OpenAPI
(ex-Budapest Bank)

Open Banking interface (ex-Budapest Bank OpenAPI)

Introduction & common parts

Version 2.2

Valid from 2023.05.01.

Please visit openbanking.mbhbank.hu to get more information on MBH Bank Open Banking, including operational information on the live service.

Contact

In case you have a question or any problem regarding our services, please send us a message using the following email address and we will reply in 2 working days:

openbanking@mbhbank.hu

1 Introduction

This documentation describes MBH Bank Open Banking interface. The documentation is divided into four main parts:

1. **MBH Bank Open Banking interface_Version 2.2**: introduction, summary of the services, common structures, error handling
2. **MBH Bank Open Banking interface – AccountInfo_Version 3.2:** description of account information operations
3. **MBH Bank Open Banking interface – PaymentInitiation_Version 2.2:** description of payment initiation operations
4. **MBH Bank Open Banking interface – TechnicalRegistration_Version 2.2:** descriptions of first TPP interactions with the Bank

2 Definitions

Name	Definition
TPP	Based on the Payment Service Directive (PSD) 2 – 2015/2366, a third party provider who uses MBH Bank's Open Banking interface (ex-Budapest Bank OpenAPI).
Account	Through Open Banking interface, we handle and support all payment accounts identified by IBAN account number, held with MBH Bank and accessible through one of our online banking channels, including bank accounts, retail and corporate bank / current accounts.
PSU	Payment service user is the owner of the accounts held by MBH Bank and gives accreditations to the TPP in order to access his or her accounts information or initiate payment from these accounts.
AISP	Account information service provider who can initiate read only service calls to PSU's accounts.
PISP	Payment initiation service provider who can initiate read only service calls and payment requests to PSU's accounts.
PIISP	Payment instrument issuer service providers who can initiate confirmation of funds service calls to PSU's accounts.
SCA	Strong customer authentication.
Consent token	Unique id of the approved access (consent) to a given account linked to the TPP who initiated the request. This token is generated by MBH Bank, based on PSU's SCA.

3   Supported Services

The following table shows the scope of Open Banking interface.

Service Name	Version	Operation Name	Description	Role of the TPP	SCA Approval required by PSU(s)	Consent token required
TechnicalRegistration	1	initiateTechnicalRegistration	TPP's first interaction with the Bank. Must share basic information, including the role of TPP and the public key used to TLS and sign messages. This API has to be called periodically aligned with TPP's certificate renewal process. In case of successful technical registration, the Bank grants access to the defined services (see in roles column).	AISP, PISP, PIISP	no	no
TechnicalRegistration	1	getRegistrationStatus	Inquiry status of the registration.	AISP, PISP, PIISP	no	no
AccountInfo	1	initiateConsentRequest	Get access to account information services for a limited period. It requires to complete approval process on the Bank's side. During consent request, the Bank determines the selection of accessible services based on TPP's technical registration.	AISP, PISP, PIISP	yes	no
AccountInfo	1	getStatus	Inquiry status of approval or order completion process. In case of consent, if the initiateConsentRequest was approved it will contain the consent token in the response message.	AISP, PISP, PIISP	no	no
AccountInfo	1	getBalance	Inquiry available balance of the given account.	AISP	no	yes
AccountInfo	1	getConfirmationOfFunds	Inquiry fund availability of the given account and amount.	PISP, PIISP	no	yes
AccountInfo	1	getTransactionsList	Inquiry transaction history up to the last 90 days.	AISP	no	yes

AccountInfo	1	getStandingOrderList	Inquiry standing order list.	AISP	no	yes
AccountInfo	1	revokeApproval	Revoke the given approval process.	AISP, PISP, PIISP	no	no
PaymentInitiation	1	initiateCreditTransfer	Initiate a new transfer order.	PISP	yes	no
PaymentInitiation	1	createStandingOrderHUF	Initiate a new HUF standing order.	PISP	yes	no
PaymentInitiation	1	deleteStandingOrder	Delete an existing standing order.	PISP	yes	no
PaymentInitiation	1	revokeCreditTransfer	Revoke / recall an order	PISP	yes	no

All operations are available via SOAP over HTTPS interface.

4 Security

The Open Banking interface requires security in 2 levels:

1. Transport layer security: TLS v1.2 must be applied with one-way authentication.
2. Message level signature - Web Services Security X.509 Certificate Token Profile. During webservice calls, TPPs must sign every message by their PSD2 certificate. Requirements of ws-security policy is defined in wsdls.

TPP's certificate has to be sent in initiateTechnicalRegistration request. Same certificate has to be used for both levels of security.

MBH Bank's certificate will be published on Open Banking website.

5 Operation methods

5.1 Without SCA Approval

While calling these types of services the TPP will get synchronous answer in the service response with the result of the request, such as available balance, transaction history, status etc.
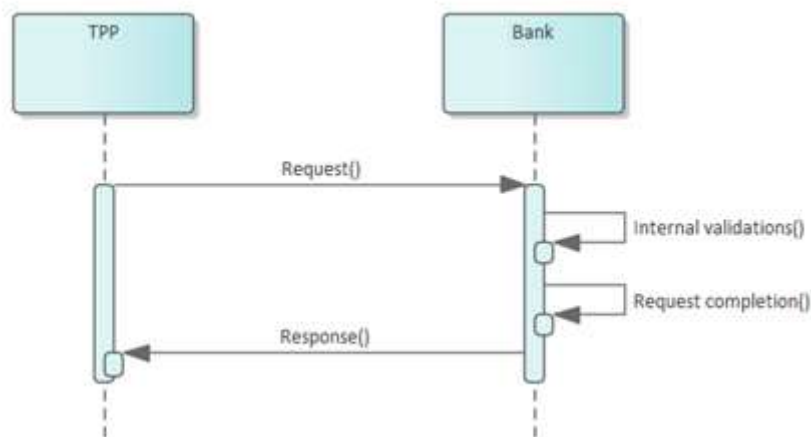
5.2   With SCA Approval

While calling these types of services the TPP will get synchronous answer in the service response with receiving acknowledgement. In order to the TPP gather additional information from the request's further lifecycle, must call the getStatus service.
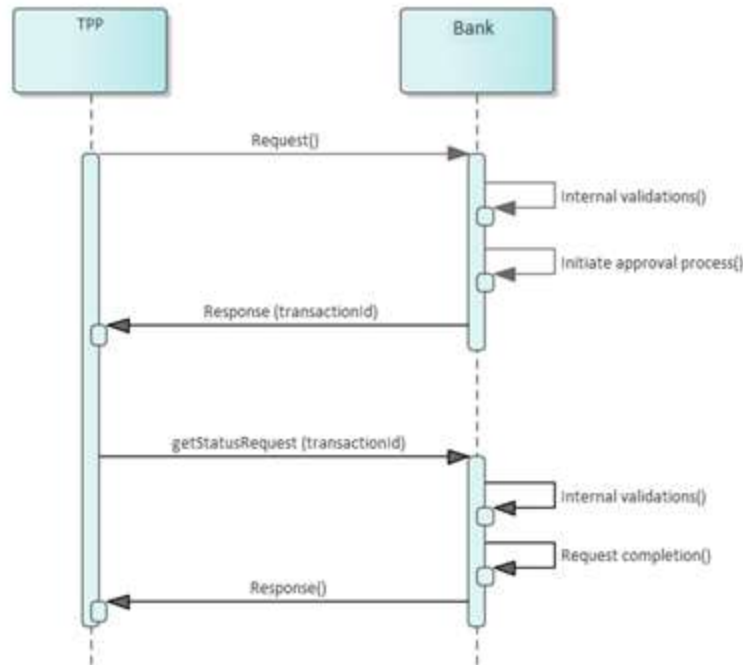
The Bank applies decoupled approach during SCA, however PSU(s) (after a successful login) can reach the approval list page in MBH Netbank (previously BB) via a direct link. This link will be included in the receiving acknowledgement response message, so TPP can redirect the PSU. The TPP must gather information from getStatus service (the Bank won't redirect back to TPP's side with approval result).

6   Common structures

This chapter describes the APIs common parts such as soap header structure and response codes.

6.1   Common Header Fields

All requests share a common header. Most fields in the header are the same. This is true for both the request and the response. Therefore the header fields are listed here separately.

6.1.1   Request Header Common Fields

The request header fields are listed below:

#	Name	Type	Num	Description
1	Head	HEADTYPE	[1..1]	The header of the request
1.1	RequestDate	xs:date	[1..1]	The date of the request sent by the TPP in UTC
1.2	RequestTime	xs:time	[1..1]	The time of the request sent by the TPP in UTC
1.3	RequestId	RequestId Type	[1..1]	Request identifier. Should be unique with the given ApplicationId. String, maximum length: 36 characters.

#				
1.4	ApplicationId	AppName HeaderType	[0..1]	The id of the application approved by the bank during the app registration process. String, maximum length: 36 characters.
1.10	ClientIPAddress	ClientIPAddressType	[0..1]	The IP address of the PSU. String, maximum length 45 characters, ###IPv6 address format
1.11	ClientGeoLocation	ClientGeoLocationType	[0..1]	The geolocation of the PSU as detected by the TPP. String, maximum 50 characters.

All fields not listed in the table are ignored if sent in request.

6.1.2   Response Header Common Fields

The response header fields are listed below:

#	Name	Type	Num	Description
1	Head	HEADTYPE Response	[1..1]	The header of the response
1.1	RequestDate	xs:date	[1..1]	Original RequestDate
1.2	RequestTime	xs:time	[1..1]	Original RequestTime
1.3	RequestId	RequestId Type	[1..1]	Original RequestId
1.4	ResponseDate	xs:date	[1..1]	The date of the response sent by the API in UTC
1.5	ResponseTime	xs:time	[1..1]	The time of the response sent by the API in UTC
1.6	ResponseCode	tns:ResponseCodeType	[1..1]	The response code from the API see: 6.2 Response Codes
1.7	ResponseDescription	ResponseDescriptionType	[1..1]	The description belonging to the response code. String, maximum length 200 characters.

6.2   Response Codes

The service has the following response codes. The message codes are categorized:

- "0" – Normal run, no error, no warning
- "4" – Normal run with warning, for example the result is too big
- "8" – Business error
- "C" – Technical error

Response message details:

SOAP message body content	responseCode	responseDescription	faultcode	faultstring	errorCode	errorDescription
normal response	0					
normal response	4	There are more results in the backend system, please change input parameters to get all data				
fault	C	Technical error	soapenv:server	Security error	C001	Certificate expired
fault	C	Technical error	soapenv:server	Security error	C002	Certificate blocked or revoked
fault	C	Technical error	soapenv:server	Security error	C003	The contents of the signature certificate are not matching

fault						PSD2 attribute requirements
fault	C	Technical error	soapenv:server	Security error	C004	Signature check failed – signature is invalid
fault	C	Technical error	soapenv:server	Error occurred at message validation	C005	Format of certain request fields are not matching the requirements
fault	C	Technical error	soapenv:server	Error occurred at message validation	C006	Invalid IBAN, checksum not match
fault	C	Technical error	soapenv:server	Error occurred at role validation	C007	TPP doesn't have right to call the requested operation
fault	C	Technical error	soapenv:server	Error occurred at role validation	C008	Invalid consent
fault	C	Technical error	soapenv:server	Error occurred at internal message processing	C009	Internal system error
fault	8	Business error	soapenv:server	Error occurred at account validation	8001	Requested account is not scope of PSD2 openAPI
fault	8	Business error	soapenv:server	Error occurred at account validation	8002	Requested account is blocked
fault	8	Business error	soapenv:server	Error occurred at account validation	8003	SCA requirement cannot be satisfied, one or more approver disabled openAPI channel
fault	8	Business error	soapenv:server	Error occurred at internal message processing	8004	Internal business error
fault	8	Business error	soapenv:server	Error occurred at internal message processing	8005	Revoke not allowed
fault	8	Business error	soapenv:server	Error occurred at validation	8006	Transaction not exists
fault	8	Business error	soapenv:server	Error occurred at validation	8007	Transaction data failed