|  |
| --- |
| ***BILL Technical specifications*** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Reference :** | **Bill-technical-specifications\_1.0.0.docx** | | |
| **Version :** | 1.0.0 | **date :** | 20/12/2016 |

| **Version** | **Written** | | **Purpose** | **Checked** | | **Validated** | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| date | by | date | by | date | by |
| 1.0.0 | 2016/12/16 |  | Document translation and update |  |  |  |  |

Content

[1 INTRODUCTION 3](#_Toc470026261)

[1.1 DOCUMENT PURPOSE 3](#_Toc470026262)

[1.2 FUNCTIONAL CONTEXT 3](#_Toc470026263)

[2 WEB SERVICES 3](#_Toc470026264)

[2.1 HEADER REQUEST PARAMETERS 3](#_Toc470026265)

[3 Payment API definition 4](#_Toc470026266)

[3.1 Resources Summary 4](#_Toc470026267)

[3.2 Structures 6](#_Toc470026268)

[3.2.1 Type: PaymentTransactionList 6](#_Toc470026269)

[3.2.2 Type: PaymentTransactionNotification 7](#_Toc470026270)

[3.2.3 Type: AmountTransaction 7](#_Toc470026271)

[3.2.4 Type: AmountReservationTransaction 9](#_Toc470026272)

[3.2.5 Type: PaymentAmount 11](#_Toc470026273)

[3.2.6 Type: ChargingMetaData 11](#_Toc470026274)

[3.2.7 Type: ChargingInformation 13](#_Toc470026275)

[3.2.8 Type: Subscription 14](#_Toc470026276)

[3.2.9 Type: SubscriptionInformation 15](#_Toc470026277)

[3.2.10 Type: SubscriptionMetaData 15](#_Toc470026278)

[3.3 Enumerations 16](#_Toc470026279)

[3.3.1 Enumeration: TransactionOperationStatus 16](#_Toc470026280)

[3.4 Values of the Link “rel” attribute 16](#_Toc470026281)

[3.5 Sequence diagrams 17](#_Toc470026282)

[3.5.1 Amount charge and refund transaction 17](#_Toc470026283)

[3.5.2 Amount reservation transaction 17](#_Toc470026284)

[3.5.3 Charge or refund amount and check transaction status 18](#_Toc470026285)

[3.5.4 Charge or refund amount and notify transaction status 19](#_Toc470026286)

[4 Detailed specification of the resources 20](#_Toc470026287)

[4.1 Resource: All payment transactions for an end user 21](#_Toc470026288)

[4.1.1 Request URL variables 21](#_Toc470026289)

[4.1.2 One phase payment (DIRECT DEBIT) 21](#_Toc470026290)

[4.1.3 Two phases payment (AUTORIZATION - CONFIRMATION) 25](#_Toc470026291)

[4.1.4 Client notification about payment transaction amount (not implemented yet) 32](#_Toc470026292)

[4.1.5 RETRIEVE TRANSACTIONS 32](#_Toc470026293)

[4.1.6 Create refund amount 37](#_Toc470026294)

[4.1.7 RETRIEVE SUBSCRIPTIONS 39](#_Toc470026295)

[4.2 STATUS CODES & EXCEPTIONS 42](#_Toc470026296)

[4.2.1 Status Codes 42](#_Toc470026297)

[4.2.2 Structure for error messages 42](#_Toc470026298)

[4.2.3 Service Exceptions 44](#_Toc470026299)

[4.2.4 Policy Exceptions 45](#_Toc470026300)

# INTRODUCTION

## DOCUMENT PURPOSE

This document aims to provide a communication technical interface between an agregator and SFR to bill services / content / products on the customer SFR invoice.

## FUNCTIONAL CONTEXT

A set of Web Services is available to the aggregator in a REST architecture. All Web Services are accessible via HTTP POST requests.

These Web Services are designed to bill services / content / products on the customer phone bill. Payment for such services / content / products is made possible by the creation of a new service and the registration of the aggregator as distributor.

The aggregator will have a login to be able to bill a customer from an ACR (or MSISDN) for the related service.

Payments are made in two steps. The first request is sent to reserve the amount followed by a second request to validate this transaction after confirmation by the aggregator of content delivery to the customer. Both URLs available are described in the paragraphs below.

# WEB SERVICES

As stated in the introduction, the aggregator must be registered as a distributor and so has:

* Login/password,
* Editor name,
* Service name,
* A service id (dedicated to billing).

## HEADER REQUEST PARAMETERS

The authentication method selected is the HTTP Basic authentication (using the login / password above).

The data format exchanged is json. It should therefore be specified in the header Content-Type.

Here is an example of header:

Accept:\*/\*

Accept-Charset:ISO-8859-1,utf-8;q=0.7,\*;q=0.3

Accept-Encoding:gzip,deflate,sdch

Accept-Language:fr-FR,fr;q=0.8,en-US;q=0.6,en;q=0.4

**Authorization:Basic aGlnaGNvOjEyMzQ1**

Connection:keep-alive

Content-Length:584

**Content-Type:application/json**

User-Agent:Mozilla/5.0 (Windows NT 6.1) AppleWebKit/535.2 (KHTML, like Gecko) Chrome/15.0.874.121 Safari/535.2

# Payment API definition

## Resources Summary

The “apiVersion” URL variable SHALL have the value “v2.1” to indicate that the API corresponds to this version of the specification.

The figure below visualizes the resource structure defined by this specification.

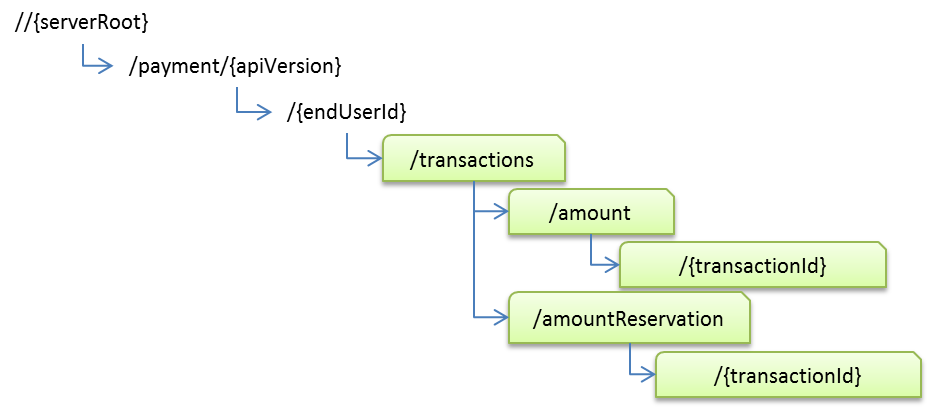


Figure 1 Resource structure defined by this specification

The following tables give detailed overview of the resources defined in this specification, the data type of their representation and the allowed HTTP methods.

Tableau 1 Purpose: to allow client to retrieve all payment transactions

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Resource | URL  Base URL:  //{serverRoot}/payment/{apiVersion} | Data Structures | HTTP verbs | | | |
| GET | PUT | POST | DELETE |
| All payment transactions for an end user | /{endUserId}/transactions | PaymentTransactionList | Return all completed and pending payment transactions (amount, amountReservation) | no | No | no |

Tableau 2 Purpose: to allow client to perform amount charge and refund transactions

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Resource | URL  Base URL:  //{serverRoot}/payment/{apiVersion} | Data Structures | HTTP verbs | | | |
| GET | PUT | POST | DELETE |
| All amount charge and refund transactions for an end user | /{endUserId}/transactions/amount | PaymentTransactionList (used for GET)  AmountTransaction  (used for POST) | Return all amount transactions details for a given end user | no | Create new transaction for a given end user | no |
| Individual amount charge or refund transaction for an end user | /{endUserId}/transactions/amount/{transactionId} | AmountTransaction | Return amount transaction details (charge, refund) | no | no | no |

Tableau 3 Purpose: to allow client to perform amount reservation transactions

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Resource | URL  Base URL:  //{serverRoot}/payment/{apiVersion} | Data Structures | HTTP verbs | | | |
| GET | PUT | POST | DELETE |
| All amount transactions for an end user | /{endUserId}/transactions/amountReservation | PaymentTransactionList (used for GET)  AmountReservationTransaction  (used for POST) | Return all amount reservation transactions details for a given end user | no | Create new reservation transaction for a given end user | no |
| Individual amount reservation transaction for an end user | /{endUserId}/transactions/amountReservation/{transactionId} | AmountReservationTransaction | Return amount reservation transaction details (amount, charge) | no | Charge reserved amount , increase reservation amount, release reserved amount | no |

Tableau 4 Purpose: to allow client to receive notifications about payment transactions

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Resource | URL  <specified by the client>  Base URL:  <specified by the client> | Data Structures | HTTP verbs | | | |
| GET | PUT | POST | DELETE |
| Client notification about payment transaction | specified by the client when transaction is created or during provisioning process | PaymentTransactionNotification | no | no | Notifies client about transaction status | no |

|  |  |  |
| --- | --- | --- |
| HTTP Method Operation | POST  Create a charge or a reservation | Occurs |
| resourceURL | [https://payment.sd-sfr.fr/payment/v2.1/{endUserId}/transactions/amount](https://payment.sd-sfr.fr/payment/v2.1/%7bendUserId%7d/transactions/amount)  [https://payment.sd-sfr.fr/payment/v2.1/{endUserId}/transactions/amountReservation](https://payment.sd-sfr.fr/payment/v2.1/%7bendUserId%7d/transactions/amountReservation) | 1-phase  2-phases |

|  |  |  |
| --- | --- | --- |
| HTTP Method  Operation | GET  Retrieve a transaction status | Occurs |
| resourceURL | [https://payment.sd-sfr.fr/payment/v2.1/{endUserId}/transactions/amount/{transactionId}](https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/%7btransactionId%7d)  https://payment.sd-sfr.fr/payment/v2.1/{endUserId}/transactions/amountReservation/{transactionId} | 1-phase  2-phases |

The {endUserId} is a variable representing the user to be CHARGED.

Use the last resourceURL return by server.

|  |  |  |
| --- | --- | --- |
| HTTP Method Operation | GET  Retrieve a subscription status | Occurs |
| resourceURL | https://payment.sd-sfr.fr/customerProfile/v1/{endUserId}/subscriptions/{mandateId} | 1-phase |
|  |  |  |
| HTTP Method Operation | GET  **Retrieve an endUser subscription list** | **Occurs** |
| resourceURL | https://payment.sd-sfr.fr/customerProfile/v1/{endUserId}/subscriptions | 1-phase |
|  |  |  |
| HTTP Method Operation | **GET**  **Retrieve the distributor subscription list** | **Occurs** |
| resourceURL | https://payment.sd-sfr.fr/customerProfile/v1/subscriptions | 1-phase |

|  |  |  |
| --- | --- | --- |
| HTTP Method Operation | POST  Renew a subscription | Occurs |
| resourceURL | https://renewal.sd-sfr.fr/payment/v2.1/{endUserId}/transactions/amount | 1-phase |

|  |  |  |
| --- | --- | --- |
| HTTP Method Operation | DELETE  Release a subscription | Occurs |
| resourceURL | https://payment.sd-sfr.fr/customerProfile/v1/{endUserId}/subscriptions/{mandateId} | 1-phase |

|  |  |  |
| --- | --- | --- |
| HTTP Method Operation | POST  Confirm or Release a transaction | Occurs |
| resourceURL | https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/{transactionId} | 2-phases |

## Structures

The subsections below define the data structures used in the RESTful Payment API.

Some of the structures can be instantiated as so-called root elements.

For structures that contains elements which describe a user identifier, the statements in section 3 regarding ‘tel’, ‘ip’ and ‘acr’ URI schemes apply.

### Type: PaymentTransactionList

List of all payment transactions

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Optional | Description |
| amountTransaction | AmountTransaction  [0..unbounded] | Yes | Collection of AmountTransaction |
| amountReservationTransaction | AmountReservationTransaction  [0..unbounded] | Yes | Collection of AmountReservationTransaction |

Table 5: Type PaymentTransactionList

A root element named paymentTransactionList of type PaymentTransactionList is allowed in response bodies.

### Type: PaymentTransactionNotification

Payment transaction notification

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Optional | Description |
| callbackData | string | Yes | The ‘callbackData’ element if it was passed by the application in the ‘callbackReference’ element when creating a payment transaction.  See [REST\_NetAPI\_Common] for details. |
| amountTransaction | AmountTransaction | Choice | AmountTransaction |
| amountReservationTransaction | AmountReservationTransaction | Choice | AmountReservationTransaction |
| link | Link  [0..unbounded] | Yes | Provided by the server and points to other resources that are in relationship with the current resource.  (Reserved) |

### Type: AmountTransaction

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | Optional |
| endUserId | anyURI | The identifier of the end user’s account to be charged (e.g. ‘tel’ URI, ‘acr’ URI).  If the endUserId is also part of the request URL, the two MUST have the same value. | No |
| paymentAmount | PaymentAmount | Information on the amount charge to be made. | No |
| transactionOperationStatus | TransactionOperationStatus | e.g. CHARGED, REFUNDED, etc. | No |
| referenceCode | string | Textual information to uniquely identify the request, e.g. in case of disputes. Used for business logic, not for operational logic | No |
| serverReferenceCode | string | A unique reference to the request, provided by the server, and meaningful to the server’s backend system for correlation purposes (e.g. to be used in case of a subsequent refund request related to this charge request) | Yes |
| originalServerReferenceCode | string | This can be used to reconcile a refund request with the original charge that is intended to be refunded. In case the server included a serverReferenceCode in the response to a charge request, then any subsequent client request to refund that charge SHOULD include that serverReferenceCode value in an originalServerReferenceCode field. If the client omits it from the refund request then the server MAY throw a policy exception. | Yes |
| clientCorrelator | string | A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server.  This field SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids duplicate transaction creation in such situations.  In case the field is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it. |  |
| callbackReference | CallbackReference | Client’s notification endpoint and parameters | Yes  (Reserved) |
| resourceURL | anyURL | Self-referring URL. The resourceURL SHALL NOT be included in POST requests by the client, but must be included in POST requests representing notifications by the server to the client, when a complete representation of the resource is embedded in the notification. The resourceURL MUST also be included in responses to any HTTP method that returns an entity body, and input requests. | Yes |
| link | Link  [0..unbounded] | Provided by the server and points to other resources that are in relationship with the current resource. | Yes  (Reserved) |

Table 6: Type AmountTransaction

A root element named amountTransaction of type AmountTransaction is allowed in request and/or response bodies.

Note that the clientCorrelator is used for purposes of error recovery as specified in [REST\_NetAPI\_Common], and internal client purposes. The server is NOT REQUIRED to use the clientCorrelator value in any form in the creation of the URL of the resource. The specification [REST\_NetAPI\_Common] provides a recommendation regarding the generation of the value of this field.

### Type: AmountReservationTransaction

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | Optional |
| endUserId | anyURI | The identifier of the end user’s account to be charged (e.g. ‘tel’ URI, ‘acr’ URI). It MUST be present in the initial reservation request.  If the endUserId is also part of the request URL, the two MUST have the same value. | Yes |
| paymentAmount | PaymentAmount | Information on the amount charge to be made. | No |
| subscription | Subscription | Information about the subscription to be made. | Yes |
| transactionOperationStatus | TransactionOperationStatus | e.g. RESERVED, CHARGED, RELEASED, etc. | No |
| referenceSequence | int | Sequential number generated by client application for every transaction state change.  The client will increment reference sequence with every new request to the server. If request failed the client can repeat the request with the same sequence number. This allows the server to distinguish easily between new and repeated requests (e.g. ignore repeated requests, in the case they completed on the server side).  Example of a sequence of usable referenceSequence values: use 1 to reserve amount X, then 2 to reserve additional amount Y, then 3 to charge reserved amount, etc. | No |
| referenceCode | String | Textual information to uniquely identify the request, e.g. in case of disputes. Used for business logic, not for operational logic | Yes |
| serverReferenceCode | String | A unique reference to the request, provided by the server, and meaningful to the server’s backend system for correlation purposes | Yes |
| originalServerReferenceCode | String | This may be used to reconcile a current transaction request with an original transaction | Yes |
| clientCorrelator | String | A correlator that the client can use to tag this particular resource representation during a request to create a resource on the server.  This field SHOULD be present. Note: this allows the client to recover from communication failures during resource creation and therefore avoids duplicate transaction creation in such situations.  In case the field is present, the server SHALL not alter its value, and SHALL provide it as part of the representation of this resource. In case the field is not present, the server SHALL NOT generate it. | Yes |
| callbackReference | CallbackReference | Client’s notification endpoint and parameters | Yes  (Reserved) |
| resourceURL | anyURL | Self-referring URL. The resourceURL SHALL NOT be included in POST requests by the client, but must be included in POST requests representing notifications by the server to the client, when a complete representation of the resource is embedded in the notification. The resourceURL MUST also be included in responses to any HTTP method that returns an entity body, and input requests. | Yes |
| link | Link  [0..unbounded] | Provided by the server and points to other resources that are in relationship with the current resource. | Yes  (Reserved) |

Table 7: Type AmountReservationTransaction

A root element named amountReservationTransaction of type AmountReservationTransaction is allowed in request and/or response bodies.

Regarding the clientCorrelator field, the note in section Type: AmountTransaction applies.

### Type: PaymentAmount

Payment amount specific information

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Optional | Description |
| chargingInformation | ChargingInformation | No | Holds the charge with amount, currency and description text. |
| totalAmountCharged | decimal | Yes | The total amount which has been charged |
| totalAmountRefunded | decimal | Yes | The total amount which has been refunded |
| amountReserved | decimal | Yes | The amount which has been reserved |
| chargingMetaData | ChargingMetaData | No | Metadata about the charging, such as e.g. information about the merchant, the product, taxation, etc. |

Table 8: Type PaymentAmount

A root element named paymentAmount is allowed in request and/or response bodies.

### Type: ChargingMetaData

Information about charging data

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Optional | Description |
| onBehalfOf | string | No | String parameter to allow aggregator or acquiring partners to specify who the payment is really by. This provides visibility to the true merchant in an aggregation scenario and is key in dealing with customer queries where an aggregator submits requests on behalf of another entity.  (Editor name)  ISO8859-1 30 caracters max |
| purchaseCategoryCode | string | No | A category defining the type of service, product or media being purchased. A standard list of category codes would be helpful together with the ability to extend as required. This provides multiple uses including correct taxation, service blocking and service based spending limits.  Possible values :   * Apps * Music * Gaming * Video * Book * Presse * Social * Mobile * Infos * Astro * Meteo * Trafic * Vote * Acces * Don * Store * Parking * Vending * Divers |
| channel | string | No | The channel over which the requester is interacting with the merchant, based on a pre-defined list of channels (e.g. WAP, WEB, SMS …) with the ability to extend the channel list as required.  This is useful if the operator needs to interact with the subscriber to authorize the charge (advice of charge) and provides details on how such interaction should be done. Some payment show the value in centrally controlled advice of charge and the channel is an important data element.  Possible values :   * WEB * WAP * SMS |
| taxAmount | decimal | Yes | The tax amount charged by the merchant if the charge has tax already included. This also provides an indicator to the downstream billing system.  It is important to know if the amount submitted has been pre-taxed so the subscribers do not get double taxed on their bill. |
| mandateId | string | Yes | The ID representing the subscription for which this charge applies. Allows operators to track charges and group them based on the subscription service which they belong to. Also allows operators to block subscription payments if they have been requested to by the subscriber. |
| serviceID | string | No | The ID of the partner/merchant service being purchased.  (partnerId) |
| productId | string | No | Combines with the serviceID to uniquely identify the product being purchased.  (CCS) |
|  |  |  |  |

Table 9: Type ChargingMetaData

### Type: ChargingInformation

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Optional | Description |
| amount | decimal | No | Amount to be charged/refunded/reserved. |
| currency | string | No | Currency identifier as defined in [ISO4217] |
| description | string | No | Description text to be used for information and billing text  (Service name)  ISO8859-1 30 caracters max  4 refund reasons are available :   * GCO (Commercial initiative) * PBT (Technical problem) * SNL (undelivered service) * AUT (other) |
| code | string | Yes | Charging code, referencing a contract under which the charge is applied.  (Not Used) |

Table 10: Type ChargingInformation

### Type: Subscription

Subscription specific information

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Optional | Description |
| subscriptionInformation | SubscriptionInformation | No | Holds the subscriptionwith amount, currency and description text. |
| subscriptionMetaData | SubscriptionMetaData | No | Metadata about the subscription, such as e.g. information about the merchant, the product, taxation, the recurring rule, etc. |
| mandateId | string | Yes | The ID representing the subscription for which this charge applies. Allows operators to track charges and group them based on the subscription service which they belong to. Also allows operators to block subscription payments if they have been requested to by the subscriber. |
| endUserId | anyURI | Yes | The identifier of the end user’s account to be charged (e.g. ‘tel’ URI, ‘acr’ URI). It MUST be present in the initial reservation request.  If the endUserId is also part of the request URL, the two MUST have the same value. |

Table 11: Type Subscription

### Type: SubscriptionInformation

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Optional | Description |
| amount | decimal | No | Normal amount when no discount is appliyed. |
| currency | string | No | Currency identifier as defined in [ISO4217] |
| description | string | No | Description text to be used for information and billing text  (Service name)  ISO8859-1 30 caracters max |
| code | string | Yes | Charging code, referencing a contract under which the charge is applied.  (Not Used) |

Table 12: Type SubscriptionInformation

### Type: SubscriptionMetaData

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Optional | Description |
| freq | string | No | “DAILY”, “MONTHLY”, “WEEKLY” |
| interval | int | No | Numbers of “freq“ intervals |
| dtstart | date | Yes | Subscription start date  (server info) |
| dtend | date | Yes | Subscription end date  (server info) |
| until | date | Yes | date until the subscription has to be renewed  (server info) |

Table 13: Type SubscriptionMetaData

## Enumerations

### Enumeration: TransactionOperationStatus

Enumeration of supported status resulting from transaction operations

|  |  |
| --- | --- |
| Enumeration | Description |
| CHARGED | In the request: charge the amount  In the response: the amount has been charged |
| REFUNDED | In the request: refund the amount  In the response: the amount has been refunded |
| RESERVED | In the request: reserve the amount  In the response: the amount has been reserved |
| RELEASED | In the request: release the reservation  In the response: the reservation is released  (explicitly by release operation, “not yet” implicitly by expiration) |
| DENIED | Not implemented  This value only occurs in responses to a GET operation and means that the operation has been denied by the back-end payment server because of an issue with the user’s account (e.g. insufficient balance, potential fraud flag on the user’s account). |
| REFUSED | Not used  This value only occurs in responses to a GET operation and means that the operation has been refused (i.e. rejected by end user). |
| PROCESSING | Future use  This value only occurs in responses to POST or GET operations and means that the operation is not done yet (e.g. the charge has not been yet added, the reservation is still processing). |

## Values of the Link “rel” attribute

The “rel” attribute of the Link element is a free string set by server implementation, to indicate a relationship between the current resource and an external resource. The following are strings (list is non-exhaustive, and can be extended):

* PaymentTransactionList
* AmountTransaction
* AmountReservationTransaction
* PaymentAmount

These values indicate the kind of resource that link points to.

## Sequence diagrams

### Amount charge and refund transaction

The figure below shows a scenario to create an amount charge or refund transaction for an end user.

The resource:

To create an amount charge or refund transaction, create new resource under

https://{serverRoot}/payment/{apiVersion}/{endUserId}/transactions/amount

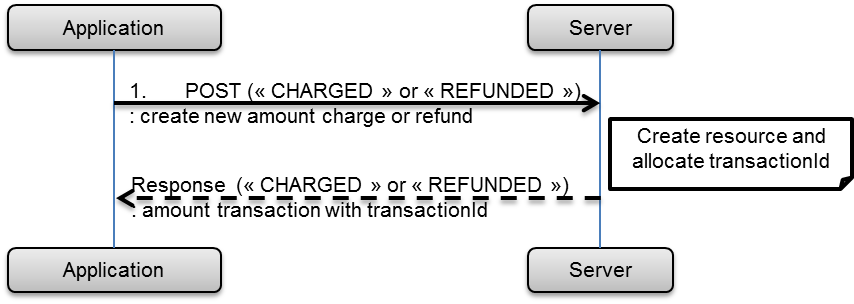


Figure 2 Amount charge and refund transaction

Outline of flow:

1. An application asks for the creation of new amount charge or refund transaction for an end user using POST and receives the response with resource URL containing transactionId.

### Amount reservation transaction

The figure below shows a scenario to create, to charge and to release an amount reservation transaction for an end user.

The resource:

To create an amount reservation for an end user, create new resource under

https://{serverRoot}/payment/{apiVersion}/{endUserId}/transactions/amountReservation

To update an amount reservation for an end user, use resource

https://{serverRoot}/payment/{apiVersion}/{endUserId}/transactions/amountReservation/{transactionId}

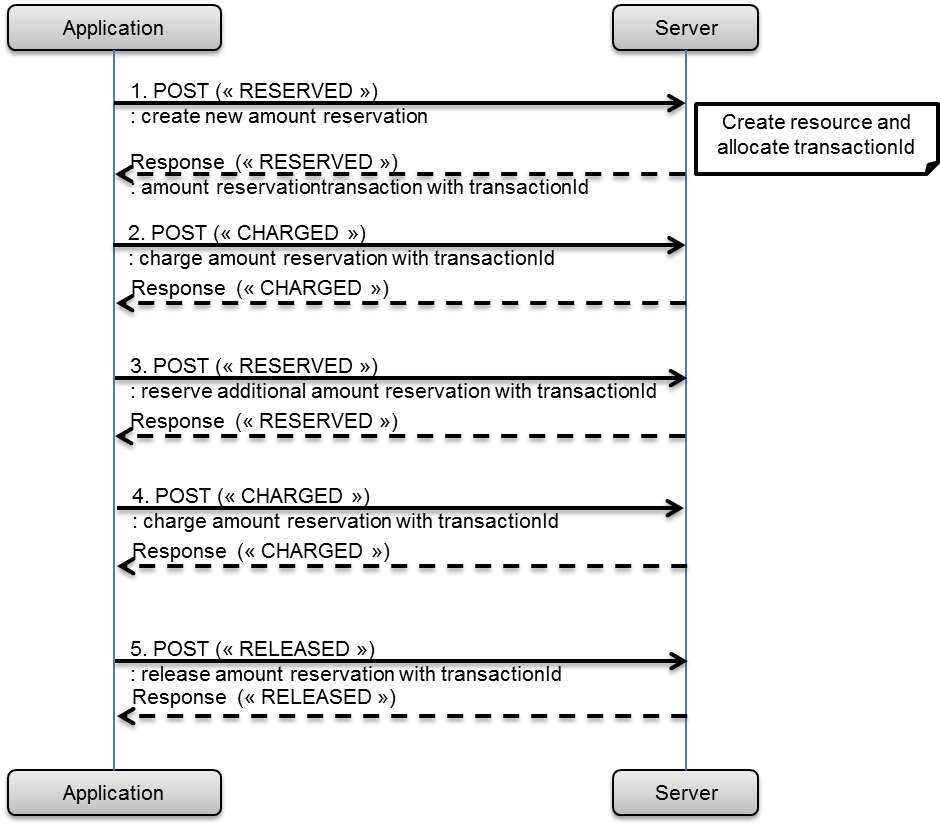


Figure 3 Amount reservation transaction

Outline of flow:

1. Application creates a new amount reservation transaction for an end user. (POST)
2. Application charges an amount reservation with transactionId. (POST)
3. In case that add to the existing amount reservation, application reserves an additional amount reservation with transactionId. (POST)
4. Application charges an amount reservation with transactionId. (POST)
5. Application releases an amount reservation transaction for an end user. (POST)

### Charge or refund amount and check transaction status

The figure below shows a scenario for charging an amount and get the transaction status of the transaction.

The resource:

* To create an amount charge or refund transaction, create new resource under
  + https://{serverRoot}/payment/{apiVersion}/{endUserId}/transactions/amount
* To get the status of an amount charge or refund transaction for an end user, use resource
  + https://{serverRoot}/payment/{apiVersion}/{endUserId}/transactions/amount/{transactionId}

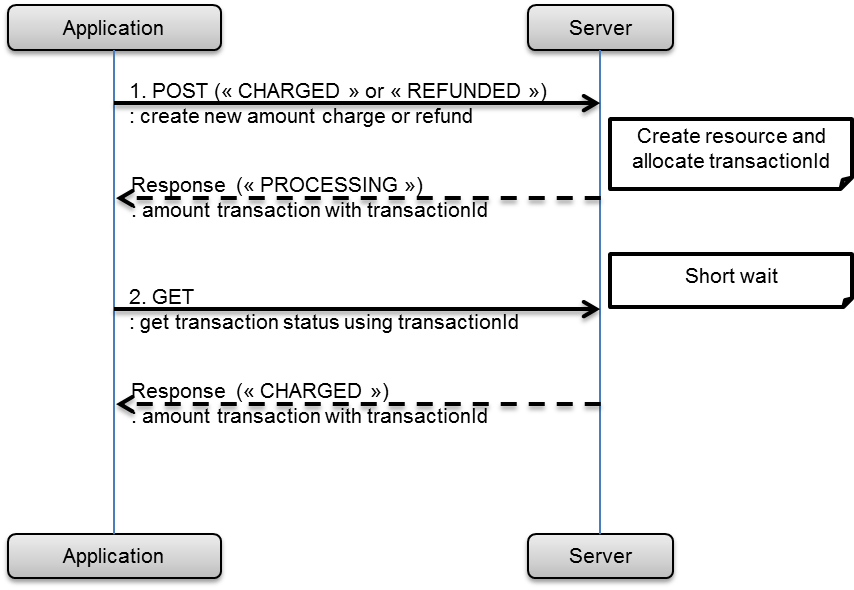


Figure 4 Amount transaction and status checking

Outline of flow:

1. An application asks for the creation of new amount charge or refund transaction for an end user using POST and receives the response containing the amount transaction with a “PROCESSING” status.
2. The application requests the transaction status with the given transactionId using GET and gets the final status.

### Charge or refund amount and notify transaction status

The figure below shows a scenario for charging an amount and notifying the transaction status of the transaction.

The optional notification URL passed by the client during the creation of the transaction step is a Client-side Notification URL.

The resources:

* To create an amount charge or refund transaction, create new resource under
  + https://{serverRoot}/payment/{apiVersion}/{endUserId}/transactions/amount
* To notify the application about the transaction status, POST notification to the client supplied notifyURL.

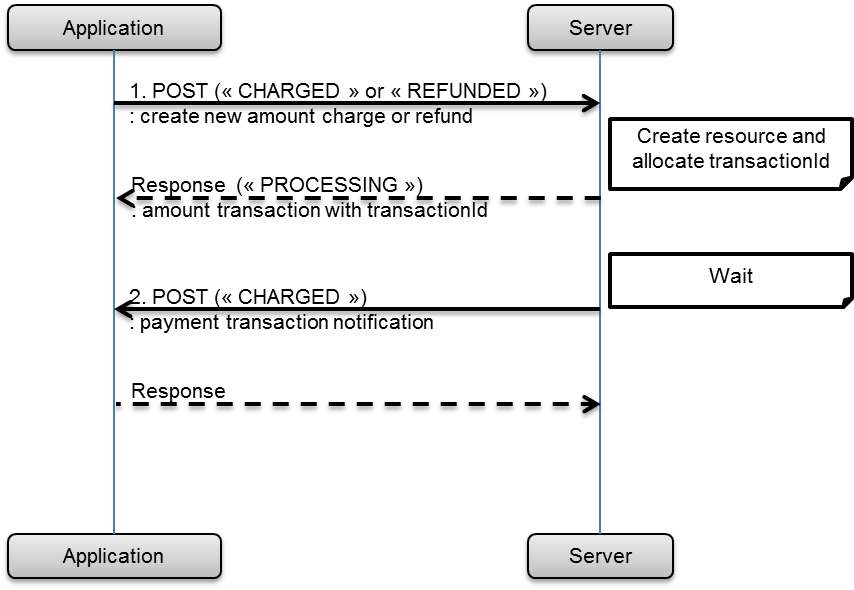


Figure 5 Amount transaction with notification

Outline of flow:

1. An application asks for the creation of new amount charge or refund transaction for an end user using POST and provides a callbackReference for notifications. The application receives the response containing the amount transaction with a “PROCESSING” status.
2. When the transaction reaches its final status, the server notifies the application the new transaction status.

# Detailed specification of the resources

The following applies to all resources defined in this specification for only one representation format (JSON):

* Reserved characters in URL variables (parts of a URL denoted below by a name in curly brackets) MUST be percent-encoded according to [RFC3986]. Note that this always applies, no matter whether the URL is used as a Request URL or inside the representation of a resource (such as in “resourceURL” and “link” elements).
* If a user identifier (e.g. address, userId, etc) of type anyURI is in the form of an MSISDN, it MUST be defined as a global number according to [RFC3966] (e.g. <tel:+33616700005>). The use of characters other than digits and the leading “+” sign SHOULD be avoided in order to ensure uniqueness of the resource URL. This applies regardless of whether the user identifier appears in a URL variable or in parameter in the body of an HTTP message.
* If a user identifier (e.g. address, userId, etc) of type anyURI is in the form of an IP, it MUST be prefixed by “ip:”, IPv6 must be neasted between brackets and suffixed by the port “:XXXX” and we notice no specific notation for IPv4.
* If a user identifier (e.g. address, userId, etc) of type anyURI is in the form of an Anonymous Customer Reference (ACR), it MUST be prefixed by “acr:”. ACR could be, a network @lias, a bill accountId or a land-line crmId.

## Resource: All payment transactions for an end user

The resource used is: https://{serverRoot}/payment/{apiVersion}/{endUserId}/transactions

This resource is used to provide access all completed and pending transactions (amount, amount reservation) for an end user.

### Request URL variables

The following request URL variables are common for all HTTP commands:

|  |  |
| --- | --- |
| Name | Description |
| serverRoot | Server base url: hostname + port + base path.  Port and base path are OPTIONAL.  Example: payment.sd-sfr.fr/payment |
| apiVersion | Version of the API client wants to use. The value is defined in section 3.1 |
| endUserId | The end user’s account identifier.  Examples:   * <tel:+33616700005> * ip:127.0.0.1 * ip:[2001:db8::1]:80 * acr:1-AKB12 * acr:212345645879624 * acr:375ee5ca-c6e2-11e6-bae5-b72336fcf494 |

### One phase payment (DIRECT DEBIT)

This operation is used to charge a currency amount to an end-user account.

Method:POST

URL: https://{serverRoot}/payment/v2.1/{endUserId}/transactions/amount

Where

* **{serverRoot}** is the domain name were Web Services are hosted : payment.sd-sfr.fr
* **{endUserId}** is the user ID who will be charged. Here this ID matches his ACR or MSISDN.

#### Create charge amount using ‘tel’ URI

##### Request

POST /payment/v2.1/tel:+33616700005/transactions/amount HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

{

"amountTransaction": {

"clientCorrelator": "55594",

"endUserId": "tel:+33616700005",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "SFR Pay",

"purchaseCategoryCode": "Gaming",

"channel": "WAP",

"taxAmount": 0,

"serviceID": "AF0010",

"productId": "3291"

}

},

"referenceCode" : "RefCode123",

"transactionOperationStatus": "CHARGED"

}

}

##### Response

HTTP/1.1 201 Created

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 16:06:55 GMT

Location: https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/2958e7ce-c605-11e6-b63f-0050568302c6

Server: nginx

Transfer-Encoding: chunked

Content-Type: application/json

Content-Length: 12345

Date: Thu, 04 Jun 2009 02:51:59 GMT

Location: <https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/2958e7ce-c605-11e6-b63f-0050568302c6>

{

"amountTransaction": {

"clientCorrelator": "55594",

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceCode": "RefCode123",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/2958e7ce-c605-11e6-b63f-0050568302c6",

"serverReferenceCode": "testP\_2958e7ce-c605-11e6-b63f-0050568302c6",

"transactionOperationStatus": "CHARGED",

"paymentAmount": {

"totalAmountCharged": 0.1,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": ""

}

}

}

}

#### unexpected login or password

{

"requestError": {

"policyException": {

"messageId": "POL-0008",

"text": "unexpected login or password"

}

}

}

#### Create charge amount with callbackReference and “PROCESSING” status returned (not implemented yet)

##### Request

POST /payment/v2.1/tel:+33616700005/transactions/amount HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

{

"amountTransaction": {

"callbackReference": {

"callbackData": "12345",

"notifyURL": "http://application.example.com/notifications/FinalResult"

},

"clientCorrelator": "55594",

"endUserId": "tel:+33616700005",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "SFR Pay",

"purchaseCategoryCode": "Gaming",

"channel": "WAP",

"taxAmount": 0,

"serviceID": "AF0010",

"productId": "3291"

}

},

"referenceCode" : "RefCode123",

"transactionOperationStatus": "CHARGED"

}

}

##### Response

HTTP/1.1 202 Accepted

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 16:18:17 GMT

Location: https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/bfadd86e-c606-11e6-aeed-0050568302c6

Server: nginx

Transfer-Encoding: chunked

{

"amountTransaction": {

"clientCorrelator": "55594",

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceCode": "RefCode123",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/bfadd86e-c606-11e6-aeed-0050568302c6",

"serverReferenceCode": "testP\_bfadd86e-c606-11e6-aeed-0050568302c6",

"transactionOperationStatus": "PROCESSING",

"paymentAmount": {

"totalAmountCharged": 0.1,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": ""

}

}

}

}

#### Client retries POST with same clientCorrelator (not implemented yet)

Retries are idempotent.

##### Request

POST /payment/v2.1/tel:+33616700005/transactions/amount HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

{

"amountTransaction": {

"clientCorrelator": "55594",

"endUserId": "tel:+33616700005",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "SFR Pay",

"purchaseCategoryCode": "Gaming",

"channel": "WAP",

"taxAmount": 0,

"serviceID": "AF0010",

"productId": "3291"

}

},

"referenceCode" : "RefCode123",

"transactionOperationStatus": "CHARGED"

}

}

##### Response

HTTP/1.1 200 OK

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 16:06:55 GMT

Location: https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/2958e7ce-c605-11e6-b63f-0050568302c6

Server: nginx

Transfer-Encoding: chunked

{

"amountTransaction": {

"clientCorrelator": "55594",

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceCode": "RefCode123",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/2958e7ce-c605-11e6-b63f-0050568302c6",

"serverReferenceCode": "testP\_2958e7ce-c605-11e6-b63f-0050568302c6",

"transactionOperationStatus": "CHARGED",

"paymentAmount": {

"totalAmountCharged": 0.1,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": ""

}

}

}

}

#### Unsuccessful charge request because of denial/refusal by back-end system

##### Request

POST /payment/v2.1/tel:+33616700005/transactions/amount HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

{

"amountTransaction": {

"callbackReference": {

"callbackData": "12345",

"notifyURL": "http://application.example.com/notifications/FinalResult"

},

"clientCorrelator": "55594",

"endUserId": "tel:+33616700005",

"paymentAmount": {

"chargingInformation": {

"amount": 99999,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "SFR Pay",

"purchaseCategoryCode": "Gaming",

"channel": "WAP",

"taxAmount": 0,

"serviceID": "AF0010",

"productId": "3291"

}

},

"referenceCode" : "RefCode123",

"transactionOperationStatus": "CHARGED"

}

}

##### Response

HTTP/1.1 403 Forbidden

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 16:34:06 GMT

Server: nginx

Transfer-Encoding: chunked

{

"requestError": {

"policyException": {

"messageId": "POL-1001",

"text": "SimpaNG Transaction denied due to excessive monthly payment amount for secondary threshold DDT"

}

}

}

### Two phases payment (AUTORIZATION - CONFIRMATION)

Create reserve amount

The format of the URL is as follows :

https://{serverRoot}/payment/v2.1/{endUserId}/transactions/amountReservation

where

* **{serverRoot}** is the domain name were Web Services are hosted : payment.sd-sfr.fr
* **{endUserId}** is the user ID who will be charged. Here this ID matches his ACR or MSISDN.

The method of data transmission is the POST method.

#### Create reserve amount (Reservation)

##### Request

POST /payment/v2.1/tel:+33616700005/transactions/amountReservation HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

{

"amountReservationTransaction": {

"clientCorrelator": "55594",

"endUserId": "tel:+33616700005",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "SFR Pay",

"purchaseCategoryCode": "Gaming",

"channel": "WAP",

"taxAmount": 0,

"serviceID": "AF0010",

"productId": "3291"

}

},

"referenceCode" : "RefCode123",

"referenceSequence": "1",

"transactionOperationStatus": "RESERVED"

}

}

##### Response

HTTP/1.1 201 Created

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 16:45:37 GMT

Location: https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/91887a9e-c60a-11e6-8e14-0050568302c6

Server: nginx

Transfer-Encoding: chunked

{

"amountReservationTransaction": {

"clientCorrelator": "55594",

"endUserId": "tel:+33616700005",

"referenceCode": "RefCode123",

"referenceSequence": "1",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/91887a9e-c60a-11e6-8e14-0050568302c6",

"transactionOperationStatus": "RESERVED",

"paymentAmount": {

"amountReserved": 0.1,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": ""

}

}

}

}

#### Charge amount for amount reservation (Capture)

##### Request

POST /payment/v2.1/transactions/amountReservation/91887a9e-c60a-11e6-8e14-0050568302c6 HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

{

"amountReservationTransaction": {

"clientCorrelator": "55594",

"endUserId": "tel:+33616700005",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "SFR Pay",

"purchaseCategoryCode": "Gaming",

"channel": "WAP",

"taxAmount": 0,

"serviceID": "AF0010",

"productId": "3291"

}

},

"referenceCode" : "RefCode123",

"referenceSequence": "2",

"transactionOperationStatus": "CHARGED"

}

}

##### Response

HTTP/1.1 200 OK

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 16:49:13 GMT

Server: nginx

Transfer-Encoding: chunked

{

"amountReservationTransaction": {

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceCode": "RefCode123",

"referenceSequence": "2",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/91887a9e-c60a-11e6-8e14-0050568302c6",

"serverReferenceCode": "testP\_91887a9e-c60a-11e6-8e14-0050568302c6",

"transactionOperationStatus": "CHARGED",

"paymentAmount": {

"totalAmountCharged": 0.1,

"amountReserved": 0,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": ""

}

}

}

}

#### Repeat charge request with same referenceSequence for an amount reservation (idempotency) (not implemented yet)

##### Request

POST /payment/v2.1/transactions/amountReservation/91887a9e-c60a-11e6-8e14-0050568302c6 HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

{

"amountReservationTransaction": {

"clientCorrelator": "55594",

"endUserId": "tel:+33616700005",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "SFR Pay",

"purchaseCategoryCode": "Gaming",

"channel": "WAP",

"taxAmount": 0,

"serviceID": "AF0010",

"productId": "3291"

}

},

"referenceCode" : "RefCode123",

"referenceSequence": "2",

"transactionOperationStatus": "CHARGED"

}

}

##### Response

HTTP/1.1 200 OK

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 16:49:13 GMT

Server: nginx

Transfer-Encoding: chunked

{

"amountReservationTransaction": {

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceCode": "RefCode123",

"referenceSequence": "2",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/91887a9e-c60a-11e6-8e14-0050568302c6",

"serverReferenceCode": "testP\_91887a9e-c60a-11e6-8e14-0050568302c6",

"transactionOperationStatus": "CHARGED",

"paymentAmount": {

"totalAmountCharged": 0.1,

"amountReserved": 0,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": ""

}

}

}

}

#### Release amount reservation (Annulation)

##### Request

POST /payment/v2.1/transactions/amountReservation/5900649a-c60d-11e6-9104-0050568302c6 HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

{

"amountReservationTransaction": {

"clientCorrelator": "55594",

"endUserId": "tel:+33616700005",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "SFR Pay",

"purchaseCategoryCode": "Gaming",

"channel": "WAP",

"taxAmount": 0,

"serviceID": "AF0010",

"productId": "3291"

}

},

"referenceCode" : "RefCode123",

"referenceSequence": "2",

"transactionOperationStatus": "RELEASED"

}

}

##### Response

HTTP/1.1 200 OK

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 16:49:13 GMT

Server: nginx

Transfer-Encoding: chunked

{

"amountReservationTransaction": {

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceSequence": "2",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/5900649a-c60d-11e6-9104-0050568302c6",

"serverReferenceCode": "testP\_5900649a-c60d-11e6-9104-0050568302c6",

"transactionOperationStatus": "RELEASED",

"paymentAmount": {

"amountReserved": 0,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": ""

}

}

}

}

#### Release an already charged reservation

##### Request

POST /payment/v2.1/transactions/amountReservation/91887a9e-c60a-11e6-8e14-0050568302c6 HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

{

"amountReservationTransaction": {

"clientCorrelator": "55594",

"endUserId": "tel:+33616700005",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "SFR Pay",

"purchaseCategoryCode": "Gaming",

"channel": "WAP",

"taxAmount": 0,

"serviceID": "AF0010",

"productId": "3291"

}

},

"referenceCode" : "RefCode123",

"referenceSequence": "2",

"transactionOperationStatus": "RELEASED"

}

}

##### Response

HTTP/1.1 400 Bad Request

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 16:49:13 GMT

Server: nginx

Transfer-Encoding: chunked

{

"requestError": {

"serviceException": {

"text": "transaction already managed",

"messageId": "SVC0007"

}

}

}

#### Charge partial amount for amount reservation (not implemented yet)

##### Request

POST [https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/ctc123 HTTP/1.1](https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/ctc123%20HTTP/1.1)

Accept: application/json

Host: payment.sd-sfr.fr:80

Content-Type: application/json

Content-Length: 12345

Date: Thu, 04 Jun 2009 02:51:59 GMT

{

"amountReservationTransaction": {

"endUserId": "tel:+33603100000",

"paymentAmount": {

"chargingInformation": {

"amount": 0.5,

"currency": "EUR",

"description": "Service yyy"

}

},

"referenceCode": "REF-12346",

"referenceSequence": "2",

"transactionOperationStatus": "CHARGED"

}

}

##### Response

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: 12345

Date: Thu, 04 Jun 2009 02:51:59 GMT

{

"amountReservationTransaction": {

"endUserId": "tel:+33603100000",

"paymentAmount": {

"amountReserved": 0.25,

"chargingInformation": {

"amount": 0.5,

"currency": "EUR",

"description": "Service yyy"

},

"totalAmountCharged": 0.5

},

"referenceCode": "XXXXX",

"referenceSequence": "2",

"resourceURL": "http://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/abc123-fgt-458", "transactionOperationStatus": "CHARGED"

}

}

#### Release remaining amount reservation

##### Request

POST /payment/v2.1/transactions/amountReservation/1f75a9e6-c60e-11e6-9aee-0050568302c6 HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

{

"amountReservationTransaction": {

"endUserId": "tel:+33616700005",

"referenceSequence": "3",

"transactionOperationStatus": "RELEASED"

}

}

##### Response

HTTP/1.1 200 OK

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 16:49:13 GMT

Server: nginx

Transfer-Encoding: chunked

{

"amountReservationTransaction": {

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceSequence": "3",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/1f75a9e6-c60e-11e6-9aee-0050568302c6",

"serverReferenceCode": "testP\_1f75a9e6-c60e-11e6-9aee-0050568302c6",

"transactionOperationStatus": "RELEASED",

"paymentAmount": {

"amountReserved": 0,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": ""

}

}

}

}

#### Reserve additional amount for amount reservation

##### Request

POST /payment/v2.1/transactions/amountReservation/d771cde0-c60e-11e6-b928-0050568302c6 HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

{

"amountReservationTransaction": {

"clientCorrelator": "55594",

"endUserId": "tel:+33616700005",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "SFR Pay",

"purchaseCategoryCode": "Gaming",

"channel": "WAP",

"taxAmount": 0,

"serviceID": "AF0010",

"productId": "3291"

}

},

"referenceCode" : "RefCode123",

"referenceSequence": "2",

"transactionOperationStatus": "RESERVED"

}

}

##### Response

HTTP/1.1 200 OK

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 16:49:13 GMT

Server: nginx

Transfer-Encoding: chunked

{

"amountReservationTransaction": {

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceCode": "RefCode123",

"referenceSequence": "2",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/d771cde0-c60e-11e6-b928-0050568302c6",

"serverReferenceCode": "testP\_d771cde0-c60e-11e6-b928-0050568302c6",

"transactionOperationStatus": "RESERVED",

"paymentAmount": {

"amountReserved": 0.2,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": ""

}

}

}

}

#### Unsuccessful charge amount for amount reservation because of denial/refusal by back-end system (“link” not implemented yet)

##### Request

POST [https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/ctc123 HTTP/1.1](https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/ctc123%20HTTP/1.1)

Accept: application/json

Host: payment.sd-sfr.fr:80

Content-Type: application/json

Content-Length: 12345

Date: Thu, 04 Jun 2009 02:51:59 GMT

{

"amountReservationTransaction": {

"endUserId": "tel:+33603100000",

"paymentAmount": {

"chargingInformation": {

"amount": 0.5,

"currency": "EUR",

"description": "Service yyy"

}

},

"referenceCode": "REF-12346",

"referenceSequence": "2",

"transactionOperationStatus": "CHARGED"

}

}

##### Response

HTTP/1.1 400 Bad Request

Date: Thu, 04 Jun 2009 02:51:59 GMT

Content-Type: application/json

Content-Length: 12345

{

"requestError": {

"link":{

"href": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/ctc123",

"rel": "AmountReservationTransaction"

},

"serviceException": {

"messageId": "SVC0270",

"text": "Charging operation failed, the charge was not applied."

}

}

}

### Client notification about payment transaction amount (not implemented yet)

##### Request

POST [/notifications/PaymentTransactionFinalResult HTTP/1.1](https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/ctc123%20HTTP/1.1)

Accept: application/json

Host: application.example.com

Content-Type: application/json

Content-Length: 12345

Date: Thu, 04 Jun 2009 02:51:59 GMT

{

"paymentTransactionNotication": {

"amountReservationTransaction": {

"endUserId": "tel:+33603100000",

"paymentAmount": {

"chargingInformation": {

"amount": 0.5,

"currency": "EUR",

"description": "Service yyy"

}

},

"referenceCode": "REF-12346",

"transactionOperationStatus": "CHARGED"

}

}

}

##### Response

HTTP/1.1 204 No Content

Date: Thu, 04 Jun 2009 02:51:59 GMT

### RETRIEVE TRANSACTIONS

Retrieve can be by endUserId and / or by date, when no endUserId is specified we warmly recommend setting a date range.

#### Get all transactions

##### Request

GET /payment/v2.1/acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a/transactions?startDate=2016-12-19&endDate=2016-12-19 HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

##### Response

HTTP/1.1 200 OK

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 17:30:22 GMT

Server: nginx

Transfer-Encoding: chunked

{

"paymentTransactionList": {

"amountTransaction": [

{

"clientCorrelator": "55594",

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceCode": "RefCode123",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/bfadd86e-c606-11e6-aeed-0050568302c6",

"serverReferenceCode": "testP\_bfadd86e-c606-11e6-aeed-0050568302c6",

"transactionOperationStatus": "CHARGED",

"paymentAmount": {

"totalAmountCharged": 0.1,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": "",

"TransactionTimestamp": "2016-12-19T16:18:17Z"

}

}

},

{

"clientCorrelator": "55594",

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceCode": "RefCode123",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/2958e7ce-c605-11e6-b63f-0050568302c6",

"serverReferenceCode": "testP\_2958e7ce-c605-11e6-b63f-0050568302c6",

"transactionOperationStatus": "CHARGED",

"paymentAmount": {

"totalAmountCharged": 0.1,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": "",

"TransactionTimestamp": "2016-12-19T16:06:55Z"

}

}

}

],

"amountReservationTransaction": [

{

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceCode": "RefCode123",

"referenceSequence": "2",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/d771cde0-c60e-11e6-b928-0050568302c6",

"serverReferenceCode": "testP\_d771cde0-c60e-11e6-b928-0050568302c6",

"transactionOperationStatus": "RESERVED",

"paymentAmount": {

"amountReserved": 0.2,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": "",

"TransactionTimestamp": "2016-12-19T17:17:06Z"

}

}

},

{

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"serverReferenceCode": "3019281082",

"transactionOperationStatus": "RELEASED",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "ARRETE FACTURE 10 FCT",

"productId": "3291",

"serviceID": "AF0010",

"TransactionTimestamp": "2016-12-19T18:17:05.000+01:00",

"billingLabel": "TIERS:internet+:ACTE:3291"

}

}

},

{

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"serverReferenceCode": "3019281082",

"transactionOperationStatus": "RESERVED",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "ARRETE FACTURE 10 FCT",

"productId": "3291",

"serviceID": "AF0010",

"TransactionTimestamp": "2016-12-19T18:17:05.000+01:00",

"billingLabel": "TIERS:internet+:ACTE:3291"

}

}

},

{

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceSequence": "2",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/1f75a9e6-c60e-11e6-9aee-0050568302c6",

"serverReferenceCode": "testP\_1f75a9e6-c60e-11e6-9aee-0050568302c6",

"transactionOperationStatus": "RELEASED",

"paymentAmount": {

"amountReserved": 0,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": "",

"TransactionTimestamp": "2016-12-19T17:12:22Z"

}

}

},

{

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"serverReferenceCode": "2218087011",

"transactionOperationStatus": "RESERVED",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "ARRETE FACTURE 10 FCT",

"productId": "3291",

"serviceID": "AF0010",

"TransactionTimestamp": "2016-12-19T18:12:22.000+01:00",

"billingLabel": "TIERS:internet+:ACTE:3291"

}

}

},

{

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceSequence": "2",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/5900649a-c60d-11e6-9104-0050568302c6",

"serverReferenceCode": "testP\_5900649a-c60d-11e6-9104-0050568302c6",

"transactionOperationStatus": "RELEASED",

"paymentAmount": {

"amountReserved": 0,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": "",

"TransactionTimestamp": "2016-12-19T17:06:20Z"

}

}

},

{

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"serverReferenceCode": "3000588025",

"transactionOperationStatus": "RESERVED",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "ARRETE FACTURE 10 FCT",

"productId": "3291",

"serviceID": "AF0010",

"TransactionTimestamp": "2016-12-19T18:06:20.000+01:00",

"billingLabel": "TIERS:internet+:ACTE:3291"

}

}

},

{

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceCode": "RefCode123",

"referenceSequence": "2",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/91887a9e-c60a-11e6-8e14-0050568302c6",

"serverReferenceCode": "testP\_91887a9e-c60a-11e6-8e14-0050568302c6",

"transactionOperationStatus": "CHARGED",

"paymentAmount": {

"totalAmountCharged": 0.1,

"amountReserved": 0,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": "",

"TransactionTimestamp": "2016-12-19T16:49:13Z"

}

}

},

{

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"serverReferenceCode": "2157041144",

"transactionOperationStatus": "RESERVED",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"onBehalfOf": "ARRETE FACTURE 10 FCT",

"productId": "3291",

"serviceID": "AF0010",

"TransactionTimestamp": "2016-12-19T17:49:13.000+01:00",

"billingLabel": "TIERS:internet+:ACTE:3291"

}

}

}

],

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a/transactions?startDate=2016-12-19&endDate=2016-12-19"

}

}

#### Get all transactions “Bad Request”

##### Request

GET /payment/v2.1/acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a/transactions?startDate=2014-1120&endDate=2015-01-05 HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

##### Response

HTTP/1.1 400 Bad Request

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 17:21:55 GMT

Server: nginx

Transfer-Encoding: chunked

{

"requestError": {

"serviceException": {

"text": "unauthorized format for value 2014-1120 for startDate",

"messageId": "SVC3000"

}

}

}

#### Get amount reservation (not implemented yet)

##### Request

GET /payment/v2.1/tel:+33616700005/transactions/amountReservation/2ae2a3de-c612-11e6-b40d-0050568302c6 HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

##### Response

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: 12345

Date: Thu, 04 Jun 2009 02:51:59 GMT

{

"amountReservationTransaction": {

"clientCorrelator": "55594",

"endUserId": "tel:+33616700005",

"referenceCode": "RefCode123",

"referenceSequence": "1",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation/2ae2a3de-c612-11e6-b40d-0050568302c6",

"transactionOperationStatus": "RESERVED",

"paymentAmount": {

"amountReserved": 0.1,

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "test Achat"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"taxAmount": 0,

"billingLabel": ""

}

}

}

}

#### Get all amount reservation transactions (not implemented yet)

##### Request

GET /payment/v2.1/tel:+33616700005/transactions/amountReservation HTTP/1.1

Accept: application/json

Host: payment.sd-sfr.fr

##### Response

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: 12345

Date: Thu, 04 Jun 2009 02:51:59 GMT

{

"paymentTransactionList": {

"amountReservationTransaction": [

{

"endUserId": "tel:+33603100000",

"paymentAmount": {

"amountReserved": 0,

"chargingInformation": {

"amount": 2.99,

"currency": "EUR",

"description": "test SFR Pay"

},

"chargingMetaData": {

"TransactionTimestamp":"2015-06-01T17:23:20.000+01:00"

},

"totalAmountCharged": 2.99

},

"referenceCode": "refCode123",

"referenceSequence": "2",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/ctc123",

"serverReferenceCode": "srefCode123",

"transactionOperationStatus": "CHARGED"

},

{

"endUserId": "tel:+ 33603100000",

"paymentAmount": {

"amountReserved": 1.50,

"chargingInformation": {

"amount": 1.50,

"currency": "EUR",

"description": "test SFR Pay"

},

"chargingMetaData": {

"TransactionTimestamp":"2015-04-05T18:03:21.000+01:00"

},

"totalAmountCharged": 0

},

"referenceSequence": "1",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/ctc124",

"transactionOperationStatus": "RESERVED"

}

],

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amountReservation"

}

}

#### Get amount charge (not implemented yet)

##### Request

GET /payment/v2.1/tel:+33616700005/transactions/amount/ctc123 HTTP/1.1

Accept: application/json

Host: payment.sd-sfr.fr

##### Response

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: 12345

Date: Thu, 04 Jun 2009 02:51:59 GMT

{

"amountTransaction": {

"clientCorrelator": "54321",

"endUserId": "tel:+33603100000",

"paymentAmount": {

"chargingInformation": {

"amount": 0.75,

"currency": "EUR",

"description": "Service yyy"

},

"chargingMetaData": {

"onBehalfOf": "Editeur xxx",

"purchaseCategoryCode": "Contenu",

"channel": "SMS",

"taxAmount": 0,

"serviceID": "X01234",

"productId": "1234"

},

"totalAmountCharged": 0.75

},

"referenceCode": "Chips-slc123",

"referenceSequence": "1",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/ctc123",

"serverReferenceCode": "partner\_ctc123",

"transactionOperationStatus": "CHARGED"

}

}

### Create refund amount

The format of the URL is as follow :

https://{serverRoot}/payment/v2.1/transactions/amount

Where

* **{serverRoot}** is the domain name where Web services are hosted: **payment.sd-sfr.fr**

The selected data transmission method is the POST method.

#### Example of refund request

##### Request

POST /payment/v2.1/transactions/amount HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

{

"amountTransaction":{

"clientCorrelator": "55594",

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceCode": "2958e7ce-c605-11e6-b63f-0050568302c6",

"originalServerReferenceCode": "testP\_2958e7ce-c605-11e6-b63f-0050568302c6",

"transactionOperationStatus": "REFUNDED",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "PBT"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010"

}

}

}

}

##### Response

HTTP/1.1 200 OK

Connection: keep-alive

Content-Type: application/json

Date: Mon, 19 Dec 2016 18:30:36 GMT

Server: nginx

Transfer-Encoding: chunked

{

"amountTransaction": {

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"originalServerReferenceCode": "testP\_2958e7ce-c605-11e6-b63f-0050568302c6",

"referenceCode": "2958e7ce-c605-11e6-b63f-0050568302c6",

"resourceURL": "https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/3c17e1b2-c619-11e6-882f-0050568302c6",

"transactionOperationStatus": "REFUNDED",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "PBT"

},

"chargingMetaData": {

"channel": "WAP",

"onBehalfOf": "SFR Pay",

"productId": "3291",

"purchaseCategoryCode": "Gaming",

"serviceID": "AF0010",

"billingLabel": ""

},

"totalAmountRefunded": 0.1

}

}

}

#### Missing amountTransaction root element

##### Request

POST /payment/v2.1/tel:+33616700005/transactions/amount HTTP/1.1

Host: payment.sd-sfr.fr

Accept: application/json

Content-Type: application/json

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

Postman-Token: a72ba2cd-3013-06f4-c212-51a5684a2242

{

"clientCorrelator": "55594",

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a",

"referenceCode": "RefCode123",

"originalServerReferenceCode": "testP\_bfadd86e-c606-11e6-aeed-0050568302c6",

"transactionOperationStatus": "REFUNDED",

"paymentAmount": {

"chargingInformation": {

"amount": 0.1,

"currency": "EUR",

"description": "AUT"

}

}

}

##### Response

HTTP/1.1 400 Bad Request

Date: Thu, 04 Jun 2015 02:51:59 GMT

Location: https://payment.sd-sfr.fr/payment/v2.1/transactions/amount/ref123

Content-Type: application/json

Content-Length: 12345

{

"requestError": {

"serviceException": {

"text": "Parameter amountTransaction/endUserId is required",

"messageId": "SVC3000"

}

}

}

### RETRIEVE SUBSCRIPTIONS

Retrieve can be by endUserId and / or by date, when no endUserId is specified we warmly recommend setting a date range.

By partner (no endUserId):

https://{serverRoot}/customerProfile/v1/subscriptions?startDate=2016-12-01&endDate=2016-12-09

By endUserId:

https://{serverRoot}/customerProfile/v1/{endUserId}/subscriptions

Where

* **{serverRoot}** is the domain name where Web services are hosted: **payment.sd-sfr.fr**

The selected data transmission method is GET.

#### Example retrieving subscriptions by partner (login)

##### Request

GET /customerProfile/v1/subscriptions?startDate=2016-12-13&endDate=2016-12-20 HTTP/1.1

Host: payment.sd-sfr.fr

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

##### Response

HTTP/1.1 200 OK

Connection: keep-alive

Content-Type: application/json

Date: Tue, 20 Dec 2016 09:16:59 GMT

Server: nginx

Transfer-Encoding: chunked

{

"subscriptions": [

{

"subscriptionInformation": {

"amount": 6,

"currency": "EUR",

"description": "Test SFR DEV"

},

"subscriptionMetaData": {

"rrule": {

"freq": "DAILY",

"interval": 30,

"dtstart": "2016-12-14T11:11:01.000+01:00",

"until": "2017-01-13T11:11:01.000+01:00"

},

"productId": "3292",

"serviceID": "AF0010",

"onBehalfOf": "SFR DEV"

},

"mandateId": "9d709522-c1e5-11e6-ab5f-005056831b41",

"endUserId": "acr:9e565fc7-e9cd0bc3-686a1f74-8254dfcb-1ed8e185"

},

{

"subscriptionInformation": {

"amount": 35,

"currency": "EUR",

"description": "test Abo"

},

"subscriptionMetaData": {

"rrule": {

"freq": "MONTHLY",

"interval": 1,

"dtstart": "2016-12-14T17:20:49.000+01:00",

"until": "2017-01-13T17:20:49.000+01:00"

},

"productId": "3292",

"serviceID": "AF0010",

"onBehalfOf": "SFR Pay"

},

"mandateId": "468c0320-c219-11e6-b8f9-0050568302c6",

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a"

},

{

"subscriptionInformation": {

"amount": 35,

"currency": "EUR",

"description": "test Abo"

},

"subscriptionMetaData": {

"rrule": {

"freq": "MONTHLY",

"interval": 1,

"dtstart": "2016-12-15T10:31:04.000+01:00",

"until": "2017-01-14T10:31:04.000+01:00"

},

"productId": "3292",

"serviceID": "AF0005",

"onBehalfOf": "SFR Pay"

},

"mandateId": "32ca007c-c2a9-11e6-a521-005056831b41",

"endUserId": "acr:1a33fb6b-6a299ff7-e1b8f827-152ed209-0e89b8a4"

},

{

"subscriptionInformation": {

"amount": 10,

"currency": "EUR",

"description": "Test Vanessa"

},

"subscriptionMetaData": {

"rrule": {

"freq": "WEEKLY",

"interval": 1,

"dtstart": "2016-12-15T15:03:16.000+01:00",

"until": "2016-12-22T15:03:16.000+01:00"

},

"productId": "3292",

"serviceID": "AF0005",

"onBehalfOf": "SFR Vanessa"

},

"mandateId": "3a0a43bc-c2cf-11e6-8df8-0050568302c6",

"endUserId": "acr:1a33fb6b-6a299ff7-e1b8f827-152ed209-0e89b8a4"

}

]

}

#### Example retrieving subscriptions by endUserId

##### Request

GET /customerProfile/v1/tel:+33616700005/subscriptions HTTP/1.1

Host: payment.sd-sfr.fr

Authorization: Basic dFA6dGVxdGVxdFA=

Cache-Control: no-cache

##### Response

HTTP/1.1 200 OK

Connection: keep-alive

Content-Type: application/json

Date: Tue, 20 Dec 2016 09:16:59 GMT

Server: nginx

Transfer-Encoding: chunked

{

"subscriptions": [

{

"subscriptionInformation": {

"amount": 35,

"currency": "EUR",

"description": "test Abo"

},

"subscriptionMetaData": {

"rrule": {

"freq": "WEEKLY",

"interval": 1,

"dtstart": "2016-12-08T15:36:58.000+01:00",

"until": "2016-12-15T15:36:58.000+01:00"

},

"productId": "3292",

"serviceID": "AF0010",

"onBehalfOf": "SFR Pay"

},

"mandateId": "c6347172-bd53-11e6-8002-005056831b41",

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a"

},

{

"subscriptionInformation": {

"amount": 35,

"currency": "EUR",

"description": "test Abo"

},

"subscriptionMetaData": {

"rrule": {

"freq": "MONTHLY",

"interval": 1,

"dtstart": "2016-12-14T17:20:49.000+01:00",

"until": "2017-01-13T17:20:49.000+01:00"

},

"productId": "3292",

"serviceID": "AF0010",

"onBehalfOf": "SFR Pay"

},

"mandateId": "468c0320-c219-11e6-b8f9-0050568302c6",

"endUserId": "acr:0d249698-520c031e-dc7196a3-777f0cff-ac15f31a"

},

{

"subscriptionInformation": {

"amount": 35,

"currency": "EUR",

"description": "test Abo"

},

"subscriptionMetaData": {

"rrule": {

"freq": "MONTHLY",

"interval": 1,

"dtstart": "2016-12-15T10:31:04.000+01:00",

"until": "2017-01-14T10:31:04.000+01:00"

},

"productId": "3292",

"serviceID": "AF0005",

"onBehalfOf": "SFR Pay"

},

"mandateId": "32ca007c-c2a9-11e6-a521-005056831b41",

"endUserId": "acr:1a33fb6b-6a299ff7-e1b8f827-152ed209-0e89b8a4"

},

{

"subscriptionInformation": {

"amount": 10,

"currency": "EUR",

"description": "Test Vanessa"

},

"subscriptionMetaData": {

"rrule": {

"freq": "WEEKLY",

"interval": 1,

"dtstart": "2016-12-15T15:03:16.000+01:00",

"until": "2016-12-22T15:03:16.000+01:00"

},

"productId": "3292",

"serviceID": "AF0005",

"onBehalfOf": "SFR Vanessa"

},

"mandateId": "3a0a43bc-c2cf-11e6-8df8-0050568302c6",

"endUserId": "acr:1a33fb6b-6a299ff7-e1b8f827-152ed209-0e89b8a4"

}

]

}

## STATUS CODES & EXCEPTIONS

### Status Codes

The http status code categorizes the answer:

|  |  |
| --- | --- |
| **200 Success** | OK |
| **201 Created** | The ressource has been created |
| **400 Bad request** | See the error message for more informations |
| **401 Authentication failure** | Check your authentification |
| **403 Forbidden** | Provide your authentication settings |
| **404 Not found** | hostname or path error in the service URI |
| **405 Method not supported** | Invalid use of the method, HTTP GET instead of POST for example. |
| **409 Conflict** | The request cannot be processed at the current state |
| **429 Too Many Requests** | The customer has made too many requests in a given time |
| **500 The server encountered an unexpected condition** | This may be caused to a misconfigured account or limited user permissions |
| **502 Bad Gateway** | Wrong answer sent to an intermediate server by another server |
| **503 Server busy and service unavailable** | Try again later |

### Structure for error messages

HTTP/1.1 400 Bad Request

Content-Type: application/json

Content-Length: 1234

Date: Thu, 04 Jun 2009 02:51:59 GMT

{

"requestError": {

"serviceException": {

"messageId": "SVC0002",

"text": "Invalid input value for message part %1",

"variables": "tel:+33603100000"

}

}

}

In case of error when calling the requests described above, the message structure is as follows:

**Table 12: Type requestError**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Type** | **Description** | **Optional** |
| **serviceException ou**  **policyException** | **serviceException ou**  **policyException** | Data structure containing information about the error.  **(see table 12.1)**. | No |

#### Table 12.1: Type serviceException ou policyException

|  |  |  |  |
| --- | --- | --- | --- |
| **Paramètre** | **Type** | **Description** | **Optionnel** |
| **messageId** | string | Error identifier | No |

**Text** string Error description No

|  |  |  |  |
| --- | --- | --- | --- |
| **Variables** | string | Parameter value at the origin of the error | Yes |

### Service Exceptions

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **EXCEPTION TEXT** | **VARIABLES** | **HTTP RESPONSE CODE** |
| **SVC0001** | An error has occurred following a customer call on the service. Contact the support team. |  | 500 The server encountered an unexpected condition |
| **SVC0002** | An input parameter is not as expected type. Check the types of parameters and resubmit your request. |  | 400 Bad Request |
| **SVC0004** | The customer’s MSISDN does not exist in SIMPA. |  | 400 Bad Request |
| **SVC0007** | Invalid charging information | none | 400 Bad Request |
| **SVC0270** | Charging operation failed, the charge was not applied | none | 400 Bad Request |

**SVC0271** Refunds not supported Guidance from the implementation on what to do 400 Bad Request

instead should be provided

|  |  |  |  |
| --- | --- | --- | --- |
| **SVC0273** | Refund failed | The reason the refund failed. Valid reasons include:    The user did not accept the refund  The refund request is for an amount greater than the original charge | 400 Bad Request |
| **SVC2000** | The customer’s MSISDN does not exist in SIMPA. | | 400 Bad Request |
| **SVC3000** | Parameters error | | 400 Bad Request |
| **SVC3002** | The customer has a Postpaid subscription, but the balance check states that he is over the credit limit. | | 409 Conflict |
| **SVC3003** | The customer has a prepaid subscription, but has insufficient amount on the IN node | | 409 Conflict |
| **SVC3004** | The customer has a prepaid subscription, but no contact can be made to the IN node | | 502 Bad Gateway |
| **SVC3005** | No authorization event could be found to allow the transaction | | 401 Unauthorized |
| **SVC3006** | The authorization transaction failed with this transaction Id. | | 401 Unauthorized |
| **SVC3008** | Number of transactions limit is reached for postpaid customer. | | 429 Too Many Requests |
| **SVC3009** | Number of transactions limit is reached for prepaid customer. | | 429 Too Many Requests |
| **SVC3999** | Un paramètre a une valeur invalide | | 400 Bad Request |
| **SVC4000** | Database error | | 500 The server encountered an unexpected condition |
| **SVC5000** | Internal error | | 500 The server encountered an unexpected condition |

Guidance from the implementation on what to do

### Policy Exceptions

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **EXCEPTION TEXT** | **VARIABLES** | **HTTP RESPONSE CODE** |
| **POL-0006** | The maximum number of transactions per second is exceeded. Make sure your number of requests is within the limits set in your SLA, for example, 10 TPS (transactions per second). |  | 429 Too Many Requests |
| **POL-0008** | The authentication parameter of the third party application are incorrect. Check the username and password "basic auth" are correct and resubmit your request. |  | 401 Authentication  Failure |
| **POL-0014** | Une liste blanche est appliquée et le numéro n'est pas dans la liste blanche. Vérifiez les informations de votre SLA. |  | 500 The server encountered an unexpected condition |
| **POL-0015** | A whitelist is applied and the number is not in the whitelist. Check the information in your SLA. |  | 500 The server encountered an unexpected condition |
| **POL-0016** | The maximum number of requests for this service is exceeded. Contact the support team. |  | 429 Too Many Requests |
| **POL-0017** | The method / operation is not supported in your SLA. Check your SLA and use a method that is supported. |  | 404 Not Found |
| **POL-0040** | A maximum destination address is applied and it was exceeded. Check your SLA for the the limit and resubmit your request. |  |  |
| **POL-0064** | RESERVED TRANSACTION NOT FOUND |  | 400 Bad Request |
| **POL-0253** | Payment operation refused by user | aucun | 403 Forbidden |
| **POL-0254** | The amount exceeds the operator limit for a single charge | aucun | 403 Forbidden |
| **POL-1000** | User has insufficient credit for transaction | aucun | 403 Forbidden |
| **POL-1001** | The %1 operator charging limit for this user has been exceeded | %1 – the time period for which the charging limit has been reached, eg hourly, daily, weekly, monthly. | 403 Forbidden |
| **POL-1002** | The charge happened too soon after the previous one | aucun | 403 Forbidden |
| **POL-1003** | The refund exceeds the original amount charged %1 | %1 – the original amount charged | 403 Forbidden |
| **POL-1004** | Refund below minimum refundable amount supported %1 | %1 – the minimum refundable amount supported | 403 Forbidden |
| **POL-1005** | A refund request requires the originalServerReferenceCode for the charge that is being refunded | aucun | 403 Forbidden |
| **POL-1006** | originalServerReferenceCode not valid | aucun | 403 Forbidden |
| **POL-1007** | Refunds not supported | aucun | 403 Forbidden |
| **POL-1008** | Alternative payment (credit card / other No-operator payment) method was declined | aucun | 403 Forbidden |
| **POL-2001** | The status of the customer is not active. |  |  |
| **POL-2002** | The baring level is set for all services |  | 401 Unauthorized |
| **POL-0100** | MSISDN is not eligible for this service. |  | **401 Unauthorized** |

|  |  |  |
| --- | --- | --- |
| **POL-2003** | The customer has a baring on operator services | 401 Unauthorized |
| **POL-4904** | Maximum number of requests per second achieved | 429 Too Many Requests |
| **POL-5002** | The requested action has been performed. | 400 Bad Request |