

**Skandiabanken**  
**Technical documentation Open Banking**

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	<i>Background</i>	3
1.2	<i>Status</i>	3
1.3	<i>Contact us and channels for communication</i>	3
<b>2</b>	<b>Test environment and sandbox</b>	<b>4</b>
2.1	<i>General information</i>	4
2.2	<i>Sandbox with static data</i>	4
<b>3</b>	<b>Onboarding of TPP</b>	<b>5</b>
3.1	<i>Process overview</i>	5
3.2	<i>Create an account in the portal</i>	5
3.3	<i>How top setup your apps and subscriptions</i>	5
<b>4</b>	<b>Security and authentication and of TPP including SCA of PSU</b>	<b>9</b>
4.1	<i>OAuth2</i>	9
4.2	<i>Request Access code and ultimately Access token</i>	9
<b>5</b>	<b>General information about API invocation</b>	<b>11</b>
5.1	<i>Gateway</i>	11
5.2	<i>Headers</i>	11
<b>6</b>	<b>AIS-operations</b>	<b>12</b>
6.1	<i>Introduction</i>	12
6.2	<i>Overview of available accounts and customers</i>	12
6.3	<i>Read Account List</i>	13
6.4	<i>Read Account Details</i>	15
6.5	<i>Read Balance List</i>	17
6.6	<i>Read transaction list of an account</i>	19
<b>7</b>	<b>PIS-operations</b>	<b>23</b>
7.1	<i>Introduction</i>	23
7.2	<i>Overview of possible payments to initiate</i>	23
<b>8</b>	<b>Document revision history</b>	<b>24</b>

## 1 Introduction

Welcome to Skandiabanken Open Banking-solutions!

### 1.1 Background

This document and information found at our information-site is created to provide the information you need to access our APIs. If you have questions or suggestions, you are welcome to contact us [here](#).

**Info site:** [www.skandia.se/openbanking](http://www.skandia.se/openbanking)

**Portal:** <https://developer.skandia.se/open-banking/core-bank/>

This document is found on the information site. A glossary of abbreviations and acronyms is also found on the information site.

### 1.2 Status

This section will clarify what we have in place and the status of the functionality. Please read the rest of this documentation for further information and details.

Functionality	Available	Comments
SCA	Yes	A redirect solution using Swedish BankID. We are investigating embedded/decoupled solutions.
TPP-identification (eIDAS)	Yes	We support eIDAS/QWAC. QSealC is currently not required but we are investigating it.
Relevant standards	Yes	Berlin Group NextGenPSD2 Framework version 1.3.3 OAuth2 including OpenID Connect
AIS	Yes	Payment account information for private customers over 18 years is available for 90 days. We have no business customers with payment accounts.
PIS	No	Domestic transfers with same date will be available soon.  Other PIS such as bill payments, international payments etc will be available later.
PIIS	No	Not currently available

### 1.3 Contact us and channels for communication

If you have questions, feedback or general problems you are welcome to contact us via mail [openbanking@skandia.se](mailto:openbanking@skandia.se). This mailbox is mainly monitored during business hours. Please use this for technical questions or business inquiries.

If you have very urgent issues outside business hours, please contact our customer and TPP support: <https://www.skandia.se/kontakta-skandia/kontakta-oss/>.

Mainly use phone: +46771555500

Open hours 06-24 all days, year around.

## 2 Test environment and sandbox

### 2.1 General information

We currently have a very limited test environment where we offer test with static mocked data for three operations of the AIS-services.

We currently do not have a test portal available.

### 2.2 Sandbox with static data

Testing with static mocked data is performed by calling the endpoints from Postman or your client of choice. Note that responses are static, you will get the same response regardless of input. However, you will get a good notion what the requests and responses look like.

The endpoint is anonymously exposed:

- No registration of TPP/app/subscription in Portal is required
- No use of eIDAS certificate or OAuth2 tokens
- No use of SCA for PSU

HTTPS is the only supported protocol in the static mocked services.

Note: The response will include two product specific headers, X-RateLimit-Limit and X-RateLimit-Remaining that should be ignored.

The following operations for AIS in mocked service are currently available:

- Read Account List
- Read Balance
- Read transaction list of an account

General information for all static AIS operations:

Host: api.test.skandia.se

Schemes: https

Basepath: /open-banking/sandbox

For each operation in this document you will find an example for calling the static test operation.

PIS operations will be available later.

### 3 Onboarding of TPP

#### 3.1 Process overview

This section provides a process overview how a TPP can gain access our production environment including portal and API gateway.

1. Get an approval from a local NCA (Finansinspektionen in Sweden) as an AISP or PISP. If you are a bank you already have an approval.
2. Obtain an eIDAS-certificate from a QTSP. QWAC is required.
3. Register in our portal and create apps and subscriptions. You will need the public key of your eIDAS QWAC certificate in this process
4. When you add a subscription to an app there will be a manual step for us to grant your app access. You will receive an e-mail when your app is granted access.
5. You can access the APIs (AIS and/or PIS) your certificate allows you to invoke

Business partners have a similar process which is not covered in this documentation.

#### 3.2 Create an account in the portal

Browse to the portal (<https://developer.skandia.se/open-banking/core-bank>). We recommend that you use Firefox as a browser when using the portal.

#### Step by step to register an account

Navigate to the portal.

Click the "CREATE ACCOUNT" button (top right corner). Enter the fields:

- user name (please make a note of the selected user name, it is required for login)
  - email address (will be used for communication with your organization)
  - first name
  - last name
  - phone (optional but recommended)
  - consumer organization (defines the name of the collective under which client applications will be created)
  - password and confirm password and captcha
1. Press the "Sign up" button
  2. Open the email received from the portal and follow the activation link
    - Following the activation link of the e-mail, the user account becomes enabled
  3. Login using the username and password selected by you

#### 3.3 How to setup your apps and subscriptions

##### Product

The API Products view provides an overview of the products available in the portal. A product acts as a collection of APIs. To enable invocation of an API, you must subscribe with an app to a product's plan containing the desired API.

##### API

An API is a set of functions and procedures allowing the creation of applications that access the features or data of an operating system, application, or other service. The API can be protected by various security mechanism, including, but not limited to, client id identification, MTLS and OAuth.

We expose a set of APIs which makes it possible for third parties to create applications using payment account information and payment initiation.

##### Plan

A plan represents a collection of APIs from a specific product. Subscribing to a plan allows the consuming app to invoke the APIs included in the plan given the correct security protocols are satisfied. A plan can contain terms of the usage of its APIs, including invocation rate limits and debit.

### Apps

Apps are the components that are used to invoke the APIs. An app has its own set of security credentials, client id and client secret. To create an app, navigate to the apps page and click the “Create new app” button.

To create an app, the following fields are required:

Field	Comment
Title	The name of the app (only visible to you and us)
QWAC Certificate	<p>Paste the content of the public part of your eIDAS QWAC x509 Certificate in PEM format with base 64 encoding. Please provide PEM encapsulation (BEGIN CERTIFICATE, END CERTIFICATE) on separate lines with the base 64 encoded certificate on a single line between the encapsulations. This certificate will be used for both MTLS and to determine your TPP accessibility.</p> <p>The certificate should contain the whole certificate chain.</p>
QSEAL Certificate (optional)	<p>Paste the content of the public part of your eIDAS QSEAL x509 Certificate in PEM format with base 64 encoding.</p> <p>We currently do not validate QSEAL but will do later, by entering this field the process will be easier for you when we require QSEAL.</p> <p>The certificate should contain the whole certificate chain.</p>
Application OAuth Redirect URL(s)	<p>A list of redirect urls to be used in the OAuth Authorization code flow. These addresses will be used to receive the authorization code required when claiming the access token for the API invocation.</p> <p>The url:</p> <ul style="list-style-type: none"> <li>• Must follow the format (?=\w+:\V?*\w+)</li> <li>• Cannot contain intermediate white spaces</li> </ul>
Sign Redirect Url (optional)	<p>The provided url will be used when redirecting the user after a successful payment. If no value is provided, the user will maintain at the default page after signing a payment.</p> <p>The url:</p> <ul style="list-style-type: none"> <li>• Must follow the format (?=\w+:\V?*\w+)</li> <li>• Cannot contain intermediate white spaces</li> </ul>

```
-----BEGIN-CERTIFICATE-----CRLF
MIIH0DCCBoigAwIBAgIKBllY7rSIPwDBDANBgkqhkiG9w.....CRLF
-----END-CERTIFICATE-----CRLF
```

**Above:** example of a correctly PEM-formatted public part of a QWAC Certificate

After creating an app, client id and client secret will be provided. Take note, this is the only time the client secret will be shown for you. It is possible to reset the client secret by navigating to apps, select an app, click subscriptions, select the menu present in the top right corner of the credentials window and click reset client secret.

It is possible to update an app. Updating an app requires the same input parameters as when creating it. However, it is not possible to change the QWAC Certificate provided when the app was initially created. I.e. if you desire to use another QWAC Certificate, a new app must be created, entailing a new client id and client secret. This is for example how you as a TPP handle updating a certificate about to expire.

#### Step by step to create an App

1. While logged in, click the “APPS” label from the top menu
2. Click the “Create new app” button

3. Enter mandatory fields title, QWAC-certificate<sup>1</sup>, application OAuth redirect URL(s) (minimum 1) and click the 'Submit' button
4. Take note of the API Key (Client-Id) and Secret and click the "Continue" button

#### Step by step to update an App

1. While logged in, click the "APPS" label from the top menu
2. Select and click the App to edit
3. Click the three vertical dots in the top right corner and select "Edit" from the drop-down menu
4. Change any of the available fields, including title, application OAuth Redirect URL(s) and Sign Redirect Url.
5. Click the 'Submit' button

#### Step by step to reset client secret

1. While logged in, click the "APPS" label from the top menu
2. Select and click the app to which you would like to reset client secret
3. Click the "Subscription" link located below the app name
4. Click the three vertical dots located in the top right corner of the credentials window and select "Reset Client Secret" from the drop-down menu
5. Click the "Reset" button
6. Check the 'Show' checkbox at to the top of the page
7. Take note of the new client secret

#### Step by step to delete an App

1. While logged in, click the "APPS" label from the top menu
2. Select and click the app to be deleted
3. Click the three vertical dots in the top right corner and select "Delete" from the drop-down menu
4. Click the "Delete" button

---

<sup>1</sup> The eIDAS-certificate holds authorization/the role you have a regulates if you can access AIS and/or PIS. If you as an ASIP creates an app/subscription for PIS the subscription will be denied.

## Subscription

A subscription defines a connection between a plan and an app. The plan defines the how and what of the subscription, including what APIs and how the APIs are accessible. The app defines the which of the subscriptions and hence details what credentials that are to be used against the gateway when attempting to invoke any of the APIs. Without a valid subscription, an app can never invoke a published API in the production environment.

To subscribe to a plan, go to the product overview page, select a product, click the plan you desire to subscribe to. Choose the App to which the subscription should be created for and confirm the subscription.

The act of requesting a new subscription will require a manual approval from us before the client app is enabled to invoke the API. It is possible to track the approval status by navigating to the app's subscriptions overview. If you know in advance that you need to get a fast approval, please contact us via mail and let us know a business day ahead.

### Step by step to create a subscription

1. While logged in, click "API PRODUCTS" from the top menu
2. Select and click the product that the subscription should regard
3. Select and click the 'Subscribe' button of the desired Plan
4. Select and click the 'Select App' button of the desired app
5. Overview the subscription details and press the "Next" button
6. Click the "Done" button

### Step by step to view Subscription status

1. While logged in, click the "APPS" label from the top menu
2. Select and click the app to which the subscription belongs to
3. Click the 'Subscription' link located below the app name
4. View the status of subscriptions in the subscriptions window

### Step by step to remove a Subscription

1. While logged in, click the "APPS" label from the top menu
2. Select and click the app to which the subscription belongs to
3. Click the "Subscription" link located below the app name
4. Click the three vertical dots located to the right of the specific subscription located in the Subscriptions window
5. Select "Unsubscribe" from the drop-down menu
6. Click the "Unsubscribe" button

#### 4 Security and authentication and of TPP including SCA of PSU

##### 4.1 OAuth2

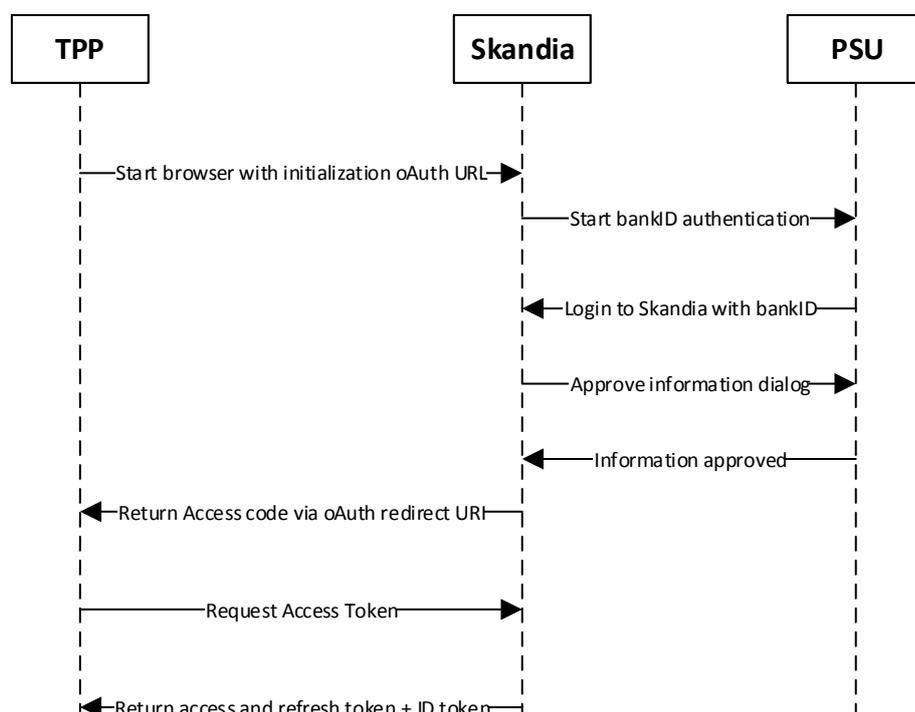
We use OAuth 2.0 for authentication and authorization. Our APIs are protected by the OAuth Authorization Code Grant type which is depicted in detail at <https://oauth.net/2/>.

##### 4.2 Request Access code and ultimately Access token

To enable the you access to the PSU:s accounts the PSU need to be authenticated by us. This is performed using a redirect approach. We are however investigating supporting other approaches such as decoupled and embedded. Swedish BankID is currently the only supported SCA.

##### 4.2.1 Authorization flow using redirect for authentication

This section provides information how to perform SCA with the redirect solution.



1. The authentication of the PSU starts upon request from you (TPP). You need to provide your ClientID, a redirect URI and requested scopes.
2. The PSU gets identified using Swedish BankID (\*note 1)
3. We display an information dialogue to the PSU. The PSU simply clicks OK. (\*note 2)
4. If BankID sign and PSU clicked OK in the information dialogue, we return an access code to the redirect URI provided by you
5. TPP can now request access token using the access code and client secret

##### Note 1:

If the PSU cancels/aborts the authentication with BankID or the authentication in BankID fails for some reason, we will show a default error page to the PSU. Currently we cannot redirect to an URI's given by you.

##### Note 2:

If the PSU choose to click cancel in the information dialogue Skandia will use the redirect URI given by the TPP with an added query parameter indicating an error.

Ex. <redirect\_uri>?error=access\_denied&description=null

#### **4.2.2 Token types and its uses**

To access the PSD2 APIs you need a valid access token. The access token is a short-lived token valid only for a short time. If the access token has expired, you use the refresh token to get a new access token.

The refresh token is valid for 90 days. After the refresh token has expired the PSU need to be authenticated again as previously described.

The ID token (of type Open ID-Connect) is a representation of the identity (ssn, "*personnummer*") of the authenticated PSU. This enables you as a TPP to validate that the PSU using your services corresponds to the PSU authenticated by us.

#### **4.2.3 AIS consent**

You as a TPP is responsible of having and handle the consent the customer gives you and act according to it. We do not

We will inform the PSU in the SCA process for the AIS the he/she will give you as TPP access to account information. The PSU simply clicks OK that he/she understands. You will after this have access to account information for 90 days without a renewed authentication by the PSU. You will have access to the accounts present at the time of the SCA and including new payment accounts the PSU creates.

## 5 General information about API invocation

### 5.1 Gateway

Invoking the PSD2 REST APIs requires 3 security measures, including:

1. Http header named "client-id" containing a client id of an app created in the portal. The app holds a valid subscription containing the specific API.
2. MTLS with the eIDAS QWAC certificate provided when creating the App
3. An Authorization header containing a bearer access token of type reference containing the scope demanded by the API definition

### 5.2 Headers

There are required headers which is found in the swagger file. Here are some notes about them.

Header	Comments
X-Request-ID	The unique ID to the call. TPP should generate this typically as UUID
X-Client-Certificate	The TLS certificate of your application, private key of the eIDAS-certificate.

## 6 AIS-operations

### 6.1 Introduction

The AIS API is available for you to get payment account information.

Pre-requisites:

1. Have registered a user account in the portal
2. Have an app setup in the portal with an eIDAS-certificate of QWAC-type telling us you are an AISP

Flow for first time access for a give customer:

1. The PSU agrees with you to use your services to retrieve account information from us
2. OAuth2/SCA – customer is informed about giving information to the you
  - a. The OAuth token will contain the ID of the customer
3. AISP invokes the AIS-operation of choice and receives a response with the requested information

Please view the swagger file for more information.

### 6.2 Overview of available accounts and customers

The AIS expose payment accounts owned by private consumers over the age of 18.

Summary:

- payments accounts owned by private consumer where the consumer is at least 18 years old
  - “Allt i Ett-konto”, typically used for everyday transactions and bill payments
  - “Sparkonto”, typically a savings account. Only domestic transfers possible.
- the consumer must own the account
  - We have no co-owned accounts
- payment accounts in SEK
- savings account in SEK (no fixed rate accounts are available)
  - This is also an “AiE-konto” but only domestic transfers are allowed (not bill payments etc)
- accounts in EUR

The AIS will allow a TPP to access all payments accounts for a given PSU. Only the account owner can give consent to an account. We do not offer shared ownership of accounts.

A customer AIS-consent will give a TPP access to the customer’s payment accounts (present and future created) for 90 days.

### 6.3 Read Account List

The operation will return a list of all payment accounts for a given PSU. Only the accounts which is owned by the PSU will be listed.

#### GET/v1/accounts

##### 6.3.1 Request description

Name	Type	Information
X-Request-ID	string	In: header Required: true Format: uuid Description: ID of the request, unique to the call, as determined by the initiating party.
PSU-Device-ID	String	In: header Required: recommended by Skandia Description: UUID for a device, which is used by the PSU
PSU-Geo-Location	String	In: header Required: recommended by Skandia Description: The forwarded Geo Location of the corresponding http request between PSU and TPP
X-Client-Certificate	String	In: header Required: true Description: The TLS certificate of your application

##### 6.3.2 Response description

Name	Type	Description
accountList	list	The list of accounts for the given customer
<b>Parameters of accountList</b>		
accounts		
resourceId	string	Id of an account (used to call other AIS operations)
bban	string	Bban identifier (clearing + number)
bic	string	BIC code of Skandiabanken: SKIASSESS
cashAccountType	string	Always "CACC"
currency	string	The currency code of the account ISO 4217
iban	string	Iban of an account
name	string	Our product name of the account type Example: <ul style="list-style-type: none"> <li>"Allt i Ett-konto" (transaction account)</li> <li>"Sparkonto" (savings account)</li> </ul>
usage	string	Always "PRIV" (private personal account, natural person)
_links		linked resources
<b>_links</b>		
balances	string	link to get balances
transactions	string	link to get transactions

### 6.3.3 Response example

**Body:**

```
{
  "accounts": [
    {
      "resourceId": "957054871102373",
      "bban": "91598570120",
      "bic": "SKIASESS",
      "cashAccountType": "CACC",
      "currency": "SEK",
      "iban": "SE079150000091598570120",
      "name": "Allt i Ett-konto",
      "usage": "PRIV",
      "_links": {
        "balances": {
          "href": "/v1/accounts/957054871102373/balances"
        },
        "transactions": {
          "href": "/v1/accounts/957054871102373/transactions"
        }
      }
    }
  ]
}
```

#### 6.4 Read Account Details

Read account details of an account.

GET /v1/accounts/{accountId}

##### 6.4.1 Request description

Name	Type	Information
X-Request-ID	string	In: header Required: true Format: uuid Description: ID of the request, unique to the call, as determined by the initiating party.
PSU-Device-ID	String	In: header Required: recommended by Skandia Description: UUID for a device, which is used by the PSU
PSU-Geo-Location	String	In: header Required: recommended by Skandia Description: The forwarded Geo Location of the corresponding http request between PSU and TPP
X-Client-Certificate	String	In: header Required: true Description: The TLS certificate of your application
accountId	String	In: path Required: true Description: This identification is denoting the addressed account. The account-id is retrieved by using a "Read Account List" call.

##### 6.4.2 Response description

Name	Type	Description
accountList	list	The list of accounts for the given customer
<b>Parameters of accountList</b>		
accounts		
resourceId	string	Id of an account (used to call other AIS operations)
bban	string	Bban identifier (clearing + number)
bic	string	BIC code of Skandiabanken: SKIASSESS
cashAccountType	string	Always "CACC"
currency	string	The currency code of the account ISO 4217
iban	string	Iban of an account
name	string	Our product name of the account type Example: <ul style="list-style-type: none"> <li>"Allt i Ett-konto" (transaction account)</li> <li>"Sparkonto" (savings account)</li> </ul>
usage	string	Always "PRIV" (private personal account, natural person)
_links		linked resources
<b>_links</b>		
balances	string	link to get balances
transactions	string	link to get transactions

### 6.4.3 Response example

**Body:**

```
{
  "resourceId": "957054871102373",
  "bban": "91598570120",
  "bic": "SKIASESS",
  "cashAccountType": "CACC",
  "currency": "SEK",
  "iban": "SE079150000091598570120",
  "name": "Allt i Ett-konto",
  "usage": "PRIV",
  "_links": {
    "balances": {
      "href": "/v1/accounts/957054871102373/balances"
    },
    "transactions": {
      "href": "/v1/accounts/957054871102373/transactions"
    }
  }
}
```

### 6.5 Read Balance List

The Operation will return the balances of an account.

**GET /v1/accounts/{accountId}/balances**

#### 6.5.1 Request description

Name	Type	Information
X-Request-ID	string	In: header Required: true Format: uuid Description: ID of the request, unique to the call, as determined by the initiating party.
PSU-Device-ID	String	In: header Required: recommended by Skandia Description: UUID for a device, which is used by the PSU
PSU-Geo-Location	String	In: header Required: recommended by Skandia Description: The forwarded Geo Location of the corresponding http request between PSU and TPP
X-Client-Certificate	String	In: header Required: true Description: The TLS certificate of your application
accountID	String	In: path Required: true Description: This identification is denoting the addressed account. The account-id is retrieved by using a "Read Account List" call.

#### 6.5.2 Response description

Name	Type	Description
<b>account</b>		
iban	String	Iban of an account
currency	String	the currency code of the account ISO 4217
<b>balances</b>		<b>list of balances</b>
balanceType "closingBooked"	List	booked transactions
<b>balanceAmount</b>		
amount		Balance amount of booked credit and debit transactions when the request was initiated
currency	String	The currency of the amount
<b>balances</b>		
balanceType "available"	List	booked transactions
<b>balanceAmount</b>		
amount		Balance amount of booked credit and debit transactions when the request was initiated. Reserved card authorizations are not included.
currency	String	The currency of the amount
referenceDate	Date	Now

### 6.5.3 Response example

**Body:**

```
{
  "account": {
    "iban": "SE0791500000091598570120",
    "currency": "SEK"
  },
  "balances": [
    {
      "balanceAmount": {
        "amount": "-1333.26",
        "currency": "SEK"
      },
      "balanceType": "closingBooked"
    },
    {
      "balanceAmount": {
        "amount": "8566.74",
        "currency": "SEK"
      },
      "balanceType": "available",
      "referenceDate": "2019-02-22T00:00:00"
    }
  ]
}
```

**6.6 Read transaction list of an account**

Read transaction reports or transaction lists of a given account addressed by "account-id", depending on the steering parameter "bookingStatus" together with balances. For a given account, additional parameters are e.g. the attributes "dateFrom" and "dateTo"

**GET /v1/accounts/{accountId}/transactions****6.6.1 Request description**

The below parameters are the only required parameters.

Name	Type	Information
X-Request-ID	string	In: header Required: true Format: uuid Description: ID of the request, unique to the call, as determined by the initiating party.
PSU-Device-ID	String	In: header Required: recommended by Skandia Description: UUID for a device, which is used by the PSU
PSU-Geo-Location	String	In: header Required: recommended by Skandia Description: The forwarded Geo Location of the corresponding http request between PSU and TPP
X-Client-Certificate	String	In: header Required: true Description: The TLS certificate of your application
accountID	String	In: path Required: true Description: This identification is denoting the addressed account. The account-id is retrieved by using a "Read Account List" call.
dateFrom	string (date-time)	In: query Required: false Description: Starting date (inclusive the date dateFrom) of the transaction list. For booked transactions, the relevant date is the booking date. For pending transactions, the relevant date is the entry date.
dateTo	string (date-time)	In: query Required: false Description: End date (inclusive the data dateTo) of the transaction list, default is "now" if not given.
bookingStatus	String	In: query Required: true Description: Permitted codes are * "booked", * "pending. To get both booked and pending transactions you will need to call the operation twice.

**6.6.2 Response description**

Information for each transaction will differ depending on payment type.

Name	Type	Description
transactions	list	list of transactions for the given account
<b>Account</b>		
iban	string	Iban of the account
currency	string	Currency code of the account
<b>Transactions</b>		
booked	list	List of booked transactions (based on input parameter)
pending	list	List of pending transactions (based on input parameter)
transactionsid	string	The Id of the payment initiation request
entryReference	string	A reference based on execution date and time
<b>transactionAmount</b>		
amount	string	
currency	string	
creditorAccount	list	
bban	string	Bban identifier (clearing + account number)
debtorName	string	
debtorAccount	string	
iban	string	Iban of an account
currency	string	
remittanceInformation Unstructured	string	message to a payee
<b>Parameters of pending and booked list</b>		
bookingDate	string	Date when entry was posted to the account
valueDate	string	Date when assets become available (credit)

### 6.6.3 Response example

**Body:**

```
{
  "account": {
    "iban": "SE0791500000091598570120",
    "currency": "SEK"
  },
  "transactions": {
    "booked": [
      {
        "transactionId": "957054871102373@YGCB0169@2019-01-02@2019-01-02-20.57.38.059727",
        "entryReference": "2019-01-02-20.57.38.059727",
        "bookingDate": "2019-01-02T00:00:00",
        "valueDate": "2019-01-01T00:00:00",
        "transactionAmount": {
          "amount": "-200",
          "currency": "SEK"
        },
        "creditorAccount": {
          "bban": "91598570120"
        },
        "debtorName": "Eva Testsson",
        "debtorAccount": {
          "iban": "SE0791500000091598570120",
          "currency": "SEK"
        },
        "remittanceInformationUnstructured": "Avgift"
      },
      {
        "transactionId": "957054871102373@YICB0149@2019-01-01@2019-01-02-15.03.22.103711",
        "entryReference": "2019-01-02-15.03.22.103711",
        "bookingDate": "2019-01-01T00:00:00",
        "valueDate": "2019-01-02T00:00:00",
        "transactionAmount": {
          "amount": "-842.89",
          "currency": "SEK"
        },
        "creditorAccount": {
          "bban": "91598570120"
        },
        "debtorName": "Eva Testsson",
        "debtorAccount": {
          "iban": "SE0791500000091598570120",
          "currency": "SEK"
        },
        "remittanceInformationUnstructured": "Kontraktsränta"
      }
    ],
    {
      "transactionId": "957054871102373@YICB0149@2018-12-31@2019-01-02-15.03.22.095046",
```

```
"entryReference": "2019-01-02-15.03.22.095046",
"bookingDate": "2018-12-31T00:00:00",
"valueDate": "2019-01-01T00:00:00",
"transactionAmount": {
  "amount": "-7.07",
  "currency": "SEK"
},
"creditorAccount": {
  "bban": "91598570120"
},
"debtorName": "Eva Testsson",
"debtorAccount": {
  "iban": "SE0791500000091598570120",
  "currency": "SEK"
},
"remittanceInformationUnstructured": "Ränta på skuld"
}
]
}
}
```

## **7 PIS-operations**

### **7.1 Introduction**

**PIS is currently not available. Documentation will be added later.**

The PIS API is available to you to initiate payments.

Pre-requisites:

1. Have registered a user account in the portal
2. Have an app setup in the portal with an eIDAS-certificate of QWAC-type telling us you are an PISP

Note: due to technical issues we are currently not able to expose operations for foreign payments, bill payments or future dated transfers. Currently we only support domestic transfers in SEK in the PIS API.

Please view the swagger file for more information.

### **7.2 Overview of possible payments to initiate**

Payments can be initiated for the same customers and accounts exposed in the AIS.

**8 Document revision history**

<b>Version</b>	<b>Description</b>
1.02	Updates for production portal registration and APIs
1.01	Updates for testing with mocked data
1.00	Initial version