

## Functional Description Sandbox

### Table of Content

|  |    |
|--|----|
| Versioning.....  | 2  |
| 1. Sandbox environment – description .....   | 3  |
| 1.1 Test our APIs – instructions .....   | 3  |
| 2. Sandbox Accounts API – description .....  | 4  |
| 2.1 Account Information API.....   | 4  |
| 2.1.1 Test users - GET /Accounts.....  | 4  |
| 2.1.2 Test users - GET /Accounts/{accountId} .....                                 | 5  |
| 2.1.3 Test users - GET /Accounts/{accountId}/balances.....                         | 6  |
| 2.1.4 Test users - GET /Accounts/{accountId}/transactions.....                     | 6  |
| 2.1.5 Test users - GET /Accounts/{accountId}/transactions/{transactionId}.....     | 7  |
| 3 Sandbox Payment API - description.....   | 8  |
| 3.1 Payments Initiation API .....  | 8  |
| 3.1.1 Test users – POST /Payments/{paymentProduct} .....                           | 9  |
| 3.1.2 Test users – GET /Payments/{paymentProduct}/{paymentId}.....                 | 10 |
| 3.1.3 Test users – DELETE /Payments/{paymentProduct}/{paymentId}.....              | 11 |
| 3.1.4 Test users – GET /Payments/{paymentProduct}/{paymentId}/status.....          | 11 |
| 3.1.5 Test users – POST /PeriodicPayments/{paymentsProduct} .....                  | 13 |
| 3.1.6 Test users – GET /PeriodicPayments/{paymentsProduct}/{payemntId} .....       | 13 |
| 3.1.7 Test users – DELETE /PeriodicPayments/{paymentProduct}/{paymentId} .....     | 14 |
| 3.1.8 Test users – GET /PeriodicPayments/{paymentProduct}/{paymentId}/status ..... | 15 |
| 3.1.9 Test users – POST /SigningBasket.....  | 16 |
| 3.1.10 Test users – GET /SigningBasket /{basketId}.....                            | 17 |
| 3.1.11 Test users – PUT /SigningBasket/{basketId}.....                             | 18 |

## Versioning

| Version | Release Day     |
|---------|-----------------|
| V1.0    | 5th May, 2020   |
| V1.1    | 2nd June, 2020  |
| V1.2    | 23rd June, 2020 |
| V1.3    | 3rd June, 2021  |
|         |                 |

**Note:** Functionality described in grey text below is as of for now not available to test with different test users, in Sandbox Payment API.

## 1. Sandbox environment – description

Sandbox API environment gives TPP an opportunity to test your application before accessing production API's. When calling Sandbox API we provide you with predefined data in the access token which allows you to perform tests below.

Sandbox API's environment you don't need to register a certificate or use BankID as authentication. To get access to Sandbox API environment you need to [SignIN](#) on Developer Portal.

### 1.1 Test our APIs – instructions

The test users in the dynamic sandbox consist private individuals and gives you opportunity to test different cases by using social security number (SSN) provided below in Sandbox API descriptions, when to generate a test access token by following client credentials code grant flow in Developer Portal. A test access token needs to be provided in authorization flow to access the various test user's data.

Test access token has a lifetime of 1 hour.

Please follow steps below on how to get access token in Sandbox.

- Sign-in to developer account. <https://apim-icabanken.ica.se/store>
- Go to "Applications" tab and create application "psd2\_sandbox" in Developer Portal.
- Go to "Subscriptions" tab and click on "API listing store" and select the Sandbox API, select application and subscribe to it
- To view subscription, select "View subscription" button
- Go to "Sandbox Key" tab and click on Generate button to get client id, client secret
- Use any REST Client to get the access token from below mentioned REST call and test-SSN number you can find in detailed description of services for each API. Please see pages 4-17.

*Request:*

```
curl-X POST https://ims.icagruppen.se/oauth/v2/token
-H "content-type:application/x-www-form-urlencoded"
-d "grant_type=client_credentials&client_id=client_id&client_secret=client_secret&scope=psd2_sandbox:<SSN>"
```

*Response:*

```
{"access_token":"f8bfe807-xxxx-xxxx-xxxx-xxxxxx","scope":"psd2_sandbox:<SSN>","token_type":"bearer","expires_in":3600}
```

\*Replace <SSN> with 12-digit social security number, e.g. 19880809xxxx

- Select "API Console" tab to get the list of all endpoints in API.
- Provide your access token in header "Authorization : Bearer "
- Select the endpoint you want to test.
- Select "Try it out".

Note, in the services where there are fields *\*required* enter any value. Sandbox data is hardcoded so please follow instructions below for which data you need to provide in request to be able to get successful response.

- Select "Execute"
- Get the endpoint response

## 2. Sandbox Accounts API – description

### Description

The Account Information API gives you access to account data for accounts within the PSD2 scope, belonging to customers of ICA Banken. Account data access requires consent from the customer.

The API includes the following main features: retrieve a list of available accounts, retrieve balances and transaction details for a given account.

Accounts that is included in the scope are accounts with one or two account holders (joint accounts) as well as accounts connected to a card at ICA Banken.

### 2.1 Account Information API

The Account Information API gives you access to account data for payment accounts within the PSD2 scope, belonging to customers of ICA Banken. Account data access requires consent from the customer.

[https://apim-icabanken.ica.se/store/apis/info?name=ICA.Bank.Services.PSD2.Accounts.Sandbox&version=1.0.0&provider=ICA.S  
E%2Fexfjrr%40icabanken.tenant&tenant=icabanken.tenant](https://apim-icabanken.ica.se/store/apis/info?name=ICA.Bank.Services.PSD2.Accounts.Sandbox&version=1.0.0&provider=ICA.SE%2Fexfjrr%40icabanken.tenant&tenant=icabanken.tenant)

The API includes the following main features:

#### **GET /Accounts** - Valid accounts for user

##### **Description:**

Returns a list of a user's accounts, where the user is account holder or disposition holder. ICA Banken will always return all active accounts that are PSD2 supported - there is no detailed consent that limits account accessibility. Terminated accounts will not be shown in the list.

The following account types are available for ICA Banken:

**ICA Konto** - A transaction account, with the possibility to connect a card and/or payment services.

Account information services can be called unlimited times if the end user is actively requesting information. According to PSD2 regulation account information services can be called 4 times a day without the end user actively requesting information and this is 4 times per day. To indicate that the end user is actively requesting account information, please provide the IP address of the end user in the header 'PSU-IP-Address' for the relevant operation call; otherwise omit IP address in the header.

#### 2.1.1 Test users - GET /Accounts

- **SSN:** 197301175209
- **withBalance:** false
- **Sandbox data:** customer with accounts and no balance
- **SSN:** 195505063676
- **withBalance:** true
- **Sandbox data:** customer with accounts and with balance
- **SSN:** 196311210063
- **withBalance:** true
- **Sandbox data:** customer with accounts with balance

- SSN: [201011119789](#)
  - Sandbox data: customer under 16 years old returns error message (customer must be 16 or older to access)
  - SSN: [198312112678](#)
  - Sandbox data: customer with invalid KYC information returns error message
  - ACCESS TOKEN: Leave the field empty
  - Sandbox data: You will get an error message
- 

## GET /Accounts/{accountId} - Detailed account information

### Description:

Returns detailed account information, with or without balances for one specific account. This endpoint returns more detailed information about the account than the account list.

### 2.1.2 Test users - GET /Accounts/{accountId}

- SSN: [195505063676](#)
  - AccountID: 12345678910
  - withBalance: true
  - Sandbox data: customer with accounts, with credit and balance
  - SSN: [196311210063](#)
  - AccountID: 19283764502
  - withBalance: true
  - Sandbox data: customer with accounts, without credit and with balance
  - SSN: [197301175209](#)
  - AccountID: 98765432103
  - withBalance: false
  - Sandbox data: customer with accounts and no balance
  - SSN: [198904153866](#)
  - AccountID: 12345678919
  - withBalance: true
  - Sandbox data: customer with accounts and with balance
  - SSN: [201011119789](#)
  - Sandbox data: customer under 16 years old returns error message (customer must be 16 or older to access)
  - SSN: [198312112678](#)
  - Sandbox data: customer with invalid KYC information returns error message
  - ACCESS TOKEN: Leave the field empty
  - Sandbox data: You will get an error message
-

## GET /Accounts/{accountId}/balances - Balances of requested account

### Description:

Returns balances for booked and reserved transactions and available amount (including any credit limit) for the account.

### 2.1.3 Test users - GET /Accounts/{accountId}/balances

- SSN: **195505063676**
- AccountID: 12345678910
- Sandbox data: returns information about a specific account, with credit
- SSN: **197301175209**
- AccountID: 98765432103
- Sandbox data: returns information about a specific account, without credit
- SSN: **201011119789**
- Sandbox data: customer under 16 years old returns error message (customer must be 16 or older to access)
- SSN: **198312112678**
- Sandbox data: customer with invalid KYC information returns error message
- ACCESS TOKEN: Leave the field empty
- Sandbox data: You will get an error message

---

## GET /Accounts/{accountId}/transactions – Transactions for a specific account

### Description:

Returns a list of transactions for one specific account. You can search on account with or without a date period. Transactions in the last 30 days is presented default if no dates specified. If no transaction within 30 days, please specify dates. The client may experience timeouts if the date span is too large. In case of a timeout, please shorten the date span. From (start date) cannot be older than 18 months from the date of the request. The transaction list contains both booked transactions and pending transactions (reserved amounts)

### 2.1.4 Test users - GET /Accounts/{accountId}/transactions

- SSN: **195505063676**
- AccountID: 12345678910
- Status: booked
- Sandbox data: customer with account, without credit. Transactions included: PG/BG and card transaction abroad
- SSN: **197301175209**
- AccountID: 98765432103
- Status: booked
- Sandbox data: customer with account, without credit. Transactions included: internal transfers, both between own accounts and to another person's account, card transaction and Swish.
- SSN: **196311210063**
- AccountID: 19283764502
- Status: both
- Sandbox data: customer with account. Transactions included: internal transfers between own accounts and pending card transactions.

- SSN: [198904153866](#)
- AccountID: 10000000001  
Status: Booked
- Sandbox data: customer with account. Transactions included: ATM withdrawal, internal transfers between own accounts, direct debit and cross border payments EUR.
- SSN: [201011119789](#)
- Sandbox data: customer under 16 years old returns error message (customer must be 16 or older to access)
- SSN: [198312112678](#)
- Sandbox data: customer with invalid KYC information returns error message
- ACCESS TOKEN: Leave the field empty
- Sandbox data: You will get an error message

---

## GET /Accounts/{accountId}/transactions/{transactionId} – Details of a specific transaction

### Description:

Returns transaction details on a specific transaction.

Depending on transaction type, different fields with details will be returned, e.g. place of purchase, original amount, exchange rate etc.

### 2.1.5 Test users - GET /Accounts/{accountId}/transactions/{transactionId}

- SSN: [195505063676](#)
- AccountID: 12345678910
- TransactionID: 00000003116265606
- Sandbox data: customer with account. Returns detailed information about a specific PG transaction.
- SSN: [197301175209](#)
- AccountID: 12345678910
- TransactionID: 00000003114459127
- Sandbox data: customer with account. Returns detailed information about a specific BG transaction
- SSN: [196311210063](#)
- AccountID: 12345678910
- TransactionID: 00000003116364886
- Sandbox data: customer with account. Returns detailed information about a specific Internal transfer
- SSN: [198904153866](#)
- AccountID: 12345678910
- TransactionID: 00000003116116872
- Sandbox data: customer with account. Returns detailed information about a specific Swish transaction
- SSN: [194112010014](#)
- AccountID: 12345678910
- TransactionID: 00000003116262615
- Sandbox data: customer with account. Returns detailed information about a specific External transfer
- SSN: [197002022692](#)
- AccountID: 12345678910
- TransactionID: 00000003109932323

- **Sandbox data:** customer with account. Returns detailed information about a specific Cross border payment
  - **SSN:** [201011119789](#)
  - **Sandbox data:** customer under 16 years old returns error message (customer must be 16 or older to access)
  - **SSN:** [198312112678](#)
  - **Sandbox data:** customer with invalid KYC information returns error message
  - **ACCESS TOKEN:** Leave the field empty
  - **Sandbox data:** You will get an error message
- 

## 3 Sandbox Payment API - description

The Payment initiation API gives you the possibility to help ICA Banken customer to initiate payments from ICA Banken accounts through your application or service, allowing the customer to transfer money to any Swedish account, transfer money between ICA Banken accounts, initiate Bankgiro and Plusgiro payments, initiate International payments.

### 3.1 Payments Initiation API

The Payment initiation API gives you the possibility to help ICA Banken customer to initiate payments from ICA Banken accounts through your application or service, allowing the customer to transfer money to any Swedish account, transfer money between ICA Banken accounts, initiate Bankgiro and Plusgiro payments, initiate International payments.

<https://apim-icabanken.ica.se/store/apis/info?name=ICA.Bank.Services.PSD2.Payments.Sandbox&version=1.0.0&provider=ICA.SE%2Fexfjrr%40icabanken.tenant&tenant=icabanken.tenant>

The Payment API also include endpoint for Funds Confirmation, which can be used for checking whether a specific requested amount is available on a given payment account. Returns true or false to indicate if funds are available.

#### POST /FundsConfirmation

##### Description

The Funds confirmation API can be used for checking whether a specific requested amount is available on a given payment account. Returns true or false to indicate if funds are available.

Test users not available due to Fundsconfirmation API not yet implemented in production.

---



## POST /Payments/{paymentProduct} – Initialize a new payment

### Description

Creates a payment initiation from the provided payment information. This is the first step in the API to initiate the related payment.

---

### 3.1.1 Test users – POST /Payments/{paymentProduct}

- **SSN:** 195505063676
  - **PaymentProduct:** SepaCreditTransfers
  - **AccountID:** 12345678910
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about initiated a specific payment bankgiro {BG} payment with OCR number
- 
- **SSN:** 197301175209
  - **PaymentProduct:** SepaCreditTransfers
  - **AccountID:** 12345678910
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about initiated a specific payment plusgiro {PG} payment with text message
- 
- **SSN:** 196311210063
  - **PaymentProduct:** SepaCreditTransfers
  - **AccountID:** 12345678910
  - **TransactionStatus:** ACSC
  - **Sandbox data:** customer with account. Returns detailed information about initiated a specific transfer, internal transfer (transfer between customers own ICA accounts)
- 
- **SSN:** 198904153866
  - **PaymentProduct:** SepaCreditTransfers
  - **AccountID:** 12345678910
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about initiated a specific transfer, external transfer with no requested execution date
- 
- **SSN:** 194112010014
  - **PaymentProduct:** CrossBorderCreditTransfers
  - **AccountID:** 12345678910
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about initiated a specific international payment in EUR
- 
- **ACCESS TOKEN:** 468c85a8-1a20-4626-81bf-6acf66334451 or any value
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get error message access token expired.
- 
- **SSN:** 201011119789
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** customer with wrong date in requested execution date, returns error message
- 
- **SSN:** 197209139240
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** customer have entered wrong currency code.
-

**GET /Payments/{paymentProduct}/{paymentId}** – Fetches a specific payment

## Description

Returns a specific payment that were previously posted. Any additional fields that are not part of the chosen payment product will be stripped away.

### 3.1.2 Test users – GET /Payments/{paymentProduct}/{paymentID}

- **SSN:** 195505063676
  - **PaymentProduct:** SepaCreditTransfers
  - **AccountID:** 12345678910
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about initiated a specific payment, bankgiro {BG} with OCR number
  - **SSN:** 197301175209
  - **PaymentProduct:** SepaCreditTransfers
  - **AccountID:** 12345678910
  - **TransactionStatus:** ACTC
  - **Sandbox data:** customer with account. Returns detailed information about initiated a specific payment, plusgiro {PG} with text message
  - **SSN:** 196311210063
  - **PaymentProduct:** SepaCreditTransfers
  - **AccountID:** 12345678910
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about initiated a specific payment, internal transfer (transfer between customers own ICA accounts)
  - **SSN:** 198904153866
  - **PaymentProduct:** SepaCreditTransfers
  - **AccountID:** 12345678910
  - **TransactionStatus:** ACTC
  - **Sandbox data:** customer with account. Returns detailed information about initiated a specific payment, external transfer with no requested execution date
  - **SSN:** 194112010014
  - **PaymentProduct:** CrossBorderCreditTransfers
  - **AccountID:** 12345678910
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about a specific payment, international payment in EUR.
  - **ACCESS TOKEN:** 468c85a8-1a20-4626-81bf-6acf66334451 or any value
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get error message access token expired.
  - **SSN:** 201011119789
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** customer with wrong status on the specific payment, returns error message
  - **ACCESS TOKEN:** Leave the field empty
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get an error message
-

## **DELETE /Payments/{paymentProduct}/{paymentId}** – Delete a payment

### **Description**

Deletes a specific payment. Only future dated payments that has not been processed can be deleted.

Note: In order to delete a signed payment a valid signed token (scope payment delete:<ID>) is required.

### 3.1.3 Test users – DELETE /Payments/{paymentProduct}/{paymentId}

- **SSN:** 195505063676
  - **PaymentProduct:** SepaCreditTransfers
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information when deleting a specific payment, bankgiro {BG} with OCR number
  - **SSN:** 198904153866
  - **PaymentProduct:** SepaCreditTransfers
  - **TransactionStatus:** ACTC
  - **Sandbox data:** customer with account. Returns detailed information when deleting a specific payment, external transfer
  - **SSN:** 194112010014
  - **PaymentProduct:** CrossBorderCreditTransfers
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information when deleting a specific international payment in EUR
  - **ACCESS TOKEN:** 468c85a8-1a20-4626-81bf-6acf66334451 or any value
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get error message access token expired.
  - **SSN:** 201011119789
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** customer with wrong status on the specific payment, returns error message
  - **ACCESS TOKEN:** Leave the field empty
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get an error message
- 

## **GET /Payments/{paymentProduct}/{paymentId}/status** - Get status of a specific payment

### **Description**

Get status of a specific payment that was previously posted.

### 3.1.4 Test users – GET /Payments/{paymentProduct}/{paymentId}/status

- **SSN:** 195505063676
- **PaymentProduct:** SepaCreditTransfers
- **Status:** ENTERED
- **TransactionStatus:** RCVD
- **Sandbox data:** customer with account. Returns detailed information about status of a specific payment, bankgiro {BG} with OCR number

- **SSN:** 197301175209
  - **PaymentProduct:** SepaCreditTransfers
  - **Status:** PROCESSED
  - **TransactionStatus:** ACSC
  - **Sandbox data:** customer with account. Returns detailed information about status of a specific payment, plus giro {BG} with text message
  - **SSN:** 198904153866
  - **PaymentProduct:** SepaCreditTransfers
  - **Status:** VERIFIED
  - **TransactionStatus:** ACTC
  - **Sandbox data:** customer with account. Returns detailed information about status of a specific payment, external transfer, with an execution date
  - **SSN:** 196311210063
  - **PaymentProduct:** SepaCreditTransfers
  - **Status:** CANCELLED
  - **TransactionStatus:** CANC
  - **Sandbox data:** customer with account. Returns detailed information about status of a specific payment, internal transfer
  - **SSN:** 194112010014
  - **paymentProduct:** CrossBorderCreditTransfers
  - **Statatus:** ENTERED
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with an account. Returns information about status of a specific payment, international payment.
  - **ACCESS TOKEN:** 468c85a8-1a20-4626-81bf-6acf66334451 or any value
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get error message access token expired.
  - **SSN:** 201011119789
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** customer with wrong paymentID, returns error message
  - **ACCESS TOKEN:** Leave the field empty
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get an error message
-

## **POST /PeriodicPayments/{paymentProduct}** - Creates a recurring payment

### **Description**

Creates a standing order initiation for recurring i.e. periodic payment. This is the first step in the API to initiate the related recurring/periodic payment.

Note! Recurring payments cannot be applied to the following payment products: Bankgiro Plusgiro or International Payments.

### 3.1.5 Test users – POST /PeriodicPayments/{paymentsProduct}

- **SSN:** 195505063676
  - **PaymentProduct:** SepaCreditTransfers
  - **Status:** ENTERED
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about initiated specific recurrent payment, internal recurring transfer between customers own ICA Accounts.
  - **SSN:** 198904153866
  - **PaymentProduct:** SepaCreditTransfers
  - **Status:** ENTERED
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about initiated specific recurrent payment, external recurring transfer
  - **ACCESS TOKEN:** 468c85a8-1a20-4626-81bf-6acf66334451 or any value
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get error message access token expired.
  - **ACCESS TOKEN:** Leave the field empty
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get an error message
- 

## **GET /PeriodicPayments/{paymentProduct}/{paymentId}** - Details about a specific recurring payment

### **Description**

Returns details about a specific recurring payment. Any additional fields that are not part of the chosen payment product will be stripped away.

### 3.1.6 Test users – GET /PeriodicPayments/{paymentsProduct}/{payemntId}

- **SSN:** 195505063676
- **PaymentProduct:** SepaCreditTransfers
- **Status:** ENTERED
- **TransactionStatus:** RCVD
- **Sandbox data:** customer with account. Returns detailed information about a specific recurrent payment, internal transaction between customer own ICA accounts
- **SSN:** 198904153866
- **PaymentProduct:** SepaCreditTransfers
- **Status:** VERIFIED
- **TransactionStatus:** ACTC

- **Sandbox data:** customer with account. Returns detailed information about a specific recurrent payment, external transaction
  - **ACCESS TOKEN:** 468c85a8-1a20-4626-81bf-6acf66334451 or any value
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get error message access token expired.
  - **SSN:** 201011119789
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** customer with wrong paymentID, returns error message
  - **ACCESS TOKEN:** Leave the field empty
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get an error message
- 

## **DELETE /PeriodicPayments/{paymentProduct}/{paymentId}** - Delete a recurring payment mandate

### **Description**

Deletes a recurring payment mandate.

Note: In order to delete a signed mandate a valid signed token (scope payment delete:<ID>) is required.

### 3.1.7 Test users – DELETE /PeriodicPayments/{paymentProduct}/{paymentId}

- **SSN:** 195505063676
  - **PaymentProduct:** SepaCreditTransfers
  - **Status:** CANCELLED
  - **TransactionStatus:** CANC
  - **Sandbox data:** customer with account. Returns detailed information about deleting a specific recurrent payment, internal transaction between customer own ICA accounts
  - **SSN:** 198904153866
  - **PaymentProduct:** SepaCreditTransfers
  - **Status:** CANCELLED
  - **TransactionStatus:** CANC
  - **Sandbox data:** customer with account. Returns detailed information about deleting a specific recurrent payment, external transaction
  - **ACCESS TOKEN:** 468c85a8-1a20-4626-81bf-6acf66334451 or any value
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get error message access token expired.
  - **SSN:** 201011119789
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** customer with wrong status on the payment, returns error message
  - **ACCESS TOKEN:** Leave the field empty
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get an error message
-

## GET /PeriodicPayments/{paymentProduct}/{paymentID}/status – Status of a specific payment

### Description

Get status of a specific recurring payment mandate.

### 3.1.8 Test users – GET /PeriodicPayments/{paymentProduct}/{paymentID}/status

- **SSN:** 195505063676
  - **PaymentProduct:** SepaCreditTransfers
  - **Status:** UNSIGNED
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about status of a specific recurring payment, internal transaction between customers own ICA accounts
  - **SSN:** 198904153866
  - **PaymentProduct:** SepaCreditTransfers
  - **Status:** CANCELLED
  - **TransactionStatus:** CANC
  - **Sandbox data:** customer with account. Returns detailed information about status of a specific recurring payment, external transfer
  - **SSN:** 197301175209
  - **PaymentProduct:** SepaCreditTransfers
  - **Status:** OPEN
  - **TransactionStatus:** ACTC
  - **Sandbox data:** customer with account. Returns detailed information about status of a specific payment, internal transaction between two accounts in the ICA Banken
  - **ACCESS TOKEN:** 468c85a8-1a20-4626-81bf-6acf66334451 or any value
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get error message access token expired.
  - **SSN:** 201011119789
  - **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** customer with wrong paymentID, returns error message
- ACCESS TOKEN:** Leave the field empty
- **PaymentProduct:** SepaCreditTransfers
  - **Sandbox data:** You will get an error message
-

## **POST /SigningBasket** – Creates a new signing basket

### **Description**

Creates a signing basket to enable authorization of several payments with one SCA method.

Note: After creating a signing Basket a call to /ApproveSigningBasket with a valid signed token (scope Basket:<ID>) is required.

### 3.1.9 Test users – POST /SigningBasket

- **SSN:** 195505063676
  - **BasketID:** 427
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about created signing basket with 1 paymentID
  - **SSN:** 198904153866
  - **BasketID:** 589
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about created signing basket with 3 paymentsIDs
  - **SSN:** 197301175209
  - **BasketID:** 620
  - **TransactionStatus:** RCVD
  - **Sandbox data:** customer with account. Returns detailed information about created signing basket with different kind of payments, paymentsIDs
  - **ACCESS TOKEN:** 468c85a8-1a20-4626-81bf-6acf66334451 or any value
  - **BasketID:** 111
  - **Sandbox data:** You will get error message access token expired.
  - **SSN:** 201011119789
  - **Sandbox data:** customer with wrong paymentID, returns error message
  - **ACCESS TOKEN:** Leave the field empty
  - **Sandbox data:** You will get an error message
-



## GET /SigningBasket /{basketId} - Get a specific signing basket

### Description

Get a specific signing basket that were previously posted.

#### 3.1.10 Test users – GET /SigningBasket /{basketId}

- SSN: **195505063676**
  - BasketID: 427
  - TransactionStatus: RCVD
  - Sandbox data: customer with account. Returns detailed information about specific signing basket with 1 payment, paymentID
  - SSN: **198904153866**
  - BasketID: 589
  - TransactionStatus: ACTC
  - Sandbox data: customer with account. Returns detailed information about specific signing basket with 3 payments, paymentsIDs
  - SSN: **197301175209**
  - BasketID: 620
  - TransactionStatus: RCVD
  - Sandbox data: customer with account. Returns detailed information about created signing basket with different kind of payments, paymentsIDs
  - ACCESS TOKEN: 468c85a8-1a20-4626-81bf-6acf66334451 or any value
  - BasketID: 111
  - Sandbox data: You will get error message access token expired.
  - SSN: **201011119789**
  - BasketID: 523
  - Sandbox data: customer with wrong basketID, returns error message
  - ACCESS TOKEN: Leave the field empty
  - BasketID: 756
  - Sandbox data: You will get an error message
-

## **PUT /SigningBasket/{basketId}** - Sign a specific signing basket

### **Description**

Validates signed token (scope Basket:<ID>) and signs all the payments in the basket for execution.

#### 3.1.11 Test users – PUT /SigningBasket/{basketId}

- **SSN:** 195505063676
  - **BasketID:** 427
  - **TransactionStatus:** ACTC
  - **Sandbox data:** customer with account. Returns detailed information about a specific signing basket with 1 payment, paymentID
  
  - **SSN:** 198904153866
  - **BasketID:** 589
  - **TransactionStatus:** ACTC
  - **Sandbox data:** customer with account. Returns detailed information about a specific signing basket with 3 paymentsIDs
  
  - **SSN:** 197301175209
  - **BasketID:** 620
  - **TransactionStatus:** ACTC
  - **Sandbox data:** customer with account. Returns detailed information about a specific signing basket with different kind of payments, paymentsIDs
  
  - **ACCESS TOKEN:** 468c85a8-1a20-4626-81bf-6acf66334451 or any value
  - **BasketID:** 111
  - **Sandbox data:** You will get error message access token expired.
  
  - **ACCESS TOKEN:** Leave the field empty
  - **BasketID:** 756
  - **Sandbox data:** You will get an error message
-