup>1</sup>.
3. Verwenden des Wertes aus dem Gebietsschema-Parameter.
4. Verwenden des Wertes aus dem Sprach-Parameter.
5. Abrufen des Gebietsschemas aus dem Browser-Header.
6. Verwenden des Fallback-Gebietsschemas (de\_de). Es ist deshalb nicht erforderlich, ein Gebietsschema-Parameter zu bestimmen.

## 9. Beispielzahlung

In diesem Kapitel wird ein Testszenario mit Beispieldaten vorgestellt. In der Praxis wird der Ablauf von Geschäftspartner zu Geschäftspartner variieren, je nachdem, ob eine oder mehrere Abbuchungen oder Statusabfragen durchgeführt werden. Verwenden Sie für Ihre Tests nicht die Daten aus diesem Beispiel! Jeder Geschäftspartner erhält zum Testen einen einheitlichen Testdatensatz.

### 9.1 createDisposition

Der Geschäftspartner initiiert den Zahlungsprozess durch Versenden eines „createDisposition“ Request.

**Beispiel-Request:**

```
<urn:createDisposition>
  <urn:username>USER</urn:username>
  <urn:password>PASSWORD</urn:password>
  <urn:18b02d230-a6822f-4cbb-ae9-0bc07d90cfa4</urn:
    <!--Zero or more repetitions-->
    <urn:subId></urn:subId>
    <urn:amount>10.00</urn:amount>
    <urn:currency>EUR</urn:currency>
    <urn:okUrl>http%3a%2f%2fwww%2epaysafecardokURL%2ecom</urn:okUrl>
    <urn:nokUrl>http%3a%2f%2fwww%2epaysafecardnokURL%2ecom</urn:nokUrl>
    <urn:merchantclientid>clD_919191</urn:merchantclientid>
    <urn:pnUrl> http%3a%2f%2fwww%2emerchantpnURL%2ecom </urn:pnUrl>
    <!--Zero or more repetitions-->
    <urn:dispositionRestrictions>
      <urn:value>FR</urn:value>
    <urn:dispositionRestrictions>
      <urn:key>MIN_AGE</urn:key>
      <urn:value>18</urn:value>
    </urn:dispositionRestrictions>
    <!--Optional-->
  </urn:createDisposition>
```

### Beispiel-Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns1:createDispositionResponse xmlns:ns1="urn:psscservice">
      <ns1:createDispositionReturn>
        <ns1:>18b02d230-a6822f-4cbb-ae9-0bc07d90cfa4</ns1:>
        <ns1:mid>1000001234</ns1:mid>
        <ns1:resultCode>0</ns1:resultCode>
        <ns1:errorCode>0</ns1:errorCode>
      </ns1:createDispositionReturn>
    </ns1:createDispositionResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

## 9.2 getCustomerPanel

„createDisposition“ wurde erfolgreich ausgeführt. Der Kunde kann nun zum paysafecard Zahlungsfenster weitergeleitet werden, um der Disposition Karten zuzuweisen.

### Beispiel-URL Testsystem:

```
https:// customer.test.at.paysafecard.com/pssccustomer/GetCustomerPanelServlet
```

### Beispiel-URL Produktivsystem:

```
https:// customer.cc.at.paysafecard.com/pssccustomer/GetCustomerPanelServlet
?mid=1000001234
&=18b02d230-a6822f-4cbb-ae9-0bc07d90cfa4
&amount=10.00
&currency=EUR
```

### Beispiel Eingabeparameter:

```
PIN:      0000 0000 1234 5678
Terms of Use: <checkbox, default unchecked>
```

## 9.3 pnUrl request

Das paysafecard System übermittelt ein „HTTP POST“ Request an das System des Geschäftspartners (pnUrl), um dieses über die erfolgreiche Zuweisung der paysafecard PINs des Kunden zu informieren.

### Beispiel-URL:

```
http://www.merchantpnURL.com/notifyME
?=3516-6s4dfsad41
&eventType=ASSIGN_CARDS
&serialNumbers=0000000001200000;EUR;100.00;DE00002
```

### Response:

```
HTTP 200
```

### 9.3.1 In Zahlungsbenachrichtigungen unterstützte Ländercodes

Als zusätzlicher Informationsparameter bildet der Ländercode (des Landes, in dem paysafecard verkauft wird) einen Bestandteil des Standard API-Requests der Zahlungsbenachrichtigung. Der Parameter „cardTypeld“ in der Zahlungsbenachrichtigung enthält eine Kombination aus dem Standard-ISOLändercode und der ID des Kartentyps. Ausnahme: Manche Karten sind keinem bestimmten Land zugeordnet. Deshalb wird hier kein Ländercode übermittelt.

#### Grundlegender pnUrl Response

```
mtid=<Mtid>
&eventType=<eventType>
&serialNumbers=<serialNr1>;<currency1>;<amount1>;<cardTyp1>;
<serialNr2>;<currency2>;<amount2>;<cardType2>;
Beispiel pnUrl
```

#### Beispiel - pnUrl Response

```
mtid=123456
&eventType=ASSIGN_CARDS
&serialNumbers=0000000001200000; EUR; 50.00; XX00004;
0000000001200001; EUR; 50.00; XX00004
```

### 9.4 getSerialNumbers

Der Geschäftspartner prüft den Transaktionsstatus mit dem „HTTP-GET“ - Request.

#### Beispiel-URL:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:psscservice">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:getSerialNumbers>
      <urn:username>USER</urn:username>
      <urn:password>PASSWORD</urn:password>
      <urn:mtid>transactionID123456</urn:mtid>
      <!--Zero or more repetitions-->
      <urn:subId></urn:subId>
      <urn:currency>EUR</urn:currency>
    </urn:getSerialNumbers>
  </soapenv:Body>
</soapenv:Envelope>
```



### Response:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns1="urn:psscservice">
  <soap:Body>
    <ns1:getSerialNumbersResponse>
      <ns1:getSerialNumbersReturn>
        <ns1:mtid>transactionID123456</ns1:mtid>
        <ns1:subld/>
        <ns1:resultCode>0</ns1:resultCode>
        <ns1:errorCode>0</ns1:errorCode>
        <ns1:amount>1.0</ns1:amount>
        <ns1:currency>EUR</ns1:currency>
        <ns1:dispositionState>R</ns1:dispositionState>
        <ns1:serialNumbers/>
      </ns1:getSerialNumbersReturn>
    </ns1:getSerialNumbersResponse>
  </soap:Body>
</soap:Envelope>
```

## 9.5 executeDebit

Nachdem der Kunde der Disposition erfolgreich Karten zugewiesen hat, führt der Geschäftspartner die Abbuchung durch und belastet die paysafecard des Kunden mit dem jeweiligen Betrag.

### Beispiel-Request:

```
<urn:executeDebit>
  <urn:username>USER</urn:username>
  <urn:password>PASSWORD</urn:password>
  <urn:18b02d230-a6822f-4cbb-ae9-0bc07d90cfa4</urn:
    <urn:subld/></urn:subld>
  <urn:amount>10.00</urn:amount>
  <urn:currency>EUR</urn:currency>
  <urn:close>1</urn:close>
</urn:executeDebit>
```

### Beispiel-Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns1:executeDebitResponse xmlns:ns1="urn:psscservice">
      <ns1:executeDebitReturn>
        <ns1:18b02d230-a6822f-4cbb-ae9-0bc07d90cfa4</ns1:
          <ns1:subld/>
          <ns1:resultCode>0</ns1:resultCode>
          <ns1:errorCode>0</ns1:errorCode>
        </ns1:executeDebitReturn>
      </ns1:executeDebitResponse>
    </soapenv:Body>
  </soapenv:Envelope>
```

## 10. Fehlercodes

Können Sie einen Fehlercode nicht in der Liste finden, wenden Sie sich bitte an [integration@paysafecard.com](mailto:integration@paysafecard.com)

### Beschreibung von resultcodes (Ergebniscodes):

Name Ergebnis	Beschreibung
resultCode	0 : erfolgreich
	1 : logisches Problem
	2 : technisches Problem
errorCode	Enthält eine Fehlernummer, wenn der resultCode nicht gleich 0 ist.

### Die gängigsten Fehlercodes sind:

2001=Transaktion (%1/%2) bereits vorhanden. Bitte wenden Sie sich an Ihren Webshop.

2017=Transaktion (%1/%2) in ungültigem Status %3; erwartet wurde %4 oder %5.

3001=Merchant %1 ist nicht aktiv. Bitte wenden Sie sich an Ihren Webshop.

3007=Abbuchungsversuch nach Ablauf des Dispositionszeitfensters.

3014=Reporting Criterion %1 für Merchant %2 existiert nicht.

10007= Allgemeiner technischer Fehler.

10008= Authentifizierung fehlgeschlagen.

10015= Währung für SOPG-User nicht gültig.

10028= Ein Anfrageparameter konnte nicht validiert werden.

3017= Es ist verpflichtend, eine Merchant Client ID mitzuschicken.

3019= Die MCID enthält einen ungültigen Wert.