

# ZooZ Extended Server API

Version: 2.0.2

*This document may be used in its complete form only, and is solely for the use of Tactus Mobile employees and authorized personnel. The material herein is proprietary to Tactus Mobile and any unauthorized reproduction of any part thereof is strictly prohibited. In consideration of the receipt of this document and the associated literature, the recipient agrees not to disclose or otherwise use or make this document available to individuals outside the group directly responsible for the evaluation of its contents, or any third party.*

## Table of Contents

<b>Versions .....</b>	<b>4</b>
<b>Extended Server API .....</b>	<b>6</b>
About .....	6
Version .....	7
Get Started with ZooZ .....	7
Http Addresses .....	8
Http Request .....	8
Http Response .....	8
• All responses' content type set as: "application/json" .....	8
• All responses return with the following structure: .....	8
Security .....	9
Security Header Parameters .....	9
<b>API Commands .....</b>	<b>10</b>
1. Get Transaction Details with the Transaction ID .....	10
Notice .....	10
Parameters .....	10
Response .....	10
2. Get Transaction Details By Payer Email Address .....	18
Notice .....	18
Parameters .....	18
Response .....	18
3. Get Transaction Details By Parameters .....	20
Notice .....	20
Parameters .....	20
Query specific parameters .....	20
Response .....	21
4. Commit Transaction .....	21
Notice .....	21
Parameters .....	21
Response .....	22
5. Refund Transaction .....	23
Notice .....	23
Parameters .....	23
Response .....	23
6. Cancel Transaction .....	24
Notice .....	24
Parameters .....	24
Response .....	24
7. Get Subscription Details .....	25
Notice .....	25
Parameters .....	25
Response .....	25
8. Cancel Subscription .....	30
Notice .....	30
Parameters .....	30
Response .....	30
9. Check Subscription .....	32
Notice .....	32
Parameters .....	32
Response .....	32
10. Get Transaction Details By User ID .....	33



---

<i>Response.....</i>	<i>33</i>
<b>Callback API .....</b>	<b>34</b>
<i>Version .....</i>	<i>34</i>
<i>ZooZ Callback.....</i>	<i>34</i>
<i>Callback Structure Example.....</i>	<i>42</i>
<i>Callback parser code example .....</i>	<i>43</i>
<b>Errors .....</b>	<b>47</b>

## Versions

Date	V.	Description
1 May 2012	0	First Release
15 Aug 2012	1.0.0	Naming Convention. Adjust ZooZ parameters' naming convention, input and output parameters.
15 Aug 2012	1.0.0	Add transactionFee parameter for response object.
15 Aug 2012	1.0.0	Add addresses parameters for response object.
24 Mar 2013	1.0.1	Changed address parameters to include billing and shipping addresses.
22 April 2013	1.0.2	1) Added ability to update payment invoice details in Commit command. 2) Added the ability to get transaction details by dates when using get transaction details by payee email address command.
20 May 2013	1.0.3	Added cancel transaction command
24 June 2013	1.0.3	Added card holder name field for credit/debit cards.
3 July 2013	1.0.4	1) Added get subscription details command. 2) Added cancel subscription command. 3) Added check subscription command.
	1.0.5	1) Added response parameters to Get Transaction Details with the Transaction ID API and Callback API. 2) Added Get Transaction Details By User ID API.
16 Sep 2013	1.0.6	Added shipping method parameters to Get Transaction Details API and Callback API
21 Oct 2013	1.0.7	New country code to address object
7 Nov 2013	1.0.8	Added new response parameter to getTransactionDetails and getSubscriptionDetails APIs . Parameter name: riskScore
25 Nov 2013	1.0.9	Update response content type. All responses now return "application/json" as content type.
11 Dec 2013	1.1.0	Adding error description to responses
1 Jan 2014	1.1.1	1) Adding get transaction by parameters 2) Adding response parameters to getTransactionDetails APIs Parameter names: shippingCarrierCode, shippingCarrierName
2 Feb 2014	1.1.2	1) Added more options to get transaction by parameters 2) For WEB3.0 and HTML5 checkout, you can also use the token returned on "openTrx" on GetTransactionDetails API. 3) For HTML5 checkout, you can also use the token returned on "openSubscription". On GetSubscriptionDetails API. 4) Updated the ZooZ servers IP addresses for security checks.
4 March 2014	1.1.3	1) Added the option to get transaction by customer login name to get transaction by parameters. 2) Add new SDK for Extended Server API in: 2.1) Java 2.2) C# 2.3) Python 2.4) PHP



Enabling Mobile Payments

www.ZooZ.com

18 March 2014	1.1.4	1) Added newTransactionStatus and newTransactionStatus Code to Get Transaction Details API and Callback API
8 April 2014	1.1.5	1) Limit the number of payments returned by API to 5000
8 June 2014	1.1.6	1) Change in Subscription Response to support indefinite subscription
10 June 2014	1.1.7	Change street to address1 + address2 in all API Commands
2 July 2014	1.1.8	1) Added the ability to partially commit multiple times an authorized payment. 2) Added the ability to partially refund multiple times a captured payment.
9 July 2014	2.0.0	A new version for the callback API
6 Oct 2014	2.0.1	Added ZooZUniqueid to all the get payment details operations
13 Nov 2014	2.0.2	Additional IP addresses to the Callback API section were added.

## Extended Server API

### About

The Zooz platform is a sophisticated payment solution that enables you to accept payments across online retail channels. Zooz's payment technology envelops and enhances your existing payment, e-commerce, and merchant acquirer infrastructures using smart routing algorithms to optimize your processing and conversion fees and providing you with deeper consumer insights from every acquirer and payment method you offer. Integrating the Zooz SDK is easy and seamless, and enables you to access and manage your payment methods, acquirers, and payment data in one centralized location.

ZooZ Extended server API allows you to extend your control and manageability of your payment transactions.

With the Extended Server API you can:

1. Defer payments - approve your users' purchases after they authorize ZooZ transactions
2. Get full details on transactions by transaction ID or payee email address
3. Refund transactions

Prior to starting the integration, you must open a merchant account with a payment processor – if you already have a merchant account and acquirer(s), we will connect our platform to your existing infrastructures.

This guide aims to provide you all of the information necessary for a seamless integration. However, if you have any questions, you are welcome to contact us at [support@zooz.com](mailto:support@zooz.com).

### Version

The version of this API is: 2.0.2

Notice:

Please make sure to pass this version number to the API.

### Get Started with ZooZ

1. **Register and create app at ZooZ**
2. **Integrate the ZooZ SDK in your App**
3. **Generate transactions using ZooZ SDK**
4. **Create Http call to one of the APIs described below for a specific transaction**

## Http Addresses

Use the following host addresses for the Extended Server APIs:

1. Production: <https://app.ZooZ.com/mobile/ExtendedServerAPI>
2. Sandbox: <https://sandbox.zooz.co/mobile/ExtendedServerAPI>  
Note: The Sandbox URL ends with “.co” and not “.com”

Extended Server API supports multiple http methods, both POST and Get, but we advise you to use POST.

Note: Character encoding must be UTF-8

## Http Request

Each Server API request includes:

- Security header parameters
- API specific parameters

## Http Response

- All responses' content type set as: "application/json"
- All responses return with the following structure:

Parameter Name	Type	Description
ResponseStatus	Integer	If code equals zero (0), the request succeeded. Otherwise, the request failed.
ResponseObject	JSON object	The response object for the request. In case of success, according to the API response object In case of failure according to Error Response Object described below

## Error Response Object

Parameter Name	Type	Description
ResponseErrorCode	Integer	Contains the specific code for the error description.
errorDescription	String	In case there is more information about the error, this field will contain the error description.



## Security

ZooZ does it's best to protect your details and your users' details. It is your responsibility to protect them as well when you retrieve information through the ZooZ Extended Server API. In order to afford you the highest level of security, we separated your credentials from the various platforms you are using (ZooZ's developer portal / Mobile web API / ZooZ's extended server API). In order to give you access to the ZooZ extended Server API we generated a new API Key for you to use.

You can find this API Key in My Account → Page header.

Please keep this API Key secure and encrypted in your server.

## Security Header Parameters

All of our APIs include a common set of required header parameters. Additional parameters are required depending on the API you use.

For each http call you will need to add the following header parameters:

Parameter Name	Mandatory	Description
ZooZDeveloperId	Yes	Developer email as entered in ZooZ Developers portal.
ZooZServerAPIKey	Yes	<p>ZooZ Server API Key assigned to you upon registration. The ZooZ server API Key is part of ZooZ's security mechanism to ensure strictly authorized access to ZooZ's secured server.</p> <p>Your ZooZ Server API Key can be found in the ZooZ portal, My Account, page header.</p> <p><u>Note:</u> You should store the Server API Key only on your secured server storage (if you've one), and not distribute the key with your apps.</p>

## API Commands

### 1. Get Transaction Details with the Transaction ID

Get transaction details with the transaction ID API. This allows you to retrieve complete transaction details and transaction statuses before and/or after commit or refund. This call does not change the transaction state.

#### Notice

Make sure to follow the [Http Addresses](#) & [Security instructions](#)

#### Parameters

Parameter Name	Mandatory	Length	Description
cmd	Yes	100	Command Name. cmd="getTransactionDetails"
ver	Yes	11	Extended Server API version number. This number affect on the naming convention and the response return by ZooZ service. Current version is: 2.0.2
transactionID	Yes	50	Transaction ID. Transaction ID is a unique identifier on ZooZ's servers as you get from the response whether it's from ZooZ's SDK response or callback response, after the user authorized or commit a transaction. For WEB3.0 and HTML5 checkout, you can also use the token returned on "openTrx".

#### Response

Response Objects will be returned in JSon format

Parameter Name	Mandatory	Length	Description
appName	Yes	50	Application name that created the transaction.
ZooZUniqueID	Yes	100	The App Unique ID, as registered in the Zooz Developer Portal.
transactionID	Yes	50	Transaction ID is unique identifier on ZooZ's servers as you get from the response whether it's from ZooZ's SDK response or callback response, after the user authorized or commit a transaction.

isSandbox	Yes	true/ false	Whether the response comes from the Sandbox Server or Production Server.
transactionStatus (Deprecated)	Yes		Transaction status should be one of the following: <ul style="list-style-type: none"> <li>• Pending – Transaction has not yet been committed. Merchant cannot commit this transaction.</li> <li>• TPCPending – Transaction has not yet been committed. It's up the merchant to commit this transaction.</li> <li>• AuthorizationProcess – A transaction is in authorization process.</li> <li>• Succeed – Transaction succeeded!</li> <li>• Failed – Transaction Failed.</li> </ul>
fundSourceType	No		Fund Source type indicates which payment method was selected to pay for this transaction. Those are the values: <ul style="list-style-type: none"> <li>• VISA</li> <li>• MasterCard</li> <li>• AmericanExpress</li> <li>• Discover</li> <li>• Diners</li> <li>• JCB</li> <li>• LeumiMasterCard</li> <li>• LeumiVisa</li> <li>• PayPal</li> <li>• Other</li> </ul>
lastFourDigits	No	4	Last 4 digits, If a credit card was selected to pay for this transaction
cardHolderName	No	100	Name on credit/debit card.
amount	Yes	9	The original transaction amount. Expected format: dddddd.cc (e.g. 105.15)
paidAmount	No	9	The actual paid amount. Expected format: dddddd.cc (e.g. 105.15)
currencyCode	Yes	3	The currency code used, for example: USD, EUR, GBP
transactionFee	No	9	ZooZ's fee. Expected format: dddddd.cc (e.g. 105.15)

transactionTimestamp			Yes	19	The creation date of the transaction. Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this date.
userId			Yes	50	A unique identifier on ZooZ’s servers for the user who processed this transaction.
mobileDeviceId			Yes	50	A unique identifier on ZooZ’s servers for the device on which this transaction was processed.
mobileDeviceName			Yes	50	The name of the device on which this transaction was processed.
ipAddress			Yes	50	The ip address of this transaction.
mobileDeviceOsVersion			Yes	30	The OS version of the device on which this transaction was processed.
shippingMethod			No	100	The shipping method that was selected by the user for this transaction.
shippingAmount			No		The price of the selected shipping method.
shippingCarrierCode			No	200	The carrier code that matches the shipping method selected by the user
shippingCarrierName			No	200	The carrier name that matches the shipping method selected by the user
user	firstName		No	50	Payer’s first name
	lastName		No	50	Payer’s last name
	email		No	50	Payer’s Email address
	phone	phoneNumber	No	10	Payer’s phone number
		countryCode	No	3	Payer’s phone number’s country code
	additionalDetails		No	400	Payer’s additional details
invoice	number		No	100	Invoice number as registered by the merchant
	additionalDetails		No	200	Invoice additional details as registered by the merchant
	items	id	No	100	Item’s id
		name	No	50	Item’s name
		price	No	9	Item’s price. Expected format: dddddd.cc (e.g. 105.15)
		quantity	No	5	Number of items Expected format: ddd.c (e.g. 105.1)
		taxAmount	No	9	Item’s tax amount. Expected format: dddddd.cc (e.g. 105.15)
		additionalDetails	No	200	Invoice’s item additional details as registered by the merchant

addresses	billing	firstName	No	50	Billing address associated first name as registered by the merchant/user
		lastName	No	50	Billing address associated last name as registered by the merchant/user
		address1	No	50	Billing address (line 1) as registered by the merchant/user
		address2	No	50	Billing address (line 2) as registered by the merchant/user
		address3	No	50	Billing address (line 3) as registered by the merchant/user
		zipCode	No	50	Billing address zip code as registered by the merchant/user
		city	No	50	Billing address city as registered by the merchant/user
		state	No	50	Billing address state as registered by the merchant/user
		country	No	50	Billing address country as registered by the merchant/user
		countryCode	No	50	Billing address country code as registered by the merchant/user (Country code in ISO 3166 format)
	shipping	firstName	No	50	Shipping address associated first name as registered by the merchant/user
		lastName	No	50	Shipping address associated last name as registered by the merchant/user
		address1	No	50	Shipping address (line 1) as registered by the merchant/user
		address2	No	50	Shipping address (line 2) as registered by the merchant/user
		address3	No	50	Shipping address (line 3) as registered by the merchant/user
		zipCode	No	50	Shipping address zip code as registered by the merchant/user
		city	No	50	Shipping address city as registered by the merchant/user
		state	No	50	Shipping address state as registered by the merchant/user
		country	No	50	Shipping address country as registered by the merchant/user
		countryCode	No	50	Shipping address country code as registered by the merchant/user (Country code in ISO 3166 format)

riskScore	Yes	5	<p>The riskScore represents the likelihood that a given transaction is fraudulent. Merchants use the riskScore to determine whether to accept, reject, manually review, or submit transactions to complementary services for further screening.</p> <p>The riskScore is given as a percentage, and as such it ranges from 0.01 to 100.00. For example, an order with a riskScore of 20.00 has a 20% chance of being fraudulent, while an order with a riskScore of 0.10 has a 0.1% chance of being fraudulent.</p>
newTransactionStatus	Yes	100	<p>newTransactionStatus should be one of the following:</p> <ul style="list-style-type: none"> <li> <p>“Payment Approved” code: 1001 Transaction completed successfully.</p> </li> <li> <p>“Payment Authorized, Pending completion” code: 1002 Transaction Authorized and waiting for the merchant to commit.</p> </li> <li> <p>"Payment Authorized, Pending user completion" code: 1003 Transaction Authorized, has not yet been committed. User still hasn't completed the transaction. (Merchant cannot commit this transaction)</p> </li> <li> <p>“Refunded” code: 1004 The original transaction that was refunded.</p> </li> <li> <p>“Refund due chargeback” code: 1005 Refund transaction due to chargeback received.</p> </li> </ul>

			<ul style="list-style-type: none"> <li>• "Charged Back" code: 1006 The original transaction that was charged back.</li> <li>• "Refund" code: 1007 Refund transaction</li> <li>• "Pending user payment completion" code: 1008 Transaction is waiting for user action. (Merchant cannot commit this transaction)</li> <li>• "Voided" code: 1009 Transaction has been voided by the merchant.</li> <li>• "Payment Declined" code: 1010 Transaction was declined by the processor.</li> <li>• "Refund Declined" code: 1011 Refund transaction was declined</li> </ul>
newTransactionStatusCode	Yes		Refer to newTransactionStatus
processorReferenceId	No		Reference id for the request, received from the processor.
authorizationNumber	No		An authorization code is a numerical or alphanumeric code sent directly by a credit card issuer or issuing bank that is uniquely ascribed to a sales transaction that indicates approval or denial of the transaction.
taxAmount	Yes		The tax amount of the payment. Please note: amount = subtotal + taxAmount. This value is used to calculate

			the subtotal for the invoice.
customerLoginID	No		customerLoginId is the customer's identifier on the merchant's server, allowing one click checkout from all devices.
paymentType	Yes		Can be one of the following: <ul style="list-style-type: none"> <li>• Payment</li> <li>• Refund</li> <li>• Chargeback</li> </ul>
avsCode			Authorization avsCode response
fraudDetectionResponse	No		Fraud detection response as received from the processor.
country	No		The country from which the user processed the transaction, according to the ipAddress.
captures (JSON array)	amount	No	The captured amount.
	referenceId		Reference id for the request, received from the processor.
	timestamp		The capture date. Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this date.
refunds (JSON array)	amount	No	The refunded amount.
	referenceId		Reference id for the request, received from the processor.
	timestamp		The refund date. Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this date.



### Example Response:

```
{
  "responseObject": {
    "currencyCode": "USD",
    "fundSourceType": "MasterCard",
    "invoice": {
      "items": [
        {
          "id": "100101",
          "price": 12.5,
          "name": "Coca Cola",
          "additionalDetails": "Can",
          "quantity": 2
        },
        {
          "id": "100102",
          "price": 10,
          "name": "Sprite",
          "additionalDetails": "Can",
          "quantity": 2
        }
      ],
      "additionalDetails": "Invoice additional details",
      "number": "Invoice Number"
    },
    "transactionID": "7JU3VMGBXC4XJE343ASWF3W6C4",
    "appName": "Kartis",
    "ZooZUniqueID": "zooz.unique.id",
    "amount": 45,
    "isSandbox": "false",
    "lastFourDigits": "4444",
    "newTransactionStatus": "Payment Approved",
    "newTransactionStatusCode": 1001,
    "paidAmount": 45,
    "transactionFee": 0.17,
    "transactionTimestamp": 1344953560718,
    "mobileDeviceId": 287153,
    "mobileDeviceName": "GT-N7000",
    "mobileDeviceOsVersion": "10",
    "userId": 287239,
    "riskScore": 1.5,
    "ipAddress": "127.255.255.255",
    "addresses": [
      {
        "billing.street": "Main St. 1",
        "billing.city": "New York",
        "billing.state": "New York",
        "billing.zip": "643321",
        "billing.country": "USA"
      }
    ],
    "transactionStatus": "Succeed",
    "user": {
      "lastName": "Jones",
      "phone": {
        "phoneNumber": "05481234567",
        "countryCode": "001"
      },
      "email": "support@zooz.com",
      "additionalDetails": "Additional User Details 1234",
      "firstName": "Tom"
    }
  },
  "ResponseStatus": 0
}
```

## 2. Get Transaction Details By Payer Email Address

Getting transactions details by payer's email address API allows you to retrieve transaction full details and transaction status before or/and after commit or refund for all transactions associated with the above email. This call does not change the transaction state.

### Notice

Make sure to follow the [Http Addresses](#) & [Security instructions](#)

### Parameters

Parameter Name	Mandatory	Length	Description
cmd	Yes	100	Command Name. cmd="getTransactionDetailsByPayerEmail"
ver	Yes	11	Extended Server API version number. This number affect on the naming convention and the response return by ZooZ service. Current version is: 2.0.2
email	Yes	100	Payer email address.
fromDate	No	10	Filter the results by dates. Expected format: yyyy-mm-dd
toDate	No	10	

### Response

Response Objects will be returned in JSon format

Parameter Name	Mandatory	Length	Description
payments	Yes	N/A	A Json array containing 0 or more payment objects. A payment object is an object as described in "Get transaction details by transaction ID" – see above

Please note: There is a limit of 5000 payments on the number of payments returned.

## Example Response:

```

"responseObject": { "payments": [
  { "addresses": { "shipping": {
    "city": "New York", "country": "United States", "countryCode": "US", "firstName": "John", "lastName": "Doe",
    "state": "NY", "address1": "Main St. 1", "address2": "Apt. 1A", "zipCode": "643321"
  } },
  "amount": 4.98,
  "appName": "XXXXXX",
  "ZooZUniqueID": "zooz.unique.id",
  "authorizationNumber": "220020",
  "cardHolderName": "John Doe",
  "currencyCode": "USD",
  "fundSourceType": "MasterCard",
  "invoice": { "items": [ { "id": "100101", "price": 12.5, "name": "Soft drink", "additionalDetails": "Can", "quantity": 2 },
    { "id": "100102", "price": 10, "name": "Sprite", "additionalDetails": "Can", "quantity": 2 } ],
    "additionalDetails": "Invoice additional details", "number": "Invoice Number",
  "ipAddress": "XXX.XXX.X.XXX",
  "isSandbox": false,
  "lastFourDigits": "4444",
  "mobileDeviceId": 287153,
  "mobileDeviceName": "GT-N7000",
  "mobileDeviceOsVersion": "10",
  "newTransactionStatus": "Payment Approved",
  "newTransactionStatusCode": 1001,
  "paidAmount": 4.98,
  "riskScore": 0,
  "shippingAmount": 3.99,
  "shippingCarrierCode": "Fixed",
  "shippingCarrierName": "Fixed",
  "shippingMethod": "FedEx ground",
  "transactionFee": 0.278,
  "transactionID": "7JU3VMGBXC4XJE343ASWF3W6C4",
  "transactionTimestamp": 1394982854000,
  "user": { "lastName": "Doe", "phone": { "phoneNumber": "06481234567", "countryCode": "001" },
    "email": "support@zooz.com", "additionalDetails": "Additional User Details 1234", "firstName": "John" },
  "userId": "aaoWlJy2k+gJdLnwhgp9cw=="
  },
  { "amount": 1,
    "appName": "XXXXXX",
    "authorizationNumber": "ET156982",
    "cardHolderName": "John Doe",
    "currencyCode": "EUR",
    "fundSourceType": "MasterCard",
    "invoice": { "number": "" },
    "ipAddress": "XXX.XXX.X.XXX",
    "isSandbox": false,
    "lastFourDigits": "1118",
    "mobileDeviceId": 293859,
    "mobileDeviceName": "Macintosh",
    "mobileDeviceOsVersion": "10_8_4",
    "newTransactionStatus": "Payment Approved",
    "newTransactionStatusCode": 1001,
    "paidAmount": 1,
    "riskScore": 0,
    "transactionFee": 0.03,
    "transactionID": "VCE4YRAU2ZDKCNYKRMIMUUIG4I",
    "transactionTimestamp": 1394999695000,
    "user": {
      "email": "John@zooz.com",
      "phone": {}
    },
    "userId": "aaoWlJy2k+gJdLnwhgp9cw=="
  }
  ],
  "ResponseStatus": 0
}

```

### 3. Get Transaction Details By Parameters

Getting transactions details by parameters API allows you to retrieve transaction full details and transaction status before or/and after commit or refund for all transactions corresponding to the selected parameters. This call does not change the transaction state.

#### Notice

Make sure to follow the [Http Addresses](#) & [Security instructions](#)

#### Parameters

Parameter Name	Mandatory	Length	Description
cmd	Yes	100	Command Name. cmd="getTransactionDetailsByParameters"
ver	Yes	11	Extended Server API version number. This number affect on the naming convention and the response return by ZooZ service. Current version is: 2.0.2
queryType	Yes	100	The name of the wanted query (as listed in the table below).

#### Query specific parameters

queryType	Parameter Name	Parameter Type	Description
invoiceNumber	invoiceNumber	String	The invoice number of the requested transaction.
closedBetweenDates	startDate	Date in format: yyyy-MM-dd	The earliest date for the requested transactions.
	endDate	Date in format: yyyy-MM-dd	The latest date for the requested transactions.
statusAndDates	paymentStatus	String	Can be one of the following: Pending, TPCPending, AuthorizationProcess, Succeed, Failed, UserRejected, POSCanceled, Timeout
	startDate	Date in format: yyyy-MM-dd	The earliest date for the requested transactions.
	endDate	Date in format: yyyy-MM-dd	The latest date for the requested transactions.
customerLoginID	customerLoginID	String	The customer login ID on the merchant's app.
customerLoginName	customerLoginName	String	The customer login name on the merchant's app.

## Response

Same as in “Get Transaction Details By Payer Email Address” – see above

### 4. Commit Transaction

Commit Transaction API – If your app is set to the deferred payments setting (App setting in the ZooZ portal) then the flow of payments is separated to the following two phases:

1. User authorizes the transaction with ZooZ (Over Mobile Device)
2. Your server approves the purchase

This flexibility allows merchants to review the user order before making the actual charge. Using ZooZ developer’s portal or using this command API, you can choose to commit or ignore that transaction after the user authorizes it.

You can call this API up to 14 days after the transaction was authorized.

In case you want to do partial commit, send the amount parameter.

#### Partial commits allowed

Instead of capturing an entire payment amount, merchants can capture partial payment amounts, as they have the ability to fulfill products in an order. Up to 5 transaction captures can be performed on a particular payment; however, the total amount captured cannot exceed the original payment amount authorized (according to the processor’s/ acquirer’s policy) .

## Notice

- Make sure to follow the [Http Addresses](#) & [Security instructions](#)
- When using deferred payments with PayPal, commit timeframe is up to 3 hours.
- 

## Parameters

Parameter Name	Mandatory	Length	Description
cmd	Yes	100	Command Name. cmd=“commitTransaction”
ver	Yes	11	Extended Server API version number. This number affect on the naming convention and the response return by ZooZ service. Current version is: 2.0.2
transactionID	Yes	50	Transaction ID. Transaction ID is unique identifier on ZooZ’s servers as you get from the response whether it’s from ZooZ’s SDK response or callback response, after the user authorized or commit a

			transaction.
amount	No	9	Optional value. If you wish to commit the transaction on a smaller amount than the original amount the transaction was authorized, or if you were approved by ZooZ to commit the transaction on higher amount than the original amount. Expected format: dddddd.cc (e.g. 105.15)

invoice (JSON Object)	number		No	100	If you wish to commit the transaction on a different invoice. Invoice number as registered by the merchant
	additionalDetails		No	200	Invoice additional details as registered by the merchant
	items (JSON Array of JSON Objects)	id	No	100	Item's id
		name	Yes	50	Item's name
		price	No	9	Item's price. Expected format: dddddd.cc (e.g. 105.15)
		quantity	No	5	Number of items Expected format: ddd.c (e.g. 105.1)
		additionalDetails	No	200	Invoice's item additional details as registered by the merchant
		taxAmount	No	6	Item's tax. Expected format: ddd.cc (e.g. 105.15)

## Response

Response Objects will be returned in JSon format.

Parameter Name	Mandatory	Length	Description
boolean	Yes	5	true/false

### Example Response:

```
{
  "responseObject": {"boolean": true},
  "ResponseStatus": 0
}
```

## 5. Refund Transaction

The Refund Transaction API allows you to refund a specific transaction.

You can call this API up to 60 days after the transaction was executed.

In case you want to do partial refund, send the amount parameter.

### Partial refunds allowed

Instead of refunding an entire payment, merchants can refund partial payment amounts, as different products from an order are returned. Up to 5 refund transactions can be performed on the original payment; however, the total amount refunded cannot exceed the original payment amount authorized.

### Notice

Make sure to follow the [Http Addresses](#) & [Security instructions](#)

### Parameters

Parameter Name	Mandatory	Length	Description
cmd	Yes	100	Command Name. cmd="refundTransaction"
ver	Yes	11	Extended Server API version number. This number affect on the naming convention and the response return by ZooZ service. Current version is: 2.0.2
transactionID	Yes	50	Transaction ID. Transaction ID is unique identifier on ZooZ's servers as you get from the response whether it's from ZooZ's SDK response or callback response, after the user authorized or commit a transaction.
amount	No	9	Optional value. If you wish to refund the transaction on a smaller amount than the original amount the transaction was authorized. Expected format: dddddd.cc (e.g. 105.15)

### Response

Response Objects will be returned in JSon format.

Parameter Name	Mandatory	Description
boolean	Yes	true/false

#### Example Response:

```
{
  "responseObject": {"boolean": true},
  "ResponseStatus": 0
}
```

## 6. Cancel Transaction

The cancel transaction API is enabled only if your app is set to deferred payments. It allows you to cancel a specific pending transaction.

### Notice

Make sure to follow the [Http Addresses](#) & [Security instructions](#)

### Parameters

Parameter Name	Mandatory	Length	Description
cmd	Yes	100	Command Name. cmd="cancelPayment"
ver	Yes	11	Extended Server API version number. This number affect on the naming convention and the response return by ZooZ service. Current version is: 2.0.2
transactionID	Yes	50	Transaction ID. Transaction ID is unique identifier on ZooZ's servers as you get from the response whether it's from ZooZ's SDK response or callback response, after the user authorized or commit a transaction.

### Response

Response Objects will be returned in JSon format.

Parameter Name	Mandatory	Description
boolean	Yes	true/false

#### Example Response:

```
{
  "responseObject": {"boolean": true},
  "ResponseStatus": 0
}
```



## 7. Get Subscription Details

Get subscription details with the subscription ID API. This allows you to retrieve complete subscription details and its transactions details. This call does not change the subscription state.

### Notice

Make sure to follow the [Http Addresses](#) & [Security instructions](#)

### Parameters

Parameter Name	Mandatory	Length	Description
cmd	Yes	100	Command Name. cmd="getSubscriptionDetails"
ver	Yes	11	Extended Server API version number. This number affect on the naming convention and the response return by ZooZ service. Current version is: 2.0.2
subscriptionID	Yes	50	Subscription ID. Subscription ID is a unique identifier on ZooZ's servers as you get from the response whether it's from ZooZ's SDK response or callback response, after the user subscribes to your plan. For HTML5 checkout, you can also use the token returned on "openSubscription".

### Response

Response Objects will be returned in JSON format

Parameter Name	Mandatory	Length	Description
appName	Yes	50	Application name that created the subscription.
ZooZUniqueID	Yes	100	The App Unique ID, as registered in the Zooz Developer Portal.
subscriptionId	Yes	50	Subscription ID. Subscription ID is a unique identifier on ZooZ's servers as you get from the response whether it's from ZooZ's SDK response or callback response, after the user subscribes to your plan.
isSandbox	Yes	true/ false	Whether the response comes from the Sandbox Server or Production Server.
subscriptionStatus	Yes		Sunscription status should be one of the

			<p>following:</p> <ul style="list-style-type: none"> <li>• Pending – Subscription has not yet been validated.</li> <li>• Valid – Subscription is valid.</li> <li>• Expired – Subscription is no longer valid. There will be no more payments.</li> <li>• CanceledByUser – The subscription was canceled by the user. The subscription will remain valid until the end of the period that was already paid for.</li> <li>• CanceledByMerchant – The subscription was canceled by the merchant. Either because a payment was refunded or charged back. The subscription is not valid.</li> </ul>
fundSourceType	No		<p>Fund Source type indicates which payment method was selected to pay for this subscription. Those are the values:</p> <ul style="list-style-type: none"> <li>• VISA</li> <li>• MasterCard</li> <li>• AmericanExpress</li> <li>• Discover</li> <li>• Diners</li> <li>• JCB</li> <li>• LeumiMasterCard</li> <li>• LeumiVisa</li> <li>• PayPal</li> <li>• Other</li> </ul>
lastFourDigits	No	4	Last 4 digits, If a credit card was selected to pay for this subscription
cardHolderName	No	100	Name on credit/debit card.
amount	Yes	9	<p>The original subscription amount.</p> <p>Expected format: dddddd.cc (e.g. 105.15)</p>
currencyCode	Yes	3	The currency code used, for example: USD, EUR, GBP
startDate	No	19	<p>The creation date of the subscription.</p> <p>Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented</p>

					by this date.
endDate			Yes	19	The date the subscription will expire. Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this date. In case of an indefinite subscription plan, this field will be set to -1.
recurring			Yes		The recurrence number defined for this subscription on creation. In case of an indefinite subscription plan, this field will be set to -1.
periodUnit			Yes	5	The period unit defined for this subscription on creation. Possible values: day, week, month, year
periodNumber			Yes		The period number defined for this subscription on creation.
Payments			No		A Json array contains Json objects. Every Json object represents a payment of this subscription. The structure of these objects is the same as in the “getTransactionDetails” command.
user	firstName		No	50	Payer’s first name
	lastName		No	50	Payer’s last name
	email		No	50	Payer’s Email address
	phone	phoneNumber	No	10	Payer’s phone number
		countryCode	No	3	Payer’s phone number’s country code
	additionalDetails		No	400	Payer’s additional details
invoice	number		No	100	Invoice number as registered by the merchant
	additionalDetails		No	200	Invoice additional details as registered by the merchant
	items	id	No	100	Item’s id
		name	No	50	Item’s name
		price	No	9	Item’s price. Expected format: dddddd.cc (e.g. 105.15)
		quantity	No	5	Number of items Expected format: ddd.c (e.g. 105.1)
		additionalDetails	No	200	Invoice’s item additional details as registered by the merchant
addresses	billing	firstName	No	50	Billing address associated first name as registered by the merchant/user

		lastName	No	50	Billing address associated last name as registered by the merchant/user
		address1	No	50	Billing address (line 1) as registered by the merchant/user
		address2	No	50	Billing address (line 2) as registered by the merchant/user
		zipCode	No	50	Billing address zip code as registered by the merchant/user
		city	No	50	Billing address city as registered by the merchant/user
		state	No	50	Billing address state as registered by the merchant/user
		country	No	50	Billing address country as registered by the merchant/user
	shipping	firstName	No	50	Shipping address associated first name as registered by the merchant/user
		lastName	No	50	Shipping address associated last name as registered by the merchant/user
		address1	No	50	Billing address (line 1) as registered by the merchant/user
		address2	No	50	Billing address (line 2) as registered by the merchant/user
		zipCode	No	50	Shipping address zip code as registered by the merchant/user
		city	No	50	Shipping address city as registered by the merchant/user
		state	No	50	Shipping address state as registered by the merchant/user
		country	No	50	Shipping address country as registered by the merchant/user
riskScore			Yes	5	<p>The riskScore represents the likelihood that a given transaction is fraudulent. Merchants should use the riskScore to determine whether to accept, reject, manually review, or submit transactions to complementary services for further screening.</p> <p>The riskScore is given as a percentage, and as such it ranges from 0.01 to 100.00. For example, an order with a riskScore of 20.00 has a 20% chance of being fraudulent, while an order with a riskScore of 0.10 has a 0.1% chance of being fraudulent.</p>

## Example Response:

```
{
  "ResponseObject": {
    "amount": 0.5,
    "appName": "Kartis",
    "ZooZUniqueID": "zooz.unique.id",
    "cardHolderName": "first Approved",
    "currencyCode": "EUR",
    "endDate": "1344953560718",
    "fundSourceType": "VISA",
    "invoice": {
      "additionalDetails": "Power Ups for user 12345",
      "items": [{
        "additionalDetails": " ",
        "id": " ",
        "name": "Orange",
        "price": 0.5,
        "quantity": 1,
        "taxAmount": 0
      }],
      "number": "5512-FA"
    },
    "isSandbox": false,
    "lastFourDigits": "4580",
    "payments": [{
      "amount": 0.5,
      "appName": "Kartis ",
      "cardHolderName": "first Approved",
      "currencyCode": "EUR",
      "fundSourceType": "VISA",
      "invoice": {
        "additionalDetails": "Power Ups for user 12345",
        "items": [{
          "additionalDetails": " ",
          "id": " ",
          "name": "Orange",
          "price": 0.5,
          "quantity": 1,
          "taxAmount": 0
        }],
        "number": "5512-FA"
      },
      "isSandbox": false,
      "lastFourDigits": "4580",
      "paidAmount": 0.5,
      "transactionFee": 0.164,
      "transactionID": "UVU3GUVB66T4PG3ZIRDYDIG02Q",
      "transactionStatus": "Succeed",
      "transactionTimestamp": 1372767243000,
      "user": {
        "email": "support@zooz.com",
        "firstName": "first",
        "lastName": "Approved",
        "phone": {"phoneNumber": "01522113356"}
      }
    }],
    "periodUnit": "day",
    "periodNumber": 5,
    "recurring": 2,
    "startDate": "1372767243000",
    "status": "Valid",
    "subscriptionId": "BSSIWLNDNE42PNQA5YRPM6T64M",
    "user": {
      "email": "support@zooz.com",
      "firstName": "first",
      "lastName": "Approved",
      "phone": {"phoneNumber": "01522113356"}
    },
    "riskScore": 1.5
  },
  "ResponseStatus": "Success"
}
```

## 8. Cancel Subscription

This allows you to cancel a subscription.

The subscription status will be "CanceledByMerchant".

### Notice

Make sure to follow the [Http Addresses](#) & [Security instructions](#)

### Parameters

Parameter Name	Mandatory	Length	Description
cmd	Yes	100	Command Name. cmd="cancelSubscription"
ver	Yes	11	Extended Server API version number. This number affect on the naming convention and the response return by ZooZ service. Current version is: 2.0.2
subscriptionID	Yes	50	Subscription ID. Subscription ID is a unique identifier on ZooZ's servers as you get from the response whether it's from ZooZ's SDK response or callback response, after the user subscribes to your plan.

### Response

Parameter Name	Mandatory	Length	Description
endDate	Yes	19	The date the subscription will expire. Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this date. In case of an indefinite subscription plan, this field will be set to -1.
subscriptionId	Yes	50	Subscription ID. Subscription ID is a unique identifier on ZooZ's servers as you get from the response whether it's from ZooZ's SDK response or callback response, after the user subscribes to your plan.
subscriptionDisplayId	Yes		An easy to view id for the subscription.
isSandbox	Yes	true/ false	Whether the response comes from the Sandbox Server or Production Server.
subscriptionStatus	Yes		Sunscription status should be one of the

			<p>following:</p> <ul style="list-style-type: none"> <li>• Pending – Subscription has not yet been validated.</li> <li>• Valid – Subscription is valid.</li> <li>• Expired – Subscription is no longer valid. There will be no more payments.</li> <li>• CanceledByUser – The subscription was canceled by the user. The subscription will remain valid until the end of the period that was already paid for.</li> <li>• CanceledByMerchant – The subscription was canceled by the merchant. Either because a payment was refunded or charged back. The subscription is not valid.</li> </ul>
--	--	--	--

Response Objects will be returned in JSON format

Example Response:

```
{
  "responseObject": {
    "endDate": "1372767243000",
    "isSandbox": false,
    "status": "CanceledByMerchant",
    "subscriptionId": "BSSIWLNDNE42PNQA5YRPM6T64M"
  },
  "ResponseStatus": 0
}
```

## 9. Check Subscription

This allows you to check a subscription's status.

### Notice

Make sure to follow the [Http Addresses](#) & [Security instructions](#)

### Parameters

Parameter Name	Mandatory	Length	Description
cmd	Yes	100	Command Name. cmd="checkSubscription"
ver	Yes	11	Extended Server API version number. This number affect on the naming convention and the response return by ZooZ service. Current version is: 2.0.2
subscriptionID	Yes	50	Subscription ID. Subscription ID is a unique identifier on ZooZ's servers as you get from the response whether it's from ZooZ's SDK response or callback response, after the user subscribes to your plan.

### Response

The same as in CancelSubscription command.



## 10. Get Transaction Details By User ID

Getting transactions details by user ID API allows you to retrieve transaction full details and transaction status before or/and after commit or refund for all transactions associated with the above email. This call does not change the transaction state.

Notice

Make sure to follow the [Http Addresses](#) & [Security instructions](#)

Parameters

Parameter Name	Mandatory	Length	Description
cmd	Yes	100	Command Name. cmd="getTransactionDetailsByUserId"
ver	Yes	11	Extended Server API version number. This number affect on the naming convention and the response return by ZooZ service. Current version is: 2.0.2
userId	Yes	100	Unique identifier on ZooZ's servers for the user who processed this transaction. This value is returned in the Callback API response.
fromDate	No	10	Filter the results by dates.
toDate	No	10	Expected format: yyyy-mm-dd

### Response

Same as in "Get Transaction Details By Payer Email Address" – see above

## Callback API

ZooZ Transaction Callback is sent on each success transaction. It allows you to keep track of your purchases.

### Version

If you would like to upgrade your callback version, please contact ZooZ support.

### ZooZ Callback

1. Expose URL to be called by ZooZ (i.e. <http://www.myDomain.com/MyCallbackService>). You can easily change this address later.  
Note: Character encoding: UTF-8;  
Support Http methods: POST.
2. Supply this callback URL to ZooZ. (App settings in the ZooZ Portal)
3. ZooZ will call this URL with additional parameters in JSON format.
4. In order to authenticate the callback, you can validate it one of the following ways, we recommend using both:
  - Validate the source IP address of the call. You should expect to get it from the following IP addresses:
    - 54.200.242.136
    - 54.200.93.153
    - 54.201.49.185
    - 54.200.38.78
    - 54.69.146.25
    - 54.68.2.155
    - 54.68.182.134
    - 54.148.9.225
    - 54.69.192.240
  - Validate the correctness of the transaction using our `getTransactionDetails` API.
5. ZooZ Callback JSon object sturcture:

Parameter Name	Description	Mandatory	Length
appName	Name of application that created the payment as defined in Zooz portal.	Yes	50

ZooZUniqueID	The App Unique ID, as registered in the Zooz Developer Portal.	Yes	100
paymentId	Payment Id is a unique identifier on Zooz's servers for this payment.	Yes	50
isSandbox	Whether the response comes from the Sandbox Server or Production Server.	Yes	true/ false
paymentStatus	<p>Can be one of the following:</p> <ul style="list-style-type: none"> <li>• "Payment Approved" code: 1001 Transaction completed successfully.</li> <li>• "Payment Authorized, Pending completion" code: 1002 Transaction authorized and waiting for the merchant to commit.</li> <li>• "Payment Authorized, Pending user completion" code: 1003 Transaction authorized, has not yet been committed. User still hasn't completed the payment. (Merchant cannot commit this transaction)</li> <li>• "Refunded" code: 1004 The original transaction that was refunded.</li> <li>• "Refund due chargeback" code: 1005 Refund transaction due to chargeback received.</li> <li>• "Charged Back" code: 1006 The original transaction that was charged back.</li> <li>• "Refund" code: 1007 Refund transaction</li> <li>• "Pending user payment completion" code: 1008 Transaction is waiting for user action. (Merchant cannot commit this transaction)</li> <li>• "Voided" code: 1009 Transaction has been voided by the merchant.</li> <li>• "Payment Declined"</li> </ul>	Yes	

		code: 1010 Transaction was declined by the processor. <ul style="list-style-type: none"> <li>“Refund Declined” code: 1011 Refund transaction was declined</li> </ul>		
paymentStatusCode		Refer to paymentStatus above	Yes	
amount		The original payment amount. Expected format: dddddd.cc (e.g. 105.15)	Yes	
paidAmount		The actual paid amount. Expected format: dddddd.cc (e.g. 105.15)	Yes	
currencyCode		The currency code used, for example: USD, EUR, GBP	Yes	
taxAmount		The tax amount of the payment. Please note: amount = subtotal + taxAmount. This value is used to calculate the subtotal for the invoice.	Yes	
zoozFee		Zooz’s fee. Expected format: dddddd.cc (e.g. 105.15)	Yes	
paymentTimestamp		The creation date of the payment. Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this date.	Yes	
customerLoginID		customerLoginId is the customer’s identifier on the merchant’s server, allowing one click checkout from all devices.	No	
ipAddress		The IP address of this transaction.		
shippingMethod		The shipping method selected by the user for this payment.	No	100
shippingAmount		The price of the selected shipping method.	No	
shippingCarrierCode		The carrier code that matches the shipping method selected by the user.	No	200
shippingCarrierName		The carrier name that matches the shipping method selected by the user.	No	200
user	firstName	Payer’s first name	No	50
	lastName	Payer’s last name	No	50
	email	Payer’s email address	No	50

	phone	phoneNumber	Payer’s phone number		10	
		countryCode	Payer’s phone number’s country code		3	
	additionalDetails		Payer’s additional details	No	400	
	idNumber		Payer’s ID Number	No	20	
invoice	number		Invoice number as registered by the merchant	No	100	
	additionalDetails		Invoice additional details as registered by the merchant	No	200	
	items	id	Item’s ID	No	100	
		name	Item’s name	No	50	
		price	Item’s price. Expected format: dddddd.cc (e.g. 105.15)	No		
		quantity	Number of items Expected format: ddd.c (e.g. 105.1)	No		
		additionalDetails	Invoice item’s additional details as registered by the merchant	No	200	
		taxAmount	Invoice item’s tax amount.	No		
addresses	billing	firstName	Billing address associated with the first name registered by the merchant/user	No	50	
		lastName	Billing address associated with the last name registered by the merchant/user	No	50	
		address1	User's billing address – line 1	No	250	
		address2	User's billing address – line 2	No	250	
		address3	User's billing address – line 3	No	250	
		phone	phone Number	Billing address phone number	No	20
			country Code	Billing address phone number’s country code	No	3
		zipCode		Billing address zip code as registered by the merchant/user	No	50
		city		Billing address city as registered by the merchant/user	No	50
		state		Billing address state as registered by the merchant/user	No	50

		country	Billing address country as registered by the merchant/user	No	50
		countryCode	Billing address country code as registered by the merchant/user (Country code in ISO 3166 format)	No	2
	shipping	firstName	Shipping address associated with the first name registered by the merchant/user	No	50
		lastName	Shipping address associated with the last name registered by the merchant/user	No	50
		address1	User's shipping address – line 1	No	250
		address2	User's shipping address – line 2	No	250
		address3	User's shipping address – line 3	No	250
		phone	phone Number	No	20
			country Code	No	3
		zipCode	Shipping address zip code as registered by the merchant/user	No	50
		city	Shipping address city as registered by the merchant/user	No	50
		state	Shipping address state as registered by the merchant/user	No	50
		country	Shipping address country as registered by the merchant/user	No	50
		countryCode	Shipping address country code as registered by the merchant/user (Country code in ISO 3166 format)	No	2
	riskScore		<p>The riskScore represents the likelihood that a given payment is fraudulent. Merchants use the riskScore to determine whether to accept, reject, manually review, or submit payments to complementary services for further screening.</p> <p>The riskScore is given as a percentage, and, as such, it ranges from 0.01 to 100.00. For example, an order with a riskScore of 20.00 has a 20% chance of being fraudulent, while an order with a riskScore of 0.10 has a 0.1% chance of being fraudulent.</p> <p>If your order returns a risk score of –1, the risk score service is not active for your account, please contact Zooz support to</p>	Yes	

		activate the service.		
authorizationCode		Raw authorization code as received from the acquirer.	No	
processorReferenceId		Reference id for the request, received from the processor.	No	
country		The country from which the user processed the transaction, according to the ipAddress.	No	50
paymentType		Can be one of the following: <ul style="list-style-type: none"> <li>• Payment</li> <li>• Refund</li> <li>• Chargeback</li> </ul>	Yes	
authResult		Original authorization response as sent by the selected Acquirer.	No	
eci		The Numeric Commerce Indicator.	No	
paresStatus		Payer Authentication Response status.	No	
cavv		Cardholder Authentication Verification Value.	No	
fraudDetectionResponse		Fraud detection response as received from the processor.	No	
threeDSecureStatus		3D secure flow status: 0 – Success, 1 – Failure	No	
avsCode		Authorization avsCode response	No	
userId		A unique identifier on Zooz’s servers for the payment’s user.		
captures (JSON array)	amount	The captured amount.	No	
	referenceId	Reference id for the request, received from the processor.	No	
	timestamp	The capture date. Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this date.	No	
refunds (JSON array)	amount	The refunded amount.	No	
	referenceId	Reference id for the request, received from the processor.	No	
	timestamp	The refund date. Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this date.	No	
paymentMet	cardHolderName	Name on credit card.	No	100

hod	validDate		The expiration date of the credit card.  Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this date.	No	
	subtype		These are the values: <ul style="list-style-type: none"> <li>• VISA</li> <li>• MasterCard</li> <li>• Maestro</li> <li>• AmericanExpress</li> <li>• Discover</li> <li>• Diners</li> <li>• JCB</li> <li>• UnionPay</li> </ul>	No	
	paymentMethodToken		Representation of the credit card in Zooz's systems.	No	
	lastFourDigits		Last 4 digits of the credit card.	No	4
	paymentMethodType		These are the values: <ul style="list-style-type: none"> <li>• CreditCard</li> <li>• AliPay</li> </ul>	No	
	paymentMethodLastUsedTimestamp		The date of the last time the payment method was used.  Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this date.		
	billing Addresses	countryCode	User's billing country. Country code should be in ISO 3166 format.	No	2
		state	User's billing state	No	50
		city	User's billing city address	No	50
		address1	User's billing address – line 1	No	250
		address2	User's billing address – line 2	No	250
		address3	User's billing address – line 3	No	250
		zipCode	User's billing zip code	No	50
		firstName	User's billing first name	No	50
		lastName	User's billing last name	No	50
	phone	phoneNumber	User's billing address phone number	No	20



			countryC ode	User's billing address phone number's country code	No	3
--	--	--	-----------------	---	----	---

## Callback Structure Example

```
{
  "amount": 153.99,
  "appName": "Test App",
  "ZooZUniqueID": "zooz.unique.id",
  "authorizationCode": "A1106535",
  "captures": [{
    "amount": 153.99,
    "referenceId": "A1106536",
    "timestamp": 1404906259000
  }],
  "currencyCode": "EUR",
  "invoice": {
    "additionalDetails": "Great fruit",
    "items": [{
      "additionalDetails": " In season ",
      "id": "001",
      "name": " Orange ",
      "price": 50,
      "quantity": 2,
      "taxAmount": 0
    }],
    "number": "oo1"
  },
  "ipAddress": "123.123.123.123",
  "isSandbox": false,
  "paidAmount": 153.99,
  "paymentId": "UG4DAKZSL62QVVAVCV4SJ7355A",
  "paymentMethod": {
    "cardHolderName": "fv ffff",
    "lastFourDigits": "4580",
    "paymentMethodLastUsedTimestamp": 1404906255000,
    "paymentMethodToken": "PO2ONA26DGCZ7W5PRK6DBFEF4I",
    "paymentMethodType": "CreditCard",
    "subtype": "VISA",
    "validDate": 1522530000000
  },
  "paymentStatus": "Payment Approved",
  "paymentStatusCode": 1001,
  "paymentTimestamp": 1404906249000,
  "paymentType": "Payment",
  "processorReferenceId": "A1106535",
  "refunds": [],
  "riskScore": -1,
  "shippingAmount": 3.99,
  "shippingCarrierCode": "Fixed",
  "shippingCarrierName": "Fixed",
  "shippingMethod": "FedEx ground",
  "taxAmount": 10,
  "threeDSecureStatus": 0,
  "transactionID": 386857,
  "user": {
    "additionalDetails": "sdfds sdfs",
    "email": "john@doe.com",
    "firstName": "John ",
    "lastName": "Doe",
    "phone": {
      "countryCode": "123",
      "phoneNumber": "123456789"
    }
  },
  "userId": "3tS8WKgaHIN7IVImUXZVgg==",
  "zoozFee": 7.7
}
```

### Callback parser code example

The following code is an example how you should parse this callback.

This code is written in Java:

```
public void parseCallback(HttpServletRequest request) {
```

```
    BufferedReader br = null;
    String body = null;
    try {
        // read body as String
        br = request.getReader();
        String line = null;
        StringBuffer envelope = new StringBuffer();
        while ((line = br.readLine()) != null) {
            envelope.append(line);
        }
        body = envelope.toString();
    } catch (IOException e) {
        e.printStackTrace();
    } finally {
        if (br != null) {
            try {
                br.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
}
```

```
String applicationName;
String currencyCode;
double amount;
double paidAmount;
double taxAmount;
double zoozFee;
long paymentTimeStamp;
String paymentStatus;
int paymentStatusCode;
String ipAddress;
String paymentType;
String comment;
boolean isSandBox;
String userId;
String customerLoginId;
String country;
double shippingAmount;
String shippingMethod;
String shippingCarrierCode;
String shippingCarrierName;
double riskScore;
String authorizationCode;
String processorReferenceId;
JSONArray captures;
JSONArray refunds;
String avsCode;
String authResult;
String eci;
String paresStatus;
String cavv;
String transactionFdsResponse;
int threeDSecureStatus;
JSONObject userJson = null;
JSONObject invoiceJson = null;
JSONObject paymentMethodJson = null;
JSONObject billingAddressJson = null;
JSONObject shippingAddressJson = null;
```

```
JSONObject obj = new JSONObject(body);

if (!obj.isNull("addresses")) {
    JSONObject addressesJson = obj.getJSONObject("addresses");

    if (!addressesJson.isNull("shipping")) {
        shippingAddressJson = addressesJson.getJSONObject("shipping");
    }
    if (!addressesJson.isNull("billing")) {
        billingAddressJson = addressesJson.getJSONObject("billing");
    }
}

if (!obj.isNull("appName")) {
    applicationName = obj.getString("appName");
}

if (!obj.isNull("currencyCode")) {
    currencyCode = obj.getString("currencyCode");
}

if (!obj.isNull("paymentStatus")) {
    paymentStatus = obj.getString("paymentStatus");
}

if (!obj.isNull("paymentStatusCode")) {
    paymentStatusCode = obj.getInt("paymentStatusCode");
}

if (!obj.isNull("paymentType")) {
    paymentType = obj.getString("paymentType");
}

if (!obj.isNull("comment")) {
    comment = obj.getString("comment");
}

if (!obj.isNull("paidAmount")) {
    paidAmount = obj.getDouble("paidAmount");
}
if (!obj.isNull("amount")) {
    amount = obj.getDouble("amount");
}

if (!obj.isNull("taxAmount")) {
    taxAmount = obj.getDouble("taxAmount");
}

if (!obj.isNull("zoozFee")) {
    zoozFee = obj.getDouble("zoozFee");
}
if (!obj.isNull("paymentTimeStamp")) {
    paymentTimeStamp = obj.getLong("paymentTimeStamp");
}

if (!obj.isNull("invoice")) {
    invoiceJson = obj.getJSONObject("invoice");
}

if (!obj.isNull("user")) {
    userJson = obj.getJSONObject("user");
}

if (!obj.isNull("isSandBox")) {
    isSandBox = obj.getBoolean("isSandBox");
}
```

```
if (!obj.isNull("userId")) {
    userId = obj.getString("userId");
}

if (!obj.isNull("customerLoginId")) {
    customerLoginId = obj.getString("customerLoginId");
}

if (!obj.isNull("ipAddress")) {
    ipAddress = obj.getString("ipAddress");
}

if (!obj.isNull("country")) {
    country = obj.getString("country");
}

if (!obj.isNull("shippingMethod")) {
    shippingMethod = obj.getString("shippingMethod");
}

if (!obj.isNull("shippingAmount")) {
    shippingAmount = obj.getDouble("shippingAmount");
}

if (!obj.isNull("shippingCarrierCode")) {
    shippingCarrierCode = obj.getString("shippingCarrierCode");
}

if (!obj.isNull("shippingCarrierName")) {
    shippingCarrierName = obj.getString("shippingCarrierName");
}

if (!obj.isNull("riskScore")) {
    riskScore = obj.getDouble("riskScore");
}

if (!obj.isNull("authorizationCode")) {
    authorizationCode = obj.getString("authorizationCode");
}

if (!obj.isNull("processorReferenceId")) {
    processorReferenceId = obj.getString("processorReferenceId");
}

if (!obj.isNull("captures")) {
    captures = obj.getJSONArray("captures");
}

if (!obj.isNull("refunds")) {
    refunds = obj.getJSONArray("refunds");
}

if (!obj.isNull("avsCode")) {
    avsCode = obj.getString("avsCode");
}

if (!obj.isNull("authResult")) {
    authResult = obj.getString("authResult");
}

if (!obj.isNull("eci")) {
    eci = obj.getString("eci");
}

if (!obj.isNull("paresStatus")) {
    paresStatus = obj.getString("paresStatus");
}

if (!obj.isNull("cavv")) {
    cavv = obj.getString("cavv");
}

if (!obj.isNull("transactionFdsResponse")) {
    transactionFdsResponse = obj.getString("transactionFdsResponse");
}
```

```
if (!obj.isNull("threeDSecureStatus")) {  
    threeDSecureStatus = obj.getInt("threeDSecureStatus");  
}  
  
if (!obj.isNull("paymentMethod")) {  
    paymentMethodJson = obj.getJSONObject("paymentMethod");  
}  
  
System.out.println(obj.toString(5));
```

## Errors

The Extended server API may return error response. An error response is returned in Json. The format of the error response is as follows:

```
{
  "responseObject": {
    "responseErrorCode": 197121,
    "errorDescription": "Payment 548 does not exist, please contact support.",
    "errorMessage": "Payment 548 does not exist, please contact support."
  },
  "ResponseStatus": -1
}
```

Parameter Name	Type	Description
ResponseErrorCode	Integer	An error code != 0 that identifies the error.
errorMessage	String	This field will contain the old message.
errorDescription	String	This field will contain the error description.

The possible error codes are:

Error Code	Error Description
65793	Cannot login, your app unique-id or application key or platform does not match. Please make sure you registered your app correctly in the developer portal.
66049	Cannot login, your app-key is corrupted.
66050	Cannot login, your app-key is corrupted.
65794	Cannot login, your app-key does not match. Please make sure you use same app-key as given to you in the developer portal.
66051	Cannot login, your password is corrupted.
65795	Cannot login, password is incorrect.
66305	Cannot login, your partner id was not found. please contact support.
66306	Cannot login, your partner account is unauthorized. please contact support.
66307	Cannot login, your account id was not found. please contact support.
66308	Cannot login account for the specific command. Missing credential!
66309	Unable to find header parameter 'ZooZDeveloperId'. Please make sure you pass it and it's correct!
66310	Unable to find header parameter 'ZooZServerPassword'. Please make sure you pass it and it's correct!
66561	Authentication Failed!
66817	You can't login, it seems your device doesn't exist. Please contact the app developer to resolve this.
262549	Invoice total amount is different from the payment amount.
66562	Your account was locked since you exceeded the maximum login attempts. Please contact the app developer for more details.
66818	The username you entered doesn't exist. Please contact the app developer for help.
67073	Your account was disabled. Please contact the app developer for help.
67074	Your account is inactive, please contact the app developer for details.
66563	Login failed. Please try again shortly or contact the app developer for help.
65797	The passcode you typed is incorrect. Please try again to complete the payment.
67075	To keep your account secure, we had to block this device due to too many failed attempts. Please contact the app developer for help.
67076	Your mobile device is not active or blocked. Please contact the app developer for help.
65798	Cannot login, Token is missing.
66564	Authentication key is expired, Please login and press the link to resend authentication key.
66565	Please make sure you are using the latest authentication email we sent you.
66311	You exceeded the maximum CVV validation attempts, please contact support.

131329	Incorrect environment. A request was made to server, please make sure to match the the environment request from both your client side and your server side.
131844	Cannot open payment. on closed invoice.
131331	Credit card number is not valid. Please use test credit card number on sandbox environment.
131585	Payment token is corrupted.
131586	Payment token is mandatory.
131587	Unique app Device ID is mandatory.
131588	Unique app pin code is mandatory.
131589	Displayed image's link is incorrect.
131590	Shipping Methods is mandatory when requiring shipment
131591	Shipping Methods format is invalid
131592	Shipping Method "name" is missing
131593	Shipping Method "price" is missing
131600	Shipping Method "price" format is invalid
132097	Subscription's period number is missing or incorrect.
132098	Subscription's recurring number is missing or incorrect.
132099	Subscription's period unit is missing or incorrect.
131841	Invalid amount format or amount exceeded maximum value.
131330	You cannot run production transactions before get approved by ZooZ, please submit your app [{0}]to approval.
131842	Your version is outdated, please contact support to update your device. Command is invalid.
131843	Your version is outdated, please make sure you are using the correct API. Command is invalid.
132100	Subscription is not supported by processor. Please contact support.
196865	There is data inconsistency. Unable to retrieve transaction GUID, please contact support.
197121	Payment does not exist, please contact support.
197122	Invoice does not exist, please contact support.
197123	Point Of Sale does not exist, please contact support.
197124	Fund Source does not exist, please contact support.
197377	Fund Source was not verified!
197125	Sorry, the Loyalty Club does not exist, please contact the app developer.
197126	Inbox Message does not exist, please try to refresh the Inbox and try again.
197127	Sorry, cannot verify transaction, timeout exceeded. Please try again.
197128	User does not exist, please contact support.
197129	Username doesn't exist.
197136	Email address was not found, please try again.
197137	Mobile device does not exist, please contact support.
197138	Fund Source was already deleted.
197379	Currently this payment option is not enable for this payment, please contact support.
197379	Unable to identify processor, please contact support.
197139	Cannot find user with phone number X.
197140	Cannot find account.
197380	Payment option does not exist.
197141	Unable to locate program for developer.
197142	Unable to locate application for developer.
197143	Sorry, cannot open payment for user ID , the user does not exist.
196866	Unable to add mobile device, it seems that your device is already registered to another user. Please contact the app developer.
196867	Unable to update email, the email is already assigned to a different user. Select a different email or contact the app developer if you feel that your email was used without your consent.
196868	Unable to update ID Number. ID Number is already attached to a different user. Select a different ID Number or contact the app developer if you feel that your ID Number was used without your consent.
197381	ID Number does not match the ID Number that was stored for the user.
197382	Cannot create a new user, the phone number is already in use. Select a different phone number or contact the app developer if you feel that your phone number was used without your consent.
197144	Sorry, cannot find registered user with given phone number.
196869	Sorry, we were unable to process your request. Please contact the app developer.
196870	The system is inconsistent. Multiple SystemUser entries were found.
196871	The system is inconsistent. Multiple phone numbers entries were found.



196872	The system is inconsistent. Multiple ID Number entries were found.
197145	Developer was not found.
262401	Currency code has invalid format.
262402	Currency code is not supported at this time.
262532	Currency code cannot be empty.
262403	Pay amount has invalid format.
262404	Tax amount has invalid format.
262405	Tip cannot be negative. Please try again.
262406	A mismatch between the payment details and the total amount was found.
262407	Invalid date format .
262408	Authentication code is missing.
1033	Email is a mandatory field.
262416	Sorry, but there is a problem with the card number. Please check the number and try again.
262417	Card number cannot be empty!
262418	Card number is incorrect. Only digits are accepted.
393753	ZIP code length is incorrect. ZIP code length Should be 5 or 9 digits.
262419	Platform is incorrect!
262420	Platform is a mandatory field!
262421	Price range is incorrect!
262422	Price range is a mandatory field!
262657	PARAMETER Cannot be empty.
262658	PARAMETER has invalid value.
262659	VALUE is out of range.
262660	VALUE is invalid number.
262661	VALUE is not in a valid range. Should be between X and Y
262662	PARAMETER has invalid pattern
262663	Invalid Email format or email is too long!
262664	ID Number have to be with 7 to 9 digits only.
262665	ID is invalid. It should be a 7-9 digits number.
262672	ZIPCODE is invalid. It should be a 5 digit number.
262673	VALUE is not valid. It should be up to 20 characters long.
262674	Password is invalid. It should be 6-20 characters [a-z][A-Z][0-9][_!@#\$\$%^]
262675	Phone number is incorrect.
262676	Username is invalid.
262677	Name value is invalid. Name should be up to 50 characters!
262678	Bundle Id value is invalid. Only alpha numeric characters are allowed.
262679	Bundle Id already used by another app.
262680	Transaction GUID already exists.
262681	Payment Id already exists.
262688	Application description value is invalid. Only alpha numeric characters are allowed.
262689	Application name value is invalid. Only alpha numeric characters are allowed.
262690	Application name is mandatory!
262691	Invalid application link. Please enter a valid application link again.
262692	Invalid boolean type
262693	Cannot create a new payment for this operation. payment already exists!
262694	Country code should be with 1-3 digits only.
262695	Invalid URL format.
262696	Invalid name length, It should be up to 50 characters.
262697	Invalid account name length, It should be up to 50 characters.
262704	Invalid state length. It should be up to 50 characters.
262705	Invalid city length. It should be up to 50 characters.
262706	Invalid address 1 length. It should be up to 50 characters.
262707	Invalid address 2 length. It should be up to 50 characters.
262708	Invalid zip length. It should be up to 50 characters.
262709	Invalid info length. It should be up to 2000 characters.
262710	Invalid developer first name length. It should be up to 50 characters.
262711	Invalid developer last name length. It should be up to 50 characters.
262712	Invalid application description length. It should be up to 400 characters.

262713	Invalid bundle id length. It should contain 6 to 100 characters.
262720	Invalid application link length. It should be up to 100 characters.
262721	Invalid application name length. It should be up to 20 characters.
262722	Invalid User Agent. Please contact ZooZ support!
262723	PARAMETER should have X to Y characters.
262724	Email is mandatory.
262550	User ID is mandatory
262725	Field is mandatory
262726	Invalid website format, or website too long. (Website should not contains http://)
262727	Invalid ZIP format.
262728	PIN Code is mandatory.
262729	Username is mandatory.
262736	First name is mandatory.
262737	Last name is mandatory.
262738	CVV is mandatory.
262551	The data entered is too long.
262552	The data entered is too short.
262423	Transaction GUID cannot be empty.
262424	Phone number cannot be empty. Please insert a valid phone number and try again.
262425	Invalid format for phone number. Please insert a valid phone number and try again.
262739	Phone number is too long.
262740	Unable to complete operation. It seems that you entered a value that already exists in the system.
262432	Cannot add device to your account, invalid SMS code.
262433	Cannot add device to your account, invalid Passcode.
262434	Sorry, cannot complete transaction, the passcode you entered is incorrect, please try again.
262435	The user ID Number or Zip Code is missing!
262436	Logo is 60*60 png file, up to 100K.
262437	Invalid file extension, must be PNG or JPEG.
262521	The height and the width are different.
262528	The image width/height has to be between 57 to 120 pixels.
262438	Account email already exists. Select a different email or contact the app developer if you feel that your email was used without your consent.
262439	Selected Fund Source is not accepted by merchant. Please select a different fund source and try again.
262657	Sorry, cannot complete transaction , you exceeded the daily limitation on selected fund source. The available amount for transaction is X
262658	Sorry, cannot complete transaction - the amount entered exceeds daily limitation on selected merchant.
262659	Sorry, cannot complete transaction - the amount entered exceeds the transaction amount limitation.
262660	Sorry, cannot complete transaction , the amount entered exceeds monthly limitation on selected fund source. The available amount for transaction is X
262661	Sorry, cannot complete transaction - the maximum amount number X of money transfers was exceeded.
262662	Sorry, cannot complete transaction, the amount limitation for a single transaction was exceeded. The available amount for transaction is X
262663	Sorry, cannot complete transaction, the amount limitation was exceeded. The amount available for the transaction is X
262664	Cannot complete transaction, you exceeded the amount limitation for a single transaction. Please use other payment source
262440	The currency you selected is not supported. Please contact the app developer.
262665	Unable to execute refund. It seems that you exceeded the number of refunds allowed for this month. Please contact Support!
262441	Refund cannot be executed. your balance is lower than the refund amount. Please contact Support!
262448	Payment cannot be committed. Only 2 phase commit payment can be paid by merchant!
262449	You cannot commit an amount which is higher for more than 10 percent from the original amount.
262450	You cannot commit an amount which is higher for more than 50 percent from the original amount.
262451	Refund amount cannot be higher than the original transaction amount.
262452	Sorry, refund amount cannot be negative.
262453	Sorry, cannot reject payment. The payment is in process or was canceled. Please try again later.
262454	Payment was failed.
262455	Sorry, cannot update payment. The payment is in process or was canceled. Please try again later.

262456	Sorry, cannot cancel payment. The payment is in process or was canceled. Please try again later.
262457	Sorry, cannot cancel payment. Payment was already canceled.
262464	Payment amount cannot be zero. Please enter a different amount and try again.
262465	Sorry, you cannot add tip to refund. Please try again.
262466	Sorry, unsubscribing from loyalty club failed. Please try again later.
262467	Payment option was not verified or application does not accept this payment option.
262468	Incorrect point of sale type.
262469	Your app is not authorized yet to run transactions in production, please submit your app for approval.
262537	Your app is not authorized yet to initiate subscription request.
262470	Subscription failed. Please contact the app developer.
262471	Subscription expired. Please contact the app developer.
262472	Subscription does not exist for application.
262473	Unable to open new subscription , it was canceled or expired.
262480	Payment option not supported.
262481	This transaction is lower than the required minimum. Please change the amount and try again.
262482	Cannot complete payment. The merchant canceled the payment. Please refresh your view.
262483	Cannot completed payment, you already rejected this payment. Please refresh your view.
262484	Cannot complete payment, this payment was already paid. Please refresh your view.
262485	Cannot complete payment, payment's timeout exceeded. Please try again.
262486	Cannot complete payment, card expired. Please try again with a different payment source.
262487	Cannot complete payment, Paid amount does not equals to original amount.
262529	Cannot cancel payment. The payment has already been captured.
262530	Cannot cancel payment, the user rejected this payment.
262531	This payment is pending. A non-authorized payment cannot be canceled.
262488	Sorry, cannot cancel invoice. The invoice is in process or was canceled. Please try again later.
262489	Sorry, cannot update invoice. The invoice is in process or was canceled. Please try again later.
262496	Sorry, cannot close invoice. The invoice is in process or was canceled. Please try again later.
262497	Invoice total amount cannot be equal to zero. Please change the amount and try again.
262549	Invoice total amount is different from the payment amount.
262519	Sorry, cannot update invoice. Please check the invoice details.
262498	Sorry, you cannot transfer money to yourself.
262504	Sorry, Cannot complete registration, no more available attempts.
262505	Cannot add device. Authentication code was not found for this device
262512	Cannot add device, please contact the app developer.
262513	Cannot change PIN code, the old PIN code is incorrect.
262514	Forgot your passcode? Please contact the app developer in order to reset your passcode and continue the purchase.
262515	Failed to create account at Payoneer.
262516	Chargeback cannot be completed.
262517	Chargeback amount cannot be negative.
262518	Password is incorrect.
262533	UDID cannot be empty.
262534	Authorization failed. Please try again later or contact support.
262535	Failed to check subscription.
262536	This subscription was already canceled.
262544	Recurrence must be more than 1.
262545	Subscription id is mandatory.
262546	Unable to send IPN. Please enable IPN on your app's configuration.
262547	Unable to send IPN. Please define callback url on your app's configuration.
262548	Unable to send IPN. Please contact support.
327937	Operation Not allowed!
327938	Sorry, but we were unable to process your request. Please contact the app developer.
327939	Unable to load logo. Please try again.
327940	We're sorry, but something went wrong. Please contact the app developer for more details.
327941	Illegal state - no constraint was found in error.
327942	Action is not supported.
327943	No logo can be found!
327944	Cannot find resource name.

327945	Error to encrypt string.
327952	Error to decrypt string.
327953	Failed to parse Expiration Date
327954	There was problem to create csv file.
327955	Failed to parse user ID
393729	Transaction declined by the processing bank. Please verify that the card details were entered correctly or use another payment source.
393730	Sorry, cannot process transaction. Your card's CVV number is incorrect.
393731	Sorry, cannot complete transaction. You typed an invalid card number.
393732	It seems your card has expired. Please verify your details or use other payment source.
393733	Card was declined. We will not be able to process any funds for this card at this moment.
393734	The transaction cannot be completed at this time , your card was declined. Please try a different card.
393735	Sorry, your credit card is blacklisted. Please try a different card.
393985	Sorry, cannot complete transaction. Please verify that the bank account details were entered correctly.
393986	Sorry, your country is blacklisted.
393736	Sorry, cannot complete transaction, please verify you typed the card details correctly.
393761	Cannot complete transaction, please verify that the billing address was entered correctly
393473	Sorry, but we were unable to process your request at this time. Please try again later or contact the app developer.
393737	Failed to verify card.
393744	Card exceeded its limitation. Please use another payment method.
393987	Card is not approved to make transactions since there is no Checking Account. Please use another payment source.
393988	Card is not approved to make transactions since there is no Savings Account. Please use another payment source.
393745	Transaction declined by the processing bank. Please use another payment source.
393746	Transaction declined by the processing bank. Please use another payment source.
393747	Card is not approved to make transactions. Please use another payment source.
393748	Card is not approved to make transactions at the moment due to excessive authorization. Please use another payment source or try again later.
393752	Card was declined. We will not be able to process any funds for this card at this moment.
393474	Transaction was already reversed.
393475	Transaction already voided.
393476	Sorry, your transaction cannot be completed. Please contact the app developer.
393477	Sorry, transaction cannot be completed due to security timeout. Please try again.
393478	Sorry, transaction cannot be completed. Please try again.
393479	Sorry, transaction cannot be completed. Please try again or contact the app developer.
393749	Sorry, the card type you entered is not-supported.
393752	American Express is currently not supported.
393760	Prepaid is currently not supported.
394241	Sorry, transaction declined by the processing bank. Please use another payment source.
394242	Sorry, cannot complete transaction - missing funds in your prepaid account. Please use another payment source.
393750	Card type is not accepted.
393751	The card you're using has already been rejected once. We recommend that you either try another card or wait at least 30 minutes before trying again, to avoid automatic blocking by your card provider.
393480	Sorry, cannot complete refund process. Contact the app developer for help.
393481	Transaction was already refunded.
393488	Sorry, but refunds are not allowed at the moment. Please try again later.
393489	Sorry, cannot refund. The transaction was chargedback.
393490	Sorry, but partial refund is not allowed.
393491	Cannot refund payment. Refund timeout exceeded.
393494	Transaction is refund pending.
393492	Sorry, but we were unable to process your request. Please try again. If the problem persists contact the app developer.
393493	We were unable to process your request. Please try again. If the problem persists then contact support.
394497	Sorry, PayPal payment was failed.
394498	Unable to verify PayPal account. Please make sure to enter your email, first name and last name exactly

	as registered with PayPal.
394499	Cannot complete transaction, PayPal payment timeout has exceeded.
394500	This transaction cannot be processed by PayPal. The country in the shipping address must match the buyer's country of residence.