Functional description Production

Versioning

Version	Release Day
V1.0	*June, 2019*
V1.1	*3rd July, 2019*
V1.2	*3rd July, 2019*
V1.3	*11th July, 2019*
V1.4	*12th November, 2019*
V1.5	*5th March, 2020*
V1.6	*7th May, 2020*
V1.7	*12th November, 2019*
V1.8	*23rd Sept, 2020*
V1.9	*14th Dec, 2020*
V2.0	*26th Aug, 2021*
V2.1	*18th March 2022*
V2.2	***17th August, 2022***

Note: Functionality described in grey text below is not yet available in Sandbox and Production APIs.

Published 2022-08-17

1 Authorization

During authorization, a scope is used to describe the type of access requested. The scopes allowed for a TPP are set by ICA Banken. The following scopes are available:

Scope	Description
account:	account is a scope to use for AISPs. It enables the authenticated user to fetch information via a TPP about the user's payment account/s and the transactions on the chosen account/s. https://ims.icagruppen.se/oauth/v2/authorize?response_type=code&client_id=CLIENT_ID&scope=account&redirect_uri=REDIRECT_URI&ssn=SSN
payment	payment is a scope to use for PISPs. It provides the ability for end users via a TPP to initiate transfers to other accounts and it indicates if funds are available. https://ims.icagruppen.se/oauth/v2/authorize?response_type=code&client_id=CLIENT_ID&scope=payment&redirect_uri=REDIRECT_URI&ssn=SSN
basket:<ID>	basket:<ID> is a scope to use for PISPs. This needs to be provided when approving a basket. Note, scope need to be provided with a prefix, **basket:<ID>** https://ims.icagruppen.se/oauth/v2/authorize?response_type=code&client_id=CLIENT_ID&scope=basket:<ID>&redirect_uri=REDIRECT_URI&ssn=SSN
payment_delete:<ID>	payment_delete:<ID> is a scope to use for PISP, needs to be provided when deleting a payment or a periodic mandate. Note, scope need to be provided with a prefix, **payment_delete:<ID>** https://ims.icagruppen.se/oauth/v2/authorize?response_type=code&client_id=CLIENT_ID&&scope=payment_delete:<ID>&redirect_uri=REDIRECT_URI&ssn=SSN
openid	Standard openid scope to get id token containing standard openid claims and consent signing method (ccr).

Note that the end user is required to sign the order with one of ICA Banken's valid authentication methods.

Valid authentication methods for ICA Banken are:
Mobile BankID
Personal device
Card and device

Please also note that same consent- and signing method need to be used during the initial signing of account and payment scope, as for the second signing where basket is to be signed. The used signing method can be found in id-token as claim ccr.

1.1 Register Certificate

The certificate is registered in ICA Banken API Store on the Application *only for the APIs using real customer data. A certificate is not necessary for using the Sandbox APIs.*

The valid certificate type is QWAC.

A QWAC certificate requested from a QTSP will contain a public part and a private part. The private key/part is generated on the TPP side when requesting a certificate from the QTSP, the private key will NEVER leave the TPP instead the certificate request will be sent to the QTSP that will check the validity to issue a digital certificate to the TPP and a public key/part will be issued. The public key is the part we request from ICA Banken to be uploaded in the ICA Banken API Store; ICA Banken do not require the private key as this is used as a proof of possession when using the certificate to establish a secure communication channel to ICA Banken.

ICA Banken require the TPP to possess a valid QWAC certificate to be able to establish a secure communication channel to ICA Banken. This is accomplished by leveraging Transport Layer Security (TLS) on the transport level, also called Mutual TLS, where ICA Banken will present its certificate and the TPP will be required to present its certificate. To establish a Mutual TLS session, the TPP establishing the session needs to configure the initiating client with both the public key and the private key.

ICA Banken require Mutual TLS sessions in the following steps:

- After submitting the public part of the QWAC certificate in the ICA Banken API Store we require the TPP to initiate a Mutual TLS call to our registration API to finish the registration of the TPP Application, to be able to consume PSD2 production APIs. In the guidance in the API Store step 3.1-3.10 we have exemplified this by using Postman to initiate this API call configuring the Postman client with the private and public key. If you don´t want to use Postman, please follow the steps for Registering the client using CURL.
- We also require Mutual TLS to be used in parts of the Authorization Code Grant flow and in those cases, shown below, we expect the TPP backend to be configured with the public/private key to initiate the Mutual TLS session.
    - When exchange a Code for an Access/Refresh token (/token endpoint)
    - When calling an ICA Banken PSD2 Production API

1.1.1 Registering the Client using CURL

The curl command can also be used for registering the client after register certificate in the developer store page. As described below

Prerequisites:

1. OS Windows 10
2. Curl compiled with OpenSSL. Download from here: https://curl.haxx.se/windows/

Execution

Extract the downloaded file and go to this location curl-7.65.1_3-win64-mingw/curl-7.65.1-win64-mingw/bin and open command prompt (cmd) at this location

```
curl  POST  --cert C:\xxxx\XXXX.crt.pem --key C:\ xxxx\XXXX.key.pem --pass XXXX -H "Content-Type: application/json" -d "{\"oid\":\"XXXX\", \"callbackUri\":[\"https://xxxx\", \"https://xxxx\"]}" https://mtls-apimgw-icabanken.ica.se/t/icabanken.tenant/mtls_client/2.0.0/client
```

Parameters:

- **cert** : The TPP client certificate to use for establishing the mTLS session in PEM format.

- **oid**: OID number obtain while Register Certificate in API store. It is a number from an FSA in any EU/EEA country and can be different formats. For example, PSDSE-FINA-xxxxx.
- **callbackUri** : https://xxxx - is the callback URL.
- **pass :** passphrase for the TPP private key if any.
- **key** : The TPP private key for the certificate to use for establishing the mTLS session in PEM format.
  Note: ICA Banken do not require the private key as this is used as a proof of possession when using the certificate to establish a secure communication channel to ICA Banken.

1.2 Access Token

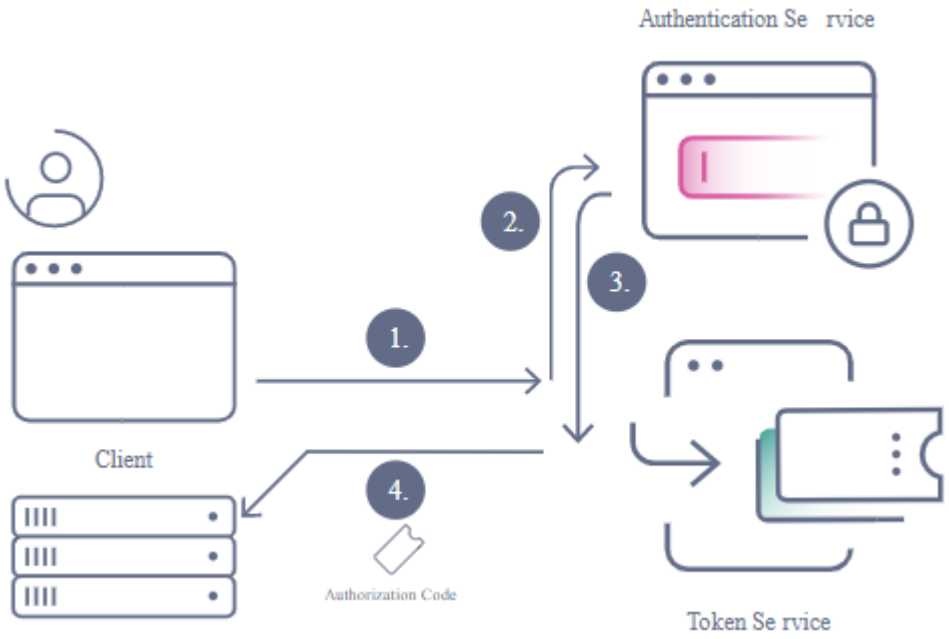
To get access to our APIs for ***using real customer data*** an access token is required. To get an access token for an end user, the user must both be authenticated with one of ICA Banken valid authentication methods and grant the calling client access to the requested scope. A test-access token is necessary in the Sandbox Environment.

ICA Banken is applying OAuth2 as a standard for authorization, for more detailed description about the OAuth2 standard we refer to RFC 6749 (https://oauth.net/2/)

1.2.1 OAuth Code Flow

The code flow is the most advanced flow in OAuth. It is also the most flexible, that allows both mobile and web clients to obtain tokens securely. It is split into two parts, the Authorization flow that runs in the browser where the client redirects to the OAuth server and the OAuth server redirects back when done, and the Token flow which is a back-channel call from the Client to the Token endpoint of the OAuth server.

1.2.1.1 Authorization Endpoint



1. Browser redirects to the Authorize endpoint of the OAuth Server
2. If the user isn't authenticated the OAuth Server redirect to the Authentication Service. Note that these two entities, while running in the same product, are separate conceptually.
3. The User authenticates, and is redirected back to the OAuth Server
4. The OAuth Server issues a one-time token called an Authorization Code

1.2.1.2 Token Endpoint

1. The Client backend makes a POST request to the Token endpoint with the Authorization Code and Client Credentials
2. The OAuth Server validates the code and the credentials and returns an access token and optionally a refresh token if configured on the client.

1.2.1.3 User Authentication

The user is authenticated during the Authorization part of the flow. This may involve multiple factors and is not defined by the OAuth specification. All user authentication is configured in the Authentication Service and is configured per client.

1.2.1.4 Client Authentication

The client authenticates in the Token part of the flow. Client authentication can be done in many ways, the most common being client secret. The following authentication mechanisms are supported is:

- Mutual TLS (mTLS) client certificate

1.2.1.5 The Authorization Code

Once the Authorization flow is done, the redirect back to the Client contains an Authorization Code. This is a Nonce, Not-more-than-once token, that is for single use. It has a short lifespan (usually less than 30 seconds) and must be presented in the token part of the flow.

1.2.1.6 The Access Token

The Access Token is returned by the token endpoint. It is the token that later can be used to call the API and gain access. It is a Bearer token and must not be sent to untrusted parties.
The access token has a lifetime of 10 minutes.

1.2.1.7 The Refresh Token

The Refresh Token is issued if the client is configured to have refresh tokens. This token can be used to obtain more access tokens once the first one expires. The refresh token may have a very long lifetime, ranging from hours to years. The refresh token has a lifetime of 90 days. Note, it can only be used to redeem another refresh-tone once. So, when you have used it once, you need to use a new refresh token which you get in response of first refresh-token request.

1.2.2 The Authorization Endpoint Request

The process is started by calling the authorize operation. The authorize operation should be called from a Web/APP GUI. It will not work using the URL in the backend-code. The reason for this is ICA Banken has three different authorization alternatives that need to be presented in the authorization flow. The call will return a pending authorization code and a redirect URL. The calling client (system user) is responsible for forwarding the end user to the redirect URL.

1.2.2.1 Request Parameters

If you wish to test an API call in production environment access below URL in web browser.

Initial consent signing:

Use this URL to initiate the signing for the person with SSN:

https://ims.icagruppen.se/oauth/v2/authorize?response_type=code&client_id=="OID"&scope=account&redirect_uri="REDIRECTURI"&ssn="SSN

As TPP you must change to your TPP-value in following fields:

• client ID
• redirect URI
• SSN

E.g. https://ims.icagruppen.se/oauth/v2/authorize?response_type=code&client_id=PSDSE-FINA-XXX&scope=account&redirect_uri=https://localhost:9443/oauth2&ssn=197202XXXXXX

- Method: **GET**
- Agent: Browser
- url: https://ims.icagruppen.se/oauth/v2/authorize

Response Type: Redirect to pre-registered callback at client with query parameters

Second signing:

Use this URL to initiate the signing for the person with previously issued access token:

https://ims.icagruppen.se/oauth/v2/authorize?response_type=code&client_id=="OID"&scope=account&redirect_uri="REDIRECTURI"&access_token="ACCESS_TOKEN"

As TPP you must change to your TPP-value in following fields:

• client ID
• redirect URI
• access_token

E.g. https://ims.icagruppen.se/oauth/v2/authorize?response_type=code&client_id=PSDSE-FINA-XXX&scope=account&redirect_uri=https://localhost:9443/oauth2&access_token=_0XBPWQQ_d83233c6-6316-43ce-b368-xxxxxxxxxxxx

- Method: **GET**
- Agent: Browser
- url: https://ims.icagruppen.se/oauth/v2/authorize
- Response Type: Redirect to pre-registered callback at client with query parameters

Parameter	Value	Mandatory	Description
client_id	The Organisation Identification Number	yes	The OID generated at the time of registration
response_type	code	yes	Defines the flow type: authorization code flow
state	a random value	no	Will be provided back to the client. Useful to keep track of the session in the client or to prevent unsolicited flows.
scope	Space separated string of scopes	yes	List the scopes the client is requesting access to. (account)
redirect_uri	The client callback URL	yes	The redirect_uri the client wants to redirect to. - Mandatory if multiple redirect URIs are configured on the client.

ssn	User social security number /personal number	Conditional*	Use ssn number register with ICA Banken Mandatory during initial signing of account and payment scope when access token is not available. Always mandatory when using the payment_delete scope.
ccr_values	icabanken-ob-bankid icabanken-ob-personligdosa icabanken-ob-kortdosa	no	Space separated string of available signing consent methods* Single value redirects to consent-screen, multiple values list provided consentors from values.
access_token	Previously issued token during signing of account and payment scope	conditional	Mandatory during second signing of payment basket when ssn is not requried

***Please note**: Same consent- and signing method need to be used during the initial signing of account and payment scope, as for the second signing where basket is to be signed. The used signing method can be found in id-token as claim ccr.

1.2.2.2 Response Parameters

In the response URL the authorization code is returned.

E.g.**https://localhost:9443/oauth2?code=qXUr1kCOF7ggtMn5cCuDQ7gD6xxxx**

A redirect back to the "redirect_uri". Response parameters are provided on the query string.

Parameter	Value	Mandatory	Description
state	The same value as given in the request	yes*	The same value as the client sent in the request. Use to match request to the redirect response. *Mandatory if the state was sent in the request
code	An Authorization Code	yes	An authorization code nonce, to be used in the token request.

ICA Banken

1.2.3 The Token Endpoint Request

1.2.3.1 Request Parameters

https://mtls-ims.icagruppen.se/oauth/v2/mtls-token

- Method: POST
- Content-Type: application/x-www-form-urlencoded
- URL: https://mtls-ims.icagruppen.se/oauth/v2/mtls-token
- Response Type: json

Parameter	Value	Mandatory	Description
client_id	The Organisation Identification Number	yes	The OID generated at the time of registration *for example, PSDE-SE-FINA-xxxxx*
mtls_certificate	Valid TPP production certificate	yes	This is TPP valid production certificate issue buy finance authority
grant_type	authorization_code	yes	Tells the token endpoint to do the second part of the code flow.
code	The authorization code	yes	The code given in the response of the Authorization request
redirect_uri	The callback URL of the Client	yes*	The same redirect URI as was sent in the authorize request. *Required if redirect_uri was sent in the authorize request.
code_verifier	The verifier that matches the code_challenge	no*	*Mandatory if code_challenge was used in the authorize request.

1.2.3.2 Response Parameters

Response Type: `application/json`

Parameter	Value	Mandatory	Description
access_token	A newly issued access token	yes	The resulting access token for the code flow
refresh_token	A newly issued refresh token	no	Only issued if the client is configured to receive refresh tokens
expires_in	Expiration in seconds	yes	The time to live of the access token in seconds
scope	Space separated string	no	If not present the requested scopes where issued. If present the issued scopes may differ from the requested scopes.
id_token	A newly issued id token	no	If open id scope was requested, id token will be issued.

Eg:

```
{
   "access_token": "cccda12a-fc90-4ce6-9e7d-xxxxxxxxx",
   "refresh_token": "4c615252-e2d4-4377-9a46-xxxxxxxx ",
   "scope": "account",
   "token_type": "bearer",
   "expires_in": 600
}
```

1.2.4 The Access Token from Refresh Token Endpoint Request

1.2.4.1 Request Parameters

https://mtls-ims.icagruppen.se/oauth/v2/mtls-token?grant_type=refresh_token&client_id=OID &refresh_token={ 1.2.3.2 Response parameter refresh_token}

- Method: POST
- Content-Type: application/x-www-form-urlencoded
- URL: https://mtls-ims.icagruppen.se/oauth/v2/mtls-token
- Response Type: json

Parameter	Value	Mandatory	Description
client_id	The Organisation Identification Number	Yes	The OID generated at the time of registration
grant_type	refresh_token	yes	Tells the token endpoint to do the refresh token call.
refresh_token	Refresh Token you received in 1.2.3.2 response	yes	Refresh token is valid for 90 days access token can be generated multiple times using this.
mtls_certificate	Valid TPP production certificate	yes	This is TPP valid production certificate issue buy finance authority

1.2.4.2 Response Parameters

Response Type: application/json

Parameter	Value	Mandatory	Description
access_token	A newly issued access token	yes	The resulting access token for the code flow
refresh_token	Return refresh token send in request	no	Only issued if the client is configured to receive refresh tokens
expires_in	Expiration in seconds	yes	The time to live of the access token in seconds
scope	Space separated string	no	If not present the requested scopes where issued. If present the issued scopes may differ from the requested scopes.

Eg:{


```
  "access_token": "432e484a-4110-462f-b586-xxxxxxxx",
  "refresh_token": "4c615252-e2d4-4377-9a46-xxxxxxxx",
  "scope": "account",
  "token_type": "bearer",
  "expires_in": 600
}
```

1.3 Access API:s

To get access to our APIs both an access token and scopes are required.
The issued access token should be sent in the **Authorization header in the format "Bearer [TOKEN]”**.
Scope is also required as a parameter in the API call to the authorize endpoint.
Requests outside the scope of those approved by the customer are unauthorized and will be rejected.

If the customer have outstanding KYC ("Know Your Customer") questions to answer, the operation will return an error with information. In this case the customer needs to log in at www.icabanken.se or log in to the ICA Banken mobile app and complete the KYC-steps. The following scopes are available:

Account and Payments

We are making production data for the Account API and Payments API available. Work continues to deliver the production data for Fundsconfirmation therefore, information related to this endpoint is described, but marked as grey parts in this documentation.

2 Account

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.

Accounts that are 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.

https://apim-icabanken.ica.se/store/apis/info?name=ICA.Bank.Services.PSD2.Accounts&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 service.

The type of cards available in ICA Banken and for which this applies to are ICA Bankort Maestro, ICA Bankkort, ICA Bankkort Plus. Note, account types called Direktkonto and Fakturakonto are both linked to ICA Bankkort Plus, but Fakturakonto is not the type of transaction account from which enables sepa-credit-transfers or cross-border-credit- transfers, due to the same functionality in our regular end user interface (www&app).

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.

GET /Accounts/{accountid} - Detailed account information, optionally with balance

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.

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.

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).

GET /Accounts/{accountId}/transactions-paginated – Paginated 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 are presented default if no dates specified. If no transaction within 30 days, please specify dates. Page size and page token can be used to paginate the responses. PageSize for responses per page, and PageToken for next page. As long as PageToken is returned a new page can be fetched with it. The page token is a string, maximum length 40 characters. The client may experience timeouts if the date span or page size is too large. Maximum page size is 1000 but recommended size is 100. In case of a timeout, please shorten the date span or page size. 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).

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.

Published 2022-08-17

3 Payments

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 (cross border payments).

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

The Payment API also include endpoint for Funds Confirmation.

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.

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.

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.

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.

Good-to know: An end-user/customer can delete a payment in a few cases, without involving an initiating TPP. This depending on which status the payments have. Cancellation of payments initiated via an TPP, can be done by end-user/customer via the telephone bank, mobile bank(app) or internet bank(www) and only applies to payments with “future date” or “recurring payments”, not transaction amount that is immediately reserved on the account. Customer can delete (Cancel) the payments which are in ENTERED (RCVD) or VERIFIED (ACTC) status.

When a payment is initiated via a TPP and stated by the customer to be ‘current date’ or without the specified payment date, (requestedExecutionDate), transaction amount is immediately reserved on the account.

Once the transaction amount has been reserved on the account, this cannot be canceled by the end-user/customer.

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

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

Description

Get status of a specific payment that was previously posted.

As there are different payment statuses according to Berlin Group (which you also find presented in "Enum" in API Console tab, on our API store page) below are the type of payments status that are included in our API solution:

- RCVD - Entered

- PDNG - Verified

- ACSP - Ongoing

- ACSC - Processed

- CANC - Cancelled

- RJCT - Cancelled/sys

Status= ACTC means that, the payment is correct and can be completed the current banking day, ie that the request is received before the cutoff time.

Status= ACWC means that, bank has changed the payment date to the next banking day.

VERIFIED status of payment is mapped to status ACTC
OPEN status of Recurring payment is also mapped to ACTC.

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.

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.

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

Published 2022-08-17

Description
Deletes a recurring payment mandate.

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

GET /PeriodicPayments/{paymentProduct}/{paymentId}/Status - Status of a specific payment

Description
Get status of a specific recurring payment mandate.

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 PUT /SigningBasket/{basketId} with a valid signed token (scope basket:<ID>) is required.

GET /SigningBasket /{baskedId} – Get a specific signing basket

Description
Get a specific signing basket that were previously posted.

PUT /SigningBasket/{baskedId} - Sign a specific signing basket.

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

4 HTTP error codes

API	HTTP Code	Title	Code	Detail
General	401	UnAuthorized	Not Authenticated	User is not Authenticated
	401	UnAuthorized	Consent_invalid	Consent is missing or invalid
	401	UnAuthorized	Consent_expired	Consent Expired (longer than 90 days)
	401	UnAuthorized	Invalid Client	Client_id not valid
	403	Resource Forbidden	RESOURCE_BLOCKED	Old KYC information
	500	Internal server Error	INTERNAL_SERVICE_ERROR	Cutomer Validation check failed
Accounts	400	Bad Request	FORMAT_ERROR	Can not retrieve personal number information
	400	Bad Request	RESOURCE_BLOCKED	Account is not valid
	400	Error received from underlying system service	SERVICE_INVALID	Error received from underlying system service
	404	Not found	RESOURCE_UNKNOWN	Accountinformation not found
	404	Not Found	RESOURCE_UNKNOWN	Account not found
	500	Error while calling underlying system service Account	INTERNAL_SERVICE_ERROR	Call to underlying system failed
Payments	400	Bad Request	FORMAT_ERROR	Payment information is null
	400	Bad Request	FORMAT_ERROR	Can not retrieve personal number or payment Id
	404	Not found	RESOURCE_UNKNOWN	Payment Information not found
	400	Error while calling underlying system service	FORMAT_ERROR	Call to underlying system failed
	404	Not found	RESOURCE_UNKNOWN	Call to underlying system failed
	500	Error while calling underlying system service	FORMAT_ERROR	Call to underlying system failed
	404	Resource not valid	Resource not valid	RESOURCE_UNKNOWN
	400	Could not create a siging basket	RESOURCE_UNKNOWN	Could not create a siging basket
	400	Bad Request	FORMAT_ERROR	Can not retrieve personal number or basket Id
	400	Bad Request	FORMAT_ERROR	Frequency should be monthly
	400	Error while calling underlying system service	FORMAT_ERROR	Call to underlying system failed
	404	Not found	RESOURCE_UNKNOWN	Call to underlying system failed
	404	Error response from underlying system	RESOURCE_UNKNOWN	Error response from underlying system

Registration/ onboarding	400	Bad request	ISSUER_MISSING	Unable to get issuer certificate
	400	Bad request	CERTIFICATE_EXPIRED	The certificate has expired.
	400	Bad request	ISSUER_MISSING	Unable to get local issuer certificate
	400	Bad request	CERTIFICATE_REVOKED	The certificate has been revoked by QSTP.
	400	Bad request	CERTIFICATE_MISSING	Certificate was not available in the request but is mandated for the corresponding
	400	Bad request	CERTIFICATE_BLOCKED	Certificate has been blocked by the ASPSP
	400	Bad request	FAILED	The Organization ID is not provided correctly in the request. Please include it in the request.
	400	Bad request	FAILED	The callback URI List is not provided correctly in the request. Please include it in the request.
	400	Bad request	FAILED	The callback URI List is empty, please provide valid URI List
	400	Bad request	FAILED	The callback URI List consists invalid URI, please provide valid URI List
	400	Bad request	FAILED	The Organization ID does not exist.
	400	Bad request	FAILED	Client: <<oid>> could not be registered in Curity. Contact the Support Team.
	400	Bad request	FAILED	Client: <<oid>> could not be registered. Contact the Support Team.
	400	Bad request	FAILED	OID : <<oid>> is already registered
	400	Bad request	FAILED	The Organization ID does not exist.
	400	Bad request	FAILED	The Organization ID does not match
	400	Bad request	FAILED	The uploaded certificate does not match
	400	Bad request	FAILED	Error occurred while registering client. Contact the Support Team
	400	Bad request	FAILED	Internal application error, Contact the Support Team.