PIS API

PSD2 interface PIS de Volksbank

April 2019

Colophon

Label	Data
Owner	Service Centre KBS de Volksbank N.V.
Authors	ITC VO KWB Open Banking
Status	PIS BG final
Project	PSD2

Version

Version	Date	Changes
1.0	2019-04-04	Final version

References

Version	Date	Description	Author	Reference
	October 2012	The OAuth 2.0 Authorization Framework	D. Hardt, Ed.	RFC 6749
		OAuth 2.0 Servers	Aaron Parecki	
	2014-07-21	An Introduction to OAuth 2	Mitchell Anicas	
	2015-07-03-07	OAuth 2.0 Token Introspection	J. Richer, Ed.	RFC 7662
1.1	2009-12-18	Sepa Requirements For An Extended Character Set	European Payments Council (EPC)	EPC217-08

TABLE OF CONTENTS

1 Introduction

This document describes the PIS (Payment Initiation Service) interface offered by de Volksbank under PSD2. It explains the process of the consent a PSU (Payment Service User) must give to allow a TPP (Third Party Provider), in its role of PISP (Payment Initiation Service Provider), to submit a payment debiting the PSU's account.
It should be noted that this interface:

- complies with Berlin Group standards (NextGenPSD2 XS2A Framework Implementation Guidelines V1.3);
- supports the initiation of a single SEPA Credit Transfer (SCT).

The remainder of this document will be organized as follows:

- Chapter 2 describes the conditions de Volksbank applies to the use of its payment initiation services, the character set used for the payment information to be exchanged between the PISP and de Volksbank in its role of ASPSP, the datatypes defined for the individual pieces of information and the URLs to be used by the PISPs for the different brands of de Volksbank.
- Chapter 3 sheds some light on the requirements PISPs must meet to access the systems controlled by de Volksbank.
- Chapter 4 not only lays out the fine details of the Berlin Group payment initiation flow, but also describes some payment initiation services specific to de Volksbank.

2 Payment Initiation Services offered by de Volksbank

2.1 Conditions on the use of de Volksbank's payment initiation services

De Volksbank offers 3 payment services:

1. One-time direct payments. This payment service is referred to as *payments* by the Berlin Group (POST /v1/payments/{payment-product});
2. Deferred payments. In contrast to the Berlin Group requirements, the scheduling of deferred payments lies with the PISPs. With respect to the data structure and most of the process steps, the deferred payment of de Volksbank complies with the Berlin Group standard.
3. Recurring payments. In contrast to the Berlin Group requirements, the scheduling of recurring payments lies with the PISPs. With respect to the data structure and most of the process steps, the recurring payment of de Volksbank complies with the Berlin Group standard.

The following conditions apply on the usage of all of these payment initiation services:

1. The authorization code is valid for a duration of 10 minutes;
2. The access token is valid for a duration of 10 minutes;
3. The refresh token is valid for 90 days.

These services also have their own specific requirements which must be met by the PISP. They are listed below per specific payment service:

One-time direct payments

1. A one-time direct payment cannot be cancelled by neither the PISP nor the PSU;
2. A one-time direct payment never has an *endDate* in the request body;
3. A one-time direct payment cannot be re-submitted by the PISP with the same paymentId, even if the payment request cannot be processed by the ASPSP for technical reasons or because of insufficient balance.

Deferred payments

1. The execution date for a deferred payment as recorded in the attribute *endDate* cannot be after 13 months counted from and including the month where the payment request was received by the ASPSP and replied to with the status *RCVD* (RCVD means *received*);
2. The PISP (not the ASPSP) is responsible for the submission of a deferred payment for execution;
3. The PSU (customer) can withdraw the permission for the execution of a deferred payment up to and including the date as recorded in the attribute *endDate* in the original payment request;
4. Withdrawal of the permission by the PSU can only be done in the online banking environment of the ASPSP;

5. The permission to execute a deferred payment expires automatically after the date as recorded in the attribute *endDate*;
6. The PISP can offer a deferred payment for execution before the date as recorded in the *endDate* in the original payment request;
7. A deferred payment can only be submitted once by the PISP with the same *paymentId*, even if the payment request cannot be processed by the ASPSP for technical reasons or because of insufficient balance.

Recurring payments

1. A recurring payment can be delivered with the attribute *endDate* filled with a date or without the attribute *endDate*. In the latter case we are dealing with an *infinite* or *perpetual* recurring payment;
2. In a series of recurring payments, the PISP (not the ASPSP) is responsible for submitting every individual payment for execution by the ASPSP;
3. A PISP can only submit one recurring payment for execution by the ASPSP per week, provided that the execution of the payment is successful;
4. If submission or execution of an individual payment in a series of recurring payments fails, the PISP is allowed to re-submit the payment for a period of 7 calendar days with a maximum of one attempt per calendar day;
5. The PSU is entitled to withdraw the permission for a series of recurring payments up to and including the date as recorded in the attribute *endDate* delivered in the original payment request;
6. The PSU is entitled to withdraw the permission for a series of recurring payments lacking an endDate at any moment;
7. Withdrawal of a permission can only be done in the online banking environment of the ASPSP;
8. The permission for the execution of a series of recurring payments expires automatically on the date as recorded in the attribute *endDate* delivered in the original payment request;
9. A PSU is allowed to view individual payments in a series of recurring payments, even if the permission has been withdrawn.

2.2 Character set

The used character set is the Latin character set of the UTF-8 character encoding standard. This is in accordance with the character set as defined by the European Payments Council (EPC) Implementation Guidelines (EPC217-08). This character set is defined below:

a b c d e f g h i j k l m n o p q r s t u v w x y z
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
0 1 2 3 4 5 6 7 8 9
/ - ? : ( ) . , ' +
Space

2.3 Data types

The APIs as defined by de Volksbank consume and produce JSON (Java Script Object Notation) structures. JSON accepts the following data types:

1. A string;
2. A number;
3. An object (JSON object);
4. An array;
5. A boolean.

2.4 URLs

De Volksbank supports PSD2 APIs for three different brands: ASN Bank, RegioBank and SNS. There is one specific URL per brand.

- URL to start the PSU's SCA and approval process:
  - for TPPs in the role of PISP to start the approval process for the PSU, use :
    - **psd.bancairediensten.nl/psd2/asnbank/v1/authorize**
    - **psd.bancairediensten.nl/psd2/regiobank/v1/authorize**
    - **psd.bancairediensten.nl/psd2/snsbank/v1/authorize**

  - for TPPs in the role of PISP to redeem a one-off authorization code or a recurring refresh token for an access token, use:
    - **psd.bancairediensten.nl/psd2/asnbank/v1/token**
    - **psd.bancairediensten.nl/psd2/regiobank/v1/token**
    - **psd.bancairediensten.nl/psd2/snsbank/v1/token**

- URL for executing permission, the so-called bank-URL:
  - for ASN Bank, use: **api.asnbank.nl**
  - for RegioBank, use: **api.regiobank.nl**
  - for SNS, use: **api.snsbank.nl**

With respect to the data types, de Volksbank adheres closely to the datatypes and formats used in pain messages as defined by the ISO 20022 norm and adopted by the EPC for SEPA payments. This means that for alpha-numerical, decimal and date fields the datatype **string** with some additional formatting will be used:

Datatype	Length/Format	Description
String	Maxtext34	Maximum length of the alpha-numerical string is 34
	Maxtext35	Maximum length of the alpha-numerical string is 35
	Maxtext70	Maximum length of the alpha-numerical string is 70
	Maxtext140	Maximum length of the alpha-numerical string is 140
	ISO 8601 date format	Dates are of the data type string, but must comply with the ISO 8601 date format. This implies that dates have the following format: **YYYY-MM-DD**.
	ISO 8601 datetime format	Dates are of the data type string, but must comply with the ISO 8601 datetime format.
	Decimal format	Amount fields are of the data type *string*, but have the format of a *decimal* where the following format requirements hold: 1. The number of fractional digits must comply with the ISO 4217 minor unit of currency (for instance, the number of fractional digits for the currency EUR is 2); 2. The digits denoting integers and the digits denoting fractions are separated by a **dot**.
Number	Integer format	Number is an integer starting at 0, 1, 2, ...

3 Access

The PISP can only use the PSD2 APIs as authorized by de Volksbank. The PISP must be registered with the Competent Authority with a license to perform payment initiation services (refer to payment service 7 as described in Annex of the Payment Services Directive (2015/2366),
PISPs that wish to use the PSD2 APIs of de Volksbank are required to go through an onboarding process. Part of this onboarding process is the exchange of a so-called **client-id**, **client-secret** and **redirect_uri.** The redirect_uri is needed to return the response to the payment initiation request, the subsequent authorization request and token exchange request to the appropriate address of the PISP.

3.1 Certificates

The connections between the TPP and de Volksbank endpoints are secured by a mutual TLS authentication, as required by the PSD2 regulations. This means that the TLS connection can only be established including client (i.e. TPP) authentication. For this authentication the TPP has to use a qualified certificate for website authentication. This qualified certificate has to be issued by a qualified trusted service provider (QTSP) according to the eIDAS regulation [eIDAS].
The content of the certificate has to be compliant with the requirements as specified in article 34 of the EBA Regulatory Technical Standards on Strong Customer Authentication and common and secure communication under article 98 of Directive 2015/2366 (PSD2).

The public key of this certificate has to be presented to de Volksbank during the onboarding process of the TPP.

3.2 Authentication by OAuth2

De Volksbank has chosen the OAuth2 authentication method for its PSD2 interface, an authentication method that does not require users to share their bank passwords with third-party apps. More details on the OAuth2 authentication method can be found in the standard OAuth2 flows or in one of the many tutorials on the internet.

3.3 Authorization

De Volksbank is using the so-called *Authorization Code* grant flow. The authorization code grant type is used to obtain both access tokens and refresh tokens and is optimized for confidential clients.

The ASPSP (the PSU's bank) delivers an authorization code to the TPP on behalf of the customer. The code is issued only once by the ASPSP and is needed for using the PSD2 functions. Next, the TPP will exchange the authorization code for an access and refresh token. The access token can subsequently be used in each PSD2 API service, but only once.

4 The APIs for submitting a payment request on behalf of a PSU

The PISPs must use the following APIs for initiating and executing a payment request :

1. Payment initiation request with JSON encoding (JSON means Java Script Object Notation);
2. Authorization request;
3. Access token request: access token and refresh token based on an authorization code;
4. New access token request: new access and refresh tokens based on a refresh token;
5. Payment execution request with JSON for deferred and recurring payments;

The API endpoints usually consist of the following elements:

1. Method and URL;
2. Path parameters;
3. Query parameters;
4. Request header;
5. Request body;
6. Response code;
7. Response header;
8. Response body.

For every individual endpoint de Volksbank offers, we will point out which of these elements they have and explain them in depth.

4.1 Payment initiation request: PISP requesting permission to submit a payment on behalf of a PSU

By issuing a payment initiation request, the PISP seeks permission from an ASPSP to submit a payment debiting the account a PSU is holding with the addressed ASPSP on behalf of that PSU.

In the sub-sections to come, we will discuss at length the parts which make up the payment initiation endpoint.

4.1.1 Method and URL

Method	URL	Description
POST	https://psd.bancairediensten.nl/psd2/[snsbank|asnbank|regiobank]/v1/payments/{payment-product}	Payment initiation endpoint for one **time direct payments** as defined by the Berlin Group in the implementation guide version 1.3.
POST	https://psd.bancairediensten.nl/psd2/[snsbank|asnbank|regiobank]/v1/deferred-payments/{payment-product}	Volksbank-specific payment initiation endpoint for **deferred payments** with a make-up conform to the structure as laid down by the Berlin Group in the implementation guide version 1.3.
POST	https://psd.bancairediensten.nl/psd2/[snsbank|asnbank|regiobank]//v1/recurring-payments/{payment-product}	Volksbank-specific payment initiation endpoint for **recurring payments** with a make-up conform to the structure as laid down by the Berlin Group in the implementation guide version 1.3.

4.1.2 Path parameters

Attribute	Type	Mandatory	Description
payment-product	String	Y	The attribute refers to the payment product associated with the credit transfer payment method. The Berlin Group distinguishes the following payment products: 1. sepa-credit-transfers; 2. instant-sepa-credit-transfers; 3. target-2-payments; 4. cross-border-credit-transfers. It is up to the ASPSP to decide which of these payment products it supports. At the moment, de Volksbank only supports the following product: 1. sepa-credit-transfers.[1]

4.1.3 Query parameters

The payment initiation endpoint does not have any query parameters.

[1] De Volksbank processes sepa-credit-transfers instantly, provided that the bank of the creditor is reachable for instant payments. So, there is no difference in the settlement of these payments with the processing via our PSU interfaces.

4.1.4 Request header

Attribute	Type	Mandatory	Description
Content-Type	String	Y	Attribute invariably filled with the value "*application/json*".
X-Request-ID	String	Y	Attribute filled with the id of the request, unique to the call, as determined by the initiating party (the PISP).
Authorization	String	Y	Attribute consists of *client_id* and *client_secret* separated by a colon (**:**) in a **base64** encoded string. ▪ Format: Basic base64 (<client_id>:<client_secret>); ▪ client_id: Identification of the PISP as registered with de Volksbank; ▪ client_secret: secret agreed between the PISP and de Volksbank.
PSU-IP-Address	String	Y	Attribute filled with the IP-address of the PSU as recorded in the HTTP request from the PSU to the PISP. If the PSU has not sent its IP-address to the PISP, the PISP has to send its own IP-address.
TTP_redirect_uri	url	Y	Attribute filled with the URI of the PISP, where the transaction flow is redirected to. Mandatory for the implicit SCA method (including OAuth2) used by de Volksbank.

4.1.5 Request body

Attribute	Type	Mandatory	Description
endToEndIdentification	String	N	Attribute filled with the unique identification of the payment request as provided by the PISP. Max35Text.
debtorAccount iban currency	Account Reference Object String String	N	iban: Attribute *iban* is part of the object *Account Reference* as defined by the Berlin Group. ISO 20022 pattern: [A-Z]{2,2}[0-9]{2,2}[a-zA-Z0-9]{1,30}. currency: Attribute *currency* is part of the object *Account Reference* as defined by the Berlin Group. ISO 4217 Alpha 3 currency code.

Attribute	Type	Mandatory	Description
instructedAmount currency amount	Amount Object String String	Y	currency: Attribute *currency* is part of the object *Amount* as defined by the Berlin Group. ISO 4217 Alpha 3 currency code. amount: Attribute *amount* is part of the object *Amount* as defined by the Berlin Group. The amount is given with fractional digits, if needed. The decimal separator is a dot (.). The number of fractional digits (or minor unit of currency) must comply with ISO 4217. totalDigits 18 fractionDigits 5.
creditorAccount iban currency	Account Reference Object String String	Y	iban: ISO 20022 pattern: [A-Z]{2,2}[0-9]{2,2}[a-zA-Z0-9]{1,30}. currency: ISO 4217 Alpha 3 currency code.
creditorAgent	String	N	Attribute is filled with a BIC. ISO 20022 definition BIC: [A-Z]{6,6}[A-Z2-9][A-NP-Z0-9]([A-Z0-9]{3,3}){0,1}.
creditorName	String	Y	Party to which an amount of money is due. Max70Text.
ultimateCreditor	String	N	Ultimate party to which an amount of money is due. Max70Text. This attribute is optional. Nevertheless it is highly recommended to provide this information in case the TPP is acting as Collecting Service Provider. The TPP is temporarily in the possession of the collected funds (after the initiated payment is executed and settled) and transfers the collected funds from his “escrow” creditor account to the ultimate receiver/creditor account.
ultimateCreditorId	String	N	The attribute *ultimateCreditorId* is de Volksbank-specific attribute *ultimate_receiver_id*. The attribute *ultimateCreditorId* is not on the list of attributes as defined by the Berlin Group. Max35Text. This attribute is optional. Nevertheless it is highly recommended to provide this information in case the TPP is acting as Collecting Service Provider.
remittanceInformationUnstructured	String	N	Max140Text.

Attribute	Type	Mandatory	Description
remittanceInformationStructured	String	N	Remittance information according to the list of Currence (“CUR”) or ISO-20022 (“ISO”). Max35Text.
issuerSRI	String	N	The attribute *issuerSRI* is a Volksbank-specific attribute required whenever the attribute *remittanceInformationStructured* is used. The attribute *issuerSRI* is not on the list of attributes as defined by the Berlin Group. It can, for instance, have the following values: • CUR; • ISO. Max35Text.
endDate	String	N	The attribute *endDate* is not allowed with payments of the payment service *one time direct* (called *payments* by the Berlin Group) The attribute *endDate* is mandatory for payments of the payment service *deferred payments*. The *endDate* marks the ultimate date on which the PISP can submit a payment for execution by the ASPSP. The attribute *endDate* is optional for payments of the payment service *recurring payments*, because de Volksbank also allows for recurring payments with no end date, the so-called infinite or perpetual recurring payments. If the *endDate* is filled, it is the last date where the PISP can submit a payment in a series of payments for execution by the ASPSP. Attribute *endDate* has the ISO 8601 Date format (YYYY-MM-DD).

4.1.6 Examples payment initiation request

The **payment initiation request** described in the previous sub-sections is illustrated below. We give two examples: one with a filled attribute *remittanceInformation**Structured*** and one with a filled attribute *remittanceInformation**Unstructured***. Both attributes are mutually exclusive in accordance with the EPC rule stating that “*Either ‘Structured’ or ‘Unstructured’ may be present*”

```
POST https://psd.bancairediensten.nl/psd2/snsbank/v1/deferred-payments/sepa-credit-transfers
Content-Type: application/json
```

```
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Authorization: TTP-X:something secret
PSU-IP-Address: 192.168.8.78
TPP-Redirect-URI: https://merchant.com/0123456789"
{
   "endToEndIdentification": "ID234567",
   "debtorAccount": {"iban": "NL64MAART0948305290", "currency": "EUR"},
   "instructedAmount": {"currency": "EUR", "amount": "123.50"},
   "creditorAccount": {"iban": "NL55WIND0000012345", "currency": "EUR"},
   "creditorAgent": "WINDNL2A",
   "creditorName": "Adyen",
   "ultimateCreditor": "Krentebol dot com",
   "ultimateCreditorId": "1234",
   "remittanceInformationStructured": "1234 5678 9012 3456",
   "issuerSRI": "CUR",
   "endDate": "2099-01-01"
 }


POST https://psd.bancairediensten.nl/psd2/snsbank/v1/deferred-
payments/sepa-credit-transfers
Content-Type: application/json
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Authorization: TTP-X:something secret
PSU-IP-Address: 192.168.8.78
TPP-Redirect-URI: https://merchant.com/0123456789"
{
   "endToEndIdentification": "ID234567",
   "debtorAccount": {"iban": "NL64MAART0948305290", "currency": "EUR"},
   "instructedAmount": {"currency": "EUR", "amount": "123.50"},
   "creditorAccount": {"iban": "NL55WIND0000012345", "currency": "EUR"},
   "creditorAgent": "WINDNL2A",
   "creditorName": "Adyen",
   "ultimateCreditor": "Krentebol dot com",
   "ultimateCreditorId": "1234",
   "remittanceInformationUnstructured": "payment for 11 currant buns",
   "endDate": "2099-01-01"
 }
```

4.1.7 Response code

Code	Description
201	Created POST response code where Payment Initiation was correctly performed.

4.1.8 Response header

Attribute	Type	Mandatory	Description
Content-Type	String	Y	Attribute invariably filled with the value "*application/json*".
Location	String	Y	Attribute contains the location of the created resource.
X-Request-ID	String	Y	Attribute filled with the id of the request, unique to the call, as determined by the initiating party (the PISP).
ASPSP-SCA-Approach	String	Y	The attribute ASPSP-SCA-Approach is invariably filled with the value REDIRECT.

4.1.9 Response body

Attribute	Type	Mandatory	Description
transactionStatus	String	Y	Value of the attribute is conform with the ISO 20022 **ExternalPaymentTransactionStatus1Code** list. Enumeration: RCVD (*RCVD* means received).
paymentId	String	Y	Max16Text. **N.B**.: ▪ relationship paymentId - one time direct payment is 1:1; ▪ relationship paymentId - deferred payment is 1:1; ▪ relationship paymentId – recurring payment is 1:n. This means that the paymentId cannot be used as correlation id for individual transactions in a series of payments of the type recurring-payments.
_links	Links	Y	**Remark:** All links can be relative or full links. The choice to be made is up to the discretion of the ASPSP. **"scaOAuth":** In case of a SCA OAuth2 Approach, the ASPSP is transmitting the URI where the configuration of the Authorisation Server can be retrieved. The configuration follows the OAuth 2.0 Authorisation Server Metadata specification. "**status**": the link to retrieve the transaction status of the payment initiation.

4.1.10 Example(s) payment initiation response

The response of the service **payment initiation request** is illustrated below:

```
HTTP/1.x 201 Created
Content-Type:       application/json
Location:           v1/consents/6ba7b811-9dad-11d1-80b4-00c04fd430c8
X-Request-ID:       99391c7e-ad88-49ec-a2ad-99ddcb1f7756
ASPSP-SCA-Approach: REDIRECT
{
   "transactionStatus": "RCVD",
   "paymentId": "REB0000123456789",
   "_links": {
      "scaOAuth": {"href": "https://www.devolksbank.com/authorize"},
      "status": {"href": "/v1/payments/REB0000123456789/status"}
   }
}
```

4.2 Authorize request: PSU is requested to approve the execution of the payment

The PISP issues a request with the purpose to receive a URL which re-directs the PSU to the local bank environment in order to allow the PSU to authorize its bank, the ASPSP, to execute the payment submitted by the PISP.

In the next sub-sections, we will take a closer look at the elements which constitute the authorization endpoint.

4.2.1 Method and URL

Method	URL	Description
GET	https://psd.bancairediensten.nl/psd2/ [snsbank|asnbank|regiobank]/v1/authorize?	Authorization endpoint as defined by de Volksbank

4.2.2 Path parameters

The authorization endpoint does not have any path parameters.

4.2.3 Query parameters

Attribute	Type	Mandatory	Description
response_type	String	Y	Attribute invariably filled with the value "*code*".
paymentId	String	Y	Attribute filled with the value of the *paymenId* as received in the response body of the payment initiation request.
scope	String	Y	Attribute specifies the level of access that the application is requesting. Invariably filled with the value "*PIS*".
state	String	Y	Attribute contains the unique identification of the request issued by the PISP. The Berlin Group calls this attribute *X-Request-ID*.
redirect_uri	url	Y	Attribute filled with the value where the service redirects the user-agent to after granting the authorization code. No wildcards can be used in the call-back URL. De Volksbank validates the exact call-back URL.
client_id	String	Y	Attribute filled with the value of the client_id

4.2.4 Request header

The authorization endpoint does not have a request header.

4.2.5 Request body

The authorization endpoint does not have a request body.

4.2.6 Example authorize request

The authorize request is illustrated below:

```
GET
https://psd.bancairediensten.nl/psd2/snsbank/v1/authorize?response_type=code&payment_id=SNS012345678912&scope=PIS&state=111111&redirect_uri=https://thirdparty.com/callback&client_id=<client_id>
```

4.2.7 Response code

Code	Description
302	Redirect

4.2.8 Response header

Attribute	Type	Mandatory	Description
location	String	Y	This attribute contains: 1. The URL leading to the login page of the ASPSP; 2. Session data stored in a JWT object (JWT stands for *Json WebToken*).

4.2.9 Response body

The authorize endpoint does not have a response body.

4.2.10 Example authorize response

The authorize response is illustrated below:

```
HTTP/1.x 302
location:
https://api.snsbank.nl/online/toestemminggeven/#/login?action=display&sessionID=<sessionID>&sessionData=<sessionData>
```

4.3 PSU approving the payment request

PSUs clicking on the link leading them to the ASPSP will log on to the service to authenticate their identity. Next, the PSU approves the PISP's request to execute the payment. In case of success, the service returns an authorization code and redirects the user-agent to the application defined by the redirect URI.

The PSU's authentication and the PSU's approval are processes internal to de Volksbank, which we will not describe here. The return of the authorization code, though, that we will discuss below.

4.3.1 Response code

Code	Description
302	Redirect

4.3.2 Response parameters

Attribute	Type	Mandatory	Description
code	String	Y	Attribute filled with the authorization code needed to obtain an access and a refresh token. This code can only be used once and exchanged within a configurable time window (currently set to 10 minutes).
state	String	Y	This attribute is filled with the value which the PISP has delivered in the attribute **state** in the **Authorize** request

The authorization code is then passed on to the PISP via the re-direct URL the PSU has to its disposition.

4.3.3 Example authorization response

The authorization response is illustrated below:

```
HTTP/1.x 302
https://fintechapplication/redirect?code=869af7df-4ea4-46cf-8bed-3de27624b29e&state=12345
```

4.4 Access token request: PISP requesting an access token

The access token and the refresh token are provided on the basis of the authorization code. The PISP requests an access token from the API by passing the authorization code along with authentication details, including the client secret, to the API token endpoint.

4.4.1 Method and URL

Method	URL	Description
POST	https://psd.bancairediensten.nl/psd2/[snsbank|asnbank|regiobank]/v1/token?	Authorization endpoint as defined by de Volksbank

4.4.2 Path parameters

The token endpoint does not have any path parameters.

4.4.3 Query parameters

Attribute	Type	Mandatory	Description
grant_type	String	Y	Filled with the fixed value "*authorization_code*"; defines the OAuth2 flow.
code	String	Y	Authorization code needed to obtain an access and a refresh token.
redirect_uri	String	Y	The service redirects the user-agent to the application redirect URI. No wildcards can be used in the call back URL. De Volksbank validates the exact call back URL.

4.4.4 Request header

Attribute	Type	Mandatory	Description
Content-Type	String	Y	filled with the value "*application/x-www-form-urlencoded*".
Authorization	String	Y	Consist of *client_id* and *client_secret* separated by a colon (**:**) in a **base64** encoded string. – Format: Basic base64 (<client_id>:<client_secret>); – client_id: Identification of the PISP as registered with de Volksbank; – client_secret: secret agreed between the PISP and de Volksbank.

4.4.5 Request body

The token endpoint does not have a request body.

4.4.6 Example token request

The token request is illustrated below:

```
POST
https://psd.bancairediensten.nl/psd2/snsbank/v1/token?grant_type=authorization_code&code=AUTORIZATION_CODE&redirect_uri=https://thirdparty.com/callback

Content-Type: application/x-www-form-urlencoded

Authorization: Basic base64(<client_id>:<client_secret>)
```

4.4.7 Response code

If the authorization is valid, the ASPSP will return a response containing the access token (and optionally, a refresh token) to the application. The response will look like this:

Code	Description
200	Ok

4.4.8 Response header

Attribute	Type	Mandatory	Description
Content-Type	String	Y	Attribute is invariably filled with the value "*application/json*".

4.4.9 Response body

Attribute	Type	Mandatory	Description
access_token	String	Y	Attribute filled with the access token needed to call the PSD2 interface, in this case PIS.
token_type	String	Y	Attribute filled with the fixed value "*bearer*".
expires_in	Number	Y	Attribute filled with the lifetime in seconds of the access token.
refresh_token	String	Y	Value in the attribute can be used to obtain a new access token using the same authorization grant in the situation where the current token has expired.
scope	String	Y	Attribute filled with the scope of the access token. In this context *PIS*.

4.4.10 Example token response

The token response is illustrated below:

```
HTTP/1.x 200
Content-Type: application/json
   {
     "access_token": "<ACCESS_TOKEN>",
     "token_type": "bearer",
     "expires_in": 600,
     "refresh_token": "<REFRESH_TOKEN>",
     "scope": "PIS"
   }
```

At this point, the PISP has been authorized. It is allowed to use the token to retrieve the status of the payment via the service until the token expires or is revoked. A refresh token may be used to request new access tokens, if the original token has expired.

4.5 New access token request: PISP requesting a new access token

When the original token has expired, the PISP can request a new access token. A PISP using an expired token in a payment status information request will receive an "Invalid Token Error" response. When this happens, the refresh token can be used to request a fresh access token from the authorization server. The authorization server issues a new refresh token, in which case the client must dispose of the old refresh token and replace it with the new refresh token.

4.5.1 Method and URL

See 4.4.1.

4.5.2 Path parameters

The token endpoint does not have any path parameters.

4.5.3 Query parameters

Attribute	Type	Mandatory	Description
grant_type	String	Y	Attribute filled with filled with the fixed value "*refresh_token*"; defines the OAuth2 flow.
code	String	Y	Refresh token code needed to obtain the new access and refresh token.
redirect_uri	String	Y	The service redirects the user-agent to the application redirect URI. No wildcards can be used in the call back URL. De Volksbank validates the exact call back URL.

4.5.4 Request header

See 4.4.4.

4.5.5 Request body

The token endpoint does not have a request body.

4.5.6 Example token request

The token request is illustrated below:

```
POST
https://psd.bancairediensten.nl/psd2/snsbank/v1/token?grant_type=
refresh_token&code=REFRESH_TOKEN&redirect_uri=https://thirdparty.com/call
back
Content-Type: application/x-www-form-urlencoded
Authorization: Basic base64(<client_id>:<client_secret>)
```

4.5.7 Response code

If the authorization is valid, the ASPSP will return a response containing the access token (and optionally, a refresh token) to the application. The response will look like this:

Code	Description
200	Ok

4.5.8 Response header

Attribute	Type	Mandatory	Description
Content-Type	String	Y	Attribute is invariably filled with the value "*application/json*".

4.5.9 Response body

Attribute	Type	Mandatory	Description
access_token	String	Y	Attribute filled with the access token needed to call PSD2 interface, in this case PIS.
token_type	String	Y	Attribute filled with the fixed value "*bearer*".
expires_in	Number	Y	Attribute filled with the lifetime in seconds of the access token.
refresh_token	String	Y	Value of the attribute can be used to obtain a new access token using the same authorization grant in the situation where the current token has expired.
scope	String	Y	Attribute filled the scope of the access token. In this context *PIS*.

4.5.10 Example token response

The token response is illustrated below:

```
HTTP/1.x 200
Content-Type: application/json
   {
     "access_token": "<ACCESS_TOKEN>",
     "token_type": "bearer",
     "expires_in": 600,
     "refresh_token": "<REFRESH_TOKEN>",
     "scope": "PIS"
   }
```

Now, the PISP has been authorized again.

4.6 Payment execution request

The approval of payments of the type *deferred payments* and *recurring payments* and the subsequent execution of these payments is a disjunct process in the sense that the execution is done in a separate service call. By issuing a payment execution request, the PISP explicitly requests the ASPSP to process the submitted credit transfer payment for which the PSU has given approval.

In the sub-sections to come, we will discuss at length the parts which make up the payment initiation endpoint.

4.6.1 Method and URL

Method	URL	Description
POST	https://psd.bancairediensten.nl/psd2/ [snsbank|asnbank|regiobank]/v1/ {payment-service}/{payment-product}/{paymentId}	Payment execution endpoint for de Volksbank specific payment services **deferred payments** and **recurring payments**.

4.6.2 Path parameters

Attribute	Type	Mandatory	Description
payment-service	String	Y	Attribute refers to the type of payment service. For this particular endpoint de Volksbank only supports the proprietary payments services *deferred payments* and *recurring payments*. Therefore, the enumeration is: 1. deferred-payments; 2. recurring-payments.
payment-product	String	Y	The attribute refers to the payment product associated with the credit transfer payment method. The Berlin Group distinguishes the following payment products: 1. sepa-credit-transfers; 2. instant-sepa-credit-transfers; 3. target-2-payments; 4. cross-border-credit-transfers. It is up the ASPSP to indicate which of these payment products it supports. At the moment, de Volksbank only supports the following product: 1. sepa-credit-transfers.[2]
paymentId	String	Y	Attribute hosts the unique identification assigned by the ASPSP to the payment, when the initiation request was sent in by the PISP.

4.6.3 Query parameters

The payment execution request endpoint does not have any query parameters.

4.6.4 Request header

Attribute	Type	Mandatory	Description
Content-Type	String	Y	Attribute invariably filled with the value "*application/json*".
X-Request-ID	String	Y	Attribute filled with the id of the request, unique to the call, as determined by the initiating party (the PISP).
Authorization	String	Y	Attribute contains the access token acquired by the PISP as a result of calling the token endpoint.

[2] De Volksbank processes sepa-credit-transfers instantly, provided that the bank of the creditor is reachable for instant payments. So, there is no difference in the settlement of these payments with the processing via our PSU interfaces.

4.6.5 Request body

Attribute	Type	Mandatory	Description
endToEndIdentification	String	N	Unique identification as provided by the PISP. Max35Text.
remittanceInformationUnstructured	String	N	Max140Text.
remittanceInformationStructured	String	N	Max35Text.
issuerSRI	String	N	The attribute issuerSRI is a Volksbank-specific attribute required whenever the attribute remittanceInformationStructured is used. The attribute issuerSRI is not on the list of attributes as defined by the Berlin Group. Max35Text.

4.6.6 Example(s) payment execution request

The **payment execution request** described in the previous sub-sections is illustrated below. We give two examples: one with a filled attribute *remittanceInformation**Structured*** and one with a filled attribute *remittanceInformation**Unstructured***. Both attributes are mutually exclusive in accordance with the EPC rule stating that "*Either 'Structured' or 'Unstructured' may be present"*

```
POST https://psd.bancairediensten.nl/psd2/snsbank/v1/recurring-
payments/sepa-credit-transfers/REB0000123456789
Content-Type: application/json
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Authorization: xyz
{
   "endToEndIdentification": "ID234567",
   "remittance Information Structured": "1234 5678 9012 3456",
   "issuerSRI": "CUR"
 }

POST https://psd.bancairediensten.nl/psd2/snsbank/v1/recurring-
payments/sepa-credit-transfers/REB0000123456789
Content-Type: application/json
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Authorization: xyz
{
   "endToEndIdentification": "ID234567",
   "remittanceInformationUnstructured": "payment for oodles of buns"
}
```

4.6.7 Response code

Code	Description
201	Created POST response code where Payment Initiation was correctly performed.

4.6.8 Response header

Attribute	Type	Mandatory	Description
Content-Type	String	Y	Invariably filled with the value "application/json".
X-Request-ID	String	Y	Attribute filled with the id of the request, unique to the call, as determined by the initiating party (the PISP).

4.6.9 Response body

Attribute	Type	Mandatory	Description
transactionStatus	String	Y	Value of the attribute is conform with the ISO 20022 **ExternalPaymentTransactionStatus1Code** list.
paymentId	String	Y	Max16Text. **N.B**.: ▪ relationship paymentId - one time direct payment is 1:1; ▪ relationship paymentId - deferred payment is 1:1; ▪ relationship paymentId – recurring payment is 1:n. This means that the payment Id cannot be used as correlation id for individual transactions in a series of payments of the type recurring-payments.
resourceId	String	Y	Unique identification as assigned by the ASPSP to uniquely identify the payment execution resource.
endToEndIdentification	String	N	Het datatype en opmaak van het attribuut is conform de ISO 20022 definitie: Max35Text.

4.6.10 Example payment execution response

The response of the service **payment execution request** is illustrated below:

```
HTTP/1.x 201 Created
Content-Type:       application/json,
X-Request-ID:       99391c7e-ad88-49ec-a2ad-99ddcb1f7756
{
   "transactionStatus": "ACCC",
   "paymentId": "REB0000123456789",
   "resourceId": "XYZ",
   "endToEndIdentification": "ID234567"
}
```

4.7 HTTP codes for failure (error handling)

Code	Description
400	Bad request The server cannot or will not process the request due to something that is perceived to be a client error (e.g., malformed request syntax, invalid request message framing, or deceptive request routing).
401	Unauthorized The request has not been applied because it lacks valid authentication credentials for the target resource.
403	Forbidden The server understood the request but refuses to authorize it.
404	Not found The origin server did not find a current representation for the target resource or is not willing to disclose that one exists.
500	Internal server error The server encountered an unexpected condition that prevented it from fulfilling the request.