

TELIA OPERATOR SERVICE PLATFORM

OMA Payment REST API Guide

Copyright © 2026 Aepona Limited, and copyright © 2026 Telia

All rights reserved by respective owners.

Revision: 6.7

Legal Information

Legal Information

This document is made available by Telia Sverige AB (Telia) under Delivery contract No:4800010170 between Telia Sverige AB and Aepona Limited.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Telia and Aepona each disclaims all warranties, express and implied, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

The products and services described may contain defects or errors which may cause deviations from published specifications.

Telia and the Telia logo are trademarks of Telia AB in Nordics and other countries.

Aepona and the Aepona logo are trademarks of Aepona in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2026 Aepona Limited and copyright © 2026 Telia.

All rights reserved by respective owners.

Revision History

Revision History

Revision Number	Description	Revision Date
1.0	Initial release containing Release 3 scope for payment service (charge and reserve).	September 04 2015
2.0	Applied template and changed legal and copy right information from the PDF shared by Ray.	November 26 2015
2.1	Changed Accelerite Corporation to Aepona Limited	December 7, 2015
3.0	Language review	February 3, 2016
4.0	Added information about the request parameters maximum supported characters	May 20, 2016
5.0	Final release	July 14, 2016
5.1	Updated the description for Amount element	Nov 23, 2016
6.0	CR0018	April 25, 2017
6.1	Updated for CR0021 which for PSD compliance of Opaali platform	July 30, 2019
6.2	Copyright Year updated to 2020	February 14, 2020
6.4	Payment status api and api and http header parameters	April 14 2020
6.5	update the TRX_REF with generic value	Nov 18 2020
6.6	Updated the description of the "Description" parameter of table 6	Nov 19 2021
6.6	Copyright Year updated to 2022	May 20, 2022
6.7	Updated the Policy Exceptions section (Table 17) by adding new Error Code POL0252 which is introduced with CR034 Updated Copyright Year to 2023	

Contents

1	Payment REST Overview	6
1.1	Methods - Scope of this Document	6
1.2	Authentication	6
1.3	Using HTTP Headers	7
1.3.1	ORG_UNIT	7
1.3.2	TRX_REF	8
2	Resource Summary	8
2.1	URI and Operations List	8
2.2	URI Variables	9
2.3	Restrictions	9
2.4	Format Requirement	11
2.5	Parameter Data Structures	11
2.6	Use Cases for acr:URI	12
3	One-Step Charge	12
3.1	Charge a Subscriber	12
3.1.1	Request	12
3.1.2	Request Parameters	13
3.1.3	Response	15
3.1.4	Response Parameters	17
3.2	One Step Charge with acr: URI	17
3.2.1	Request	17
3.2.2	Response	19
3.3	Charge Amount Split	21
3.3.1	Request	21
3.3.2	Request Parameters	23
3.3.3	Response	24
3.3.4	Response Parameters	26
4	Reserve and Charge a Subscriber	27
4.1	Reserve Amount	27
4.1.1	Request	27
4.1.2	Request Parameters	28
4.1.3	Response	29
4.1.4	Response Parameters	31
4.2	Get Reserved Amount	32
4.2.1	Request	32
4.2.2	Request Parameters	33
4.2.3	Response	33
4.2.4	Response Parameters	33
4.3	Increase Reserved Amount	34

Contents

4.3.1	Request.....	34
4.3.2	Request Parameters	35
4.3.3	Response.....	35
4.3.4	Response Parameters	37
4.4	Charge Against Reservation.....	37
4.4.1	Request.....	37
4.4.2	Request Parameters	38
4.4.3	Response.....	38
4.4.4	Response Parameters	40
4.5	Release Reserved Amount.....	40
4.5.1	Request.....	40
4.5.2	Request Parameters	42
4.5.3	Response.....	42
4.5.4	Response Parameters	43
4.6	Reservation Operation with acr:URI	44
4.6.1	Request.....	44
4.6.2	Response.....	45
4.7	Charge Against Reservation with acr:URI.....	47
4.7.1	Request.....	47
4.7.2	Response.....	49
5	Payment status API.....	52
5.1	Get payment status	52
5.1.1	Request	52
5.1.2	Request Parameters	52
5.1.3	Response.....	53
5.1.4	Response Parameters	53
5.1.5	Error response	53
6	ResponseCodesandExceptions	55
6.1	Response Codes.....	55
6.2	Exceptions	55
6.2.1	Service Exceptions	56
6.2.2	Policy Exceptions.....	59

1 Payment REST Overview

The Telia-OMA Payment RESTful interface allows an application to perform payment operations including one-step charging, split-charging and reserve, and commit charging.

The API is implemented based on the following OMA specifications.

- OMA-TS-REST_NetAPI_Common-V1_0-20120417-C.pdf
- OMA-TS-REST_NetAPI_Payment-V1_0-20130924-A.pdf

1.1 Methods - Scope of this Document

This issue of the API Guide contains descriptions of the following Payment resources:

- Charge an amount to a subscriber
- Charge an amount shared amongst subscribers
- Reserve an amount for later charging, and by using the transaction ID and applicable transaction operation status values -
 - ↓ get the reserved amount
 - ↓ increase the reserved amount
 - ↓ release the reserved amount
 - ↓ charge against the reserved amount
- Get status of payment transactions

Use the GET and POST operations in OMA Payments.

Restrictions which apply to certain elements are described in [section 2.3](#).

1.2 Authentication

OSP supports one-way, server-side SSL, enabling HTTPS access sessions with a server-side certificate loaded on OSP. It does not support two-way SSL.

OMA Payment does not support `application/x-www-form-urlencoded` encoding format.

Operator Service Platform requires an access token, inserted in the Authorization request header field with the Bearer authorization scheme, as shown below and in the examples throughout this document.

Payment REST Overview

```
Authorization: Bearer <token value>
```

Obtain the access token by calling the OMA Authorization API, as described in *API_Telia-OMA_OAuth-REST.pdf*.

NOTE: Operator Service Platform API's support chunked transfer coding (Resource ref: HTTP 1.1 specification: <ftp://ftp.isi.edu/in-notes/rfc2616.txt>) in line with HTTP 1.1 protocol.

1.3 Using HTTP Headers

NSEE implements a number of HTTP headers which may (optionally) be present for payment operation types. This section describes the headers and the operation for which you use them. Use of the headers in operations other than those described here have no effect.

1.3.1 ORG_UNIT

Example JSON request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+358369102/transactions/amount
HTTP 1.1
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/json
Content-Type: application/json
ORG_UNIT: Finance
{
  "amountTransaction": {
    "clientCorrelator": "10000",
    "endUserId": "tel:+358369102",
    "paymentAmount": {
      "chargingInformation": {
        "amount": 8.99,
        "code": "TEST-012345",
        "currency": "EUR",
        "description": ["Default description"]
      }
    },
    "referenceCode": "REF-12345",
    "transactionOperationStatus": "Charged"
  }
}
```

This is an optional http header accepted in payment API. Payment transactions are counted for given header value and presented as a summary report in portal. Special characters such as Finish, and Cyrillic are not accepted.

Payment REST Overview

1.3.2 TRX_REF

Example JSON request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+358369102/transactions/amount
HTTP 1.1
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/json
Content-Type: application/json
TRX_REF: CompanyX_124fgh566
{
  "amountTransaction": {
    "clientCorrelator": "10000",
    "endUserId": "tel:+358369102",
    "paymentAmount": {
      "chargingInformation": {
        "amount": 8.99,
        "code": "TEST-012345",
        "currency": "EUR",
        "description": ["Default description"]
      }
    },
    "referenceCode": "REF-12345",
    "transactionOperationStatus": "Charged"
  }
}
```

This is an optional http header accepted in payment API. Applications can use this header to attach internal transaction id to the request. Special characters such as Finish, and Cyrillic are not accepted. The value of the header is recorded in reporting server.

2 Resource Summary

2.1 URI and Operations List

The URIs of the resources are as follows:

- One-Step Charge - section 3
 - Charge a subscriber - section 3.1
POST
<https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amount>
 - One Step Charge with acr:URI – section 3.2
POST
<https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amount>
 - Charge an amount split between subscribers - section 3.3
POST
<https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountsplit>
- Reserve and Charge a Subscriber - section 4
 - Reserve Amount - Section 4.1
POST
<https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation>
 - Get amount reserved for transaction - section 4.2
GET
<https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation/{transactionId}>
 - Increase the reserved amount - section 4.3
POST
<https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation/{transactionId}>
 - Charge against the reservation - section 4.4
POST
<https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation/{transactionId}>
 - Release reserve amount – section 4.5
POST

Resource Summary

`https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation/{transactionId}`

- Reservation Operation with acr:URI- section 4.6

POST

`https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation`

- Charge Against Reservation with acr:URI – section 4.7

POST

`https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation/{transactionId}`

- Get status of payment transaction section 5.1

GET

`https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/paymentStatus/{clientCorrelator}`

2.2 URI Variables

The variables used in the URIs are described below:

Table 1 URI Variable

Name	Description
{endUserId}	The unique reference ID for the user. In this case, the subscriber number, equivalent to the 'address' in a messaging service. See Section 2.3 Restrictions .
{transactionId}	The unique transaction ID generated by the server. The value returns in a successful response when the API creates a transaction.
{clientCorrelator}	This uniquely identifies the create charge request.

2.3 Restrictions

The following restrictions apply to Telia-OMA Payment API:

Table 2 Restriction

Parameter/element	Restriction
-------------------	-------------

Resource Summary

endUserId	<p>The following restrictions apply:</p> <ul style="list-style-type: none">❖ Only <i>tel:</i> and <i>acr</i> URIs are allowed (no support for <i>sip:</i> URI), when <code>TransactionOperationStatus = "Reserved"</code> and the request is the first in the sequence (no <code>transactionID</code> is present)❖ <i>tel:</i> which must be in international format beginning with + , followed by a minimum 9 digits including the country code. For example, <code>tel:+358123456</code>. Rejects numbers without any prefix.❖ only <i>tel:</i> URI allowed when,
-----------	--

Resource Summary

Parameter/element	Restriction
	TransactionOperationStatus = "Charged" or "Released" <ul style="list-style-type: none"> ❖ only <i>tel:</i> URI allowed when, TransactionOperationStatus = "Reserved" The request is not the first in the sequence (transactionIDis present) <ul style="list-style-type: none"> ❖ <i>sip:</i>, <i>:</i> and <i>short:</i> URIs are not supported.
callbackReference	This parameter is not supported.
chargingMetadata	This OMA data structure is not supported.
TransactionOperationStatus - Refunded	TransactionOperationStatus of Refunded is specifically not supported.

Violations of these rules returns an exception. See [Section 5.2 Exceptions](#) for details.

2.4 Format Requirement

The Telia-OMA Payment API supports JSON and XML content types only, for both request and response message bodies.

Throughout this document, the URL examples containing addresses may be shown WITHOUT URL encoding for readability purposes, for example, URL encoding for ':' and '+' are %3A and %2B respectively. Therefore, in the URL examples, tel:+1234567890 may be shown instead of "tel%3A%2B1234567890".

2.5 Parameter Data Structures

A defined data structure contains parameters sent in the body of the message, some of which are nested doubly or triply within other layers of data structure. These structures are noted by their Type, and parameters are described per data structure type, in separate tables.

For example, in the Charge a transaction sample in [section 3.1](#), the top level structure *transactions/amount* contains a number of simple elements as well as two objects each of which contains one or more sub-elements:

Direct elements of the primary structure, including the secondary structures, are presented in [Table 3](#); and elements of the respective secondary-structures are described in [Table 4](#) and [Table 5](#).

Resource Summary

2.6 Use Cases for acr:URI

Valid Case	Invalid Case
<p>The acr: URI is permitted in the following use cases:</p> <ul style="list-style-type: none">❖ amountTransaction❖ amountReservationTransaction (in initial reservation and in subsequent operations in the sequence where the initial reservation was made using acr: URI)	<p>The acr: URI is not permitted in the following use cases:</p> <ul style="list-style-type: none">❖ amountSplitTransaction❖ amountReservationTransaction (in subsequent operations where the initial reservation was made using the tel: URI)❖ any use case associated with the OMA Messaging API

3 One-Step Charge

3.1 Charge a Subscriber

Use the POST operation to charge a subscriber in a single step for a service provided by your application.

POST

`https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amount`

3.1.1 Request

Example JSON request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amount
HTTP 1.1
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/json
Content-Type: application/json
{
  "amountTransaction": {
    "clientCorrelator": "10000",
    "endUserId": "tel:+358369102",
    "paymentAmount": {
      "chargingInformation": {
        "amount": 8.99,
        "code": "TEST-012345",
        "currency": "EUR",
        "description": ["Default description"]
      }
    },
    "referenceCode": "REF-12345",
    "transactionOperationStatus": "Charged"
  }
}
```

One-Step Charge

Example XML request:

```

POST
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amount
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/xml
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- AMOUNT CHARGE TRANSACTION -->
<payment:amountTransaction
xmlns:payment="urn:oma:xml:rest:netapi:payment:1">
  <endUserId>tel:+358369102</endUserId>
  <paymentAmount>
    <chargingInformation>
      <description>Test idempotency-
"Charged"</description>
      <currency>EUR</currency>
      <amount>10</amount>
      <code>TEST-1</code>
    </chargingInformation>
  </paymentAmount>

<transactionOperationStatus>Charged</transactionOperationSta
tus>
  <referenceCode>REF-12345</referenceCode>
  <clientCorrelator>54321</clientCorrelator>
</payment:amountTransaction>

```

3.1.2 Request Parameters

Table 3 Transaction for charge amount (AmountTransaction Type)

Parameter	Data Type	Description	Optional
endUserId	anyURI	Only <i>tel:</i> URI is allowed, which must be in international format beginning with + , followed by a minimum 9 digits including the country code. For example, tel:+358123456. Numbers without any prefix is rejected. <i>sip:</i> , and <i>short:</i> URIs are not supported.	No

One-Step Charge

paymentAmount	paymentAmount	Provides a breakdown of the payment information. See Table 4.	No
transactionOperationStatus	TransactionOperationStatus	This indicates the desired resource state, with the value 'Charged'. See Table 5.	No
referenceCode	string	This should be unique per charge event. It provides a reference for reconciliation purposes. The operator should include it in reports so that they can match their view of what has been sold with yours. This parameter supports maximum 255 characters.	No
clientCorrelator	string	This uniquely identifies the create charge request. If there is a communication failure during the charge request, using the same clientCorrelator when retrying the request allows the operator to avoid applying the same charge twice. This parameter supports maximum 255 characters.	Yes

Table 4 Payment amount specific information (PaymentAmount Type)

Parameter	Data Type	Description	Optional
chargingInformation	ChargingInformation	Holds the charge with amount, currency and description text. See Table 6.	No
totalAmountCharged	decimal	The total amount which has been charged. Not used in Request; returned in Response.	Yes

Table 5 Enumeration of supported status resulting from transaction operations (TransactionOperationStatusType)

Enumeration	Description
Charged	In the request: indicates that the amount to charge. In the response: the amount has been charged, as stored as the final status of the operation of the transaction in the resource representation on the server.

Table 6 Charge Amount to End User's Bill Request Parameters (ChargingInformation)

Parameter	Data Type	Description	Optional
description	string, one or many	This parameter can be used as partner own internal reference to identify what the amount is for the item purchased, whether charged or reserved, and so on. Cyrillic characters are not supported. This parameter supports maximum 255 characters. Please note that the value of this parameter is NOT incorporated on end-user invoice.	No
currency	string	This is the 3 figure code as per ISO 4217. This parameter supports maximum 20 characters.	No
amount	decimal	This is the actual amount being charged or reserved, depending on the resource. It can be a whole number or decimal.	No

One-Step Charge

		<p>NOTE: The amount defined here for service price is excluding VAT.</p> <p>For values under €1, the decimal point must be proceeded by 0. For example, to denote 50 EURO CENT, enter 0.50, not .50.</p>	
code	string	Charging code, referencing a contract under which the charge is applied. This parameter supports maximum 255 characters.	Yes

3.1.3 Response

Example JSON response:

```
HTTP/1.1 201 Created
Accept: application/json
Accept-Encoding: gzip, deflate
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Content-Type: application/json; charset=UTF-8
Date: Thu, 26 May 2016 15:07:32 GMT
Location:
https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B358369102/transactions/amount/79530069-0ff0-45cc-8575-7cf819f9c4a7

{
  "amountTransaction": {
    "endUserId": "tel:+358369102",
    "paymentAmount": {
      "chargingInformation": {
        "description": [
          "Default description"
        ],
        "currency": "EUR",
        "amount": 8.99,
        "code": "TEST-012345"
      },
      "totalAmountCharged": 8.99
    },
    "transactionOperationStatus": "Charged",
    "referenceCode": "REF-12345",
    "clientCorrelator": "10000",
    "resourceURL":
      "https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B358369102/transactions/amount/79530069-0ff0-45cc-8575-7cf819f9c4a7"
  }
}
```

One-Step Charge

```
}  
}
```

Example XML response:

```
HTTP/1.1 201 Created  
accept: application/xml  
accept-encoding: gzip, deflate  
authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b  
content-type: application/xml; charset=UTF-8  
date: Mon, 06 Jun 2016 11:57:46 GMT  
location:  
https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B3  
58369102/transactions/amount/98c6013e-2874-  
4465967balee0bc3dc4f  
  
<?xml version="1.0" encoding="UTF-8"?>  
<payment:amountTransaction  
xmlns:payment="urn:oma:xml:rest:netapi:payment:1"  
xmlns:common="urn:oma:xml:rest:netapi:common:1">  
  <endUserId>tel:+358369102</endUserId>  
  <paymentAmount>  
    <chargingInformation>  
      <description>Test idempotency-  
"Charged"</description>  
      <currency>EUR</currency>  
      <amount>10</amount>  
      <code>TEST-1</code>  
    </chargingInformation>  
    <totalAmountCharged>10</totalAmountCharged>  
  </paymentAmount>  
  
<transactionOperationStatus>Charged</transactionOperationSta  
tus>  
  <referenceCode>REF-12345</referenceCode>  
  <clientCorrelator>54321</clientCorrelator>  
  
<resourceURL>https://api.opaali.telia.fi/production/payment/  
v1/tel%3A%2B358369102/transactions/amount/98c6013e-2874-  
4465967balee0bc3dc4f</resourceURL>  
</payment:amountTransaction>
```

One-Step Charge

3.1.4 Response Parameters

Returns the parameters passed in the request, with the addition of the resourceURL, containing the ID of the transaction created, and the totalAmountCharged.

Table 7 Transaction for charge amount (AmountTransaction Type)

Parameter	Data Type	Description	Optional
endUserId	anyURI	Only <i>tel:</i> URI is allowed, which must be in international format beginning with + , followed by a minimum 9 digits including the country code. For example, tel:+358123456. Numbers without any prefix is rejected. <i>sip:</i> and <i>short:</i> URIs are not supported.	No
paymentAmount	payment Amount	Provides a breakdown of the payment information, including totalAmountCharged. See Table 4.	No
transactionOperationStatus	TransactionOperationStatus	The resource state, with the value 'Charged'. See Table 5.	No
referenceCode	string	This is unique per charge event. It provides a reference for reconciliation purposes. The operator should include it in reports so that they can match their view of what has been sold with yours.	No
clientCorrelator	string	This uniquely identifies the create charge request. If there is a communication failure during the charge request, using the same clientCorrelator when retrying the request allows the operator to avoid applying the same charge twice.	Yes
resourceURL	anyURI	Self-referring URL. Contains the charge transaction ID.	Yes

3.2 One Step Charge with acr: URI

Use the POST operation to create charge operation when MSISDN is in encrypted format, where the charge is for a service provided by your application.

NOTE: If the acr value contains "/" character (ex :
acr:Fd7Yy3Z4GnHOG2QYDu//Tij/PVG3RPyHyvG0Xcw3cHU=), that character in the {endUserId} field in url should be replace with %2F (ex :
acr:Fd7Yy3Z4GnHOG2QYDu%2F%2FTij%2FPVG3RPyHyvG0Xcw3cHU=).
However, do not do the same for the endUserId field in request body.

3.2.1 Request

Example JSON request with *acr:* string:

One-Step Charge

```
POST
https://api.opaali.telia.fi/production/payment/v1/acr:wb8Wtc
pg5hxdNvt3LUDzR3kZwbX2+71U+LTag7npMXA=/transactions/amount
HTTP/1.1
Accept-Encoding: gzip,deflate
content-type: application/json
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
accept: application/json

{
  "amountTransaction": {
    "clientCorrelator": "ac40474a-be8e-49e4-b049-
f1de61685394",
    "endUserId":
"acr:wb8Wtcpg5hxdNvt3LUDzR3kZwbX2+71U+LTag7npMXA=",
    "paymentAmount": {
      "chargingInformation": {
        "amount": "1.2",
        "code": "FN-PAY-AMT-CRG-01-b",
        "currency": "EUR",
        "description": ["FN-PAY-AMT-CRG-01-b charge json"]
      }
    },
    "referenceCode": "FN-PAY-AMT-CRG-01-b",
    "transactionOperationStatus": "Charged"
  }
}
```

Example XML request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/acr:wb8Wtc
pg5hxdNvt3LUDzR3kZwbX2+71U+LTag7npMXA=/transactions/amount
HTTP/1.1
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/xml
Content-Type: application/xml
<?xml version="1.0" encoding="UTF-8"?>
<!-- AMOUNT CHARGE TRANSACTION -->
<payment:amountTransaction
xmlns:payment="urn:oma:xml:rest:netapi:payment:1">
<endUserId>acr:wb8Wtcpg5hxdNvt3LUDzR3kZwbX2+71U+LTag7npMXA=<
/endUserId>
<paymentAmount>
```

One-Step Charge

```
<chargingInformation>
<description>FN-PAY-AMT-CRG-01-a charge</description>
<currency>EUR</currency>
<amount>10</amount>
<code>TEST-1</code>
</chargingInformation>
</paymentAmount>
<transactionOperationStatus>Charged</transactionOperationSta
tus>
<referenceCode>REF-12345</referenceCode>
<clientCorrelator>54321</clientCorrelator>
</payment:amountTransaction>
```

NOTE: The request parameters are same as mentioned in 3.1.2 except endUserId field. It will be encrypted format of telephone number.

3.2.2 Response

Example JSON response with *acr*: string:

```
HTTP/1.1 201 Created
Accept: application/json
Accept-Encoding: gzip, deflate
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Content-Type: application/json; charset=UTF-8
Date: Thu, 26 May 2016 15:07:32 GMT
Location:
https://api.opaali.telia.fi/production/payment/v1/acr%3Awb8W
tcpg5hxdNvt3LUDzR3kZwbX2%2B71U%2BLTag7npMXA=/transactions/am
ount/765c5ac3-a207-49af-a1f2-c22f68026725
msisdn: +358405767255
{
  "amountTransaction" : {
    "endUserId" :
"acr:wb8Wtcpg5hxdNvt3LUDzR3kZwbX2+71U+LTag7npMXA=",
    "paymentAmount" : {
      "chargingInformation" : {
        "description" : [ "FN-PAY-AMT-CRG-01-b charge
json" ],
        "currency" : "EUR",
        "amount" : 1.2,
        "code" : "FN-PAY-AMT-CRG-01-b"
      },
    },
  },
}
```

One-Step Charge

```
    "totalAmountCharged" : 1.2
  },
  "transactionOperationStatus" : "Charged",
  "referenceCode" : "FN-PAY-AMT-CRG-01-b",
  "clientCorrelator" : "ac40474a-be8e-49e4-b049-
f1de61685394",
  "resourceURL" :
  "https://api.opaali.telia.fi/production/payment/v1/acr%3Awb8
Wtcpg5hxdNvt3LUDzR3kZwbX2%2B71U%2BLTag7npMXA=/transactions/a
mount/765c5ac3-a207-49af-a1f2-c22f68026725"
  }
}
```

Example XML response:

```
HTTP/1.1 201 Created
accept: application/xml
accept-encoding: gzip, deflate
authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
content-type: application/xml; charset=UTF-8
date: Mon, 06 Jun 2016 11:57:46 GMT
location:
https://api.opaali.telia.fi/production/payment/v1/acr:wb8Wtc
pg5hxdNvt3LUDzR3kZwbX2+71U+LTag7npMXA=/transactions/amount/9
8c6013e-2874-4465967ba1ee0bc3dc4f
Msisdn: +358405767255

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<payment:amountTransaction
xmlns:payment="urn:oma:xml:rest:netapi:payment:1"
xmlns:common="urn:oma:xml:rest:netapi:common:1">
<endUserId>acr:wb8Wtcpg5hxdNvt3LUDzR3kZwbX2+71U+LTag7npMXA=<
/endUserId>
<paymentAmount>
<chargingInformation>
<description>FN-PAY-AMT-CRG-01-a charge</description>
<currency>EUR</currency>
<amount>10</amount>
<code>TEST-1</code>
</chargingInformation>
<totalAmountCharged>10</totalAmountCharged>
</paymentAmount>
<transactionOperationStatus>Charged</transactionOperationSta
tus>
```

One-Step Charge

```
<referenceCode>REF-12345</referenceCode>
<clientCorrelator>54321</clientCorrelator>
<resourceURL>https://api.opaali.telia.fi/production/payment/
v1/acr:wb8Wtcpg5hxdNvt3LUDzR3kZwbX2+71U+LTag7npMXA=/transact
ions/amount/72c7efcc-e149-4c44-b591-
3aa7a5be2a8e</resourceURL>
</payment:amountTransaction>
```

NOTE: The response parameters are same as mentioned in 3.1.4.

NOTE: The msisdn header contains the decrypted value of acr. Http headers are case-insensitive, so Msisdn = msisdn = MSisdn = MslsDn.

3.3 Charge Amount Split

Use the POST operation to create an amount split transaction, where the charge is shared by multiple subscribers for a service provided by your application.

NOTE: Note: acr: URI is not supported with AmountSplitTransaction.

POST <https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountsplit>

3.3.1 Request

Example JSON request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amountSplit
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/json
Content-Type: application/json

{
  "amountSplitTransaction": {
    "clientCorrelator": "5000",
    "endUserShare": [
      {
        "endUserId": "tel:+358369102",
        "percent": "30"
      },
      {
        "endUserId": "tel:+358369103",
        "percent": "20"
      }
    ]
  }
}
```

One-Step Charge

```
    },
    {
      "endUserId": "tel:+358369104",
      "percent": "50"
    }
  ],
  "paymentAmount": {
    "chargingInformation": {
      "amount": 100,
      "code": "TEST-012345",
      "currency": "EUR",
      "description": [
        "Split Charge"
      ]
    }
  },
  "referenceCode": "REF-12345",
  "transactionOperationStatus": "Charged"
}
}
```

Example XML request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amountSplit
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/xml
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- AMOUNT SPLIT CHARGE TRANSACTION -->
<payment:amountSplitTransaction
xmlns:payment="urn:oma:xml:rest:netapi:payment:1">
  <endUserShare>
    <endUserId>tel:+358369102</endUserId>
    <percent>30</percent>
  </endUserShare>
  <endUserShare>
    <endUserId>tel:+358369103</endUserId>
    <percent>70</percent>
  </endUserShare>
</paymentAmount>
```

One-Step Charge

```

    <chargingInformation>
      <description>Test amount transaction
    "Charged"</description>
      <currency>EUR</currency>
      <amount>100</amount>
      <code>TEST-012345</code>
    </chargingInformation>
  </paymentAmount>

<transactionOperationStatus>Charged</transactionOperationSta
tus>
  <referenceCode>REF-12345</referenceCode>
  <clientCorrelator>54431</clientCorrelator>
</payment:amountSplitTransaction>

```

3.3.2 Request Parameters

Table 8 Transaction for charge a split amount (AmountSplitTransaction Type)

Parameter	Data Type	Description	Optional
endUserShare	EndUserShare [1..unbounded]	An array of endUserId and percentage split of the amount charged. See Table 9.	No
paymentAmount	PaymentAmount	Provides a breakdown of the payment information. See Table 4.	No
transactionOperationStatus	TransactionOperationStatus	This indicates the desired resource state, with the value 'Charged'. See Table 5.	No
referenceCode	string	Textual information to uniquely identify the request, such as in case of disputes. Used for business logic, not for operational logic	No
clientCorrelator	string	This uniquely identifies the create charge request. If there is a communication failure during the charge request, using the same clientCorrelator when retrying the request allows the operator to avoid applying the same charge twice.	Yes

Table 9 EndUserShare specific information (EndUserShare Type)

Parameter	Data Type	Description	Optional
endUserId	Any URI	The ID of the subscriber to be charged. Only <i>tel:</i> URI is allowed, which must be in international format beginning with + , followed by a minimum 9 digits including the country code. For example, tel:+358123456. Numbers without any prefix is rejected.	No

One-Step Charge

percent	Integer (positive value)	The percentage of the end user's share in the split amount transaction. The sum of all shares must equal 100.	No
---------	-----------------------------	---	----

3.3.3 Response

Example JSON response:

```
Accept-Encoding: gzip, deflate
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Content-Type: application/json; charset=UTF-8
Date: Thu, 26 May 2016 15:07:32 GMT
Location:
https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B4
47900000001/transactions/amountSplit/800a4794-cb99-4f67-
a6a0-7ce0401bc0c0
```

```
{
  "amountSplitTransaction": {
    "endUserShare": [
      {
        "endUserId": "tel:+358369102",
        "percent": 30
      },
      {
        "endUserId": "tel:+358369103",
        "percent": 20
      },
      {
        "endUserId": "tel:+358369104",
        "percent": 50
      }
    ],
    "paymentAmount": {
      "chargingInformation": {
        "description": [
          "Split Charge"
        ],
        "currency": "EUR",
        "amount": 100,
        "code": "TEST-012345"
      },
      "totalAmountCharged": 100
    }
  }
}
```

One-Step Charge

```
    },
    "transactionOperationStatus": "Charged",
    "referenceCode": "REF-12345",
    "clientCorrelator": "5000",
    "resourceURL":
    "https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B
    447900000001/transactions/amountSplit/800a4794-cb99-4f67-
    a6a0-7ce0401bc0c0"
  }
}
```

Example XML response:

```
HTTP/1.1 201 Created
accept: application/xml
authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
content-type: application/xml; charset=UTF-8
date: Mon, 06 Jun 2016 11:57:46 GMT
location:
https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B4
47900000001/transactions/amountSplit/ffc73ce7-8924-434e-
8ede-fec6178c0e19

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<payment:amountSplitTransaction
  xmlns:payment="urn:oma:xml:rest:netapi:payment:1"
  xmlns:common="urn:oma:xml:rest:netapi:common:1">
  <endUserShare>
    <endUserId>tel:+358369102</endUserId>
    <percent>30</percent>
  </endUserShare>
  <endUserShare>
    <endUserId>tel:+358369103</endUserId>
    <percent>70</percent>
  </endUserShare>
  <paymentAmount>
    <chargingInformation>
      <description>Test amount transaction
      &quot;Charged&quot; </description>
      <currency>EUR</currency>
      <amount>100</amount>
      <code>TEST-012345</code>
    </chargingInformation>
    <totalAmountCharged>100</totalAmountCharged>
```

One-Step Charge

```

    </paymentAmount>

    <transactionOperationStatus>Charged</transactionOperationSta
    tus>
    <referenceCode>REF-12345</referenceCode>
    <clientCorrelator>54431</clientCorrelator>

    <resourceURL>https://api.opaali.telia.fi/production/payment/
    v1/tel%3A%2B447900000001/transactions/amountSplit/ffc73ce7-
    8924-434e-8ede-fec6178c0e19</resourceURL>
    </payment:amountSplitTransaction>
  
```

3.3.4 Response Parameters

Returns the parameters passed in the request, with the addition of the resourceURL, containing the ID of the transaction created, and the totalAmountCharged.

Table 10 Transaction for charge a split amount (AmountSplitTransaction Type)

Parameter	Data Type	Description	Optional
endUserShare	EndUserShare [1..unbounded]	An array of endUserId and percentage split of the amount charged. See Table 9.	No
paymentAmount	PaymentAmount	Provides a breakdown of the payment information, including totalAmountCharged. See Table 4.	No
transactionOperationstatus	TransactionOperationStatus	The resource state, with the value 'Charged'. See Table 5.	No
referenceCode	string	Textual information to uniquely identify the request, such as in case of disputes. Used for business logic, not for operational logic	No
clientCorrelator	string	This uniquely identifies the create charge request. If there is a communication failure during the charge request, using the same clientCorrelator when retrying the request allows the operator to avoid applying the same charge twice.	Yes
resourceURL	anyURI	Self-referring URL. Contains the split charge transaction Id.	Yes

4 Reserve and Charge a Subscriber

4.1 Reserve Amount

- Use the POST operation to reserve an amount for charging the subscriber in a two-step operation. The transactionOperationStatus should be set to Reserved.
- **POST**
<https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation>

4.1.1 Request

Example JSON request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amountReservation
Content-type: application/json
Accept: application/json
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b

{"amountReservationTransaction":{
  "endUserId":"tel:+ 358369102",
  "paymentAmount":{"chargingInformation":{
    "currency":"EUR",
    "amount":"8.50",
    "code":"RES02Reserved",
    "description":["Test amount reservation
transaction - Reserved"]
  }},
  "transactionOperationStatus":"Reserved",
  "referenceSequence":"1",
  "referenceCode":"Reserve JSON"
}}
```

Example XML request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amountReservation
```

Reserve and Charge a Subscriber

```

Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/xml
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- AMOUNT RESERVATION TRANSACTION -->
<payment:amountReservationTransaction
xmlns:payment="urn:oma:xml:rest:netapi:payment:1">
<endUserId>tel:+358369102</endUserId>
<paymentAmount>
<chargingInformation>
<description>Test amount reservation transaction
"Reserved"</description>
<currency>EUR</currency>
<amount>10</amount>
<code>TEST-012345</code>
</chargingInformation>
</paymentAmount>
<transactionOperationStatus>Reserved</transactionOperationSt
atus>
<referenceSequence>1</referenceSequence>
<clientCorrelator>55555</clientCorrelator>
</payment:amountReservationTransaction>

```

4.1.2 Request Parameters

Table 11 Transaction for reserve or release amount (AmountReservationTransaction Type)

Parameter	Data Type	Description	Optional
endUserId	anyURI	Only tel: URI is allowed, which must be in international format beginning with + , followed by a minimum 9 digits including the country code. For example, tel:+358123456. Numbers without any prefix is rejected. sip:, acr: and short: URIs are not supported.	Yes
paymentAmount	PaymentAmount	Information on the amount reservation to be made. See Table 12.	No
transactionOperationStatus	TransactionOperationStatus	The desired resource state, with the value 'Reserved'. See Table 13.	No
referenceSequence	int	Sequential number generated by client application for every transaction state change.	No

Reserve and Charge a Subscriber

		<p>The client increments reference sequence with every new request to the server. If request failed, the client can repeat the request with the same sequence number. This allows the server to distinguish easily between new and repeated requests (such as ignore repeated requests, in the case they completed on the server side).</p> <p>Example of a sequence of usable referenceSequence values: use 1 to reserve amount X, then 2 to reserve additional amount Y, then 3 to charge reserved amount, and so on.</p>	
clientCorrelator	string	This uniquely identifies the reserve amount request. If there is a communication failure during the charge request, using the same clientCorrelator when retrying the request allows the operator to avoid applying the same charge twice.	Yes

Table 12 Payment amount specific information (PaymentAmount Type)

Parameter	Data Type	Description	Optional
chargingInformation	ChargingInformation	Holds the charge with amount, currency and description text. See Table 6.	No
totalAmountCharged	decimal	The total amount which has been charged. Used only in responses in various Reserve and Charge a Subscriber methods.	Yes
amountReserved	decimal	The amount which has been reserved. Used only in responses in various Reserve and Charge a Subscriber methods.	Yes

Table 13 Enumeration of supported status resulting from transaction operations (TransactionOperationStatusType)

Enumeration	Description
Charged	In the request: charge the amount In the response: the amount has been charged
Reserved	In the request: reserve the amount In the response: the amount has been reserved
Released	In the request: release the reservation In the response: the reservation is released (implicitly by expiration or explicitly by release operation)

4.1.3 Response

Example JSON response:

```
HTTP/1.1 201 Created
```

Reserve and Charge a Subscriber

```
Accept-Encoding: gzip, deflate
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Content-Type: application/json; charset=UTF-8
Date: Thu, 26 May 2016 15:07:32 GMT
Location:
https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B35842356001/transactions/amountReservation/ec08112a-243b-4af0-bdce-9f9ea5158730

{
  "amountReservationTransaction" : {
    "endUserId" : "tel:+35842356001",
    "paymentAmount" : {
      "chargingInformation" : {
        "description" : [ "Test amount reservation
transaction - Reserved" ],
        "currency" : "EUR",
        "amount" : 8.50,
        "code" : "RES02Reserved"
      },
      "totalAmountCharged" : 0,
      "amountReserved" : 8.50
    },
    "transactionOperationStatus" : "Reserved",
    "referenceSequence" : 1,
    "referenceCode" : "Reserve JSON",
    "resourceURL" :
    "https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B35842356001/transactions/amountReservation/ec08112a-243b-4af0-bdce-9f9ea5158730"
  }
}
```

Example XML response:

```
HTTP/1.1 201 Created
accept: application/xml
authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
content-type: application/xml; charset=UTF-8
date: Mon, 06 Jun 2016 11:57:46 GMT
location:
https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B35842356001/transactions/amountReservation/453c6986-1d46-4312-b32d-6c77f09e29c
```

Reserve and Charge a Subscriber

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<payment:amountReservationTransaction
xmlns:payment="urn:oma:xml:rest:netapi:payment:1"
xmlns:common="urn:oma:xml:rest:netapi:common:1">
  <endUserId>tel:+35842356001</endUserId>
  <paymentAmount>
    <chargingInformation>
      <description>Test amount reservation transaction
Reserved</description>
      <currency>EUR</currency>
      <amount>10</amount>
      <code>TEST-012345</code>
    </chargingInformation>
    <totalAmountCharged>0</totalAmountCharged>
    <amountReserved>10</amountReserved>
  </paymentAmount>
  <transactionOperationStatus>Reserved</transactionOperati
onStatus>
  <referenceSequence>1</referenceSequence>
  <referenceCode>Reserve XML</referenceCode>
  <resourceURL>
https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B3584
2356001/transactions/amountReservation/453c6986-1d46-4312-b32d-
6c77f09e29c8</resourceURL>
</payment:amountReservationTransaction>
```

4.1.4 Response Parameters

Table 14 Transaction for reserve or release amount (AmountReservationTransaction Type)

Parameter	Data Type	Description	Optional
endUserId	anyURI	Only <i>tel:</i> URI is allowed, which must be in international format beginning with + , followed by a minimum 9 digits including the country code. For example, tel:+358123456. Numbers without any prefix is rejected.	Yes
paymentAmount	PaymentAmount	Information on the amount charge to be made. See Table 12.	No
transactionOperationStatus	TransactionOperationStatus	The desired resource state, with value as appropriate. See Table 13.	No
referenceSequence	int	Sequential number generated by client application for every transaction state change.	No

Reserve and Charge a Subscriber

		<p>The client increments reference sequence with every new request to the server. If request failed the client can repeat the request with the same sequence number. This allows the server to distinguish easily between new and repeated requests (such as ignore repeated requests, in the case they completed on the server side).</p> <p>Example of a sequence of usable referenceSequence values: use 1 to reserve amount X, then 2 to reserve additional amount Y, then 3 to charge reserved amount, and so on.</p>	
clientCorrelator	string	This uniquely identifies the reserve amount request. If there is a communication failure during the charge request, using the same clientCorrelator when retrying the request allows the operator to avoid applying the same charge twice.	Yes
resourceURL	anyURI	Self-referring URL. The resourceURL Contains the amount reservation transaction ID.	Yes

4.2 Get Reserved Amount

Use the GET operation to return the reservation transaction details for a subscriber, by referencing the transaction ID of the initial reservation.

- **GET**
- <https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation/{transactionId}>

4.2.1 Request

Example JSON request:

```
GET
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amountReservation/53918e2a-3080-4469-8ceb-
f75fd20b6b49
HTTP/1.1
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/json
Content-Type: application/json
```

Example XML request:

```
GET
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amountReservation/53918e2a-3080-4469-8ceb-
f75fd20b6b49
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/xml
```

Reserve and Charge a Subscriber

Content-Type: application/xml

4.2.2 Request Parameters

The transaction ID received in the response to Reserve Amount call is required for the URL. There is no body in this request.

4.2.3 Response

The GET response is in exactly the same format as the one returned for Reserve Amount request but the contents may differ. This is determined by the previous request which may not always be a 'Reserve'.

Table 15 Transaction for get reserved amount (AmountReservation Type)

Parameter	Data Type	Description	Optional
endUserId	anyURI	Only <i>tel:</i> URI is allowed, which must be in international format beginning with + , followed by a minimum 9 digits including the country code. For example, tel:+358123456. Numbers without any prefix is rejected.	Yes
paymentAmount	PaymentAmount	This is the representation of the most recent successful "Reserved" or "Charged" operation;	No
transactionOperationStatus	TransactionOperationStatus	Taken from the most recent successful "Reserved" or "Charged" operation.	No
referenceSequence	int	Taken from the most recent successful "Reserved" or "Charged" operation.	No
clientCorrelator	string	Taken from the most recent successful "Reserved" or "Charged" operation.	Yes
totalAmountCharged	decimal	The cumulative amount that has been charged across all "Charged" operations for this reservation	Yes
amountReserved	decimal	The remaining reservation amount.	Yes
resourceURL	anyURI	Self-referring URL. The resourceURL Contains the amount reservation transaction Id.	Yes

4.2.4 Response Parameters

Returns the same parameters as for the response to the Reserve Amount.

Reserve and Charge a Subscriber

4.3 Increase Reserved Amount

Use the POST operation to add to a reservation made earlier, by referencing the transaction ID of the initial reservation made for the subscriber.

- POST
`https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation/{transactionId}`

4.3.1 Request

Example JSON request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amountReservation/9ebf0b73-a346-4b9d-bc42-
6ac66c49df78
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/json
Content-Type: application/json

{
  "amountReservationTransaction": {
    "clientCorrelator": "1234",
    "endUserId": "tel:+358369102",
    "paymentAmount": {
      "chargingInformation": {
        "amount": 100.00,
        "code": "TEST-012345",
        "currency": "EUR",
        "description": ["Increase Reserve description"]
      }
    },
    "referenceSequence": "2",
    "transactionOperationStatus": "Reserved"
  }
}
```

Example XML request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+358369102/transact
ions/amountReservation/444a9f87-9cee-45b7-95f5-3b0ed8b6ef3a

Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
```

Reserve and Charge a Subscriber

```
Accept: application/xml
Content-Type: application/xml
<?xml version="1.0" encoding="UTF-8"?>
<payment:amountReservationTransaction
xmlns:payment="urn:oma:xml:rest:netapi:payment:1">
<endUserId>tel:+358369102</endUserId>
<paymentAmount>
<chargingInformation>
<description>Test increase reservation</description>
<currency>EUR</currency>
<amount>15</amount>
<code>TEST012345</code>
</chargingInformation>
</paymentAmount>
<transactionOperationStatus>Reserved</transactionOperationSt
atus>
<referenceSequence>2</referenceSequence>
<referenceCode>REF-12345</referenceCode>
</payment:amountReservationTransaction>
```

4.3.2 Request Parameters

Use the same parameters as for the request for Reserve Amount. Use the transaction ID received in the response to the Reserve Amount.

4.3.3 Response

Example JSON response:

```
HTTP/1.1 200 Created
Accept-Encoding: gzip,deflate
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Content-Type: application/json; charset=UTF-8
Date: Thu, 26 May 2016 15:07:32 GMT

{"amountReservationTransaction": {
  "endUserId": "tel:+358369102",
  "paymentAmount": {
    "chargingInformation": {
      "description": ["increase Reserve description"],
      "currency": "EUR",
      "amount": 100,
      "code": "TEST-012345"
```

Reserve and Charge a Subscriber

```
    },
    "totalAmountCharged": 0,
    "amountReserved": 100
  },
  "transactionOperationStatus": "Reserved",
  "referenceSequence": 2,
  "clientCorrelator": "12314",
  "resourceURL":
  "https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B
  447900000001/transactions/amountReservation/9ebf0b73-a346-
  4b9d-bc42-6ac66c49df78"
}}
```

Example XML response:

```
HTTP/1.1 200 Created
accept: application/xml
authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
content-type: application/xml; charset=UTF-8
date: Mon, 06 Jun 2016 11:57:46 GMT

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<payment:amountReservationTransaction
  xmlns:payment="urn:oma:xml:rest:netapi:payment:1"
  xmlns:common="urn:oma:xml:rest:netapi:common:1">
  <endUserId>tel:+358369102</endUserId>
  <paymentAmount>
    <chargingInformation>
      <description>Test increase
reservation</description>
      <currency>EUR</currency>
      <amount>15</amount>
      <code>TEST012345</code>
    </chargingInformation>
    <totalAmountCharged>0</totalAmountCharged>
    <amountReserved>5</amountReserved>
  </paymentAmount>

  <transactionOperationStatus>Reserved</transactionOperationSt
atus>
  <referenceSequence>2</referenceSequence>
  <referenceCode>REF-12345</referenceCode>
  <clientCorrelator>55555</clientCorrelator>
  <resourceURL>
```

Reserve and Charge a Subscriber

```
https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B9
12890369102/transactions/amountReservation/d1f0da5e-0348-
478b-99f8-5a91bf750ef1</resourceURL>
</payment:amountReservationTransaction>
```

4.3.4 Response Parameters

Returns the same parameters as for the response to the Reserve Amount, with the updated reservedAmount.

4. Charge Against Reservation

Use the POST operation to charge the subscriber for the reserved amount, by referencing the transaction ID created for the reservation.

- **POST**

`https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation/{transactionId}`

4.4.1 Request

Example JSON request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amountReservation/ec08112a-243b-4af0-bdce-
9f9ea5158730
Content-type: application/json
Accept: application/json
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b

{"amountReservationTransaction":{
  "endUserId":"tel:+358369102",
  "paymentAmount":{"chargingInformation":{
    "currency":"EUR",
    "amount":"8.50",
    "code":"RES02Charged",
    "description":["Test amount reservation
transaction - Charged"]
  }},
  "transactionOperationStatus":"Charged",
  "referenceSequence":"2",
  "referenceCode":"Charged JSON"
}
```

Reserve and Charge a Subscriber

```
}
```

Example XML request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amountReservation/d1f0da5e-0348-478b-99f8-
5a91bf750ef1
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/xml
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<payment:amountReservationTransaction
xmlns:payment="urn:oma:xml:rest:netapi:payment:1">
<endUserId>tel:+358369102</endUserId>
<paymentAmount>
<chargingInformation>
<description>Test increase reservation</description>
<currency>EUR</currency>
<amount>5</amount>
<code>TEST012345</code>
</chargingInformation>
</paymentAmount>
<transactionOperationStatus>Charged</transactionOperationSta
tus>
<referenceSequence>2</referenceSequence>
<referenceCode>REF-12345</referenceCode>
</payment:amountReservationTransaction>
```

4.4.2 Request Parameters

Use the same parameters as for the request for Reserve Amount. Use the transaction ID received in the response to the Reserve Amount call, and set the transactionOperationStatus to *Charged*.

4.4.3 Response

Example JSON response:

```
HTTP/1.1 200 Created
Accept-Encoding: gzip,deflate
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Content-Type: application/json; charset=UTF-8
```

Reserve and Charge a Subscriber

```
Date: Thu, 26 May 2016 15:07:32 GMT

"amountReservationTransaction" : {
  "endUserId" : "tel:+358369102",
  "paymentAmount" : {
    "chargingInformation" : {
      "description" : [ "Test amount reservation
transaction - Charged" ],
      "currency" : "EUR",
      "amount" : 8.50,
      "code" : "RES02Charged"
    },
    "totalAmountCharged" : 8.50,
    "amountReserved" : 0.00
  },
  "transactionOperationStatus" : "Charged",
  "referenceSequence" : 2,
  "referenceCode" : "Charged JSON",
  "resourceURL" :
  "https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B
358369102/transactions/amountReservation/ec08112a-243b-4af0-
bdce-9f9ea5158730"
}
}
```

Example XML response:

```
HTTP/1.1 200 Created
accept: application/xml
authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
content-type: application/xml; charset=UTF-8
date: Mon, 06 Jun 2016 11:57:46 GMT

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<payment:amountReservationTransaction
  xmlns:payment="urn:oma:xml:rest:netapi:payment:1"
  xmlns:common="urn:oma:xml:rest:netapi:common:1">
  <endUserId>tel:+358369102</endUserId>
  <paymentAmount>
    <chargingInformation>
      <description>Test increase
reservation</description>
      <currency>EUR</currency>
```

Reserve and Charge a Subscriber

```
        <amount>5</amount>
        <code>TEST012345</code>
    </chargingInformation>
    <totalAmountCharged>5</totalAmountCharged>
    <amountReserved>5</amountReserved>
</paymentAmount>

<transactionOperationStatus>Charged</transactionOperationSta
tus>
    <referenceSequence>2</referenceSequence>
    <referenceCode>REF-12345</referenceCode>
    <clientCorrelator>55555</clientCorrelator>
    <resourceURL>
https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B3
58369102/transactions/amountReservation/19f68fb3-2e5f-4113-
82a9-de75bb2b79c5</resourceURL>
</payment:amountReservationTransaction>
```

4.4.4 Response Parameters

Returns the same parameters as for the response to the Reserve Amount, with updated values for `totalAmountCharged` and `amountReserved`. The `transactionOperationStatus` value is *Charged*.

4.5 Release Reserved Amount

Use the POST operation to remove a reservation made earlier, by referencing the transaction ID of the reservation made for the subscriber. Set the `transactionOperationStatus` to `Released`.

- **POST**
`https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation/{transactionId}`

4.5.1 Request

Example JSON request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amountReservation/9ebf0b73-a346-4b9d-bc42-
6ac66c49df78
HTTP/1.1
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/json
```

Reserve and Charge a Subscriber

```
Content-Type: application/json

{
  "amountReservationTransaction": {
    "endUserId": "tel:+358369102",
    "paymentAmount": {
      "chargingInformation": {
        "code": "TEST012345",
        "description": ["Test amount release"]
      }
    },
    "referenceSequence": "3",
    "transactionOperationStatus": "Released"
  }
}
```

Example XML request:

```
POST
https://api.opaali.telia.fi/production/payment/v1/tel:+35836
9102/transactions/amountReservation/08ed554e-ddda-44ca-87db-
6110e8c0aa28
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/xml
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<payment:amountReservationTransaction
xmlns:payment="urn:oma:xml:rest:netapi:payment:1">
<endUserId>tel:+358369102</endUserId>
<paymentAmount>
<chargingInformation>
<description>Test release reservation</description>
<code>TEST012345</code>
</chargingInformation>
</paymentAmount>
<transactionOperationStatus>Released</transactionOperationSt
atus>
<referenceSequence>3</referenceSequence>
<referenceCode>REF-12345</referenceCode>
</payment:amountReservationTransaction>
```

Reserve and Charge a Subscriber

4.5.2 Request Parameters

Use the same parameters as for the request for Reserve Amount, for `currency` and `amount` where these are omitted. Use the transaction ID received in the response to the Reserve Amount call, and set the `transactionOperationStatus` to *Released*.

4.5.3 Response

Example JSON response for tel:

```
HTTP/1.1 200 Created
Accept-Encoding: gzip,deflate
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Content-Type: application/json; charset=UTF-8
Date: Thu, 26 May 2016 15:07:32 GMT

{"amountReservationTransaction": {
  "endUserId": "tel:+358369102",
  "paymentAmount": {
    "chargingInformation": {
      "description": ["Test amount release"],
      "amount": 50.74,
      "code": "TEST012345"
    },
    "totalAmountCharged": 50.25,
    "amountReserved": 0
  },
  "transactionOperationStatus": "Released",
  "referenceSequence": 3,
  "clientCorrelator": "12314",
  "resourceURL":
  "https://api.opaali.telia.fi/production/payment/v1/tel%3A%2B
  358369102/transactions/amountReservation/cb4a8f94-4df7-4cdb-
  8182-63bc23e4e58d"
}}
```

Example XML response for tel:

```
HTTP/1.1 200 Created
accept: application/xml
authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
content-type: application/xml; charset=UTF-8
date: Mon, 06 Jun 2016 11:57:46 GMT
```

Reserve and Charge a Subscriber

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<payment:amountReservationTransaction
  xmlns:payment="urn:oma:xml:rest:netapi:payment:1"
  xmlns:common="urn:oma:xml:rest:netapi:common:1">
  <endUserId>tel:+358369102</endUserId>
  <paymentAmount>
    <chargingInformation>
      <description>Test release
reservation</description>
      <amount>5</amount>
      <code>TEST012345</code>
    </chargingInformation>
    <totalAmountCharged>5</totalAmountCharged>
    <amountReserved>0</amountReserved>
  </paymentAmount>

  <transactionOperationStatus>Released</transactionOperationSt
atus>
  <referenceSequence>3</referenceSequence>
  <referenceCode>REF-12345</referenceCode>
  <clientCorrelator>55555</clientCorrelator>

  <resourceURL>https://api.opaali.telia.fi/production/payment/
v1/tel%3A%2B358369102/transactions/amountReservation/08ed554
e-ddda-44ca-87db-6110e8c0aa28</resourceURL>
</payment:amountReservationTransaction>
```

4.5.4 Response Parameters

Returns the same parameters as for the response to the Reserve Amount, with the `amountReserved` value updated to 0. The `transactionOperationStatus` value is *Released*.

Reserve and Charge a Subscriber

4.6 Reservation Operation with acr:URI

Use the POST operation to create reservation operation.

POST

https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation

NOTE: If the acr value contains "/" character

(ex : acr:Fd7Yy3Z4GnHOG2QYDu//Tij/PVG3RPyHyvG0Xcw3cHU=)
that character in {endUserId} field in the url should be replace with %2F (ex :
acr:Fd7Yy3Z4GnHOG2QYDu%2F%2FTij%2FPVG3RPyHyvG0Xcw3cHU=).
However, do not do the same for the endUserId field in request body.

4.6.1 Request

Example JSON request with *acr: endUserId*:

```
POST
https://api.opaali.telia.fi/production/payment/v1/
acr:mO+TEEnwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=/transaction
s/amountReservation
Content-type: application/json
Accept: application/json
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b

{
  "amountReservationTransaction": {
    "clientCorrelator": "f29a42ed-d482-44e9-81fa-
8f4c1459a5cb",
    "endUserId":
"acr:mO+TEEnwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=",
    "paymentAmount": {
      "chargingInformation": {
        "amount": 5,
        "code": "TEST-012345",
        "currency": "EUR",
        "description": ["FN-CR010-AMT-RES-01- reserve json"]
      }
    },
    "referenceSequence": 1,
    "transactionOperationStatus": "Reserved"
  }
}
```

Reserve and Charge a Subscriber

Example XML request with *acr: endUserId*:

```
POST https://api.opaali.telia.fi/production/payment/v1/
acr:mO+TEEnwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=/transaction
s/amountReservation
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/xml
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- AMOUNT RESERVATION TRANSACTION -->
<payment:amountReservationTransaction
xmlns:payment="urn:oma:xml:rest:netapi:payment:1">
<endUserId>acr:mO+TEEnwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=<
/endUserId>
<paymentAmount>
<chargingInformation>
      <description>FN-CR010-AMT-RES-01- reserved
xml</description>
      <currency>EUR</currency>
      <amount>5</amount>
      <code>TEST-012345</code>
</chargingInformation>
</paymentAmount>
<transactionOperationStatus>Reserved</transactionOperationSt
atus>
<referenceSequence>1</referenceSequence>
</payment:amountReservationTransaction>
```

NOTE: The request parameters are same as mentioned in 4.4.2 except *endUserId* field. It will be encrypted format of telephone number.

4.6.2 Response

Returns the response

Example JSON response with *acr: string*:

```
HTTP/1.1 201 Created
Accept-Encoding: gzip, deflate
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Content-Type: application/json; charset=UTF-8
Date: Thu, 26 May 2016 15:07:32 GMT
```

Reserve and Charge a Subscriber

```
Location: https://api.opaali.telia.fi/production/payment/v1/
acr:mO+TEEnwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=/transaction
s/amountReservation/ec08112a-243b-4af0-bdce-9f9ea5158730
Content-type: application/json
Accept: application/json
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b

{"amountReservationTransaction":{
  "endUserId":"acr:mO+TEEnwiwbNGz44gEj0ewFXM2GZI9xByQpniT6N
MqIQ=",
  "paymentAmount":{"chargingInformation":{
    "currency":"EUR",
    "amount":"8.50",
    "code":"RES02Charged",
    "description":["Test amount reservation
transaction - Charged"]
  },
  "totalAmountCharged" : 0,
  "amountReserved" : 8.50
},
  "transactionOperationStatus":"Charged",
  "referenceSequence":"2",
  "referenceCode":"Charged JSON"
}
}
```

Example XML response:

```
HTTP/1.1 201 Created
accept: application/xml
authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
content-type: application/xml; charset=UTF-8
date: Mon, 06 Jun 2016 11:57:46 GMT
location: https://api.opaali.telia.fi/production/payment/v1/
acr:mO+TEEnwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=/transaction
s/amountReservation/d1f0da5e-0348-478b-99f8-5a91bf750ef1
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/xml
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
```

Reserve and Charge a Subscriber

```
<payment:amountReservationTransaction
xmlns:payment="urn:oma:xml:rest:netapi:payment:1">
<endUserId>acr:mO+TEnwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=<
/endUserId>
<paymentAmount>
<chargingInformation>
<description>Test increase reservation</description>
<currency>EUR</currency>
<amount>5</amount>
<code>TEST012345</code>
</chargingInformation>
<totalAmountCharged>0</totalAmountCharged>
<amountReserved>5</amountReserved>
</paymentAmount>
<transactionOperationStatus>Charged</transactionOperationSta
tus>
<referenceSequence>2</referenceSequence>
<referenceCode>REF-12345</referenceCode>
</payment:amountReservationTransaction>
```

NOTE: The response parameters are same as mentioned in 4.4.4.

4.7 Charge Against Reservation with acr:URI

Use the POST operation to charge the subscriber for the reserved amount, by referencing the transaction ID created for the reservation. After succesfull request, msisdn header is returned with decrypted msisdn.

- **POST**
<https://api.opaali.telia.fi/production/payment/v1/{endUserId}/transactions/amountReservation/{transactionId}>

NOTE: If the acr value contains "/" character
(ex : acr:Fd7Yy3Z4GnHOG2QYDu//Tij/PVG3RPyHyvG0Xcw3cHU=)
that character in {endUserId} field in the url should be replace with %2F (ex :
acr:Fd7Yy3Z4GnHOG2QYDu%2F%2FTij%2FPVG3RPyHyvG0Xcw3cHU=).
However, do not do the same for the endUserId field in request body.

4.7.1 Request

Example JSON request:

Reserve and Charge a Subscriber

```
POST
https://api.opaali.telia.fi/production/payment/v1/
acr:mO+TENwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=/transaction
s/amountReservation/ec08112a-243b-4af0-bdce-9f9ea5158730
Content-type: application/json
Accept: application/json
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b

{"amountReservationTransaction":{
  "endUserId": "
acr:mO+TENwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=",
  "paymentAmount":{"chargingInformation":{
    "currency":"EUR",
    "amount":"8.50",
    "code":"RES02Charged",
    "description":["Test amount reservation
transaction - Charged"]
  }},
  "transactionOperationStatus":"Charged",
  "referenceSequence":"2",
  "referenceCode":"Charged JSON"
}
}
```

Example XML request:

```
POST https://api.opaali.telia.fi/production/payment/v1/
acr:mO+TENwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=/transaction
s/amountReservation/d1f0da5e-0348-478b-99f8-5a91bf750ef1
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/xml
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<payment:amountReservationTransaction
xmlns:payment="urn:oma:xml:rest:netapi:payment:1">
<endUserId>acr:mO+TENwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=<
/endUserId>
<paymentAmount>
<chargingInformation>
<description>Test increase reservation</description>
<currency>EUR</currency>
<amount>5</amount>
<code>TEST012345</code>
```

Reserve and Charge a Subscriber

```
</chargingInformation>
</paymentAmount>
<transactionOperationStatus>Charged</transactionOperationSta
tus>
<referenceSequence>2</referenceSequence>
<referenceCode>REF-12345</referenceCode>
</payment:amountReservationTransaction>
```

NOTE: The request parameters are same as mentioned in 4.4.2 except endUserId field. It will be encrypted format of telephone number.

4.7.2 Response

Example JSON response:

```
HTTP/1.1 200 Created
Accept-Encoding: gzip,deflate
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Content-Type: application/json; charset=UTF-8
Date: Thu, 26 May 2016 15:07:32 GMT
```

```
msisdn: +358405767255
```

```
"amountReservationTransaction" : {
  "endUserId" : "
acr:mO+TENwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=",
  "paymentAmount" : {
    "chargingInformation" : {
      "description" : [ "Test amount reservation
transaction - Charged" ],
      "currency" : "EUR",
      "amount" : 8.50,
      "code" : "RES02Charged"
    },
    "totalAmountCharged" : 8.50,
    "amountReserved" : 0.00
  },
  "transactionOperationStatus" : "Charged",
  "referenceSequence" : 2,
  "referenceCode" : "Charged JSON",
  "resourceURL" :
  "https://api.opaali.telia.fi/production/payment/v1/
```

Reserve and Charge a Subscriber

```
acr:mO+TEEnwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=/transaction
s/amountReservation/ec08112a-243b-4af0-bdce-9f9ea5158730"
  }
}
```

Example XML response:

```
HTTP/1.1 200 Created
accept: application/xml
authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
content-type: application/xml; charset=UTF-8
date: Mon, 06 Jun 2016 11:57:46 GMT
msisdn: +358405767255

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<payment:amountReservationTransaction
  xmlns:payment="urn:oma:xml:rest:netapi:payment:1"
  xmlns:common="urn:oma:xml:rest:netapi:common:1">
  <endUserId>
acr:mO+TEEnwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=</endUserId>
  <paymentAmount>
    <chargingInformation>
      <description>Test increase
reservation</description>
      <currency>EUR</currency>
      <amount>5</amount>
      <code>TEST012345</code>
    </chargingInformation>
    <totalAmountCharged>5</totalAmountCharged>
    <amountReserved>5</amountReserved>
  </paymentAmount>

<transactionOperationStatus>Charged</transactionOperationSta
tus>
  <referenceSequence>2</referenceSequence>
  <referenceCode>REF-12345</referenceCode>
  <clientCorrelator>55555</clientCorrelator>
  <resourceURL>
https://api.opaali.telia.fi/production/payment/v1/
acr:mO+TEEnwiwbNGz44gEj0ewFXM2GZI9xByQpniT6NMqIQ=/transaction
s/amountReservation/19f68fb3-2e5f-4113-82a9-
de75bb2b79c5</resourceURL>
</payment:amountReservationTransaction>
```

Payment status

NOTE: The response parameters are same as mentioned in 4.4.4.

NOTE: The msisdn header contains the decrypted value of acr. Http headers are case-insensitive, so Msisdn = msisdn = MSisdn = MslsDn.

5 Payment status API

5.1 Get payment status

Payment applications can invoke payment status API and get status of the payment transactions, the application sent previously. All the payment APIs return either success or fail status in API call response. In addition, application can request status of sent payment transactions using payment status api as well.

Within 24 hours of the original transaction, application can invoke the payment status API and get status of the transaction.

GET <https://api.opaali.telia.fi/production/payment/v1/paymentStatus/{clientCorrelator}>

5.1.1 Request

Example JSON request:

```
GET
https://api.opaali.telia.fi/production/payment/v1/tel:+358369
102/transactions/paymentStatus/53918e2a-3080-4469-8ceb-
f75fd20b6b49 HTTP/1.1
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Accept: application/json
Content-Type: application/json
```

5.1.2 Request Parameters

Parameter	Data Type	Description	Optional
clientCorrelator	string	This is a unique identifier sent in payment requests.	No
endUserId	String	The unique reference ID for the user. In this case, the subscriber number	No

Response codes and exceptions

5.1.3 Response

Example JSON response:

```
HTTP/1.1 200 Ok
Accept:application/json
Accept-Encoding: gzip,deflate
Authorization: Bearer 4ba50412-9c7e-42e3-b1b4-a6b8ed75565b
Content-Type: application/json; charset=UTF-8
Date: Thu, 26 May 2016 15:07:32 GMT
{
  "paymentStatus": {
    "opaaliTrxID": "1de02cdd-6420-425b-93fd-
c6f95fedd3d3",
    "trxReference": "Apple_fkdi434300",
    "clientCorrelator" : "e850e002-f690-4b9a-b76d-
6b6747601996",
    "status": "Success"
  }
}
```

5.1.4 Response Parameters

Charge / Split charge / charge against reservations are considered as payment requests.,

Parameter	Data Type	Description	Optional
opaaliTrxID	string	This is the transaction id returned in payment call response. Charge / Split charge / charge against reservations are considered as payment requests.	Yes
trxReference	string	This is the application transaction reference (TRX_REF) send in payment api calls.	Yes
clientCorrelator	string	This is the unique id sent in each payment request	No
status	string	Status of the payment transaction. Either Success or Fail	No

5.1.5 Error response

Opaali bound API requests pass through several layers of security before it reaches actual payment applications. Malformed requests

Response codes and exceptions

such as invalid json formats may fail on those layers and return error response to the calling application. If you send a 'payment status' call to such a request Opaali may not be able to find such payment request in its records hence will return bellow error.

```
HTTP/1.1 400 Bad Request
{
  "requestError":{
    "serviceException":{
      "messageId":"SVC5000",
      "text":"Unknown transaction for the clientCorrelator
%1",
      "variables":[
        "e850e002-f690-4b9a-b76d-6b6747601996"
      ]
    }
  }
}
```

6 Response Codes and Exceptions

6.1 Response Codes

HTTP response codes are used to indicate:

- **200** – Success!
- **201** – Created
- **400** – Bad request; check the error message for details. Generally returned for a service exception.
- **401** – Authentication failure, check your authentication details.
- **403** – Forbidden; please provide authentication credentials. Returned for policy exceptions.
- **404** – Not found: mistake in the host or path of the service URI
- **405** – Method not allowed: The actual HTTP method (such as GET, PUT, POST, and DELETE) is not supported by the resource.
- **409** – Conflict: Occurs in situations when two instances of an application are trying to modify a resource in parallel, in a non- synchronized way.
- **500** – Internal Error; the server encountered an unexpected condition which prevented it from fulfilling the request
- **503** – Server busy and service unavailable. Please retry the request.

For more details, refer to <http://www.ietf.org/rfc/rfc2616.txt>.

6.2 Exceptions

This section lists the available error codes, the possible reasons why the exception may have occurred, and possible solutions.

The example below shows a service exception, when the client correlator is duplicated:

```
HTTP/1.1 409 Conflict
Connection: Close
content-type: application/json
date: Fri, 12 Feb 2016 08:43:40 GMT
Content-Length: 221
{
  "requestError" : {
    "serviceException" : {
      "messageId" : "SVC0005",
```

Response codes and exceptions

```
"text" : "Correlator %1 specified in message part %2 is a
duplicate",
"variables" : [ "121211sdsds55", "clientCorrelator" ]
}
}
```

All error response return the messageId and text lines; some, like above, have the variables line.

The example below shows a policy exception, for POL0251:

```
{
  "requestError" : {
    "policyException" : {
      "messageId" : "POL0251",
      "text" : "Split Charging is not supported",
    }
  }
}
```

6.2.1 Service Exceptions

A service exception describes the reason why the service cannot accept the request.

The following service exceptions may be thrown for message requests:

Table 16 Payment Request Exceptions

HTTP Exception Codes	Explanation
400 SVC0002	Invalid input value, for transactionId, endUserId, and so on: Invalid input value for message part %1 Followed by <variables> with the parameter name. endUserId: : only tel: and acr: URIs allowed (no support for sip: URI) Unable to decode acr: URI acr: present where transactionOperationStatus = "Charged" or Released" acr: present where transactionOperationStatus = "Reserved" and transactionID also present. acr: In case of Split Charging, SVC0002 is received.
404 SVC0004	Missing element: No valid addresses provided in message part %1

Response codes and exceptions

	Followed by <variables> with the parameter name.
403 SCV0270	General Payment Error (failure to take payment): Charging operation failed, the charge was not applied.
400 SVC0271	Total of EndUserShare not equal to 100%: Sum of percentage allocation is not equal to 100
403 POL1000	Prepaid Subscriber with Insufficient Credit or the charged amount exceeds the reservation amount.
409 SVC0005	Duplicate clientCorrelator: See example above. Correlator %1 specified in message part %2 is a duplicate Followed by <variables> with the value and the parameter name.
403 SVC0009	Strong authentication failure for subscriber <MSISDN>.Reason is <Reason> Parameter values are returned as a variable array in the order as defined in error text. This exception is thrown for following reasons. <ul style="list-style-type: none">- Subscriber does not have a mobile certificate in strong authentication service side.- Subscriber rejects strong authentication.
403 SVC4000	Strong authentication failure for subscriber <MSISDN>.Reason is <Reason>.Try again later. Parameter values are returned as a variable array in the order as defined in error text. This exception is thrown for following reasons. <ul style="list-style-type: none">- Subscriber enters wrong PIN code for strong authentication request.- Strong authentication request has already been sent to the subscriber
503 SVC4000	Strong authentication failure for subscriber <MSISDN>.Reason is <Reason>.Try again later. Parameter values are returned as a variable array in the order as defined in error text. This exception is thrown for following reasons. <ul style="list-style-type: none">- Subscriber authentication fails due to timeout
400 SVC4000	Strong authentication failure for subscriber <MSISDN>.Reason is <Reason>.Try again later. Parameter values are returned as a variable array in the order as defined in error text. This exception is thrown for following reasons. <ul style="list-style-type: none">- Strong Identification endpoint not available

Response codes and exceptions

	<p>Unkown transaction for the clientCorrelator <client correlator ></p> <p>Parameter values are returned as a variable array in the order as defined in error text.</p> <p>This exception is thrown for following reasons.</p> <ul style="list-style-type: none">- Opaali cannot find a payment transaction for the given client correlator
--	---

Response codes and exceptions

6.2.2 Policy Exceptions

A policy exception means that the request syntax is valid, but Operator ServicePlatform does not support it.

- **POL0250** – Too many endUserIdentifiers; HTTP response code 403
- **POL0251** – Split charging policy not supported; HTTP response code 403
- **POL0254** – Charge limit policy; amount exceeded limit; HTTP response code 403
- **POL200x** – General policy error occurred; HTTP response code 403
- **POL30xx** – Operator Service Platform specific service policy error occurred; HTTP response code 403
- **POL31xx** – Operator Service Platform specific subscriber related error occurred; HTTP code 403

POL2000 and 3000 range of exceptions may be thrown to indicate a fault relating to a policy associated with the service, as listed in [Table 17](#):

Table 17 Policy Exceptions

Error Code	Error Text/Action Required
POL2000 Resource not supported	The following policy error occurred: OMA Resource <resource name> not supported. Error code is 2000.
POL2000 Verb not supported	The following policy error occurred: Verb <verb name> not supported in OMA resource <resource name>. Error code is 2000.
POL2000 Element not supported	The following policy error occurred: Element <element name> not supported in OMA resource <resource name>. Error code is 2000.
POL2000 Value not supported	The following policy error occurred: Value <value> not supported in OMA element <element name>. Error code is 2000.
POL1002: Max transactions per interval is enforced, and you have made more requests.	The charge happened too soon after the previous one.: Maximum Transactions per Interval Exceeded. Error code is 1002. Check your SLA for the maximum transaction rate allowed, for example 10 TPS. Ensure that the rate of your requests is within the limits.
POL3004: Max transactions is enforced, and max requests has been exceeded	The following policy error occurred: Maximum Transactions Exceeded. Error code is 3004. Check your SLA for the maximum transactions allowed,
POL3006:	The following policy error occurred: Destination Whitelist. Error code is 3006.

Response codes and exceptions

Destination White List is enforced, and address is not in Destination White List	Enforces a destination white list and the number is not in the white list. Rejects the address. Check your SLA details.
POL3007: Destination Black List is enforced and address is in Destination Black List.	The following policy error occurred: Destination Blacklist. Error code is 3007. Enforces a destination black list and the number is on the list. Rejects the address.
POL3008: Sender White List is enforced and address is not in Sender White List.	The following policy error occurred: Sender Address Whitelist. Error code is 3008. Enforces a sender white list and the number is not in the list. Rejects the address.
POL3009: Sender Black List is enforced and address is in Sender Black List.	The following policy error occurred: Sender Address Blacklist. Error code is 3009. Enforces a sender black list and the number is on the list. Rejects the address.
POL3012: AmountTransaction policy	The following policy error occurred: AmountTransaction not allowed. Error code is 3012. Returns the error when the policy/operation is disabled for the application.
POL3013: AmountReservationTransaction policy	The following policy error occurred: AmountReservationTransaction not allowed. Error code is 3013.
POL3014: Charge Currency policy	The following policy error occurred: Only Partner's own currency allowed. Error code is 3014. Returns the error when the policy is set to the partner's currency only, and the transaction uses another currency.
POL3017: Purchase limit exceeded for billing period	Subscriber <MSISDN> purchase limit exceeded for billing period. Parameter values are returned as a variable array in the order as defined in error text. This error is thrown when a subscriber has exceeded the total purchase limit for billing period.
POL3018: Prepaid subscribers block policy	Subscriber <MSISDN> Service Disabled.Reason is <Reason>. Parameter values are returned as a variable array in the order as defined in error text. The error is returned when a payment act transaction is made on behalf of a prepaid subscriber. If an application is having attribute APPLICATION_PAYMENT_ACT=yes or APPLICATION_PAYMENT_ACT=no and message price with vat >= 50 is considered as a payment act transaction.
POL3101: Missing Subscriber Data	Subscriber <endUserId> is not found. Error code is 3101. Returns the error when the subscriber's entry is not found in the database.

Response codes and exceptions

<p>POL3102: Service Barred</p>	<p>Subscriber <endUserId> Service Barred. Error Code 3102</p> <p>EITHER the <i>Subscriber_Access_Status</i> is PASSIVE and the service cannot proceed;</p> <p>OR the <i>Subscriber_Access_Status</i> is ACTIVE but the subscriber is barred from receiving the particular content type.</p>
<p>POL3103: Service Disabled</p>	<p>Subscriber <endUserId> Service Disabled. Error Code 3103.</p> <p>Returns the error when the subscription is no longer active.</p>
<p>POL3204: Unknown Transaction ID</p>	<p>Unknown TransactionID. Error Code is 3204.</p> <p>Returns the error when the GET reserved transaction call includes a transaction ID that is not found, or a valid transaction but not one related to the calling application.</p>
<p>POL0252: Fraud Detection/Aggregated Charge Limit Policy</p>	<p>An exception is returned when the subscriber's aggregated charge amount exceeds the aggregated charge limit for a given period.</p> <p>HTTP response code: 403</p> <p>Error Text: "Subscriber <MSISDN> exceeded the maximum of mobile payment purchases allowed for given period"</p>