PSD2

This document enters into force on the date specified in the Directive (EU) 2015/2366 of the European
Parliament and of the Council on payment services in the internal market.

Table of Contents

Changelog

Version	Release date	Changes
02.08	2021-09-17	- Added endpoint for batch payments: /api/v1/payments/standard/isoBatch - /api/v1/payments/submission endpoint was modified to also submit batch payments -web submission via “orderId=” was modified to also submit batch payments
02.09	2021-09-30	-Endpoints /iso and /isoBatch modified, XML tag <Iban> changed to <IBAN> to better comply with ISO 20022, pain.001.001.03
02.10	2022-04-14	-Added separate endpoint for getting status of batch payments: /api/v1/payments/<batchOrderId>/statusBatch
02.11	2022-06-10	-Added the “Differences between Live and Sandbox environments” section - Added the “Support and troubleshooting” section
02.12	2022-08-19	-Restored section “Payment submission via Prima bank web page”
02.13	2022-10-20	-Added the option to initiate SEPA payment without explicitly stating debtor IBAN, the PSU will be able to choose it on our web -Specifics added to section “PISP Operation: Standard payment initialization (XML) - SEPA payments”, page 33.
02.14	2022-12-08	-Added status “IBAN” to /status endpoint, page 41
02.15	2023-06-23	-Removal of outdated external web links
02.16	2023-08-01	-Added “orderId” to /token endpoint page 47

Terminology

For the purposes of this document, the following terms have the following meanings:

Term	Meaning
AISP	Account Information Service Provider.
Alternative implementation	The ASPSP is required to implement at least one of the alternatives.
ASPSP	Account Servicing Payment Service Provider.
Authentication	TPP Identity confirmation.
Authorization	Verification of access to ASPSP resources.
Certificate	Qualified certificate in the sense of e-IDAS.
Directive	PSD2 Directive. Directive of the European Parliament and of the Council (EU) 2015/2366.
EV	Extended Validation certificate
IBAN	International Bank Account Number.
JOSE	JSON Object Signing and Encryption.
OIDC	OpenID Connect
Optional implementation	The ASPSP may implement this functionality or process.
Optional input parameter	TPP can ignore this parameter.
Optional output Parameter	The ASPSP may fill the parameter value.
PIISP (CBII,CISP)	Payment Instrument Issuer Service Provider (Card Based Payment Instrument Issuer or Card Issuer Service Prvider)
PISP	Payment Initiation Service Provider.
PSU	Payment Service User.
Resource	All access points of the ASPSP API for TPP access within PSD2.
RTS	Regulatory technical standards of the Europan Banking Authority
SBA	Slovak banking association.
SCA	Strong Customer Authentication. Authentication of a payment service user means authentication based on the use of two or more elements that are categorized as knowledge (something the user knows only), ownership (something that only the user has), and inherence (something, the user is) and are independent in the sense that the violation of one element does not impair the reliability of the other elements, while being created in such a way as to protect the confidentiality of the authentication data.
The Slovak Banking API Standard	Common initiative of Slovak banking association and its members. The aim of this initiative is to develop common specifications for the communication interface between ASPSPs and third party providers within the meaning of Directive (EU) 2015/2366.
TPP	Third Party Provider, i.e., a third party that is a payment service provider providing payment service users with a payment initiation or account information service or a payment service provider issuing card based payment facilities.

Introduction

This document defines secure communication between the TPP and the ASPSP and between the PSU and the ASPSP, in particular to ensure the integrity of the transmitted data and the identity of the communicating entities. The SCA process drawn in the process flow of the individual processes diagrams serves for demonstration purposes and a better understanding of process flow. List of services described by the standard:

Service Provider	Service	Description
AISP	Accounts information	Account information – service provide information and balances related to an account
AISP	Accounts transactions	Account transactions – service provide list of transactions in defined date range related to an account to which the client has given a mandate to specific TPP (not a list of all client accounts) without balances
PISP	Standard payment initialization (XML)	Standard payment initialization – service allows to initialize payment in XML format (PAIN.001)
PISP	Standard payment submission	Standard payment submission – service allows to authorization of initialized payment
PISP	Payment order status	Payment order status – service provides actual information about initialized payment
PISP	Request to cancel payment	Request to cancel payment – service allows to cancel payment, that were initiated through the same PISPby services Standard Payment Initializaton (XML) or Standard Payment Initializaton (JSON)
PISP	Standard payment initialization (JSON)	Standard payment initialization – service allows to initialize payment in JSON sufficient balance with the yes/no answer
PISP	Balance check	Balance check – service provide information about sufficient balance with the yes/no answer

Requesting access to the API

Access to the API is granted to TPPs by communication keys. Following are the steps a TPP should take to be granted communication keys:

1. Fill in the registration form.
2. Send the filled in above-mentioned form to api.banking@primabanka.sk.
3. Send a qualified certificate for website autenthication, containing a license number matching your national authority registered license number, to api.banking@primabanka.sk in a .zip compressed format. The certificate has to be signed by an accredited certification authority.
4. Once the provided information is verified, Primabanka will generate the appropriate communication keys. The keys will be sent to the TPP via the email addressed provided in the registration form. The keys will be sent in the form of a client_id and a client secret, encrypted by the provided certificate.
5. After successful registration and receiving the keys, the TPP will be able to test and use the API using the provided links.

Once a TPP is satisfied with their implementation in the test environment and wants access to the production environment, they should contact us at api.banking@primabanka.sk and send a certificate similarily to the above point 3. They will be provided with communication keys to the production API.

Once a registration form and the required certificate are received by Primabanka, we will respond with the communication keys within 14 days.

Support and troubleshooting

Support is available at api.banking@primabanka.sk. Our banking API team should respond within seven days of an inquiry, with troubleshooting time depending on the issue.

Securing communication

A TLS version 1.2 is used to secure the communication layer. In order to reduce the vulnerability of block ciphers, only AEAD (Authenticated Encryption with Additional Data) is allowed, specifically:
AES_GCM (128,256)

NIST	OpenSSL equivalent
TLS_RSA_WITH_AES_128_GCM_SHA256	AES128-GCM-SHA256
TLS-RSA-WITH-AES-256-GCM-SHA384	AES256-GCM-SHA384
TLS-RSA-WITH-CAMELLIA-128-GCM-SHA256	NA
TLS-RSA-WITH-CAMELLIA-256-GCM-SHA384	NA
TLS-DHE-RSA-WITH-AES-128-GCM-SHA256	DHE-RSA-AES128-GCM-SHA256
TLS-DHE-RSA-WITH-AES-256-GCM-SHA384	DHE-RSA-AES256-GCM-SHA384
TLS-DHE-RSA-WITH-CAMELLIA-128-GCM-SHA256	NA
TLS-DHE-RSA-WITH-CAMELLIA-256-GCM-SHA256	NA
TLS-ECDHE-RSA-WITH-AES-128-GCM-SHA256	ECDHE-RSA-AES128-GCM-SHA256
TLS-ECDHE-RSA-WITH-AES-256-GCM-SHA384	ECDHE-RSA-AES256-GCM-SHA384

TLS-ECDH-RSA-WITH-AES-128-GCM-SHA256	ECDH-RSA-AES128-GCM-SHA256
TLS-ECDH-RSA-WITH-AES-256-GCM-SHA384	ECDH-RSA-AES256-GCM-SHA384
TLS-ECDH-RSA-WITH-CAMELLIA-128-GCM-SHA256	NA
TLS-ECDH-RSA-WITH-CAMELLIA-256-GCM-SHA384	NA
TLS-ECDHE-ECDSA-WITH-AES-128-GCM-SHA256	ECDHE-ECDSA-AES128-GCM-SHA256
TLS-ECDHE-ECDSA-WITH-AES-256-GCM-SHA384	ECDHE-ECDSA-AES256-GCM-SHA384
TLS-ECDHE-ECDSA-WITH-CAMELLIA-128-GCM-SHA256	NA
TLS-ECDHE-ECDSA-WITH-CAMELLIA-256-GCM-SHA384	NA
TLS-ECDH-ECDSA-WITH-AES-128-GCM-SHA256	ECDH-ECDSA-AES128-GCM-SHA256
TLS-ECDH-ECDSA-WITH-AES-256-GCM-SHA384	ECDH-ECDSA-AES256-GCM-SHA384
TLS-ECDH-ECDSA-WITH-CAMELLIA-128-GCM-SHA256	NA
TLS-ECDH-ECDSA-WITH-CAMELLIA-256-GCM-SHA384	NA

AES_CCM (128,256)

NIST	OpenSSL equivalent
TLS-RSA-WITH-AES-128-CCM	AES128-CCM
TLS-RSA-WITH-AES-256-CCM	AES256-CCM
TLS-RSA-WITH-AES-128-CCM-8	AES128-CCM8
TLS-RSA-WITH-AES-256-CCM-8	AES256-CCM8
TLS-DHE-RSA-WITH-AES-128-CCM	DHE-RSA-AES128-CCM
TLS-DHE-RSA-WITH-AES-256-CCM	DHE-RSA-AES256-CCM
TLS-DHE-RSA-WITH-AES-128-CCM-8	DHE-RSA-AES128-CCM8
TLS-DHE-RSA-WITH-AES-256-CCM-8	DHE-RSA-AES256-CCM8
TLS-ECDHE-ECDSA-WITH-AES-128-CCM	ECDHE-ECDSA-AES128-CCM
TLS-ECDHE-ECDSA-WITH-AES-256-CCM	ECDHE-ECDSA-AES256-CCM
TLS-ECDHE-ECDSA-WITH-AES-128-CCM-8	ECDHE-ECDSA-AES128-CCM8
TLS-ECDHE-ECDSA-WITH-AES-256-CCM-8	ECDHE-ECDSA-AES256-CCM8

CHACHA20_POLY1305

NIST	OpenSSL equivalent
TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	ECDHE-RSA-CHACHA20-POLY1305
TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256	ECDHE-ECDSA-CHACHA20-POLY1305
TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256	DHE-RSA-CHACHA20-POLY1305

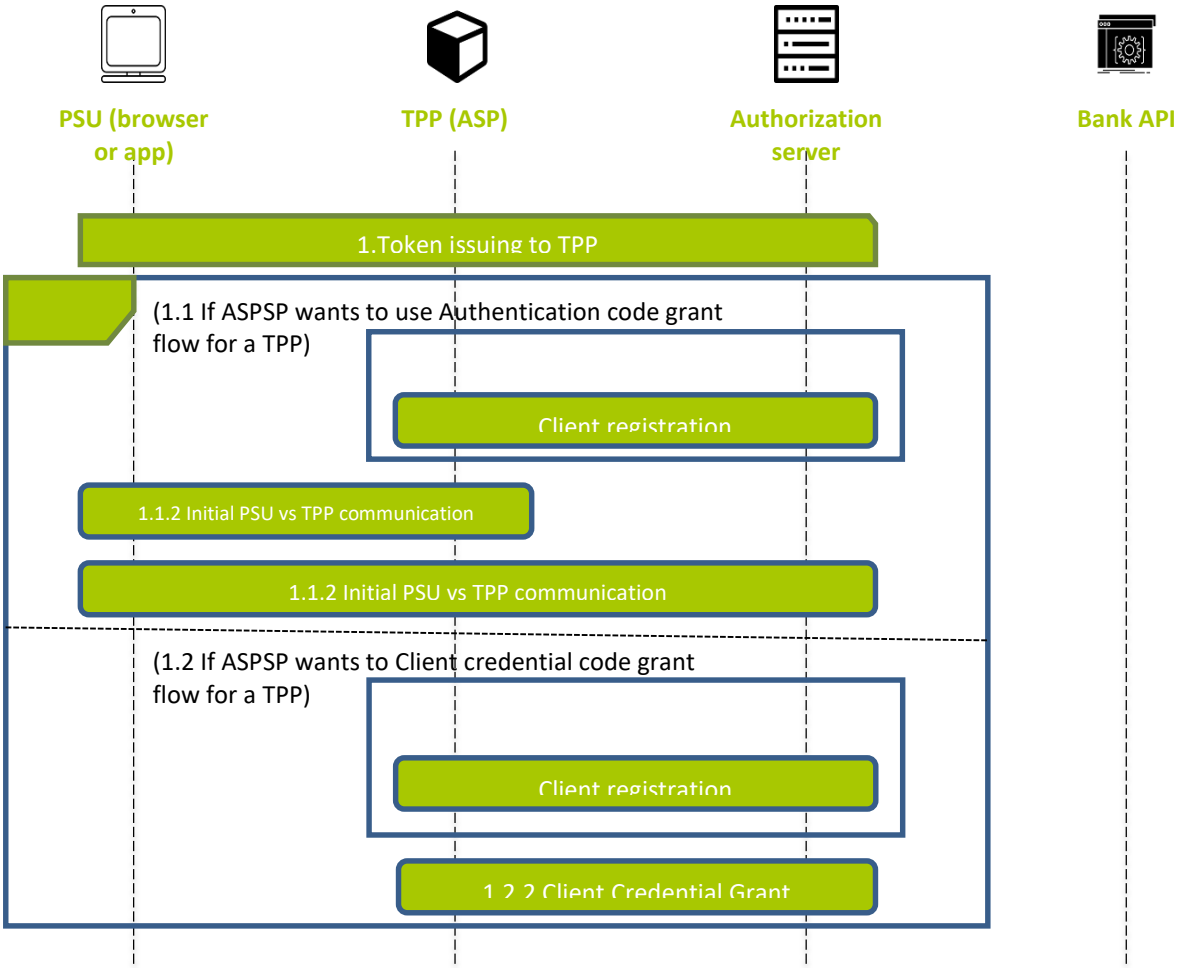
Differences between Live and Sandbox environments

Differences between Live and Sandbox environments are outlined in the following points:

- Default accepted SMS code in sandbox is “0”. This mitigates the need of the TPPs to register a mobile phone number with Prima Bank for development/testing.

TPP and ASPSP authentication

For the authentication of the ASPSP as a resource provider, the eIDAS-based site authentication certificate is used. For the authentication of the TPP as a client, the eIDAS-based site authentication certificate is used as well. The certificate used must be issued in accordance with ETSI TS 119 495 (Qualified Certificate Profiles and TSP Policy Requirements under the payment services Directive (EU) 2015/2366). All TPP requests, where technically possible, must be protected by TLS protocol with mutual authentication where PKI certificates are used.



General Design approach

After successful TPP PSD2 registration, the TPP is given by license number and PKI certificate which contains the **license number**. Since this point, the TPP is allowed to perform both communication with an ASPSP, use development portal and access the **API documentation**.
ASPSP will manage communication with TPP using the IETF RFC 6749 - The OAuth 2.0 Authorization Framework („OAuth framework“, hereafter only). Therefore, in order to access ASPSP's API, TPP must be given by an **access token** which must be presented when performing a call. The access token usage is defined by the ITEF RFC 6750 - The OAuth 2.0 Authorization Framework: Bearer Token Usage („**access_token“**, hereafter only). All TPP requests, where technically possible, must be protected by TLS protocol with mutual authentication, where PKI certificates used are in accordance to definitions in the Section TPP and ASPSP authentication.

Section Assigning a technical identifier.
The technical identifier consists of **client_id** and **client_secret** and is used for automated communication with the ASPSP to obtain valid **access_token** and **refresh_token**. Assigning a technical identifier is not required. In the absence of a technical identifier, only the client credentials grant method with a valid PKI certificate can be used.

TPP requests

Upon TPP being issued a valid access token, it may call only the services defined by Prima banka standard and made available by a particular ASPSP. If the access token becomes invalid, the TPP must perform actions to obtain new access token.

Assigning a technical identifier

The technical identifier consists of client_id and client_secret and is used for automated communication with the ASPSP to obtain valid access_token and refresh_token. Assigning a technical identifier could be required by ASPSP. In the absence of a technical identifier, only the client credentials grant method with a valid PKI certificate can be used.

Account Servicing Payment Service Provider (AISP)

Chapter defines list of methods and alternative of flows provided for AISPs.
Prerequisites:

a) The TPP is registered for the AISP role and valid AISP scope
b) The TPP has been successfully checked and authenticated
c) The TPP has presented its "OAuth2 Authorization Code Grant" access token which allows the ASPSP to identify the relevant PSU

Endpoints definition

Following sections describes technical definition of provided endpoints for AISPs.

Account information – service provide information and balances related to an account

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/api/v1/accounts/information
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/api/v1/accounts/information

Account transactions – service provide list of transactions in defined date range related to an account

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/api/v1/accounts/transactions
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/api/v1/accounts/transactions

Standard header definition

Set of request and response headers for AISP endpoints

Request header definition

Attribute	Optionality	Type	Description
Host	Mandatory	String	Domain name of the server and optional TCP port number
Content-Type	Mandatory	String	application/json or application/xml
Authorization	Mandatory	String	Authorization is defined in RFC 6750 - The OAuth 2.0 Authorization Framework: Bearer Token Usage
Request-ID	Mandatory	String	An unique identifier of a particular request message. Although it may be arbitrary string, it is strongly (UUID) version 4 form (RFC4122).
Correlation-ID	Optional	String	An unique correlation identifier correlates the request and the response messages as a pair especially useful for audit logs. Although it may be arbitrary string, it is Ifentifier (UUID) version 4 form (RFC4122).
Process-ID	Optional	String	Identifier of a business or technical process to what the set of requests and response pairs are organized (e.g. paging of transaction history should have same Process-ID). Although it may be arbitrary string, it is strongly (UUID) version 4 form (RFC4122).
PSU–IP-Address	Mandatory	String	Identifier of a customer's IP address from which he/she is connected to the TPP infrastructure. It might be in the format of IPv4 o IPv6 address. ASPSP shall indicate which values are acceptable.
PSU-Device-OS	Mandatory	String	A customer's device and/or operating system identification from which he/she is connected to the TPP

			infrastructure.
PSU-User-Agent	Mandatory	String	A customer's web browser of other client device identification from which he/she is connected to the TPP infrastructure. Agent header field of the http request between PSU and TPP.)

Response header definition

Attribute	Optionality	Type	Description
Content-Type	Mandatory	String	application/json or application/xml
Response-ID	Mandatory	String	An unique identifier of a particular request message. Although it may be arbitrary string, it is strongly (UUID) version 4 form (RFC4122).
Correlation-ID	Optional	String	An unique correlation identifier correlates the request and the response messages as a pair especially useful for audit logs. Although it may be arbitrary string, it is Ifentifier (UUID) version 4 form (RFC4122).
Process-ID	Optional	String	Identifier of a business or technical process to what the set of requests and response pairs are organized (e.g. paging of transaction history should have same Process-ID). Although it may be arbitrary string, it is strongly (UUID) version 4 form (RFC4122).

AISP Operation: Account information

The operation provides the relevant data about PSU account identified by IBAN and two types of account balances: Interim booked and interim available balance. Only AISP is allowed to use current endpoint.

Endpoint: POST

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/api/v1/accounts/information
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/api/v1/accounts/information

Request

Attributes structure Level 1	Optionality	Type	Description
iban	Mandatory	String [34]	International Bank Account Number (IBAN)

Response (if no error)

Attributes sturcture Level 1	Level 2	Level 3	Optionality	Type	Description
account	name		Mandatory	String [70]	Account name - usually client name
account	productName		Optional	String [70]	Product name - commercial product designation
account	type		Optional	Enum	Account type is enumeration: ISO 20022 - Cash Account Type Code e.g. (CACC - Current account)
account	baseCurrency		Mandatory	String [3]	Account currency (currency code

					according to ISO 4217 - 3 capital letters) Balance type is enumeration: ISO 20022
balances	typeCodeOrProprietary		Mandatory	Enum	Balance Type Code. Following mandatory balances are published: ITBD (Interim booked balance) ITAV (Interim available balance) BLCK (Blocked) CRDL (CreditLine)
balances	amount	value	Mandatory	Number Float [12.2]	Balance amount. Numeric value of the amount as a fractional number. The fractional part has a maximum of two digits
balances	amount	currency	Mandatory	String [3]	Balance currency (currency code according to ISO 4217 - 3 capital letters)
balances	creditDebitIndicator		Mandatory	Enum	Credit/Debit indicator is enumeration: CRDT (Credit) DBIT (Debit)
balances	dateTime		Mandatory	DateTime	Timestamp of balances (official local date and time of Slovak republic in RFC 3339 format)

Error codes

Set of HTTP Status codes and corresponding custom error codes:

HTTP Status	Error code	Description
400	parameter_missing	Mandatory Headers parameter is missing
400	parameter_invalid	Value of input parameter is not valid
500	server_error	Internal Server Error
403	forbidden	Authorization IBAN server error, client can not access requested IBAN
401	authorization_error	Authorization server error

AISP Operation: Account transactions

The operation provides the list of financial transactions performed on a bank account in Prima banka within a date period. Transaction history include transactions that affect the balance (reserved and booked transaction). Transactions are be ordered from the most recent to the oldest. The range of attributes provided for transactions is based on ISO 20 022 - CAMT.054. Only AISP is allowed to use current operation.

Endpoint: POST

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/api/v1/accounts/transactions
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/api/v1/accounts/transactions

Request

Attributes structure Level 1	Optionality	Type	Description
iban	Mandatory	String [34]	International Bank Account Number (IBAN)
dateFrom	Optional	Date	The starting date of a date period for transaction history. Default value is actual day.
dateTo	Optional	Date	The end date of a date period for transaction history. ASPSPs provide transaction's history for at least 13 months. Default value is actual day.
pageSize	Optional	Integer	The number of records included in one page for displaying. Default value is 50 records. ASPSP has to supports maximum 100 records on page.
page	Optional	Integer	The sequence number of a page in regards to page size for a record set. Because it starts at number 0, it should be considered as an offset from the beginning from a page set. Default value is 0.
status	Optional	Enum	Transaction status indicator is enumeration: BOOK (booked transactions) INFO (settled transactions) ALL (ALL transactions) Default value is ALL

Response (if no error)

Collection of information sets about customer's financial transactions executed at their bank account.

Attributes structure					Optionality	Type	Description
Level0	level 1	Level 2	Level 3	Level 4			
pageCount					Optional	Number	Number of pages in the selected range
transactions	amount	value			Mandatory	Number Float [12.2]	Transaction amount value in account currency. Numeric value of the amount as a fractional number.
	amount	currency			Mandatory	String [3]	Transaction amount currency. Formated in Alphabetic codes from ISO 4712.
	creditDebitIndicator				Mandatory	Enum	Credit/Debit indicator is enumeration: - CRDT (Credit) - DBIT (Debit)
	reversalIndicator				Optional	boolean	The flag determining that it is the reversal transaction for some previous one.
	status				Mandatory	Enum	The status of a transaction, related to the query parameter 'transactionStatus'. Transaction status indicator is enumeration: - BOOK (booked transactions) - INFO (settled transactions)
	bookingDate				Mandatory for booked tnx.	Date	Transaction booking date. The requested date by a bank customer to execute the transaction.
	valueDate				Mandatory	Date	Transaction value date. The date of the execustion of the transaction.
	bankTransactionCode				Optional	String [11]	The category code of the transaction type from the SBA's code list.
	bankTransactionCode	bankTransactionCodeDomain			Optional	String [11]	The domain code of the transaction type from the SBA's code list.
	bankTransactionCode	bankTransactionCodeFamily			Optional	String [11]	The family code of the transaction type from the SBA's code list.
	bankTransactionCode	bankTransactionCodeSubFamily			Optional	String [11]	The SubFamily code of the transaction type from the SBA's code list.
	transactionDetails	references	accountServiceReference		Optional	String [35]	The unique identifier of the transaction generated by a ASPSP that it should be considered as a ASPSP reference.

transactionDetails	references	instructionIdentification		Optional	String [35]	Technical identification of the payment generated by a client.
transactionDetails	references	endToEndIdentification		Mandatory in case this attribute is provided by client	String [35]	Unique identification defined by a requestor.
transactionDetails	references	transactionIdentification		Optional	String [35]	The payment reference for related fees.
transactionDetails	references	mandateIdentification		Mandatory for Direct debit tnx.	String [35]	The mandate referece as its reference number.
transactionDetails	references	chequeNumber		Optional	String [35]	For card transactions, this is the card number in format **** **** **** 1111
transactionDetails	counterValueAmount	amount	value	Optional	Number Float [12.2]	Transaction amount value in account currency
transactionDetails	counterValueAmount	amount	currency	Optional	String [3]	Transaction amount currency. Formated in Alphabetic codes from ISO 4712.
transactionDetails	counterValueAmount	currencyExchange	exchangeRate	Optional	Number Float [12.6]	The used exchange rate for conversion from the instructed currency to the target account currency.
transactionDetails	relatedParties	debtor	name	Optional	String [140]	Name of the debtor
transactionDetails	relatedParties	debtorAccount	identification	Optional	String [34]	Unique identification of the debtor account, usually IBAN.
transactionDetails	relatedParties	creditor	name	Optional	String [140]	Name of the creditor
transactionDetails	relatedParties	creditor	identification	Optional	String [35]	The creditor identifier (CID) in the direct debit transaction.
transactionDetails	relatedParties	creditorAccount	identification	Optional	String [34]	Unique identification of the creditor account, usually IBAN.
transactionDetails	relatedParties	tradingParty	name	Optional	String [140]	Name of a third party. For card transaction, this is the name of merchant.
transactionDetails	relatedParties	tradingParty	identification	Optional	String [35]	Unique identification of a third party. For card transaction, this is ID of merchant.

transactionDetails	relatedParties	tradingParty	address	Optional	String [70]	Merchant cummulative address identification usually containing concatenation of street name, street number, etc.
transactionDetails	relatedParties	tradingParty	CountryCode	Optional	String [2]	The two letter merchant country code adopted from ISO3166.
transactionDetails	relatedParties	tradingParty	merchantCode	Optional	String [4]	A Merchant Category Code (MCC) coordinated by MasterCard and Visa.
transactionDetails	relatedAgents	debtorAgent	financialInstitutionIdentification	Optional	String [11]	Corresponding identification of a debtor bank managing the account, usually Bank Identification Code (BIC).
transactionDetails	relatedAgents	creditorAgent	financialInstitutionIdentification	Optional	String [11]	Corresponding identification of a creditor bank managing the account, usually Bank Identification Code (BIC).
transactionDetails	remittanceInformation			Mandatory in case this attribute is provided by client	String [140]	The text aimed as the information for a receiver of the transaction.
transactionDetails	relatedDates	acceptanceDateTime		Optional	Date	Transaction entry date. The date of receiving the transaction in a bank.
transactionDetails	additionalTransactionInformation			Optional	String [140]	Bank transaction description.

Error codes

Set of HTTP Status codes and corresponding custom error codes:

HTTP Status	Error code	Description
400	parameter_missing	Mandatory parameter is missing
400	parameter_invalid	Value of input parameter is not valid
500	server_error	Authorization server error.
403	forbidden	Authorization IBAN server error, client can not access requested IBAN
401	authorization_error	Authorization server error

AISP Operation: List of accounts

The operation provides the list of accounts to which the client has given a long-term mandate to specific TPP (not a list of all client accounts) without balances.

Endpoint: GET

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/api/v1/accounts
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/ api/v1/accounts

Request

Payload is empty.

Response

Attributes sturcture Level 1	**Level 2**	**Optionality**	**Type**	**Description Level 3**
creationDateTime		Mandatory	DateTime	The **date and time** in RFC3339 format at which a particular action has been requested or executed.
accounts	*iban*	Mandatory	String	International Bank Account Number (IBAN)
accounts	*name*	Mandatory	String [70]	**Account name** - usually client name
accounts	*productName*	Optional	String [70g]	**Product name** - commercial product designation
accounts	*type*	Optional	Enum	**Account type** is enumeration: ISO 20022 - Cash Account Type Code e.g. (CACC - Current account)
accounts	*baseCurrency*	Mandatory	String [3]	**Account currency** (currency code according to ISO 4217 - 3 capital letters)
accounts	*servicer*	Mandatory	String [11]	Corresponding identification of a servicing bank managing the account, usually **Bank Identification Code** (BIC).
accounts	*consent*	Mandatory	Array [String]	Consent containts set of particular account's scopes for TPP. Formated as array of following enumerations: AISP, PISP, PIISP.

Error codes

Set of HTTP Status codes and corresponding custom error codes:

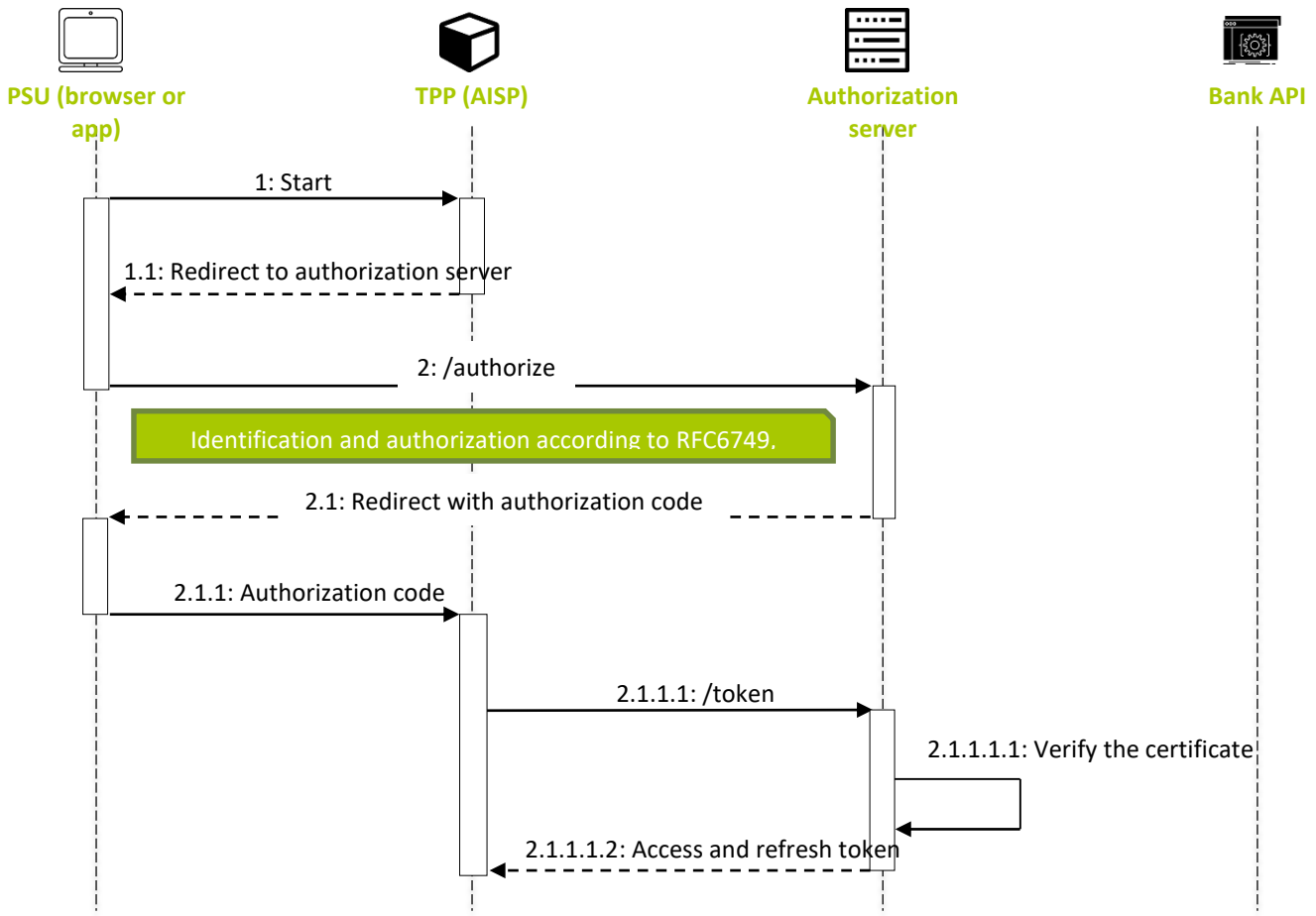
HTTP Status	Error code	Description
400	parameter_missing	Mandatory parameter is missing
401	authorization_error	Authorization server error
500	server_error	Internal server error

Token for AISP services

To access ASPSP API, the TPP must use a valid **access_token** with AISP scope. According to OAuth framework, valid **access_token** and **refresh_token** are used to access ASPSP and PSU' s resources. TPP uses a short-term **access_token** to communicate with the API of the ASPSP if the ASPSP requires it and it MAY use **refresh_token** to request a new **access_token**.
The OAuth2 PKCE extension (RFC 7636 https://tools.ietf.org/html/rfc7636) is used to issue **access_token** using **code_challenge** and **code_verifier** technique.

Basic properties

- **access_token** is issued as short-term (300 s) and be canceled (by PSU, TPP or ASPSP)
- **refresh_token** can not be directly used to communicate with the API, it has a long validity (180 days) and the ASPSP has the option to cancel it and this option can also allow to the PSU.
- **iban –** is used for the generate token only for a selected IBAN - optional
- The ASPSP and the TPP application share a common "secret" **client_secret**
- The result of identification and authentication is the code that the TPP application must replace with the client secret for **refresh_token** and **access_token**
- the code itself without **client_secret** knowledge CAN NOT be used
- Under the code grant flow, the ASPSP is not required to execute the SCA of PSU of the ASPSP to authorize the TPP's access to ASPSP resources related to that PSU

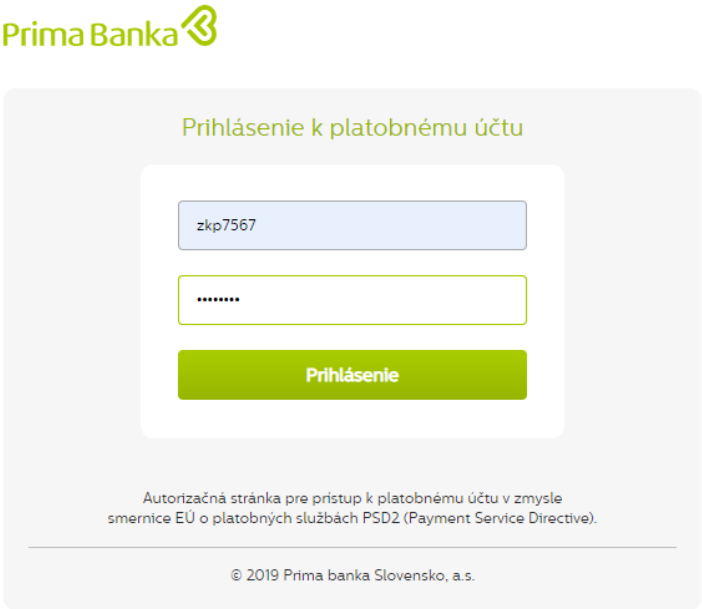


Authorization

The AISP creates an Authorization request for the PSU to consent to the AISP request. The request is an Oauth 2.0. Authorization Code Grant with PKCE extension (requesting for Code)

Endpoint: GET

LIVE	https://psd2.primabanka.sk/loginAPI/authorize
DEV	https://ib.primabanka.sk/loginAPI_test/authorize



Request

Attribute	Optionality	Type	Description
response_type	Mandatory	Code	Mandatory parameter. Specifies the authentication flow used. In this case, a code grant. For the authentication process, this means that, as a result

			of successful identification and authentication, a one-time auth_code is expected instead of access_token.
client_id	Mandatory	String	Unique TPP application identifier issued by the ASPSP, eg. using the process defined in Section Automated assigning of a technical identifier
redirect_uri	Mandatory	URL	The URL to which the authentication flow is redirected at the end. This URL is set when client_id is issued, and this parameter is validated against the URL introduced to client_id in the ASPSP. The value should match one of the values introduced using registration e.g. using the process defined in Section 4.5.1
scope	Mandatory	String	Underscore/plus sign separated string of attributes of the application required scope, eg.: AISP/PISP/AISP_PISP
login_hint	Optional	User identification for automation	Hint to the Authorization Server about the login identifier the End-User might use to log in (http://openid.net/specs/openid-connect-core-1_0.html)
state	Mandatory	Random string [min 128 bits]	With this parameter, TPP needs to enrich redirect_uri when redirecting. It protects against CSRF attacks and passes information from the application through authentication flow. Requested CSRF token length is min. 128 bits. TPP has to verify this value upon receipt authorization code.
code_challenge	Mandatory	String	code_challenge = BASE64URLENCODE(SHA256(ASCII(code_verifier))) see. RFC 7636 (OAuth PKCE)
code_challenge_method	Mandatory	String	S256

Response

Attribute	Optionality	Type	Description
code	Mandatory	String	Authorization code
state	Mandatory	String	Attribute state from TPP request

2: HTTP Request example: GET /authorize

Endpoints

DEV:
https://ib.primabanka.sk/loginAPI_test/authorize?response_type=code&scope=AISP&client_id=8FEF489748174876B059DE9386378467&state=VsH0TiAB1d3t7yR6VvD31DpUZEVrBXAQ&redirect_uri=http://www.primabanka.sk&code_challenge=oO77bZ2WVsphzUSIihF1VUB2H0AE5auo8uP_x8axjW0&code_challenge_method=S256

LIVE:
https://psd2.primabanka.sk/loginAPI_test/authorize?response_type=code&scope=AISP&client_id=8FEF489748174876B059DE9386378467&state=VsH0TiAB1d3t7yR6VvD31DpUZEVrBXAQ&redirect_uri=http://www.primabanka.sk&code_challenge=oO77bZ2WVsphzUSIihF1VUB2H0AE5auo8uP_x8axjW0&code_challenge_method=S256

Header

```
GET /authorize HTTP/1.1
Content-Type: application/x-www-form-urlencoded
```

HTTP Response example: GET /authorize

Header

```
HTTP/1.1 303
Content-Type: application/x-www-form-urlencoded
Location:https://www.tpp.sk/index?code=gCyAymoimg0L1bEI&state=VsH0TiA
B1d3t7yR6VvD31DpUZEVrBXAQ
```

The PSU is redirected to the AISP with Authorization code and state parameters in URL.

Get token

The AISP will now possess the Authorization code and state parameter from the ASPSP. State parameter value must be identical as requested by AISP in the previous request otherwise, the response is invalid. AISP will proceed to obtain an Access Token from the ASPSP using the Authorization Code. The AISP will present its Authorization Code together with CLIENT_ID and CLIENT_SECRET in authorization header.
The Access Token is required by the AISP in order to access PSU Account information. The AISP scope should already be associated with the Authorization Code generated in the previous step.

Endpoint: POST

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/token
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/token

Request

Attribute	Optionality	Type	Description
code	Mandatory	String	Authorization code returned from the code grant
redirect_uri	Mandatory	URL	The redirect URL matches the URL passed in the authentication request.
grant_type	Mandatory	String	Under the existing OAuth2 definition, this value will be the authorization_code if the TPP requested refresh_token.
code_verifier	Mandatory	String	Code_verifier used to generate code_challenge from a previous request with a minimum length of 43 characters and a maximum length of 128 characters
iban	Optional	string	Token for the IBAN – simple array separator comma

Response

Attribute	Optionality	Type	Description
access_token	Mandatory	String	Short-term 300 seconds token, which can be reissued using refresh_token. This token serves to authorize TPP request on ASPSP API.
expires_in	Mandatory	Number	The remaining time to expiration of access_token - in seconds.
refresh_token	Optional	String	Long-term token 180 days issued as a replacement for authorization_code.
token_type	Mandatory	String	Type of token „Bearer“
scope	Optional	String	List of permissions separated by the space for which the token is issued.

HTTP Request example: POST /token

Header

```
POST /token HTTP/1.1
Content-Type:      application/x-www-form-urlencoded
Authorization:     Basic YTBiMjUyOTFmMDpCQmprazQ1c2Q3OGFkNDU0Z2RkZDg3MTJfNDU
1NWc1ZzVnNWdn //Basic BASE64(CLIENT_ID + ":" + CLIENT_SECRET)
```

Body

```
grant_type=authorization_code&code=03ab43f0-0257-477c-b06c-
ab61b2b1080d&redirect_uri=https:\/\/ib.primabanka.sk\/ib\/default.aspx&code_verifier=04e10457-7e67-4d9e-9b49-
121f7eee5396&iban = SK12 5600 0000 1324 4560
```

HTTP Response example: POST /token

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
```

Body

```
{
"access_token":" IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX",
"token_type":"bearer",
"expires_in":300,
"refresh_token":" be9eef9b0af42c674d0b1c1128c37c2g"
}
```

Access token renew

The TPP can save the **refresh_token** from the Get token resource and ask for a new **access_token** after the expiration of **access_token** through this token. Therefore, TPP can use Get token resource with these parameters:

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/token
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/token

Request

Attribute	Optionality	Type	Description
grant_type	Mandatory	String	Under the existing OAuth2 definition, this value will be the authorization_code if the TPP requested refresh_token.
refresh_token	Mandatory	String	Valid refresh_token for which the exchange takes place.
scope	Mandatory	String	The scope of the access request.

Response

Attribute	Optionality	Type	Description
access_token	Mandatory	String	Short-term (e.g. 3600 seconds, in some cases, onetime) token, which can be reissued using refresh_token. This token serves to authorize TPP request on ASPSP API.
token_type	Mandatory	String	Type of token „Bearer“
expires_in	Mandatory	Number	The remaining time to expiration of access_token - in seconds
refresh_token	Optional	String	Valid refresh_token for which the exchange takes place

HTTP Request example:

Header

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Authorization: Basic YTBiMjUyOTFmMDpCQmprazQ1c2Q3OGFkNDU0Z2RkZDg3MTJfNDU
1NWc1ZzVnNWdn // Basic BASE64(CLIENT_ID + ":" + CLIENT_SECRET)
```

Body

```
grant_type=refresh_token&
refresh_token= be9eef9b0af42c674d0b1c1128c37c2g&
scope= AISP PISP
```

HTTP Response example:

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
```

Body

```
{
"access_token": "0F7HZ1OBL0KTXXXIOUNFOFK6ZKR5T2TH",
"token_type":"bearer",
"expires_in":300,
"refresh_token":" r986fxs7elrtvzzn7kj8xrmnlv5zkwss"
}
```

Revoke refresh token

The TPP can revoke a **refresh_token** at will using this endpoin. Revoking a **refresh_token** also revokes its associated **access_tokens.**

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/revokeRefreshToken
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/revokeRefreshToken

Request

Attribute	Optionality	Type	Description
refresh_token	Mandatory	String	Valid refresh_token to be revoked.

Response

Attribute	Optionality	Type	Description
result	Mandatory	String	Value indicating success of the revocation. The value is „RVKD“ in case of success. An unsuccessful attempt results in an error.

HTTP Request example:

Header

```
POST /token HTTP/1.1
Content-Type: application/json;charset=UTF-8
Authorization: Basic YTBiMjUyOTFmMDpCQmprazQ1c2Q3OGFkNDU0Z2RkZDg3MTJfNDU
1NWc1ZzVnNWdn // Basic BASE64(CLIENT_ID + ":" + CLIENT_SECRET)
```

Body

```
{
"refreshToken":"17bef4c8-b0a4-4852-9bed-b5d2ba82a447"
}
```

HTTP Response example:

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
```

Body

```
{
"result":"RVKD"
}
```

Usage Example of AISP Operation: Account information

HTTP Request example: POST /api/v1/accounts/information

Endpoints

LIVE	*https://psd2.primabanka.sk/ServiceAPI/API.svc/ api/v1/accounts/information*
DEV	*https://ib.primabanka.sk/ServiceAPI_test/API.svc/api/v1/accounts/information*

Header

```
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX
Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
PSU-Last-Logged-Time: 2019-02-16T14:54:32+01:00
PSU–IP-Address: 192.168.0.100
```

Body


```
{
   "iban":"SK3456000000000533628001"
}
```


HTTP Response example: POST /api/v1/accounts/information

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
```

Body


```
{
   "account":
   {
      "name":"Jan Knoška",
      "productName":"KNOSKA JAN",
      "type":"CACC",
      "baseCurrency":"EUR"
   },
   "balances":
   [
      {
         "typeCodeOrProprietary":"ITAV",
         "amount":
         {
            "value":2440.66,
            "currency":"EUR"
         },
         "creditDebitIndicator":"CRDT",
         "dateTime":"2019-07-12T14:41:42+02:00"
      },
      {
         "typeCodeOrProprietary":"ITBD",
         "amount":
         {
            "value":2440.66,
            "currency":"EUR"
         },
         "creditDebitIndicator":"CRDT",
         "dateTime":"2019-07-12T14:41:42+02:00"
```

```
    },
    {
      "typeCodeOrProprietary":"BLCK",
      "amount":
      {
        "value":10,
        "currency":"EUR"
      },
      "creditDebitIndicator":"DBIT",
      "dateTime":"2019-07-12T14:41:42+02:00"
    }
    {
      "typeCodeOrProprietary":"CRDL",
      "amount":
      {
        "value":500,
        "currency":"EUR"
      },
      "creditDebitIndicator":"CRDT",
      "dateTime":"2019-07-12T14:41:42+02:00"
    }
  ]
}
```

Usage Example of AISP Operation: Account *transactions*

https://www.iso20022.org/standardsrepository/public/wqt/Description/mx/dico/codesets/_bbFhQNp-Ed-ak6NoX_4Aeg_142948041

HTTP Request example: POST /api/v1/accounts/transactions

Endpoints:

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/api/v1/accounts/transactions
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/api/v1/accounts/transactions

Header

```
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX
```

...

Body


```
{
  "iban":"SK3456000000000533628001",
  "dateFrom":"2019-01-01",
  "dateTo":"2019-05-29",
  "pageSize":50,
  "page":0,
  "status":"ALL"
}
```


HTTP Response example: POST /api/v1/accounts/transactions


```
{
  "pageCount":62,
  "transactions":
  [
    {
      "amount":
      {
        "value":1.00,
        "currency":"EUR"
      },
      "creditDebitIndicator":"DBIT",
      "reversalIndicator":"",
      "status":"INFO",
      "bookingDate":"",
      "valueDate":"2019-05-01",
      "bankTransactionCode": {
        "bankTransactionCodeDomain": "PMNT",
        "bankTransactionCodeFamily": "ICDT",
        "bankTransactionCodeSubFamily": "ESCT"
        },
       "transactionDetails":
      {
        "references":
        {
          "accountServiceReferenc":"",
          "instructionIdentification":"",
          "endToEndIdentification":"",
          "transactionIdentification":"",
          "mandateIdentification":"2EC6050E-DEA6-462F-BD19-A0136F572290",
          "chequeNumber":""
        },
```

Usage Example of AISP Operation: List of accounts

HTTP Request example: POST /api/v1/accounts/information

Endpoints

LIVE	*https://psd2.primabanka.sk/ServiceAPI/API.svc/ api/v1/accounts*
DEV	*https://ib.primabanka.sk/ServiceAPI_test/API.svc/api/v1/accounts*

Header

```
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX
Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
PSU-Last-Logged-Time: 2019-02-16T14:54:32+01:00
PSU–IP-Address: 192.168.0.100
```

Body

```
{
}
```

HTTP Response example: POST /api/v1/accounts

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
```

Body


```
{
  "creationDateTime":"2020-12-08T17:03:04+01:00",
  "accounts":
  [
    {
      "name":"PaedDr. Vzorka Specimen MBA",
      "productName":"SPECIMEN VZORKA",
      "type":"CACC",
      "baseCurrency":"EUR",
      "iban":"SK12 5600 0000 0092 2594 7001",
      "servicer":"KOMASK2X",
      "consent":"[\"AISP\", \"PISP\"]"
    }
  ]
}
```

Payment Initiation Service Provider (PISP)

Prerequisites:
a) The TPP is registered for the PISP role and valid PISP scope
b) The TPP has been successfully authenticated
c) The TPP has presented its access token to call PISP services.

Endpoints definition

In following sections describe technical definition of provided endpoints for PISPs.

Method Optionality Description	Method	Optionality	Description
/api/v1/payments/standard/iso	POST	Mandatory	Standard payment initialization – service allows to initialize payment in XML format (PAIN.001)
/api/v1/payments/standard/isoBatch	POST	Optional	Standard batch payment initialization – service allows to initialize batch payment in XML format (PAIN.001)
/api/v1/payments/submission	POST	Mandatory	Standard payment submission – service allows to authorization of initialized payments and batch payments
/api/v1/payments/{orderId}/status	GET	Mandatory	Payment order status – service provides actual information about initialized payment
/api/v1/payments/{orderId}/statusBatch	GET	Mandatory	Batch payment order status – service provides actual information about initialized batch payment and its individual payments
/api/v1/payments/{orderId}/rcp	DELETE	Optional	Request to cancel paymnet – service allows to cancel payment, that were initiated through the same PISP by services Standard Payment Initializaton (XML or Standard Payment Initializaton
/api/v1/payments/foreign			Standard payment initialization – service allows to initialize non- sepa payment in JSON format (PAIN.001)

Standard header definition

Set of request and response headers for PISP endpoints

Request header definition

Attribute	Optionality	Type	Description
Host	Mandatory	String	Domain name of the server and optional TCP port number
Content-Type	Mandatory	String	application/json or application/xml
Authorization	Mandatory	String	Authorization is defined in RFC 6750 - The OAuth 2.0 Authorization Framework: Bearer Token Usage
Request-ID	Mandatory	String	An unique identifier of a particular request message.
Correlation-ID	Optional	String	An unique correlation identifier correlates the request
Process-ID	Optional	String	Identifier of a business or technical process to what the set of requests and response pairs are organized (e.g. paging of transaction history should have same Process-ID).
PSU–IP-Address	Mandatory	String	Identifier of a customer's IP address from which he/she is connected to the TPP infrastructure. It might be in the format of IPv4 o IPv6 address. ASPSP shall indicate which values are acceptable.
PSU-Device-OS	Mandatory	String	A customer's device and/or operating system identification from which he/she is connected to the TPP infrastructure.
PSU-User-Agent	Mandatory	String	A customer's web browser of other client device identification from which he/she is connected to the TPP infrastructure. Agent header field of the http request between PSU and TPP.)
PSU-Geo-Location	Optional	String	The GPS coordinates of the current customer's location in the moment of connection to the TPP infrastructure. (Required GPS format)
PSU-Last-Logged-Time	Optional	DateTime	Last date and time when user was logged to TPP app (RFC3339 format)
PSU-Presence	Optional	Enum	The presence status of user (PSU) during an API call. The value of the parameter could be „true“ (PSU is present) or „false“ (PSU is not present).

Response header definition

Attribute	Optionality	Type	Description
Content-Type	Mandatory	String	application/json or application/xml
Response-ID	Mandatory	String	An unique identifier of a particular request message.
Correlation-ID	Optional	String	An unique correlation identifier correlates the request and the response messages as a pair especially used for audit logs.
Process-ID	Optional	String	Identifier of a business or technical process to what the set of requests and response pairs are organized (e.g. paging of transaction history should have same Process-ID).

PISP Operation: Standard payment initialization (XML) - SEPA payments

The operation allows initialize payment in XML format (PAIN.001.001.03). The PISP sends a ISO20022 pain.001.001.03 based structure that specifies the payment activation request that is related to a commercial transaction between a PSU and the merchant.The ISO20022 pain.001.001.03 structure is also described in national standard for **SEPA** . Operation is used only for the SEPA payments.

If PSU/debtor IBAN is not present, status IBAN will be returned. TPP should then redirect to PSU to our Web where the PSU logs in and chooses their desired IBAN the payment is to be carried out from. The redirect URL is: https://psd2.primabanka.sk/loginAPI/authorize?redirect_iban={orderId}.

Endpoint: POST /api/v1/payments/standard/iso

Request

Message contains xml: pain.001.001.03

Response (if no error)

Message contains xml: pain.002.001.03

Attribute	XML structuremapping	Optionality	Type	Description
orderId	TxInfAndSts/ AcctSvcrRef	Mandatory	String [36]	OrderId is Unique reference, as assigned by the account servicing institution, to unambiguously identify the instruction.
status	TxInfAndSts/ TxSts	Mandatory	Enum	Transaction status indicator is enumeration: ACTC (AcceptedTechnicalValidation) ACWC (AcceptedWithChange) RJCT (Rejected) IBAN(Debtor IBAN not present)
reasonCode	TxInfAndSts/ StsRsnInf/Rsn	Optional	Enum	ISO 20022 Status Reason Code
statusDateTime	GrpHdr/CreDtTm	Mandatory	DateTime	Transaction entry date. The date of receiving the transaction in a bank.

Set of HTTP Status codes and corresponding custom error codes:

HTTP Status	Error code	Description
400	parameter_missing	Mandatory parameter is missing
400	parameter_invalid	Value of input parameter is not valid
500	server_error	Authorization server error.
403	forbidden	Authorization IBAN server error, client can not access requested IBAN
401	authorization_error	Authorization server error

PISP Operation: Standard batch payment initialization (XML) - SEPA payments

The operation allows initialize payment in XML format (PAIN.001.001.03). The PISP sends a ISO20022 pain.001.001.03 based structure that specifies the payment activation request that is related to a commercial transaction between a PSU and the merchant. Operation is used only for batch SEPA payments.

Endpoint: POST /api/v1/payments/standard/isoBatch

Request

Message contains xml: pain.001.001.03

Response (if no error)

Attributes sturcture		Optionality	Type	Description
Level 1	Level 2			
batchOrderId		Mandatory	String [40]	A unique reference to the batch payment. Used to submit the payments in a batch.
result		Mandatory	Enum	Overall result of the batch payment. Enumerations are: ACTC: All payments in the batch were accepted. RJCT: All payments in the batch were rejected. PRTL: At least one payment was accepted, and at least one was rejected.
paymentStatuses		Mandatory	List of /iso results	Lists responses for each payment in the batch. Identical to the Response of the /iso endpoint.

Set of HTTP Status codes and corresponding custom error codes:

HTTP Status	Error code	Description
400	parameter_missing	Mandatory parameter is missing
400	parameter_invalid	Value of input parameter is not valid
500	server_error	Authorization server error.
403	forbidden	Authorization IBAN server error, client can not access requested IBAN
401	authorization_error	Authorization server error

PISP Operation: Standard payment initialization (JSON) - NON SEPA payments

The operation allows initialize payment in JSON format. The PISP sends JSON structure message. Operation is used for the non sepa payments.

Request

Attribute	Optionality	Type	Description
instructionIdentification	Mandatory	String	Mandatory parameter. Specifies the unique guid
creationDateTime	Mandatory	String	Payment Date time creation
debtor	Mandatory		
name	Mandatory	String	Name of the debtor
account	Mandatory	String	Account number
creditor	Mandatory	String	Creditor segment
name	Mandatory		
account	Mandatory		
address	Optional		
country	Optional		
bank	Mandatory		Creditor bank segment
name	Mandatory		
swift	Mandatory		
address	Optional		
instructedAmount	Mandatory		Payments
value	Mandatory		Amount
currency	Mandatory	String	Currency
endToEndIdentification	Mandatory	String	
remittanceInformation	Mandatory	String	
requestedExecutionDate		Datetime	Requested execution datetime
purposeCode		String	
fee		String	
isExpress	Optional	boolean	Express payment (max 12:00 AM)

Response (if no error)

Attribute	XML structuremappin	Optionality	Type	Description
orderId	TxInfAndSts/ AcctSvcrRef	Mandatory	String [35]	OrderId is Unique reference, as assigned by the account servicing institution, to unambiguously identify the instruction.
status	TxInfAndSts/ TxSts	Mandatory	Enum	Transaction status indicator is enumeration: ACTC (AcceptedTechnicalValidation) ACWC (AcceptedWithChange) RJCT (Rejected)
reasonCode	TxInfAndSts/ StsRsnInf/Rsn	Optional	Enum	ISO 20022 Status Reason Code
statusDateTime	GrpHdr/CreD tTm	Mandatory	DateTime	Transaction entry date. The date of receiving the transaction in a bank.

Error codes

Set of HTTP Status codes and corresponding custom error codes:

HTTP Status	Error code	Description
400	parameter_missing	Mandatory parameter is missing
400	parameter_invalid	Value of input parameter is not valid
500	server_error	Authorization server error.
401	authorization_error	Authorization server error

PISP Operation: Standard payment submission SEPA/ NON-SEPA

The operation provides authorization of initialized payment.

Endpoint: POST /api/v1/payments/submission

Request

Request (if authorization is requeired – dependend on the user setup SMS/ VASCO)

Attributes structure	Optionality	Type	Description
code	Optional	String [20]	SMS /VASCO verification code
orderId	Mandatory	String [36]	OrderId is Unique reference, as assigned by the account servicing institution, to unambiguously identify a payment or a batch of payments.

The authorization header will contain a "bearer token" that coresppond to " orderId".

Response (if no error)

Attributes structure Level 1	Optionality	Type	Description
orderId	Mandatory	String [36]	OrderId is Unique reference, as assigned by the account servicing institution, to unambiguously identify the instruction.
status	Mandatory	Enum	Transaction status indicator is enumeration: ACTC (AcceptedTechnicalValidation) ACWC (AcceptedWithChange) RJCT (Rejected) ACCR (AcceptedCancellationRequest), RJCR (RejectedCancellationRequest)
reasonCode	Optional	Enum	Additional transaction indicator enumeration: SEPA (Accepted SEPA payment) FOREIGN (Accepted foreign payment) SMS(Rejected payment due to incorrect SMS code) ORDERID(Rejected payment due to incorrect orderId)
statusDateTime	Mandatory	DateTime	The date and time in RFC3339 format at which a particular action has been requested or executed.

Error codes

Set of HTTP Status codes and corresponding custom error codes:

HTTP Status	Error code	Description
400	parameter_missing	Mandatory parameter is missing
400	parameter_invalid	Value of input parameter is not valid
500	server_error	Internal server error
401	authorization_error	Authorization server error

PISP Operation: Payment order status

The operation provides information about processing status of a received payment instruction based on payment orderId identification.

Endpoint: GET /api/v1/payments/{orderId}/status

Request

Payload is empty.

Response (if no error)

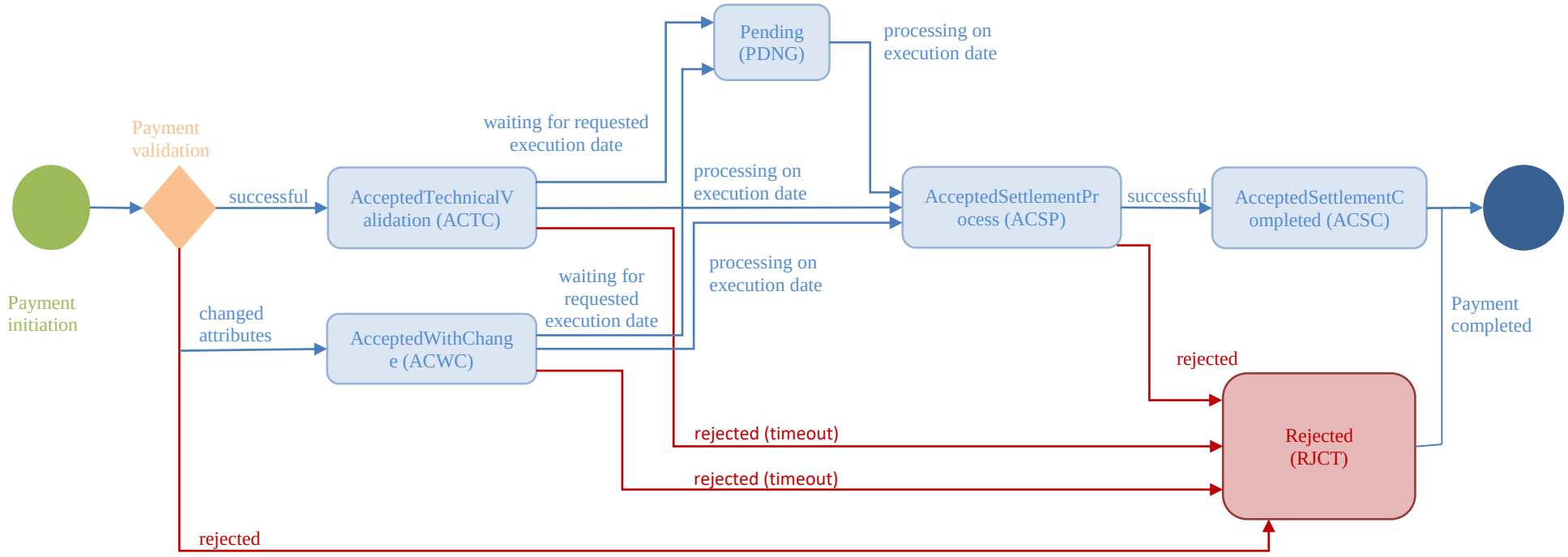
Attributes structure Level 1	Optionality	Type	Description
orderId	Mandatory	String [36]	OrderId is Unique reference, as assigned by the account servicing institution, to unambiguously identify the instruction.
status	Mandatory	Enum	Transaction status indicator is enumeration: ACTC (AcceptedTechnicalValidation) ACWC (AcceptedWithChange) RJCT (Rejected) PDNG (Pending) ACSP (AcceptedSettlementInProcess) ACSC (AcceptedSettlementCompleted) IBAN – returned when PSU initializes a payment without explicit debtor IBAN, before choosing IBAN
reasonCode	Optional	Enum	ISO 20022 Status Reason Code
statusDateTime	Mandatory	DateTime	The date and time in RFC3339 format at which a particular action has been requested or executed.

Error codes

Set of HTTP Status codes and corresponding custom error codes:

HTTP Status	Error code	Description
400	parameter_missing	Mandatory Headers parameter is missing
400	parameter_invalid	Value of input parameter is not valid
500	server_error	Internal Server Error
401	authorization_error	Authorization server error

Expected flow of payment's states:



This operation provides following payment status codes:

Attribute	Description
ACTC	AcceptedTechnicalValidation - Authentication and syntactical and semantical validation are successful.
ACWC	AcceptedWithChange - Instruction is accepted but a change will be made, such as date or remittance not change
PDNG	Pending – payment initiation or individual transaction included in the payment initiation is pending. Further checks and status update will be performed.
ACSP	AcceptedSettlementInProcess - All preceding checks such as technical validation and customer profile were successful and therefore the payment initiation has been accepted for execution.
ACSC	AcceptedSettlementCompleted – Settlement on the debtor's account has been completed. Usage: this can be used by the first agent to report to the

debtor that the transaction has been completed. Warning: this status is provided for transaction status reasons, not for financial information. It can only be used after bilateral agreement.

RJCT Rejected - Payment initiation or individual transaction included in the payment initiation has been rejected

Endpoint: GET /api/v1/payments/{orderId}/statusBatch

Request

Payload is empty.

Response (if no error)

Attributes structure		Optionality	Type	Description
Level 1	Level 2	Mandatory	String [36]	BatchOrderId is Unique reference, as assigned by the account servicing institution, to unambiguously identify the payment batch.
batchOrderId				
result		Mandatory	Enum	Overall result of the batch payment. Enumerations are: ACTC: All payments in the batch were accepted. RJCT: All payments in the batch were rejected. PRTL: At least one payment was accepted, and at least one was rejected.
paymentStatuses	orderId	Mandatory	String [36]	OrderId is Unique reference, as assigned by the account servicing institution, to unambiguously identify the instruction.
	status	Mandatory	Enum	Transaction status indicator is enumeration: ACTC (AcceptedTechnicalValidation) ACWC (AcceptedWithChange) RJCT (Rejected) PDNG (Pending) ACSP (AcceptedSettlementInProcess) ACSC (AcceptedSettlementCompleted)

reasonCode	Optional	Enum	ISO 20022 Status Reason Code
statusDateTime	Mandatory	DateTime	The date and time in RFC3339 format at which a particular action has been requested or executed.

PISP Operation: Request to cancel payment

The operation allows to cancel payment, that were initiated through the same PISP by services Standard Payment Initializaton (XML) or Standard Payment Initializaton (JSON).

Endpoint: GET /api/v1/payments/{orderId}/rcp

Request

Payload is empty

Response (if no error)

Attribute structure Level 1	Optionality	Type	Description
orderId	Mandatory	String [36]	OrderId is Unique reference (different from the payment order_ID) , as assigned by the account servicing institution, to unambiguously identify the instruction.

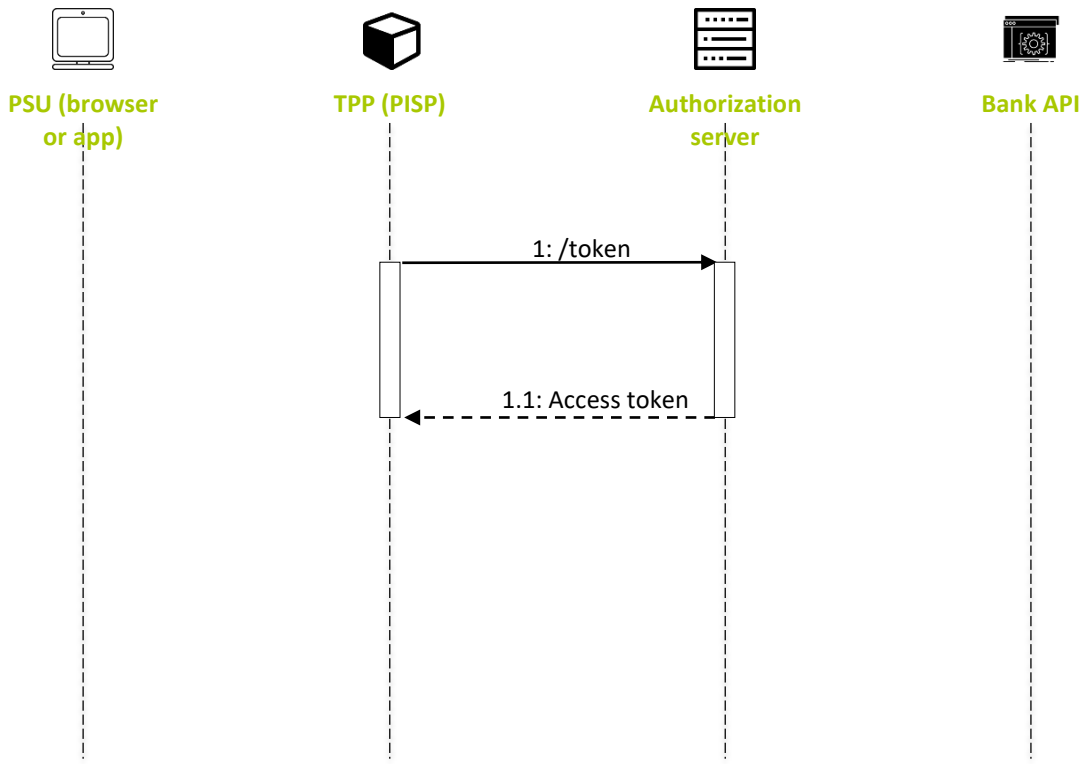
Error codes

Set of HTTP Status codes and corresponding custom error codes:

HTTP Status	Error code	Description
400	parameter_missing	Mandatory Headers parameter is missing
400	parameter_invalid	Value of input parameter is not valid
500	server_error	Internal Server Error
401	authorization_error	Authorization server error

Token for PISP services

STEP 1: To setup a single payment the Client Credentials Grant **access_token** The PISP initiates an Authorization request using valid Client Credentials Grant type and scope(s). The ASPSP Authorization Server validates the Client Authentication request from the PISP and generates an Access Token response where the request is valid



PISP obtains an Access Token using a Client Credentials Grant Type with valid **client_id** and **client_secret** in authorization header. The scope PISP must be used. When an Access Token expires, the PISP will need to re-request for another Access Token using the same request below. This step can be ommited in case of valid AISP **access_token** and **refresh_token.**

Endpoint

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/token
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/token

Request

Attribute	Optionality	Type	Description
grant_type	Mandatory	String	client_credentials exclusively to assign one-time access_token
scope	Mandatory	String	Required scope: "PISP"

Response

Attribute	Optionality	Type	Description
access_token	Mandatory	String	Short-term (one-time) token. This token is used to authorize the API request.
expires_in	Mandatory	Number	The remaining time to expiration of access_token - in seconds.
token_type	Mandatory	String	Type of token „Bearer“

scope	Optional	String	PISP

HTTP Request example: POST /token

Header

```
Content-Type: application/x-www-form-urlencoded
Authorization: Basic YTBiMjUyOTFmMDpCQmprazQ1c2Q3OGFkNDU0Z2RkZDg3MTJfNDU
1NWc1ZzVnNWdn // Basic BASE64(CLIENT_ID + ":" + CLIENT_SECRET)
```

Body

```
grant_type=client_credentials&
scope=PIS2&orderId={orderId}
```

HTTP Response example: POST /token

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
```

Body

```
{
"access_token":"IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX",
"token_type":"bearer",
"expires_in":3600
"scope":"PISP"
}
```

The Client Credentials Grant may optionally be used by the PISP in Step 4 to retrieve the status of a Payment or Payment-Submission where no active Access Token is available

Usage Example of PISP Operation: Standard payment initialization (XML)

STEP 2: The PISP uses the Access Token (with PISP scope) from the ASPSP to invoke the Payments API resource against the ASPSP Resource Server. The ASPSP Resource server responds with the OrderId (and rest of data according specification).

HTTP Request example: POST /api/v1/payments/standard/iso

Header

```
POST /api/v1/payments/standard/iso HTTP/1.1
Content-Type: application/xml;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX
Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
PSU-Device-OS: win
PSU–IP-Address: 192.168.0.100
PSU-User-Agent: curl
```

Body


```
<Document xmlns="urn:iso:std:iso:20022:tech:xsd:pain.001.001.03">
 <CstmrCdtTrfInitn>
  <GrpHdr>
   <MsgId>MCCT1708164657382965</MsgId>
   <CredDtTm>2019-02-16T11:59:20+0100</CredDtTm>
   <NbOfTxs>1</NbOfTxs>
   <CtrlSum>1234.56</CtrlSum>
   <InitgPty>
    <Nm>Johne Doe</Nm>
    <Id>
     <OrgId>
      <Othr>
       <Id>NOTPROVIDED</Id>
      </Othr>
     </OrgId>
    </Id>
   </InitgPty>
  </GrpHdr>
  <PmtInf>
   <PmtInfId>17081600001</PmtInfId>
   <PmtMtd>TRF</PmtMtd>
   <PmtTpInf>
    <InstrPrty>NORM</InstrPrty>
    <SvcLvl>
     <Cd>NURG</Cd>
    </SvcLvl>
    <CtgyPurp>
     <Cd>SEPA</Cd>
    </CtgyPurp>
   </PmtTpInf>
   <ReqdExctnDt>2019-06-28</ReqdExctnDt>
   <Dbtr>
    <Nm> Johne Doe </Nm>
    <Id>
     <OrgId>
      <Othr>
       <Id>NOTPROVIDED</Id>
      </Othr>
     </OrgId>
    </Id>
   </Dbtr>
   <DbtrAcct>
    <Id>
     <IBAN>SK3456000000000533628001 </Iban>
     <Othr>
      <Id>1109532451/7500</Id>
     </Othr>
    </Id>
    <Issr>Issuer</Issr>
   </DbtrAcct>
   <DbtrAgt>
    <FinInstnId>
     <BIC>CEKOSKBX</BIC>
    </FinInstnId>
   </DbtrAgt>
   <ChrgBr>SLEV</ChrgBr>
   <CdtTrfTxInf>
    <PmtId>
     <InstrId>9b766084-57de-48b2-be53-1bd2804ae0b7</InstrId>
     <EndToEndId>/VS123/SS456/KS0308</EndToEndId>
    </PmtId>
    <Amt>
```

```
        <InstdAmt>1234.56</InstdAmt>
        <Ccy>EUR</Ccy>
      </Amt>
      <CdtrAgt>
        <FinInstnId>
          <BIC>TATRSKBX</BIC>
        </FinInstnId>
      </CdtrAgt>
      <Cdtr>
        <Nm>ABC Ltd.</Nm>
        <Id>
          <OrgId>
            <Othr>
              <Id>NOTPROVIDED</Id>
            </Othr>
          </OrgId>
        </Id>
      </Cdtr>
      <CdtrAcct>
        <Id>
          <IBAN>SK7811000000001111111111</Iban>
          <Othr>
            <Id>111111111/1100</Id>
          </Othr>
        </Id>
        <Issr>Issuer</Issr>
      </CdtrAcct>
      <UltmtCdtr>
        <Nm>ABC Ltd.</Nm>
        <Id>
          <OrgId>
            <Othr>
              <Id>4748027</Id>
            </Othr>
          </OrgId>
        </Id>
      </UltmtCdtr>
      <Purp>
        <Cd>RINP</Cd>
      </Purp>
      <RmtInf>
        <Ustrd>Payment for a utility service.</Ustrd>
      </RmtInf>
    </CdtTrfTxInf>
  </PmtInf>
 </CstmrCdtTrfInitn>
</Document>
```


HTTP Response example: POST /api/v1/payments/standard/iso

Header

```
HTTP/1.1 200 OK
Content-Type: application/xml;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
```

Body


```
{
  "orderId":"61357f33-7d09-42ba-aaec-16b95fdf01ea",
  "status":"ACTC",
  "statusDateTime":"2019-08-23T16:33:51+02:00"
}
```

Error response example

```
{
   "Detail":"Exception of type 'System.Exception' was thrown.",
   "Error":"Mandatory parameter is missing"
}
```

Usage Example of PISP Operation: Standard payment initialization (JSON) – NON SEPA payments

STEP 2: The PISP uses the Access Token (with PISP scope) from the ASPSP to invoke the Payments API resource against the ASPSP Resource Server. The ASPSP Resource server responds with the OrderId (and rest of data according specification).

HTTP Request example: POST /api/v1/payments/foreign

Header

```
POST / api/v1/payments/foreign HTTP/1.1
Content-Type: application/xml;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX
Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
PSU-Device-OS: win
PSU–IP-Address: 192.168.0.100
PSU-User-Agent: curl
```

Body

```
{
   "instructionIdentification": "9b766084-57de-48b2-be53-1bd2804ae0b7",
   "creationDateTime": "2019-02-16T11:59:20+01:00",
   "debtor":
   {
      "name": "John Doe",
      "account": "SK1475000000001109532451"
   },
   "creditor":
   {
      "name": "ABC Ltd.",
      "account": "SK7811000000001111111111",
      "address": "test",
      "country": "SK"
   },
   "bank":
   {
      "name": "ABC Ltd.",
      "swift": "KOMAX2X",
      "address": "test"
   },
   "instructedAmount":
   {
```

```
      "value": 1234.56,
      "currency": "EUR"
    },
    "endToEndIdentification": "/VS123/SS456/KS0308",
    "remittanceInformation": "Payment for a utility service.",
    "requestedExecutionDate": "2019-02-18",
    "purposeCode": "RINP",
    "fee": "SHA",
   "isExpress": true
}
```


HTTP Response example: POST /api/v1/payments/foreign

Body


```
{
"orderId": "aichz8i8z4c2ynabqtkymddhx2raw29zrzj",
"status": "RJCT",
"reasonCode": "MONY",
"reasonDescription": "MONY",
"statusDateTime": "2019-02-16T11:59:27+01:00"
}
```


ReasonCode : https://www.iso20022.org/15022/uhb/mt567-13-field-24b.htm

Error response example


```
{
   "Detail":"Exception of type 'System.Exception' was thrown.",
   "Error":"Mandatory parameter is missing"
}
```

Usage Example of PISP Operation: Request to cancel payment

STEP 2 : The PISP uses the Access Token (with PISP scope) from the ASPSP and the Order_Id for request to cancel payment. The ASPSP Resource server responds with the new Order_Id.

HTTP Request example: DELETE /api/v1/payments/{orderId}/rcp

Header

```
DELETE api/v1/payments/aichz8i8z4c2ynabqtkymddhx2raw29zrzj/rcp HTTP/1.1
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX
Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
PSU–IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5
```

HTTP Response example: DELETE /api/v1/payments/{orderId}/rcp

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
```

Body

```
{
"orderId": "6j74qbrt7bufixd2yw6jr3kgbvb7yd3dizf"
}
```

Usage Example of PISP Operation: Standard payment submission

Authorization

STEP 3: Payment authorization is initiated at the end of Step 2 by the PISP after the OrderId is generated by the ASPSP and returned to the PISP. This is used in a redirect across the PSU and ASPSP in Step 3 in order for the PSU to authorize the transaction.

Endpoint: POST

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/ api/v1/payments/submission
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/ api/v1/payments/submission

Request

Empty

Response

Attributes structure Level 1	Optionality	Type	Description
orderId	Mandatory	String [36]	OrderId is Unique reference, as assigned by the account servicing institution, to unambiguously identify the instruction.
status	Mandatory	Enum	Transaction status indicator is enumeration: ACTC (AcceptedTechnicalValidation) RJCT (Rejected)
reasonCode	Optional	Enum	ISO 20022 Status Reason Code
statusDateTime	Mandatory	DateTime	The date and time in RFC3339 format at which a particular action has been requested or executed.

HTTP Request example: POST /api/v1/payments/submission

Header

```
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX
Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
PSU–IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5
```

Body

```
{
"orderId":"aichz8i8z4c2ynabqtkymddhx2raw29zrzj",
"code" : “987456 ”

}
```

HTTP Response example: POST /api/v1/payments/submission

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
```

Body

```
{
```

```
"orderId":"aichz8i8z4c2ynabqtkymddhx2raw29zrzj",
"status": "RJCT",
"reasonCode": "MONY",
"statusDateTime": "2019-02-18T09:59:27+01:00"
}
```


Get token

Once the state and code validations have been confirmed as successful by use of the ID token, the PISP will proceed to obtain an Access Token from the ASPSP using the Authorization Code they now possess. The PISP will present its Authorization Code.

Endpoint: POST

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/token
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/token

Request

Attribute	Optionality	Type	Description
code	Mandatory	String	Authorization code returned from the code grant
redirect_uri	Mandatory	URL	The redirect URL matches the URL passed in the authentication request.
grant_type	Mandatory	String	Under the existing OAuth2 definition, this value will be the authorization_code if the TPP requested refresh_token.
code_verifier	Mandatory	String	Code_verifier used to generate code_challenge from a previous request with a minimum length of 43 characters and a maximum length of 128 characters
iban	Optional	string	Token for the IBAN – simple array separator comma

Response

Attribute	Optionality	Type	Description
access_token	Mandatory	String	Short-term (e.g. 3600 seconds, in some cases, onetime) token, which can be used to submit the initialized payment.
expires_in	Mandatory	Number	The remaining time to expiration of access_token - in seconds.
token_type	Mandatory	String	Type of token „Bearer“

HTTP Request example: POST /token

Header

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Authorization:Basic YTBiMjUyOTFmMDpCQmprazQ1c2Q3OGFkNDU0Z2RkZDg3MTJfNDU1
NWc1ZzVnNWdn // Basic BASE64(CLIENT_ID + ":" + CLIENT_SECRET)
```

Body

```
code=gCyAymoimg0L1bEI&
redirect_uri=https://www.paypay.sk/index&
grant_type=authorization_code&
code_verifier=yDWNhLugLI3BqUvXDYWE3DPrggSEyXCR
```

Note: All attributes are mandatory

The Access Token is required by the PISP in order to submit the Payment on behalf of the PSU. The payments scope should already be associated with the Authorization Code generated in the previous step.

HTTP Response example: POST /token

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
```

Body


```
{
"access_token":"1VVKPKO9IJUBFFXUKLW8JDVWM3B5XUBG",
"token_type":"bearer",
"expires_in": 300
}
```

Payment submission

The PISP has an Access Token which can be used to Create a Payment submission. The PISP must obtain the OrderId so that the Payment request is associated with the correct OrderId. OrderId is sourced from the **OrderId claim** of signed ID Token. The PISP will need to decode the ID Token JWT and locate the claim attribute associated with the OrderId. The PISP can now invoke the payment submissions endpoint to commit the Payment using the Access Token and OrderId in the payload of the request.

HTTP Request example: POST /api/v1/payments/submission

Header

```
POST /api/v1/payments/paymentSubmission HTTP/1.1
Content-Type: application/json;charset=UTF-8
Authorization: Bearer 1VVKPKO9IJUBFFXUKLW8JDVWM3B5XUBG
Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
PSU-Last-Logged-Time: 2019-02-16T11:56:32+01:00
PSU–IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3)
```

HTTP Response: POST /api/v1/payments/submission

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
```

Body

```
{
"orderId": "aichz8i8z4c2ynabqtkymddhx2raw29zrzj",
"status":"ACTC",
"statusDateTime": "2019-02-16T12:02:12+01:00"
}
```

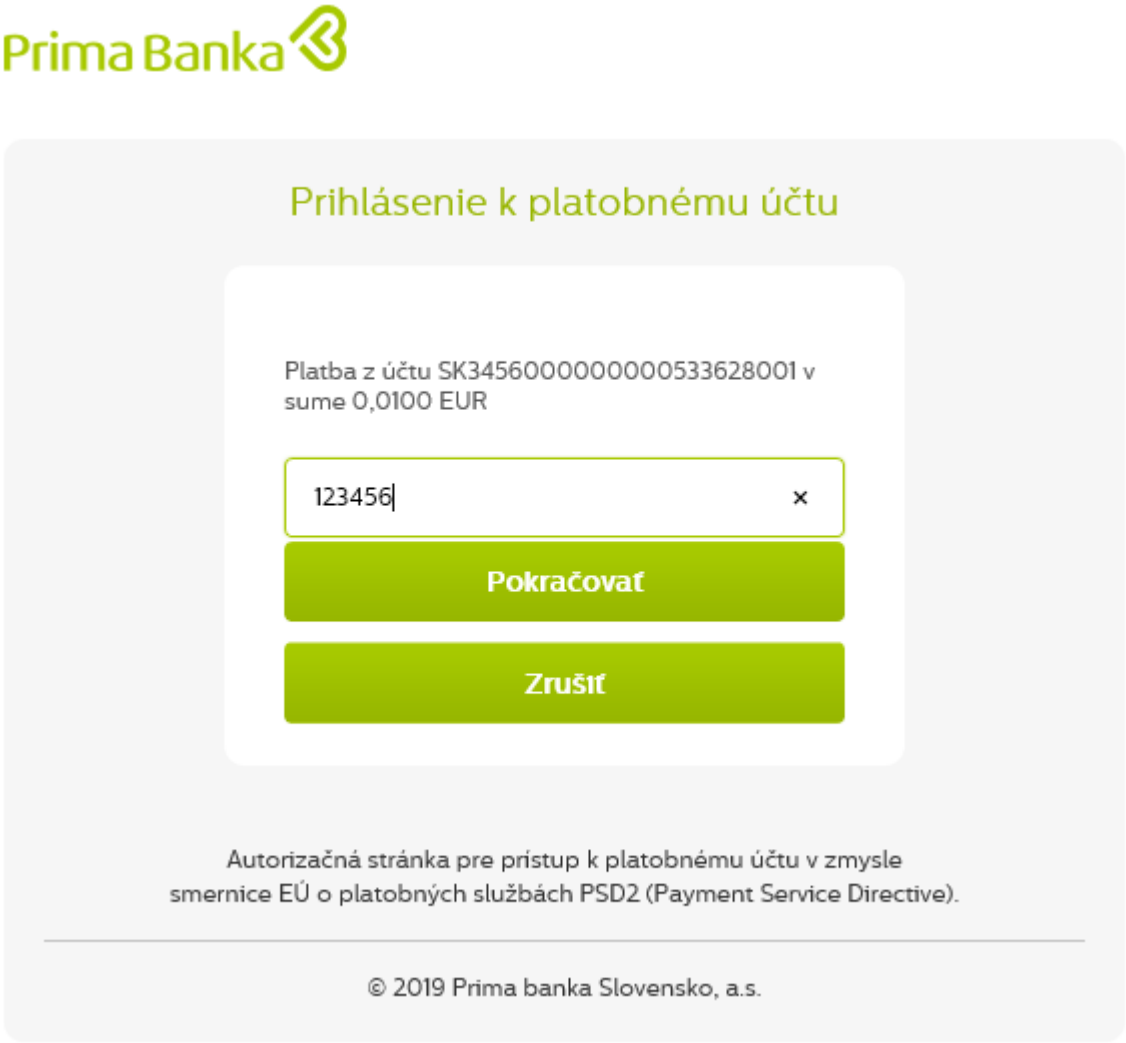
Payment submission via Prima bank web page

TPP can also choose to redirect the PSU to bank's web page for payment submition.

TPP only needs to send the original /authorize URL with added orderid=XXX, see example:

https://psd2.primabanka.sk/loginAPI/authorize?orderid=9d7d5482-f70a-4604-bdbe-58f9d058825b&response_type=code&scope=AISP&client_id=8FEF489748174876B059DE9386378467&state=VsH0TiAB1d3t7yR6VvD31DpUZEVrBXAQ&redirect_uri=http%3a%2f%2fwww.primabanka.sk&code_challenge=oO77bZ2WVsphzUSIihF1VUB2H0AE5auo8uP_x8axjW0&code_challenge_method=S256

The PSU will be redirected to the following page with appropriate payment information:


Prima Banka
Prihlásenie k platobnému účtu
Platba z účtu SK3456000000000533628001 v sume 0,0100 EUR
123456
Pokračovať
Zrušiť
Autorizačná stránka pre prístup k platobnému účtu v zmysle smernice EÚ o platobných službách PSD2 (Payment Service Directive).
© 2019 Prima banka Slovensko, a.s.

Usage Example of PISP Operation: Payment order status

STEP 4: The PISP can query for the status of a Payment submission by invoking the payment submissions using the known OrderId. This can use an existing access token with payments scope or the PISP can obtain a fresh access token by replaying the client credentials grant request as per Step 1 – Setup Single Payment Initiation.

HTTP Request example: GET/api/v1/payments/{orderId}/status

Header

```
GET /api/v1/payments/aichz8i8z4c2ynabqtkymddhx2raw29zrzj/status HTTP/1.1
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX
Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
PSU-Last-Logged-Time: 2019-02-16T11:56:32+01:00
PSU–IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3)
```

HTTP Response example:GET/api/v1/payments/{orderId}/status

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
```

Body

```
{
"orderId":"aichz8i8z4c2ynabqtkymddhx2raw29zrzj",
"status": "RJCT",
"reasonCode": "MONY",
"statusDateTime": "2019-02-18T09:59:27+01:00"
}
```

HTTP Request example: GET/api/v1/payments/{orderId}/statusBatch

Header

```
GET /api/v1/payments/ 710d9f2f-300f-474a-b6d8-a5fa6f04c6db/statusBatch HTTP/1.1
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX
Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
PSU-Last-Logged-Time: 2019-02-16T11:56:32+01:00
PSU–IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3)
```

HTTP Response example:GET/api/v1/payments/{orderId}/ statusBatch

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
```

Body

```
{
  "batchOrderId": "710d9f2f-300f-474a-b6d8-a5fa6f04c6db",
  "result": "ACTC",
  "paymentStatuses":
```

```
  {
    "orderId": "710d9f2f-300f-474a-b6d8-a5fa6f04c6db",
    "status": "ACTC",
    "statusDateTime": "2022-04-14T09:20:53+02:00",
    "reasonCode": null
  },
  {
    "orderId": "710d9f2f-300f-474a-b6d8-a5fa6f04c6db",
    "status": "ACTC",
    "statusDateTime": "2022-04-14T09:20:53+02:00",
    "reasonCode": null
  },
  {
    "orderId": "710d9f2f-300f-474a-b6d8-a5fa6f04c6db",
    "status": "ACTC",
    "statusDateTime": "2022-04-14T09:20:53+02:00",
    "reasonCode": null
  }
}
```

Payment Instrument Issuer Service Provider (PIISP)

Chapter defines list of services and alternative of flows provided for PIISPs and PISPs.
Prerequisites:

a) The TPP is registered for the PIISP or PISP role and valid PIISP or PISP scope
b) The TPP has been successfully authenticated
c) The TPP has presented its "OAuth2 Authorization Client Credential Grant" access token which allows the ASPSP to identify the TPP

Endpoints definition

In following sections describe technical definition of provided endpoints for PIISPs.

Endpoint	Method	Optionality	Description
/api/v1/accounts/balanceCheck	POST	Mandatory	Balance check – service provide information about sufficient balance with the yes/no answer

Standard header definition

Set of request and response headers for PIISP or PISP endpoints

Request header definition

Attribute	Optionality	Type	Description
Host	Mandatory	String	Domain name of the server and optional TCP port number
Content-Type	Mandatory	String	application/json or application/xml
Authorization	Mandatory	String	Authorization is defined in RFC 6750 - The OAuth 2.0 Authorization Framework: Bearer Token Usage
Request-ID	Mandatory	String	An unique identifier of a particular request message.
Correlation-ID	Optional	String	An unique correlation identifier correlates the request
Process-ID	Optional	String	Identifier of a business or technical process to what the set of requests and response pairs are organized (e.g. paging of transaction history should have same Process-ID).
PSU–IP-Address	Mandatory	String	Identifier of a customer's IP address from which he/she is connected to the TPP infrastructure. It might be in the format of IPv4 o IPv6 address. ASPSP shall indicate which values are acceptable.
PSU-Device-OS	Mandatory	String	A customer's device and/or operating system identification from which he/she is connected to the TPP infrastructure.
PSU-User-Agent	Mandatory	String	A customer's web browser of other client device identification from which he/she is connected to the TPP infrastructure. Agent header field of the http request between PSU and TPP.)
PSU-Geo-Location	Optional	String	The GPS coordinates of the current customer's location in the moment of connection to the TPP infrastructure. (Required GPS format)
PSU-Last-Logged-Time	Optional	DateTime	Last date and time when user was logged to TPP app (RFC3339 format)
PSU-Presence	Optional	Enum	The presence status of user (PSU) during an API call. The

			value of the parameter could be „true“ (PSU is present) or „false“ (PSU is not present).

Response header definition

Attribute	Optionality	Type	Description
Content-Type	Mandatory	String	application/json or application/xml
Response-ID	Mandatory	String	An unique identifier of a particular request message.
Correlation-ID	Optional	String	An unique correlation identifier correlates the request and the response messages as a pair especially useful for audit logs.
Process-ID	Optional	String	Identifier of a business or technical process to what the set of requests and response pairs are organized (e.g. paging of transaction history should have same Process-ID).

PIISP Operation: Balance check

The operation provides the resolution whether the balance of a bank customer's account identified by IBAN is sufficient for asked amount.

Endpoint: POST /api/v1/accounts/balanceCheck

Request

Attributes structure			Optionality	Type	Description
Level 1	Level 2	Level 3			
instructionIdentification			Mandatory	String	Technical identification of payment, generated by the PIISP
creationDateTime			Optional	DateTime	The date and time in RFC3339 format at which a particular action has been requested or executed.
iban			Mandatory	String [34]	International Bank Account Number (IBAN)
amount	value		Mandatory	Number Float [12.2]	Transaction amount value. Numeric value of the amount as a fractional number.
amount	currency		Mandatory	String [3]	Transaction amount currency. Formated in Alphabetic codes from ISO 4712.
relatedParties	tradingParty	identification	Optional	String [35]	Unique identification of a third party. For card transaction, this is ID of merchant.
relatedParties	tradingParty	name	Optional	String [140]	Name of a third party. For card transaction, this is the name of merchant.
relatedParties	tradingParty	address	Optional	String [70]	Merchant cummulative address identification usually containing concatenation of street name, street number, etc.
relatedParties	tradingParty	countryCode	Optional	String [2]	The two letter merchant country code adopted from ISO3166.
relatedParties	tradingParty	merchantCode	Optional	String [4]	A Merchant Category Code (MCC) coordinated by MasterCard and Visa.

references	chequeNumber	Optional	String [35]	For card transactions, this is the card number in format **** **** **** 1111
references	holderName	Optional	String[35]	Card holder name

Response (if no error)

Attributes structure Level 1	Optionality	Type	Description
response	Mandatory	Enum	response is enumeration: APPR (sufficient funds on the account) DECL (insufficient funds in the account)
dateTime	Mandatory	DateTime	The date and time in RFC3339 format at which a particular action has been requested or executed.

Error codes

HTTP Status	Error code	Description
400	parameter_missing	Mandatory Headers parameter is missing
400	parameter_invalid	Value of input parameter is not valid
500	server_error	Internal Server Error
403	forbidden	Authorization IBAN server error, client can not access requested IBAN
403	forbidden	Authorization PIISP server error
401	authorization_error	Authorization server error

Get token

The AISP will now possess the Authorization code and state parameter from the ASPSP. State parameter value must be identical as requested by AISP in the previous request otherwise, the response is invalid. AISP will proceed to obtain an Access Token from the ASPSP using the Authorization Code. The AISP will present its Authorization Code together with CLIENT_ID and CLIENT_SECRET in authorization header.
The Access Token is required by the AISP in order to access PSU Account information. The AISP scope should already be associated with the Authorization Code generated in the previous step.

Endpoint: POST

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/token
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/token

Request

Attribute	Optionality	Type	Description
code	Mandatory	String	Authorization code returned from the code grant
redirect_uri	Mandatory	URL	The redirect URL matches the URL passed in the authentication request.
grant_type	Mandatory	String	Under the existing OAuth2 definition, this value will be the authorization_code if the TPP requested refresh_token.
code_verifier	Mandatory	String	Code_verifier used to generate code_challenge from a previous request with a minimum length of 43 characters and a maximum length of 128 characters
iban	Optional	string	Token for the IBAN – simple array separator comma

Response

Attribute	Optionality	Type	Description
access_token	Mandatory	String	Short-term 300 seconds token, which can be reissued using refresh_token. This token serves to authorize TPP request on ASPSP API.
expires_in	Mandatory	Number	The remaining time to expiration of access_token - in seconds.
refresh_token	Optional	String	Long-term token 90 days issued as a replacement for authorization_code.
token_type	Mandatory	String	Type of token „Bearer“
scope	Optional	String	List of permissions separated by the space for which the token is issued.

HTTP Request example: POST /token

Header

```
POST /token HTTP/1.1
Content-Type:      application/x-www-form-urlencoded
Authorization:     Basic YTBiMjUyOTFmMDpCQmprazQ1c2Q3OGFkNDU0Z2RkZDg3MTJfNDU
1NWc1ZzVnNWdn //Basic BASE64(CLIENT_ID + ":" + CLIENT_SECRET)
```

Body

```
grant_type=authorization_code&code=03ab43f0-0257-477c-b06c-
ab61b2b1080d&redirect_uri=https:\/\/ib.primabanka.sk\/ib\/default.aspx&code_verifier=04e10457-7e67-4d9e-9b49-
121f7eee5396&iban = SK12 5600 0000 1324 4560
```

HTTP Response example: POST /token

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
```

Body


```
{
"access_token":" IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX",
"token_type":"bearer",
"expires_in":300,
"refresh_token":" be9eef9b0af42c674d0b1c1128c37c2g"
}
```

Usage Example of PIISP Operation: Balance check

Endpoints

LIVE	https://psd2.primabanka.sk/ServiceAPI/API.svc/api/v1/accounts/BalanceCheck
DEV	https://ib.primabanka.sk/ServiceAPI_test/API.svc/api/v1/accounts/BalanceCheck

HTTP Request example: POST /api/v1/accounts/balanceCheck

Header

```
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX
Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
PSU–IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3)
```

Body


```
 {  "instructionIdentification":null,
   "creationDateTime":null,
   "iban":" SK0431000000002333363431",
   "amount":
   {
      "value":5,
      "currency":"EUR"
   },
   "relatedParties":null,
   "references":
   {
      "chequeNumber":"1234567890",
      "holderName":"Peter Test"
   }
}
```


HTTP Response example: POST /api/v1/accounts/balanceCheck

Headers

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
Correlation-ID: 292163f5-4eee-4447-9292-5672fdf0013b
Process-ID: 4b88bf95-e129-42b8-a17d-1d2379810fbe
```

Body


```
{
   "response":"DECL",
   "dateTime":"2019-07-12T15:44:53"
}
```