BUDAPEST BANK

Open Banking interface

Account information services

Version 1.0

Please visit openbanking.budapestbank.hu to get more information on Budapest Bank OpenBanking, 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@budapestbank.hu

1 Introduction

This document describes the AccountInfo services provided by Budapest Bank Open Banking interface. These services are for AISPs to get information on the PSU's account. All service access requires approval from the target account's approvers. The approval results in a consent, which provides access to the account services for a maximum of 90 days. In these 90 days the TPP does not need an approval to use the services.

2 Supported Operations

Operation Name	Description	Role of the TPP	SCA Approval required by PSU(s)	Consent token required
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
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
getBalance	Inquiry available balance of the given account.	AISP, PISP	no	yes
getConfirmationOfFunds	Inquiry fund availability of the given account and amount.	PIISP	no	yes
getTransactionsList	Inquiry transaction history up to the last 90 days.	AISP, PISP	no	yes
revokeApproval	Revoke the given approval process.	AISP, PISP, PIISP	no	no

All operations are available via SOAP over HTTPS interface.

3 Consent handling

The usage of consent based services requires an approval from the target account's approvers. The TPP needs to send an initiateConsentRequest request. This generates approval tasks that are sent to

the users who have approval rights over the given account. If the request is approved, it results in a consent. TPP must gather initiateConsentRequest's status via getStatus request. If the request is approved the getStatus response will contain the new consent ID. The TPP can use the consent ID to use AccountInfo services without further approval from the PSU.

Consents are valid for all AccountInfo services that the TPP has access to. The services that the consent is valid for are returned to the TPP.

The consent is valid for maximum 90 days from the time of the approval. For example if a consent is approved on 2019-09-14T13:00:01+0200 then it is valid for 2019-12-13T13:00:01+0100. In this period the TPP does not need to ask for the PSU's approval to request information about the account.

4   Service Definitions

This chapter provides information about the AccountInfo_V1 services. The base definition of the service can be found in the AccountInfo_V1.wsdl and referred xsd files. This document provides more details on input and output fields. Requirements or restrictions that cannot be put into the WSDL can be found here.

For input fields only the fields listed in this service definition are used. All other fields that may be the service definition wsdl will be ignored by the API.

For output fields the fields presented in this service description are the only fields that are guaranteed to return relevant values. Values returned in other fields (if any) should be used as informational data.

The c: namespace here refers to the bbcommon: namespace in the WSDL definition.

All business date/datetime fields are Budapest local time CET or CDT if timezone is not present. All header fields are UTC timezone if not marked otherwise.

4.1 getBalance Operation

4.1.1 Description

The `GetBalance` service is used to retrieve the balance of a given account. The TPP must have a valid consent to successfully use this service. If the TPP does not have a valid consent, the `initiateConsentRequest` service should be called first for the given IBAN and the `GetBalance` service to acquire Consent. If the TPP does not have a valid Consent, the result of the request is a fault.

The service is eligible for a maximum of 90 day consent.

4.1.2 Input Fields

#		Name	Type	Num	Description
1		consent	complex type	[1..1]	Request target
	1.1	type	c:consentType	[1..1]	The type of the consent, "IBAN" for this service
	1.2	target	c:targetType	[1..1]	The IBAN account number
	1.3	id	c:consentIdType	[1..1]	The id of the consent

4.1.3 Output Fields:

#			Name	Type	Num	Description
1			consent	complex type	[1..1]	Request target
	1.1		type	c:consentType	[1..1]	The type of the consent, "IBAN" for this service
	1.2		target	c:ibanType	[1..1]	The IBAN account number
	1.3		id	c:consentIdType	[1..1]	The id of the consent
2			transactionId	c:transactionType	[1..1]	The transaction id given by the API.
3			balances	complex type	[1..1]	The balances of the requested account
	3.1		balance	complex type	[1..*]	Balance object to hold the balance of the given account. One row returned per balance type
		3.1.1	balanceType	c: balanceTypeType	[1..1]	Type of the given balance value. At the moment the only type available is "availableBalance"
		3.1.2	amount	c:amountType	[1..1]	Balance amount
		3.1.3	currency	c: currencyType	[1..1]	ISO code of the account's currency. 'HUF' in case of Hungarian Forint account.

4.2 getConfirmationOfFunds Operation

4.2.1 Description

The purpose of this service is to ask for a confirmation from the bank that the required funds are available on the target account at the time of the request. This is an informational service only, the service does not place a hold on the amount requested.

The TPP must have a valid consent to successfully use this service. If the TPP does not have a valid consent, the `initiateConsentRequest` service should be called first for the given IBAN and the `GetConfirmationOfFunds` service to acquire Consent. If the TPP does not have a valid Consent, the result of the request is a fault.

The service is eligible for a maximum of 90 day consent.

4.2.2 Input Fields

getConfirmationOfFunds Operation - Input Fields				
#	**Name**	**Type**	**Num**	**Description**
1	consent	complex type	[1..1]	Request target
1.1	type	c:consentType	[1..1]	The type of the consent, "IBAN" for this service
1.2	target	c:targetType	[1..1]	The IBAN account number
1.3	id	c:consentIdType	[1..1]	The id of the consent
2	instructedAmount	complex type	[1..1]	The funds the service is asking for
2.1	amount	c:amountType	[1..1]	The instructed amount
2.2	currency	c:currencyType	[1..1]	ISO code of the instructed amount's currency, 'HUF' in case of Hungarian Forint. Instructed amount's and account's currency have to be matched, otherwise the request will be declined.

4.2.3 Output Fields

getConfirmationOfFunds Operation - Output Fields				
#	**Name**	**Type**	**Num**	**Description**
1	consent	complex type	[1..1]	Request target
1.1	type	c:consentType	[1..1]	The type of the consent, "IBAN" for this service
1.2	target	c:ibanType	[1..1]	The IBAN account number

getConfirmationOfFunds Operation - Output Fields				
#	Name	Type	Num	Description
1.3	id	c:consentIdType	[1..1]	The id of the consent
2	transactionId	c:transactionType	[1..1]	The transaction id given by the API.
3	fundsAvailable	xs:boolean	[1..1]	Is the required funds available on the target account.
4	queryDate	xs:dateTime	[1..1]	The date and time of the fund check. Time zone is CET (or CDT).

4.3 getStandingOrderList Operation

4.3.1 Description

The purpose of this service is to retrieve the list of standing orders for the account.

The TPP must have a valid consent to successfully use this service. If the TPP does not have a valid consent, the `initiateConsentRequest` service should be called first for the given IBAN and the `GetStandingOrderList` service to acquire Consent. If the TPP does not have a valid Consent, the result of the request is a fault.

The service is eligible for a maximum of 90 day consent.

4.3.2 Input Fields

getStandingOrderList Operation - Input Fields				
#	Name	Type	Num	Description
1	consent	complex type	[1..1]	Request target
1.1	type	c:consentType	[1..1]	The type of the consent, "IBAN" for this service
1.2	target	c:ibanType	[1..1]	The IBAN account number
1.3	id	c:consentIdType	[1..1]	The id of the consent

4.3.3 Output Fields

#			Name	Type	Num	Description
getStandingOrderList Operation - Output Fields						
1			consent	complex type	[1..1]	Request target
	1.1		type	c:consentType	[1..1]	The type of the consent, "IBAN" for tis service
	1.2		target	c:ibanType	[1..1]	The IBAN account number
	1.3		id	c:consentIdType	[1..1]	The id of the consent
2			transactionId	c:transactionType	[1..1]	The transaction id given by the API.
3			standingOrders	complex type	[0..1]	The element to hold the list of standing orders
	3.1*		standingOrder	complex type	[0..*]	The standing order data. Can be zero or more elements.
		3.1.1	startDate	xs:date	[1..1]	First payment date
		3.1.2	endDate	xs:date	[0..1]	Last payment date, if there is any
		3.1.3	nextDate	xs:date	[0..1]	Next payment date if there is any
		3.1.4	lastDate	xs:date	[0..1]	Last processed payment date
		3.1.5	instructedAmount	complex type	[0..1]	The amount of the standing order
		3.1.5.1	amount	c:amountHUFType	[1..1]	Instructed amount
		3.1.5.2	currency	c:currencyHUFType	[1..1]	ISO code of the instructed amount's currency.
		3.1.6	threshold	xs:string	[0..1]	Threshold of the order
		3.1.7	creditorName	c:creditorName	[0..1]	The name of the creditor
		3.1.8	creditorAccount	xs:string	[0..1]	The creditor account for the standing order
		3.1.9	remittanceInformationUnstructured	c:remittanceInformationUnstructuredType	[0..1]	The remittance information of the standing order.
		3.1.10	executionRule	c: executionRuleType	[0..1]	The execution rule for the standing order. Possible values are: - "until revoked" - "to date" - "until a number of fulfilled orders"
		3.1.11	standingOrderId	c: standingOrderIdType	[0..1]	The id of the standing order.
		3.1.12	processedPaymentCount	xs:string	[0..1]	The count of processed payments
		3.1.13	processedPaymentQuantity	xs:string	[0..1]	Total amount of processed payments

getStandingOrderList Operation - Output Fields				
#	Name	Type	Num	Description
3.1.14	newPaymentAmount	xs:string	[0..1]	New payment value after modification
3.1.15	newPaymentDate	xs:string	[0..1]	New payment date after modification
3.1.16	status	xs:string	[0..1]	The status of the standing order. The possible values are: - 'A' -> active - 'R' -> creation in progress - 'M' -> modification in progress - 'L' -> closing in progress - 'D' -> closed - 'H' -> archive - 'T' -> deleted - 'E' -> error

4.4   getStatus Operation

4.4.1   Description

The purpose of this service is to retrieve the status of the approval process and the status of the payment order.

4.4.2   Input Fields

getStatus Operation - Input Fields				
#	Name	Type	Num	Description
1	transactionId	c:tansactionIdType	[1..1]	The id of the transaction the TPP is asking for.

4.4.3 Output Fields

#		Name		Type	Num	Description
1		transactionId		c:tansactionIdType	[1..1]	The transaction id given by the API.
2		status		c:statusType	[1..1]	The status of the consent request?. the possible values are: - "received" - "pending" - "approved" - "inprogress" - "completed" - "declined"
3		consent		complex type	[1..1]	Request target (choice)
	3.1	id		c:consentIdType	[0..1]	The id of the consent (present if the related initiateConsentRequest was approved)
	3.2	expiration		xs:dateTime	[0..1]	Expiration of the consent (present if the related initiateConsentRequest was approved)
	3.3	grantedAPIs		complex type	[1..1]	List of available services related to consent
		3.3.1	grantedAPI	complex type	[1..*]	Service details
		3.3.3.1	service	xs:string	[1..1]	Service name, example: AccountInfo
		3.3.3.2	serviceVersion	xs:string	[1..1]	Service version, example: 1
		3.3.3.3	operation	xs:string	[1..1]	Operation name, example: getBalance
4		order		complex type	[1..1]	The order data (choice)
	4.1	type		c:orderType	[1..1]	The type of the order. The possible values are: - PromtHUF -> Instant payment - NormalHUF -> Usual HUF transaction - FX -> Foreign transfer
	4.2	transferStatus		Q1: CustomerPaymentStatusReportV09	[0..1]	Transaction status information for initiateCreditTransfer requests. Included when approval is completed and the first status update received from transaction processing system
		4.2.1	GrpHdr	GroupHeader74	[1..1]	
		4.2.1.1	MsgId	Max35Text	[1..1]	The message id generated by the Bank's payment system.

getStatus Operation - Output Fields

#	Name		Type		Num	Description
4.2.1.2		CreDtTm		ISODateTime	[1..1]	The creation time of the message created by the Bank's payment system
		InitgPty		PartyIdentification125	[0..1]	The initiating party of the message
4.2.1.1.1		Nm		Max140Text	[1..1]	Always "BB PE"
4.2.2	OrgnlGrpInfAndSts			OriginalGroupHeader13	[1..1]	
4.2.2.1		OrgnlMsgId		Max35Text	[1..1]	The message id of the original message
4.2.2.2		OrgnlMsgNmId		Max35Text	[1..1]	The type of the message that initiated the transfer
4.2.2.3		OrgnlCreDtTm		ISODateTime	[0..1]	The create time sent with the original message
4.2.2.4		OrgnlNbOfTxs		Max15NumericText	[0..1]	The number of transactions in the original message
4.2.2.5		OrgnlCtrlSum		DecimalNumber	[0..1]	The control sum field form the original message
4.2.2.6		GrpSts		ExternalPaymentGroupStatus1Code	[0..1]	Status of the package received. Possible values: - "RCVD" The payment initiation has been received by the payment system . "ACCP" the validation of the message is complete, parts of the package has been accepted - "RJCT" the whole package has been rejected
4.2.2.7		StsRsnInf		StatusReasonInformation11	[0..1]	Mandatory if the GrpSts is "RJCT"
4.2.2.7.1		Rsn		StatusReason6Choice	[0..1]	
4.2.2.7.1.1		Cd		ExternalStatusReason1Code	[1..1]	External status code
4.2.2.7.2		AddtlInf		Max105Text	[0..*]	In case of error, the description of the error message.
4.2.3	OrgnlPmtInfAndSts			OriginalPaymentInstruction27	[0..*]	Contains information about the first level payment block.
4.2.2.9		OrgnlPmtInfId		Max35Text	[1..1]	The id of the original payment block
4.2.2.10		OrgnlNbOfTxs		Max15NumericText	[0..1]	The number of transactions in the original messge
4.2.2.11		OrgnlCtrlSum		DecimalNumber	[0..1]	The control sum in the original message
4.2.2.12		PmtInfSts		ExternalPaymentGroupStatus1Code	[0..1]	External payment status group code. Possible values: . "ACCP" the validation of the message is complete, parts of the package has been accepted - "RJCT" the whole package has been rejected
4.2.2.13		StsRsnInf		StatusReasonInformation11	[0..1]	Mandatory if the PmtInfSts is "RJCT"

#	Name		Type	Num	Description
	getStatus Operation - Output Fields				
4.2.2.13.1	Rsn		StatusReason6Choice	[0..1]	
	4.2.2.13.1.1	Cd	ExternalStatusReason1Code	[1..1]	External status code
4.2.2.13.2	AddtlInf		Max105Text	[0..*]	In case of error, the description of the error message.
4.2.2.14	TxInfAndSts		PaymentTransaction92	[0..*]	Status information of the original orders
4.2.2.13.2	OrgnlInstrId		Max35Text	[0..1]	The id of the original order
4.2.2.13.3	OrgnlEndToEndId		Max35Text	[0..1]	
4.2.2.13.4	TxSts		ExternalPaymentTransactionStatus1Code	[0..1]	Status of the package received. Possible values: . "ACCP" the validation of the message is complete, parts of the package has been accepted - "ACSC" The payment is booked on the account. - "RJCT" the whole package has been rejected
4.2.2.13.5	StsRsnInf		StatusReasonInformation11	[0..*]	Mandatory if the TxSts is "RJCT"
	4.2.2.13.5.1	Rsn	StatusReason6Choice	[0..1]	
	4.2.2.13.5.1.1	Cd	ExternalStatusReason1Code	[1..1]	External status code
	4.2.2.13.5.2	AddtlInf	Max105Text	[0..*]	In case of error, the description of the error message.
4.2.2.13.6	AccptncDtTm		ISODateTime	[0..1]	The booking date on the account. Mandatory if the TxSts is "ACSC".
4.2.2.13.7	AcctSvcrRef		Max35Text	[0..1]	The reference number of the booking. Mandatory if the TxSts is "ACSC".
4.2.2.13.8	ClrSysRef		Max35Text	[0..1]	The international reference number of the order. Mandatory if the TxSts is "ACSC" and the order is sent through SWIFT.

4.5 getTransactionList Operation

4.5.1 Description
This service retrieves the transaction history of the account.

The TPP must have a valid consent to successfully use this service. If the TPP does not have a valid consent, the `initiateConsentRequest` service should be called first for the given IBAN and the `getTransactionList` service to acquire Consent. If the TPP does not have a valid Consent, the result of the request is a fault.

The service is eligible for a maximum of 90 day consent.

4.5.2   Input Fields

#		Name	Type		Num	Description
1		consent	complex type		[1..1]	Request target
	1.1	type	c:consentType		[1..1]	The type of the consent. The only values accepted is "IBAN".
	1.2	target	c:ibanType		[1..1]	The IBAN account number
	1.3	id	c:consentIdType		[1..1]	The id of the consent
2		resultCount	c:resultCountType		[1..1]	The maximum returned result requested. Integer minimum value is 1 maximum value is 1000
3		startDate	xs:date		[1..1]	Start date of the query period. Earliest date is: today-90 days' date
4		endDate	xs:date		[1..1]	End date of the query period. Latest date is current date.

getTransactionList Operation - Input Fields

4.5.3   Output Fields

getTransactionList Operation - Output Fields

#			Name	Type	Num	Description
1			consent	complex type	[1..1]	Request target
	1.1		type	c:consentType	[1..1]	The type of the consent, "IBAN" for this service
	1.2		target	c:ibanType	[1..1]	The IBAN account number
	1.3		id	c:consentIdType	[1..1]	The id of the consent
2			transactionId	c:transactionIdType	[1..1]	The transaction id given by the API.
3			transactions	complex type	[0..1]	List of transactions
	3		transaction	complex type	[0..*]	Transaction history element
	3.1		valueDate	xs:date	[0..1]	Value date of the transaction
	3.2		cardUsageDate	xs:date	[0..1]	In case of card transaction the date of the card usage
	3.3		cardNumber	xs:string	[0..1]	In case of card transaction the card number in masked format
	3.4		cardUsageLocation	xs:string	[0..1]	Location of card transaction
	3.5		creditorAccount	xs:string	[0..1]	Account of the creditor

getTransactionList Operation - Output Fields				
#	**Name**	**Type**	**Num**	**Description**
3.6	creditorName	xs:string	[0..1]	Name of the creditor
3.7	bookingDate	xs:date	[0..1]	Booking date of the transaction
3.8	remittanceInformationUnstructured	xs:string	[0..1]	Remittance information of the transaction
3.9	actuonDescHU	xs:string	[0..1]	Description of the transaction type in Hungarian. Examples: 'Pénztári befizetés', 'Internetbank átutalás', 'Fogadott tétel', 'Vásárlás', etc.
3.10	actuonDescEN	xs:string	[0..1]	Description of the transaction type in English. Examples: 'Cashier pay-in', 'Internetbank remittance', 'Received item', 'Purchase', etc.
3.11	bookingReferenceNumber	xs:string	[0..1]	Booking reference number of the transaction
3.12	accountCurrency	xs:string	[0..1]	ISO code of the account's currency. 'HUF' in case of Hungarian Forint account.
3.13	transactionChannelDescription	xs:string	[0..1]	Description of the origination transaction channel in Hungarian. Examples: 'Fiok', 'NetBank', 'Bankon kivulrol erkezo', 'Kartya', etc.
3.14	transactionAmount	complex type	[0..1]	The amount of the transaction
3.14.1	currency	xs:string	[0..1]	ISO code of the account's currency.
3.14.2	value	xs:string	[0..1]	Amount of the transaction with negative sign in case of debit transaction

4.6 initiateConsentRequest Operation

4.6.1 Description

The `initiateConsentRequest` service is used to start an approval for the given IBAN for the given TPP. The Consent acquired this way can be used by the TPP to get information on the target account. All account related inquiry services can be used with a valid consent.

4.6.2 Input Fields

#	**Name**	**Type**	**Num**	**Description**
1	consent	complex type	[1..1]	Request target
1.1	type	c:consentType	[1..1]	The type of the consent, "IBAN" for tis service
1.2	target	c:targetType	[1..1]	The IBAN account number the Consent is requested for.
1.3	validityPeriod	c:validityperiodType	[1..1]	The requested validity period for the consent in days. Integer, value range is [1..90]

4.6.3   Output Fields

#	Name	Type	Num	Description
1	transactionId	c:transactionIdType	1..1]	The transaction id given by the API.
1	consent	complex type	[1..1]	Request target
1.1	type	c:consentType	[1..1]	The type of the consent, "IBAN" for this service
1.2	target	c:ibanType	[1..1]	The IBAN account number
1.4	status	c:statusType	[1..1]	The status of the consent request

4.7   revokeApproval Operation

4.7.1   Description

The `RevokeApproval` service is used to cancel an ongoing approval related to consent or order.

In the case of approval, the workflow is stopped, any approvals are discarded. The service can be used to cancel an order that has been approved but not fulfilled yet. In this case the order is not going to be fulfilled.

4.7.2   Input Fields

#	Name	Type	Num	Description
1	transactionId	c:transactionIdType	[1..1]	The id of the transaction that the TPP would like to revoke.

4.7.3   Output Fields

#	Name	Type	Num	Description
1	transactionId	c:transactionIdType	[1..1]	The transaction id given by the API.
2	status	c:statusType	[1..1]	The status of the approval after the request