Scenarios for working with the service

Authentication

For authentication, when sending most requests, the authentication attribute `Authorization` must be passed in the request header containing the value `accessToken` (security token). A list of the requests that require authentication is available in the Connection URLs section. The simplified procedure for obtaining a security token is as follows.

1. The merchant (application) searches for the customer by the phone number.
2. If the customer is found, the payment gateway returns their identifier (uuid).
3. The merchant makes a request for an access code (accessCode), using the merchant's name (merchantLogin) and the received customer ID (uuid).
4. The payment gateway returns the access code (`accessCode`) and sends an SMS message from the number 900 containing a one-time password to the customer's phone (`otp`).

The lifetime of the security code is 60 seconds. If the customer does not enter a one- time password during this time, the request must be repeated.

5. The customer enters a one- time password into the form and confirms it.
6. The merchant sends the one-time password value (otp) and the access code (accessCode).
7. The payment gateway returns a security token (accessToken) that should be used in the header of requests for transactions with bindings.

The lifetime of the security token (`accessToken`) is 2592000 seconds, i.e., 30 days.

The lifetime of this parameter can also be configured and changed in agreement with the Bank.

The security token is returned as a JSON document. The received security token is a symmetric application secret, so the application developer must take measures to protect it: store the token in encrypted form, provide access to it only after the user is authorized in the application. One application can receive only one security token for one user. Reauthorization (with the same user ID value) cancels previously issued permissions.

Below, the procedure for getting the security token (accessToken) is presented with links to the required requests.

Step number	**Description**	**Endpoint**	**Input data**	**Output data to be used in further steps.**	**Link to detailed description of the request**
Preconditions: no					
Step 1	Request to check whether a customer with the specified phone number is in the system.	/api/clients/find	phone	client/uuid requestId status	Request to search for a customer

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Step 2	Getting an access code.	/api/auth/otp	merchantLogin uuid	accessCode requestId status	Request an access code
Step 3	Getting a security token.	/api/auth/token	accessCode otp	accessToken requestId status	Request for a security token

Payment – card data gathered on Merchant side, 3DS payment

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Preconditions: the merchant has authenticated (see section Authentication).					
Step 1	Start of the payment procedure	/api/payment/start	transactionNumber source/card/pan source/card/expiryDate source/card/cardholderName source/card/cvc amount/transactionAmount amount/transactionFee amount/currency	transactionNumber mdOrder info redirectUrl paReq requestId status	Start of payment
Step 2	Displaying the info message to the customer.				
Step 3	Redirecting the client to the received `redirectUrl` (ACS of the issuer's bank) with paReq specified. 1. The customer's browser requests a user authentication form from ACS. 2. ACS sends this form to the customer's browser. 3. The customer fills in the form and sends it to ACS. 4. ACS processes the completed form and passes redirect to the return URL to the browser. At the same time, encrypted parameters of the authentication result are passed in the URL.				
Step 4	Receiving the response from ACS with paRes.				
Step 5	Completion of the payment procedure.	/api/payment/finish	mdOrder transactionNumber paRes	mdOrder requestId status	Payment completion
Step 6	The merchant requests the result of the transaction.	/api/find	`transactionNumber` or `mdOrder`	mdOrder mdOrderStatus transactionNumber transactionType transactionDate source/maskedPan source/binding amount/transactionAmount amount/transactionFee amount/currency requestId status	Request for information about the result of an operation

Payment – use of a previously saved binding, 3DS card

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Preconditions: • The merchant has authenticated (for more details, see the section Authentication). • The Customer has at least one card saved in the profile. The customer selected the saved card in the application interface.					

Step 1	Start of the payment procedure	/api/payment/start	`transactionNumber`
			`source/binding`
			`amount/transactionAmount`
			`amount/transactionFee`
			`amount/currency`
				`transactionNumber`
				`mdOrder`
				`info`
				`redirectUrl`
				`paReq`
				`requestId`
				`status`
					Start of payment

Step 2	Displaying the info message to the customer.				
Step 3	Redirecting the customer to the received redirectUrl (ACS of the issuing bank) with the indication of `pa_req`. 1. The customer's browser requests a user authorization form from ACS. 2. ACS sends this form to the customer's browser. 3. The customer fills in the form and sends it to ACS. 4. ACS processes the completed form and passes `redirect` to the return URL to the browser. The encrypted parameters of the authentication result are passed along with this URL.				
Step 4	Getting a response from ACS with `paRes`.				

Step 5	Completion of the payment procedure.	/api/payment/finish	`mdOrder`
			`transactionNumber`
			`paRes`
				`mdOrder`
				`requestId`
				`status`
					Payment completion

Step 6	The merchant requests the result of the transaction.	/api/find	`transactionNumber` or `mdOrder`	`mdOrder`
				`mdOrderStatus`
				`transactionNumber`
				`transactionType`
				`transactionDate`
				`source/maskedPan`
				`amount/transactionAmount`
				`amount/transactionFee`
				`amount/currency`
				`requestId`
				`status`
					Request for information about the result of an operation

Payment – use of a previously saved binding, SSL card

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Preconditions: • The merchant has authenticated and received a security token (for more details, see the section Authentication). • The Customer has at least one card saved in the profile.					

Step 1	Start of the payment procedure.	/api/payment/start	`transactionNumber`
			`source/binding`
			`amount/transactionAmount`
			`amount/transactionFee`
			`amount/currency`
				`transactionNumber`
				`mdOrder`
				`requestId`
				`status`
					Start of payment

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request

Step 2	Completing the payment with specifying `mdOrder` received on Step 1.	/api/payment/finish	`mdOrder`
			`transactionNumber`
				`transactionNumber`
				`mdOrder`
				`requestId`
				`status`
					Payment completion

Step 3	The merchant requests the result of the transaction.	/api/find	`transactionNumber` or `mdOrder`	`mdOrder`
				`mdOrderStatus`
				`transactionNumber`
				`transactionType`
				`transactionDate`
				`source/maskedPan`
				`amount/transactionAmount`
				`amount/transactionFee`
				`amount/currency`
				`requestId`
				`status`
					Request for information about the result of an operation

Payment – collection of card data on the merchant's side, 3DS card

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Preconditions: • The merchant has authenticated (for more details, see the section Authentication). • The client has an issued virtual card (see sectionCreating a prepaid virtual card).					

Step 1	Start of the transfer procedure.	/api/transfer/start	`transactionNumber`
			`source/card/pan`
			`source/card/expiryDate`
			`source/card/cardholderName`
			`source/card/cvc`
			`amount/transactionAmount`
			`amount/transactionFee`
			`amount/currency`
				`transactionNumber`
				`mdOrder`
				`info`
				`redirectUrl`
				`paReq`
				`requestId`
				`status`
					Start of transfer

Step 2	Displaying the `info` message to the customer.				
Step 3	Redirect the client to the received `redirectUrl` (ACS of the issuing bank) with `paReq`. 1. The customer's browser requests a user authentication form from ACS. 2. ACS sends this form to the customer's browser. 3. The customer fills in the form and sends it to ACS. 4. ACS processes the completed form and passes `redirect` to the return URL to the browser. At the same time, encrypted parameters of the authentication result are passed in the URL.				
Step 4	Getting a response from ACS with `paRes`.				

Step 5	Completion of the transfer procedure.	/api/payment/finish	`mdOrder`
			`transactionNumber`
			`paRes`
				`mdOrder`
				`requestId`
				`status`
					Transfer completion

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request

Step 6	The merchant requests the result of the transaction.	/api/find	`transactionNumber` or `mdOrder`	`mdOrder`
				`mdOrderStatus`
				`transactionNumber`
				`transactionType`
				`transactionDate`
				`source/maskedPan`
				`source/binding`
				`amount/transactionAmount`
				`amount/transactionFee`
				`amount/currency`
				`requestId`
				`status`
					Request for information about the result of an operation

Transfer – collection of card data on the merchant's side, SSL card

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Preconditions: • The merchant has authenticated (for more details, see the section Authentication). • The client has an issued virtual card (see sectionCreating a prepaid virtual card).					

Step 1	Start of the transfer procedure.	/api/transfer/start	`transactionNumber`
			`source/card/pan`
			`source/card/expiryDate`
			`source/card/cardholderName`
			`source/card/cvc`
			`amount/transactionAmount`
			`amount/transactionFee`
			`amount/currency`
				`transactionNumber`
				`mdOrder`
				`requestId`
				`status`
					Start of transfer

Step 2	Completing the payment with specifying `mdOrder` received on Step 1.	/api/transfer/finish	`mdOrder` `transactionNumber`	N/A	Transfer completion

Step 3	The merchant requests the result of the transaction.	/api/find	`transactionNumber` or `mdOrder`	`mdOrder`
				`mdOrderStatus`
				`transactionNumber`
				`transactionType`
				`transactionDate`
				`source/maskedPan`
				`amount/transactionAmount`
				`amount/transactionFee`
				`amount/currency`
				`requestId`
				`status`
					Request for information about the result of an operation

Transfer – use of a previously saved binding, 3DS card

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Preconditions: • The merchant has authenticated and received a security token (for more details, see Authentication). • The Customer has at least one card saved in the profile. A virtual card has been issued (for more details, see Issuing a virtual card).					

Step 1	Start of the transfer procedure.	/api/transfer/start	`transactionNumber`
			`source/binding`
			`target/binding`
			`amount/transactionAmount`
			`amount/transactionFee`
			`amount/currency`
				`transactionNumber`
				`mdOrder`
				`info`
				`redirectUrl`
				`paReq`
				`requestId`
				`status`
					Start of transfer

Step 2	Displaying the `info` message to the customer.				
Step 3	Redirecting the customer to the received redirectUrl (ACS of the issuing bank) with the indication of `pa_req`. 1. The customer's browser requests a user authorization form from ACS. 2. ACS sends this form to the customer's browser. 3. The customer fills in the form and sends it to ACS. 4. ACS processes the completed form and passes `redirect` to the return URL to the browser. The encrypted parameters of the authentication result are passed along with this URL.				
Step 4	Getting a response from ACS with `paRes`				

Step 5	Completion of the transfer procedure.	/api/transfer/finish	`mdOrder`
			`transactionNumber`
			`paRes`
				`mdOrder`
				`requestId`
				`status`
					Transfer completion

Step 6	The merchant requests the result of the transfer.	/api/find	`transactionNumber` or `mdOrder`	`mdOrder`
				`mdOrderStatus`
				`transactionNumber`
				`transactionType`
				`transactionDate`
				`source/maskedPan`
				`target/maskedPan`
				`amount/transactionAmount`
				`amount/transactionFee`
				`amount/currency`
				`requestId`
				`status`
					Request for information about the result of an operation

Transfer – use of a previously saved binding, SSL card

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Preconditions: • The merchant has authenticated and received a security token (for more details, see Authentication). • The Customer has at least one card saved in the profile. A virtual card has been issued (for more details, see Issuing a virtual card).					

Step 1	Start of the transfer procedure.	/api/transfer/start	`transactionNumber`
			`source/binding`
			`target/binding`
			`amount/transactionAmount`
			`amount/transactionFee`
			`amount/currency`
				`transactionNumber`
				`mdOrder`
				`requestId`
				`status`
					Start of transfer

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Step 2	Transfer completion with the `mdOrder` received on Step 1.	/api/transfer/finish	`mdOrder` `transactionNumber`	N/A	Transfer completion

Step 3	The merchant requests the result of the transfer.	/api/find	`transactionNumber` or `mdOrder`	`mdOrder`
				`mdOrderStatus`
				`transactionNumber`
				`transactionType`
				`transactionDate`
				`source/maskedPan`
				`target/maskedPan`
				`amount/transactionAmount`
				`amount/transactionFee`
				`amount/currency`
				`requestId`
				`status`
					Request for information about the result of an operation

Issuing a virtual card

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Preconditions: • The merchant has authenticated and received a security token (for more details, see Authentication). • The Customer has at least one card saved in the profile.					

Step 1	Request for card creation	/api/vcard/create	`transactionNumber`	`transactionNumber`
				`binding/uuid`
				`binding/createdDate`
				`binding/paymentType`
				`binding/paymentSystem`
				`card/maskedPan`
				`card/extendedInfo`
				`requestId`
				`status`
					Issuing a virtual card

Step 2	The beginning of the transfer procedure to the issued card. The `target/binding` specified is the `binding/uuid` obtained at Step 1.	/api/transfer/start	`transactionNumber`
			`source/binding`
			`target/binding`
			`amount/transactionAmount`
			`amount/transactionFee`
			`amount/currency`
				`ransactionNumber`
				`mdOrder`
				`info`
				`redirectUrl`
				`paReq`
				`requestId`
				`status`
					Start of transfer

Step 3	Displaying of the `info` message (obtained in Step 2) to the client.				
Step 4	Redirect the client to the received `redirectUrl` (ACS of the issuing bank) with paReq. 1. The customer's browser requests a user authorization form from ACS. 2. ACS sends this form to the customer's browser. 3. The customer fills in the form and sends it to ACS. 4. ACS processes the completed form and passes `redirect` to the return URL to the browser. Along with the URL the encrypted parameters of the authorization result are passed.				
Step 5	Getting a response from ACS with paRes				

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request

Step 6	Completion of the transfer procedure	/api/transfer/finish	`mdOrder`
			`transactionNumber`
			`paRes`
				`mdOrder`
				`requestId`
				`status`
					Transfer completion

Step 7	The merchant requests the result of the transaction.	/api/find	`transactionNumber` or `mdOrder`	`mdOrder`
				`mdOrderStatus`
				`transactionNumber`
				`transactionType`
				`transactionDate`
				`source/maskedPan`
				`target/maskedPan`
				`amount/transactionAmount`
				`amount/transactionFee`
				`amount/currency`
				`requestId`
				`status`
					Request for information about the result of an operation

Step 8	Request for changing the binding name. The `binding/uuid` specified is the `binding/uuid` obtained at Step 1.	/api/change	`transactionNumber`
			`binding/uuid`
			`binding/mnemonic`
				`transactionNumber`
				`requestId`
				`status`
					Change of the displayed card name

Getting a list of performed operations

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Preconditions: The merchant has authenticated and received a security token (for more details, see Authentication).					

Step 1	Change request	/api/transactions/	`dateFrom`
			`dateTo`
			`responseType`
			`local`
				`transactions`
				`mdOrder`
				`transactionNumber`
				`transactionDate`
				`transactionType`
				`transactionWay`
				`amount`
				`transactionAmount`
				`currency`
				`partner`
				`merchantLogin`
				`merchantInfo`
				`requestId`
				`status`
					Getting a list of performed operations

Getting information about a performed operation

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Preconditions: • The merchant has authenticated and received a security token (for more details, see Authentication). • A payment or transfer operation has been performed.					

Step 1	Change request	/api/transactions/find	`mdOrder`
			`transactionNumber`
				`mdOrder`
				`mdOrderStatus`
				`transactionNumber`
				`transactionType`
				`transactionWay`
				`transactionDate`
				`transactionDetails`
				`source`
				`card`
				`maskedPan`
				`binding`
				`target`
				`card`
				`maskedPan`
				`binding`
				`amount`
				`transactionAmount`
				`transactionFee`
				`currency`
				`requestId`
				`status`
					Request for information about the result of an operation

Change of the displayed name of the binding

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Preconditions: • The merchant has authenticated and received a security token (for more details, see Authentication). • The Customer has at least one card saved in the profile. The customer selected the saved card in the application interface.					

Step 1	Change request.	/api/change	`transactionNumber`
			`binding/uuid`
			`binding/mnemonic`
				`transactionNumber`
				`requestId`
				`status`
					Change of the displayed name of the binding

Binding deletion

Step number	Description	Endpoint	Input data	Output data to be used in further steps.	Link to detailed description of the request
Preconditions: • The merchant has authenticated and received a security token (for more details, see Authentication). • The Customer has at least one card saved in the profile. The customer selected the saved card in the application interface.					

Step 1	Request for deletion	/api/delete	`transactionNumber`
			`type`
			`uuid`
				`transactionNumber`
				`requestId`
				`status`
					Binding deletion