

SDP Solution API Reference (MobileMoney, Security)





Change History

Changes between document issues are cumulative. The latest document issue contains all the changes made in earlier issues.

Issue 25 (2017-09-11)

This issue is the 25th official release, and includes the following changes:
Remove the extension field from the request example as xsd definition in 2.5.4 listbatchtransfer.

Issue 24 (2017-07-21)

This issue is the 24th official release, and includes the following changes:
Add a new API in 2.6.4 refundrequest as per MTNSDP-DB-0322C.

Issue 23 (2017-06-20)

This issue is the 23th official release, and includes the following changes:
Update the old API in 2.2.1 login as per MTNSDP-DB-0322A

Issue 22 (2017-05-25)

This issue is the 22th official release, and includes the following changes:
Add a new API named channelOrchestrator in 2.4.7 channelOrchestrator as per MTNSDP-DB-0322B

Issue 21 (2017-01-19)

This issue is the 21th official release, and includes the following changes:
Add a new API named gettransactionstatus for 1.2 version in 2.4.3 gettransactionstatusV1.2 as per MTNSDP-CR-0322.

Issue 20 (2016-08-29)

This issue is the twentieth official release, and includes the following changes:
Update the error code in 2.7 API Error Response.

Issue 19 (2016-08-25)

This issue is the nineteenth official release, and includes the following changes:
Add a new API named getaccountholderinfo in 2.2.7 getaccountholderinfo as per MTNSDP-DB-0184A.

Issue 18 (2016-08-25)

This issue is the eighteenth official release, and includes the following changes:
Update the error code in 1.2.5 Error Codes for RequestPayment and 1.4.5 Error Codes for DepositMobileMoney.

Issue 17 (2016-03-03)

This issue is the seventeenth official release, and includes the following changes:
Add a new API named getaccountholderidentification as per MTNSDP-DB-0300.

Issue 16 (2016-02-29)

This issue is the sixteenth official release, and includes the following changes:
Add new parameter named CurrCode in 1.2.3 Request for RequestPayment as per MTNSDP-CR-0317.
Add new parameter named CurrCode in 1.3.3 Request for ConfirmThirdPartyPayment as per MTNSDP-CR-0317.
Add new parameter named CurrCode in 1.4.3 Request for DepositMobileMoney as per MTNSDP-CR-0317.



Change History

Add new parameter named CurrCode in 1.5.4 Response for QueryThirdPartyAccount as per MTNSDP-CR-0317.

Issue 15 (2015-12-28)

This issue is the fifteenth official release, and includes the following changes:

Add new parameter named profilename in 2.6.3 validatePayee response as per MTNSDP-CR-0231A.

Issue 14 (2015-12-18)

This issue is the fourteenth official release, and includes the following changes:

Modify Cookie definition in the MT HTTP Header for HTTP APIs.

Issue 13 (2015-11-20)

This issue is the thirteenth official release, and includes the following changes:

Password encryption algorithm supports SHA256.

Issue 12 (2015-09-29)

This issue is the twelfth official release, and includes the following changes:

Add the validatePayee API to 2.6.3 validatePayee.

Issue 11 (2015-06-18)

This issue is the eleventh official release, and includes the following changes:

Optimize the process for RequestPayment.

Add the following APIs to 2 HTTP APIs:

- cancelpreapproval
- getTransactionhistory
- queryBalance

Issue 10 (2015-04-30)

This issue is the tenth official release, and includes the following changes:

Optimize the process for preapproval, getPreApprovalStatus and preapprovalcompleted.

Issue 09 (2015-01-12)

This issue is the ninth official release, and includes the following changes:

Change the examples for the following APIs:

- Response example for validateaccountholder
 - Request and response examples for getfinancialresourceinformation
 - Request and response examples for verifyfinancialresourceinformation
 - Response example for preapproval
 - Request and response examples for getPreApprovalStatus
 - Request and response examples for preapprovalcompleted
 - Request and response examples for Paymentcompleted
 - Request and response examples for gettransactionstatus
 - Request and response examples for reversetransaction
 - Request and response examples for getreversetransactiondetails
-



Change History

- Request and response examples for withdraw
- Request and response examples for startbatchtransfer
- Request and response examples for approvebatchtransfer
- Request and response examples for cancelbatchtransfer
- Request and response examples for listbatchtransfer
- Request and response examples for getbatchtransfer

Issue 08 (2014-09-30)

This issue is used for eighth office app.

Add one error code **1000** in the RequestPayment for Phase3.3.

Add the description of destPartnerId.

Add the chapter of MO Message Header for A2A Scenario.

Change the **Level of Requirement** of the **SenderID** to **Optional**.

Delete the **CurrCode** field for requestPaymentRequest, ConfirmThirdPartyPaymentRequest and depositMobileMoneyRequest.

Issue 07 (2014-05-04)

This issue is the seventh official release, and includes the following changes:

Update the type of the **serviceld** field in the RequestPayment, DepositMobileMoney, ConfirmThirdpartyPayment, and QueryThirdPaytyAccount requests.

Change the description of **serviceld** and **bundleID**.

Issue 06 (2014-04-23)

This issue is the sixth official release, and includes the following changes:

Update the description and the example of serviceld field in the RequestPayment, DepositMobileMoney, ConfirmThirdpartyPayment, and QueryThirdPaytyAccount request.

Issue 05 (2014-04-09)

This issue is the fifth official release, and includes the following changes:

Update the description of StatusDesc field.

Issue 04 (2014-03-27)

This issue is the fourth official release, and includes the following changes:

Modify the "fee deduction" to "bill payment".

Modify the description of Narration field in DepositMoney interface.

Add a note in the description of Cookie field.

Issue 03 (2014-03-19)

This issue is the third official release, and includes the following changes for Phase3.1:

Add the MerchantName field in the ConfirmThirdPartyPayment request.

Issue 02 (2014-03-10)

This issue is the second official release, and includes the following changes for Phase2.3:

Update the process flow for Paymentcompleted API.



Change History

Issue 01 (2013-12-27)

This issue is the first official release.



Contents

1 SOAP APIs	1
1.1 Overviews	1
1.1.1 API Functions	1
1.1.2 Level of Requirement for Parameters	2
1.1.3 Request Format	2
1.1.4 Response Format	3
1.1.5 Namespaces	4
1.2 Bill Payment API initiated by Partner	5
1.2.1 Process	5
1.2.2 RequestPayment	7
1.2.3 Request for RequestPayment	7
1.2.4 Response for RequestPayment	14
1.2.5 Error Codes for RequestPayment	17
1.3 Bill Payment API initiated by Subscriber	34
1.3.1 Process	34
1.3.2 ConfirmThirdPartyPayment	35
1.3.3 Request for ConfirmThirdPartyPayment	35
1.3.4 Response for ConfirmThirdPartyPayment	41
1.3.5 Error Codes for ConfirmThirdPartyPayment	44
1.4 Deposit APIs	46
1.4.1 Process	46
1.4.2 DepositMobileMoney	46
1.4.3 Request for DepositMobileMoney	47
1.4.4 Response for DepositMobileMoney	55
1.4.5 Error Codes for DepositMobileMoney	59
1.5 APIs for Querying Subscriber Account Balances	76
1.5.1 Process	76
1.5.2 QueryThirdPartyAccount	77
1.5.3 Request for QueryThirdPartyAccount	77
1.5.4 Response for QueryThirdPartyAccount	80
1.5.5 Error Codes for QueryThirdPartyAccount	84
1.6 APIs for notifying requestPaymentCompleted	84
1.6.1 Process	84
1.6.2 requestPaymentCompleted	85
1.6.3 Request for requestPaymentCompleted	85
1.6.4 Response for requestPaymentCompleted	88
1.6.5 Error Codes for requestPaymentCompleted	89
1.7 API Error Responses	90
2 HTTP APIs	91
2.1 Overviews	91
2.1.1 MT Message Header	91
2.1.2 MO Message Header	96
2.1.3 MO Message Header for A2A Scenario	100



Contents

2.2 Account Validation & Information	102
2.2.1 login	102
2.2.2 Logout	104
2.2.3 validateaccountholder	106
2.2.4 getfinancialresourceinformation	108
2.2.5 verifyfinancialresourceinformation	111
2.2.6 getaccountholderidentification	114
2.2.7 getaccountholderinfo	116
2.3 Approval	119
2.3.1 preapproval	119
2.3.2 getPreApprovalStatus	122
2.3.3 preapprovalcompleted	125
2.3.4 cancelpreapproval	128
2.4 Transaction Details	130
2.4.1 Paymentcompleted	130
2.4.2 gettransactionstatus	134
2.4.3 gettransactionstatusV1.2	136
2.4.4 reversetransaction	139
2.4.5 getreversetransactiondetails	143
2.4.6 withdraw	146
2.4.7 channelOrchestrator	149
2.5 Batch Processing	151
2.5.1 startbatchtransfer	152
2.5.2 approvebatchtransfer	155
2.5.3 cancelbatchtransfer	157
2.5.4 listbatchtransfer	159
2.5.5 getbatchtransfer	165
2.6 Financial	168
2.6.1 getTransactionhistory	168
2.6.2 queryBalance	184
2.6.3 validatePayee	186
2.6.4 refundrequest	191
2.7 API Error Response	194
3 Appendix	222
3.1 The Currency and Atomic Charging Unit of each OpCo country	222
3.2 ISO 4217 Codes for the representation of currencies and funds	223
3.3 OpCoID List	224

01

SOAP APIs

1.1 Overviews

1.1.1 API Functions

The SDP provides Mobile Money capability application programming interfaces (APIs) for third-party applications (App for short) to connect to it and use its Mobile Money capability for. Bill Payment, Depositing Money in Subscribers account & Querying Subscriber's account in a third party system. The App is generally developed by various partners of the SDP.



NOTE

Partners are the enterprises and individuals who sign a contract and cooperate with carriers in utilizing the SDP. Partners include service Partners, Developers, and API Partners. In this document, partners are mainly the service Partners, Developers, and API Partners who use APIs for secondary development.

Table 1-1 describes functions of Mobile Money capability APIs provided by the SDP.

Table 1-1 Functions of Mobile Money capability APIs

Function	Description	API
Bill Payment API	Bill payment initiated by Partner: The App initiates the requestPayment API to debit mobile money from subscribers mobile money account. Ex : Utility Payments , Recurrent Payments etc.	1.2.2 RequestPayment
	Bill payment initiated by Subscriber: The subscriber through ECW initiates the ConfirmThirdPartyPayment API to pay their Bill on third party system. Ex : School Payment, Bill Payment etc	1.3.2 ConfirmThirdParty Payment
Deposit	The App initiates the depositMobileMoney API to deposit mobile money in Subscriber s mobile money account. Ex : Cash Deposit	1.4.2 DepositMobileMoney
QueryingSubscriberaccount balance	The SDP initiates the QueryThirdPartyAccount API to query subscriber account balance on third party system. Ex: check Outstanding Amount etc	1.5.2 QueryThirdPartyAccount



1 SOAP APIs

1.1.2 Level of Requirement for Parameters

The App must develop APIs based on the level of requirement for each parameter.

Table 1-2 Level of requirement for parameters

Type	Description
Mandatory	A parameter is always mandatory in a request. Parameters with the Mandatory requirement are used for access authentication or service processing. If a parameter with the Mandatory requirement is left empty in a request, access authentication or service processing fails and the request fails.
Conditional	A parameter is mandatory or optional in specified conditions. Parameters with the Conditional requirement are used for access authentication or service processing in specified conditions. If the specified conditions is met but a parameter with the Conditional requirement is left empty in a request, access authentication or service processing fails and the request fails.
Optional	A parameter is always optional. Parameters with the Optional requirement are not used for service processing.

1.1.3 Request Format

The SDP provides the SOAP request in the following format:

```
<soapenv:Envelope>
  <soapenv:Header>
    <parameter>...</parameter>
  ...
</soapenv:Header>
  <soapenv:Body>
    <parameter>...</parameter>
  ...
</soapenv:Body>
</soapenv:Envelope>
```



1 SOAP APIs

Table 1-3 Request format

Element	Description
<soapenv:Envelope>	Root element in a request, which specifies the namespace.
<soapenv:Header>	Request header. Parameters in this element are defined by the SDP and are mainly information to be processed by the SDP services, including access authentication parameters.
<soapenv:Body>	Request body. Parameters in this element comply with the SOAP protocol.

1.1.4 Response Format

Success Response Format

The SDP provides the SOAP API success responses in the following format:

```
<soapenv:Envelope>
  <soapenv:Body>
    <parameter>...</parameter>
  ...
</soapenv:Body>
</soapenv:Envelope>
```

Table 1-4 Success response format

Element	Description
<soapenv:Envelope>	Root element in a success response, which specifies the namespace.
<soapenv:Body>	Success response body. Parameters in this element comply with the SOAP protocol.

Error Response Format

The SDP provides the SOAP API error responses in the following format:

```
<soapenv:Envelope>
  <soapenv:Body>
    <soapenv:Fault>
      <parameter>...</parameter>
    ...
    <detail>
      <parameter>...</parameter>
```



1 SOAP APIs

```
...
    </detail>
  </soapenv:Fault>
</soapenv:Body>
</soapenv:Envelope>
```

Table 1-5 Error response format

Element	Description
<soapenv:Envelope>	Root element in an error response, which specifies the namespace.
<soapenv:Body>	Error response body, contains the <soapenv:Fault> and <detail> elements. This element specifies the error code and error details.
<soapenv:Fault>	Error code and description. For details about error responses, see 1.7 API Error Responses.
<detail>	Error details, which are the same as the <soapenv:Fault> element information.

1.1.5 Namespaces

When coding API functions, partners must follow the namespaces of Mobile Money capability APIs and data types in the APIs.

- Table 1-6 describes namespaces of Mobile Money capability APIs.
- The namespace of data types in Mobile Money capability APIs is <http://www.w3.org/2001/XMLSchema>.



NOTE

The Mobile Money capability APIs involve the following data types: Parameter[0..20], and Return[0..20]. For details about the structure of the data types, see the parameter description for the matching API requests or responses.

Table 1-6 Namespaces of Mobile Money capability APIs

Namespace	API
http://schemas.xmlsoap.org/soap/encoding/	• 1.2.2 RequestPayment
http://schemas.xmlsoap.org/wsdl/	• 1.3.2 ConfirmThirdPartyPayment
http://schemas.xmlsoap.org/wsdl/soap/	• 1.4.2 DepositMobileMoney
http://b2b.mobilemoney.mtn.zm_v1.0/	• 1.5.2 QueryThirdPartyAccount



1 SOAP APIs

1.2 Bill Payment API initiated by Partner

1.2.1 Process

Figure 1-1 shows the synchronous process of bill payment for ECW.

Figure 1-1 Synchronous Process of bill payment for ECW

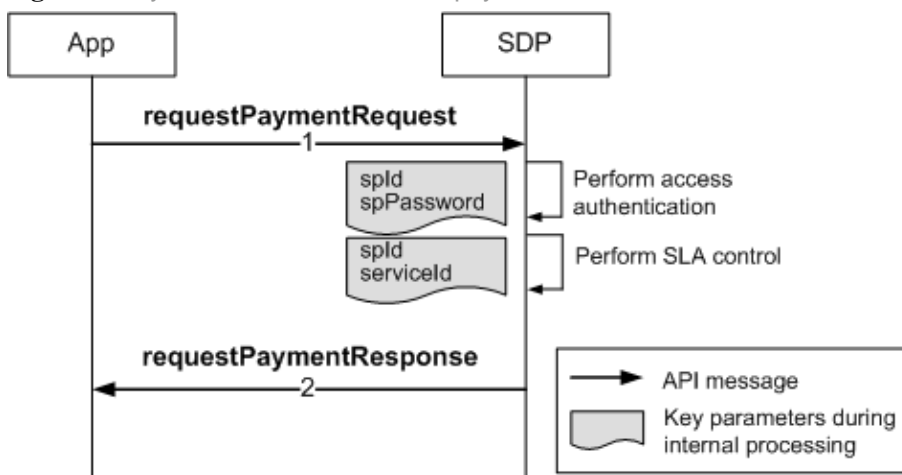


Table 1-7 describes the synchronous process for ECW.

Table 1-7 Description for the synchronous process of bill payment for ECW

Step	Description
1	The App sends bill payment request to debit mobile money from subscribers account to SDP.
2	<ul style="list-style-type: none"> The SDP performs authentication and service level agreement (SLA) control based on fields in the request. The SDP sends response to the App.

Figure 1-2 shows the asynchronous process of bill payment for ECW.



1 SOAP APIs

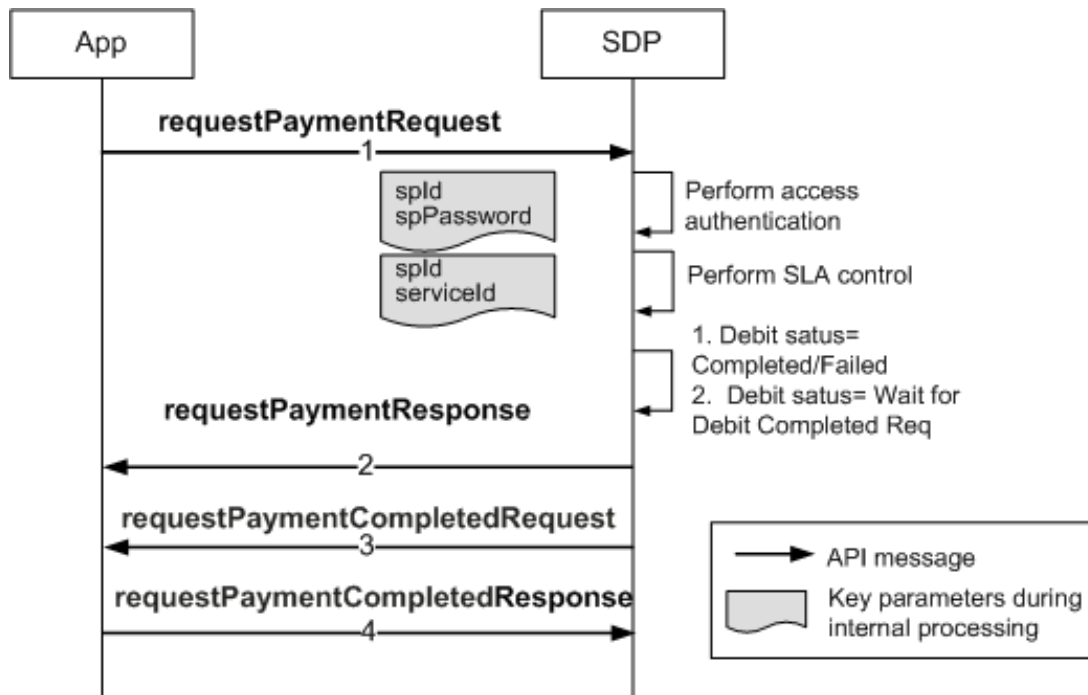
Figure 1-2 Asynchronous Process of bill payment for ECW

Table 1-8 describes the asynchronous process for ECW.

Table 1-8 Description for the asynchronous process of bill payment for ECW

Step	Description
1	The App sends bill payment request to debit mobile money from subscribers account to SDP.
2	<ul style="list-style-type: none"> The SDP performs authentication and service level agreement (SLA) control based on fields in the request. The SDP sends response to the App.
3	The SDP sends requestPaymentCompleted request to inform the App the result of the requestPayment in case asynchronism requestPayment.
4	The APP processes the requestPaymentCompleted request and sends a response to the SDP.



1 SOAP APIs

1.2.2 RequestPayment*Function*

The App (functioning as the client) invokes the requestPayment API to debit mobile money from subscribers account.

Partners must code the App based on the API field requirements so that the App can send correct requests to the SDP. The SDP must send a response within 180 seconds by default in synchronous process, or 30 seconds by default in asynchronous process.

Request URI

The request URI is the destination URI of requestPayment messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:PortThirdPartyServiceUMMImpl/UMMSERVICE/RequestPayment/v17

In the format, **IP** and **Port** indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

1.2.3 Request for RequestPayment

The App functions as the client and sends a requestPaymentRequest message to the SDP for bill payment.

Example

```
POST /ThirdPartyServiceUMMImpl/UMMSERVICE/RequestPayment/v17 HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Signature:43AD232FD45FF
Cookie: sessionId=default8fcee064690b45faa9f8f6c7e21c5e5a

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:b2b="http://b2b.mobilemoney.mtn.zm_v1.0">
  <soapenv:Header>
    <RequestSOAPHeader xmlns="http://www.huawei.com.cn/schema/common/v2_1">
      <spId>35000001</spId>
      <spPassword>de96d901b3bad1db2aab76b7b0b202f2</spPassword>
      <bundleID>256000039</bundleID>
      <serviceId>35000001000035</serviceId>
      <timeStamp>20100727</timeStamp>
    </RequestSOAPHeader>
  </soapenv:Header>
  <soapenv:Body>
    <b2b:processRequest>
      <serviceId>200</serviceId>
      <parameter>
```



1 SOAP APIs

```

        <name>DueAmount</name>
        <value>10</value>
    </parameter>
    <parameter>
        <name>MSISDNNum</name>
        <value> FRI:< MSISDNNum >/MSISDN </value>
    </parameter>
    <parameter>
        <name>ProcessingNumber</name>
        <value>555</value>
    </parameter>
    <parameter>
        <name>serviceId</name>
        <value>101</value>
    </parameter>
    <parameter>
        <name>AcctRef</name>
        <value> FRI:< AcctRef > @water.company/SP </value>
    </parameter>
    <parameter>
        <name>AcctBalance</name>
        <value>555</value>
    </parameter>
    <parameter>
        <name>MinDueAmount</name>
        <value>121212</value>
    </parameter>
    <parameter>
        <name>Narration</name>
        <value>121212</value>
    </parameter>
    <parameter>
        <name>PrefLang</name>
        <value>121212121</value>
    </parameter>
    <parameter>
        <name>OpCoID</name>
        <value>0</value>
    </parameter>
    <parameter>
        <name>CurrCode</name>
        <value>USD</value>
    </parameter>
</b2b:processRequest>
</soapenv:Body>
</soapenv:Envelope>

```



1 SOAP APIs

Message HTTP Header Parameters

Table 1-9 Describes parameters in a requestPaymentRequest message HTTP header.

Table 1-9 Parameters in a requestPaymentRequest message HTTP header

Header Field	Parameter	Type	Length	Level of Requirement	Description
Cookie	sessionid	String		Conditional	<p>The session id is used as a cookie in the HTTP header in the requests towards the Mobile Money Manager when the Partner use Login/Logout mode.</p> <p>[Example] default8fcee064690b45faa9f8f6c7e21c5e5a</p> <p>NOTE Cookie is valid only for ECW and not for fundamo.</p>
Signature/		String		Conditional	<p>Signature.</p> <p>The field indicates the summary when the Partner uses HMAC security mode.</p> <p>[Example] 43AD232FD45FF</p>

Message SOAP Header Parameters

Table 1-10 describes parameters in a requestPaymentRequest message SOAP header.

Table 1-10 Parameters in a requestPaymentRequest message SOAP header

Parameter	Type	Length	Level of Requirement	Description
spId	xsd: string	21	Mandatory	<p>Partner ID.</p> <p>The ID is automatically allocated by the SDP to partners after successful registration. To obtain the ID:</p> <ul style="list-style-type: none"> A service Partner and API Partner can log in to the SDP management portal and query account information. A Developer can log in to the Developer Portal and query account information. <p>[Example] 000201</p>



1 SOAP APIs

Parameter	Type	Length	Level of Requirement	Description
spPassword	xsd: string	100	Conditional	<p>Authentication key for the SDP to authenticate partners.</p> <p>The SDP supports authentication by SP ID + Password, SP ID + IP address + Password, or SP ID + IP address. Partners select an authentication mode during registration. If a partner selects authentication by SP ID + Password or SP ID + IP address + Password, this parameter is mandatory in requests sent by this partner.</p> <p>The value is a character string encrypted. The encryption formula is as follows:</p> <ul style="list-style-type: none"> • SHA-256: spPassword = Base64(SHA-256(spId + Password + timeStamp)) • MD5: spPassword = MD5(spId + Password + timeStamp) <p>In the formula:</p> <ul style="list-style-type: none"> • spId and timeStamp: authentication ID and timestamp. • Password: password for partners to access the SDP. <ul style="list-style-type: none"> – A service Partner and API Partner can obtain the password from the email notification received after successful registration. – A Developer can log in to the Developer Portal, choose Member Center > Account > Registration Information > Invoke Password, and set the password. <p> NOTE To retain features of earlier versions, the SP uses the MD5 algorithm in the connection to the SDP, which might cause security risks.</p> <p>[Example] e6434ef249df55c7a21a0b45758a39bb</p>



1 SOAP APIs

Parameter	Type	Length	Level of Requirement	Description
bundleID	xsd: string	21	Conditional	<p>Bundle ID.</p> <p>When SDP creates a capability bundle, SDP allocates a bundleID to capability bundle.</p> <p>The bundleID must not be contained during invocation of a service interface developed by service partners and other partners, and must be contained during invocation of a capability interface developed by API partners, other partners, and developers.</p> <p>[Example] 256000039</p>
serviceld	xsd: string	21	Conditional	<p>Service ID.</p> <p>The ID is automatically allocated by the SDP to services after successful release. A service Partner can log in to the SDP management portal and query service information for the ID.</p> <p>The serviceld must be contained during invocation of a service interface developed by service partners and other partners, and must not be contained during invocation of a capability interface developed by API partners, other partners, and developers.</p> <p>[Example] 35000001000001</p>
timestamp	xsd: string	14	Conditional	<p>Timestamp (UTC time).</p> <p>The value is used in MD5 encryption of spPassword. This parameter is mandatory when the spPassword parameter is required.</p> <p>[Format] yyyyMMddHHmmss</p> <p>[Example] 20100731064245</p>



1 SOAP APIs

Message Body Parameters

Table 1-11 describes parameters in a requestPaymentRequest message body.

Table 1-11 Parameters in a requestPaymentRequest message body

Parameter	Type	Length	Level of Requirement	Description
serviceld	xsd: string	4	Mandatory	Holds the identification number to identify the service request. This is set as 200.
parameter	Parameter[0..20]		Optional	<p>Holds the request parameter details in the form of name/value pair to identify the service details.</p> <p>Valid parameter names are:</p> <ul style="list-style-type: none"> • DueAmount • MSISDNNum • ProcessingNumber • serviceld • AcctRef • AcctBalance • MinDueAmount • Narration • PrefLang • OpCoID • CurrCode <p>For the detailed parameter name information, see Table 1-13.</p> <p>The parameter parameter is of the Parameter[0..20] type and contains multiple sub-parameters. For details about the Parameter[0..20] type, see Table 1-12.</p>

Table 1-12 describes the parameter structure of the Parameter[0..20] type.

Table 1-12 Parameter structure of the Parameter[0..20] type

Parameter	Type	Length	Level of Requirement	Description
name	xsd:string	-	Mandatory	Holds the parameter name.
value	xsd:string	-	Mandatory	Holds the parameter value.



1 SOAP APIs

Table 1-13 describes the parameter name lists.

Table 1-13 Parameter name lists

name	value			
Parameter	Type	Length	Level of Requirement	Description
DueAmount	decimal	4	Mandatory	The amount that the subscriber is due to pay on the App system.
MSISDNNum	xsd:string	30	Mandatory	<p>An Id to uniquely identify the subscriber on the App system.</p> <p>Subscriber's MSISDN number or Fake ID.</p> <p>[Format]</p> <ul style="list-style-type: none"> • Mobile number: [Country code][Mobile number] • Fake ID: [Prefix]- [opcoid]-[sequence]. <p>[Example]</p> <ul style="list-style-type: none"> • Mobile number: 8612312345678 • Fake ID: f-245-11900000007639 <p>Partner needs to convert MSISDN to FRI format before sending it as per below rules:</p> <p>FRI:< MSISDNNum >/MSISDN</p>
ProcessingNumber	xsd:string	140	Mandatory	An Id to uniquely identify the transaction on the App system.
serviceld	xsd:string	140	Optional	This relates to the Code configured in the Mobile Money system. Properties file is used to uniquely identify each service provided.
AcctRef	xsd:string	140	Mandatory	<p>Subscriber's account number on the App system.</p> <p>If the subscriber does not have account on the App system, you need confirm it with the Mobile Money system.</p> <p>The Partner should send the request in format like:</p> <p>FRI:< AcctRef > @water.company/SP</p>
AcctBalance	xsd:int	4	Optional	<p>Subscriber's balance on the App system.</p> <p>If there is no subscriber's balance on the App system, you need confirm it with the Mobile Money system.</p>



1 SOAP APIs

name	value			
Parameter	Type	Length	Level of Requirement	Description
MinDueAmount	xsd:int	4	Optional	The minimum amount that the subscriber is due to pay on the App system. If there is no minimum due on App system, you need confirm it with the Mobile Money system.
Narration	xsd:string	140	Optional	Any text that the vendor would like to display to the subscriber.
PrefLang	xsd:string	140	Optional	The language locale to be used.
OpCoID	xsd:string	140	Optional	Identifier of the country local to be used. Please refer to 3.3 OpCoID List.
CurrCode	xsd:string	3	Optional	The currency, as an ISO 4217 formatted string.

1.2.4 Response for RequestPayment

The SDP functions as the server, processes requestPaymentRequest messages received from the App, and sends requestPaymentResponse messages to the App.

This topic provides a success response example. If a request fails, the SDP sends an error response that contains an error code. For details about error responses, see 1.7 API Error Responses.

Example

HTTP/1.1 200 OK
 Signature:43AD232FD45FF
 Content-Length: 0
 Server: Jetty(7.5.4.v20111024)

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <ns1:processRequestResponse xmlns:ns1="http://b2b.mobilemoney.mtn.zm_v1.0">
      <return>
        <name>ProcessingNumber</name>
        <value>555</value>
      </return>
      <return>
        <name>ThirdPartyAcctRef</name>
        <value>444</value>
      </return>
      <return>
        <name>senderID</name>
```



1 SOAP APIs

```

        <value>MOM</value>
      </return>
    </return>
    <name>StatusCode</name>
    <value>222</value>
  </return>
  <return>
    <name>StatusDesc</name>
    <value>PENDING</value>
  </return>
  <return>
    <name>MOMTransactionID</name>
    <value>111</value>
  </return>
</ns1:processRequestResponse>
</soapenv:Body>
</soapenv:Envelope>

```

Message HTTP Header Parameters

Table 1-14 Describes parameters in a requestPaymentResponse message HTTP header.

Table 1-14 Parameters in a requestPaymentResponse message HTTP header

Header Field	Parameter	Type	Length	Level of Requirement	Description
Signature/		String		Conditional	Signature. The field indicates the summary when the Partner uses HMAC security mode. [Example] 43AD232FD45FF

Message Body Parameters

Table 1-15 describes parameters in a requestPaymentResponse message body.



1 SOAP APIs

Table 1-15 Parameters in a requestPaymentResponse message body

Parameter	Type	Length	Level of Requirement	Description
return	Return[0..20]	-	Mandatory	<p>Holds the response parameter details in the form of name/value pair to identify the service details.</p> <p>Valid parameter names are:</p> <ul style="list-style-type: none"> ProcessingNumber ThirdPartyAcctRef SenderID StatusCode StatusDesc MOMTransactionID <p>For the detailed parameter name information, see Table 1-17.</p> <p>The return parameter is of the Return[0..20] type and contains multiple sub-parameters. For details about the Return[0..20] type, see Table 1-16.</p>

Table 1-16 describes the parameter structure of the Return[0..20] type.

Table 1-16 Parameter structure of the Return[0..20] type

Parameter	Type	Length	Level of Requirement	Description
name	xsd:string	-	Mandatory	Holds the parameter name.
value	xsd:string	-	Mandatory	Holds the parameter value.

Table 1-17 describes the return name lists.

Table 1-17 Return name lists

name	value			
Parameter	Type	Length	Level of Requirement	Description
ProcessingNumber	xsd:string	140	Mandatory	An Id to uniquely identify the transaction on the App system.
ThirdPartyAcctRef	xsd:string	140	Optional	<p>Subscriber's account number on the App system.</p> <p>The SDP will fill this parameter as the same value which was received from AcctRef in the request.</p>



1 SOAP APIs

name	value			
Parameter	Type	Length	Level of Requirement	Description
SenderId	xsd:string	140	Optional	An Id to identify sender. This is set as MOM.
StatusCode	xsd:int	2	Mandatory	A predetermined code to determine the status of the transaction.
StatusDesc	xsd:string	140	Optional	The Mobile Money system supplied description of the response code. The value are as follow: <ul style="list-style-type: none"> • SUCCESSFUL • PENDING • FAILED
MOMTransactionID	xsd:string	140	Optional	An Id to uniquely identify the transaction on the Mobile Money system. The SDP will forward this parameter to the App system.

1.2.5 Error Codes for RequestPayment

Table 1-18 describes requestPayment error codes that the SDP may return upon an exception. For details about the error codes, see the *SDP Solution Error Code Reference*.

Table 1-18 requestPayment error codes

Error Code	Description
01	Successfully processed transaction.
02	Invalid account details.
03	Transaction not allowed.
04	Payment below minimum.
05	Invalid Fundamo userid/password.
06	Payment Entry not found.
07	Missing transaction/message ID.
08	Missing application version.
100	General failure.



1 SOAP APIs

Error Code	Description
	Get addr route info failed!the baseScfType = %1,the sourceAddr = %2,the destAddr = %3. NOTE ScfType refers to Service Capability Function Type.
	MDSP AuthPrice failed, result code is %1. NOTE The MDSP is composed of the Mobile Data Control Center (MDCC) subsystem and peripheral functional subsystems such as the Portal and Mobile Data Management Center (MDMC). The MDCC subsystem consists of core modules such as the Short Message Push Agent (SMPA), Network Management System (NMS) proxy, and BOSS interface machine. The MDSP is connected to the NMS, BOSS system, Service Control Point (SCP), customer service center, and service gateways. The MDSP can implement the functions such as external entity access, mobile data management, user and service authentication, and charging.
	Waiting for response timed out, message type is %1.
	Cannot find route policy, the policy identifier is %1.
	The service status is unknown.
	Service %1 is in blacklist!
	The API %1 is not existed.
	Service ID %1 is not exist!
	The service status is suspended.
	The service status is pre-deregistered.
	The service status is deregistered.
	The API status is disabled.
	SP status is locked.
	The sp's Status is forbidden.
	The sp's status is pause.
	The RemotePassword of NE %1 is empty! NOTE NE refers to network element, such as SMSC, MMSC.
	NEInfo id %1 not in SAG. NOTE The SAG provides open interfaces for third-party partners. It authenticates and authorizes the partners, and controls the Service Level Agreement policies when it receives third-party service requests.



1 SOAP APIs

Error Code	Description
	The ScfType %1 is inactive!
	NOTE ScfType refers to Service Capability Function Type.
	The ScfType %1 is uninstalled!
	NOTE ScfType refers to Service Capability Function Type.
	The service status is configuring.
	The Created is null.
	SP ip is null!
	Sp password is null!
	The Nonce is null.
	Authentication Failed, cause by SP, because of timestamp expired.
	Service ID is null!
	Sp password is null!
	Timestamp is empty in soapheader.
	Service ID %1 is invalid!
	The PasswordDigest is not accepted.
	The sp's Status is pre-deregistered.
	The sp's Status is deregistered.
	The sp's Status is unknown.
	SP %1 is in blacklist!
	SPID %1 is not exist!
	Sp password is not accepted!
	Sp ip %1 is not accepted!
	local SP password is null!
	The authentication type is unknown!
	Authentication Failed, cause by SP, because of timestamp is not valid.
	Minimum Amount per Transaction



1 SOAP APIs

Error Code	Description
	%1 is null.
101	Field length exceeded.
102	Subscriber could not be identified. (Invalid subscriberID).
	%1 is null.
103	Duplicate transaction id.
104	Authorisation declined.
105	Account holder is not active.
106	Deposit Transfer not processed: Source & destination accounts are same
108	Deposit Transfer not processed: Insufficient funds
109	Deposit Transfer not processed: Daily limit has been exceeded.
110	Deposit Transfer not processed: Sender account not active
111	Deposit Transfer not processed: Target account not active
501	DATEFORMAT_INVALID
502	FRI_INVALID
503	IDENTITY_INVALID
504	INVALID_FIELD_VALUE
505	MANDATORY_FIELD_MISSING
506	OTHER_FINANCIAL_ERROR
507	PREAPPROVAL_NOT_FOUND
508	REFERENCE_ID_NOT_FOUND
509	REQUIRED_FIELD_MISSING
510	SERVICE_TEMPORARILY_UNAVAILABLE
511	SESSION_INVALID
512	TRANSACTION_ID_OR_ORIGINAL_TRANSACTION_ID_MISSING
513	TRANSACTION_NOT_COMPLETED
514	TRANSFER_TYPE_AND_ACCOUNT_DO_NOT_MATCH
515	ACCOUNTHOLDER_WITH_FRI_NOT_FOUND



1 SOAP APIs

Error Code	Description
516	ACCOUNTHOLDER_WITH_MSISDN_NOT_FOUND
517	AUTHORIZATION_INVALID_ACCOUNT_TYPE_COMBINATION
518	COULD_NOT_PERFORM_OPERATION
519	EXPIRED_OR_INVALID_TRANSACTION_ID
520	INCORRECT_PASSWORD
521	NOT_AUTHORIZED
522	AMOUNT_INVALID
523	CURRENCY_NOT_SUPPORTED
524	INVALID_CURRENCY
525	RESOURCE_NOT_ACTIVE
526	SOURCE_NOT_FOUND
527	RESOURCE_NOT_FOUND
529	TARGET_AUTHORIZATION_ERROR
530	This account is already a member of this group
531	Account could not be removed from Bank domain
532	The specified account holder's additional information name already exist
533	The specified account holder's additional information name cannot be deleted because it is in use
534	This account holder is already a member of this group
535	AccountHolder does not exist
536	This operation requires the user to be logged in
537	AccountHolder profile does not have requested account
538	AccountHolder status should be blocked or active
539	Account not found
540	Could not create account route since Account route with provided account route reference and currency already exists
541	Could not delete account route since Account route with provided account route reference and currency does not exists



1 SOAP APIs

Error Code	Description
542	Account type is not supported
543	The account holder already has a connection to the specified account
544	Activation of account holder failed
545	Could not perform operation since at least one of the mandatory field is required
546	Could not perform operation since at least one of the mandatory field is required
547	Could not perform operation since at least one of the mandatory field is required
548	Could not perform operation since at least one of the mandatory field is required
549	Could not perform operation since at least one of the mandatory field is required
550	Account holder could not be found
551	Operation requires account holder to have an MSISDN identity but it has none
552	Account holder is inactive
553	A specified account holder was not a child to another specified account holder
554	Account holder could not be found
555	The settings for this account holder already exists
556	Account holder settings are mandatory
557	Account holder settings not found
558	Account holder could not be registered because an account holder with submitted Alias already exists
559	Account holder could not be registered because an account holder with submitted E-mail already exists
560	Account holder with given e-mail could not be found
561	Account holder cannot have the same employee id as another employee working under the same employer
562	Account holder could not be registered because an account holder with submitted External ID already exists
563	Account holder IMSI could not be updated because it already exists
564	Account holder could not be registered because an account holder with submitted MSISDN already exists



1 SOAP APIs

Error Code	Description
565	Account holder could not be registered because an account holder with submitted Username already exists
566	Adjustments from account not allowed
567	Adjustments not allowed on account
568	Adjustments to account not allowed
569	No Alias with this name was found in the system
570	Could not perform operation since the resource already belongs to another Bank domain
571	The FRI refers to more than one account
572	Account holder is not allowed to perform this operation
573	Account holder is already activated
574	Insufficient amount
575	Setting an account holder as its own parent or a creating a cycle in the hierarchy is not allowed
576	Authorization failed
577	Maximum transfer account throughput amount including fee threshold exceeded
578	Maximum transfer account throughput amount including fee threshold exceeded for receiver
579	Maximum transfer account throughput amount including fee threshold exceeded for sender
580	Maximum transfer amount threshold exceeded
581	Maximum transfer amount including fee threshold exceeded
582	Maximum transfer amount including fee threshold exceeded for receiver
583	Maximum transfer amount including fee threshold exceeded for sender
584	Maximum transfer amount threshold exceeded for receiver
585	Maximum transfer amount threshold exceeded for sender
586	Maximum transfer counter threshold exceeded
587	Maximum send amount threshold exceeded
588	Authorization failed. Maximum threshold limit reached for Approver
589	Amount below minimum amount threshold
590	Amount below minimum amount threshold for account



1 SOAP APIs

Error Code	Description
591	Amount below minimum amount threshold for account reference
592	Account holder is not registered
593	Receivers account does not allow deposit
594	Could not perform operation because amount is below minimum first deposit amount threshold
595	Maximum balance threshold for receiver exceeded
596	Receivers account is not active
597	Receiving account is unavailable
598	Withdrawal from recipient account not allowed
599	Minimum balance threshold for sender exceeded
600	Sending account is unavailable
601	A bank account with given bank code and account number could not be found
602	A route to the wanted bank account could not be found
603	No bank domain set in the system.
604	Bank domain with given ID was not found
605	Could not perform operation since the Bank domain is currently in use
606	Could not perform operation since Bank domain with given name already exists
607	Account could not be blocked
608	Receiving account holder has insufficient permission to receive cashin
609	Receiving account holder has insufficient permission to receive cashout
610	Certificate already updated for the user
611	Certificate credentials already exist
612	Certificate credentials do not exist
613	Account could not be closed
614	The company information already exist
615	The company information does not exist
616	Contact was not added since it already exists
617	Contact with given FRI could not be found



1 SOAP APIs

Error Code	Description
618	Could not create account holder
619	Could not perform add account operation
620	Failed to approve the operation, please retry the initial operation again
621	ATM partial reversal operation could not be performed
622	ATM redeem voucher operation could not be performed
623	ATM rollback voucher operation could not be performed
624	Could not perform the operation, the account has to be linked with sp account.
625	Receive remittance failed
626	Could not perform update of account holder settings
627	Could not perform update of pending payment status
628	The specified counter configuration does not exist.
629	The credential is not in the expected suspended state
630	Operation not allowed because credential not in active state
631	Repeated secrets does not match each other
632	Given credential type does not exist in the system
633	For this operation currency conversion is not allowed
634	Custody Account already exists
635	Custody account certificate already exist
636	Custody account certificate could not deleted
637	Custody account certificate could not be empty
638	Custody account certificate could not be used for non-repudiation
639	Custody account certificate does not exist
640	Custody account certificate error
641	Unsupported algorithm for custody account certificate
642	Custody account certificate with same id already exist
643	Name and Bank Domain are same as before



1 SOAP APIs

Error Code	Description
644	Could not find CustodyAccount since Custody Account with provided account number and bank code does not exists
645	Custody account still contains money
646	This Custody account still have pending transaction
647	Could not update custody account as name and bank domain both are missing
648	Account holder and account must belong to the same bank domain
649	The document does not exist
650	The document rules has been incorrectly configured in the system
651	This document URL is invalid
652	This document URL is not allowed
653	End date can not occur in the future
654	This error originates from an external system
655	External service provider with given User name could not be found
656	Netting report error
657	Financial Configuration does not exist
658	Neither transaction with transaction ID nor original transaction with original transaction ID could be found
659	Using first name in password is not allowed
660	The specified float transfer exemption rule already exists
661	Setting multiple owners of an FRI is not allowed
662	FRI Reference not found.
663	Operation requires that start date has to occur before end date
664	Could not perform operation since the Group is already used by another Bank domain
665	Group member could not be removed from group
666	Group member could not be removed from Bank domain
667	Group member with given ID could not be found in this group
668	Group used by Bank domain is not allowed to be removed
669	It's not possible to mirror a High Contention account in SDP since it doesn't have a balance.



1 SOAP APIs

Error Code	Description
670	Home charging region with given ID was not found
671	Could not perform operation since the home charging region is currently in use
672	Could not perform operation since Home Charging Region with given name already exists
673	Length of identification is not within length boundaries
674	Duplicate identification
675	ID type not supported by ISO 20022
676	The credential status change is illegal
677	Format of IMSI is invalid
678	Inactive receiver account
679	Incorrect pin code
680	Security answer was not correct
681	Transaction state is incorrect for this operation
682	An internal error caused the operation to fail
683	Could not find Internal Loan Account with given Id
684	Amount Min cannot be larger than Amount Max
685	The state of this Internal Loan Application is not valid for this operation
686	Could not find Internal Loan Application
687	Maximum number of Internal Loan Applications reached
688	Internal Loan Product already exist
689	The Internal Loan Product status is not active
690	The given Internal Loan Product is not allowed to be used
691	Could not find Internal Loan Product with given identity and product code
692	Internal Loan Provider already exist
693	Could not find Internal Loan Provider with given identity
694	The FRI must be of account type
695	The specified account holder's additional information name is not valid or does not exist
696	Account holder settings are invalid



1 SOAP APIs

Error Code	Description
697	Approval transaction has invalid state
698	Invalid ATM operation
699	Certificate is invalid
700	Invalid Delivery method
701	Profile validation
702	Profile counter validation
703	Profile name validation
704	Profile threshold validation
705	Invalid receiving end user
706	Incorrect remittance status
707	A transaction can only be scheduled from a future date
708	Invalid scheduled transaction status for the requested operation
709	Settlement deposit with given ID could not be found
710	Settlement with given ID could not be found
711	The provided signature is not correct
712	Current status is not as expected
713	Invalid Transaction type
714	Invalid User (or financial) approval type or not registered
715	Bank domain could not be removed since it is the last remaining
716	Limited retries are not allowed for the specified service provider
717	Could not find Loan Application with given approval ID
718	Could not find Loan Provider with given name
719	Reached limit for max accounts
720	Maximum number of contacts threshold exceeded
721	Maximum number of documents exceeded
722	Actor MSISDN is missing
723	Exchange rate for conversion between currencies does not exist.



1 SOAP APIs

Error Code	Description
724	More than one of a set of optional but mutual parameters were set.
725	Given msisdn already used by a point of sale
726	User has no access permission
727	Using non ASCII characters in password is not allowed
728	Non-digit characters in pin code
729	It is not allowed to change bank domain for an account holder
730	One custody account must exist
731	Only one refund per debit is allowed
732	Operation not supported on external account
733	Operation not supported on a migrated account
734	Request did not contain any OTP
735	Request did not contain any OTP
736	Request did not contain any OTP
737	Format of OTP is invalid
738	OTP not found, has expired or is used for the wrong operation
739	Password has already been used before
740	Password too short
741	Could not find Pending Commission
742	Pickup with given MTCN could not be found
743	PIN code has already been used
744	PIN code denied by matching birth date
745	PIN code denied by matching identification number
746	PIN code denied by matching MSISDN
747	PIN code denied by matching regular expression
748	Wrong profile
749	Pin code too long
750	Pin code too short



1 SOAP APIs

Error Code	Description
751	The point of sale with given msisdh is not acquired.
752	No pre-approvals found
753	Profile name validation
754	Json File validation
755	A profile mismatch between for example a provided counter configuration (belonging to one profile) and provided profile reference (having a different profile)
756	Profile not found
757	Profile type validation
758	Profile version validation
759	Profile version validation
760	Tag was not found
761	Provider Category can not be deleted as it is linked with account holder
762	No provider category with this name was found in the system
763	Request is queued for approval
764	Quote has already been used
765	Quote has expired
766	Quote not found
767	Quote is not supported with the combination of parameters
768	Quote is already processing
769	Account could not be re-activated
770	Failed to process reconciliation file.
771	Recurring scheduled transaction can only be scheduled a limited number of days into the future
772	Refund amount cannot be larger than the debit amount
773	The relative does not exist
774	Remittance Id does not exist in dbase
775	Remittance not found, Invalid provider name or remittance id
776	Remittance not possible because given data is invalid.



1 SOAP APIs

Error Code	Description
777	Password criteria configuration invalid
778	FRI not available
779	Required resource is temporarily locked
780	Reversal has already been performed on this transaction
781	Reversal is not allowed on this type of transaction
782	Reversal not possible since sub transaction is invalid
783	Using reversed user id in password is not allowed
784	Transaction could not be performed since neither of the parties supports rollback
785	Transaction can only be scheduled a limited number of days into the future
786	The requested scheduled transaction could not be found
787	This self service notification already exists.
788	The notification limit exceeds the hard limit
789	Self service notificaiton not found.
790	Max number of self serviced notifications reached, cannot create another.
791	The response amount does not match with the amount in the Payment Instruction
792	Another settlement task is ongoing.
793	Balance reconciliation failed. Provided closing balance does not match with the expected closing balance of custody account
794	Certificate does not exist for custody account not able to verify signature of settlement file
795	Custody account already exists with settlement configuration
796	Could not find settlement Configuration for custody account
797	Settlement configuration for identifier already exists
798	Instruction formatter class not found
799	Settlement configuration not found
800	Settlement parser class not found
801	Unable to remove custody account id from settlement configuration
802	Currency does not match with custody account's currency.



1 SOAP APIs

Error Code	Description
803	Settlement process failed because the custody accounts information do not match.
804	The deposit has already been reversed.
805	The deposit is in an invalid state.
806	The deposit was not found.
807	The transaction details of the deposit where not found.
808	Duplicated deposit or reverse deposit record found.
809	Digest does not exist or digest verification failed for settlement file
810	Error in parsing settlement file
811	Signature does not exist or signature verification failed for settlement file
812	Payment Instruction Response caller identifier mismatch
813	Settlement identifier not found
814	Invalid or no amount in Payment Instruction Response.
815	Settlement error due to invalid receiver. There is no such account holder or custody account exists.
816	Settlement is in wrong state
817	Online settlement not enabled for custody account
818	Opening Balance reconciliation failed
819	Wrong state for Settlement PaymentInstruction.
820	Settlement PaymentInstruction not found.
821	Failed to process settlement file.
822	The receiver account does not match
823	Wrong state for Settlement CustodyAccountsTransfers.
824	Using surname in password is not allowed
825	Not possible to put account in suspended state
826	Tag could not be created because a Tag with submitted name already exists
827	Tag was not found
828	Tax report not found



1 SOAP APIs

Error Code	Description
829	The specified threshold configuration does not exist.
830	Too few lower case alphabetic characters in password
831	Too few digits in password
832	Too few special characters in password
833	Too few upper case alphabetic characters in password
834	Too few upper or lower case alphabetic characters in password
835	Too many consecutive digits in PIN code
836	Too many identical consecutive characters in password
837	Too many repeated digits in PIN code
838	Transaction not found
839	Transaction timed out
840	Operation is not supported for this account
841	Account holder could not be created
842	Interest tier could not be created
843	Interest tier could not be modified
844	Interest tier could not be removed
845	Account could not be un-blocked
846	Unresolved Remittance Id does not exist in dbase
847	The maximum attempts of credential unsuspend for the given account holder has been reached
848	Using user id in password is not allowed
849	The data provided in the request failed the validation check
850	Voucher date of birth has been set
851	Voucher has expired
852	Voucher external id has been set
853	Voucher first name has been set
854	Voucher is no longer active



1 SOAP APIs

Error Code	Description
855	Voucher with given identity could not be found in the system
856	Voucher secret contains letters that has been excluded through system configuration
857	Voucher secret contains characters that is not legal according to system configuration
858	Voucher secret contains non ASCII characters
859	Voucher secret can not contain digits
860	Voucher secret is not correct
861	Voucher secret element is empty
862	Voucher secret can not contain letters
863	The voucher secret rules has been incorrectly configured in the system
864	Voucher secret is longer than system configured maximum length
865	Too many redeem voucher attempts with incorrect secret has been performed
866	Voucher secret is shorter than system configured minimum length
867	Voucher surname has been set
868	Wrong profile
869	Number of coupons allowed on the offer for the user has already been reached
870	User is not eligible for offer
871	User profile does not exist
1000	Pending

1.3 Bill Payment API initiated by Subscriber

1.3.1 Process

Figure 1-3 shows the process of bill payment confirmation for ECW.



1 SOAP APIs

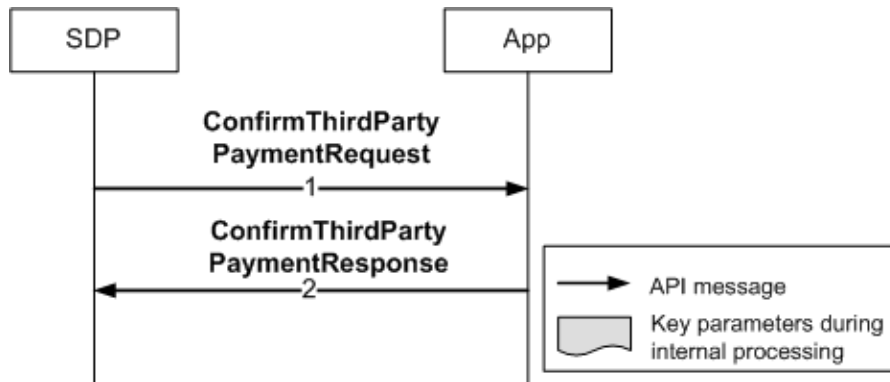
Figure 1-3 Process of bill payment confirmation for ECW

Table 1-19 describes the process for ECW.

Table 1-19 Description for the process of bill payment confirmation for ECW

Step	Description
1	The SDP sends bill payment confirmation request to the App.
2	The App parses the bill payment confirmation request and sends a response to the SDP.

1.3.2 ConfirmThirdPartyPayment

Function

The subscriber through ECW initiates the ConfirmThirdPartyPayment API to pay their Bill on third party system.

Partners must code the App based on the API field requirements so that the App can correctly parse and respond to requests received from the SDP. The App must send a response to the SDP within 30 seconds.

Request URI

The request URI is the destination URI of ConfirmThirdPartyPaymentRequest messages sent by the SDP to the App. The URI is defined by the App.

1.3.3 Request for ConfirmThirdPartyPayment

The SDP functions as the client and sends a ConfirmThirdPartyPaymentRequest message to the App to payment confirmation.



1 SOAP APIs

Example

```
POST /ThirdPartyServiceUMMImpl/UMMSERVICE/ConfirmThirdPartyPayment HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Signature:43AD232FD45FF
```

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
    <ns1:NotifySOAPHeader xmlns:ns1="http://www.huawei.com.cn/schema/common/v2_1">
      <ns1:traceUniqueID>504021505821207281129180006002</ns1:traceUniqueID>
    </ns1:NotifySOAPHeader>
  </soapenv:Header>
  <soapenv:Body>
    <ns2:processRequest xmlns:ns2="http://b2b.mobilemoney.mtn.zm_v1.0/">
      <serviceId>101</serviceId>
      <parameter>
        <name>ProcessingNumber</name>
        <value>12121212</value>
      </parameter>
      <parameter>
        <name>senderID</name>
        <value>MOM</value>
      </parameter>
      <parameter>
        <name>AcctRef</name>
        <value>46733491234</value>
      </parameter>
      <parameter>
        <name>RequestAmount</name>
        <value>1212</value>
      </parameter>
      <parameter>
        <name>paymentRef</name>
        <value>12121</value>
      </parameter>
      <parameter>
        <name>ThirdPartyTransactionID</name>
        <value>12112</value>
      </parameter>
      <parameter>
        <name>MOMAcctNum</name>
        <value> FRI:46733491234/MSISDN </value>
      </parameter>
    </ns2:processRequest>
  </soapenv:Body>
</soapenv:Envelope>
```



1 SOAP APIs

```

<parameter>
  <name>CustName</name>
  <value>121212</value>
</parameter>
<parameter>
  <name>TXNType</name>
  <value>12121</value>
</parameter>
<parameter>
  <name>StatusCode</name>
  <value>12121</value>
</parameter>
<parameter>
  <name>OpCoID</name>
  <value>0</value>
</parameter>
<parameter>
  <name>MerchantName</name>
  <value>dstv1</value>
</parameter>
<parameter>
  <name>CurrCode</name>
  <value>USD</value>
</parameter>
</ns2:processRequest>
</soapenv:Body>
</soapenv:Envelope>

```

Message HTTP Header Parameters

Table 1-20 Describes parameters in a ConfirmThirdPartyPaymentRequest message HTTP header.

Table 1-20 Parameters in a ConfirmThirdPartyPaymentRequest message HTTP header

Header Field	Parameter	Type	Length	Level of Requirement	Description
Signature/		String		Conditional	Signature. The field indicates the summary when the Partner uses HMAC security mode. [Example] 43AD232FD45FF



1 SOAP APIs

Message SOAP Header Parameters

Table 1-21 describes parameters in a ConfirmThirdPartyPaymentRequest message header.

Table 1-21 Parameters in a ConfirmThirdPartyPaymentRequest message header

Parameter	Type	Length	Level of Requirement	Description
traceUniqueID	xsd:string	30	Mandatory	Transaction ID. The ID is automatically generated by the SDP and is used only to trace messages during the SDP commissioning. The App ignores this parameter. [Example] 100001200101110623021721000011

Message Body Parameters

Table 1-22 describes parameters in a ConfirmThirdPartyPaymentRequest message body.

Table 1-22 Parameters in a ConfirmThirdPartyPaymentRequest message body

Parameter	Type	Length	Level of Requirement	Description
serviceId	xsd: string	4	Mandatory	Holds the identification number to identify the service request. This is set as 101.



1 SOAP APIs

Parameter	Type	Length	Level of Requirement	Description
parameter	Parameter[0..20]		Optional	<p>Holds the request parameter details in the form of name/value pair to identify the service details.</p> <p>Valid parameter names are:</p> <ul style="list-style-type: none"> • ProcessingNumber • SenderID • AcctRef • RequestAmount • PaymentRef • ThirdPartyTransactionID • MOMAcctNum • CustName • TXNType • StatusCode • OpCoID • MerchantName • CurrCode <p>For the detailed parameter name information, see Table 1-24.</p> <p>NOTE</p> <p>Fields RequestAmount and PaymentRef are mandatory only for serviceId 101.</p> <p>The currency is the main currency of the country where the OpCo node is located. The minimum current unit is used. For the currencies and their units in the OpCo countries, see 3.1 The Currency and Atomic Charging Unit of each OpCo country.</p> <p>The parameter parameter is of the Parameter[0..20] type and contains multiple sub-parameters. For details about the Parameter[0..20] type, see Table 1-23.</p>

Table 1-23 describes the parameter structure of the Parameter[0..20] type.

Table 1-23 Parameter structure of the Parameter[0..20] type

Parameter	Type	Length	Level of Requirement	Description
name	xsd:string	-	Mandatory	Holds the parameter name.
value	xsd:string	-	Mandatory	Holds the parameter value.



1 SOAP APIs

Table 1-24 describes the parameter lists.

Table 1-24 Parameter name lists

name		value		
Parameter	Type	Length	Level of Requirement	Description
ProcessingNumber	xsd:long	140	Mandatory	An Id to uniquely identify the transaction on the App system.
SenderID	xsd:string	140	Optional	An Id to identify sender. This is set as MOM.
AcctRef	xsd:string	140	Mandatory	Subscriber's account number on the App system. If the subscriber does not have account on the App system, you need confirm it with the Mobile Money system. The values send to SP will be in format like: <ul style="list-style-type: none"> FRI:46733491234/MSISDN FRI:12345ABC@water.company/SP
RequestAmount	decimal	4	Mandatory	The value of the amount to be paid to the App system.
PaymentRef	xsd:string	140	Optional	A description the subscriber provides of the payment. The SDP will forward this parameter to the App system.
ThirdPartyTransactionID	xsd:string	140	Optional	An Id to uniquely identify the transaction on the App system. This parameter is left blank in this instance. This is the ID sent by the App system in requestPayment request ProcessingNumber parameter.
MOMAcctNum	xsd:string	140	Optional	Subscriber's Mobile Money account number. SDP will send the date to SP like : FRI:46733491234/MSISDN 46733491234 is Subscriber MSISDN.
CustName	xsd:string	140	Mandatory	The name of the subscriber.
TXNTType	xsd:string	140	Optional	A code value to identify the type of transaction being executed.
StatusCode	xsd:int	4	Mandatory	A predetermined code to determine the status of the transaction. Value: 01



1 SOAP APIs

name	value			
Parameter	Type	Length	Level of Requirement	Description
OpCoID	xsd:string	140	Optional	Identifier of the country local to be used. Please refer to 3.3 OpCoID List.
MerchantName	xsd:string	30	Optional	It indicates the name of the destination partner.
CurrCode	xsd:string	3	Optional	The currency, as an ISO 4217 formatted string.

1.3.4 Response for ConfirmThirdPartyPayment

The App functions as the server, processes ConfirmThirdPartyPaymentRequest messages received from the SDP, and sends ConfirmThirdPartyPaymentResponse messages to the SDP.

This topic provides a success response example. If a request fails, the SDP sends an error response that contains an error code. For details about error responses, see 1.7 API Error Responses.

Example

HTTP/1.1 200 OK
 Signature:43AD232FD45FF
 Content-Length: 0
 Server: Jetty(7.5.4.v20111024)

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:b2b="http://b2b.mobilemoney.mtn.zm_v1.0/">
  <soapenv:Header/>
  <soapenv:Body>
    <b2b:processRequestResponse>
      <return>
        <name>ProcessingNumber</name>
        <value>121212</value>
      </return>
      <return>
        <name>StatusCode</name>
        <value>121212</value>
      </return>
      <return>
        <name>StatusDesc</name>
        <value>PENDING</value>
      </return>
      <return>
        <name>ThirdPartyAcctRef</name>
        <value>121212</value>
      </return>
    </b2b:processRequestResponse>
  </soapenv:Body>
</soapenv:Envelope>
```




1 SOAP APIs

```
<return>
  <name>Token</name>
  <value>121212</value>
</return>
</b2b:processRequestResponse>
</soapenv:Body>
</soapenv:Envelope>
```

Message HTTP Header Parameters

Table 1-25 Describes parameters in a ConfirmThirdPartyPaymentResponse message HTTP header.

Table 1-25 Parameters in a ConfirmThirdPartyPaymentResponse message HTTP header

Header Field	Parameter	Type	Length	Level of Requirement	Description
Signature/		String		Conditional	Signature. The field indicates the summary when the Partner uses HMAC security mode. [Example] 43AD232FD45FF

Message Body Parameters

Table 1-26 describes parameters in a ConfirmThirdPartyPaymentResponse message body.



1 SOAP APIs

Table 1-26 Parameters in a ConfirmThirdPartyPaymentResponse message body

Parameter	Type	Length	Level of Requirement	Description
return	Return[0..20]	-	Mandatory	<p>Holds the response parameter details in the form of name/value pair to identify the service details.</p> <p>Valid parameter names are:</p> <ul style="list-style-type: none"> ProcessingNumber StatusCode StatusDesc ThirdPartyAcctRef Token <p>For the detailed parameter name information, see Table 1-28.</p> <p>The return parameter is of the Return[0..20] type and contains multiple sub-parameters. For details about the Return[0..20] type, see Table 1-27.</p>

Table 1-27 describes the parameter structure of the Return[0..20] type.

Table 1-27 Parameter structure of the Return[0..20] type

Parameter	Type	Length	Level of Requirement	Description
name	xsd:string	-	Mandatory	Holds the parameter name.
value	xsd:string	-	Mandatory	Holds the parameter value.

Table 1-28 describes the return name lists.

Table 1-28 Return name lists

name	value			
Parameter	Type	Length	Level of Requirement	Description
ProcessingNumber	xsd:string	140	Mandatory	An Id to uniquely identify the transaction on the Mobile Money system.
StatusCode	xsd:int	2	Mandatory	A predetermined code to determine the status of the transaction.



1 SOAP APIs

name	value			
Parameter	Type	Length	Level of Requirement	Description
StatusDesc	xsd:string	140	Optional	The Mobile Money system supplied description of the response code. The value are as follow: <ul style="list-style-type: none"> • PENDING • COMPLETED • FAILED
ThirdPartyAccountRef	xsd:string	140	Optional	Subscriber's account number on the App system. The SDP will fill this parameter as the same value which was received from AcctRef in the request.
Token	xsd:string	140	Optional	A 32 character token representing the prepaid airtime or electricity bought.

1.3.5 Error Codes for ConfirmThirdPartyPayment

The App returns error codes to the SDP when an exception occurs in response to ConfirmThirdPartyPaymentRequest messages. The error codes are defined by partners. The following table describes ConfirmThirdPartyPayment error codes that the partners may return upon an exception.

Table 1-29 ConfirmThirdPartyPayment error codes

Error Code	Description
01	Successfully processed transaction.
02	Invalid account details.
03	Transaction not allowed.
04	Payment below minimum.
05	Invalid Fundamo userid/password.
06	Payment Entry not found.
07	Missing transaction/message ID.
08	Missing application version.
100	General failure.
101	Field length exceeded.



1 SOAP APIs

Error Code	Description
102	Subscriber could not be identified. (Invalid subscriberID).
103	Duplicate transaction id.



1 SOAP APIs

1.4 Deposit APIs

1.4.1 Process

Figure 1-4 shows the process of deposit for ECW.

Figure 1-4 Process of deposit for ECW

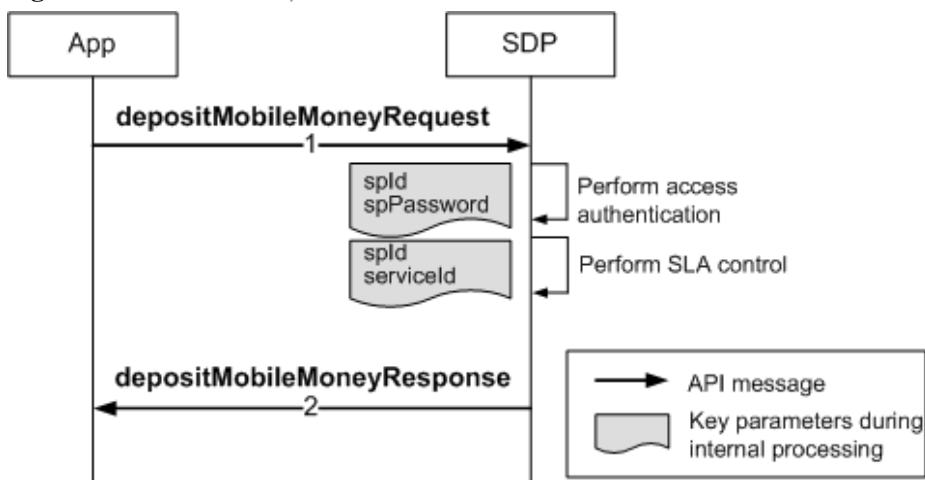


Table 1-30 describes the process for ECW.

Table 1-30 Description for the process of deposit for ECW

Step	Description
1	The App invokes the depositMobileMoney API to deposit mobile money in subscriber's mobile money account.
2	<ul style="list-style-type: none"> The SDP performs authentication and service level agreement (SLA) control based on fields in the request. The SDP sends deposit response to the App.

1.4.2 DepositMobileMoney

Function

The App invokes the depositMobileMoney API to deposit money in Subscriber's mobile money account. Partners must code the App based on the API field requirements so that the App can send correct requests to the SDP. The SDP must send a response within 180 seconds by default.



1 SOAP APIs

Request URI

The request URI is the destination URI of depositMobileMoney messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/ThirdPartyServiceUMMImpl/UMMServiceService/DepositMobileMoney/v17

In the format, **IP** and **Port** indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

1.4.3 Request for DepositMobileMoney

The App functions as the client and sends a depositMobileMoneyRequest message to the SDP to deposit money in Subscriber s mobile money account.

Example

```
POST /ThirdPartyServiceUMMImpl/UMMServiceService/DepositMobileMoney/v17 HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Signature:43AD232FD45FF
Cookie: sessionId=default8fcee064690b45faa9f8f6c7e21c5e5a
```

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:b2b="http://b2b.mobilemoney.mtn.zm_v1.0/">
  <soapenv:Header>
    <RequestSOAPHeader xmlns="http://www.huawei.com.cn/schema/common/v2_1">
      <spId>35000001</spId>
      <spPassword>de96d901b3bad1db2aab76b7b0b202f2</spPassword>
      <bundleID>256000039</bundleID>
      <serviceId>35000001000035</serviceId>
      <timeStamp>20100727</timeStamp>
    </RequestSOAPHeader>
  </soapenv:Header>
  <soapenv:Body>
    <b2b:processRequest>
      <serviceId>201</serviceId>
      <parameter>
        <name>ProcessingNumber</name>
        <value>555</value>
      </parameter>
      <parameter>
        <name>serviceId</name>
        <value>102</value>
      </parameter>
      <parameter>
        <name>SenderID</name>
```



1 SOAP APIs

```

        <value>MOM</value>
    </parameter>
    <parameter>
        <name>PrefLang</name>
        <value>121212121</value>
    </parameter>
    <parameter>
        <name>OpCoID</name>
        <value>0</value>
    </parameter>
    <parameter>
        <name>MSISDNNum</name>
        <value> FRI:46733491234/MSISDN </value>
    </parameter>
    <parameter>
        <name>Amount</name>
        <value>10</value>
    </parameter>
    <parameter>
        <name>Narration</name>
        <value> FRI:< AcctRef > @water.company/SP </value>
    </parameter>
    <parameter>
        <name>IMSINum</name>
        <value>86</value>
    </parameter>
    <parameter>
        <name>OrderDateTime</name>
        <value>20100727</value>
    </parameter>
    <parameter>
        <name>CurrCode</name>
        <value>USD</value>
    </parameter>
</b2b:processRequest>
</soapenv:Body>
</soapenv:Envelope>

```

Message HTTP Header Parameters

Table 1-31 Describes parameters in a depositMobileMoneyRequest message HTTP header.



1 SOAP APIs

Table 1-31 Parameters in a depositMobileMoneyRequest message HTTP header

Header Field	Parameter	Type	Length	Level of Requirement	Description
Cookie	sessionid	String		Conditional	<p>The session id is used as a cookie in the HTTP header in the requests towards the Mobile Money Manager when the Partner use Login/Logout mode.</p> <p>[Example] default8fcee064690b45faa9f8f6c7e21c5e5a</p> <p>NOTE Cookie is valid only for ECW and not for fundamo.</p>
Signature/		String		Conditional	<p>Signature.</p> <p>The field indicates the summary when the Partner uses HMAC security mode.</p> <p>[Example] 43AD232FD45FF</p>



1 SOAP APIs

Message SOAP Header Parameters

Table 1-32 describes parameters in a depositMobileMoneyRequest message SOAP header.

Table 1-32 Parameters in a depositMobileMoneyRequest message SOAP header

Parameter	Type	Length	Level of Requirement	Description
spId	xsd: string	21	Mandatory	<p>Partner ID.</p> <p>The ID is automatically allocated by the SDP to partners after successful registration. To obtain the ID:</p> <ul style="list-style-type: none">• A service Partner and API Partner can log in to the SDP management portal and query account information.• A Developer can log in to the Developer Portal and query account information. <p>[Example]</p> <p>000201</p>



1 SOAP APIs

Parameter	Type	Length	Level of Requirement	Description
spPassword	xsd: string	100	Conditional	<p>Authentication key for the SDP to authenticate partners.</p> <p>The SDP supports authentication by SP ID + Password, SP ID + IP address + Password, or SP ID + IP address. Partners select an authentication mode during registration. If a partner selects authentication by SP ID + Password or SP ID + IP address + Password, this parameter is mandatory in requests sent by this partner.</p> <p>The value is a character string encrypted. The encryption formula is as follows:</p> <ul style="list-style-type: none"> SHA-256: spPassword = Base64(SHA-256(spld + Password + timeStamp)) MD5: spPassword = MD5(spld + Password + timeStamp) <p>In the formula:</p> <ul style="list-style-type: none"> spld and timeStamp: authentication ID and timestamp. Password: password for partners to access the SDP. <ul style="list-style-type: none"> A service Partner and API Partner can obtain the password from the email notification received after successful registration. A Developer can log in to the Developer Portal, choose Member Center > Account > Registration Information > Invoke Password, and set the password. <p> NOTE To retain features of earlier versions, the SP uses the MD5 algorithm in the connection to the SDP, which might cause security risks.</p> <p>[Example] e6434ef249df55c7a21a0b45758a39bb</p>



1 SOAP APIs

Parameter	Type	Length	Level of Requirement	Description
bundleID	xsd: string	21	Conditional	<p>Bundle ID.</p> <p>When SDP creates a capability bundle, SDP allocates a bundleID to capability bundle.</p> <p>The bundleID must not be contained during invocation of a service interface developed by service partners and other partners, and must be contained during invocation of a capability interface developed by API partners, other partners, and developers.</p> <p>[Example] 256000039</p>
serviceID	xsd: string	21	Conditional	<p>Service ID.</p> <p>The ID is automatically allocated by the SDP to services after successful release. An service Partner can log in to the SDP management portal and query service information for the ID.</p> <p>The serviceID must be contained during invocation of a service interface developed by service partners and other partners, and must not be contained during invocation of a capability interface developed by API partners, other partners, and developers.</p> <p>[Example] 35000001000001</p>
timeStamp	xsd: string	14	Conditional	<p>Timestamp (UTC time).</p> <p>The value is used in MD5 encryption of spPassword.</p> <p>This parameter is mandatory when the spPassword parameter is required.</p> <p>[Format] yyyyMMddHHmmss</p> <p>[Example] 20100731064245</p>



1 SOAP APIs

Message Body Parameters

Table 1-33 describes parameters in a depositMobileMoneyRequest message body.

Table 1-33 Parameters in a depositMobileMoneyRequest message body

Parameter	Type	Length	Level of Requirement	Description
serviceld	xsd: string	4	Mandatory	Holds the identification number to identify the service request. This is set as 201.
parameter	Parameter[0..20]	-	Optional	<p>Holds the request parameter details in the form of name/value pair to identify the service details.</p> <p>Valid parameter names are:</p> <ul style="list-style-type: none"> • ProcessingNumber • serviceld • SenderID • PrefLang • OpCoID • MSISDNNum • Amount • Narration • IMSINum • OrderDateTime • CurrCode <p>For the detailed parameter name information, see Table 1-35.</p> <p>The parameter parameter is of the Parameter[0..20] type and contains multiple sub-parameters. For details about the Parameter[0..20] type, see Table 1-34.</p>



1 SOAP APIs

Table 1-34 describes the parameter structure of the Parameter[0..20] type.

Table 1-34 Parameter structure of the Parameter[0..20] type

Parameter	Type	Length	Level of Requirement	Description
name	xsd:string	-	Mandatory	Holds the parameter name.
value	xsd:string	-	Mandatory	Holds the parameter value.

Table 1-35 describes the parameter name lists.

Table 1-35 Parameter name lists

name	value			
Parameter	Type	Length	Level of Requirement	Description
ProcessingNumber	xsd:string	140	Mandatory	An Id to uniquely identify the transaction on the App system.
serviceId	xsd:string	140	Optional	This relates to the Code configured in the Mobile Money system. Properties file is used to uniquely identify each service provided.
SenderId	xsd:string	140	Optional	An Id to identify sender.
PrefLang	xsd:string	140	Optional	The language locale to be used.
OpCoId	xsd:int	140	Optional	Identifier of the country local to be used. Please refer to 3.3 OpCoId List.
MSISDNNum	xsd:string	30	Mandatory	<p>An Id to uniquely identify the subscriber on the App system.</p> <p>Subscriber's MSISDN number or Fake ID.</p> <p>[Format]</p> <ul style="list-style-type: none"> • Mobile number: [Country code][Mobile number] • Fake ID: [Prefix]- [opcoid]-[sequence]. <p>[Example]</p> <ul style="list-style-type: none"> • Mobile number: 8612312345678 • Fake ID: f-245-11900000007639 <p>Partner need to map the MSISDN from to FRI format expected by Mobile Money.</p> <p>e.g. FRI: 8612312345678/MSISDN</p>



1 SOAP APIs

name	value			
Parameter	Type	Length Level of Requirement		Description
Amount	decimal	4	Mandatory	The value of the amount to be deposited into the subscriber's Mobile Money account.
Narration	xsd:string	140	Optional	A descriptive message for receiver's transaction history. If present, it will also be added to the notification text sent to the receiver. Partner need to map the Narration value to FRI format expected by Mobile Money. e.g. FRI:< AcctRef > @water.company/SP
IMSINum	xsd:string	140	Optional	Subscriber's IMSI number.
OrderDateTime	xsd:dateTime	140	Optional	Time when subscriber ordered the service.
CurrCode	xsd:string	3	Optional	The currency, as an ISO 4217 formatted string.

1.4.4 Response for DepositMobileMoney

The SDP functions as the server, processes depositMobileMoneyRequest messages received from the App, and sends depositMobileMoneyResponse messages to the App.

This topic provides a success response example. If a request fails, the SDP sends an error response that contains an error code. For details about error responses, see API Error Responses.

Example

HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <ns1:processRequestResponse xmlns:ns1="http://b2b.mobilemoney.mtn.zm_v1.0/">
      <return>
        <name>ProcessingNumber</name>
        <value>123</value>
      </return>
      <return>
        <name>SenderID</name>
```



1 SOAP APIs

```

        <value>MOM</value>
      </return>
    </return>
    <name>StatusCode</name>
    <value>234</value>
  </return>
  <return>
    <name>StatusDesc</name>
    <value>FAILED</value>
  </return>
  <return>
    <name>OpCoID</name>
    <value>0</value>
  </return>
  <return>
    <name>IMSINum</name>
    <value>86</value>
  </return>
  <return>
    <name>MSISDNNum</name>
    <value>13132132000</value>
  </return>
  <return>
    <name>OrderDateTime</name>
    <value>20100727</value>
  </return>
  <return>
    <name>ThirdPartyAcctRef</name>
    <value>121212</value>
  </return>
  <return>
    <name>MOMTransactionID</name>
    <value>456</value>
  </return>
</ns1:processRequestResponse>
</soapenv:Body>
</soapenv:Envelope>

```

Message HTTP Header Parameters

Table 1-36 Describes parameters in a depositMobileMoneyResponse message HTTP header.



1 SOAP APIs

Table 1-36 Parameters in a depositMobileMoneyResponse message HTTP header

Header Field	Parameter	Type	Length	Level of Requirement	Description
Signature/		String		Conditional	Signature. The field indicates the summary when the Partner uses HMAC security mode. [Example] 43AD232FD45FF

Message Body Parameters

Table 1-37 describes parameters in a depositMobileMoneyResponse message body.

Table 1-37 Parameters in a depositMobileMoneyResponse message body

Parameter	Type	Length	Level of Requirement	Description
return	Return[0..20]	-	Mandatory	<p>Holds the response parameter details in the form of name/value pair to identify the service details.</p> <p>Valid parameter names are:</p> <ul style="list-style-type: none"> ProcessingNumber SenderID StatusCode StatusDesc OpCoID IMSINum MSISDNNum OrderDateTime ThirdPartyAcctRef MOMTransactionID <p>For the detailed parameter name information, see Table 1-39.</p> <p>The return parameter is of the Return[0..20] type and contains multiple sub-parameters. For details about the Return[0..20] type, see Table 1-38.</p>

Table 1-38 describes the parameter structure of the Return[0..20] type.



1 SOAP APIs

Table 1-38 Parameter structure of the Return[0..20] type

Parameter	Type	Length	Level of Requirement	Description
name	xsd:string	-	Mandatory	Holds the parameter name.
value	xsd:string	-	Mandatory	Holds the parameter value.

Table 1-39 describes the return name lists.

Table 1-39 Return name lists

name	value			
Parameter	Type	Length	Level of Requirement	Description
ProcessingNumber	xsd:string	140	Optional	An Id to uniquely identify the transaction on the App system.
SenderID	xsd:string	140	Optional	An Id to identify sender. This is set as MOM.
StatusCode	xsd:int	2	Mandatory	A predetermined code to determine the status of the transaction.
StatusDesc	xsd:string	140	Optional	The Mobile Money system supplied description of the response code. The value are as follow: <ul style="list-style-type: none"> • SUCCESSFUL • PENDING • FAILED
OpCoID	xsd:string	140	Optional	Identifier of the country local to be used. Please refer to 3.3 OpCoID List.
IMSINum	xsd:string	140	Optional	Subscriber's IMSI number.
MSISDNNum	xsd:string	30	Optional	Subscriber's MSISDN number or Fake ID.
OrderDateTime	xsd:dateTime	140	Optional	Time when subscriber ordered the service.
ThirdPartyAccountRef	xsd:string	140	Optional	Subscriber's account number on the App system. The SDP will fill this parameter as the same value which was received from AcctRef in the request.



1 SOAP APIs

name	value			
Parameter	Type	Length	Level of Requirement	Description
MOMTransactionID	xsd:long	140	Optional	An Id to uniquely identify the transaction on the Mobile Money system. The SDP will forward this parameter to the App system.

1.4.5 Error Codes for DepositMobileMoney

Table 1-40 describes depositMobileMoney error codes that the SDP may return upon an exception. For details about the error codes, see the *SDP Solution Error Code Reference*.

Table 1-40 depositMobileMoney error codes

Error Code	Description
01	Successfully processed transaction.
02	Invalid account details.
03	Transaction not allowed.
04	Payment below minimum.
05	Invalid Fundamo userid/password.
06	Payment Entry not found.
07	Missing transaction/message ID.
08	Missing application version.
100	General failure.
	Get addr route info failed!the baseScfType = %1,the sourceAddr = %2,the destAddr = %3.
	NOTE ScfType refers to Service Capability Function Type.
	MDSP AuthPrice failed, result code is %1.
	NOTE The MDSP is composed of the Mobile Data Control Center (MDCC) subsystem and peripheral functional subsystems such as the Portal and Mobile Data Management Center (MDMC). The MDCC subsystem consists of core modules such as the Short Message Push Agent (SMPA), Network Management System (NMS) proxy, and BOSS interface machine. The MDSP is connected to the NMS, BOSS system, Service Control Point (SCP), customer service center, and service gateways. The MDSP can implement the functions such as external entity access, mobile data management, user and service authentication, and charging.
	Waiting for response timed out, message type is %1.
	Cannot find route policy, the policy identifier is %1.



1 SOAP APIs

Error Code	Description
	The service status is unknown.
	Service %1 is in blacklist!
	The API %1 is not existed.
	Service ID %1 is not exist!
	The service status is suspended.
	The service status is pre-deregistered.
	The service status is deregistered.
	The API status is disabled.
	SP status is locked.
	The sp's Status is forbidden.
	The sp's status is pause.
	The RemotePassword of NE %1 is empty!
	NOTE NE refers to network element, such as SMSC, MMSC.
	NEInfo id %1 not in SAG.
	NOTE The SAG provides open interfaces for third-party partners. It authenticates and authorizes the partners, and controls the Service Level Agreement policies when it receives third-party service requests.
	The ScfType %1 is inactive!
	NOTE ScfType refers to Service Capability Function Type.
	The ScfType %1 is uninstalled!
	NOTE ScfType refers to Service Capability Function Type.
	The service status is configuring.
	The Created is null.
	SP ip is null!
	Sp password is null!
	The Nonce is null.
	Authentication Failed, cause by SP, because of timestamp expired.



1 SOAP APIs

Error Code	Description
	Service ID is null!
	Sp password is null!
	Timestamp is empty in soapheader.
	Service ID %1 is invalid!
	The PasswordDigest is not accepted.
	The sp's Status is pre-deregistered.
	The sp's Status is deregistered.
	The sp's Status is unknown.
	SP %1 is in blacklist!
	SPID %1 is not exist!
	Sp password is not accepted!
	Sp ip %1 is not accepted!
	local SP password is null!
	The authentication type is unknown!
	Authentication Failed, cause by SP, because of timestamp is not valid.
	Minimum Amount per Transaction
	%1 is null.
101	Field length exceeded.
102	Subscriber could not be identified. (Invalid subscriberID).
	%1 is null.
103	Duplicate transaction id.
104	Authorisation declined.
105	Account holder is not active.
106	Deposit Transfer not processed: Source & destination accounts are same
108	Deposit Transfer not processed: Insufficient funds
109	Deposit Transfer not processed: Daily limit has been exceeded.
110	Deposit Transfer not processed: Sender account not active



1 SOAP APIs

Error Code	Description
111	Deposit Transfer not processed: Target account not active
501	DATEFORMAT_INVALID
502	FRI_INVALID
503	IDENTITY_INVALID
504	INVALID_FIELD_VALUE
505	MANDATORY_FIELD_MISSING
506	OTHER_FINANCIAL_ERROR
507	PREAPPROVAL_NOT_FOUND
508	REFERENCE_ID_NOT_FOUND
509	REQUIRED_FIELD_MISSING
510	SERVICE_TEMPORARILY_UNAVAILABLE
511	SESSION_INVALID
512	TRANSACTION_ID_OR_ORIGINAL_TRANSACTION_ID_MISSING
513	TRANSACTION_NOT_COMPLETED
514	TRANSFER_TYPE_AND_ACCOUNT_DO_NOT_MATCH
515	ACCOUNTHOLDER_WITH_FRI_NOT_FOUND
516	ACCOUNTHOLDER_WITH_MSISDN_NOT_FOUND
517	AUTHORIZATION_INVALID_ACCOUNT_TYPE_COMBINATION
518	COULD_NOT_PERFORM_OPERATION
519	EXPIRED_OR_INVALID_TRANSACTION_ID
520	INCORRECT_PASSWORD
521	NOT_AUTHORIZED
522	AMOUNT_INVALID
523	CURRENCY_NOT_SUPPORTED
524	INVALID_CURRENCY
525	RESOURCE_NOT_ACTIVE
526	SOURCE_NOT_FOUND



1 SOAP APIs

Error Code	Description
527	RESOURCE_NOT_FOUND
529	TARGET_AUTHORIZATION_ERROR
530	This account is already a member of this group
531	Account could not be removed from Bank domain
532	The specified account holder's additional information name already exist
533	The specified account holder's additional information name cannot be deleted because it is in use
534	This account holder is already a member of this group
535	AccountHolder does not exist
536	This operation requires the user to be logged in
537	AccountHolder profile does not have requested account
538	AccountHolder status should be blocked or active
539	Account not found
540	Could not create account route since Account route with provided account route reference and currency already exists
541	Could not delete account route since Account route with provided account route reference and currency does not exists
542	Account type is not supported
543	The account holder already has a connection to the specified account
544	Activation of account holder failed
545	Could not perform operation since at least one of the mandatory field is required
546	Could not perform operation since at least one of the mandatory field is required
547	Could not perform operation since at least one of the mandatory field is required
548	Could not perform operation since at least one of the mandatory field is required
549	Could not perform operation since at least one of the mandatory field is required
550	Account holder could not be found
551	Operation requires account holder to have an MSISDN identity but it has none
552	Account holder is inactive



1 SOAP APIs

Error Code	Description
553	A specified account holder was not a child to another specified account holder
554	Account holder could not be found
555	The settings for this account holder already exists
556	Account holder settings are mandatory
557	Account holder settings not found
558	Account holder could not be registered because an account holder with submitted Alias already exists
559	Account holder could not be registered because an account holder with submitted E-mail already exists
560	Account holder with given e-mail could not be found
561	Account holder cannot have the same employee id as another employee working under the same employer
562	Account holder could not be registered because an account holder with submitted External ID already exists
563	Account holder IMSI could not be updated because it already exists
564	Account holder could not be registered because an account holder with submitted MSISDN already exists
565	Account holder could not be registered because an account holder with submitted Username already exists
566	Adjustments from account not allowed
567	Adjustments not allowed on account
568	Adjustments to account not allowed
569	No Alias with this name was found in the system
570	Could not perform operation since the resource already belongs to another Bank domain
571	The FRI refers to more than one account
572	Account holder is not allowed to perform this operation
573	Account holder is already activated
574	Insufficient amount
575	Setting an account holder as its own parent or a creating a cycle in the hierarchy is not allowed



1 SOAP APIs

Error Code	Description
576	Authorization failed
577	Maximum transfer account throughput amount including fee threshold exceeded
578	Maximum transfer account throughput amount including fee threshold exceeded for receiver
579	Maximum transfer account throughput amount including fee threshold exceeded for sender
580	Maximum transfer amount threshold exceeded
581	Maximum transfer amount including fee threshold exceeded
582	Maximum transfer amount including fee threshold exceeded for receiver
583	Maximum transfer amount including fee threshold exceeded for sender
584	Maximum transfer amount threshold exceeded for receiver
585	Maximum transfer amount threshold exceeded for sender
586	Maximum transfer counter threshold exceeded
587	Maximum send amount threshold exceeded
588	Authorization failed. Maximum threshold limit reached for Approver
589	Amount below minimum amount threshold
590	Amount below minimum amount threshold for account
591	Amount below minimum amount threshold for account reference
592	Account holder is not registered
593	Receivers account does not allow deposit
594	Could not perform operation because amount is below minimum first deposit amount threshold
595	Maximum balance threshold for receiver exceeded
596	Receivers account is not active
597	Receiving account is unavailable
598	Withdrawal from recipient account not allowed
599	Minimum balance threshold for sender exceeded
600	Sending account is unavailable
601	A bank account with given bank code and account number could not be found
602	A route to the wanted bank account could not be found



1 SOAP APIs

Error Code	Description
603	No bank domain set in the system.
604	Bank domain with given ID was not found
605	Could not perform operation since the Bank domain is currently in use
606	Could not perform operation since Bank domain with given name already exists
607	Account could not be blocked
608	Receiving account holder has insufficient permission to receive cashin
609	Receiving account holder has insufficient permission to receive cashout
610	Certificate already updated for the user
611	Certificate credentials already exist
612	Certificate credentials do not exist
613	Account could not be closed
614	The company information already exist
615	The company information does not exist
616	Contact was not added since it already exists
617	Contact with given FRI could not be found
618	Could not create account holder
619	Could not perform add account operation
620	Failed to approve the operation, please retry the initial operation again
621	ATM partial reversal operation could not be performed
622	ATM redeem voucher operation could not be performed
623	ATM rollback voucher operation could not be performed
624	Could not perform the operation, the account has to be linked with sp account.
625	Receive remittance failed
626	Could not perform update of account holder settings
627	Could not perform update of pending payment status
628	The specified counter configuration does not exist.
629	The credential is not in the expected suspended state



1 SOAP APIs

Error Code	Description
630	Operation not allowed because credential not in active state
631	Repeated secrets does not match each other
632	Given credential type does not exist in the system
633	For this operation currency conversion is not allowed
634	Custody Account already exists
635	Custody account certificate already exist
636	Custody account certificate could not deleted
637	Custody account certificate could not be empty
638	Custody account certificate could not be used for non-repudiation
639	Custody account certificate does not exist
640	Custody account certificate error
641	Unsupported algorithm for custody account certificate
642	Custody account certificate with same id already exist
643	Name and Bank Domain are same as before
644	Could not find CustodyAccount since Custody Account with provided account number and bank code does not exists
645	Custody account still contains money
646	This Custody account still have pending transaction
647	Could not update custody account as name and bank domain both are missing
648	Account holder and account must belong to the same bank domain
649	The document does not exist
650	The document rules has been incorrectly configured in the system
651	This document URL is invalid
652	This document URL is not allowed
653	End date can not occur in the future
654	This error originates from an external system
655	External service provider with given User name could not be found



1 SOAP APIs

Error Code	Description
656	Netting report error
657	Financial Configuration does not exist
658	Neither transaction with transaction ID nor original transaction with original transaction ID could be found
659	Using first name in password is not allowed
660	The specified float transfer exemption rule already exists
661	Setting multiple owners of an FRI is not allowed
662	FRI Reference not found.
663	Operation requires that start date has to occur before end date
664	Could not perform operation since the Group is already used by another Bank domain
665	Group member could not be removed from group
666	Group member could not be removed from Bank domain
667	Group member with given ID could not be found in this group
668	Group used by Bank domain is not allowed to be removed
669	It's not possible to mirror a High Contention account in SDP since it doesn't have a balance.
670	Home charging region with given ID was not found
671	Could not perform operation since the home charging region is currently in use
672	Could not perform operation since Home Charging Region with given name already exists
673	Length of identification is not within length boundaries
674	Duplicate identification
675	ID type not supported by ISO 20022
676	The credential status change is illegal
677	Format of IMSI is invalid
678	Inactive receiver account
679	Incorrect pin code
680	Security answer was not correct
681	Transaction state is incorrect for this operation



1 SOAP APIs

Error Code	Description
682	An internal error caused the operation to fail
683	Could not find Internal Loan Account with given Id
684	Amount Min cannot be larger than Amount Max
685	The state of this Internal Loan Application is not valid for this operation
686	Could not find Internal Loan Application
687	Maximum number of Internal Loan Applications reached
688	Internal Loan Product already exist
689	The Internal Loan Product status is not active
690	The given Internal Loan Product is not allowed to be used
691	Could not find Internal Loan Product with given identity and product code
692	Internal Loan Provider already exist
693	Could not find Internal Loan Provider with given identity
694	The FRI must be of account type
695	The specified account holder's additional information name is not valid or does not exist
696	Account holder settings are invalid
697	Approval transaction has invalid state
698	Invalid ATM operation
699	Certificate is invalid
700	Invalid Delivery method
701	Profile validation
702	Profile counter validation
703	Profile name validation
704	Profile threshold validation
705	Invalid receiving end user
706	Incorrect remittance status
707	A transaction can only be scheduled from a future date
708	Invalid scheduled transaction status for the requested operation



1 SOAP APIs

Error Code	Description
709	Settlement deposit with given ID could not be found
710	Settlement with given ID could not be found
711	The provided signature is not correct
712	Current status is not as expected
713	Invalid Transaction type
714	Invalid User (or financial) approval type or not registered
715	Bank domain could not be removed since it is the last remaining
716	Limited retries are not allowed for the specified service provider
717	Could not find Loan Application with given approval ID
718	Could not find Loan Provider with given name
719	Reached limit for max accounts
720	Maximum number of contacts threshold exceeded
721	Maximum number of documents exceeded
722	Actor MSISDN is missing
723	Exchange rate for conversion between currencies does not exist.
724	More than one of a set of optional but mutual parameters were set.
725	Given msisdn already used by a point of sale
726	User has no access permission
727	Using non ASCII characters in password is not allowed
728	Non-digit characters in pin code
729	It is not allowed to change bank domain for an account holder
730	One custody account must exist
731	Only one refund per debit is allowed
732	Operation not supported on external account
733	Operation not supported on a migrated account
734	Request did not contain any OTP
735	Request did not contain any OTP



1 SOAP APIs

Error Code	Description
736	Request did not contain any OTP
737	Format of OTP is invalid
738	OTP not found, has expired or is used for the wrong operation
739	Password has already been used before
740	Password too short
741	Could not find Pending Commission
742	Pickup with given MTCN could not be found
743	PIN code has already been used
744	PIN code denied by matching birth date
745	PIN code denied by matching identification number
746	PIN code denied by matching MSISDN
747	PIN code denied by matching regular expression
748	Wrong profile
749	Pin code too long
750	Pin code too short
751	The point of sale with given msisdn is not acquired.
752	No pre-approvals found
753	Profile name validation
754	Json File validation
755	A profile mismatch between for example a provided counter configuration (belonging to one profile) and provided profile reference (having a different profile)
756	Profile not found
757	Profile type validation
758	Profile version validation
759	Profile version validation
760	Tag was not found
761	Provider Category can not be deleted as it is linked with account holder



1 SOAP APIs

Error Code	Description
762	No provider category with this name was found in the system
763	Request is queued for approval
764	Quote has already been used
765	Quote has expired
766	Quote not found
767	Quote is not supported with the combination of parameters
768	Quote is already processing
769	Account could not be re-activated
770	Failed to process reconciliation file.
771	Recurring scheduled transaction can only be scheduled a limited number of days into the future
772	Refund amount cannot be larger than the debit amount
773	The relative does not exist
774	Remittance Id does not exist in dbase
775	Remittance not found, Invalid provider name or remittance id
776	Remittance not possible because given data is invalid.
777	Password criteria configuration invalid
778	FRI not available
779	Required resource is temporarily locked
780	Reversal has already been performed on this transaction
781	Reversal is not allowed on this type of transaction
782	Reversal not possible since sub transaction is invalid
783	Using reversed user id in password is not allowed
784	Transaction could not be performed since neither of the parties supports rollback
785	Transaction can only be scheduled a limited number of days into the future
786	The requested scheduled transaction could not be found
787	This self service notification already exists.



1 SOAP APIs

Error Code	Description
788	The notification limit exceeds the hard limit
789	Self service notificaiton not found.
790	Max number of self serviced notifications reached, cannot create another.
791	The response amount does not match with the amount in the Payment Instruction
792	Another settlement task is ongoing.
793	Balance reconciliation failed. Provided closing balance does not match with the expected closing balance of custody account
794	Certificate does not exist for custody account not able to verify signature of settlement file
795	Custody account already exists with settlement configuration
796	Could not find settlement Configuration for custody account
797	Settlement configuration for identifier already exists
798	Instruction formatter class not found
799	Settlement configuration not found
800	Settlement parser class not found
801	Unable to remove custody account id from settlement configuration
802	Currency does not match with custody account's currency.
803	Settlement process failed because the custody accounts information do not match.
804	The deposit has already been reversed.
805	The deposit is in an invalid state.
806	The deposit was not found.
807	The transaction details of the deposit where not found.
808	Duplicated deposit or reverse deposit record found.
809	Digest does not exist or digest verification failed for settlement file
810	Error in parsing settlement file
811	Signature does not exist or signature verification failed for settlement file
812	Payment Instruction Response caller identifier mismatch
813	Settlement identifier not found



1 SOAP APIs

Error Code	Description
814	Invalid or no amount in Payment Instruction Response.
815	Settlement error due to invalid receiver. There is no such account holder or custody account exists.
816	Settlement is in wrong state
817	Online settlement not enabled for custody account
818	Opening Balance reconciliation failed
819	Wrong state for Settlement PaymentInstruction.
820	Settlement PaymentInstruction not found.
821	Failed to process settlement file.
822	The receiver account does not match
823	Wrong state for Settlement CustodyAccountsTransfers.
824	Using surname in password is not allowed
825	Not possible to put account in suspended state
826	Tag could not be created because a Tag with submitted name already exists
827	Tag was not found
828	Tax report not found
829	The specified threshold configuration does not exist.
830	Too few lower case alphabetic characters in password
831	Too few digits in password
832	Too few special characters in password
833	Too few upper case alphabetic characters in password
834	Too few upper or lower case alphabetic characters in password
835	Too many consecutive digits in PIN code
836	Too many identical consecutive characters in password
837	Too many repeated digits in PIN code
838	Transaction not found
839	Transaction timed out



1 SOAP APIs

Error Code	Description
840	Operation is not supported for this account
841	Account holder could not be created
842	Interest tier could not be created
843	Interest tier could not be modified
844	Interest tier could not be removed
845	Account could not be un-blocked
846	Unresolved Remittance Id does not exist in dbase
847	The maximum attempts of credential unsuspend for the given account holder has been reached
848	Using user id in password is not allowed
849	The data provided in the request failed the validation check
850	Voucher date of birth has been set
851	Voucher has expired
852	Voucher external id has been set
853	Voucher first name has been set
854	Voucher is no longer active
855	Voucher with given identity could not be found in the system
856	Voucher secret contains letters that has been excluded through system configuration
857	Voucher secret contains characters that is not legal according to system configuration
858	Voucher secret contains non ASCII characters
859	Voucher secret can not contain digits
860	Voucher secret is not correct
861	Voucher secret element is empty
862	Voucher secret can not contain letters
863	The voucher secret rules has been incorrectly configured in the system
864	Voucher secret is longer than system configured maximum length
865	Too many redeem voucher attempts with incorrect secret has been performed



1 SOAP APIs

Error Code	Description
866	Voucher secret is shorter than system configured minimum length
867	Voucher surname has been set
868	Wrong profile
869	Number of coupons allowed on the offer for the user has already been reached
870	User is not eligible for offer
871	User profile does not exist

1.5 APIs for Querying Subscriber Account Balances

1.5.1 Process

Figure 1-5 shows the process of querying subscriber account balance for ECW.

Figure 1-5 Process of querying subscriber account balance(ECW)

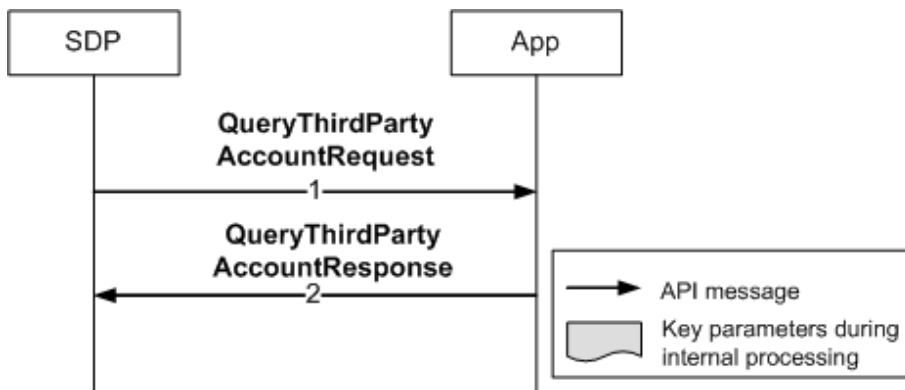


Table 1-41 describes the process.

Table 1-41 Description for the process of querying subscriber account balance(ECW)

Step	Description
1	The SDP sends querying subscriber account balance request to the SDP.
2	The App parses the querying subscriber account balance request and sends a response to the SDP.



1 SOAP APIs

1.5.2 QueryThirdPartyAccount*Function*

The SDP (functioning as the client) invokes the QueryThirdPartyAccount API to send querying subscriber account balance requests to the App (functioning as the server). Partners must code the App based on the API field requirements so that the App can correctly parse and respond to requests received from the SDP. The App must send a response to the SDP within 30 seconds.

Request URI

The request URI is the destination URI of QueryThirdPartyAccountRequest messages sent by the SDP to the App. The URI is defined by the App.

1.5.3 Request for QueryThirdPartyAccount

The SDP functions as the client and sends a QueryThirdPartyAccountRequest message to the App to query subscriber account balance.

Example

```
POST /ThirdPartyServiceUMMImpl/UMMSERVICE/QueryThirdPartyAccount HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Signature:43AD232FD45FF
```

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
    <ns1:NotifySOAPHeader xmlns:ns1="http://www.huawei.com.cn/schema/common/v2_1">
      <ns1:traceUniqueID>504021505821207271704550002002</ns1:traceUniqueID>
    </ns1:NotifySOAPHeader>
  </soapenv:Header>
  <soapenv:Body>
    <ns2:processRequest xmlns:ns2="http://b2b.mobilemoney.mtn.zm_v1.0/">
      <serviceId>100</serviceId>
      <parameter>
        <name>ProcessingNumber</name>
        <value>1212121</value>
      </parameter>
      <parameter>
        <name>senderID</name>
        <value>MOM</value>
      </parameter>
    </ns2:processRequest>
  </soapenv:Body>
</soapenv:Envelope>
```



1 SOAP APIs

```
<parameter>
  <name>AcctRef</name>
  <value> FRI:46733491234/MSISDN </value>
</parameter>
<parameter>
  <name>PrefLang</name>
  <value>121212121</value>
</parameter>
<parameter>
  <name>OpCoID</name>
  <value>0</value>
</parameter>
</ns2:processRequest>
</soapenv:Body>
</soapenv:Envelope>
```

Message HTTP Header Parameters

Table 1-42 Describes parameters in a QueryThirdPartyAccountRequest message HTTP header.

Table 1-42 Parameters in a QueryThirdPartyAccountRequest message HTTP header

Header Field	Parameter	Type	Length	Level of Requirement	Description
Signature/		String		Conditional	Signature. The field indicates the summary when the Partner uses HMAC security mode. [Example] 43AD232FD45FF

Message SOAP Header Parameters

Table 1-43 describes parameters in a QueryThirdPartyAccountRequest message header.

Table 1-43 Parameters in a QueryThirdPartyAccountRequest message header

Parameter	Type	Length	Level of Requirement	Description
traceUniqueID	xsd:string	30	Mandatory	Transaction ID. The ID is automatically generated by the SDP and is used only to trace messages during the SDP commissioning. The App ignores this parameter.



1 SOAP APIs

				[Example] 100001200101110623021721000011
--	--	--	--	---

Message Body Parameters

Table 1-44 describes parameters in a QueryThirdPartyAccountRequest message body.

Table 1-44 Parameters in a QueryThirdPartyAccountRequest message body

Parameter	Type	Length	Level of Requirement	Description
serviceld	xsd: string	4	Mandatory	Holds the identification number to identify the service request. This is set as 100.
parameter	Parameter[0..20]	-	Optional	Holds the request parameter details in the form of name/value pair to identify the service details. Valid parameter names are: <ul style="list-style-type: none"> ProcessingNumber SenderId AcctRef PrefLang OpCoIDFor the detailed parameter name information, see Table 1-46. The parameter parameter is of the Parameter[0..20] type and contains multiple sub-parameters. For details about the Parameter[0..20] type, see Table 1-45.

Table 1-45 describes the parameter structure of the Parameter[0..20] type.

Table 1-45 Parameter structure of the Parameter[0..20] type

Parameter	Type	Length	Level of Requirement	Description
name	xsd:string	-	Mandatory	Holds the parameter name.
value	xsd:string	-	Mandatory	Holds the parameter value.

Table 1-46 describes the parameter lists.



1 SOAP APIs

Table 1-46 Parameter name lists

name	value			
Parameter	Type	Length	Level of Requirement	Description
ProcessingNumber	xsd:string	140	Optional	An Id to uniquely identify the transaction on the App system.
SenderID	xsd:string	140	Optional	An Id to identify sender. This is set as MOM.
AcctRef	xsd:string	140	Mandatory	Subscriber's account number on the App system. If the subscriber does not have account on the App system, you need confirm it with the Mobile Money system. The values send to SP will be in format like: <ul style="list-style-type: none"> • FRI:46733491234/MSISDN • FRI:12345ABC@water.company/SP
PrefLang	xsd:string	140	Optional	The language locale to be used.
OpCoID	xsd:string	140	Optional	Identifier of the country local to be used. Please refer to 3.3 OpCoID List.

1.5.4 Response for QueryThirdPartyAccount

The App functions as the server, processes QueryThirdPartyAccountRequest messages received from the SDP, and sends QueryThirdPartyAccountResponse messages to the SDP.

This topic provides a success response example. If a request fails, the SDP sends an error response that contains an error code. For details about error responses, see 1.7 API Error Responses.

Example

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)
```

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:b2b="http://b2b.mobilemoney.mtn.zm_v1.0/">
  <soapenv:Header/>
```



1 SOAP APIs

```

<soapenv:Body>
  <b2b:processRequestResponse>
    <return>
      <name>DueAmount</name>
      <value>121212</value>
    </return>
    <return>
      <name>BalanceAmount</name>
      <value>2121</value>
    </return>
    <return>
      <name>MinDueAmount</name>
      <value>121212</value>
    </return>
    <return>
      <name>Narration</name>
      <value>121212</value>
    </return>
    <return>
      <name>ProcessingNumber</name>
      <value>121212</value>
    </return>
    <return>
      <name>StatusCode</name>
      <value>121212</value>
    </return>
    <return>
      <name>ThirdPartyAcctRef</name>
      <value>121212</value>
    </return>
    <return>
      <name>StatusDesc</name>
      <value>121212</value>
    </return>
    <return>
      <name>CurrCode</name>
      <value>USD</value>
    </return>
  </b2b:processRequestResponse>
</soapenv:Body>
</soapenv:Envelope>

```

Message HTTP Header Parameters

Table 1-47 Describes parameters in a QueryThirdPartyAccountResponse message HTTP header.



1 SOAP APIs

Table 1-47 Parameters in a QueryThirdPartyAccountResponse message HTTP header

Header Field	Parameter	Type	Length	Level of Requirement	Description
Signature/		String		Conditional	Signature. The field indicates the summary when the Partner uses HMAC security mode. [Example] 43AD232FD45FF

Message Body Parameters

Table 1-48 describes parameters in a QueryThirdPartyAccountResponse message body.

Table 1-48 Parameters in a QueryThirdPartyAccountResponse message body

Parameter	Type	Length	Level of Requirement	Description
return	Return[0..20]	-	Mandatory	<p>Holds the response parameter details in the form of name/value pair to identify the service details.</p> <p>Valid parameter names are:</p> <ul style="list-style-type: none"> • DueAmount • BalanceAmount • MinDueAmount • Narration • ProcessingNumber • StatusCode • ThirdPartyAcctRef • StatusDesc • CurrCode <p>For the detailed parameter name information, see Table 1-50.</p> <p>The return parameter is of the Return[0..20] type and contains multiple sub-parameters. For details about the Return[0..20] type, see Table 1-49.</p>

Table 1-49 describes the parameter structure of the Return[0..20] type.



1 SOAP APIs

Table 1-49 Parameter structure of the Return[0..20] type

Parameter	Type	Length	Level of Requirement	Description
name	xsd:string	-	Mandatory	Holds the parameter name.
value	xsd:string	-	Mandatory	Holds the parameter value.

Table 1-50 describes the return name lists.

Table 1-50 Return name lists

name	value			
Parameter	Type	Length	Level of Requirement	Description
DueAmount	decimal	4	Mandatory	The amount that the subscriber is due to pay on the App system.
BalanceAmount	xsd:int	4	Optional	Subscriber's balance on the App system. If there is no subscriber's balance on the App system, you need confirm it with the Mobile Money system.
MinDueAmount	xsd:string	4	Optional	The minimum amount that the subscriber is due to pay on the App system. If there is no minimum due on App system, you need confirm it with the Mobile Money system.
Narration	xsd:string	140	Optional	Any text that the vendor would like to display to the subscriber.
ProcessingNumber	xsd:string	140	Optional	The identifier of the transaction echoed back.
StatusCode	xsd:string	2	Optional	A predetermined code to determine the status of the transaction.
ThirdPartyAcctRef	xsd:string	140	Optional	Subscriber's account number on the App system. The SDP will fill this parameter as the same value which was received from AcctRef in the request.
StatusDesc	xsd:string	140	Optional	The App system supplied description of the response code.



1 SOAP APIs

name	value			
Parameter	Type	Length	Level of Requirement	Description
CurrCode	xsd:string	3	Optional	The currency, as an ISO 4217 formatted string.

1.5.5 Error Codes for QueryThirdPartyAccount

The App returns error codes to the SDP when an exception occurs in response to QueryThirdPartyAccountRequest messages. The error codes are defined by partners. The following table describes QueryThirdPartyAccount error codes that the partners may return upon an exception.

Table 1-51 QueryThirdPartyAccount error codes

Error Code	Description
01	Successfully processed transaction.
02	Invalid account details.
03	Transaction not allowed.
04	Payment below minimum.
05	Invalid Fundamo userid/password.
06	Payment Entry not found.
07	Missing transaction/message ID.
08	Missing application version.
100	General failure.
101	Field length exceeded.
102	Subscriber could not be identified. (Invalid subscriberID).
103	Duplicate transaction id.

1.6 APIs for notifying requestPaymentCompleted

1.6.1 Process

Figure 1-6 shows the process of notifying requestPayment completed.



1 SOAP APIs

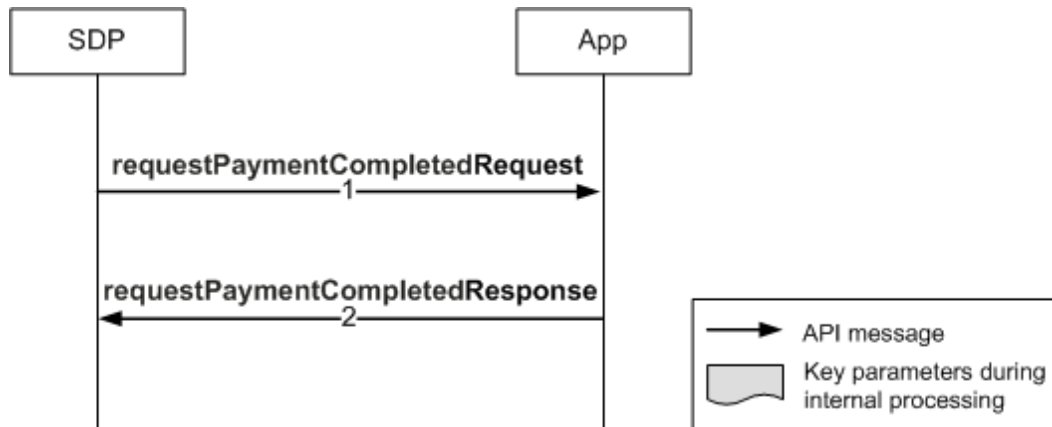
Figure 1-6 Process of notifying requestPayment completed

Table 1-52 describes the process.

Table 1-52 Description for the process of notifying requestPayment completed

Step	Description
1	The SDP sends requestPaymentCompleted request to inform the App the result of the requestPayment in case asynchronism requestPayment.
2	The APP processes the requestPaymentCompleted request and sends a response to the SDP.

1.6.2 requestPaymentCompleted

Function

The SDP invokes the requestPaymentCompleted API to inform the App the result of the requestPayment. Partners must code the App based on the API field requirements so that the App can correctly parse and respond to requests received from the SDP. The App must send a response to the SDP within 30 seconds.

Request URI

The request URI is the destination URI of requestPaymentCompleted messages sent by the SDP to the App. The URI is defined by the App.

1.6.3 Request for requestPaymentCompleted

The SDP functions as the client and sends a requestPaymentCompleted request message to the App to inform the App the result of the requestPayment.



1 SOAP APIs

Example

```
POST / HTTP/1.1
SOAPAction: ""
Content-Type: text/xml; charset=UTF-8
Host: 10.135.163.33:17309
Connection: close
Content-Length: 880

<?xml version="1.0" encoding="utf-8" ?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
    <ns1:NotifySOAPHeader xmlns:ns1="http://www.huawei.com.cn/schema/common/v2_1">
      <ns2:traceUniqueID
xmlns:ns2="http://www.csapi.org/schema/momopayment/local/v1_0">504021503411410281818220013006</n
s2:traceUniqueID>
    </ns1:NotifySOAPHeader>
  </soapenv:Header>
  <soapenv:Body>
    <ns3:requestPaymentCompleted
xmlns:ns3="http://www.csapi.org/schema/momopayment/local/v1_0">
      <ns3:ProcessingNumber>2713500010003</ns3:ProcessingNumber>
      <ns3:MOMTransactionID>2713500010002</ns3:MOMTransactionID>
      <ns3:StatusCode>01</ns3:StatusCode>
      <ns3:StatusDesc>This is a response Message!</ns3:StatusDesc>
      <ns3:ThirdPartyAcctRef>http://www.qwe.com</ns3:ThirdPartyAcctRef>
    </ns3:requestPaymentCompleted>
  </soapenv:Body>
</soapenv:Envelope>
```

Message Header Parameters

Table 1-53 describes parameters in a requestPaymentCompleted message header.



1 SOAP APIs

Table 1-53 Parameters in a requestPaymentCompleted message header

Parameter	Type	Length	Level of Requirement	Description
traceUniqueID	xsd:string	30	Mandatory	Transaction ID. The ID is automatically generated by the SDP and is used only to trace messages during the SDP commissioning. The App ignores this parameter. [Example] 100001200101110623021721000011

Message Body Parameters

Table 1-54 describes parameters in a requestPaymentCompleted request message body.

Table 1-54 Parameters in a requestPaymentCompleted request message body

Parameter	Type	Length	Level of Requirement	Description
ProcessingNumber	xsd: string	140	Mandatory	"ProcessingNumber" which came in requestPayment request.
MOMTransactionID	xsd: string	140	Mandatory	An Id to uniquely identify the transaction on the Mobile Money system.
StatusCode	xsd: string	10	Mandatory	A predetermined code to determine the status of the transaction.
StatusDesc	xsd: string	140	Optional	The Mobile Money system supplied description of the response code.
ThirdPartyAcctRef	xsd: string	140	Mandatory	"AcctRef" which came in requestPayment request.
extensionInfo	NamedParameterList	-	Optional	Extended parameter. Reversed, not used now.

Table 1-55 describes the parameter structure of the NamedParameterList type.

Table 1-55 Parameter structure of the NamedParameterList type

Parameter	Type	Length	Level of Requirement	Description
item	NamedParam	-	Optional	List of extended parameters.



1 SOAP APIs

Parameter	Type	Length	Level of Requirement	Description
	eter[]			

Table 1-56 describes the parameter structure of the NamedParameter type.

Table 1-56 Parameter structure of the NamedParameter type

Parameter	Type	Length	Level of Requirement	Description
key	xsd:string	100	Mandatory	Name of the extended parameter.
value	xsd:string	512	Mandatory	Value of the extended parameter.

1.6.4 Response for requestPaymentCompleted

The App functions as the server, processes requestPaymentCompleted messages received from the SDP, and sends requestPaymentCompletedResponse messages to the SDP.

This topic provides a success response example. If a request fails, the SDP sends an error response that contains an error code. For details about error responses, see 1.7 “API Error Responses.”

Example

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 1142
SOAPAction: ""

<?xml version="1.0" encoding="utf-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <requestPaymentCompletedResponse
xmlns="http://www.csapi.org/schema/momopayment/local/v1_0">
      <result>
        <resultCode xmlns="">00000000</resultCode>
        <resultDescription xmlns="">success</resultDescription>
      </result>
      <extensionInfo>
        <item xmlns="">
          <key>result</key>
          <value>success</value>
        </item>
      </extensionInfo>
    </requestPaymentCompletedResponse>
  </soapenv:Body>
</soapenv:Envelope>

```



1 SOAP APIs

Message Body Parameters

Table 1-57 describes parameters in a requestPaymentCompletedResponse message body.

Table 1-57 Parameters in a requestPaymentCompletedResponse message body

Parameter	Type	Length	Level of Requirement	Description
result	Result	-	Mandatory	Result
extensionInfo	NamedParameterList	-	Optional	Extended parameter. Reversed, not used now.

Table 1-58 describes the parameter structure of the Result type.

Table 1-58 Parameter structure of the Result type

Parameter	Type	Length	Level of Requirement	Description
resultCode	xsd:string	10	Mandatory	Return code; 00000000 means success.
resultDescription	xsd:string	512	Optional	the description of the resultCode

1.6.5 Error Codes for requestPaymentCompleted

The App returns error codes to the SDP when an exception occurs in response to requestPaymentCompleted messages. The error codes are defined by partners.

Table 1-59 describes requestPaymentCompleted error codes that the partners may return upon an exception.

Table 1-59 requestPaymentCompleted error codes

Error Code	Description
00000000	Success
00000001	No data found
10000001	Authentication failed
10000002	Invalid Parameter:%1. %1 indicates parameter name
10000003	Request Rate Limited
10000099	Service Error



1 SOAP APIs

1.7 API Error Responses

When an error occurs, the server sends an error response to the client. This topic provides an error response example and describes parameters in the response.

Example

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <ns1:processRequestResponse xmlns:ns1="http://b2b.mobilemoney.mtn.zm_v1.0">
      <return>
        <name>ProcessingNumber</name>
        <value>555</value>
      </return>
      <return>
        <name>ThirdPartyAcctRef</name>
        <value>444</value>
      </return>
      <return>
        <name>senderID</name>
        <value>MOM</value>
      </return>
      <return>
        <name>StatusCode</name>
        <value>222</value>
      </return>
      <return>
        <name>StatusDesc</name>
        <value>333</value>
      </return>
      <return>
        <name>MOMTransactionID</name>
        <value>111</value>
      </return>
    </ns1:processRequestResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Message Body Parameters

Table 1-15, Table 1-22, Table 1-37, and Table 1-48 define the fields in an error response sent by the server to the client.

02

HTTP APIs

2.1 Overviews

2.1.1 MT Message Header

Request Message Header Parameters

Table 2-1 describes the parameters in a MT request message header. When the 3rd Party invokes APIs exposed by the SDP, MT Message Header must present. MT Message Header is not available for following APIs: getfinancialresourceinformation, preapprovalcompleted, verifyfinancialresourceinformation, withdraw.

Table 2-1 parameters in a MT request message header

Header Field	Parameter	Type	Length	Level of Requirement	Description
Authorization		String	10	Mandatory	Authentication mode. The SDP uses the WSSE authentication mode, in which the realm and profile parameters are involved. NOTE The Authorization value indicates the policy used by the SDP to perform authentication at the application layer. The SDP authenticates a third-party in WSSE UsernameToken mode. [Example] WSSE
	realm	String	20	Mandatory	Set it to SDP . [Example] SDP
	profile	String	20	Mandatory	SDP access mechanism. Set it to UsernameToken . [Example] UsernameToken



2 HTTP APIs

Header Field	Parameter	Type	Length	Level of Requirement	Description
X-WSSE /		String	30	Mandatory	<p>WSSE authentication flag.</p> <p>To enable the SDP to perform authentication at the application layer in the WSSE UsernameToken mode, set this parameter to UsernameToken.</p> <p>[Example]</p> <p>UsernameToken</p>
	Username	String	21	Mandatory	<p>Partner ID.</p> <p>The ID is automatically allocated by the SDP to partners after successful registration. To obtain the ID:</p> <ul style="list-style-type: none"> • A service Partner and API Partner can log in to the SDP Management Portal and query account information, or log in to the mailbox used for registration and view the email notification received after successful registration. • A Developer can log in to the Developer Portal and query account information, or log in to the mailbox used for registration and view the email notification received after successful registration. <p>[Example]</p> <p>2340110001412</p>
	PasswordDigest	String	100	Mandatory	<p>Authentication key for the SDP to authenticate partners.</p> <p>The value is a character string encrypted. The encryption formula is as follows:</p> <ul style="list-style-type: none"> • SHA-256: PasswordDigest = Base64(SHA-256(Nonce + Created + Password)) • SHA-1: PasswordDigest = Base64(SHA-1(Nonce + Created + Password)) <p>In the formula:</p> <ul style="list-style-type: none"> • Nonce and Created: random number and its generation time. • Password: password for partners to access the SDP. <ul style="list-style-type: none"> – A service Partner and API Partner can obtain the password from the email notification received after successful registration. – A Developer can log in to the Developer Portal, choose Member Center > Account > Registration Information > Invoke Password, and set the password.



2 HTTP APIs

Header Field	Parameter	Type	Length	Level of Requirement	Description
					NOTE To retain features of earlier versions, the SP uses the SHA-1 algorithm in the connection to the SDP, which might cause security risks. [Example] 206D88BB7F3D154B130DD6E1E0B8828B
	Nonce	String	30	Mandatory	Random number generated by SDP when sending a message. [Example] 66C92B11FF8A425FB8D4CCFE0ED9ED1F
	Created	String	20	Mandatory	UTC time when the Nonce value is generated. [Format] yyyy-MM-dd'T'HH:mm:ss'Z' [Example] 2011-03-22T13:47:56Z
X-RequestHeader		String	10	Mandatory	Type of the message sent by a third party. Set it to request . [Example] request
	ServiceId	String	21	Conditional	Service ID. The ID is automatically allocated by the SDP to services after successful release. A service Partner can log in to the SDP Management Portal and query service information for the ID. The serviceId must be contained during invocation of a service interface developed by service partners and other partners, and must not be contained during invocation of a capability interface developed by API partners, other partners, and developers. [Example] 35000001000001
	TransId	String	30	Mandatory	Transaction ID. The ID is automatically generated by the SDP and is used only to trace messages during the SDP commissioning. The App ignores this parameter. [Example] 100001200101110623021721000011



2 HTTP APIs

Header Field	Parameter	Type	Length	Level of Requirement	Description
	LinkId	String	20	Conditional	Service order ID. The ID is automatically generated by the SDP when a user orders a service in the SDP. This parameter is mandatory during service ordering by SMS message. [Example] 12345678901111
	FA	xsd:string	30	Mandatory	Mobile number or the fake ID of the charged party. [Example] <ul style="list-style-type: none"> • Mobile number: 8612312345678 • Fake ID: f-245-11900000007639
Cookie	sessionId	String		Conditional	The session id is used as a cookie in the HTTP header in the requests towards the Mobile Money Manager when the Partner use Login/Logout mode. [Example] default8fcee064690b45faa9f8f6c7e21c5e5a NOTE Cookie is valid only for ECW and not for fundamo.
Msisdn	/	String		Mandatory	A valid unique MSISDN of the user. [Format] Countrycode + MSISDN number [Example] 23415905193050
Signature	/	String		Conditional	Signature. The field indicates the summary when the Partner uses HMAC security mode. [Example] 43AD232FD45FF
X-HW-Extension	/	/	/	Optional	Extension field. [Format] key1=value1;key2=value2
destPartnerId		String	21	Conditional	For A2A scenario, it indicates the identifier of the destination partner.



2 HTTP APIs

Response Message Header Parameters

Table 2-2 describes the parameters in a MT response message header.

Table 2-2 parameters in a MT response message header

Header Field	Parameter	Type	Length	Level of Requirement	Description
WWW-Authenticate	/	String	10	Mandatory	Authentication mode. The SDP uses the WSSE authentication mode, in which the realm and profile parameters are involved. NOTE The Authorization value indicates the policy used by the SDP to perform authentication at the application layer. The SDP authenticates a third-party in WSSE UsernameToken mode. [Example] WSSE
	realm	String	20	Mandatory	Set it to SDP . [Example] SDP
	profile	String	20	Mandatory	SDP access mechanism. Set it to UsernameToken . [Example] UsernameToken
Set-Cookie	sessionid	String		Conditional	The Cookie of the response message when the Partner invokes the Login interface. [Example] default1937332605ff4d27a1108f424ade0715
Signature	/	String		Conditional	Signature. The field indicates the summary when the Partner uses HMAC security mode. [Example] 43AD232FD45FF
X-HW-Extension	/	/	/	Optional	Extension field. [Format] key1=value1;key2=value2



2 HTTP APIs

2.1.2 MO Message Header

Request Message Header Parameters

Table 2-3 describes the parameters in a MO request message header. When the SDP sends notification messages to the 3rd party, MO Message Header will present. MO Message Header is only available for following APIs: getfinancialresourceinformation, preapprovalcompleted, verifyfinancialresourceinformation, withdraw.

Table 2-3 parameters in a MO request message header

Header Field	Parameter	Type	Length	Level of Requirement	Description
Authorization		String	10	Mandatory	Authentication mode. The SDP uses the WSSE authentication mode, in which the realm and profile parameters are involved. NOTE The Authorization value indicates the policy used by the SDP to perform authentication at the application layer. The SDP authenticates a third-party in WSSE UsernameToken mode. [Example] WSSE
	realm	String	20	Mandatory	Set it to SDP . [Example] SDP
	profile	String	20	Mandatory	SDP access mechanism. Set it to UsernameToken . [Example] UsernameToken
X-WSSE /		String	30	Mandatory	WSSE authentication flag. To enable the SDP to perform authentication at the application layer in the WSSE UsernameToken mode, set this parameter to UsernameToken . [Example] UsernameToken
	Username	String	21	Mandatory	Partner ID. The ID is automatically allocated by the SDP to partners after successful registration. To obtain the ID: <ul style="list-style-type: none"> A service Partner and API Partner can log in to the SDP Management Portal and query account information, or log in to the mailbox used for registration and view the email



2 HTTP APIs

Header Field	Parameter	Type	Length	Level of Requirement	Description
					notification received after successful registration. <ul style="list-style-type: none"> A Developer can log in to the Developer Portal and query account information, or log in to the mailbox used for registration and view the email notification received after successful registration. [Example] 2340110001412
	PasswordDigest	String	100	Mandatory	Authentication key for the SDP to authenticate partners. The value is a character string encrypted. The encryption formula is as follows: <ul style="list-style-type: none"> SHA-256: PasswordDigest = Base64(SHA-256(Nonce + Created + Password)) SHA-1: PasswordDigest = Base64(SHA-1(Nonce + Created + Password)) In the formula: <ul style="list-style-type: none"> Nonce and Created: random number and its generation time. Password: password for partners to access the SDP. <ul style="list-style-type: none"> A service Partner and API Partner can obtain the password from the email notification received after successful registration. A Developer can log in to the Developer Portal, choose Member Center > Account > Registration Information > Invoke Password, and set the password. NOTE To retain features of earlier versions, the SP uses the SHA-1 algorithm in the connection to the SDP, which might cause security risks. [Example] 206D88BB7F3D154B130DD6E1E0B8828B
	Nonce	String	30	Mandatory	Random number generated by SDP when sending a message. [Example] 66C92B11FF8A425FB8D4CCFE0ED9ED1F
	Created	String	20	Mandatory	UTC time when the Nonce value is generated. [Format]



2 HTTP APIs

Header Field	Parameter	Type	Length	Level of Requirement	Description
					yyyy-MM-dd'T'HH:mm:ss'Z' [Example] 2011-03-22T13:47:56Z
X-NotifyHeader		String	10	Mandatory	Type of the message sent by the SDP. Set it to notify . [Example] notify
	TransId	String	30	Mandatory	Transaction ID. The ID is automatically generated by the SDP and is used only to trace messages during the SDP commissioning. The App ignores this parameter. [Example] 100001200101110623021721000011
	LinkId	String	20	Conditional	Service order ID. The ID is automatically generated by the SDP when a user orders a service in the SDP. This parameter is mandatory during service ordering by SMS message. [Example] 12345678901111
Signature		String		Conditional	Signature. The field indicates the summary when the Partner uses HMAC security mode. [Example] 43AD232FD45FF
X-HW-Extension	/	/	/	Optional	Extension field. [Format] key1=value1;key2=value2

Response Message Header Parameters

Table 2-4 describes the parameters in a MO response message header.



2 HTTP APIs

Table 2-4 parameters in a MO response message header

Header Field	Parameter	Type	Length	Level of Requirement	Description
WWW-Authenticate	/	String	10	Mandatory	Authentication mode. The SDP uses the WSSE authentication mode, in which the realm and profile parameters are involved. NOTE The Authorization value indicates the policy used by the SDP to perform authentication at the application layer. The SDP authenticates a third-party in WSSE UsernameToken mode. [Example] WSSE
	realm	String	20	Mandatory	Set it to SDP . [Example] SDP
	profile	String	20	Mandatory	SDP access mechanism. Set it to UsernameToken . [Example] UsernameToken
Signature	/	String		Conditional	Signature. The field indicates the summary when the Partner uses HMAC security mode. [Example] 43AD232FD45FF
X-HW-Extension	/	/	/	Optional	Extension field. [Format] key1=value1;key2=value2



2 HTTP APIs

2.1.3 MO Message Header for A2A Scenario

For A2A scenario, when the 3rd Party invokes APIs exposed by the SDP, these headers must present. These headers are only available for following APIs:

validateaccountholder, getfinancialresourceinformation, verifyfinancialresourceinformation, preapprovalcompleted, withdraw

Request Message Header Parameters

Parameter	Type	Length	Level of Requirement	Description
version	String	5	Mandatory	Protocol version number. Set this parameter to 1.0 .
traceUniqueId	String	30	Mandatory	<p>Message ID.</p> <p>Value of traceUniqueId is in the format AAAAAAXXXBBBBBBBBBBBCCCCDDDD, which consists of 30 digits.</p> <ul style="list-style-type: none"> • AAAAAA indicates the type of the first component or module that generates traceUniqueId. The value of AAAAAA is the same as the ID of the product module. Prefix the module ID of less than 7 digits with zeros. • XXXX: identifies a node of the component that generates traceUniqueId if the component is deployed in a cluster. The value is the sum of the four octets of the IP address for the node. Assume that the IP address of a node is 192.168.10.2. The value of XXXX is 0372. • BBBBBBBBBBBB: dynamically generated by the first component that generates traceUniqueId. The entry component must ensure that the value of traceUniqueId is unique in different messages. It is recommended that BBBBBBBBBBBB should be generated as YYMMDDHHMMSS, where YY indicates the year, such as 09; the first MM indicates the month, such as 08; DD indicates the day, such as 31; HH indicates the hour, such as 13; the second MM indicates the minute, such as 31; SS indicates the second, such as 31. • CCCC: indicates the serial number that begins with 0001. The component that generates traceUniqueId needs to check whether the first 23 digits of a service process are the same. If yes, the component distinguishes the service process by using CCCC. The value of CCCC ranges from 0001 to 9999 (assume that a maximum of 9999 service



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
				operations can be performed in a second). <ul style="list-style-type: none"> • <i>DDD</i>: indicates the serial number of the message. The value of <i>DDD</i> in traceUniqueID generated by the first component is 001. Otherwise, increase the value of <i>DDD</i> in traceUniqueID by 1.
sourceDeviceType	Int4	4	Mandatory	SP type. This parameter has a fixed value of 110000 .
sourceDeviceCode	String	21	Mandatory	SP ID.
destDeviceType	Int4	4	Mandatory	SAG type. This parameter has a fixed value of 1000 .
destDeviceCode	String	21	Mandatory	SAG ID. This parameter has a fixed value of 1000 .
authenticationSource	String	256	Optional	Access password of the SP. The password encryption formula is as follows: MD5(SPID+PASSWORD+TIMESTAMP)
timestamp	String	14	Optional	Time stamp in the format YYYYMMDDHHMMSS. The time must be UTC time.
destinationPartnerId	String	10	Mandatory	For an A2A scenario, it indicates the identifier of the destination partner.
extension	String	100	Optional	Extended field. [Format] key1=value1;key2=value2



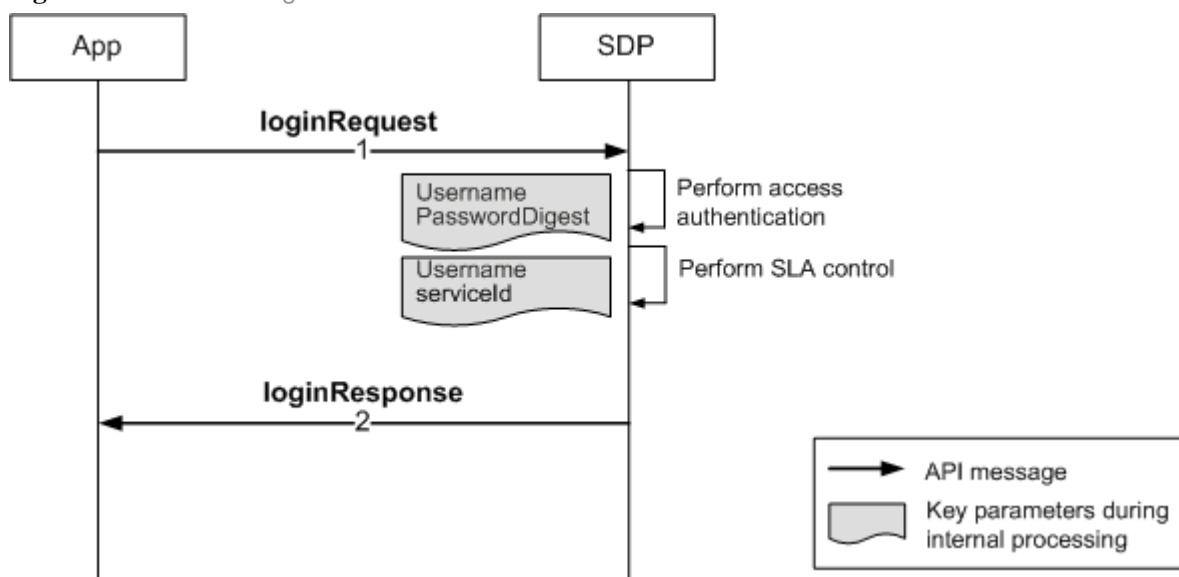
2 HTTP APIs

2.2 Account Validation & Information

2.2.1 login

Process

Figure 2-1 Process of login



Function

Identifying oneself to the system by entering one's credential.

Request URI

The request URI is the destination URI of login messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/login

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/login HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest="weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==",
Nonce="WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId="200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"

```



2 HTTP APIs

```
Signature:43AD232FD45FF
Cookie: sessionid=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 8611111111
X-HW-Extension: k1=v1;k2=v2
```

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:loginrequest xmlns:tns="http://www.ericsson.com/lwac"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<identity>Spl</identity>
<credential>
  <secret>Abc2342!</secret>
  <type>password</type>
</credential>
</tns:loginrequest>
```

Message Body Parameters

Table 2-5 describes parameters in a message body.

Table 2-5 Parameters in a message body

Parameter	Type	Length	Level of Requirement	Description
identity	string		Mandatory	Identity of user.
credential	credential		Mandatory	Credential used to verify the users identity.

Table 2-6 describes parameters of credential.

Table 2-6 Parameters of credential

Parameter	Type	Length	Level of Requirement	Description
secret	string		Mandatory	Credential secret
type	string		Mandatory	Credential type

Response

An empty response indicates success.

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)
```

Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.



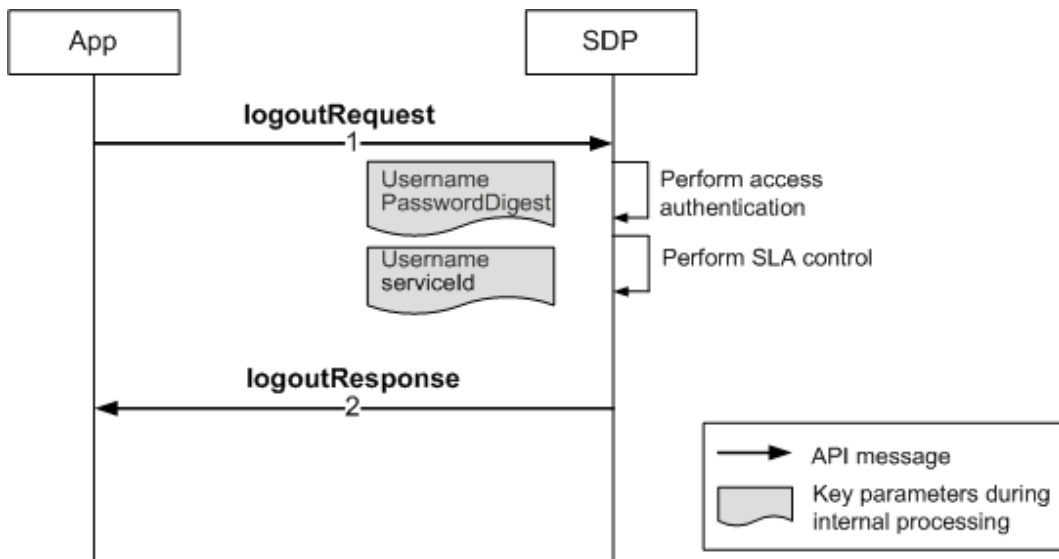
2 HTTP APIs

2.2.2 Logout

Process

Figure 2-12 shows the process of logout request.

Figure 2-2 Process of logout request



Function

The APP (functioning as the client) invokes the Logout API to log out to the SDP (functioning as the server).

Request URI

The request URI is the destination URI of Logout messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/logout

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request

The SDP functions as the client and sends a Logout message to the App.

Message Body Example

The message body is empty.

```

POST /mom/mt/logout HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
  
```



2 HTTP APIs

```
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
Cookie: sessionid=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 861111111
X-HW-Extension: k1=v1;k2=v2
```

Message Body Parameters

None

Response

An empty response indicates success.

A success response example:

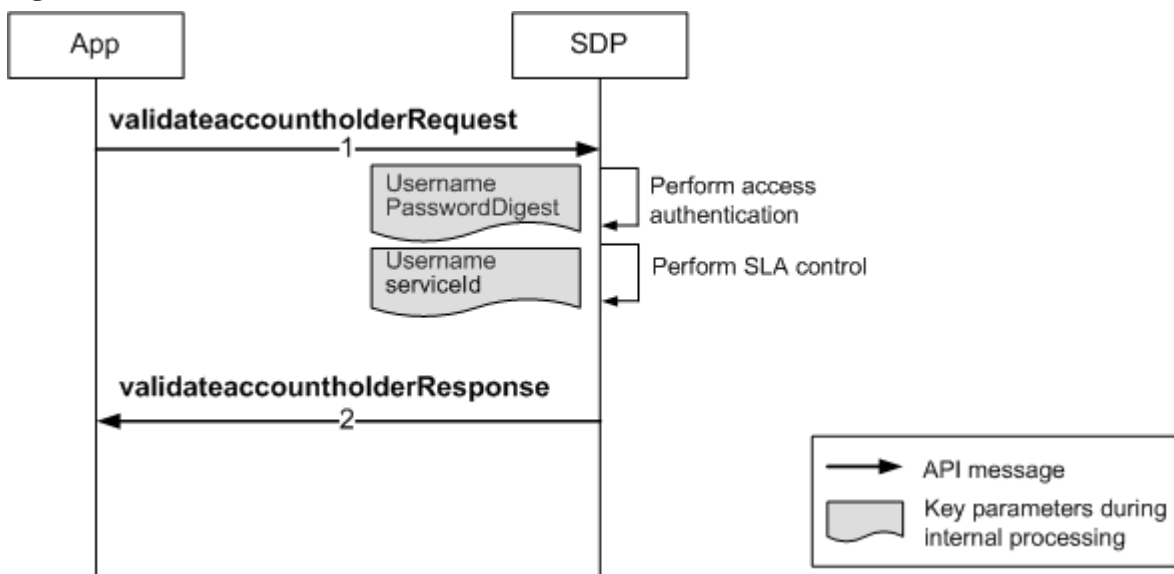
```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)
```

Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.



2 HTTP APIs

2.2.3 validateaccountholder*Process***Figure 2-3** Process of validateaccountholder*Function*

This operation is used by the Partner to check if an account holder is registered and active in the system.

Request URI

The request URI is the destination URI of `validateaccountholder` messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/validateaccountholder

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/validateaccountholder HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
  
```



2 HTTP APIs

```
Cookie: sessionId=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 8611111111
X-HW-Extension: k1=v1;k2=v2
```

```
<ns2:validateaccountholderrequest xmlns:ns2="http://www.ericsson.com/em/emm/sp/backend">
  <accountholderid>ID:20200009999/MSISDN</accountholderid>
</ns2:validateaccountholderrequest>
```

Message Body Parameters

Table 2-7 describes parameters in a validateaccountholder message body.

Table 2-7 Parameters in a validateaccountholder message body

Parameter	Type	Length	Level of Requirement	Description
accountholderid	string		Mandatory	The identity of the account holder to validate. Parameter cannot be NULL. [Example] ID:46733491234/MSISDN.

Response

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)

<?xml version="1.0" encoding="UTF-8"?>
<ns2:validateaccountholderresponse xmlns:ns2="http://www.ericsson.com/em/emm/sp/backend">
  <valid>true</valid>
</ns2:validateaccountholderresponse>
```

Table 2-8 describes parameters in a response message body.

Table 2-8 Parameters in a validateaccountholder response message body

Parameter	Type	Length	Level of Requirement	Description
valid	boolean		Mandatory	TRUE means that requested identity is an account holder that can receive any transaction. FALSE: Means that identity either not found or it is not able to receive any transaction. Parameter cannot be NULL.



2 HTTP APIs

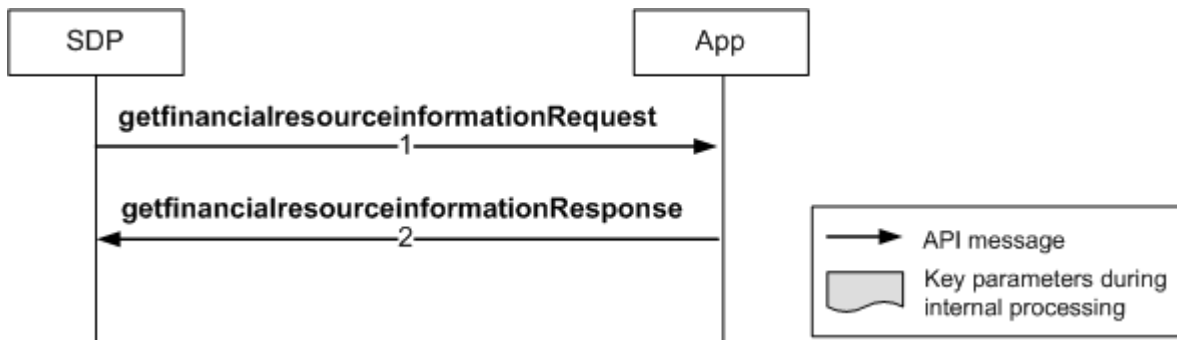
Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.

2.2.4 *getfinancialresourceinformation*

Process

Figure 2-4 Process of *getfinancialresourceinformation*



Function

This operation is used to fetch generic data for a specific resource for a customer in the Partner system. The information returned can be any data, such as loan information, electricity meter status, TV subscription details. This can be used as a complement to the *getbalance* operation, since it only returns a balance. The information in the response is just passed through the system to the end-user requesting it.

Request URI

The request URI is the destination URI of *getfinancialresourceinformation* messages sent by the SDP to the App. The URI is defined by the App.

Request Message Body Example

```

POST /mom/mo/getreversetransactiondetails HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-NotifyHeader: notify TransId=" 200903241230451000000123456789" , LinkId="A15Y89Z32T99"
Signature:43AD232FD45FF
X-HW-Extension: k1=v1;k2=v2

<?xml version="1.0" encoding="UTF-8"?>
<getfinancialresourceinformationrequest xmlns ="http://www.ericsson.com/em/emm/sp/backend" >

```



2 HTTP APIs

```
<resource>123</resource>
<accountholderid>ID:4673349175/MSISDN</accountholderid>
<extension>11</extension>
</getfinancialresourceinformationrequest>
```

Message Body Parameters

Table 2-9 describes parameters in a message body.

Table 2-9 Parameters in a message body

Parameter	Type	Length	Level of Requirement	Description
resource	string		Mandatory	The Financial Resource Identifier of the sending account. Parameter cannot be NULL.
accountholderid	string		Mandatory	The MSISDN identity of the account holder that is performing the operation. For example: ID:46733491234/MSISDN
extension	extensiontype	-	Optional	Used for sending any additional information to Partner. For details about extensiontype, see Table 2-32.

Response

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)

<?xml version="1.0" encoding="UTF-8"?>
<getfinancialresourceinformationresponse xmlns="http://www.ericsson.com/em/emm/sp/backend" >
  <message>11</message>
  <extension>11</extension>
</getfinancialresourceinformationresponse>
```

Table 2-40 describes parameters in a response message body.



2 HTTP APIs

Table 2-10 Parameters in a response message body

Parameter	Type	Length	Level of Requirement	Description
message	string	-	Optional	The list of sub transactions for the requested transaction. Contains information describing a single financial instruction For more details about financialinstructiondetails, see Table 2-38.
extension	extensiontype		Optional	Any additional information received from Partner. For details about extensiontype, see Table 2-32.

Error Codes

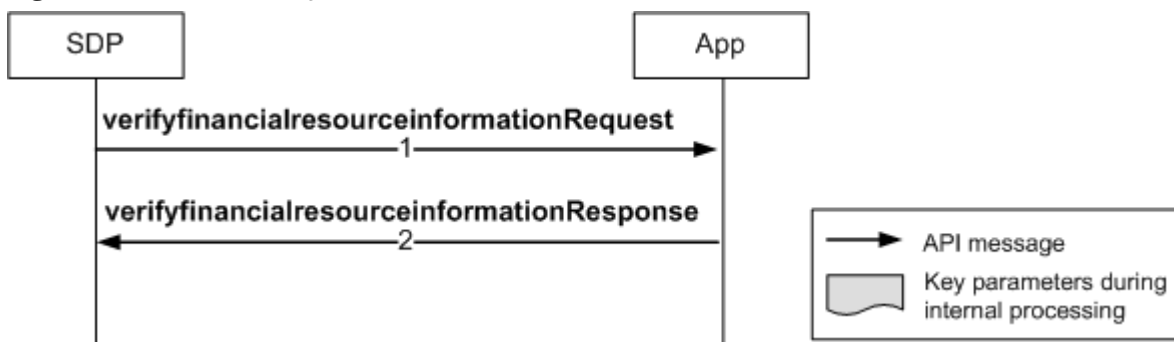
The App returns error codes to the SDP when an exception occurs in response to getfinancialresourceinformation messages. The error codes are defined by partners. Table 2-11 describes getfinancialresourceinformation error codes that the partners may return upon an exception.

Table 2-11 getfinancialresourceinformation error codes

Error Code	Description
01	Successfully processed transaction.
02	Invalid account details.
03	Transaction not allowed.
04	Payment below minimum.
05	Invalid Mobile Money platform userid/password.
06	Payment Entry not found.
07	Missing transaction/message ID.
08	Missing application version.
100	General failure.
101	Field length exceeded.
102	Subscriber could not be identified. (Invalid subscriberID).
103	Duplicate transaction id.



2 HTTP APIs

2.2.5 verifyfinancialresourceinformation*Process***Figure 2-5** Process of verifyfinancialresourceinformation*Function*

This operation is used to ask the Partner if the account in their system may be linked to the account holders eWallet in the Mobile Money Manager. This is primarily relevant for micro finance services, like loans and savings accounts. Once the Partner confirms the link, this external account is added to the account holder in system, and they can start using this account through operations such as **BalanceRequest** and **Withdrawrequest**.

Request URI

The request URI is the destination URI of verifyfinancialresourceinformation messages sent by the SDP to the App. The URI is defined by the App.

Request Message Body Example

```

POST /mom/mo/getreversetransactiondetails HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNvksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-NotifyHeader: notify TransId=" 200903241230451000000123456789" , LinkId="A15Y89Z32T99"
Signature:43AD232FD45FF
X-HW-Extension: k1=v1;k2=v2

<?xml version="1.0" encoding="UTF-8"?>
<ns2:verifyfinancialresourceinformationrequest
xmlns:ns2="http://www.ericsson.com/em/emm/sp/backend">
<resource>FRI:20200009999@savings.sp1/SP</resource>
<accountholderid>ID:20200009999/MSISDN</accountholderid>
</ns2:verifyfinancialresourceinformationrequest>
  
```



2 HTTP APIs

Message Body Parameters

Table 2-12 describes parameters in a message body.

Table 2-12 Parameters in a message body

Parameter	Type	Length	Level of Requirement	Description
resource	string		Mandatory	The FRI of the account, bill or user in the partner system. Parameter cannot be NULL. For example, FRI:12345ABC@water.company/SP.
accountholderid	string		Mandatory	The MSISDN identity of the account holder. Parameter cannot be NULL. For example: ID:46733491234/MSISDN
extension	extensiontype	-	Optional	Used for sending any additional information to Partner. For details about extensiontype, see Table 2-32.

Response

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)

<?xml version="1.0" encoding="UTF-8"?>
<ns4:verifyfinancialresourceinformationresponse
xmlns:ns4="http://www.ericsson.com/em/emm/sp/backend">
  <valid>true</valid>
  <extension/>
</ns4:verifyfinancialresourceinformationresponse>
<
```

Table 2-13 describes parameters in a response message body.



2 HTTP APIs

Table 2-13 Parameters in a response message body

Parameter	Type	Length	Level of Requirement	Description
valid	boolean		Mandatory	True if account holder and service resource is valid in Partner system. Parameter cannot be NULL.
extension	extensiontype		Optional	Any additional information received from Partner. For details about extensiontype, see Table 2-32.

Error Codes

The App returns error codes to the SDP when an exception occurs in response to `verifyfinancialresourceinformation` messages. The error codes are defined by partners. Table 2-14 describes `verifyfinancialresourceinformation` error codes that the partners may return upon an exception.

Table 2-14 `verifyfinancialresourceinformation` error codes

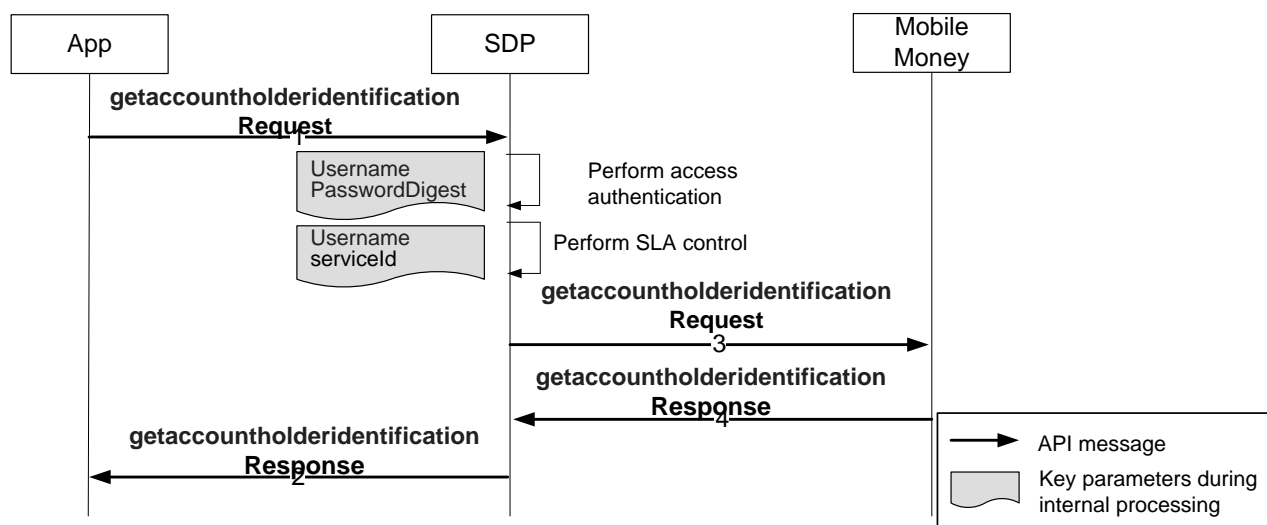
Error Code	Description
01	Successfully processed transaction.
02	Invalid account details.
03	Transaction not allowed.
04	Payment below minimum.
05	Invalid Mobile Money platform userid/password.
06	Payment Entry not found.
07	Missing transaction/message ID.
08	Missing application version.
100	General failure.
101	Field length exceeded.
102	Subscriber could not be identified. (Invalid subscriberID).
103	Duplicate transaction id.



2 HTTP APIs

2.2.6 *getaccountholderidentification*

Process

Figure 2-6 Process of *getaccountholderidentification*

Function

This operation is used by the Partner to query the account holder's identification.

Request URI

The request URI is the destination URI of *getaccountholderidentification* messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/getaccountholderidentification

In the format, IP and Port indicate the service IP address and HTTP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/getaccountholderidentification HTTP/1.1
Content-Type:text/xml
Content-Length:569
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
Cookie: sessionId=default8fcee064690b45faa9f8f6c7e21c5e5a
  
```



2 HTTP APIs

```

Msisdn: 861111111
X-HW-Extension: k1=v1;k2=v2

<ns2:getaccountholderidentificationrequest version="1.0"
xmlns:ns2="http://www.ericsson.com/em/emm/provisioning/v1_0">
  <identity>20200009999</identity>
</ns2:getaccountholderidentificationrequest>

```

Message Body Parameters

Table 2-15 describes parameters in a getaccountholderidentification message body.

Table 2-15 Parameters in a getaccountholderidentification message body

Parameter	Type	Length	Level of Requirement	Description
identity	String		Optional	A valid identity associated with the account holder. Validated with IsIdentity. This shall be user's MSISDN.
version	String		Mandatory	Default value 1.0

Response

A success response example:

```

<?xml version="1.0" encoding="UTF-8"?>
<n1:getaccountholderidentificationresponse
xmlns:n1="http://www.ericsson.com/em/emm/provisioning/v1_0">
  <identifications>
    <identification>
      <Id>a</Id>
      <IdTp>PASS</IdTp>
    </identification>
    <identification>
      <Id>a</Id>
      <IdTp>PASS</IdTp>
    </identification>
  </identifications>
</n1:getaccountholderidentificationresponse>

```

Table 2-16 describes parameters in a response message body.



2 HTTP APIs

Table 2-16 Parameters in a getaccountholderidentification response message body

Parameter	Type	Length	Level of Requirement	Description
identifications	Identification-[0...unbounded]		Optional	

Table 2-17 describes parameters in an Identification.

Table 2-17 Parameters in Identification structure

Parameter	Type	Length	Level of Requirement	Description
Id	xsd:string	35	Mandatory	
IdTp	xsd:string	4	Optional	Possible values are: PASS CPFA SRSA NRIN OTHR DRLC SOCS AREG IDCD EMID
XtndedIdTp	xsd:string	350	Optional	Either IdTp or XtndedIdTp will present.

Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.

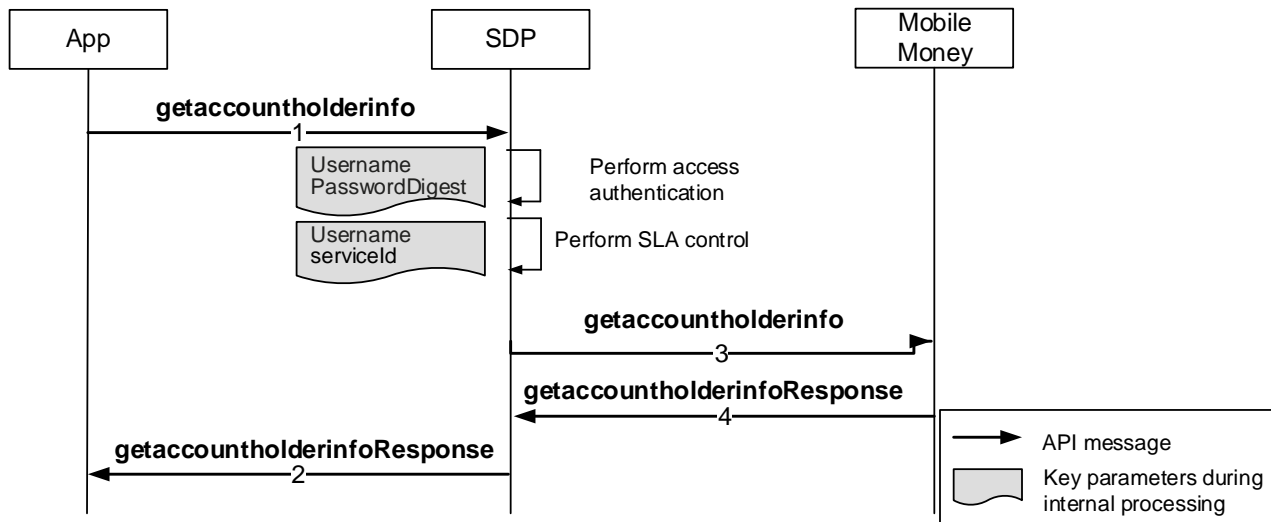
2.2.7 getaccountholderinfo

Process

Figure 2-7 Process of getaccountholderinfo



2 HTTP APIs

*Function*

This operation is used by the Partner to query the account holder's information.

Request URI

The request URI is the destination URI of `getaccountholderinfo` messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/getaccountholderinfo

In the format, IP and Port indicate the service IP address and HTTP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/getaccountholderinfo HTTP/1.1
Content-Type:text/xml
Content-Length:569
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
Cookie: sessionId=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 861111111
X-HW-Extension: k1=v1;k2=v2

<ns2:getaccountholderinfo request version="1.0"
xmlns:ns2="http://www.ericsson.com/em/emm/provisioning/v1_1">
  <identity>20200009999</identity>

```



2 HTTP APIs

```
</ns2:getaccountholderinforequest >
```

Message Body Parameters

Table 2-18 describes parameters in a getaccountholderinfo message body.

Table 2-18 Parameters in a getaccountholderinfo message body

Parameter	Type	Length	Level of Requirement	Description
identity	String		Optional	A valid identity associated with the account holder. Validated with IsIdentity. This shall be user's MSISDN.
version	String		Mandatory	Default value 1.0

Response

A success response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<n1:getaccountholderinforesponse xmlns:n1="http://www.ericsson.com/em/emm/provisioning/v1_0">
  <accountholderbasicinfo>
    <msisdn>20200009999</msisdn>
    <firstname>lili</firstname>
    <surname>Li</surname>
  </accountholderbasicinfo>
</n1:getaccountholderinforesponse>
```

Table 2-19 describes parameters in a response message body.

Table 2-19 Parameters in a getaccountholderinfo response message body

Parameter	Type	Length	Level of Requirement	Description
accountholderbasicinfo	accountholderbasicinfoType	-	Optional	

Table 2-20 describes parameters in the accountholderbasicinfoType.

Table 2-20 Parameters in accountholderbasicinfoType structure

Parameter	Type	Length	Level of Requirement	Description
msisdn	xsd:string	35	Optional	The account holder number.
firstname	xsd:string	30	Optional	The firstname of the account holder



2 HTTP APIs

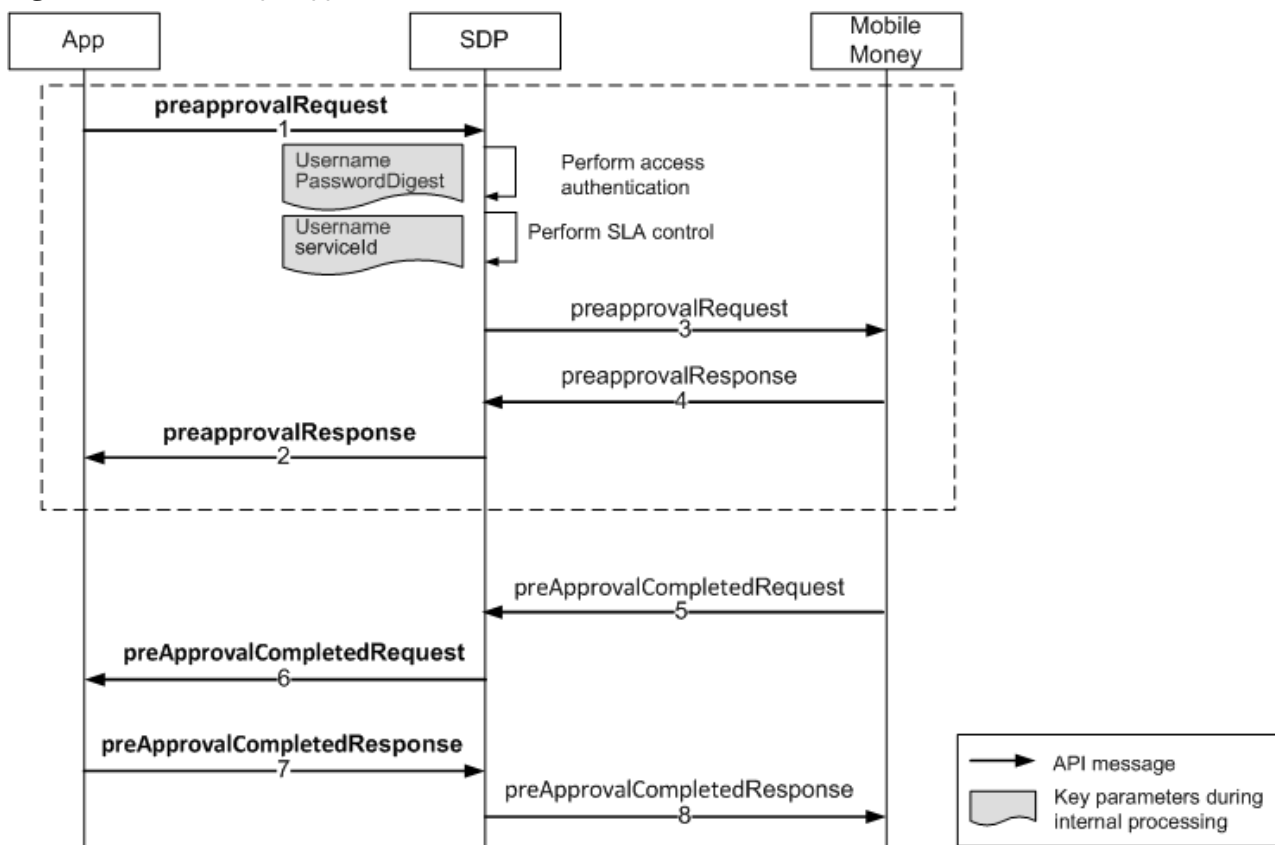
Parameter	Type	Length	Level of Requirement	Description
surname	xsd:string	30	Optional	The surname of the account holder

Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.

2.3 Approval

2.3.1 preapproval

*Process***Figure 2-8** Process of preapproval*Function*

A pre-approval enables a Partner to debit an account holder without the user having to approve each operation separately.



2 HTTP APIs

Request URI

The request URI is the destination URI of preapproval messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/preapproval

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```
POST /mom/mt/preapproval HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
Cookie: sessionid=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 861111111
X-HW-Extension: k1=v1;k2=v2

<ns2:preapprovalrequest xmlns:ns2="http://www.ericsson.com/em/emm/sp/backend">
  <fromfri>ID:2341305205559/MSISDN</fromfri>
  <tofri>ID:2341305205559/MSISDN</tofri>
  <message>abc</message>
</ns2:preapprovalrequest>
```

Message Body Parameters

Table 2-21 describes parameters in a preapproval message body.

Table 2-21 Parameters in a preapproval message body

Parameter	Type	Length	Level of Requirement	Description
fromfri	string		Mandatory	The Financial Resource Identifier of the sending account. Parameter cannot be NULL.
tofri	string		Mandatory	The Financial Resource Identifier of the receiving account. Parameter cannot be NULL.
message	string		Mandatory	Message to the end user.



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
				Parameter cannot be NULL.

Response

A success response example:

```

HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty (7.5.4.v20111024)

<?xml version="1.0" encoding="UTF-8"?>
<preapprovalresponse xmlns="http://www.ericsson.com/em/emm/sp/backend">
  <preapprovalid>123</preapprovalid>
  <status>APPROVED</status>
</preapprovalresponse>

```

Table 2-22 describes parameters in a preapproval response message body.

Table 2-22 Parameters in a preapproval response message body

Parameter	Type	Length	Level of Requirement	Description
preapprovalid	long		Mandatory	The identity of the preapproval. Parameter cannot be NULL.
status	preapproval - status		Mandatory	The status of the preapproval. Parameter cannot be NULL. [Enumeration] <ul style="list-style-type: none"> • PENDING • APPROVED • REJECTED • EXPIRED

Error Codes

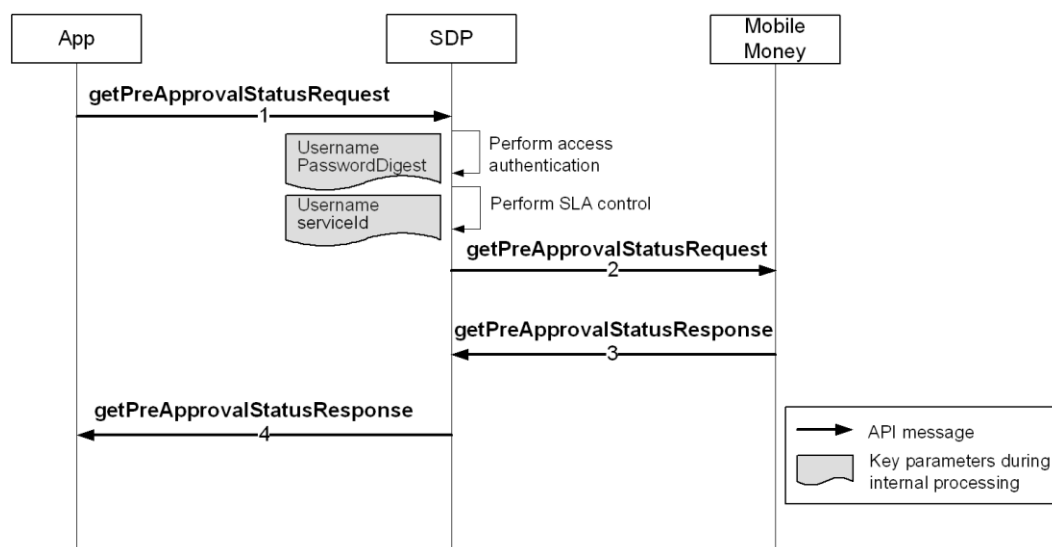
SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.



2 HTTP APIs

2.3.2 *getPreApprovalStatus*

Process

Figure 2-9 Process of *getPreApprovalStatus*

Function

It is used by the Partner to fetch the status of a previously sent *PreApprovalRequest*.

Request URI

The request URI is the destination URI of *getPreApprovalStatus* messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/getPreApprovalStatus

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/getPreApprovalStatus HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
Cookie: sessionId=defaultt8fcee064690b45faa9f8f6c7e21c5e5a
  
```



2 HTTP APIs

```

Msisdn: 861111111
X-HW-Extension: k1=v1;k2=v2

<ns2:getpreapprovalstatusrequest xmlns:ns2="http://www.ericsson.com/em/emm">
  <preapprovalid>12.23</preapprovalid>
  <fromfri>REPORT</fromfri>
  <tofri>SUMMARY</tofri>
</ns2:getpreapprovalstatusrequest>

```

Message Body Parameters

Table 2-23 describes parameters in a message body.

Table 2-23 Parameters in a message body

Parameter	Type	Length	Level of Requirement	Description
preapprovalid	long		Mandatory	ID of the pre-approval request. Parameter cannot be NULL.
fromfri	string		Mandatory	The Financial Resource Identifier of the sending account. Parameter cannot be NULL. Validated with IsFRI.
tofri	string		Mandatory	The Financial Resource Identifier of the receiving account. Parameter cannot be NULL. Validated with IsFRI.

Response

A success response example:

```

HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)

<?xml version="1.0" encoding="UTF-8"?>
<getpreapprovalstatusresponse xmlns="http://www.ericsson.com/em/emm">
  <status>1</status>
</getpreapprovalstatusresponse>

```

Table 2-24 describes parameters in a response message body.



2 HTTP APIs

Table 2-24 Parameters in a response message body

Parameter	Type	Length	Level of Requirement	Description
status	preapproval status	-	Mandatory	Current status of the preapproval. Parameter cannot be NULL. [Enumeration] <ul style="list-style-type: none">• PENDING• APPROVED• REJECTED• EXPIRED

Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.

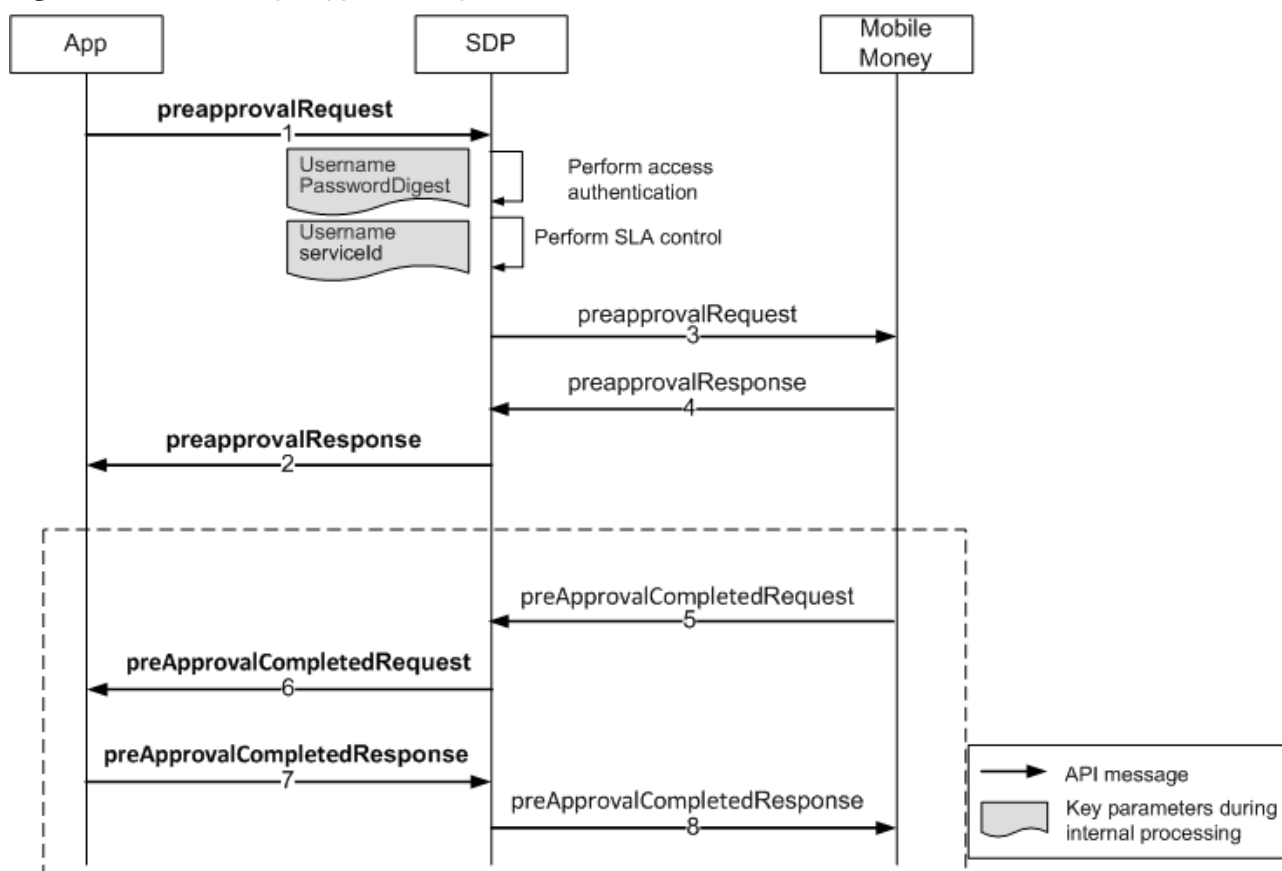


2 HTTP APIs

2.3.3 *preapprovalcompleted*

Process

Figure 2-10 Process of preapprovalcompleted



Function

The Mobile Money Manager will send this request to the Service Provider when the user has confirmed the approval, or if the pre-approval has expired.

Request URI

The request URI is the destination URI of preapprovalcompleted messages sent by the SDP to the App. The URI is defined by the App.

Request Message Body Example

```

POST /mom/mo/getreversetransactiondetails HTTP/1.1
Content-Type:text/xml
Content-Length:1069
  
```



2 HTTP APIs

```
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-NotifyHeader: notify TransId=" 200903241230451000000123456789" , LinkId="A15Y89Z32T99"
Signature:43AD232FD45FF
X-HW-Extension: k1=v1;k2=v2
```

```
<preapprovalcompletedrequest xmlns=" http://www.ericsson.com/em/emm/sp/backend">
  <preapprovalid>1.2</preapprovalid>
  <status>PENDING</status>
  <receiverinfo>
    <fri>123</fri>
    <name>12</name>
    <msisdn>12</msisdn>
    <emailaddresses>
      <email>12</email>
    </emailaddresses>
  </receiverinfo>
</preapprovalcompletedrequest>
```

Message Body Parameters

Table 2-25 describes parameters in a message body.

Table 2-25 Parameters in a message body

Parameter	Type	Length	Level of Requirement	Description
preapprovalid	long		Mandatory	ID of the preapproval. Parameter cannot be NULL.
status	preapprovalstatus		Mandatory	Current status of the preapproval. Parameter cannot be NULL. [Enumeration] <ul style="list-style-type: none"> • PENDING • APPROVED • REJECTED • EXPIRED
receiverinfo	receiverinfo	-	Mandatory	Information about the receiving party. Parameter cannot be NULL. For more details about receiverinfo, see Table 2-26.

Table 2-26 describes parameters in receiverinfo.



2 HTTP APIs

Table 2-26 Parameters in receiverinfo

Parameter	Type	Length	Level of Requirement	Description
fri	string		Mandatory	FRI for receivers account to which the amount was credited.
name	string		Optional	Name of receiving party.
msisdn	string	-	Optional	MSISDN of receiving party.
email	list<emailaddress>		Optional	Email addresses of receiving party. For more details about list<emailaddress>, see Table 2-27.
language	string		Optional	Preferred language code.

Table 2-27 describes parameters in list<emailaddress>.

Table 2-27 Parameters in list<emailaddress>

Parameter	Type	Length	Level of Requirement	Description
address	string	-	Optional	Validated with IsEmail.

Response

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty (7.5.4.v20111024)

<?xml version="1.0" encoding="UTF-8"?>
<preapprovalcompletedresponse xmlns="http://www.ericsson.com/em/emm/sp/backend"/>
```

Table 2-22 describes parameters in a response message body.

Error Codes

The App returns error codes to the SDP when an exception occurs in response to preapprovalcompleted messages. The error codes are defined by partners.

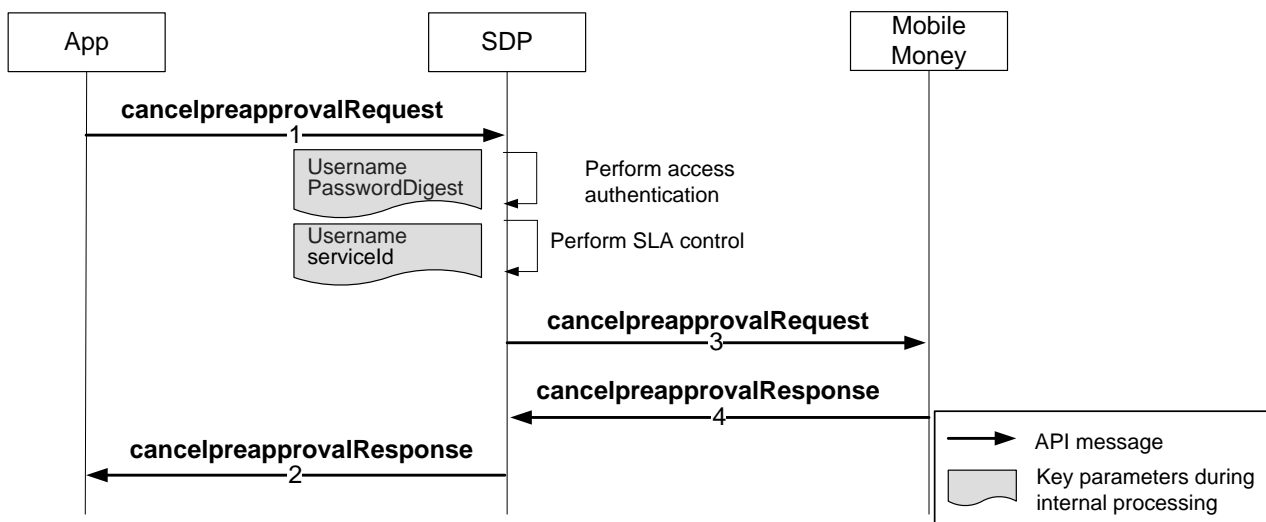
Table 2-28 describes preapprovalcompleted error codes that the partners may return upon an exception.



2 HTTP APIs

Table 2-28 Preapprovalcompleted error codes

Error Code	Description
01	Successfully processed transaction.
02	Invalid account details.
03	Transaction not allowed.
04	Payment below minimum.
05	Invalid Mobile Money platform userid/password.
06	Payment Entry not found.
07	Missing transaction/message ID.
08	Missing application version.
100	General failure.
101	Field length exceeded.
102	Subscriber could not be identified. (Invalid subscriberID).
103	Duplicate transaction id.

2.3.4 cancelpreapproval*Process***Figure 2-11** Process of cancelpreapproval



2 HTTP APIs

Function

The App can invoke this API to cancel one or all pre-approvals for the logged in account holder.

Request URI

The request URI is the destination URI of cancelpreapproval messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/cancelPreApproval

In the format, IP and Port indicate the service IP address and HTTP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```
POST /mom/mt/cancelPreApproval HTTP/1.1
Content-Type:text/xml
Content-Length:569
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-NotifyHeader: notify TransId=" 200903241230451000000123456789" , LinkId="A15Y89Z32T99"
X-HW-Extension: k1=v1;k2=v2
Signature:43AD232FD45FF

<?xml version="1.0" encoding="UTF-8"?>
<fr:cancelpreapprovalrequest
xmlns:fr="http://www.ericsson.com/em/emm/serviceprovider/v1_0/frontend" version="1.0">
  <identity>identity0</identity>
  <fri>fri0</fri>
  <preapprovalid>preapprovalid0</preapprovalid>
</fr:cancelpreapprovalrequest>
```

Message Body Parameters

Table 2-29 describes parameters in a message body.

Table 2-29 Parameters in a message body

Parameter	Type	Length	Level of Requirement	Description
identity	string		Optional	The account holder's internal identity. If set, all pre-approvals for this account holder will be cancelled. Only a logged in user with permission ROLE_CANCEL_ANY_PREAPPROVAL can cancel all pre-approvals for a specific account holder..
fri	string		Optional	The FRI to cancel all pre-approvals from. If no FRI, and no pre-approval identity is supplied, then all



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
				pre-approvals for the logged in user are cancelled.
preapprovalid	string		Optional	Omit to cancel more than one pre-approval.

Response

A success response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<fr:cancelpreapprovalresponse
xmlns:fr="http://www.ericsson.com/em/emm/serviceprovider/v1_0/frontend"/>
```

Error Codes

For details about the error codes, see 2.7 API Error Response.

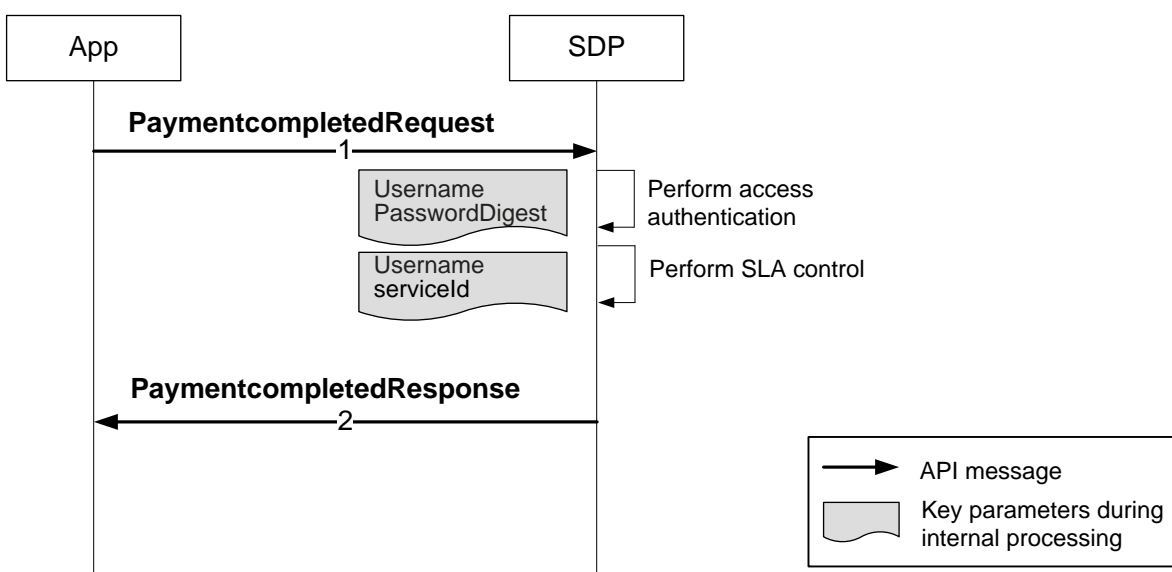
2.4 Transaction Details

2.4.1 Paymentcompleted

Process

Figure 2-12 shows the process of notifying about the completion of a previously initiated payment request.

Figure 2-12 Process of notifying about the completion of a previously initiated payment request





2 HTTP APIs

Function

The APP (functioning as the client) invokes the Paymentcompleted API to notify about the completion of a previously initiated payment request to the SDP (functioning as the server). Partners must code the App based on the API field requirements so that the App can send correct requests to the SDP. The SDP must send a response within 180 seconds by default.

Request URI

The request URI is the destination URI of Paymentcompleted messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/paymentcompleted

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request

The SDP functions as the client and sends a Paymentcompleted message to the App to query account balance.

Message Body Example

```
POST /mom/mt/paymentcompleted HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
Cookie: sessionid=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 861111111
X-HW-Extension: k1=v1;k2=v2

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns4:paymentcompletedrequest xmlns:ns4="http://www.ericsson.com/em/emm/sp/backend">
<transactionid>2110266</transactionid>
<providertransactionid>1351061132147</providertransactionid>
<newbalance>
  <amount>110</amount>
  <currency>RWF</currency>
</newbalance>
<paymenttoken>ABCDEF</paymenttoken>
<message>Your payment has been performed.</message>
<status>COMPLETED</status>
</ns4:paymentcompletedrequest>
```



2 HTTP APIs

Message Body Parameters

Table 2-30 describes parameters in a Paymentcompleted message body.

Table 2-30 Parameters in a Paymentcompleted message body

Parameter	Type	Length	Level of Requirement	Description
transactionid	long		Mandatory	System internal transaction id. Parameter cannot be NULL.
providertransactionid	string		Mandatory	Service provider transaction id.
newbalance	moneydetailstype	-	Optional	Balance of the account at the Partner. For details about moneydetailstype, see Table 2-31.
paymenttoken	string		Optional	Partner payment token.
message	string		Optional	A message to the account holder that made the payment.
status	paymentstatuses	-	Mandatory	The status of the payment. Parameter cannot be NULL. [Enumeration] <ul style="list-style-type: none"> • PENDING • COMPLETED • FAILED
extension	extensiontype	-	Optional	Used for sending any additional information to front end access module. For details about extensiontype, see Table 2-32.

Table 2-31 describes the parameter structure of the moneydetailstype.

Table 2-31 Parameter structure of moneydetailstype

Parameter	Type	Length	Level of Requirement	Description
amount	decimal		Mandatory	The monetary amount. Parameter cannot be NULL.



2 HTTP APIs

currency	string		Mandatory	The currency, as an ISO 4217 formatted string. Parameter cannot be NULL.
----------	--------	--	-----------	--

Table 2-32 describes the extensiontype.

Table 2-32 Parameter extensiontype

name	value			
Parameter	Type	Length	Level of Requirement	Description
any	list<object>	-	Optional	Use to send any other information.

Response

An empty response indicates success.

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: xxx
Server: Jetty(7.5.4.v20111024)

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns4:paymentcompletedresponse xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:op="http://www.ericsson.com/em/emm/common"
xmlns:ns4="http://www.ericsson.com/em/emm/sp/backend"/>
```

Error Codes

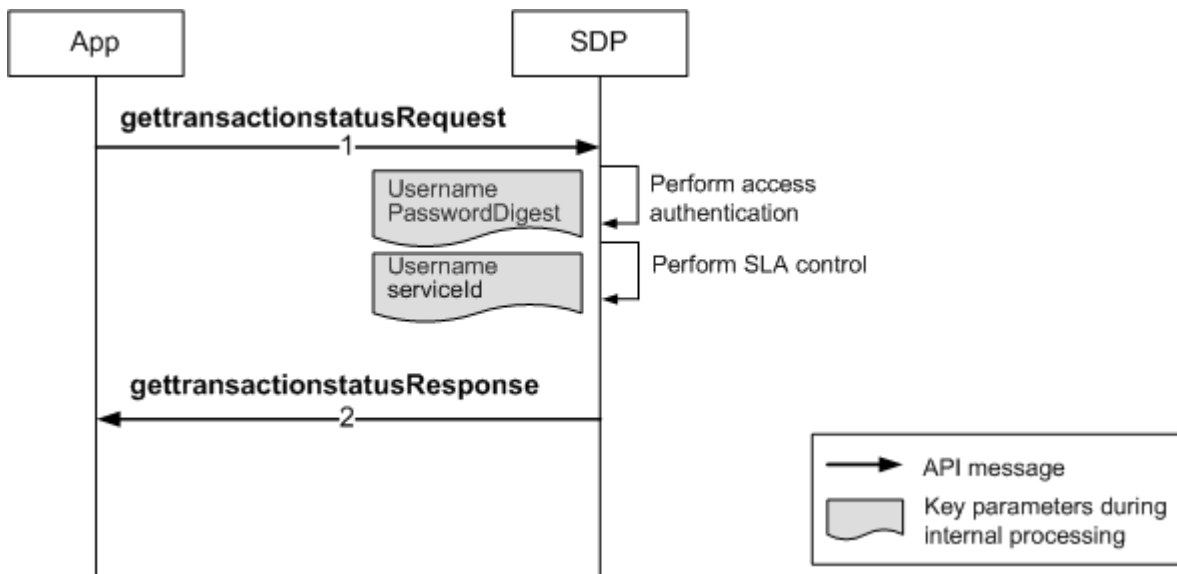
SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.



2 HTTP APIs

2.4.2 *gettransactionstatus*

Process

Figure 2-13 Process of *gettransactionstatus*

Function

This section describes the operation used to check debit status.

Request URI

The request URI is the destination URI of *gettransactionstatus* messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/gettransactionstatus

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/gettransactionstatus HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
  
```



2 HTTP APIs

```
Cookie: sessionId=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 8611111111
X-HW-Extension: k1=v1;k2=v2
```

```
<?xml version="1.0" encoding="UTF-8"?>
<ns2:gettransactionstatusrequest xmlns:ns2="http://www.ericsson.com/em/emm">
  <referenceid>12.23</referenceid>
  <identity>REPORT</identity>
</ns2:gettransactionstatusrequest>
```

Message Body Parameters

Table 2-33 describes parameters in a gettransactionstatus message body.

Table 2-33 Parameters in a gettransactionstatus message body

Parameter	Type	Length	Level of Requirement	Description
referenceid	string		Mandatory	Reference Id which has been entered by user, while initiating the transaction. Parameter cannot be NULL.
identity	string		Optional	A valid identity associated with the account holder that performed the requested transaction.

Response

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)
```

```
<?xml version="1.0" encoding="UTF-8"?>
<n1:gettransactionstatusresponse xmlns:n1="http://www.ericsson.com/em/emm">
  <transactionid>2147483647</transactionid>
  <status>SUCCESSFUL</status>
</n1:gettransactionstatusresponse>
```

Table 2-34 describes parameters in a gettransactionstatus response message body.



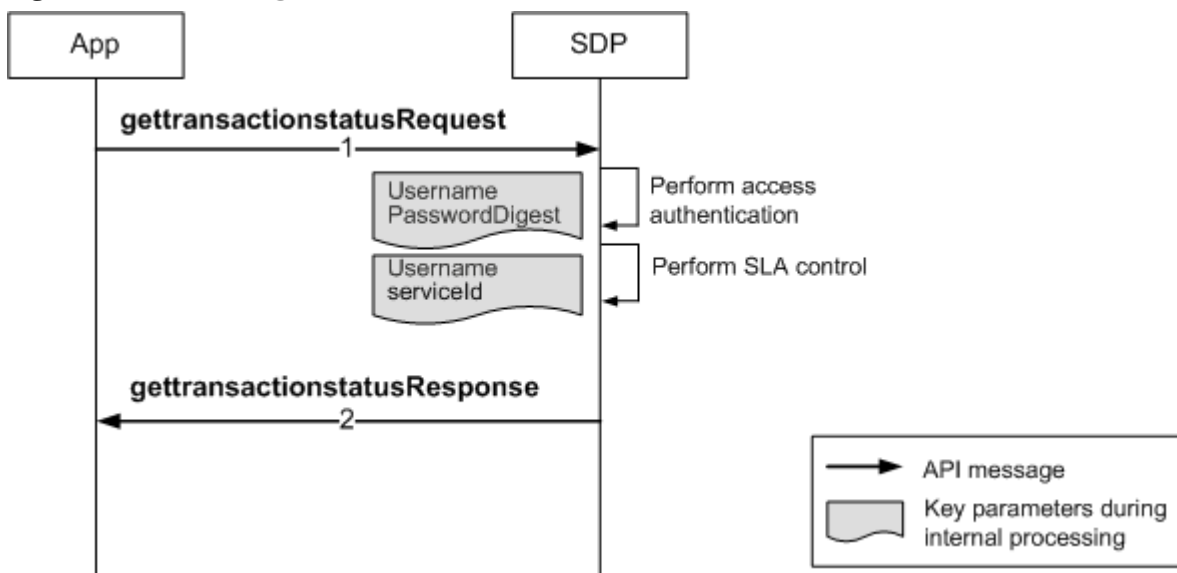
2 HTTP APIs

Table 2-34 Parameters in a gettransactionstatus response message body

Parameter	Type	Length	Level of Requirement	Description
transactionid	long		Mandatory	Transaction ID. Parameter cannot be NULL.
status	Financialtransaction status	-	Mandatory	Payment Status. . Parameter cannot be NULL. [Enumeration] <ul style="list-style-type: none"> • SUCCESSFUL • PENDING • FAILED

Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.

2.4.3 gettransactionstatusV1.2*Process***Figure 2-14** Process of gettransactionstatus*Function*

This section describes the operation used to check debit status.



2 HTTP APIs

Request URI

The request URI is the destination URI of gettransactionstatus messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/gettransactionstatus/v2

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```
POST /mom/mt/gettransactionstatus HTTP/1.1
Content-Type:text/xml
Content-Length:569
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Cookie: sessionid=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 861111111
X-HW-Extension: k1=v1;k2=v2

<?xml version="1.0" encoding="UTF-8"?>
<n:gettransactionstatusrequest xmlns:n="http://www.ericsson.com/em/emm/financial/v1_2"
xmlns:ns1="http://www.ericsson.com/em/emm/v1_0/common"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ericsson.com/em/emm/financial/v1_2">
<referenceid>111</referenceid>
<identity>ID:25612345678/MSISDN</identity>
</n:gettransactionstatusrequest>
```

Message Body Parameters

Table 2-33 describes parameters in a gettransactionstatus message body.

Table 2-35 Parameters in a gettransactionstatus message body

Parameter	Type	Length	Level of Requirement	Description
referenceid	string		Conditional	Reference ID which has been entered by user, while initiating the transaction. This field should be not appeared with the financialtransactionid field at a same time



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
identity	string		Conditional	A valid identity associated with the account holder that performed the requested transaction. This field should be not appeared with the referenceid field at a same time
financialtransactionid	string		Optional	Financial ID of the transaction, received in the previously initiated operation.

Response

A success response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<n:gettransactionstatusresponse xmlns:n="http://www.ericsson.com/em/emm/financial/v1_2"
xmlns:ns1="http://www.ericsson.com/em/emm/v1_0/common"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ericsson.com/em/emm/financial/v1_2">
<financialtransactionid>2333341</financialtransactionid>
<status>SUCCESSFUL</status>
<providertransactionid>2333341</providertransactionid>
</n:gettransactionstatusresponse>
```

Table 2-34 describes parameters in a gettransactionstatus response message body.

Table 2-36 Parameters in a gettransactionstatus response message body

Parameter	Type	Length	Level of Requirement	Description
financialtransactionid	string		Mandatory	Transaction ID. Parameter cannot be NULL.
status	Financialtransactionstatus	-	Mandatory	Payment Status. Parameter cannot be NULL. [Enumeration] • SUCCESSFUL • PENDING • FAILED
providertransactionid	string		optional	Provider transaction ID.



2 HTTP APIs

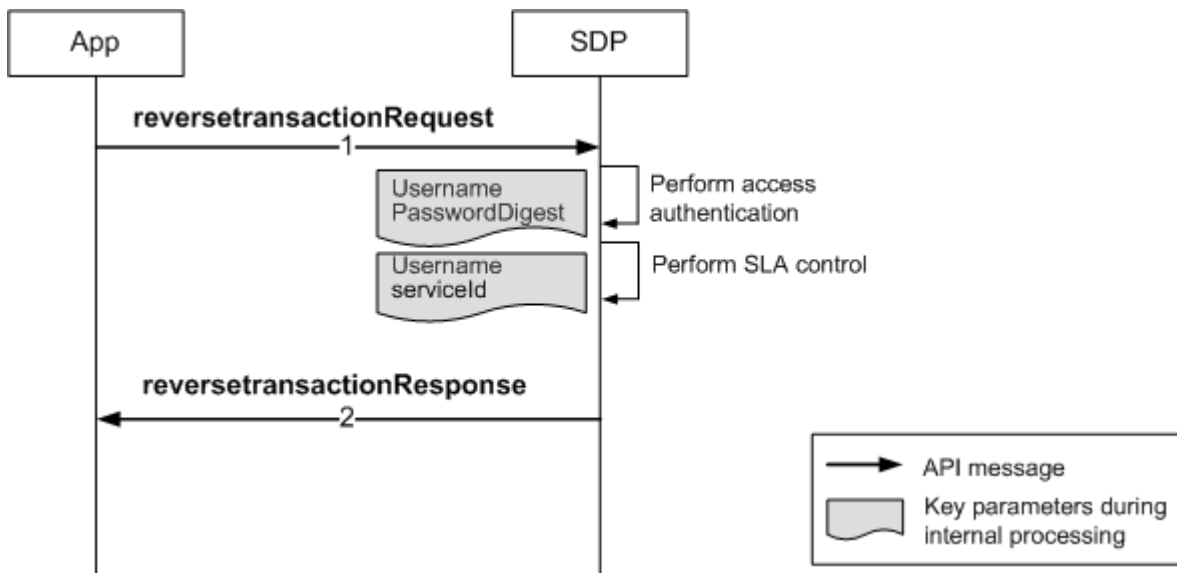
Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.

2.4.4 reversetransaction

Process

Figure 2-15 Process of reversetransaction



Function

Reverses a transaction and all of its potential sub-transactions.

Request URI

The request URI is the destination URI of reversetransaction messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/reversetransaction

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/reversetransaction HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
  
```



2 HTTP APIs

```
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
Cookie: sessionid=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 8611111111
X-HW-Extension: k1=v1;k2=v2
```

```
<?xml version="1.0" encoding="UTF-8"?>
<ns1:reversetransactionrequest xmlns:ns1="http://www.ericsson.com/em/emm">
  <financialtransactionid>1025684</financialtransactionid>
  <comment>this is a comment</comment>
  <financialinstructiondetails>
    <date>Tue Aug 13 08:13:12 UTC 2013</date>
    <fromfri>4673349175</fromfri>
    <tofri>4673349176</tofri>
    <financialinstructiontype>WITHDRAWAL</financialinstructiontype>
    <transfertype>TRANSFER</transfertype>
    <amount>10.0253654</amount>
    <frommessage>this is a frommessage</frommessage>
    <tomessage>hello</tomessage>
    <externaltransactionid>00124545</externaltransactionid>
  </financialinstructiondetails>
</ns1:reversetransactionrequest>
>
```

Message Body Parameters

Table 2-37 describes parameters in a message body.

Table 2-37 Parameters in a message body

Parameter	Type	Length	Level of Requirement	Description
financialtransactionid	long		Mandatory	The financial transaction id of the original transaction to reverse. Parameter cannot be NULL.
comment	string		Mandatory	A comment describing the reason for the reversal. Validated with NotEmpty. Validated with Length parameter(max=512). Validated with IsRestrictedString.
financialinstructiondetails	list<financialinstructiondetails>		Mandatory	A list of instructions describing how to perform the reversal. Contains information describing a single financial instruction. Validated with NotEmpty.



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
				For more details about financialinstructiondetails, see Table 2-38.

Table 2-38 describes parameters in financialinstructiondetails.

Table 2-38 Parameters in financialinstructiondetails

Parameter	Type	Length	Level of Requirement	Description
date	date		Mandatory	The time and date of the sub transaction to reverse.
fromfri	string		Mandatory	The FRI to transfer from. Validated with IsFRI. Parameter cannot be NULL.
tofri	string	-	Mandatory	The FRI to transfer to. Validated with IsFRI. Parameter cannot be NULL.
financialinstructiontype	financialinstructiontype	-	Mandatory	The type of financial instruction to perform. Parameter cannot be NULL. For more details about the financialinstructiontype, see Financial transaction type in the table below.
transfertype	transfertype	-	Mandatory	The type of transfer to perform. Parameter cannot be NULL.
amount	decimal		Mandatory	The amount to send. Parameter cannot be NULL. Validated with IsAmount.
frommessage	string		Optional	A message for the sender. Validated with IsRestrictedString.
tomessage	string		Optional	A message for the receiver. Validated with IsRestrictedString.
externaltransactionid	string		Optional	External id. Validated with Length parameter(max=256). Validated with IsExternalReferenceString.

Financial transaction type:

- DEPOSIT
- WITHDRAWAL
- TRANSFER



2 HTTP APIs

- CASH_IN
- CASH_OUT
- TRANSFER_TO_VOUCHER
- TRANSFER_FROM_VOUCHER
- PAYMENT_SEND
- PAYMENT_RECEIVE
- COMMISSIONING
- FEE
- PERIODIC_FEE
- CANCEL_VOUCHER
- VIEW_ACTIVE_TOKEN
- VIEW_TOKEN
- EXPIRE_VOUCHER
- BALANCE_ENQUIRY
- TRANSACTION_HISTORY_ENQUIRY
- REMITTANCE
- TRANSFER_TO_INVITATION
- TRANSFER_FROM_INVITATION
- CANCEL_INVITATION
- TRANSFER_FROM_VOUCHER_ATM
- PICKUP
- REVERSAL
- REVERSAL_FEE
- REVERSAL_LOY
- REVERSAL_COMMISSIONING
- ADJUSTMENT
- ADJUSTMENT_FEE
- ADJUSTMENT_LOY
- ADJUSTMENT_COMMISSIONING
- BATCH_TRANSFER
- RESOURCE_INFORMATION_ENQUIRY
- DEBIT
- RESOLVE_DEPOSIT
- SUSPEND_DEPOSIT
- PAYMENT
- CUSTODY_ACCOUNTS_TRANSFER
- EXTERNAL_TRANSFER



2 HTTP APIs

- REFUND
- INVITE_USER_REWARD
- ACCEPT_INVITE_REWARD

Response

An empty response indicates success.

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)

<?xml version="1.0" encoding="UTF-8"?>
<reversetransactionresponse xmlns="http://www.ericsson.com/em/emm"/>
```

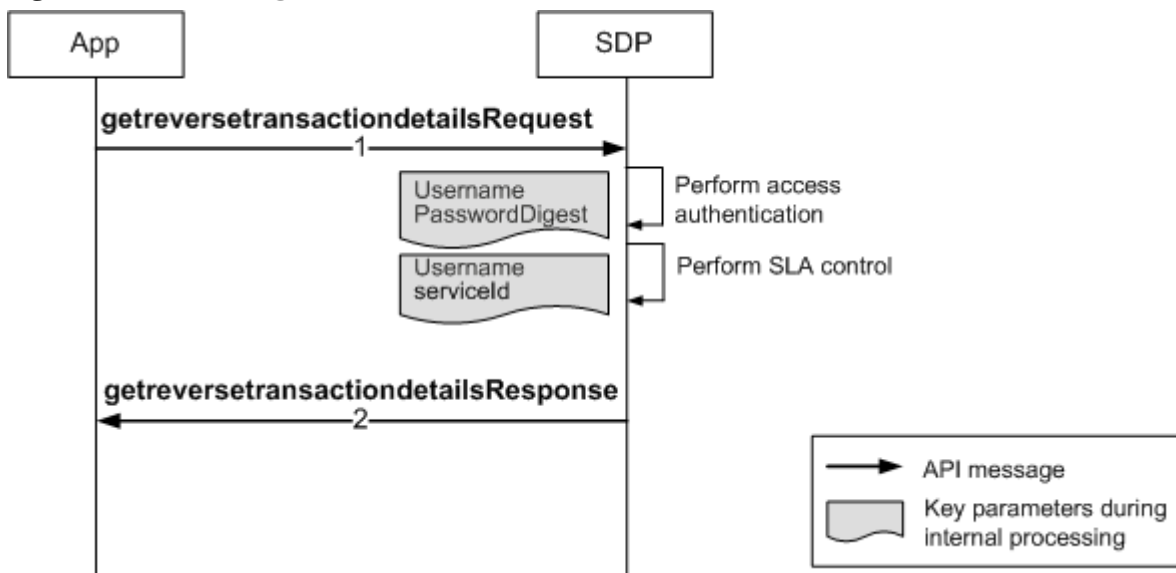
Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.

2.4.5 getreversetransactiondetails

Process

Figure 2-16 Process of getreversetransactiondetails



Function

Get status of previously initiated reversetransaction operation.



2 HTTP APIs

Request URI

The request URI is the destination URI of getreversetransactiondetails messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/getreversetransactiondetails

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```
POST /mom/mt/getreversetransactiondetails HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
Cookie: sessionid=defaultt8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 8611111111
X-HW-Extension: k1=v1;k2=v2

<?xml version="1.0" encoding="UTF-8"?>
<ns2:getreversetransactiondetailsrequest xmlns:ns2="http://www.ericsson.com/em/emm">
  <financialtransactionid>12.23</financialtransactionid>
</ns2:getreversetransactiondetailsrequest>
```

Message Body Parameters

Table 2-39 describes parameters in a message body.

Table 2-39 Parameters in a message body

Parameter	Type	Length	Level of Requirement	Description
financialtransactio nid	long		Mandatory	The Financial Resource Identifier of the sending account. Parameter cannot be NULL.

Response

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)
```



2 HTTP APIs

```
<?xml version="1.0" encoding="UTF-8"?>
<ns1:getreversetransactiondetailsresponse xmlns:n1="http://www.ericsson.com/em/emm">
  <subtransactions>
    <subtransaction>
      <date>2013</date>
      <fromfri>11</fromfri>
      <tofri>11</tofri>
      <financialinstructiontype>11</financialinstructiontype>
      <transfertype>11</transfertype>
      <amount>11</amount>
      <frommessage>11</frommessage>
      <tomessage>11</tomessage>
      <externaltransactionid>11</externaltransactionid>
    </subtransaction>
  </subtransactions>
</ns1:getreversetransactiondetailsresponse>
```

Table 2-40 describes parameters in a response message body.

Table 2-40 Parameters in a response message body

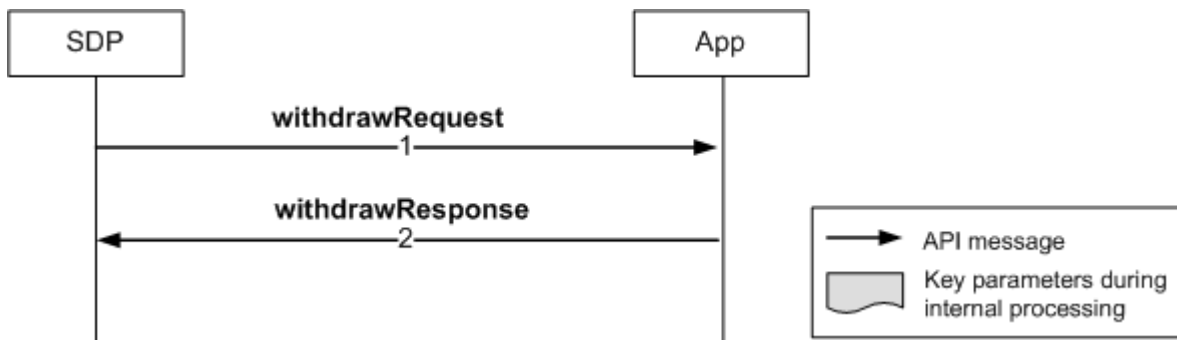
Parameter	Type	Length	Level of Requirement	Description
subtransactions	list<financialinstructiondetails>	-	Optional	The list of sub transactions for the requested transaction. Contains information describing a single financial instruction For more details about financialinstructiondetails, see Table 2-38.

Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.



2 HTTP APIs

2.4.6 withdraw*Process***Figure 2-17** Process of withdraw*Function*

The operation transfers money from the account holders account in the Partner system to an eWallet in the Mobile Money Manager.

Request URI

The request URI is the destination URI of withdraw messages sent by the SDP to the App. The URI is defined by the App.

Request Message Body Example

```

POST /mom/mo/getreversetransactiondetails HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-NotifyHeader: notify TransId=" 200903241230451000000123456789" , LinkId="A15Y89Z32T99"
Signature:43AD232FD45FF
X-HW-Extension: k1=v1;k2=v2
    
```

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns4:withdrawrequest xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:op="http://www.ericsson.com/em/emm/common"
xmlns:ns4="http://www.ericsson.com/em/emm/sp/backend">
<resource>FRI:20200009999@savings.sp1/SP</resource>
<accountholderid>ID:20200009999/MSISDN</accountholderid>
<amount>
<amount>130</amount>
<currency>RWF</currency>
    
```



2 HTTP APIs

```
</amount>
<transactionid>2110248</transactionid>
<message>Transfer to</message>
</ns4:withdrawrequest>
```

Message Body Parameters

Table 2-41 describes parameters in a message body.

Table 2-41 Parameters in a message body

Parameter	Type	Length	Level of Requirement	Description
resource	string		Mandatory	The FRI of the account, bill or user in the partner system. For example, FRI:12345ABC@water.company/SP
accountholderid	string		Mandatory	The MSISDN identity of the account holder that is performing the operation. For example: ID:46733491234/MSISDN
amount	moneydetailst - ype		Mandatory	Amount to be withdrawn. Parameter cannot be NULL. For details about moneydetailstype, see Table 2-31.
transactionid	long		Mandatory	System internal transaction id. Parameter cannot be NULL.
message	string		Optional	A descriptive note for Partner.
extension	extensiontype -		Optional	Used for sending any additional information to Partner. For details about extensiontype, see Table 2-32.

Response

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)
```



2 HTTP APIs

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns4:withdrawresponse xmlns:ns4="http://www.ericsson.com/em/emm/sp/backend">
  <transactionid>2110248</transactionid>
  <providertransactionid>1351000005264</providertransactionid>
  <status>COMPLETED</status>
  <newbalance>
    <amount>270</amount>
    <currency>RWF</currency>
  </newbalance>
  <message>THIS IS A MESSAGE TO THE END USER</message>
  <extension/>
</ns4:withdrawresponse>
```

Table 2-42 describes parameters in a response message body.

Table 2-42 Parameters in a response message body

Parameter	Type	Length	Level of Requirement	Description
transactionid	long		Mandatory	System Transaction ID. Parameter can not be NULL.
providertransactionid	string		Mandatory	Transaction id from the service provider system.
status	status		Mandatory	Status of withdrawal operation. Parameter can not be NULL. Possible values are: <ul style="list-style-type: none"> ● CREATED ● PENDING ● COMPLETED ● FAILED ● EXPIRED
newbalance	moneydetailstype		Optional	Balance of the account in the service provider. For details about moneydetailstype, see Table 2-31.
message	string	-	Optional	A descriptive note for user.
extension	extensiontype		Optional	Any additional information received from Partner. For details about extensiontype, see Table 2-32.



2 HTTP APIs

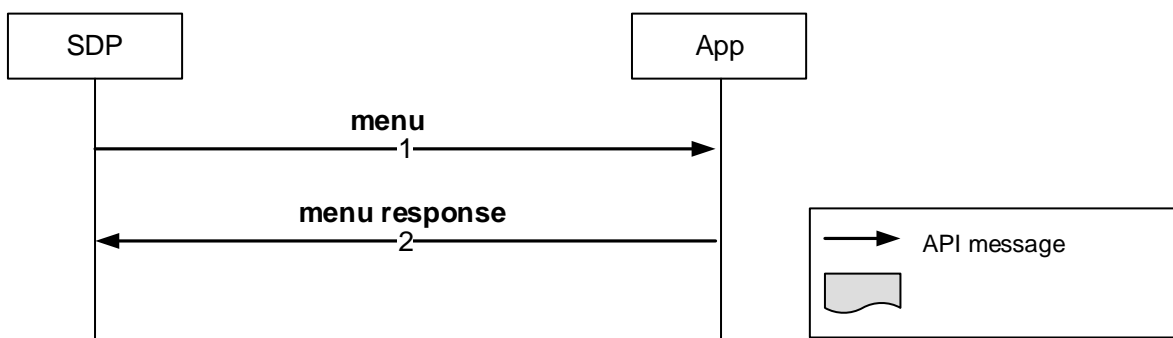
Error Codes

The App returns error codes to the SDP when an exception occurs in response to withdraw messages. The error codes are defined by partners.

Table 2-43 describes withdraw error codes that the partners may return upon an exception.

Table 2-43 withdraw error codes

Error Code	Description
01	Successfully processed transaction.
02	Invalid account details.
03	Transaction not allowed.
04	Payment below minimum.
05	Invalid Mobile Money platform userid/password.
06	Payment Entry not found.
07	Missing transaction/message ID.
08	Missing application version.
100	General failure.
101	Field length exceeded.
102	Subscriber could not be identified. (Invalid subscriberID).
103	Duplicate transaction id.

2.4.7 channelOrchestrator*Process***Figure 2-18** Process of withdraw



2 HTTP APIs

Function

This channelOrchestrator operation is sent from the channel orchestrator towards/from the menu provider (through the HSDP) to request/forward menu contents.

Request URI

The request URI is the destination URI of channelOrchestrator messages sent by the SDP to the App. The URI is defined by the App.

Request Message Body Example

```
POST /mom/mo/channelOrchestrator HTTP/1.1
Content-Type:text/xml
Content-Length:569
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-NotifyHeader: notify TransId=" 200903241230451000000123456789" , LinkId="A15Y89Z32T99"
X-HW-Extension: k1=v1;k2=v2

<?xml version="1.0" encoding="UTF-8"?>
<n:menurequest xmlns:n="http://www.ericsson.com/em/am/co/menu/v1_0/"
xmlns:ns1="http://www.ericsson.com/em/am/co/v1_0/common"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ericsson.com/em/am/co/menu/v1_0/">
<channel>SMS</channel>
<language>english</language>
</n:menurequest>
```

Message Body Parameters

Table 2-41 describes parameters in a message body.

Table 2-44 Parameters in a message body

Parameter	Type	Length	Level of Requirement	Description
channel	string	1024	Mandatory	Source channel information
language	string		Optional	The language of the menu requested for

Response

A success response example:

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



2 HTTP APIs

```
<n:menuresponse xmlns:n="http://www.ericsson.com/em/am/co/menu/v1_0/"
xmlns:ns1="http://www.ericsson.com/em/am/co/v1_0/common"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ericsson.com/em/am/co/menu/v1_0/ ">
<contents>String</contents>
<translations>
  <translation>
    <language>en</language>
    <content>String</content>
  </translation>
</translations>
<policy>String</policy>
</n:menuresponse>
```

Table 2-45 describes parameters in a response message body.

Table 2-45 Parameters in a response message body

Parameter	Type	Length	Level of Requirement	Description
contents	string		Mandatory	The menu xml content of the provider.
translations	Translation[1..unbond]		Mandatory	Listing of all translations for each language. The translation content listing all keys and values of the translation.
policy	string		Optional	The policy content of the menu definition

Table 2-46 Parameters in Translation

Parameter	Type	Length	Level of Requirement	Description
language	string	50	Mandatory	The language of the translation
content	string	2048	Mandatory	The content of the translation document

Error Codes

Please see 2.7 API Error Response.

2.5 Batch Processing

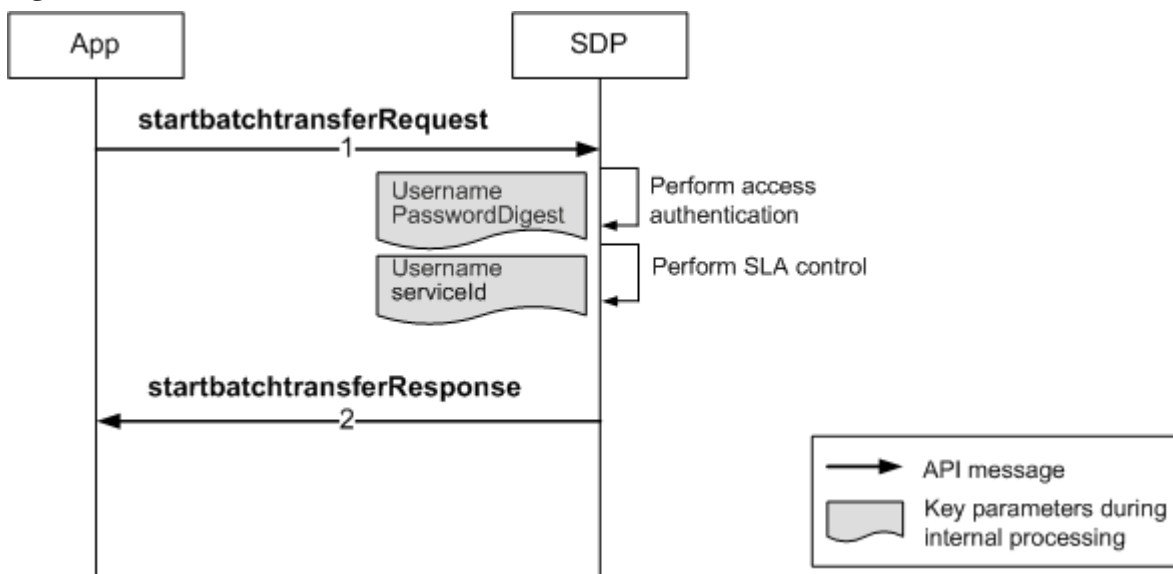


2 HTTP APIs

2.5.1 startbatchtransfer

Process

Figure 2-19 Process of startbatchtransfer



Function

Start a batch by sending a file containing transaction detail in a multipart HTTP request.

Request URI

The request URI is the destination URI of `startbatchtransfer` messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/startbatchtransfer

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/startbatchtransfer HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
  
```



2 HTTP APIs

```
Cookie: sessionId=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 861111111
X-HW-Extension: k1=v1;k2=v2
```

```
<ns2:startbatchtransferrequestheader xmlns:ns2= "http://www.ericsson.com/em/emm">
  <batchtitle>12.23</batchtitle>
  <fromaccountfri>abc</fromaccountfri>
  <numberofrecords>123</numberofrecords>
  <totalamount>
    <amount>abc</amount>
    <currency>abc</currency>
  </totalamount>
  <parsername>abc</parsername>
  <parserarguments>
    <argument>
      <name>abc</name>
      <value>abc</value>
    </argument>
  </parserarguments>
</ns2:startbatchtransferrequestheader>
```

Message Body Parameters

Table 2-47 describes parameters in a startbatchtransfer message body.

Table 2-47 Parameters in a startbatchtransfer message body

Parameter	Type	Length	Level of Requirement	Description
startbatchtransferrequestheader	Startbatchtransferrequestheader	-	Optional	See for more details.

Table 2-48 describes parameters in a Startbatchtransferrequestheader message body.

Table 2-48 Parameters in a Startbatchtransferrequestheader message body

Parameter	Type	Length	Level of Requirement	Description
batchtitle	string		Optional	The description of the batch, it is set by the person creating the file. The description will be present in the result file. Validated with Length parameter(max=128). Validated with IsRestrictedString.



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
fromaccountfri	string		Optional	The account where money is transferred from specified as an FRI. If not specified, the default account of the logged in user is used. Validated with IsFRI.
numberofrecords	long		Mandatory	The number of transaction records in the batch. Parameter cannot be NULL. Maximum value: 10000L.
totalamount	decimal		Mandatory	Total sum of amount for all transactions in transaction details file. Must be greater than 0 (zero). Parameter cannot be NULL. Validated with IsAmount.
currency	string		Optional	Amount currency. Validated with IsCurrencyCode.
parsername	string		Optional	Name of file parser that will be used to parse the batch data. System default parser will be used if this parameter is not set. Validated with IsSingleWordString.
parserarguments	list<propertytype>		Optional	Parser arguments. Contains the name and value of a property. For more details about propertytype, see Table 2-49.

Table 2-49 describes parameters of propertytype.

Table 2-49 Parameters in propertytype

Parameter	Type	Length	Level of Requirement	Description
name	string	-	Mandatory	The name of the property. Validated with NotBlank. Validated with IsASCIISimpleString.
value	string		Mandatory	The value of the property. Validated with NotBlank. Validated with IsPropertyValue.

Response

A success response example:



2 HTTP APIs

```

HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)

<?xml version="1.0" encoding="UTF-8"?>
<tns:startbatchtransferresponse xmlns:tns="http://www.ericsson.com/em/emm">
  <batchtransferid>2147483647</batchtransferid>
</tns:startbatchtransferresponse>

```

Table 2-50 describes parameters in a startbatchtransfer response message body.

Table 2-50 Parameters in a startbatchtransfer response message body

Parameter	Type	Length	Level of Requirement	Description
batchtransferid	long		Mandatory	Numerical identification of the performed batch transfer.

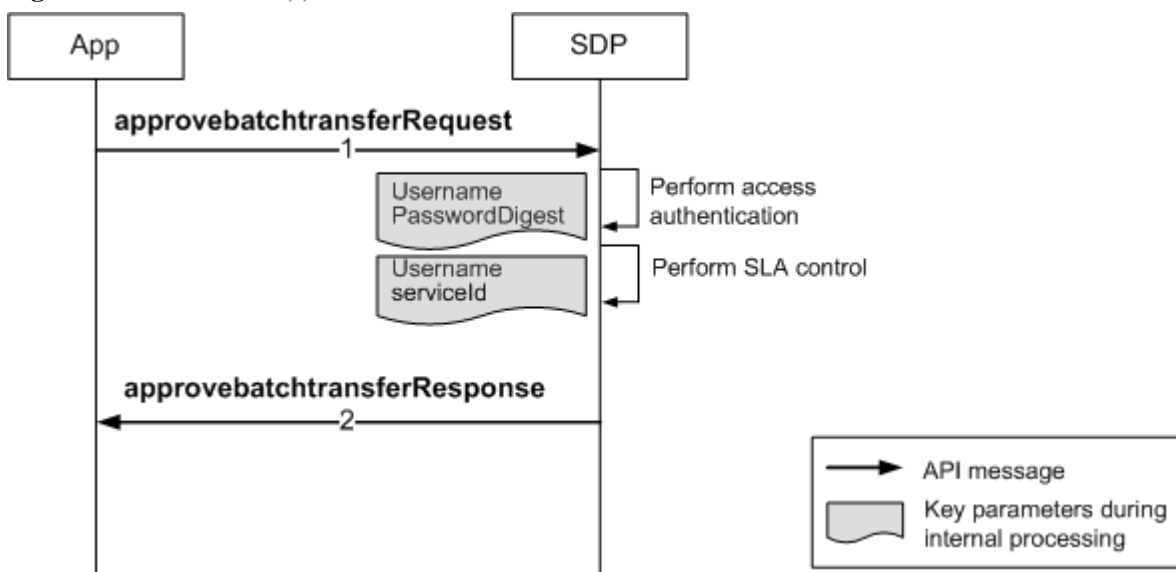
Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.

2.5.2 approvebatchtransfer

Process

Figure 2-20 Process of approvebatchtransfer





2 HTTP APIs

Function

Approves a batch transfer previously created with the startbatchtransfer operation.

Request URI

The request URI is the destination URI of approvebatchtransfer messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/approvebatchtransfer

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```
POST /mom/mt/approvebatchtransfer HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
Cookie: sessionid=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 861111111
X-HW-Extension: k1=v1;k2=v2

<?xml version="1.0" encoding="UTF-8"?>
<approvebatchtransferrequest xmlns="http://www.ericsson.com/em/emm">
  <batchid xmlns="">2147483647</batchid>
</approvebatchtransferrequest>
```

Message Body Parameters

Table 2-51 describes parameters in a approvalbatchtransfer message body.

Table 2-51 Parameters in a approvalbatchtransfer message body

Parameter	Type	Length	Level of Requirement	Description
batchid	long		Mandatory	The batch id generated by Mobile Money Manager when the batch was created. Parameter cannot be NULL.

Response

A success response example:

```
HTTP/1.1 200 OK
```



2 HTTP APIs

```
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)
```

```
<?xml version="1.0" encoding="UTF-8"?>
<approvebatchtransferresponse xmlns="http://www.ericsson.com/em/emmm"/>
```

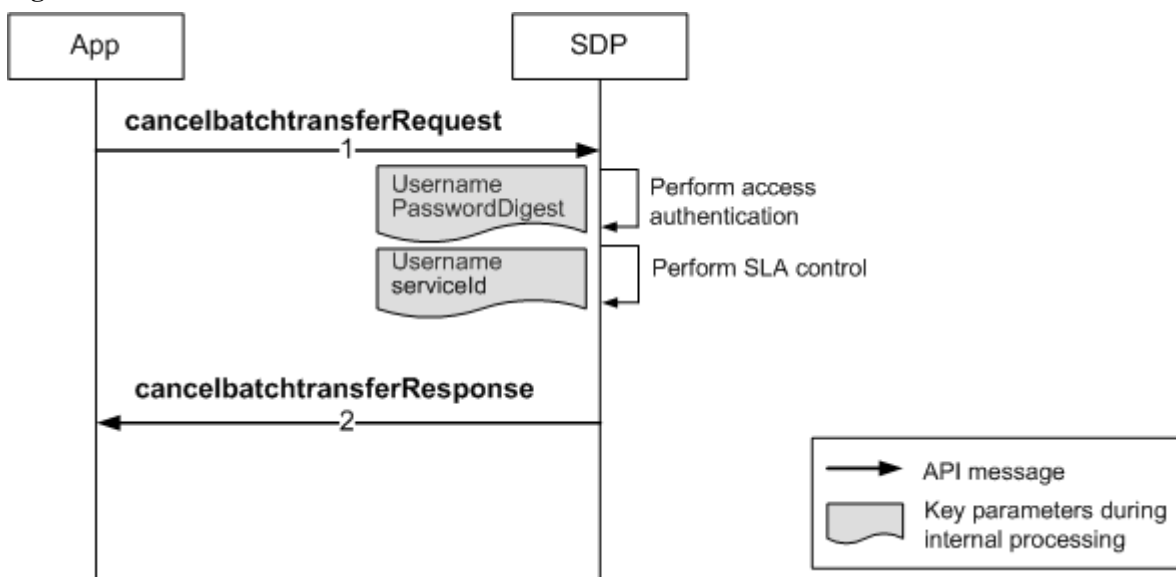
Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.

2.5.3 cancelbatchtransfer

Process

Figure 2-21 Process of cancelbatchtransfer



Function

Cancel an unprocessed batch transfer.

Request URI

The request URI is the destination URI of cancelbatchtransfer messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/cancelbatchtransfer

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.



2 HTTP APIs

Request Message Body Example

```
POST /mom/mt/cancelbatchtransfer HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
Cookie: sessionid=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 8611111111
X-HW-Extension: k1=v1;k2=v2

<?xml version="1.0" encoding="UTF-8"?>
<cancelbatchtransferrequest xmlns="http://www.ericsson.com/em/emm">
<batchid xmlns="">2147483647</batchid>
</cancelbatchtransferrequest>
```

Message Body Parameters

Table 2-52 describes parameters in message body.

Table 2-52 Parameters in message body

Parameter	Type	Length	Level of Requirement	Description
batchid	long		Mandatory	The batch transfer identification number. Parameter cannot be NULL.

Response

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)

<?xml version="1.0" encoding="UTF-8"?>
<cancelbatchtransferresponse xmlns="http://www.ericsson.com/em/emm"/>
```

Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.

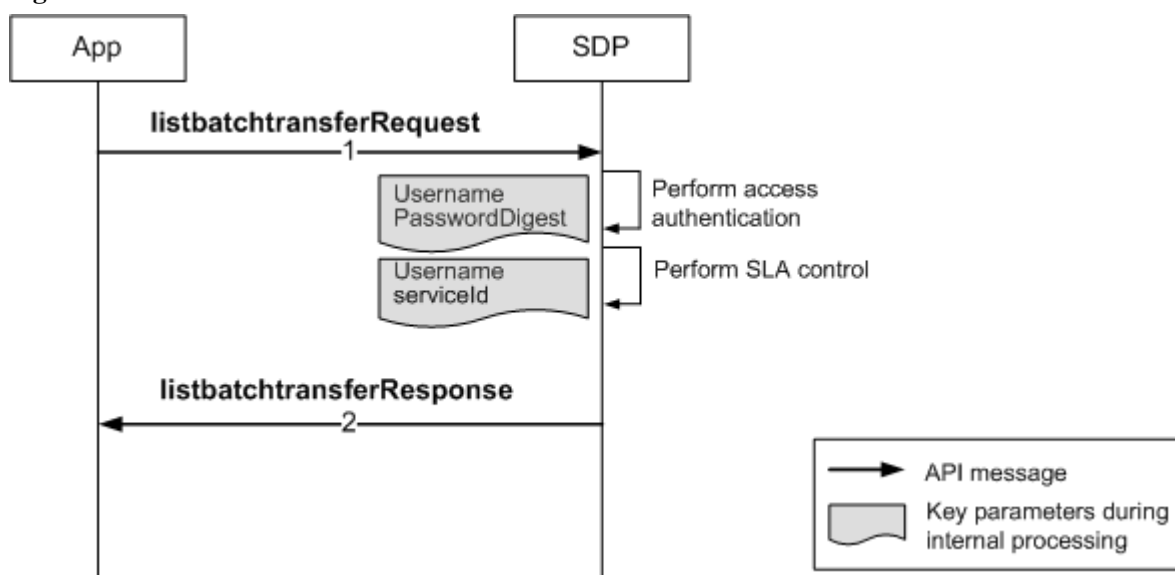


2 HTTP APIs

2.5.4 listbatchtransfer

Process

Figure 2-22 Process of listbatchtransfer



Function

Fetch batch transfers for a specific account, with possibility to also specify time interval and status.

Request URI

The request URI is the destination URI of `listbatchtransfer` messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/listbatchtransfer

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/listbatchtransfer HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
  
```



2 HTTP APIs

```
Cookie: sessionid=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 8611111111
X-HW-Extension: k1=v1;k2=v2
```

```
<?xml version="1.0" encoding="UTF-8"?>
<ns2:listbatchtransferrequest xmlns:ns2= "http://www.ericsson.com/em/emm">
  <fromaccountfri>456</fromaccountfri>
  <numberofbatches>798</numberofbatches>
  <startdate>654</startdate>
  <enddate>abc</enddate>
  <statuslist>
    <status>CREATED</status>
  </statuslist>
</ns2:listbatchtransferrequest>
```

Message Body Parameters

Table 2-53 describes parameters in a message body.

Table 2-53 Parameters in a message body

Parameter	Type	Length	Level of Requirement	Description
fromaccountfri	string		Optional	List batch transfers originating from this account. If not specified, the default account of the logged in user is used. Validated with IsFRI.
numberofbatches	integer		Optional	The maximum number of batch transfers to be returned.
startdate	string		Optional	If present, only batch transfers received from this date and onwards will be returned. Validated with IsDate parameter(format=yyyy-MM-dd).
enddate	string		Optional	If present, only batch transfers received up to this date will be returned. Validated with IsDate parameter(format=yyyy-MM-dd).



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
status	list<batchtransferstatus>	-	Optional	Batch transfer status list. Contains the different status codes for a batch transfer. [Enumeration] <ul style="list-style-type: none"> • CREATED • APPROVED • ONGOING • COMPLETED • CANCELLED

Response

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)
```

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:listbatchtransferresponse xmlns:tns="http://www.ericsson.com/em/emm" >
  <batchinfos>
    <batchinfo>
      <batchid>12</batchid>
      <title>title</title>
      <receivetimestamp>20130726111233</receivetimestamp>
      <completedtimestamp>20130726111234</completedtimestamp>
      <fromaccountfri>1</fromaccountfri>
      <batchcreatormsisdn>1</batchcreatormsisdn>
      <batchcreatorname>1</batchcreatorname>
      <batchapprovermsisdn>1</batchapprovermsisdn>
      <batchapprovername>1</batchapprovername>
      <batchapprovable>1</batchapprovable>
      <batchcancelledbysmsdn>1</batchcancelledbysmsdn>
      <batchcancelledbyname>1</batchcancelledbyname>
      <numberofrecords>1</numberofrecords>
      <totalamount>1</totalamount>
      <currency>1</currency>
      <status>1</status>
      <numberofsuccess>1</numberofsuccess>
      <numberoffailure>1</numberoffailure>
      <totalfee>1</totalfee>
      <totaltransferredamount>1</totaltransferredamount>
      <transactionrecords>
```




2 HTTP APIs

```

<transactionrecord>
  <transactionidentifier>1</transactionidentifier>
  <receiveridtype>1</receiveridtype>
  <receiverid>1</receiverid>
  <transactionid>1</transactionid>
  <transactiondatetime>1</transactiondatetime>
  <amount>1</amount>
  <result>1</result>
  <reason>1</reason>
  <message>1</message>
  <fee>1</fee>
  <transferredamount>1</transferredamount>
</transactionrecord>
</transactionrecords>
</batchinfo>
</batchinfos>
</tns:listbatchtransferresponse>

```

Table 2-54 describes parameters in a response message body.

Table 2-54 Parameters in a response message body

Parameter	Type	Length	Level of Requirement	Description
batchinfos	list<batchinfotype>	-	Optional	List of batch transfer information. Contains information about a batch transfer. For more details about batchinfotype, see Table 2-55.

Table 2-55 describes parameters in batchinfotype.

Table 2-55 Parameters in batchinfotype

Parameter	Type	Length	Level of Requirement	Description
batchid	long		Mandatory	Generated by the system when the batch is accepted.
title	string		Optional	Import CSV file header title. Validated with IsMultiWordString.
fromaccountfri	string		Mandatory	From account Financial Resource Identifier. Validated with IsFRI.
batchcreatormsisdn	string		Mandatory	Creator msisdn. Validated with IsMSISDN.
batchcreatorname	string		Optional	Creator name. Validated with IsUsername.



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
batchapprovermsisdn	string		Optional	Approver msisdn. Validated with IsMSISDN.
batchapprovername	string		Optional	Approver name. Validated with IsUsername.
batchapprovable	boolean		Optional	Batch is approvable or not, present if status is created.
batchcancelledbysisdn	string		Optional	CancelledBy msisdn. Validated with IsMSISDN.
batchcancelledbyname	string		Optional	CancelledBy name. Validated with IsUsername.
receivetimestamp	string		Mandatory	Time stamp when the system received the start batch request. Validated with IsDate parameter(format=yyyy-MM-dd HH:mm:ss).
numberofrecords	long		Mandatory	Import CSV file number of records.
totalamount	decimal		Mandatory	Import CSV file total amount.
currency	string		Mandatory	Import CSV file currency. Validated with IsCurrencyCode.
status	string		Mandatory	Import CSV file status. Validated with IsSingleWordString.
completedtimestamp	string		Mandatory	Time stamp when the system completed the batch request. Validated with IsDate parameter(format=yyyy-MM-dd HH:mm:ss).
numberofsuccess	long		Optional	The number of successfully executed transfers. Present if status is completed.
numberoffailure	long		Optional	The number of failed transfers. Present if status is completed.
totalfee	decimal		Optional	The totally withdrawn fee. Present if status is completed.
totaltransferredamount	decimal		Optional	The totally transferred amount. Present if status is completed.
transactionrecords	list<transactionrecordtype>		Optional	List of batch transaction details. Contains details about a batch transaction. For more details about transactionrecordtype, see Table 2-56.



2 HTTP APIs

Table 2-56 describes parameters in transactionrecordtype.

Table 2-56 Parameters in transactionrecordtype

Parameter	Type	Length	Level of Requirement	Description
transactionidentifier	string		Mandatory	The identifier of the transaction record. Validated with IsExternalReferenceString.
receiveridtype	string		Mandatory	The id type of the receiver, passport, identification card etc. Validated with IsRestrictedString.
receiverid	string		Mandatory	The number of the receiver identification. Validated with IsRestrictedString.
transactionid	string		Optional	Transaction id. Validated with IsNumericString.
transactiondatetime	string		Optional	Time of transaction. Validated with IsDate parameter(format=yyyy-MM-ddHH:mm:ss).
amount	decimal		Mandatory	Transaction amount.
result	string		Mandatory	The result of the transaction. Validated with IsSingleWordString.
reason	string		Optional	Free text. Validated with IsRestrictedString.
message	string		Optional	Transaction message. Validated with IsRestrictedString.
fee	decimal		Optional	Transaction fee.
transferredamount	decimal		Optional	The amount being transferred.

Error Codes

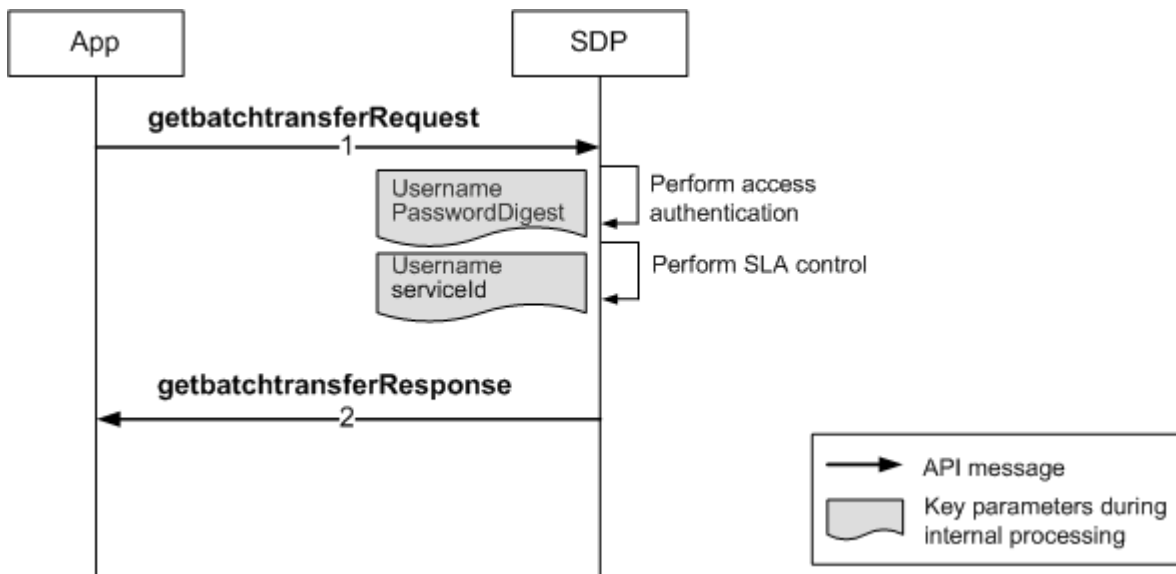
SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.



2 HTTP APIs

2.5.5 *getbatchtransfer*

Process

Figure 2-23 Process of *getbatchtransfer*

Function

Retrieves information about a specific batch.

Request URI

The request URI is the destination URI of *getbatchtransfer* messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/getbatchtransfer

In the format, IP and Port indicate the service IP address and SOAP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/getbatchtransfer HTTP/1.1
Content-Type:text/xml
Content-Length:1069
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Signature:43AD232FD45FF
  
```



2 HTTP APIs

```
Cookie: sessionid=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 861111111
X-HW-Extension: k1=v1;k2=v2
```

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:getbatchtransferrequest xmlns:tns="http://www.ericsson.com/em/emm">
  <batchid>2147483647</batchid>
  <resulttype>SUMMARY_REPORT</resulttype>
</tns:getbatchtransferrequest>
```

Message Body Parameters

Table 2-21 describes parameters in a message body.

Table 2-57 Parameters in a message body

Parameter	Type	Length	Level of Requirement	Description
batchid	long		Mandatory	The batch id generated by Mobile Money Manager when the batch was created. Parameter cannot be NULL.
resulttype	batchtransferresulttype		Optional	Controls the level of information to return. [Enumeration] <ul style="list-style-type: none"> SUMMARY_REPORT FULL_REPORT FULL_REPORT_ON_COMPLETED

Response

A success response example:

```
HTTP/1.1 200 OK
Signature:43AD232FD45FF
Content-Length: 0
Server: Jetty(7.5.4.v20111024)
```

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:getbatchtransferresponse xmlns:tns="http://www.ericsson.com/em/emm">
  <batchinfo>
    <batchid>123</batchid>
    <title>11</title>
    <receivetimestamp>11</receivetimestamp>
    <completedtimestamp>11</completedtimestamp>
    <fromaccountfri>11</fromaccountfri>
    <batchcreatormsisdn>11</batchcreatormsisdn>
```



2 HTTP APIs

```

<batchcreatorname>11</batchcreatorname>
<batchapprovermsisdn>11</batchapprovermsisdn>
<batchapprovername>11</batchapprovername>
<batchapprovable>11</batchapprovable>
<batchcancelledbymsisdn>11</batchcancelledbymsisdn>
<batchcancelledbyname>11</batchcancelledbyname>
<numberofrecords>11</numberofrecords>
<totalamount>11</totalamount>
<currency>11</currency>
<status>11</status>
<numberofsuccess>11</numberofsuccess>
<numberoffailure>11</numberoffailure>
<totalfee>11</totalfee>
<totaltransferredamount>11</totaltransferredamount>
<transactionrecords>
  <transactionrecord>
    <transactionidentifier>11</transactionidentifier>
    <receiveridtype>11</receiveridtype>
    <receiverid>11</receiverid>
    <transactionid>11</transactionid>
    <transactiondatetime>11</transactiondatetime>
    <amount>11</amount>
    <result>11</result>
    <reason>11</reason>
    <message>11</message>
    <fee>11</fee>
    <transferredamount>11</transferredamount>
  </transactionrecord>
</transactionrecords>
</batchinfo>
</tns:getbatchtransferresponse>

```

Table 2-22 describes parameters in a response message body.

Table 2-58 Parameters in a response message body

Parameter	Type	Length	Level of Requirement	Description
batchinfo	batchinfotype	-	Mandatory	Transfer Batch Info. For more details about batchinfotype, see Table 2-55.

Error Codes

SDP may return upon an exception. For details about the error codes, see 2.7 API Error Response.



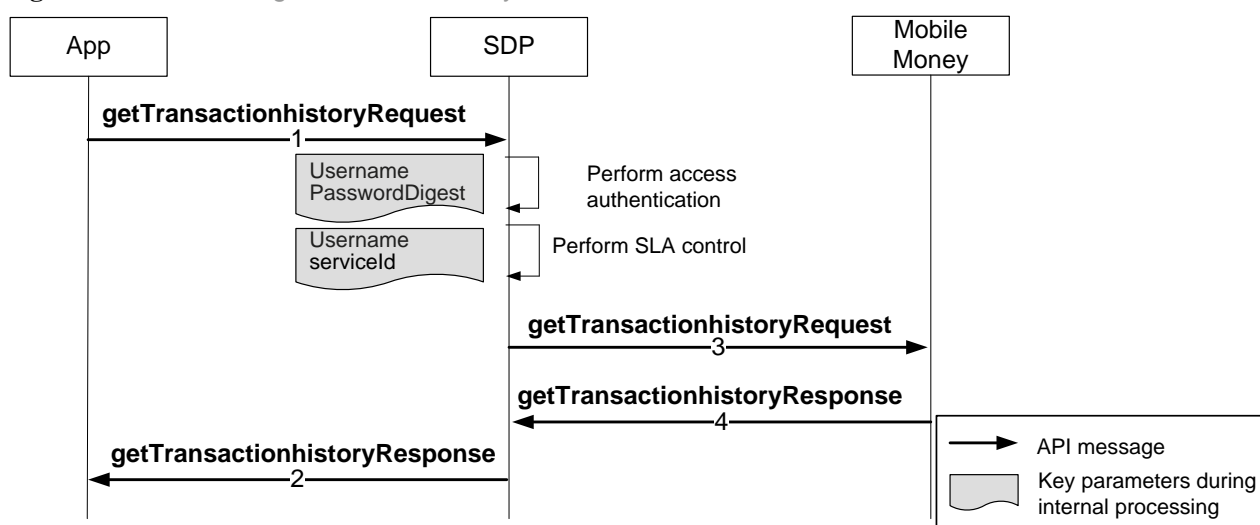
2 HTTP APIs

2.6 Financial

2.6.1 getTransactionhistory

Process

Figure 2-24 Process of getTransactionhistory



Function

A `getTransactionhistory` enables a Partner to query a list of transactions matching the parameters in the request.

Request URI

The request URI is the destination URI of `getTransactionhistory` messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/getTransactionhistory

In the format, IP and Port indicate the service IP address and HTTP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/getTransactionhistory HTTP/1.1
Content-Type:text/xml
Content-Length:569
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
  
```



2 HTTP APIs

```
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Cookie: sessionId=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 8611111111
X-HW-Extension: k1=v1;k2=v2
Signature:43AD232FD45FF
```

```
<?xml version="1.0" encoding="UTF-8"?>
<v1_0:gettransactionhistoryrequest xmlns:ns2="http://www.ericsson.com/em/emm/v1_1/common"
xmlns:fic="http://www.ericsson.com/em/emm/financial/v1_0/common"
xmlns:op="http://www.ericsson.com/em/emm/v1_0/common"
xmlns:ns1="urn:iso:std:iso:20022:tech:xsd"
xmlns:v1_0="http://www.ericsson.com/em/emm/financial/v1_0" version="1.0">
  <fri>fri0</fri>
  <numoftransactions>50</numoftransactions>
  <indexoffset>0</indexoffset>
  <financialtransactionid>1111</financialtransactionid>
  <externaltransactionid>externaltransactionid0</externaltransactionid>
  <transactionstatus>SUCCESSFUL</transactionstatus>
  <transactiontype>DEPOSIT</transactiontype>
  <transactiontype>DEPOSIT</transactiontype>
  <startdatefrom>2006-05-04T18:13:51.0Z</startdatefrom>
  <startdateto>2006-05-04T18:13:51.0Z</startdateto>
  <datefrom>2006-05-04T18:13:51.0Z</datefrom>
  <dateto>2006-05-04T18:13:51.0Z</dateto>
  <direction>IN</direction>
  <otherfri>otherfri0</otherfri>
  <quoteid>quoteid0</quoteid>
</v1_0:gettransactionhistoryrequest>
```

Message Body Parameters

Table 2-59 describes parameters in a getTransactionhistory message body.

Table 2-59 Parameters in a getTransactionhistory message body

Parameter	Type	Length	Level of Requirement	Description
fri	string		Optional	The FRI to get the transactions for. If not specified the default FRI of the calling user will be used. Validated with IsFRI.
numoftransactions	int		Optional	The number of transactions to return in the response. Default value: 50. Minimum value: 1. Parameter can not be NULL.



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
indexoffset	int		Optional	Start retrieving transaction history from this index.
financialtransactionid	long		Optional	Retrieve transaction history with this financial transaction ID. Validated with IsPositiveLong (allowZero=false).
externaltransactionid	string		Optional	Retrieve transaction history with this external transaction ID. Validated with IsExternalReferenceString.
transactionstatus	string		Optional	Retrieve transaction history with this status. Possible values are: <ul style="list-style-type: none"> • SUCCESSFUL • PENDING • FAILED
transactiontype	string[0...unbounded]		Optional	Retrieve transaction history with this transaction type. It is possible to have more than one transactiontype element to filter on. Shows the financial transaction type Possible values are: <ul style="list-style-type: none"> • DEPOSIT • REVERSE_DEPOSIT • WITHDRAWAL • TRANSFER • CASH_IN • CASH_OUT • TRANSFER_TO_VOUCHER • TRANSFER_FROM_VOUCHER • PAYMENT_SEND • PAYMENT_RECEIVE • COMMISSIONING • FEE



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
				<ul style="list-style-type: none"> • PERIODIC_FEE • CANCEL_VOUCHER • VIEW_ACTIVE_TOKEN • VIEW_TOKEN • EXPIRE_VOUCHER • BALANCE_ENQUIRY • TRANSACTION_HISTORY_ENQUIRY • REMITTANCE • TRANSFER_TO_INVITATION • TRANSFER_FROM_INVITATION • CANCEL_INVITATION • CASH_OUT_ATM • TRANSFER_FROM_VOUCHER_ATM • PICKUP • REVERSAL • REVERSAL_FEE • REVERSAL_LOY • REVERSAL_COMMISSIONING • REVERSAL_TAX • REVERSAL_DISCOUNT • REVERSAL_PROMOTION • ADJUSTMENT • ADJUSTMENT_FEE • ADJUSTMENT_LOY • ADJUSTMENT_COMMISSIONING • ADJUSTMENT_TAX • ADJUSTMENT_DISCOUNT_PROMOTION • BATCH_TRANSFER • RESOURCE_INFORMATION_ENQUIRY • DEBIT • RESOLVE_DEPOSIT



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
				<ul style="list-style-type: none"> • SUSPEND_DEPOSIT • PAYMENT • CUSTODY_ACCOUNTS_TRANSFER • EXTERNAL_TRANSFER • REFUND • INVITE_USER_REWARD • ACCEPT_INVITE_REWARD • TAX • DECLINE_INVITATION • EXPIRED_INVITATION • FLOAT_TRANSFER • TRANSFER_TO_ANY_BANK_ACCOUNT • TRANSFER_LOY • EXTERNAL_TRANSFER_LOY • PAYMENT_LOY • PAYMENT_SEND_LOY • PAYMENT_RECEIVE_LOY • DEBIT_LOY • REFUND_LOY • REVERSAL_USER_LOY • COUPONS_REDEEM_COUPON • COUPONS_ACQUIRE_COUPON • COUPONS_ISSUE_OFFER • COUPONS_REDEEMED_COUPON_RATE • DEPOSIT_NOTIFICATION • WITHDRAWAL_NOTIFICATION • REVERSAL_SENDER_NOTIFICATION • REVERSAL_RECEIVER_NOTIFICATION • LOAN_APPLICATION • VIEW_LOAN_REPAYMENTS



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
				<ul style="list-style-type: none"> VIEW_LOAN_DETAILS UNKNOWN
startdatefrom	dateTime		Optional	Retrieve transaction history created from this start date.
startdateto	dateTime		Optional	Retrieve transaction history created until this stop date
datefrom	dateTime		Optional	Retrieve transaction history finalized (SUCCESSFUL, FAILED) from this start date.
dateto	dateTime		Optional	Retrieve transaction history finalized (SUCCESSFUL, FAILED) until this stop date.
direction	string		Optional	Retrieve transaction history with the specified direction, that is, incoming or outgoing transfers. Possible values are: IN OUT
otherfri	string		Optional	The FRI of the other party in transfer, could be from or to depending on direction. Validated with IsFRI.
quoteid	string		Optional	The ID of the quote to use.

Response

A success response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<v1_0:gettransactionhistoryresponse xmlns:iso="urn:iso:std:iso:20022:tech:xsd"
  xmlns:ns2="http://www.ericsson.com/em/emm/v1_1/common"
  xmlns:fic="http://www.ericsson.com/em/emm/financial/v1_0/common"
  xmlns:ns1="http://www.ericsson.com/em/emm/v1_0/common"
  xmlns:v1_0="http://www.ericsson.com/em/emm/financial/v1_0">
  <totalcount>0</totalcount>
  <historylist>
    <entry>
```



2 HTTP APIs

```

<financialtransactionid>financialtransactionid0</financialtransactionid>
<transactionstatus>SUCCESSFUL</transactionstatus>
<transfertype>TRANSFER_FEE</transfertype>
<startdate>2006-05-04T18:13:51.0Z</startdate>
<commitdate>2006-05-04T18:13:51.0Z</commitdate>
<fxrate>0</fxrate>
<initiatinguser>initiatinguser0</initiatinguser>
<realuser>realuser0</realuser>
<initiatingaccountholder>initiatingaccountholder0</initiatingaccountholder>
<realaccountholder>realaccountholder0</realaccountholder>
<from>from0</from>
<fromaccount>fromaccount0</fromaccount>
<fromamount>
  <amount>0</amount>
  <currency>currency0</currency>
</fromamount>
<fromfee>
  <amount>0</amount>
  <currency>currency1</currency>
</fromfee>
<fromexternalfee>
  <amount>0</amount>
  <currency>currency2</currency>
</fromexternalfee>
<fromdiscount>
  <amount>0</amount>
  <currency>currency3</currency>
</fromdiscount>
<frompromotion>
  <amount>0</amount>
  <currency>currency4</currency>
</frompromotion>
<fromloyfee>0</fromloyfee>
<fromloyreward>0</fromloyreward>
<frompromotionrefund>
  <amount>0</amount>
  <currency>currency5</currency>
</frompromotionrefund>
<fromnote>fromnote0</fromnote>
<fromavailablebalance>
  <amount>0</amount>
  <currency>currency6</currency>
</fromavailablebalance>
<fromtotalbalance>
  <amount>0</amount>
  <currency>currency7</currency>
</fromtotalbalance>

```



2 HTTP APIs

```

<fromcommittedbalance>
  <amount>0</amount>
  <currency>currency8</currency>
</fromcommittedbalance>
<fromaccountholder>fromaccountholder0</fromaccountholder>
<to>to0</to>
<toaccount>toaccount0</toaccount>
<toamount>
  <amount>0</amount>
  <currency>currency9</currency>
</toamount>
<tofee>
  <amount>0</amount>
  <currency>currency10</currency>
</tofee>
<toexternalfee>
  <amount>0</amount>
  <currency>currency11</currency>
</toexternalfee>
<topromotion>
  <amount>0</amount>
  <currency>currency12</currency>
</topromotion>
<toloyfee>0</toloyfee>
<toloyreward>0</toloyreward>
<topromotionrefund>
  <amount>0</amount>
  <currency>currency13</currency>
</topromotionrefund>
<todiscountrefund>
  <amount>0</amount>
  <currency>currency14</currency>
</todiscountrefund>
<tomessage>tomessage0</tomessage>
<toavailablebalance>
  <amount>0</amount>
  <currency>currency15</currency>
</toavailablebalance>
<tototalbalance>
  <amount>0</amount>
  <currency>currency16</currency>
</tototalbalance>
<tocommittedbalance>
  <amount>0</amount>
  <currency>currency17</currency>
</tocommittedbalance>
<maininstructionid>-1073741773</maininstructionid>

```



2 HTTP APIs

```

<instructionid>-1073741773</instructionid>
<externaltransactionid>externaltransactionid0</externaltransactionid>
<transactiontext>transactiontext0</transactiontext>
<tofeerefund>
  <amount>0</amount>
  <currency>currency18</currency>
</tofeerefund>
<fromfeerefund>
  <amount>0</amount>
  <currency>currency19</currency>
</fromfeerefund>
<toaccountholder>toaccountholder0</toaccountholder>
</entry>
<entry>
  <financialtransactionid>financialtransactionid1</financialtransactionid>
  <transactionstatus>SUCCESSFUL</transactionstatus>
  <transfertype>TRANSFER_FEE</transfertype>
  <startdate>2006-05-04T18:13:51.0Z</startdate>
  <commitdate>2006-05-04T18:13:51.0Z</commitdate>
  <fxrate>0</fxrate>
  <initiatinguser>initiatinguser1</initiatinguser>
  <realuser>realuser1</realuser>
  <initiatingaccountholder>initiatingaccountholder1</initiatingaccountholder>
  <realaccountholder>realaccountholder1</realaccountholder>
  <from>from1</from>
  <fromaccount>fromaccount1</fromaccount>
  <fromamount>
    <amount>0</amount>
    <currency>currency20</currency>
  </fromamount>
  <fromfee>
    <amount>0</amount>
    <currency>currency21</currency>
  </fromfee>
  <fromexternalfee>
    <amount>0</amount>
    <currency>currency22</currency>
  </fromexternalfee>
  <fromdiscount>
    <amount>0</amount>
    <currency>currency23</currency>
  </fromdiscount>
  <frompromotion>
    <amount>0</amount>
    <currency>currency24</currency>
  </frompromotion>
  <fromloyalty>0</fromloyalty>

```



2 HTTP APIs

```

<fromloyreward>0</fromloyreward>
<frompromotionrefund>
  <amount>0</amount>
  <currency>currency25</currency>
</frompromotionrefund>
<fromnote>fromnote1</fromnote>
<fromavailablebalance>
  <amount>0</amount>
  <currency>currency26</currency>
</fromavailablebalance>
<fromtotalbalance>
  <amount>0</amount>
  <currency>currency27</currency>
</fromtotalbalance>
<fromcommittedbalance>
  <amount>0</amount>
  <currency>currency28</currency>
</fromcommittedbalance>
<fromaccountholder>fromaccountholder1</fromaccountholder>
<to>to1</to>
<toaccount>toaccount1</toaccount>
<toamount>
  <amount>0</amount>
  <currency>currency29</currency>
</toamount>
<tofee>
  <amount>0</amount>
  <currency>currency30</currency>
</tofee>
<toexternalfee>
  <amount>0</amount>
  <currency>currency31</currency>
</toexternalfee>
<topromotion>
  <amount>0</amount>
  <currency>currency32</currency>
</topromotion>
<toloyfee>0</toloyfee>
<toloyreward>0</toloyreward>
<topromotionrefund>
  <amount>0</amount>
  <currency>currency33</currency>
</topromotionrefund>
<todiscountrefund>
  <amount>0</amount>
  <currency>currency34</currency>
</todiscountrefund>

```




2 HTTP APIs

```

<tomessage>tomessage1</tomessage>
<toavailablebalance>
  <amount>0</amount>
  <currency>currency35</currency>
</toavailablebalance>
<tototalbalance>
  <amount>0</amount>
  <currency>currency36</currency>
</tototalbalance>
<tocommittedbalance>
  <amount>0</amount>
  <currency>currency37</currency>
</tocommittedbalance>
<maininstructionid>-1073741773</maininstructionid>
<instructionid>-1073741773</instructionid>
<externaltransactionid>externaltransactionid1</externaltransactionid>
<transactiontext>transactiontext1</transactiontext>
<tofeerefund>
  <amount>0</amount>
  <currency>currency38</currency>
</tofeerefund>
<fromfeerefund>
  <amount>0</amount>
  <currency>currency39</currency>
</fromfeerefund>
<toaccountholder>toaccountholder1</toaccountholder>
</entry>
</historylist>
<loyalty>
  <generated>0</generated>
  <consumed>0</consumed>
  <newbalance>0</newbalance>
</loyalty>
</v1_0:gettransactionhistoryresponse>

```

Table 2-60 describes parameters in a getTransactionhistory response message body.

Table 2-60 Parameters in a getTransactionhistory response message body

Parameter	Type	Length	Level of Requirement	Description
totalcount	integer	-	Optional	Total amount of transaction history entries available with the specified condition.
entry	gettransactionhistoryentrytype[0...unbounded]	-	Optional	Contains a list of transactions, each transaction contains type, amount, date and so on. Contains the main summary of a transaction history event.



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
loyalty	loyalty	-	Optional	

Table 2-61 describes parameters in a gettransactionhistoryentrytype response message body.

Table 2-61 Parameters in a gettransactionhistoryentrytype response message body

Parameter	Type	Length	Level of Requirement	Description
financialtransactionid	string		Optional	The financial transaction ID. It's a number, in ASCII digits. Digit groups may be separated by hyphen characters (-, ASCII code number 45). An empty string is allowed.
transactionstatus	string		Optional	The status of the transaction. Possible values are: <ul style="list-style-type: none">• SUCCESSFUL• PENDING• sFAILED



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
transfertype	string		Optional	<p>The transfer type.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • DEPOSIT • REVERSE_DEPOSIT • WITHDRAWAL • TRANSFER • CASH_IN • CASH_OUT • TRANSFER_TO_VOUCHER • TRANSFER_FROM_VOUCHER • PAYMENT_SEND • PAYMENT_RECEIVE • COMMISSIONING • FEE • PERIODIC_FEE • CANCEL_VOUCHER • VIEW_ACTIVE_TOKEN • VIEW_TOKEN • EXPIRE_VOUCHER • BALANCE_ENQUIRY • TRANSACTION_HISTORY_ENQUIRY • REMITTANCE • TRANSFER_TO_INVITATION • TRANSFER_FROM_INVITATION • CANCEL_INVITATION • CASH_OUT_ATM • TRANSFER_FROM_VOUCHER_ATM • PICKUP • REVERSAL • REVERSAL_FEE • REVERSAL_LOY • REVERSAL_COMMISSIONING • REVERSAL_TAX • REVERSAL_DISCOUNT • REVERSAL_PROMOTION • ADJUSTMENT • ADJUSTMENT_FEE • ADJUSTMENT_LOY • ADJUSTMENT_COMMISSIONING
Issue 11 (2015-06-18)		Copyright 2015 © Mobile Telephone Networks. All rights reserved.		180



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
startdate	dateTime	-	Optional	The start date and time of the transaction.
commitdate	dateTime	-	Optional	The date and time at which the transaction was completed.
fxrate	decimal	-	Optional	The foreign exchange rate.
initiatinguser	string		Optional	The execution ID of the user that initiated the transaction. It's a string, containing any number of letters or digits, at-signs, hyphens,underscores or periods (full stop). Other whitespace or punctuation characters are not allowed. An empty string is not allowed.
realuser	string		Optional	The execution ID of the real user that initiated the transaction. It's a string, containing any number of letters or digits, at-signs, hyphens,underscores or periods (full stop). Other whitespace or punctuation characters are not allowed. An empty string is not allowed.
initiatingaccountholder	string		Optional	The Identity of the account holder that initiated the transaction if it was initiated by an account holder
realaccountholder	string		Optional	The Identity of the real account holder that is effected by the transaction if it was initiated by an account holder.
from	string		Optional	The sending user's default FRI.
fromaccount	string		Optional	The sending account FRI.
fromamount	moneydetail-type	-	Optional	The amount withdrawn from the sending account.
fromfee	moneydetail-type	-	Optional	The fee charged from the sending account.
fromexternalfee	moneydetail-type	-	Optional	The external fee charged from the sending account.
fromdiscount	moneydetail-type	-	Optional	The discount amount that has been applied on the sending account.
frompromotion	moneydetail-type	-	Optional	The promotion amount that has been applied on the sending account
fromloyaltyfee	decimal	-	Optional	The sender's loyalty points fee charged for the transaction.
fromloyaltyreward	decimal	-	Optional	The sender's loyalty points reward earned for the transaction.



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
fromnote	string		Optional	The sender's note.
fromavailablebalance	moneydetail - type		Optional	The sending account's available balance after the transaction was completed.
fromtotalbalance	moneydetail - type		Optional	The sending account's total balance after the transaction was completed.
fromcommittedbalance	moneydetail - type		Optional	The sending account's committed balance after the transaction was completed.
fromaccountholder	string		Optional	The Identity of the sending accountholder.
to	string		Optional	The receiving user's FRI or the receiving account's FRI.
toaccount	string		Optional	The receiving account FRI.
toamount	moneydetail - type		Optional	The amount deposited into the receiving account.
tofee	moneydetail - type		Optional	The fee charged from the receiving account.
toexternalfee	moneydetail - type		Optional	The external fee charged from the receiving account.
topromotion	moneydetail - type		Optional	The promotion amount that has been applied on the receiving account.
toloyaltyfee	decimal	-	Optional	The receiver's loyalty points fee charged for the transaction.
toloyaltyreward	decimal	-	Optional	The receiver's loyalty points reward earned for the transaction.
tomessage	string		Optional	The receiver's message.
toavailablebalance	moneydetail - type		Optional	The receiving account's available balance after the transaction was completed.
tocommittedbalance	moneydetail - type		Optional	The receiving account's committed balance after the transaction was completed.
instructionid	long	-	Optional	The financial instruction ID
maininstructionid	long	-	Optional	The main instruction ID
externaltransactionid	string		Optional	External transaction ID for the operation.



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
transactiontext	string	-	Optional	Text describing the transaction.
totalbalance	moneydetailtype	-	Optional	The receiving account's total balance after the transaction was completed.
tofeerefund	moneydetailtype	-	Optional	Previous fee that was refunded to the receiver during this transaction.
fromfeerefund	moneydetailtype	-	Optional	Previous fee that was refunded to the sender during this transaction
frompromotionrefund	moneydetailtype	-	Optional	Previous sender promotion that was refunded to the system during this transaction.
topromotionrefund	moneydetailtype	-	Optional	Previous receiver promotion that was refunded to the system during this transaction.
todiscountrefund	moneydetailtype	-	Optional	Previous receiver discount that was refunded to the system during this transaction.
toaccountholder	string	-	Optional	The Identity of the receiving accountholder.

Table 2-62 describes parameters in a moneydetailtype type.

Table 2-62 Parameters in a moneydetailtype type.

Parameter	Type	Length	Level of Requirement	Description
amount	decimal	-	Mandatory	The monetary amount. Validated with IsAmount (allowNull=false).
currency	string	3	Mandatory	The currency, as an ISO 4217 formatted string. Validated with IsCurrencyCode. Parameter cannot be NULL.

Table 2-63 describes parameters in a loyalty type

Table 2-63 Parameters in a loyalty type

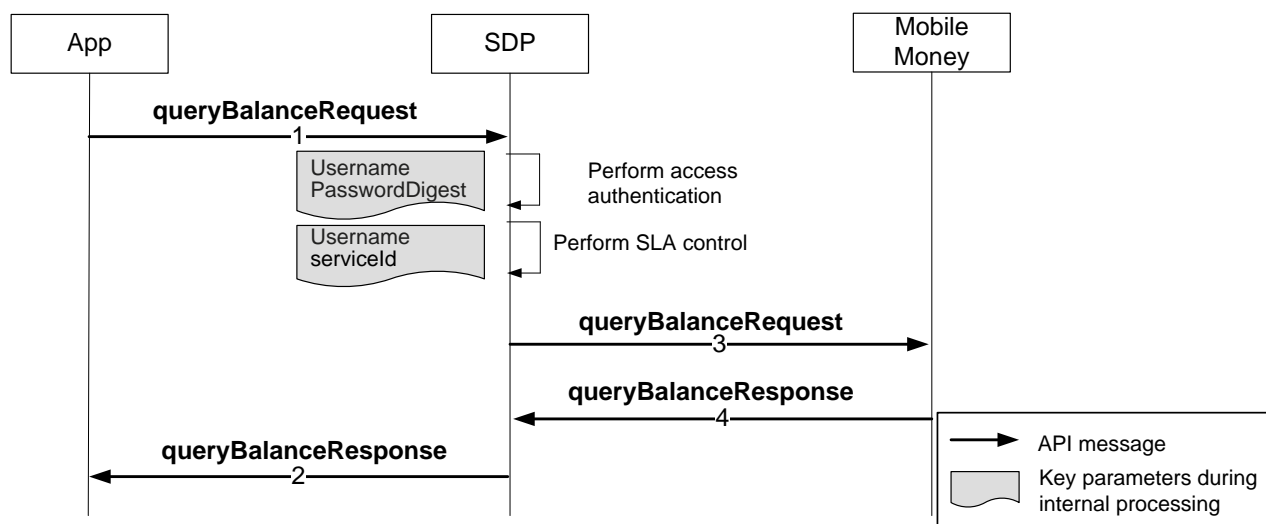
Parameter	Type	Length	Level of Requirement	Description
generated	decimal	-	Optional	The rewarded loyalty, generated amount.
consumed	decimal	-	Optional	The loyalty fee, consumed amount.
newbalance	decimal	-	Optional	The loyalty balance after execution.



2 HTTP APIs

Error Codes

For details about the error codes, see 2.7 API Error Response.

2.6.2 queryBalance*Process***Figure 2-25** Process of queryBalance*Function*

Obtain the balance for a specific account

Request URI

The request URI is the destination URI of `queryBalance` messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/queryBalance

In the format, IP and Port indicate the service IP address and HTTP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/queryBalance HTTP/1.1
Content-Type:text/xml
Content-Length:569
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
  
```



2 HTTP APIs

```
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Cookie: sessionId=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 8611111111
X-HW-Extension: k1=v1;k2=v2
Signature:43AD232FD45FF
```

```
<?xml version="1.0" encoding="UTF-8"?>
<v1_0:getbalancerequest xmlns:ns1="urn:iso:std:iso:20022:tech:xsd"
  xmlns:op="http://www.ericsson.com/em/emm/v1_0/common"
  xmlns:v1_0="http://www.ericsson.com/em/emm/financial/v1_0" version="1.0">
  <fri>fri0</fri>
  <quoteid>quoteid0</quoteid>
</v1_0:getbalancerequest>
```

Message Body Parameters

Table 2-64 describes parameters in a queryBalance message body.

Table 2-64 Parameters in a queryBalance message body

Parameter	Type	Length	Level of Requirement	Description
fri	string		Optional	Used for a specific account, for example commissioning account balance.
quoteid	string		Optional	The ID of the quote to use.

Response

A success response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<v1_0:getbalanceresponse xmlns:ns1="urn:iso:std:iso:20022:tech:xsd"
  xmlns:op="http://www.ericsson.com/em/emm/v1_0/common"
  xmlns:v1_0="http://www.ericsson.com/em/emm/financial/v1_0">
  <balance>
    <amount>0</amount>
    <currency>currency0</currency>
  </balance>
  <loyalty>
    <generated>0</generated>
    <consumed>0</consumed>
    <newbalance>0</newbalance>
  </loyalty>
</v1_0:getbalanceresponse>
```

Table 2-65 describes parameters in a queryBalance response message body.



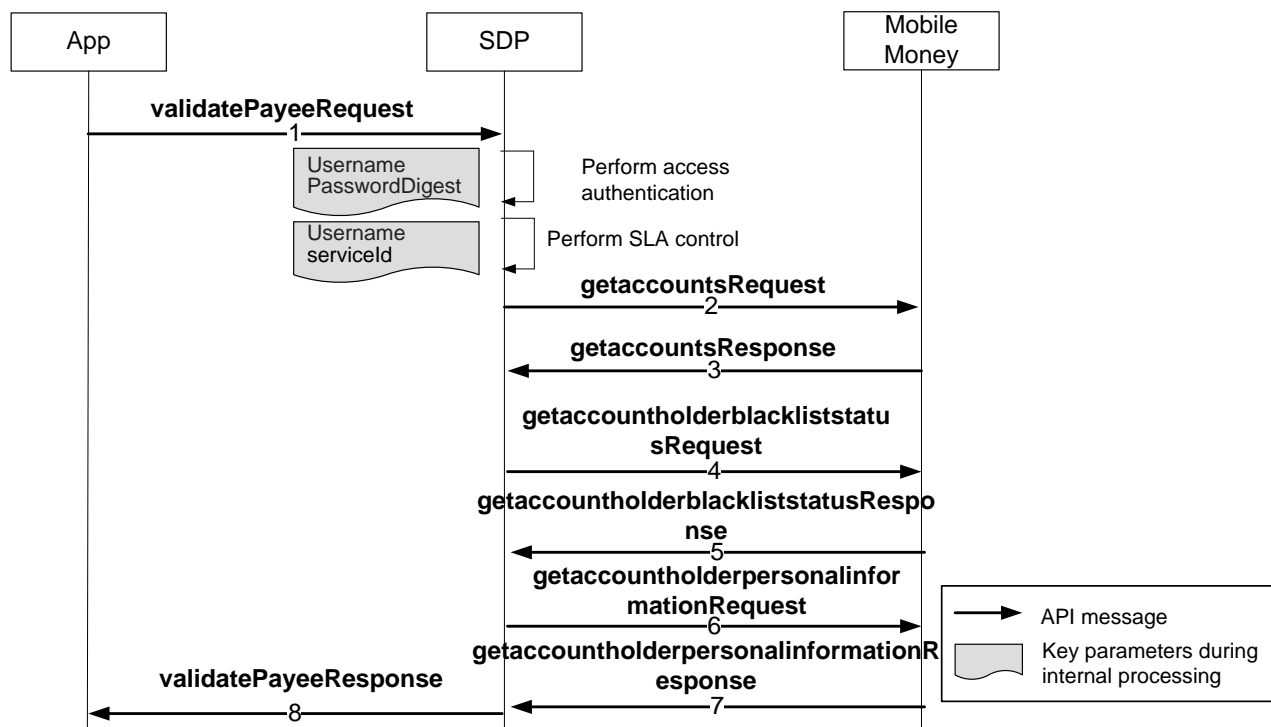
2 HTTP APIs

Table 2-65 Parameters in a queryBalance response message body

Parameter	Type	Length	Level of Requirement	Description
balance	long	-	Optional	Account holder's available account balance. Refer to Table-51
loyalty	loyalty	-	Optional	The loyalty information for the sender. Refer to Table-52

Error Codes

For details about the error codes, see 2.7 API Error Response.

2.6.3 validatePayee*Process***Figure 2-26** Process of validatePayee

3rd Party invokes the validatePayee interface towards SDP to validate if customer account is registered, active and can transact. SDP will then invoke multiple APIs towards ECW, consolidate its responses and send validatePayee response to partner.



2 HTTP APIs

Function

Validate the account holder information from 3rd party providers before executing a deposit or debiting customer account.

Request URI

The request URI is the destination URI of validatePayee messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/validatePayee

In the format, IP and Port indicate the service IP address and HTTP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```
POST /mom/mt/validatePayee HTTP/1.1
Content-Type:text/xml
Content-Length:569
Host:127.0.0.1
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Cookie: sessionId=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 8611111111
X-HW-Extension: k1=v1;k2=v2

<?xml version="1.0" encoding="UTF-8"?>
<v1_0:validatepayeerequest xmlns:ns1="http://www.ericsson.com/em/emm/v1_0/common"
xmlns:prc="http://www.ericsson.com/em/emm/provisioning/v1_0/common"
xmlns:ns2="urn:iso:std:iso:20022:tech:xsd"
xmlns:v1_0="http://www.ericsson.com/em/emm/provisioning/v1_0">
  <identity>identity0</identity>
  <extension>
  </extension>
</v1_0:validatepayeerequest>
```

Message Body Parameters

Table 2-66 describes parameters in a validatePayee message body.

Table 2-66 Parameters in a validatePayee message body

Parameter	Type	Length	Level of Requirement	Description
identity	string		Mandatory	A valid identity associated with the account holder.
extension	extensiontype	-	Optional	Used for sending any additional



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
				information to Partner. For details about extensiontype, see Table 2-32.

Response

A success response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<v1_0:validatepayeeresponse xmlns:ns1="http://www.ericsson.com/em/emm/v1_0/common"
xmlns:prc="http://www.ericsson.com/em/emm/provisioning/v1_0/common"
xmlns:ns2="urn:iso:std:iso:20022:tech:xsd"
xmlns:v1_0="http://www.ericsson.com/em/emm/provisioning/v1_0">
  <transactionStatus>false</transactionStatus>
  <accountslist>
    <account>
      <accountstatus>ACTIVE</accountstatus>
      <accounttype>MOBILE_MONEY</accounttype>
      <profilename>profilename0</profilename>
      <description>description0</description>
    </account>
    <account>
      <accountstatus>ACTIVE</accountstatus>
      <accounttype>MOBILE_MONEY</accounttype>
      <profilename>profilename1</profilename>
      <description>description1</description>
    </account>
  </accountslist>
  <blackliststatus>BLACKLISTED</blackliststatus>
  <information>
    <birth>
      <date>2006-05-04</date>
    </birth>
  </information>
  <extension>
  </extension>
</v1_0:validatepayeeresponse>
```

Table 2-67 describes parameters in a validatePayee response message body.

Table 2-67 Parameters in a validatePayee response message body

Parameter	Type	Length	Level of Requirement	Description
transactionStatus	boolean	-	Mandatory	True/False based on overall response.



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
accountslist	accountslist	-	Conditional	Returns all accounts connected to a specific account holder. Mandatory for true/success response. For details about accountslist, see Table 2-68.
blackliststatus	blackliststatus	-	Conditional	Returns the black list status of the identity sent in the request. Mandatory for true/success response. String based enumeration, Possible values are: <ul style="list-style-type: none"> • BLACKLISTED • WHITELISTED • NONE
information	personalinformation	-	Conditional	Returns personal information connected to the identity sent in the request. Mandatory for true/success response. For details about personalinformation, see Table 2-70.
extension	extensiontype	-	Optional	Used for sending any additional information to Partner. For details about extensiontype, see Table 2-32.

Table 2-68 describes parameters in an accountslist type.

Table 2-68 Parameters in an accountslist type

Parameter	Type	Length	Level of Requirement	Description
account	accountdetailtype[0...unbounded]	-	Optional	Contains information about the account and its balance. For details about accountdetailtype, see Table 2-69.

Table 2-69 describes parameters in an accountdetailtype type.



2 HTTP APIs

Table 2-69 Parameters in in an accountdetailstype type

Parameter	Type	Length	Level of Requirement	Description
accountstatus	accountstatus		Optional	The status of the account. accountstatus is string based enumeration, Possible values are: <ul style="list-style-type: none"> • ACTIVE • BLOCKED • CLOSED • UNAVAILABLE
accounttype	accounttype		Optional	The type of account. accounttype is string based enumeration, Possible values are: <ul style="list-style-type: none"> • MOBILE_MONEY • COMMISSIONING • LOYALTY_POINTS • TAX • LOAN • UNKNOWN
profilename	xs:string		Optional	The name of the profile of the account.
description	xs:string		Optional	Description of the account.

Table 2-70 describes parameters in a personalinformation type.

Table 2-70 Parameters in in a personalinformation type

Parameter	Type	Length	Level of Requirement	Description
birth	birthinformation		Optional	The account holder's birth information. For details about birthinformation, see Table 2-71.

Table 2-71 describes parameters in a birthinformation type.

Table 2-71 Parameters in in a birthinformation type

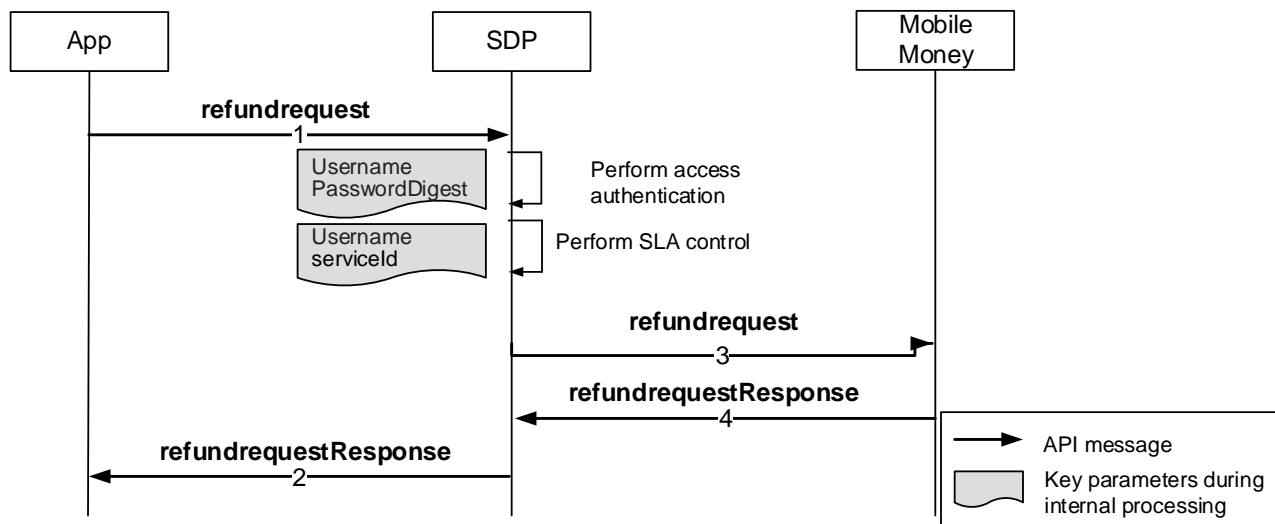
Parameter	Type	Length	Level of Requirement	Description
date	xs:date	-	Optional	The account holder's birth date. minDate=1890-01-01



2 HTTP APIs

Error Codes

For details about the error codes, see 2.7 API Error Response.

2.6.4 refundrequest*Process***Figure 2-27** Process of refundrequest

3rd Party invokes the refundrequest interface towards SDP to refundrequest if customer want to **refund** money or loyalty points to an account holder. SDP will then invoke refundrequest API towards ECW, and send refundrequest response to partner.

Function

refund money or loyalty points to an account holder.

Request URI

The request URI is the destination URI of refundrequest messages sent by the App to the SDP. The URI is provided by the SDP in the following format:

http://IP:Port/mom/mt/refundrequest

In the format, IP and Port indicate the service IP address and HTTP port number of the API provided by the SDP. Contact carriers to obtain the IP address and port number.

Request Message Body Example

```

POST /mom/mt/refundrequest HTTP/1.1
Content-Type:text/xml
Content-Length:569
Host:127.0.0.1
  
```



2 HTTP APIs

```
Authorization: WSSE realm="SDP", profile="UsernameToken"
X-WSSE: UsernameToken Username="bob", PasswordDigest=" weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==", Nonce="
WScqanjCEAC4mQoBE07sAQ==", Created="2012-07-26T11:31:26Z"
X-RequestHeader: request ServiceId="35000001000012", TransId=" 200903241230451000000123456789" ,
LinkId="A15Y89Z32T99", FA="8613912345678"
Cookie: sessionId=default8fcee064690b45faa9f8f6c7e21c5e5a
Msisdn: 8611111111
X-HW-Extension: k1=v1;k2=v2
```

```
<?xml version="1.0" encoding="UTF-8"?>
<n:refundrequest xmlns:n="http://www.ericsson.com/em/emm/financial/v1_0"
xmlns:ns1="http://www.ericsson.com/em/emm/v1_0/common"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="String">
<amount>
  <amount>3.14</amount>
  <currency>USD</currency>
</amount>
<refundernote>1234</refundernote>
<receivermessage>refund</receivermessage>
<financialtransactionid>2147483647</financialtransactionid>
<quoteid>2233</quoteid>
</n:refundrequest>
```

Message Body Parameters

Table 2-6672 describes parameters in a refundrequest message body.

Table 2-72 Parameters in a refundrequest message body

Parameter	Type	Length	Level of Requirement	Description
amount	moneydetailst ype		Mandatory	The amount to refund. Parameter can not be NULL. Validated with IsPositiveAmount. See details in Table 2-73
refundernote	string	256	Optional	A descriptive note for sender transaction history. Validated with Length (max=256). Validated with IsRestrictedString.
receivermessage	string	256	Optional	A descriptive note for receiver transaction history. Validated with Length (max=256). Validated with IsRestrictedString
financialtransactio nid	long		Mandatory	Transaction ID of the debit or payment transaction for which refund is requested. Validated with IsPositiveLong (allowNull=false) (allowZero=false).



2 HTTP APIs

Parameter	Type	Length	Level of Requirement	Description
quoteid	quoteid	-	Optional	The ID of the quote to use. See details in Table 2-74

Table 2-73 Parameters in a moneydetailstype structure

Parameter	Type	Length	Level of Requirement	Description
amount	decimal	-	Mandatory	The monetary amount. Validated with IsAmount (allowNull=false) (allowZero=true).
currency	string	10	Mandatory	The currency, as an ISO 4217 formatted string. Validated with IsCurrencyCode. Parameter can not be NULL.

Table 2-74 Parameters in a quoteid structure

Type	Length	Level of Requirement	Description
string	-	Mandatory	Validated with IsLongString (allowNegativeValue=false). Parameter can not be NULL.

Response

A success response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<n:refundresponse xmlns:n="http://www.ericsson.com/em/emm/financial/v1_0"
xmlns:ns1="http://www.ericsson.com/em/emm/v1_0/common"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" >
  <financialtransactionid>2147483647</financialtransactionid>
  <refunderbalanceafter>
    <amount>3.14</amount>
    <currency>USD</currency>
  </refunderbalanceafter>
  <refunderfee>
    <amount>3.14</amount>
    <currency>USD</currency>
  </refunderfee>
  <loyalty>
    <generated>3.14</generated>
    <consumed>3.14</consumed>
    <newbalance>3.14</newbalance>
  </loyalty>
</n:refundresponse>
```




2 HTTP APIs

Table 2-75 describes parameters in a refundrequest response message body.

Table 2-75 Parameters in a refundrequest response message body

Parameter	Type	Length	Level of Requirement	Description
financialtransactionid	long	-	Mandatory	The financial transaction ID of the transfer. Validated with IsPositiveLong (allowNull=false) (allowZero=false).
refunderbalanceafter	moneydetailtype	-	Optional	Balance of the refunder account after the transaction. Validated with IsPositiveAmount, see Table 2-73.
refunderfee	moneydetailtype	-	Optional	The fee paid by the sender, currency is the same as the senders account defined by the sender FRI. Validated with IsPositiveAmount. See Table 2-73.
loyalty	Loyalty	-	Optional	Loyalty information generated, see Table 2-76.

Table 2-76 describes parameters in an Loyalty type.

Table 2-76 Parameters in in an Loyalty type

Parameter	Type	Length	Level of Requirement	Description
generated	decimal	-	Optional	The rewarded loyalty, generated amount.
consumed	decimal	-	Optional	The loyalty fee, consumed amount.
newbalance	decimal	-	Optional	The loyalty balance after execution.

Error Codes

For details about the error codes, see 2.7 API Error Response.

2.7 API Error Response

When an error occurs, the server sends an error response to the client. This topic provides an error response example and describes parameters in the response.

Example

```
<tns:errorResponse xmlns:tns="http://www.ericsson.com/lwac" errorCode="errorCode0">
  <arguments name="name0" value="value0"/>
  <arguments name="name1" value="value1"/>
</tns:errorResponse>
```

Error Code List

Table 2-77 defines the error code.



2 HTTP APIs

Table 2-77 Parameters in error response message body

errorCode	arguments.name	arguments.value	Exception Scene
SYSTEM_OVERLOAD	resource_name	Name of the resource with dynamic flow control errors, such as CPU, memory, and queue.	Dynamic flow control error
SYSTEM_OVERLOAD	item_name	Name of the item with static flow control errors, such as SMS and MMS.	Static flow control error
EXCEED_LICENSE	item_name	Name of the item with license control errors, such as SMS and MMS.	License control error
SYSTEM_ERROR	connection	Error details, such as internal connection broken or waiting response timeout .	Internal message interaction error, such as disconnection or timeout
AUTHORIZATION_FAILED	SPID	SP ID, which is the value of UserName in the X-WSSE header.	Access authentication failure
NOT_AUTHORIZED	SPID ServiceID ApiName	SP ID, service ID, or API name.	Capability authentication failure
MANDATORY_FIELD_MISSING	Parameter_name	Name of a parameter that fails to be verified.	Parameter verification failure due to mandatory parameter missing
INVALID_PARAMETER	Parameter_name	Name of a parameter that fails to be verified.	Parameter verification failure due to incorrect format
VIOLATE_SLA	SPID ServiceID SLAID	SP ID, service ID, or SLA ID.	SLA control failure
NO_ROUTE	route	Route information not found.	Route query failure
CONNECTION_ERROR	connection	External connection	External message interaction error, such



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
		error details, such as connection broken or waiting response timeout .	as disconnection or timeout
ACCOUNT_ALREADY_MEMBER_OF_GROUP	-	-	This account is already a member of this group
ACCOUNT_COULD_NOT_BE_REMOVED_FROM_BANKDOMAIN	-	-	Account could not be removed from Bank domain
ACCOUNT_HOLDER_ADDITIONAL_INFORMATION_NAME_ALREADY_EXIST	-	-	The specified account holder's additional information name already exist
ACCOUNT_HOLDER_ADDITIONAL_INFORMATION_NAME_IN_USE	-	-	The specified account holder's additional information name cannot be deleted because it is in use
ACCOUNT_HOLDER_ALREADY_MEMBER_OF_GROUP	-	-	This account holder is already a member of this group
ACCOUNT_HOLDER_DOES_NOT_EXIST	-	-	AccountHolder does not exist
ACCOUNT_HOLDER_NOT_LOGGED_IN	-	-	This operation requires the user to be logged in
ACCOUNT_HOLDER_PROFILE_DOES_NOT_HAVE_REQUESTED_ACCOUNT_TYPE	-	-	AccountHolder profile does not have requested account
ACCOUNT_HOLDER_STATUS_IS_NOT_VALID_FOR_ADD_ACCOUNT_OPERATION	-	-	AccountHolder status should be blocked or active
ACCOUNT_NOT_FOUND	-	-	Account not found
ACCOUNT_ROUTE_ALREADY_EXISTS	-	-	Could not create account route since Account route with provided account route reference and currency already exists
ACCOUNT_ROUTE_NOT_FOUND	-	-	Could not delete account route since Account route with provided account



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
			route reference and currency does not exists
ACCOUNT_TYPE_NOT_SUPPORTED	-	-	Account type is not supported
ACCOUNTHOLDER_ACCOUNT_REFERENCE_ALREADY_EXISTS	-	-	The account holder already has a connection to the specified account
ACCOUNTHOLDER_ACTIVATION_FAILED	-	-	Activation of account holder failed
ACCOUNTHOLDER_ANY_ONE_MANDATORY_FIELD_REQUIRED	-	-	Could not perform operation since at least one of the mandatory field is required
ACCOUNTHOLDER_CANNOT_REMOVE_EMAIL	-	-	Could not perform operation since at least one of the mandatory field is required
ACCOUNTHOLDER_CANNOT_REMOVE_EXT_ID	-	-	Could not perform operation since at least one of the mandatory field is required
ACCOUNTHOLDER_CANNOT_REMOVE_MSISDN	-	-	Could not perform operation since at least one of the mandatory field is required
ACCOUNTHOLDER_CANNOT_REMOVE_USER_NAME	-	-	Could not perform operation since at least one of the mandatory field is required
ACCOUNTHOLDER_EUI_NOT_FOUND	-	-	Account holder could not be found
ACCOUNTHOLDER_MSISDN_NOT_FOUND	-	-	Operation requires account holder to have an MSISDN identity but it has none
ACCOUNTHOLDER_NOT_ACTIVE	-	-	Account holder is inactive
ACCOUNTHOLDER_NOT_CHILD	-	-	A specified account holder was not a child to another specified account holder
ACCOUNTHOLDER_NOT_FOUND	-	-	Account holder could not be found
ACCOUNTHOLDER_SETTINGS_ALREADY_EXISTS	-	-	The settings for this account holder already exists



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
ACCOUNTHOLDER_SETTINGS_MANDATORY	-	-	Account holder settings are mandatory
ACCOUNTHOLDER_SETTINGS_NOT_FOUND	-	-	Account holder settings not found
ACCOUNTHOLDER_WITH_ALIAS_ALREADY_EXISTS	-	-	Account holder could not be registered because an account holder with submitted Alias already exists
ACCOUNTHOLDER_WITH_EMAIL_ALREADY_EXISTS	-	-	Account holder could not be registered because an account holder with submitted E-mail already exists
ACCOUNTHOLDER_WITH_EMAIL_NOT_FOUND	-	-	Account holder with given e-mail could not be found
ACCOUNTHOLDER_WITH_EMPLOYEE_ID_ALREADY_EXISTS	-	-	Account holder cannot have the same employee id as another employee working under the same employer
ACCOUNTHOLDER_WITH_EXTERNALID_ALREADY_EXISTS	-	-	Account holder could not be registered because an account holder with submitted External ID already exists
ACCOUNTHOLDER_WITH_FRI_NOT_FOUND	-	-	Account holder with given FRI could not be found
ACCOUNTHOLDER_WITH_IMSI_ALREADY_EXISTS	-	-	Account holder IMSI could not be updated because it already exists
ACCOUNTHOLDER_WITH_MSISDN_ALREADY_EXISTS	-	-	Account holder could not be registered because an account holder with submitted MSISDN already exists
ACCOUNTHOLDER_WITH_MSISDN_NOT_FOUND	-	-	Account holder with given MSISDN could not be found
ACCOUNTHOLDER_WITH_USERNAME_ALREADY_EXISTS	-	-	Account holder could not be registered because an account holder with submitted Username already exists
ADJUSTMENTS_FROM_ACCOUNT_NOT_ALLOWED	-	-	Adjustments from account not allowed
ADJUSTMENTS_NOT_ALLOWED	-	-	Adjustments not allowed on account
ADJUSTMENTS_TO_ACCOUNT	-	-	Adjustments to account not allowed



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
COUNT_NOT_ALLOWED			
ALIAS_NOT_FOUND	-	-	No Alias with this name was found in the system
ALREADY_MEMBER_IN_ANOTHER_BANKDOMAIN	-	-	Could not perform operation since the resource already belongs to another Bank domain
AMBIGUOUS_CURRENCY	-	-	The FRI refers to more than one account
AMOUNT_INVALID	-	-	Either amount and currency are incompatible or currency is invalid
AUTHORIZATION_ACCOUNT_NOT_ALLOWED_TO_PERFORM_ACTION	-	-	Account holder is not allowed to perform this operation
AUTHORIZATION_ACCOUNTHOLDER_ALREADY_ACTIVATED	-	-	Account holder is already activated
AUTHORIZATION_ACCOUNTHOLDER_NOT_ACTIVE	-	-	Account holder is not active
AUTHORIZATION_CURRENCY_BALANCE_TOO_LOW	-	-	Insufficient amount
AUTHORIZATION_FAILED	-	-	Authorization failed
AUTHORIZATION_INVALID_ACCOUNT_TYPE_COMBINATION	-	-	Performing this operation with this account type combination is not allowed
AUTHORIZATION_INVALID_ACCOUNTHOLDER_HIERARCHY	-	-	Setting an account holder as its own parent or a creating a cycle in the hierarchy is not allowed
AUTHORIZATION_INVALID_DEBIT	-	-	Authorization failed
AUTHORIZATION_MAX_TRANSFER_ACCOUNT_THROUGHPUT	-	-	Maximum transfer account throughput amount including fee threshold exceeded
AUTHORIZATION_MAX_TRANSFER_ACCOUNT_THROUGHPUT_RECEIVER	-	-	Maximum transfer account throughput amount including fee threshold exceeded for receiver
AUTHORIZATION_MAX_TRANSFER	-	-	Maximum transfer account throughput



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
RANSFER_ACCOUNT_THROUGHPUT_SENDER			amount including fee threshold exceeded for sender
AUTHORIZATION_MAX_TRANSFER_AMOUNT	-	-	Maximum transfer amount threshold exceeded
AUTHORIZATION_MAX_TRANSFER_AMOUNT_FEE	-	-	Maximum transfer amount including fee threshold exceeded
AUTHORIZATION_MAX_TRANSFER_AMOUNT_FEE_RECEIVER	-	-	Maximum transfer amount including fee threshold exceeded for receiver
AUTHORIZATION_MAX_TRANSFER_AMOUNT_FEE_SENDER	-	-	Maximum transfer amount including fee threshold exceeded for sender
AUTHORIZATION_MAX_TRANSFER_AMOUNT_RECEIVER	-	-	Maximum transfer amount threshold exceeded for receiver
AUTHORIZATION_MAX_TRANSFER_AMOUNT_SENDER	-	-	Maximum transfer amount threshold exceeded for sender
AUTHORIZATION_MAX_TRANSFER_TIMES	-	-	Maximum transfer counter threshold exceeded
AUTHORIZATION_MAXIMUM_AMOUNT_ALLOWED_TO_RECEIVE	-	-	Maximum receive amount threshold exceeded
AUTHORIZATION_MAXIMUM_AMOUNT_ALLOWED_TO_SEND	-	-	Maximum send amount threshold exceeded
AUTHORIZATION_MAXIMUM_AMOUNT_TO_APPROVE	-	-	Authorization failed. Maximum threshold limit reached for Approver
AUTHORIZATION_MINIMUM_AMOUNT_ALLOWED_TO_SEND	-	-	Amount below minimum amount threshold
AUTHORIZATION_MINIMUM_AMOUNT_ALLOWED_TO_SEND_ACCOUNT	-	-	Amount below minimum amount threshold for account
AUTHORIZATION_MINIMUM_AMOUNT_ALLOWED_TO	-	-	Amount below minimum amount



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
O_SEND_ACCOUNT_REFERENCE			threshold for account reference
AUTHORIZATION_NOT_REGISTERED_STATUS	-	-	Account holder is not registered
AUTHORIZATION_RECEIVER_ACCOUNT_NO_DEPOSIT	-	-	Receivers account does not allow deposit
AUTHORIZATION_RECEIVER_FIRST_DEPOSIT_MINIMUM_AMOUNT_ACCOUNT	-	-	Could not perform operation because amount is below minimum first deposit amount threshold
AUTHORIZATION_RECEIVER_MAX_ALLOWED_BALANCE	-	-	Maximum balance threshold for receiver exceeded
AUTHORIZATION_RECEIVING_ACCOUNT_NOT_ACTIVE	-	-	Receivers account is not active
AUTHORIZATION_RECEIVING_ACCOUNT_UNAVAILABLE	-	-	Receiving account is unavailable
AUTHORIZATION_SENDER_ACCOUNT_NO_WITHDRAWAL	-	-	Withdrawal from recipient account not allowed
AUTHORIZATION_SENDER_ACCOUNT_NOT_ACTIVE	-	-	Senders account is not active
AUTHORIZATION_SENDER_MIN_ALLOWED_BALANCE	-	-	Minimum balance threshold for sender exceeded
AUTHORIZATION_SENDING_ACCOUNT_UNAVAILABLE	-	-	Sending account is unavailable
BANK_ACCOUNT_NOT_REGISTERED	-	-	A bank account with given bank code and account number could not be found
BANK_ACCOUNT_ROUTE_NOT_FOUND	-	-	A route to the wanted bank account could not be found



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
BANKDOMAIN_MISSING	-	-	No bank domain set in the system.
BANKDOMAIN_NOT_FOUND	-	-	Bank domain with given ID was not found
BANKDOMAIN_USED	-	-	Could not perform operation since the Bank domain is currently in use
BANKDOMAIN_WITH_NAME_ALREADY_EXISTS	-	-	Could not perform operation since Bank domain with given name already exists
BLOCK_ACCOUNT_FAILED	-	-	Account could not be blocked
CAN_NOT_RECEIVE_CASHIN	-	-	Receiving account holder has insufficient permission to receive cashin
CAN_NOT_RECEIVE_CASHOUT	-	-	Receiving account holder has insufficient permission to receive cashout
CERTIFICATE_ALREADY_UPDATED	-	-	Certificate already updated for the user
CERTIFICATE_CREDENTIALS_ALREADY_EXIST	-	-	Certificate credentials already exist
CERTIFICATE_CREDENTIALS_NOT_FOUND	-	-	Certificate credentials do not exist
CLOSE_ACCOUNT_FAILED	-	-	Account could not be closed
COMPANY_INFORMATION_ALREADY_EXIST	-	-	The company information already exist
COMPANY_INFORMATION_DOES_NOT_EXIST	-	-	The company information does not exist
CONTACT_ALREADY_EXISTS	-	-	Contact was not added since it already exists
CONTACT_NOT_FOUND	-	-	Contact with given FRI could not be found
COULD_NOT_CREATE_ACCOUNTHOLDER_SETTINGS	-	-	Could not create account holder
COULD_NOT_PERFORM_ADD_ACCOUNT_OPERATION	-	-	Could not perform add account operation



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
COULD_NOT_PERFORM_APPROVAL_OF_OPERATION_PLEASE_RETRY_OPERATION_FROM_BEGINNING	-	-	Failed to approve the operation, please retry the initial operation again
COULD_NOT_PERFORM_ATM_PARTIAL_REVERSAL_OPERATION	-	-	ATM partial reversal operation could not be performed
COULD_NOT_PERFORM_ATM_REDEEM_VOUCHER_OPERATION	-	-	ATM redeem voucher operation could not be performed
COULD_NOT_PERFORM_ATM_ROLLBACK_VOUCHER_OPERATION	-	-	ATM rollback voucher operation could not be performed
COULD_NOT_PERFORM_OPERATION	-	-	Could not perform the operation. Additional reason attached.
COULD_NOT_PERFORM_OPERATION_ACCOUNT_NOT_LINKED	-	-	Could not perform the operation, the account has to be linked with sp account.
COULD_NOT_PERFORM_TRANSACTION	-	-	Transaction could be performed
COULD_NOT_RECEIVE_REMITTANCE	-	-	Receive remittance failed
COULD_NOT_UPDATE_ACCOUNTHOLDER_SETTINGS	-	-	Could not perform update of account holder settings
COULD_NOT_UPDATE_PENDING_PAYMENT	-	-	Could not perform update of pending payment status
COUNTER_CONFIGURATION_NOT_FOUND	-	-	The specified counter configuration does not exist.
CREDENTIAL_IS_NOT_SUSPENDED	-	-	The credential is not in the expected suspended state
CREDENTIAL_NOT_ACTIVE	-	-	Operation not allowed because credential not in active state
CREDENTIAL_REPEATED_SECRET_DOES_NOT_MATCH	-	-	Repeated secrets does not match each other



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
ATCH			
CREDENTIAL_TYPE_NOT_FOUND	-	-	Given credential type does not exist in the system
CREDENTIALS_NOT_FOUND	-	-	The user does not have any credential of the given type
CREDENTIALS_NOT_FOUND	-	-	Given credentials could not be found
CURRENCIES_DONT_MATCH	-	-	For this operation currency conversion is not allowed
CURRENCY_NOT_SUPPORTED	-	-	Given currency is not supported in the system
CUSTODY_ACCOUNT_ALREADY_EXISTS	-	-	Custody Account already exists
CUSTODY_ACCOUNT_CERTIFICATE_ALREADY_EXISTS	-	-	Custody account certificate already exist
CUSTODY_ACCOUNT_CERTIFICATE_COULD_NOT_BE_DELETED	-	-	Custody account certificate could not deleted
CUSTODY_ACCOUNT_CERTIFICATE_COULD_NOT_BE_EMPTY	-	-	Custody account certificate could not be empty
CUSTODY_ACCOUNT_CERTIFICATE_COULD_NOT_BE_USED_FOR_NON_REPUDIATION	-	-	Custody account certificate could not be used for non-repudiation
CUSTODY_ACCOUNT_CERTIFICATE_DOES_NOT_EXIST	-	-	Custody account certificate does not exist
CUSTODY_ACCOUNT_CERTIFICATE_ERROR	-	-	Custody account certificate error
CUSTODY_ACCOUNT_CERTIFICATE_NO_SUCH_ALGORITHM	-	-	Unsupported algorithm for custody account certificate
CUSTODY_ACCOUNT_CERTIFICATE_WITH_SAME_ID	-	-	Custody account certificate with same id already exist



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
D_ALREADY_EXIST			
CUSTODY_ACCOUNT_NAME_AND_BANKDOMAIN_SAME_AS_BEFORE	-	-	Name and Bank Domain are same as before
CUSTODY_ACCOUNT_NOT_FOUND	-	-	Could not find CustodyAccount since Custody Account with provided account number and bank code does not exists
CUSTODY_ACCOUNT_STILL_CONTAINS_MONEY	-	-	Custody account still contains money
CUSTODY_ACCOUNT_STILL_HAVE_PENDING_TRANSACTION	-	-	This Custody account still have pending transaction
CUSTODY_ACCOUNT_UPDATE_FAILED_NAME_AND_BANKDOMAIN_MISSING	-	-	Could not update custody account as name and bank domain both are missing
DATEFORMAT_INVALID	-	-	DateTime string could not be parsed
DIFFERENT_BANKDOMAINS	-	-	Account holder and account must belong to the same bank domain
DOCUMENT_DOES_NOT_EXIST	-	-	The document does not exist
DOCUMENT_URL_CONFIGURATION_ERROR	-	-	The document rules has been incorrectly configured in the system
DOCUMENT_URL_INVALID	-	-	This document URL is invalid
DOCUMENT_URL_NOT_ALLOWED	-	-	This document URL is not allowed
END_DATE_IN_FUTURE	-	-	End date can not occur in the future
EXPIRED_OR_INVALID_TRANSACTION_ID	-	-	Transaction ID is either invalid or this transaction has expired
EXTERNAL_ERROR_MESSAGE	-	-	This error originates from an external system
EXTERNAL_SERVICE_PROVIDER_NOT_FOUND	-	-	External service provider with given User name could not be found



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
FAILED_TO_CREATE_NETTING_REPORT	-	-	Netting report error
FIELD_TOO_LONG	-	-	Data in element is too long
FINANCIAL_CONFIGURATION_DOES_NOT_EXIST	-	-	Financial Configuration does not exist
FINANCIAL_CONTROLLER_TRANSACTION_NOT_FOUND	-	-	Neither transaction with transaction ID nor original transaction with original transaction ID could be found
FIRST_NAME_NOT_ALLOWED_IN_PASSWORD	-	-	Using first name in password is not allowed
FLOAT_TRANSFER_EXEMPTION_ALREADY_EXISTS	-	-	The specified float transfer exemption rule already exists
FRI_CAN_ONLY_HAVE_ONE_OWNER	-	-	Setting multiple owners of an FRI is not allowed
FRI_INVALID	-	-	FRI string is incorrectly formatted
FRI_REFERENCE_NOT_FOUND	-	-	
FROM_DATE_AFTER_TO_DATE	-	-	Operation requires that start date has to occur before end date
GROUP_ALREADY_USED_BY_ANOTHER_BANKDOMAIN	-	-	Could not perform operation since the Group is already used by another Bank domain
GROUP_MEMBER_COULD_NOT_BE_REMOVED	-	-	Group member could not be removed from group
GROUP_MEMBER_COULD_NOT_BE_REMOVED_FROM_BANKDOMAIN	-	-	Group member could not be removed from Bank domain
GROUP_MEMBER_NOT_FOUND	-	-	Group member with given ID could not be found in this group
GROUP_USED_BY_BANKDOMAIN	-	-	Group used by Bank domain is not allowed to be removed
HIGH_CONTENTION_ACCOUNT_NOT_SUPPORTED_AS_MIRRORED_ACCOUNT	-	-	It's not possible to mirror a High Contention account in SDP since it doesn't have a balance.



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
HOME_CHARGING_REGION_NOT_FOUND	-	-	Home charging region with given ID was not found
HOME_CHARGING_REGION_USED	-	-	Could not perform operation since the home charging region is currently in use
HOME_CHARGING_REGION_WITH_NAME_ALREADY_EXISTS	-	-	Could not perform operation since Home Charging Region with given name already exists
ID_LENGTH_OUT_OF_BOUNDS	-	-	Length of identification is not within length boundaries
IDENTIFICATION_ID_AND_TYPE_ALREADY_EXIST	-	-	Duplicate identification
IDENTITY_INVALID	-	-	Identity string is incorrectly formatted
IDTYPE_NOT_SUPPORT	-	-	ID type not supported by ISO 20022
ILLEGAL_CREDENTIAL_STATUS_CHANGE	-	-	The credential status change is illegal
IMSI_INVALID	-	-	Format of IMSI is invalid
INACTIVE_ACCOUNT	-	-	Inactive receiver account
INCORRECT_PASSWORD	-	-	Incorrect password
INCORRECT_PIN	-	-	Incorrect pin code
INCORRECT_SECURITY_ANSWER	-	-	Security answer was not correct
INCORRECT_TRANSACTION_STATUS	-	-	Transaction state is incorrect for this operation
INTERNAL_ERROR	-	-	An internal error caused the operation to fail
INTERNAL_LOAN_ACCOUNT_NOT_FOUND	-	-	Could not find Internal Loan Account with given Id
INTERNAL_LOAN_AMOUNT_MIN_LARGER_THAN_AMOUNT_MAX	-	-	Amount Min cannot be larger than Amount Max
INTERNAL_LOAN_APPLICATION_IS_NOT_VALID	-	-	The state of this Internal Loan Application is not valid for this operation



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
INTERNAL_LOAN_APPLICATION_NOT_FOUND	-	-	Could not find Internal Loan Application
INTERNAL_LOAN_MAXIMUM_NUMBER_OF_APPLICATIONS	-	-	Maximum number of Internal Loan Applications reached
INTERNAL_LOAN_PRODUCT_ALREADY_EXIST	-	-	Internal Loan Product already exist
INTERNAL_LOAN_PRODUCT_IS_NOT_ACTIVE	-	-	The Internal Loan Product status is not active
INTERNAL_LOAN_PRODUCT_NOT_ALLOWED	-	-	The given Internal Loan Product is not allowed to be used
INTERNAL_LOAN_PRODUCT_NOT_FOUND	-	-	Could not find Internal Loan Product with given identity and product code
INTERNAL_LOAN_PROVIDER_ALREADY_EXIST	-	-	Internal Loan Provider already exist
INTERNAL_LOAN_PROVIDER_NOT_FOUND	-	-	Could not find Internal Loan Provider with given identity
INVALID_ACCOUNT_FRI	-	-	The FRI must be of account type
INVALID_ACCOUNT_HOLDER_ADDITIONAL_INFORMATION_NAME	-	-	The specified account holder's additional information name is not valid or does not exist
INVALID_ACCOUNTHOLDER_SETTINGS	-	-	Account holder settings are invalid
INVALID_APPROVAL_TRANSACTION_STATUS	-	-	Approval transaction has invalid state
INVALID_ATM_OPERATION	-	-	Invalid ATM operation
INVALID_CERTIFICATE	-	-	Certificate is invalid
INVALID_CHOSEN_DELIVERY_METHOD	-	-	Invalid Delivery method
INVALID_CURRENCY	-	-	Currency is either not valid or used incorrectly
INVALID_FIELD_VALUE	-	-	Element has invalid format



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
INVALID_PROFILE	-	-	Profile validation
INVALID_PROFILE_COUNTER	-	-	Profile counter validation
INVALID_PROFILE_NAME	-	-	Profile name validation
INVALID_PROFILE_THRESHOLD	-	-	Profile threshold validation
INVALID_RECEIVER	-	-	Invalid receiving end user
INVALID_REMITTANCE_STATUS	-	-	Incorrect remittance status
INVALID_SCHEDULED_TRANSACTION_DATE	-	-	A transaction can only be scheduled from a future date
INVALID_SCHEDULED_TRANSACTION_STATUS	-	-	Invalid scheduled transaction status for the requested operation
INVALID_SETTLEMENT_DEPOSIT_ID	-	-	Settlement deposit with given ID could not be found
INVALID_SETTLEMENTID	-	-	Settlement with given ID could not be found
INVALID_SIGNATURE	-	-	The provided signature is not correct
INVALID_STATUS	-	-	Current status is not as expected
INVALID_TRANSACTION_TYPE	-	-	Invalid Transaction type
INVALID_USER_APPROVAL_TYPE	-	-	Invalid User (or financial) approval type or not registered
LAST_BANKDOMAIN_NOT_ALLOWED_TO_REMOVE	-	-	Bank domain could not be removed since it is the last remaining
LIMITED_RETRIES_FOR_SERVICE_PROVIDER_NOT_ALLOWED	-	-	Limited retries are not allowed for the specified service provider
LOAN_APPLICATION_NOT_FOUND	-	-	Could not find Loan Application with given approval ID
LOAN_PROVIDER_NOT_FOUND	-	-	Could not find Loan Provider with given name



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
MANDATORY_FIELD_MISSING	-	-	Mandatory element is missing
MAX_ACCOUNTS_LIMIT_REACHED	-	-	Reached limit for max accounts
MAX_NUMBER_OF_CONTACTS_EXCEEDED	-	-	Maximum number of contacts threshold exceeded
MAX_NUMBER_OF_DOCUMENT_EXCEEDED	-	-	Maximum number of documents exceeded
MISSING_ACTOR_MSISDN_FOR_EXTERNAL_RATING	-	-	Actor MSISDN is missing
MISSING_EXCHANGE_RATE	-	-	Exchange rate for conversion between currencies does not exist.
MORE_THAN_ONE_MUTUALLY_EXCLUSIVE_OPTIONAL_PARAMETERS_SET	-	-	More than one of a set of optional but mutual parameters were set.
MSISDN_ALREADY_USED_BY_POINTOFSALE	-	-	Given msisdn already used by a point of sale
NO_ACCESS	-	-	User has no access permission
NONASCII_CHARACTERS_IN_PASSWORD	-	-	Using non ASCII characters in password is not allowed
NONDIGIT_CHARACTERS_IN_PINCODE	-	-	Non-digit characters in pin code
NOT_ALLOWED_TO_CHANGE_BANKDOMAIN	-	-	It is not allowed to change bank domain for an account holder
NOT_AUTHORIZED	-	-	The logged in user does not have sufficient permissions to perform this operation
NOT_ENOUGH_FUNDS	-	-	Not enough funds or loyalty points to perform operation
ONE_CUSTODY_ACCOUNT_MUST_EXIST	-	-	One custody account must exist
ONLY_ONE_REFUND_PER_DEBIT_IS_ALLOWED	-	-	Only one refund per debit is allowed



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
OPERATION_NOT_SUPPORTED_ON_EXTERNAL_ACCOUNT	-	-	Operation not supported on external account
OPERATION_NOT_SUPPORTED_ON_MIGRATED_ACCOUNT	-	-	Operation not supported on a migrated account
OTHER_FINANCIAL_ERROR	-	-	Other financial error
OTP_FOR_BANK_ACCOUNT_VERIFICATION_MISSING	-	-	Request did not contain any OTP
OTP_FOR_EMAIL_VERIFICATION_MISSING	-	-	Request did not contain any OTP
OTP_FOR_MSISDN_VERIFICATION_MISSING	-	-	Request did not contain any OTP
OTP_INVALID	-	-	Format of OTP is invalid
OTP_NOT_FOUND	-	-	OTP not found, has expired or is used for the wrong operation
PASSWORD_ALREADY_USED_BEFORE	-	-	Password has already been used before
PASSWORD_TOO_SHORT	-	-	Password too short
PENDING_COMMISSION_NOT_FOUND	-	-	Could not find Pending Commission
PICKUP_REFERENCE_ID_NOT_FOUND	-	-	Pickup with given MTCN could not be found
PINCODE_ALREADY_USED_BEFORE	-	-	PIN code has already been used
PINCODE_DENIED_BY_MATCHING_BIRTH_DATE	-	-	PIN code denied by matching birth date
PINCODE_DENIED_BY_MATCHING_IDENTIFICATION_NUMBER	-	-	PIN code denied by matching identification number
PINCODE_DENIED_BY_MATCHING_MSISDN	-	-	PIN code denied by matching MSISDN



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
PINCODE_DENIED_BY_MATCHING_REGULAR_EXPRESSION	-	-	PIN code denied by matching regular expression
PINCODE_HAS_EXPIRED	-	-	Wrong profile
PINCODE_TOO_LONG	-	-	Pin code too long
PINCODE_TOO_SHORT	-	-	Pin code too short
POINTOFSALE_NOT_ACQUIRED	-	-	The point of sale with given msisdn is not acquired.
PREAPPROVAL_NOT_FOUND	-	-	The pre-approval with given ID could not be found
PREAPPROVALS_NOT_FOUND	-	-	No pre-approvals found
PROFILE_ALREADY_EXISTS	-	-	Profile name validation
PROFILE_JSON_INVALID_FILE	-	-	Json File validation
PROFILE_MISMATCH	-	-	A profile mismatch between for example a provided counter configuration (belonging to one profile) and provided profile reference (having a different profile)
PROFILE_NOT_FOUND	-	-	Profile not found
PROFILE_TYPE_MISMATCH	-	-	Profile type validation
PROFILE_VERSION_MANDATORY	-	-	Profile version validation
PROFILE_VERSION_MISMATCH	-	-	Profile version validation
PROFILE_WITH_NAME_DOES_NOT_EXIST	-	-	Tag was not found
PROVIDER_CATEGORY_LINKED_WITH_ACCOUNT_HOLDER	-	-	Provider Category can not be deleted as it is linked with account holder
PROVIDER_CATEGORY_NAME	-	-	No provider category with this name was



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
NOT_FOUND			found in the system
QUEUED_FOR_APPROVAL	-	-	Request is queued for approval
QUOTE_ALREADY_USED	-	-	Quote has already been used
QUOTE_EXPIRED	-	-	Quote has expired
QUOTE_NOT_FOUND	-	-	Quote not found
QUOTE_NOT_SUPPORTED	-	-	Quote is not supported with the combination of parameters
QUOTE_PROCESSING	-	-	Quote is already processing
REACTIVATE_ACCOUNT_FAILED	-	-	Account could not be re-activated
RECONCILIATION_PROCESS_ERROR	-	-	Failed to process reconciliation file.
RECURRING_SCHEDULED_TRANSACTION_EXCEEDS_MAX_DAYS	-	-	Recurring scheduled transaction can only be scheduled a limited number of days into the future
REFERENCE_ID_ALREADY_IN_USE	-	-	This client reference ID is already used
REFERENCE_ID_NOT_FOUND	-	-	A client reference with this ID could not be found
REFUND_AMOUNT_CANNOT_EXCEED_THE_DEBIT_AMOUNT	-	-	Refund amount cannot be larger than the debit amount
RELATIVE_DOES_NOT_EXIST	-	-	The relative does not exist
REMITTANCE_ID_DOES_NOT_EXIST	-	-	Remittance Id does not exist in dbase
REMITTANCE_NOT_FOUND	-	-	Remittance not found, Invalid provider name or remittance id
REMITTANCE_NOT_POSSIBLE	-	-	Remittance not possible because given data is invalid.
REQUIRED_FIELD_MISSING	-	-	Element needed for this invocation is missing



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
REQUIRED_PASSWORD_CRITERIA_NOT_MATCH	-	-	Password criteria configuration invalid
RESOURCE_NOT_ACTIVE	-	-	Required resource is not active
RESOURCE_NOT_AVAILABLE	-	-	FRI not available
RESOURCE_NOT_FOUND	-	-	FRI not found
RESOURCE_TEMPORARILY_LOCKED	-	-	Required resource is temporarily locked
REVERSAL_ALREADY_PERFORMED_ON_THIS_TRANSACTION	-	-	Reversal has already been performed on this transaction
REVERSAL_NOT_ALLOWED_ON_THIS_TRANSACTION_TYPE	-	-	Reversal is not allowed on this type of transaction
REVERSAL_SUB_TRANSACTION_INVALID	-	-	Reversal not possible since sub transaction is invalid
REVERSED_USER_ID_NOT_ALLOWED_IN_PASSWORD	-	-	Using reversed user id in password is not allowed
ROLLBACK_NOT_POSSIBLE	-	-	Transaction could not be performed since neither of the parties supports rollback
SCHEDULED_TRANSACTION_EXCEEDS_MAX_DAYS	-	-	Transaction can only be scheduled a limited number of days into the future
SCHEDULED_TRANSACTION_NOT_FOUND	-	-	The requested scheduled transaction could not be found
SELF_SERVICE_NOTIFICATION_DUPLICATE	-	-	This self service notification already exists.
SELF_SERVICE_NOTIFICATION_LIMIT_EXCEEDS_HARD_LIMIT	-	-	The notification limit exceeds the hard limit
SELF_SERVICE_NOTIFICATION_NOT_FOUND	-	-	
SELF_SERVICE_NOTIFICATION_TOO_MANY	-	-	Max number of self serviced notifications reached, cannot create another.



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
SERVICE_TEMPORARILY_UNAVAILABLE	-	-	The operation could not be performed right now but might succeed later
SESSION_INVALID	-	-	The session has timed out
SETTLEMENT_AMOUNT_DO_NOT_MATCH	-	-	The response amount does not match with the amount in the Payment Instruction
SETTLEMENT_ANOTHER_SETTLEMENT_BEING_PROCESSED	-	-	Another settlement task is ongoing.
SETTLEMENT_CLOSING_BALANCE_MISMATCH	-	-	Balance reconciliation failed. Provided closing balance does not match with the expected closing balance of custody account
SETTLEMENT_CONFIGURATION_CERTIFICATE_DOES_NOT_EXIST_FOR_CUSTODY_ACCOUNT	-	-	Certificate does not exist for custody account not able to verify signature of settlement file
SETTLEMENT_CONFIGURATION_CUSTODY_ACCOUNT_ALREADY_EXISTS_WITH_SETTLEMENT_CONFIGURATION	-	-	Custody account already exists with settlement configuration
SETTLEMENT_CONFIGURATION_DOES_NOT_EXIST_FOR_CUSTODY_ACCOUNT	-	-	Could not find settlement Configuration for custody account
SETTLEMENT_CONFIGURATION_FOR_IDENTIFIER_ALREADY_EXISTS	-	-	Settlement configuration for identifier already exists
SETTLEMENT_CONFIGURATION_INSTRUCTION_FORMATTER_CLASS_NOT_FOUND	-	-	Instruction formatter class not found
SETTLEMENT_CONFIGURATION_NOT_FOUND	-	-	Settlement configuration not found
SETTLEMENT_CONFIGURATION_SETTLEMENT_PARSER_CLASS_NOT_FOUND	-	-	Settlement parser class not found



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
D			
SETTLEMENT_CONFIGURATION_UNABLE_TO_REMOVE_CUSTODY_ACCOUNT_ID	-	-	Unable to remove custody account id from settlement configuration
SETTLEMENT_CURRENCY_DO_NOT_MATCH	-	-	Currency does not match with custody account's currency.
SETTLEMENT_CUSTODY_ACCOUNT_DO_NOT_MATCH	-	-	Settlement process failed because the custody accounts information do not match.
SETTLEMENT_DEPOSIT_ALREADY_REVERSED	-	-	The deposit has already been reversed.
SETTLEMENT_DEPOSIT_IN_INVALID_STATE	-	-	The deposit is in an invalid state.
SETTLEMENT_DEPOSIT_NOT_FOUND	-	-	The deposit was not found.
SETTLEMENT_DEPOSIT_TRANSACTION_DETAILS_MISSING	-	-	The transaction details of the deposit where not found.
SETTLEMENT_DUPLICATE_RECORD_FOUND	-	-	Duplicated deposit or reverse deposit record found.
SETTLEMENT_FILE_DIGEST_VERIFICATION_ERROR	-	-	Digest does not exist or digest verification failed for settlement file
SETTLEMENT_FILE_FORMAT_ERROR	-	-	Error in parsing settlement file
SETTLEMENT_FILE_SIGNATURE_VERIFICATION_ERROR	-	-	Signature does not exist or signature verification failed for settlement file
SETTLEMENT_IDENTIFIER_DO_NOT_MATCH	-	-	Payment Instruction Response caller identifier mismatch
SETTLEMENT_IDENTIFIER_NOT_FOUND	-	-	Settlement identifier not found
SETTLEMENT_INVALID_AMOUNT	-	-	Invalid or no amount in Payment Instruction Response.



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
SETTLEMENT_INVALID_RECEIVER	-	-	Settlement error due to invalid receiver. There is no such account holder or custody account exists.
SETTLEMENT_NOT_READY	-	-	Settlement is in wrong state
SETTLEMENT_ONLINE_SETTLEMENT_NOT_ENABLED_FOR_CUSTODY_ACCOUNT	-	-	Online settlement not enabled for custody account
SETTLEMENT_OPENING_BALANCE_MISMATCH	-	-	Opening Balance reconciliation failed
SETTLEMENT_PAYMENT_INSTRUCTION_IN_WRONG_STATE	-	-	Wrong state for Settlement PaymentInstruction.
SETTLEMENT_PAYMENT_INSTRUCTION_NOT_FOUND	-	-	Settlement PaymentInstruction not found.
SETTLEMENT_PROCESS_ERROR	-	-	Failed to process settlement file.
SETTLEMENT_RECEIVING_ACCOUNT_DO_NOT_MATCH	-	-	The receiver account does not match
SETTLEMENT_TRANSFER_IN_WRONG_STATE	-	-	Wrong state for Settlement CustodyAccountsTransfers.
SOURCE_AND_TARGET_ARE_THE_SAME	-	-	Source is the same as the target
SOURCE_NOT_FOUND	-	-	Source FRI not found
SURNAME_NOT_ALLOWED_IN_PASSWORD	-	-	Using surname in password is not allowed
SUSPEND_ACCOUNT_FAILED	-	-	Not possible to put account in suspended state
TAG_WITH_NAME_ALREADY_EXISTS	-	-	Tag could not be created because a Tag with submitted name already exists
TAG_WITH_NAME_DO_NOT_EXISTS	-	-	Tag was not found



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
TARGET_AUTHORIZATION_ERROR	-	-	Target financial resource has reached counter or balance limits, has insufficient funds or is missing permissions
TARGET_NOT_FOUND	-	-	Target FRI not found
TAX_REPORT_NOT_FOUND	-	-	Tax report not found
THRESHOLD_CONFIGURATION_NOT_FOUND	-	-	The specified threshold configuration does not exist.
TOO_FEW_LOWERCASE_ALPHABETIC_CHARACTERS_IN_PASSWORD	-	-	Too few lower case alphabetic characters in password
TOO_FEW_NUMERICAL_DIGITS_IN_PASSWORD	-	-	Too few digits in password
TOO_FEW_SPECIAL_CHARACTERS_IN_PASSWORD	-	-	Too few special characters in password
TOO_FEW_UPPERCASE_ALPHABETIC_CHARACTERS_IN_PASSWORD	-	-	Too few upper case alphabetic characters in password
TOO_FEW_UPPERCASE_OR_LOWERCASE_ALPHABETIC_CHARACTERS_IN_PASSWORD	-	-	Too few upper or lower case alphabetic characters in password
TOO_MANY_CONSECUTIVE_DIGITS_IN_PINCODE	-	-	Too many consecutive digits in PIN code
TOO_MANY_IDENTICAL_CONSECUTIVE_CHARACTERS_IN_PASSWORD	-	-	Too many identical consecutive characters in password
TOO_MANY_REPEATED_DIGITS_IN_PINCODE	-	-	Too many repeated digits in PIN code
TRANSACTION_ID_OR_ORIGINAL_TRANSACTION_ID_MISSING	-	-	Neither transaction ID nor original transaction ID was given
TRANSACTION_NOT_COMPLETED	-	-	Transaction has not been completed and its state is either Pending or Failed



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
TRANSACTION_NOT_FOUND	-	-	Transaction not found
TRANSACTION_TIMED_OUT	-	-	Transaction timed out
TRANSFER_TYPE_AND_ACCOUNT_DO_NOT_MATCH	-	-	Receiving account is invalid for this type of transaction
TRANSFERTOBANK_NOT_SUPPORTED	-	-	Operation is not supported for this account
UNABLE_TO_CREATE_ACCOUNTHOLDER	-	-	Account holder could not be created
UNABLE_TO_CREATE_INTEREST_TIER	-	-	Interest tier could not be created
UNABLE_TO_MODIFY_INTEREST_TIER	-	-	Interest tier could not be modified
UNABLE_TO_REMOVE_INTEREST_TIER	-	-	Interest tier could not be removed
UNBLOCK_ACCOUNT_FAILED	-	-	Account could not be un-blocked
UNRESOLVED_REMITTANCE_ID_DOES_NOT_EXIST	-	-	Unresolved Remittance Id does not exist in dbase
UNSUSPEND_CREDENTIAL_COUNTER_THRESHOLD_REACHED	-	-	The maximum attempts of credential unsuspend for the given account holder has been reached
USER_ID_NOT_ALLOWED_IN_PASSWORD	-	-	Using user id in password is not allowed
VALIDATION_ERROR	-	-	The data provided in the request failed the validation check
VOUCHER_DATE_OF_BIRTH_SET	-	-	Voucher date of birth has been set
VOUCHER_EXPIRED	-	-	Voucher has expired
VOUCHER_EXTERNALID_SET	-	-	Voucher external id has been set
VOUCHER_FIRSTNAME_SET	-	-	Voucher first name has been set



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
ET			
VOUCHER_NOT_ACTIVE	-	-	Voucher is no longer active
VOUCHER_NOT_FOUND	-	-	Voucher with given identity could not be found in the system
VOUCHER_SECRET_CONTAINS_EXCLUDED_LETTERS	-	-	Voucher secret contains letters that has been excluded through system configuration
VOUCHER_SECRET_CONTAINS_ILLEGAL_CHARACTERS	-	-	Voucher secret contains characters that is not legal according to system configuration
VOUCHER_SECRET_CONTAINS_NONASCII_CHARACTERS	-	-	Voucher secret contains non ASCII characters
VOUCHER_SECRET_DIGITS_NOT_ALLOWED	-	-	Voucher secret can not contain digits
VOUCHER_SECRET_INVALID	-	-	Voucher secret is not correct
VOUCHER_SECRET_IS_EMPTY	-	-	Voucher secret element is empty
VOUCHER_SECRET_LETTERS_NOT_ALLOWED	-	-	Voucher secret can not contain letters
VOUCHER_SECRET_SYSTEM_CONFIGURATION_ERROR	-	-	The voucher secret rules has been incorrectly configured in the system
VOUCHER_SECRET_TOO_LONG	-	-	Voucher secret is longer than system configured maximum length
VOUCHER_SECRET_TOO_MANY_FAILED_REDEEM_ATTEMPTS	-	-	Too many redeem voucher attempts with incorrect secret has been performed
VOUCHER_SECRET_TOO_SHORT	-	-	Voucher secret is shorter than system configured minimum length
VOUCHER_SURNAME_SET	-	-	Voucher surname has been set
WRONG_PROFILE	-	-	Wrong profile



2 HTTP APIs

errorCode	arguments.name	arguments.value	Exception Scene
COUPONS_USER_ALLOWANCE_REACHED	-	-	Number of coupons allowed on the offer for the user has already been reached
COUPONS_USER_NOT_ELIGIBLE_FOR_OFFER	-	-	User is not eligible for offer
COUPONS_USER_PROFILE_NOT_EXIST	-	-	User profile does not exist

03

Appendix

3.1 The Currency and Atomic Charging Unit of each OpCo country

Country Name	Currency	Atomic Charging Unit
Afghanistan	AFN	pul
Benin	XOF	N/A
Bissau	CFA	N/A
Cameroon	XAF	XAF
Cote d'Ivoire	XOF	XOF
Cyprus	EUR	cent
Ghana	GHS	Pesewa (1GHS = 100Pesewa)
Guinea	GNF(Guinea Franc)	N/A
Liberia	LRD(Liberia Dollar)	N/A
N Sudan	SDG	Piaster
Nigeria	NGN	Kobo
Rwanda	Rwandan Franc (RWF)	RWF
SA	ZAR	N/A
Swaziland	SZL	cent
Uganda	Ugandan shilling (Code UGX)	UGX
Yemen	Yemeni Riyal	Fils (1 Riyal = 100 Fils)
Zambia	ZMK	ngwee, 1ZMK = 100Ngwee



3 Appendix

3.2 ISO 4217 Codes for the representation of currencies and funds

ENTITY	CURRENCY	ALPHABETI C CODE	NUMERI C CODE
Afghanistan	Afghani	AFA	4
Benin	CFA Franc BCEAO	XOF	952
Botswana	Pula	BWP	72
Cameroon	CFA Franc BEAC	XAF	950
Congo	CFA Franc BEAC	XAF	950
Cyprus	Cyprus Pound	CYP	196
Ethiopia	Ethiopian Birr	ETB	230
Ghana	Cedi	GHC	288
Guinea	Guinea Franc	GNF	324
Guinea-Bissau	Guinea-Bissau Peso	GWP	624
	CFA Franc BCEAO	XOF	952
Liberia	Liberian Dollar	LRD	430
Nigeria	Naira	NGN	566
Rwanda	Rwanda Franc	RWF	646
South Africa	Rand	ZAR	710
Sudan	Sudanese Dinar	SDD	736
Swaziland	Lilangeni	SZL	748
Uganda	Uganda Shilling	UGX	800
Yemen	Yemeni Rial	YER	886
Zambia	Kwacha	ZMK	894



3 Appendix

3.3 OpCoID List

MTN OpCo Name	OpCoID Value
Afghanistan	9301
Benin	22901
Bissau	24501
Botswana	26701
Cameroon	23701
Congo	24201
CoteD'Ivoire	22501
Cyprus	35701
Ghana	23301
Guinea Republic	22401
Liberia	23101
Nigeria	23401
Rwanda	25001
South Africa	2701
Sudan North	21101
Sudan South	24901
Swaziland	26801
Syria	96301
Uganda	25601
Yemen	96701
Zambia	26001

