

Merchant Documentation
Alfa Bank
SBP

Alfa Bank

only one-phase payment

Оглавление

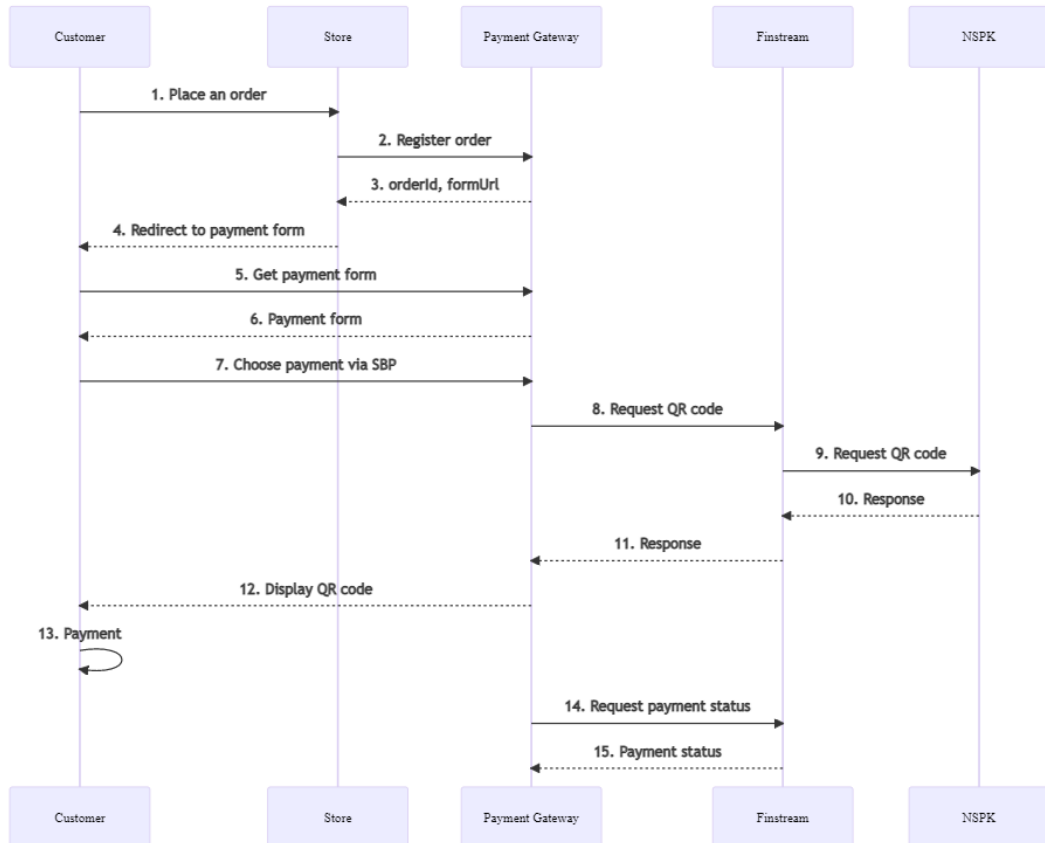
1	Terms and definitions	3
2	Payment via SBP	4
2.1	Payment on payment page.....	4
2.2	Payment with card details input on Merchant side.....	5
2.3	Payment through the "instant invoice" service	6
2.4	Payment API description	6
2.4.1	SOAP	6
2.4.2	REST	23
3	Refunds	40
3.1	Refunds via API	40
3.1.1	Scenario for a refund via API	40
3.1.2	Refund API description	41
3.2	Refunds via Merchant Portal	44
4	Looking up orders paid via SBP	45
4.1	Looking up orders	45
4.2	Order history	45
5	Fiscalization of transactions	46
6	Dummy mode on the test bench	47

1 Terms and definitions

Term	Description
Bank	Acquiring bank. A bank that implements and operates the Payment Gateway.
Merchant, Store	A retail business establishment (RBE) that sells goods or provides services through a website.
SBP	Fast Payments System provided by National System of Payment Cards (NSPK) under the Central Bank of Russian Federation.

2 Payment via SBP

2.1 Payment on payment page

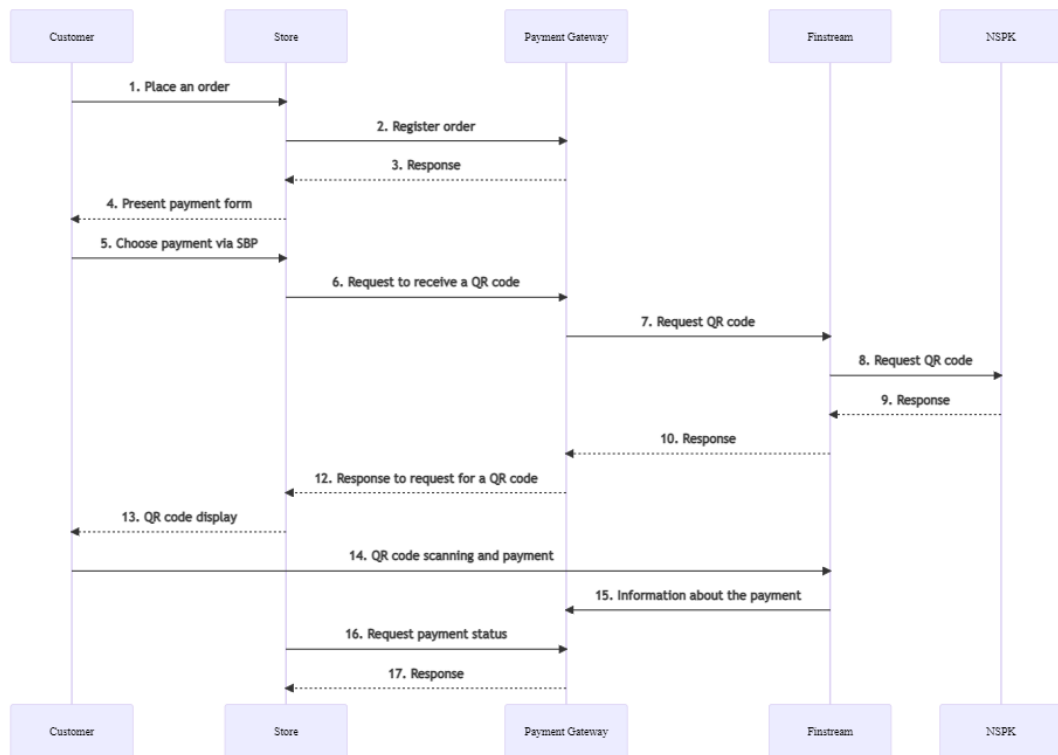


Description:

1. The buyer places an order and proceeds to payment.
2. The Merchant sends an order registration request to the Payment Gateway. Specification of request is available in the following sections:
 - "[2.4.1.1. Request for order registration \(SOAP\)](#)";
 - "[2.4.2.1. Request for order registration \(REST\)](#)".
3. In response to a registration request, the Payment Gateway returns a unique order identifier in the payment system (in the 'orderId' parameter) and a URL to which the Customer must be redirected to receive the payment form (in the 'formUrl' parameter).
4. The Merchant system should redirect the Customer's browser to the URL received from the Payment Gateway ('formUrl' parameter) in response to the request for order registration.
5. Customer's browser requests the page at received URL.
6. The browser displays the payment page, where the Customer chooses to pay via SBP.
7. The payment page requests a QR code from the Payment Gateway.

- 8-11. The payment gateway sends a request for a QR code to the Finstream system.
- 12. A QR code is displayed to the Customer on the payment page.
- 13. The client scans the QR code using specialized software and makes a payment.
- 14-15. The Payment Gateway polls Finstream for the status of the payment and, if the status is successful, completes the payment.

2.2 Payment with card details input on Merchant side



Description:

1. The buyer places an order and proceeds to payment.
2. The Merchant sends an order registration request to the Payment Gateway. Specification of request is available in the following sections:
 - "[2.4.1.1. Request for order registration \(SOAP\)](#)";
 - "[2.4.2.1. Request for order registration \(REST\)](#)".

3. In response to a registration request, the Payment Gateway returns a unique order identifier in the payment system (in the 'orderId' parameter).
4. The Merchant redirects the Customer to Merchant's payment form.
5. The Customer chooses to pay via SBP.
6. The Merchant sends a request to receive a QR code to the Payment Gateway. Request specifications are presented in the following sections:
 - "[2.4.1.2. Request to receive a QR code \(SOAP\)](#)";
 - "[2.4.2.2. Request to receive a QR code \(REST\)](#)".
- 7-10. The payment gateway sends a request for a QR code to the Fistream system.
11. The Payment Gateway returns a response to request for a QR code request to the Merchant's system.
12. The QR code is displayed to the Customer.
13. The client scans the QR code using specialized software and makes a payment.
14. The Payment Gateway receives information about the payment made.
- 15-16. The Merchant requests the payment status from the Payment Gateway. Request specifications are presented in the following sections:
 - "[2.4.1.3. Request for status of payment by QR code \(SOAP\)](#)";
 - "[2.4.2.3. Request for status of payment by QR code \(REST\)](#)".

2.3 Payment through the "instant invoice" service

Payment through the "instant invoice" service is available both on the payment page and on the Merchant side.

The payment scenarios are similar to those presented in sections "[2.1. Payment on payment page](#)" and "[2.2. Payment with card details input on Merchant side](#)".

If this functionality is enabled, the QR code displayed to the Customer also serves as a link. To make a payments, instead of scanning the QR code, the Customer clicks on it, after which the online bank opens.

2.4 Payment API description

2.4.1 SOAP

2.4.1.1 Request for order registration (SOAP)

The request for order registration is `registerOrder`, it is described in WSDL of the service.

WSDL (test): <https://web.rbsuat.com/ab/webservices/merchant-ws?wsdl>

WSDL (prod): <https://pay.alfabank.ru/payment/webservices/merchant-ws?wsdl>

Request parameters:

Name	Type	Mandatory	Description
merchantOrderNumber	ANS..32	yes	Identifier of an order in the Store system, it is unique for each Store within the system
description	ANS..598	no	<p>Description of the Order in any format.</p> <div style="border: 1px solid black; background-color: #e6f2ff; padding: 5px;"> <p>To enable sending this field to the processing system, contact the technical support service.</p> </div>
amount	N..12	yes	Amount of the payment in minimum currency units
currency	N3	no	ISO 4217 code of the payment currency. If not specified, default value is used.
language	A2	no	Language in the ISO 639-1 encoding. If the language is not specified, the default language defined in the store settings is used.
feeInput	N..8	no (see description)	<p>Commission amount in minimum currency units.</p> <div style="border: 1px solid black; background-color: #e6f2ff; padding: 5px;"> <p>The parameter is passed only when the corresponding permission is enabled: Merchant commission transfer allowed.</p> </div>
pageView	ANS..20	no	<p>The value of this parameter defines what kind of pages of the payment interface will be loaded for the customer. The possible values are:</p> <ul style="list-style-type: none"> • DESKTOP – pages designed to be displayed on desktop computers (pages with names like <code>payment_<locale>.html</code> and <code>errors_<locale>.html</code> will be searched for in the payment service archive); • DESKTOP – pages designed to be displayed on mobile devices (pages with names like <code>mobile_payment_<locale>.html</code> and <code>mobile_errors_<locale>.html</code> will be searched for in the payment service archive); • If the store has added arbitrary prefixes to their page file names, value of the desired prefix should be passed in the <code>pageView</code> parameter for the corresponding pages to be loaded. For example, if <code>iphone</code> is passed in the parameter, pages with names like <code>iphone_payment_<locale>.html</code> and <code>iphone_error_<locale>.html</code> will be searched for in the payment service archive. <p>Where:</p>

Name	Type	Mandatory	Description
			<p>locale – page language as ISO 639-1 code. For example, ru for Russian or en for English.</p> <p>If the parameter is missing or its value is in incorrect format, the default value is <code>pageView=DESKTOP</code>.</p>
sessionTimeoutSeconds	N..9	no	<p>Order lifetime in seconds.</p> <p>If the parameter is not specified, the value specified in the Merchant settings or the default value (1200 seconds = 20 minutes) will be used.</p> <p>If the request contains the <code>expirationDate</code> parameter, then the value of <code>sessionTimeoutSecs</code> parameter is disregarded.</p>
bindingId	AN..255	no	<p>Identifier of the binding created earlier. It can be used only if the store has the permission to work with bindings. If this parameter is passed in the given request, it means that:</p> <ol style="list-style-type: none"> 1. The given order can be paid only using a binding; 2. The Payer will be redirected to the payment page on which only entering CVC is required.
expirationDate	ANS	no	<p>The date and time of the order lifetime expiration. Format: <code>yyyy-MM-dd'T'HH:mm:ss</code>.</p> <p>If this parameter is not present in the request, the <code>sessionTimeoutSecs</code> parameter is used to define the time of the end of life of the order.</p>
returnUrl	ANS..512	yes	<p>The address to which the user is to be redirected in case of a successful payment. The URL must be specified in full including the protocol used (for example, https://test.ru instead of test.ru). Otherwise, the user will be redirected to the address of the following type: http://<payment_gateway_address>/<merchant_address>.</p> <div style="border: 1px solid #add8e6; padding: 5px; margin-top: 10px;"> <p>See also the 3DS-payment block for one-phase and two-phase payments. <code>returnUrl</code> is used in the same way as in Step 11 in these diagrams.</p> </div>
failUrl	ANS..512	no	<p>The address to which the user is to be redirected in case of a failed payment. The address must be specified in full including the protocol used (for example, https://test.ru instead of test.ru). Otherwise, the user will be redirected to the address of the following type: http://<payment_gateway_address>/<merchant_address>.</p> <div style="border: 1px solid #add8e6; padding: 5px; margin-top: 10px;"> <p>See also the 3DS-payment block for one-phase and two-phase payments. <code>failUrl</code> is</p> </div>

Name	Type	Mandatory	Description												
			used in the same way as in Step 11 in these diagrams.												
dynamicCallbackUrl	ANS..512	no	<p>This parameter enables dynamic callback functionality. All payment-related callbacks enabled for Merchant could be directed to a URL passed in this parameter. Payment-related notifications are callback notifications about the following events: successful hold, payment rejected by timeout, cardpresent payment rejected, successful debiting, refund, reversal. Notably, payment-unrelated callbacks that are active for the Merchant (enabling/disabling a binding, binding creation) will be sent to static callback URL.</p> <p>Note To use the functionality of dynamic callback notifications, the Merchant must have the appropriate permission enabled for them: Callback type: Dynamic (CALLBACK_TYPE = DYNAMIC).</p> <p>To receive callbacks, the Merchant must have a specific permission enabled for them: Sending callback notification is allowed.</p>												
params	See description	no	<p>Tag containing attributes to pass additional parameters of the Merchant.</p> <p>The fields for additional information and its subsequent storage. To pass N parameters, a request must contain N <code>params</code> tags, where the <code>name</code> attribute contains the parameter name and the <code>value</code> attribute contains its value.</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Mandatory</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>ANS..255 bytes</td> <td>yes</td> <td>Name of the additional parameter</td> </tr> <tr> <td>value</td> <td>ANS..1024 bytes</td> <td>yes</td> <td>Value of the additional parameter</td> </tr> </tbody> </table> <p>These fields can be passed to the Bank processing system to be subsequently displayed in the registries.*</p> <p>This functionality can be enabled upon additional agreement with the bank during the integration period.</p>	Name	Type	Mandatory	Description	name	ANS..255 bytes	yes	Name of the additional parameter	value	ANS..1024 bytes	yes	Value of the additional parameter
Name	Type	Mandatory	Description												
name	ANS..255 bytes	yes	Name of the additional parameter												
value	ANS..1024 bytes	yes	Value of the additional parameter												

Name	Type	Mandatory	Description
			<p>In case of payments for housing and communal services, the following additional parameters must be passed:</p> <ul style="list-style-type: none"> • <code>payment_doc_id</code> – the identifier of the payment document; • <code>order_num</code> - the order number. <p>If sending notifications to Customers is set up for a Merchant, the email address of a Customer must be passed in this tag in the <code>email</code> parameter.</p>
<code>clientId</code>	AN..255	no	<p>Identifier of the Customer in the Store system. Is used to facilitate bindings. It can be present if creating bindings is allowed for the Store.</p> <p>Specifying this parameter when processing payments with the use of bindings is mandatory. Otherwise, a payment will be unsuccessful.</p>
<code>merchantLogin</code>	AN..255	no	<p>To register an order on behalf of a child Merchant, specify the Merchant login in this parameter.</p>
<code>features</code>	ANS..255	no	<p>A container for <code>feature</code> parameter, where the following values can be specified.</p> <p><code>AUTO_PAYMENT</code> - if the request for order registration initiates auto-payments.</p> <p><code>VERIFY</code> - if this parameter is specified, after the request for order registration, the cardholder is to be verified without debiting funds from the cardholder account. Thus it is possible to pass a zero amount in the request. This verification allows the Merchant to ensure that a card belongs to the cardholder and to debit this card in the future without verifying authentication data (CVC, 3-D Secure) when processing subsequent payments.</p> <p>Specifics of passing the VERIFY value</p> <ul style="list-style-type: none"> • Even if the payment amount is to be passed in the request, it will not be debited from the buyer's account. • After the order has been successfully registered, it is put to the REVERSED (cancelled) status. <p><code>FORCE_TDS</code> - use of 3-D Secure is enforced for the payment. If the card does not support 3-D Secure, the transaction will fail.</p>

Name	Type	Mandatory	Description
			<p>FORCE_SSL - use of SSL is enforced for the payment (no 3-D Secure).</p> <p>FORCE_FULL_TDS - after authentication using 3-D Secure, the PaRes status must only be Y, which guarantees successful user authentication. Otherwise, the transaction will fail.</p> <p>Example of use is given below.</p> <pre style="border: 1px dashed blue; padding: 5px;"> <features> <feature>AUTO_PAYMENT</feature> </features></pre>
email	ANS..40	no	Customer's email address.
phone	ANS.12	no	<p>Customer's phone number. If the phone includes a country code, the number must begin with a plus sign ("+"). If the phone is passed without a plus sign ("+"), the country code should not be specified. Thus, the following options are valid:</p> <ul style="list-style-type: none"> • +79998887766; • 9998887766.

* By default the following fields are passed to the bank processing system:

- `orderNumber` – order number in the store system;
- `description` – order description (no more than 99 characters; %, +, carriage return \r, and line feed \n cannot be used).

Response parameters:

Name	Type	Mandatory	Description
orderId	ANS36	no	Identifier of the order in the payment system. It is unique within the system. The identifier is missing if the order registration failed due to an error detailed in <code>errorCode</code> .
formUrl	AN..512	no	URL of the payment form to which the Customer's browser is to be redirected. This parameter is not returned if the registration of the order was not successful due to the error described in <code>errorCode</code> .
errorCode	N3	no	Error code.
errorMessage	AN..512	no	Error description in the language passed in the <code>language</code> parameter in the request.

Error codes (the `errorCode` field):

Value	Description
0	Request processing took place without system errors
1	Unknown order ID
1	Order number is duplicated, order with given order number is processed already
3	Currency is not defined

Value	Description
4	Empty amount
4	Order number is empty
4	Empty return URL
5	A parameter value has been specified incorrectly
5	Access denied
5	The user must change his password
7	System error
13	It is not allowed to use both FORCE_TDS/FORCE_SSL and AUTO_PAYMENT
13	The Merchant does not have the permission to process auto-payments
13	The Merchant does not have the permission to process verification payments
14	Features are specified incorrectly.

Request example:

```

<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mer="http://engine.paymentgate.ru/webservices/merchant">
  <soapenv:Header/>
  <soapenv:Body>
    <mer:registerOrder>
      <order merchantOrderNumber="78ds901234567890" description=" "
amount="15000" currency=" " language=" " pageView="MOBILE"
sessionTimeoutSecs=" " bindingId=" " expirationDate="2014-09-08T14:14:14">
        <returnUrl>http://yoursite.com</returnUrl>
        <params name="param1" value="valueParam1"/>
        <params name="param2" value="valueParam2"/>
        <clientId>666</clientId>
        <merchantLogin> </merchantLogin>
        <features>
          <feature>AUTO_PAYMENT</feature>
        </features>
      </order>
    </mer:registerOrder>
  </soapenv:Body>
</soapenv:Envelope>

```

Response example:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:registerOrderResponse
xmlns:ns1="http://engine.paymentgate.ru/webservices/merchant">
      <return orderId="05fcbc62-7ee6-4f1a-b3d5-6ca41a982283"
errorCode="0" errorMessage="Success">
        <formUrl>
https://server/application\_context/mobile\_payment\_ru.html?mdOrder=05fcbc62-7ee6-4f1a-b3d5-6ca41a982283 </formUrl>
        </return>
      </ns1:registerOrderResponse>
    </soap:Body>
  </soap:Envelope>

```

2.4.1.2 Request to receive a QR code (SOAP)

WSDL (test): <https://web.rbsuat.com/ab/webservices/sbpc2b-ws?wsdl>

WSDL (prod): <https://pay.alfabank.ru/payment/webservices/sbpc2b-ws?wsdl>

The request used to get the QR code is 'getDynamicQr'.

Request parameters

Parameter	Type	Mandatory	Description
mdOrder	ANS..36	Yes	Order number in the Payment Gateway system.
qrHeight	String	No	The height of the QR code in pixels. Specify if <code>renderedQR</code> is required. Minimum value: 10. Maximum value: 1000
qrWidth	String	No	The width of QR code. Specify if <code>renderedQR</code> is required. Minimum value: 10. Maximum value: 1000.
qrFormat	ANS*	No	The possible values are: matrix - returns a matrix of zeros and ones; image - returns the image in base64.

Response parameters

Parameter	Type	Mandatory	Description
errorCode	ANS..3	No	Error code.
errorMessage	AN..512	No	Error description.
payload	String	No	The content of the QR code registered in the SBP. Present if value of <code>qrStatus</code> is <code>STARTED</code> .
qrId	String	No	ID of the QR code.
qrStatus	String	No	Status of the QR code request. The available values are: <ul style="list-style-type: none"> STARTED - QR code generated; CONFIRMED - the order is accepted for payment; REJECTED - payment declined; REJECTED_BY_USER - payment by QR code was rejected by the Merchant; ACCEPTED - order has been paid.
renderedQr		No	QR code in PNG format encoded in Base64. Present if the request had <code>qrHeight</code> and <code>qrWidth</code> specified, and at the same time the value of <code>qrStatus</code> is <code>STARTED</code> .

Request format

```

<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mer="http://engine.paymentgate.ru/webservices/merchant">
  <soapenv:Header/>
  <soapenv:Body>
    <mer:getDynamicQr>
      <request>
        <mdOrder>string</mdOrder>
        <qrHeight>40</qrHeight>
        <qrWidth>40</qrWidth>
        <qrFormat>image</qrFormat>
      </request>
    </mer:getDynamicQr>
  </soapenv:Body>
</soapenv:Envelope>

```

Response format

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:getDynamicQrResponse
xmlns:ns1="http://engine.paymentgate.ru/webservices/merchant">
      <return>
        <errorMessage>string</errorMessage>
        <payload>string</payload>
        <qrId>string</qrId>
        <qrStatus>STARTED</qrStatus>
        <renderedQr>string</renderedQr>
      </return>
    </ns1:getDynamicQrResponse>
  </soap:Body>
</soap:Envelope>

```

2.4.1.3 Request for status of payment by QR code (SOAP)

WSDL (test): <https://web.rbsuat.com/ab/webservices/sbpc2b-ws?wsdl>

WSDL (prod): <https://pay.alfabank.ru/payment/webservices/sbpc2b-ws?wsdl>

The requests used to get the status of an order in the SBP is `getQrStatus`.

Request parameters

Parameter	Type	Mandatory	Description
mdOrder	ANS..36	Yes	Order number in the Payment Gateway system.
qrId	String	Yes	ID of the QR code.

Response parameters

Parameter	Type	Mandatory	Description
errorCode	ANS..3	No	Error code.
errorMessage	AN..512	No	Error description.
payload	String	No	The content of the QR code registered in the SBP. Present if value of <code>qrStatus</code> is <code>STARTED</code> .

Parameter	Type	Mandatory	Description
qrStatus	String	No	Status of the QR code request. The available values are: <ul style="list-style-type: none"> STARTED - QR code generated; CONFIRMED - the order is accepted for payment; REJECTED - payment declined; REJECTED_BY_USER - payment by QR code was rejected by the Merchant; ACCEPTED - order has been paid.
qrType	String	No	QR code type: <ul style="list-style-type: none"> STATIC - static; DYNAMIC - dynamic. Currently, the value returned can be only DYNAMIC.
transactionState	String	No	Order status: <ul style="list-style-type: none"> CREATED - order is created; DECLINED - order is declined; DEPOSITED – order is paid;
transactionStateExtension	String	No	Extended status of the order: <ul style="list-style-type: none"> SBP_C2B_STARTED - order is created; SBP_C2B_CONFIRMED - order is confirmed; SBP_C2B_PENDING - the order is awaiting processing.

Request format

```

<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mer="http://engine.paymentgate.ru/webservices/merchant">
  <soapenv:Header/>
  <soapenv:Body>
    <mer:getQrStatus>
      <request>
        <mdOrder>string</qrId>
        <qrId>string</qrStatus>
      </request>
    </mer:getQrStatus>
  </soapenv:Body>
</soapenv:Envelope>

```

Response format

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:getQrStatusResponse
      xmlns:ns1="http://test.ru/webservices/merchant">
      <return>
        <errorCode>string</errorCode>
        <errorMessage>string</errorMessage>
        <payload>string</payload>
        <qrStatus>STARTED</qrStatus>
        <qrType>STATIC</qrType>
        <transactionState>CREATED</transactionState>

        <transactionStateExtension>SBP_C2B_STARTED</transactionStateExtension>
      </return>
    </ns1:getQrStatusResponse>
  </soap:Body>
</soap:Envelope>

```

2.4.1.4 Request for order status (SOAP)

WSDL (test): <https://web.rbsuat.com/ab/webservices/sbpc2b-ws?wsdl>

WSDL (prod): <https://pay.alfabank.ru/payment/webservices/sbpc2b-ws?wsdl>

To request the status of a registered order, the `getOrderStatusExtended` request is used.

Request parameters:

Name	Type	Mandatory	Description
orderId	ANS36	yes*	Order identifier in the payment system. It is unique within the system.
language	A2	no	Language in the ISO 639-1 encoding. If this parameter is not specified, it is considered that the language is Russian. The error message will be returned in that language.
merchantOrderNumber	ANS..32	yes*	Order identifier in the Store system.

* It is necessary to pass either the `orderId` or `merchantOrderNumber` parameter in the request. If both parameters are passed in the request, `orderId` has a higher priority.

Several sets of response parameters exist. The exact sets of parameters to be returned depend on the version of `getOrderStatusExtended` specified in the merchant settings.

Name	Type	Mandatory	Description	Version of <code>getOrderStatusExtended</code>
orderNumber	AN..32	yes	Order identifier in the Store system.	All versions.
orderStatus	N2	no	<p>The order status in the payment system is defined by the value of this parameter. The list of possible values is given below. It is missing if the order has not been found.</p> <ul style="list-style-type: none"> 0 - the order is registered, but not paid; 	All versions.

Name	Type	Mandatory	Description	Version of getOrderStatusExtended
			<ul style="list-style-type: none"> • 1 - the pre-authorized amount is put on hold (for two-phase payments); • 2 - a full authorization of the order amount has been performed; • 3 - the authorization was cancelled; • 4 - a refund operation has been processed for the transaction; • 5 - an authorization through ACS of the issuing bank has been initiated; • 6 - the authorization was declined 	
actionCode	N3	yes	Response code.	All versions.
actionCodeDescription	AN..512	yes	Error description in the language passed in the 'language' parameter of the request.	All versions.
errorCode	N3	no	<p>Error code. The following values are available.</p> <ul style="list-style-type: none"> • 0 - the request has been processed without system errors.; • 1 - Expected [orderId] or [orderNumber]; • 5 - Access denied; • 5 - The user must change his password; • 6 - Order not found; • 7 - System error. 	All versions.
errorMessage	AN..512	no	Error description in the language passed in the 'language' parameter of the request.	All versions.
amount	N..20	yes	Payment amount in minimum currency units (cents or kopeks)	All versions.
currency	N3	no	ISO 4217 code of the payment currency. If the code is not specified, the default value is 810 (Russian rubles).	All versions.
date	ANS	yes	Order registration date.	All versions.

Name	Type	Mandatory	Description	Version of getOrderStatusExtended
depositedDate	N	no	Date of order payment in Unix-time format (POSIX-time).	10 and higher.
orderDescription	AN..512	no	Order description passed upon its registration.	All versions.
ip	ANS..39	yes	IP address of the user who paid for the order. IPv6 is supported in all requests.	All versions.
authRefNum	AN..24	no	Registration number of the payment authorization that has been assigned to it on the payment registration.	02 and higher.
refundedDate	ANS	no	Date and time of the refund.	13 and more.
paymentWay	AS..14	yes	The way of order completion (a payment with entering card data, a payment using a binding, etc.). This parameter specifies the payment method. Possible values: <ul style="list-style-type: none"> CARD - payment with entering of card details; CARD_BINDING - payment by binding; CARD_MOTO - payment through call center; CARD_PRESENT - payment by cardPresent; ALFA_ALFACLICK – for a payment with "Alfa-click" (through the PayByClik system); UPOP – for a payment through the UPOP system, for the holders of the China Union Pay cards. FILE_BINDING - payment by file; SMS_BINDING - payment by SMS; P2P - transfer from card to card; P2P_BINDING - transfer by binding; 	09 and higher.

Name	Type	Mandatory	Description	Version of getOrderStatusExtended
			<ul style="list-style-type: none"> • PAYPAL - payment from PayPal; • MTS - payment from MTS account; • APPLE_PAY - Apple Pay; • APPLE_PAY_BINDING - payment by Apple Pay binding; • ANDROID_PAY - Android Pay; • ANDROID_PAY_BINDING - payment by Android Pay binding; • GOOGLE_PAY_CARD - Google Pay, non-tokenized; • GOOGLE_PAY_CARD_BINDING - payment by binding with non-tokenized Google Pay card; • GOOGLE_PAY_TOKENIZED - Google Pay, tokenized; • GOOGLE_PAY_TOKENIZED_BINDING - payment by binding with tokenized Google Pay card; • SAMSUNG_PAY - Samsung Pay; • SAMSUNG_PAY_BINDING - payment by Samsung Pay binding; • IPOS - iPOS payment. 	
avsCode	A1	no	<p>AVS Response Code - AVS response code (verification of card holder's registration address and postal code). The possible values are:</p> <ul style="list-style-type: none"> • A - Both address and postal code match; • B - Address matches, postal code does not match; • C - Postal code matches, address does not match; • D - Neither address nor postal code match; • E - Verification requested but not available; 	19 and higher

Name	Type	Mandatory	Description	Version of getOrderStatusExtended
			<ul style="list-style-type: none"> F - Transaction improperly requests verification. 	
<p>The <code>merchantOrderParams</code> element is present in a response if an order contains additional parameters of the Merchant. Each additional order parameter is contained in a separate <code>merchantOrderParams</code> element.</p>				
<code>name</code>	AN..20	no	Name of the additional parameter	All versions.
<code>value</code>	AN..1024	no	Value of the additional parameter	All versions.
<p><code>cardAuthInfo</code> element – element contains a structure consisting of a list of <code>secureAuthInfo</code> element and following parameters:</p>				
<code>pan</code>	N..19	no	Masked number of the card that has been used for the payment. This parameter is to be specified only after the order has been paid.	All versions.
<code>expiration</code>	N6	no	Card expiration date in YYYYMM format. This parameter is to be specified only after the order has been paid.	All versions.
<code>cardholderName</code>	A..64	no	Name of the cardholder. This parameter is to be specified only after the order has been paid.	All versions.
<code>approvalCode</code>	AN6	no	Payment authorization code. This field has a fixed length (six characters) and can contain digits and Latin letters. This parameter is to be specified only after the order has been paid.	All versions.
<code>chargeback</code>	A..5	no	Whether the funds were forcibly returned to the buyer by the bank. The available values are: <ul style="list-style-type: none"> true false 	06 and higher.
<code>paymentSystem</code>	N..10	yes	The payment system name. Available values: <ul style="list-style-type: none"> VISA; MASTERCARD; AMEX; JCB; CUP; MIR. 	08 and higher.

Name	Type	Mandatory	Description	Version of getOrderStatusExtended
product	AN..255	yes	Additional details on corporate cards. These details are filled in by the technical support service through the administrative console. If such details are missing, an empty value is returned.	08 and higher.
productCategory	string	yes	Additional details about corporate cards. These details are filled in by the technical support service through the administrative console. If such details are missing, an empty value is returned. The possible values are: DEBIT, CREDIT, PREPAID, NON_MASTERCARD, CHARGE, DIFFERED_DEBIT.	17
The <code>secureAuthInfo</code> element contains the <code>eci</code> element and the <code>threeDSInfo</code> element, which is a list of parameters <code>cavv</code> and <code>xid</code>):				
eci	N..4	no	Electronic Commerce Indicator. The indicator is specified only after an order has been paid and in case the corresponding permission is present.	All versions.
cavv	ANS..200	no	The value for the cardholder and card authentication check. The indicator is specified only after an order has been paid and in case the corresponding permission is present.	All versions.
xid	ANS..80	no	Electronic Commerce Indicator of the transaction. The indicator is specified only after an order has been paid and in case the corresponding permission is present.	All versions.
The <code>bindingInfo</code> contains the following parameters.				
clientId	AN..255	no	Identifier of the Customer in the Store system passed on the order registration. This parameter is present only if a Store has the permission to create bindings.	All versions.
bindingId	AN..255	no	Identifier of a binding created on an order payment or used to pay for an order. This parameter is present only if a Store has the permission to create bindings.	All versions.
authDateTime	ANS	no	Date and time of authorization.	02 and higher.
terminalId	AN..10	no	Terminal ID.	02 and higher.

Name	Type	Mandatory	Description	Version of getOrderStatusExtended
The <code>paymentAmountInfo</code> contains the following parameters.				
<code>approvedAmount</code>	N..20	no	The amount put on hold on the card (is used only for two-phase payments).	03 and higher.
<code>depositedAmount</code>	N..20	no	The amount authorized to be debited from the card.	03 and higher.
<code>refundedAmount</code>	N..20	no	The refund amount.	03 and higher.
<code>paymentState</code>	A..10	no	The order status.	03 and higher.
<code>feeAmount</code>	N..20	no	Fee amount.	11 and higher.
<code>totalAmount</code>	N..20	no	Order amount + fee (if applicable to the order).	18 and higher.
The <code>bankInfo</code> contains the following parameters.				
<code>bankName</code>	AN..200	no	Issuing bank name.	03 and higher.
<code>bankCountryCode</code>	AN..4	no	Country codes of the issuing bank.	03 and higher.
<code>bankCountryName</code>	AN..160	no	Name of the country of the issuing bank passed in the 'language' parameter of the request or in the language of the user who has called the method if the language has not been specified in the request.	03 and higher.

Request example:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mer="http://engine.paymentgate.ru/webservices/merchant">
  <soapenv:Header/>
  <soapenv:Body>
    <mer:getOrderStatusExtended>
      <order orderId="942e8534-ac73-4e3c-96c6-f6cc448018f7"
language="ru">
        <!--Optional:-->
        <merchantOrderNumber> </merchantOrderNumber>
      </order>
    </mer:getOrderStatusExtended>
  </soapenv:Body>
</soapenv:Envelope>
```

Response example:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:getOrderStatusExtendedResponse
xmlns:ns1="http://engine.paymentgate.ru/webservices/merchant">
      <return orderNumber="00000123" orderStatus="2" actionCode="0"
actionCodeDescription="" amount="10000" currency="810" date="2016-12-
20T18:48:49.590+03:00" ip="10.7.5.21" errorCode="0"
errorMessage="Успешно">
        <attributes name="mdOrder" value="08c87ec5-399d-49d0-85ed-
85c94bb26675"/>
        <cardAuthInfo pan="555555**5599" expiration="201912"
cardholderName="asd asd" approvalCode="123456" paymentSystem="MASTERCARD"
product="DEBIT">
          <secureAuthInfo>
            <eci>1</eci>
          </secureAuthInfo>
        </cardAuthInfo>
        <authDateTime>2016-12-20T18:52:16.276+03:00</authDateTime>
        <terminalId>000001</terminalId>
        <authRefNum>111111111111</authRefNum>
        <paymentAmountInfo paymentState="DEPOSITED"
approvedAmount="10000" depositedAmount="10000" refundedAmount="0"/>
        <bankInfo bankName="Unated State Bank of Atlanta"
bankCountryCode="US" bankCountryName="United States of America"/>
        <chargeback>false</chargeback>
        <paymentWay>CARD</paymentWay>
      </return>
    </ns1:getOrderStatusExtendedResponse>
  </soap:Body>
</soap:Envelope>

```

2.4.2 REST

2.4.2.1 Request for order registration (REST)

The `register.do` request is used to register an order

URL (test): <https://web.rbsuat.com/ab/rest/register.do>

URL (prod): <https://pay.alfabank.ru/payment/rest/register.do>

Request parameters:

Name	Type	Mandatory	Description
userName	AN..30	no (either login/passw ord or token must be specified)	AN..30
password	AN..30	no (either login/passw ord or token must be specified)	AN..30
token	AN..30	no (either login/passw ord or token must be specified)	ANS..30

Name	Type	Mandatory	Description
orderNumber	AN..32	yes	Identifier of an order in the Store system, it is unique for each Store within the system
amount	N..12	yes	Payment amount in minimum currency units (cents or kopeks)
feeInput	N..8	no (see description)	<p>Commission amount in minimum currency units.</p> <div style="border: 1px solid #add8e6; padding: 5px; margin-top: 10px;"> <p>The parameter is passed only when the corresponding permission is enabled: Merchant commission transfer allowed.</p> </div>
currency	N3	no	ISO 4217 code of the payment currency. If not specified, default value is used.
returnUrl	ANS..51 2	yes	<p>The address to which the user is to be redirected in case of a successful payment. The URL must be specified in full including the protocol used (for example, https://test.ru instead of test.ru). Otherwise, the user will be redirected to the address of the following type: http://<payment_gateway_address>/<merchant_address>.</p> <div style="border: 1px solid #add8e6; padding: 5px; margin-top: 10px;"> <p>See also the 3DS-payment block for one-phase and two-phase payments. <code>returnUrl</code> is used in the same way as in Step 11 in these diagrams.</p> </div>
failUrl	ANS..51 2	no	<p>The address to which the user is to be redirected in case of a failed payment. The address must be specified in full including the protocol used (for example, https://test.ru instead of test.ru). Otherwise, the user will be redirected to the address of the following type: http://<payment_gateway_address>/<merchant_address>.</p> <div style="border: 1px solid #add8e6; padding: 5px; margin-top: 10px;"> <p>See also the 3DS-payment block for one-phase and two-phase payments. <code>failUrl</code> is used in the same way as in Step 11 in these diagrams.</p> </div>
dynamicCallbackUrl	ANS..51 2	no	<p>This parameter enables dynamic callback functionality. All payment-related callbacks enabled for Merchant could be directed to a URL passed in this parameter. Payment-related notifications are callback notifications about the following events: successful hold, payment rejected by timeout, cardpresent payment rejected, successful debiting, refund, reversal. Notably, payment-unrelated</p>

Name	Type	Mandatory	Description
			<p>callbacks that are active for the Merchant (enabling/disabling a binding, binding creation) will be sent to static callback URL.</p> <div style="border: 1px solid #add8e6; padding: 5px; margin: 5px 0;"> <p>Note To use the functionality of dynamic callback notifications, the Merchant must have the appropriate permission enabled for them: Callback type: Dynamic (CALLBACK_TYPE = DYNAMIC).</p> </div> <div style="border: 1px solid #ffff00; padding: 5px; margin: 5px 0;"> <p>To receive callbacks, the Merchant must have a specific permission enabled for them: Sending callback notification is allowed.</p> </div>
description	ANS..598	no	<p>Description of the Order in any format.</p> <div style="border: 1px solid #add8e6; padding: 5px; margin: 5px 0;"> <p>To enable sending this field to the processing system, contact the technical support service.</p> </div>
language	A2	no	<p>Language in the ISO 639-1 encoding. If the language is not specified, the default language specified in the Store settings is used.</p>
pageView	ANS..20	no	<p>The value of this parameter defines what kind of pages of the payment interface will be loaded for the customer. The possible values are:</p> <ul style="list-style-type: none"> • DESKTOP – pages designed to be displayed on desktop computers (pages with names like <code>payment_<locale>.html</code> and <code>errors_<locale>.html</code> will be searched for in the payment service archive); • DESKTOP – pages designed to be displayed on mobile devices (pages with names like <code>mobile_payment_<locale>.html</code> and <code>mobile_errors_<locale>.html</code> will be searched for in the payment service archive); • If the store has added arbitrary prefixes to their page file names, value of the desired prefix should be passed in the <code>pageView</code> parameter for the corresponding pages to be loaded. For example, if <code>iphone</code> is passed in the parameter, pages with names like <code>iphone_payment_<locale>.html</code> and <code>iphone_error_<locale>.html</code> will be searched for in the payment service archive. <p>Where:</p>

Name	Type	Mandatory	Description						
			<p><code>locale</code> – page language as ISO 639-1 code. For example, <code>ru</code> for Russian or <code>en</code> for English.</p> <p>If the parameter is missing or its value is in incorrect format, the default value is <code>pageView=DESKTOP</code>.</p>						
<code>clientId</code>	AN..255	no	<p>Identifier of the Customer in the Store system. Is used to facilitate bindings. It can be present if creating bindings is allowed for the Store.</p> <div style="border: 1px solid red; padding: 5px; background-color: #ffe6e6;"> <p>Specifying this parameter when processing payments with the use of bindings is mandatory. Otherwise, a payment will be unsuccessful.</p> </div>						
<code>merchantLogin</code>	AN..255	no	To register an order on behalf of a child Merchant, specify the Merchant login in this parameter.						
<code>jsonParams</code>	See description	no	<p>Block for passing Merchant parameters. Fields of additional information for further storage are passed in the following form.</p> <pre>{ "<name1>": "<value1>", ..., "<nameN>": "<valueN>" },</pre> <p>These fields can be passed to the Bank processing system to be subsequently displayed in the registries.*</p> <p>This functionality can be enabled upon additional agreement with the bank during the integration period.</p> <p>If sending notifications from Merchant to Customer is enabled, the Customer's email address must be transferred in this block in the <code>email</code> parameter.</p> <p>Data type</p> <table border="1" style="width: 100%;"> <thead> <tr> <th>Name</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td><code>name</code></td> <td>ANS..255 bytes</td> </tr> <tr> <td><code>value</code></td> <td>ANS..1024 bytes</td> </tr> </tbody> </table> <div style="border: 1px solid blue; padding: 5px; background-color: #e6f2ff;"> <p>In case of payments for housing and communal services, the following additional parameters must be passed:</p> <ul style="list-style-type: none"> <code>payment_doc_id</code> – the identifier of the payment document; <code>order_num</code> - the order number. </div>	Name	Type	<code>name</code>	ANS..255 bytes	<code>value</code>	ANS..1024 bytes
Name	Type								
<code>name</code>	ANS..255 bytes								
<code>value</code>	ANS..1024 bytes								

Name	Type	Mandatory	Description
sessionTimeoutSecs	N...9	no	<p>Order lifetime in seconds.</p> <p>If the parameter is not specified, the value specified in the Merchant settings or the default value (1200 seconds = 20 minutes) will be used.</p> <p>If the request contains the <code>expirationDate</code> parameter, then the value of <code>sessionTimeoutSecs</code> parameter is disregarded.</p>
expirationDate	ANS	no	<p>The date and time of the order lifetime expiration. Format: <code>yyyy-MM-ddTHH:mm:ss</code>.</p> <p>If this parameter is not present in the request, the <code>sessionTimeoutSecs</code> parameter is used to define the time of the end of life of the order.</p>
bindingId	AN..255	no	<p>Identifier of the binding created earlier. It can be used only if the store has the permission to work with bindings. If this parameter is passed in the given request, it means that:</p> <ol style="list-style-type: none"> 1. The given order can be paid only using a binding; 2. The Payer will be redirected to the payment page on which only entering CVC is required.
features	ANS..255	no	<p>The following values can be used.</p> <p><code>AUTO_PAYMENT</code> - Payment is made without authentication of the card holder (without CVC and 3-D Secure). To make such payments, the Merchant must have the respective permissions.</p> <p><code>VERIFY</code> - if this parameter is specified, after the request for order registration, the cardholder is to be verified without debiting funds from the cardholder account. Thus it is possible to pass a zero amount in the request. This verification allows the Merchant to ensure that a card belongs to the cardholder and to debit this card in the future without verifying authentication data (CVC, 3-D Secure) when processing subsequent payments.</p> <div style="border: 1px solid black; background-color: #e6f2ff; padding: 10px; margin: 10px 0;"> <p>Specifics of passing the VERIFY value</p> <ul style="list-style-type: none"> • Even if the payment amount is to be passed in the request, it will not be debited from the buyer's account. • After the order has been successfully registered, it is put to the REVERSED (cancelled) status. </div> <p><code>FORCE_TDS</code> - use of 3-D Secure is enforced for the payment. If the card does not support 3-D Secure, the transaction will fail.</p> <p><code>FORCE_SSL</code> - use of SSL is enforced for the payment (no 3-D Secure).</p> <p><code>FORCE_FULL_TDS</code> - after authentication using 3-D Secure, the PaRes status must only be Y, which</p>

Name	Type	Mandatory	Description
			guarantees successful user authentication. Otherwise, the transaction will fail.
email	ANS..40	no	Customer's email address.
phone	ANS.12	no	Customer's phone number. If the phone includes a country code, the number must begin with a plus sign ("+"). If the phone is passed without a plus sign ("+"), the country code should not be specified. Thus, the following options are valid: <ul style="list-style-type: none"> • +79998887766; • 9998887766.

* By default the following fields are passed to the bank processing system:

- `orderNumber` – order number in the store system;
- `description` – order description (no more than 99 characters; %, +, carriage return \r, and line feed \n cannot be used).

Response parameters:

Name	Type	Mandatory	Description
orderId	ANS36	No	Identifier of the order in the payment system. It is unique within the system. The identifier is missing if the order registration failed due to an error detailed in <code>errorCode</code> .
formUrl	AN..512	No	URL of the payment form to which the Customer's browser is to be redirected. This parameter is not returned if the registration of the order was not successful due to the error described in <code>errorCode</code> .
errorCode	N3	No	Error code.
errorMessage	AN..512	no	Error description in the language passed in the <code>language</code> parameter in the request.

Error codes (`errorCode` field):

Value	Description
0	Request processing took place without system errors
1	Order number is duplicated, order with given order number is processed already
1	The order with specified number is registered but not paid
1	Unknown order ID
3	Currency is not defined
4	Order number is empty
4	Empty merchant user name
4	Empty amount
4	Empty return URL
4	Password can't be empty

Value	Description
5	Merchant login is incorrect.
5	Invalid amount
5	Invalid merchant language
5	Access denied
5	The user must change his password
5	Access denied
5	[jsonParams] is incorrect.
7	System error
13	It is not allowed to use both FORCE_TDS/FORCE_SSL and AUTO_PAYMENT
13	The Merchant does not have the permission to process auto-payments
13	The Merchant does not have the permission to process verification payments
14	Features are specified incorrectly.

POST request example

```
https://web.rbsuat.com/ab/rest/register.do?userName=loginapi&password=login*%3F1&amount=100&currency=810&language=ru&orderNumber=87654321&returnUrl=http://yoursite.com&pageView=DESKTOP&jsonParams={"param1":"value1","param2":"value2"}&expirationDate=2014-09-08T14:14:14
```

Response example:

```
{"orderId":"70906e55-7114-41d6-8332-4609dc6590f4","formUrl":"https://server/application_context//merchants/test/payment_ru.html?mdOrder=70906e55-7114-41d6-8332-4609dc6590f4"}
```

2.4.2.2 Request to receive a QR code (REST)

The request used to get the QR code is `get.do`.

URL (test): <https://web.rbsuat.com/ab/rest/sbp/c2b/qr/dynamic/get.do>

URL (prod): <https://pay.alfabank.ru/payment/rest/sbp/c2b/qr/dynamic/get.do>

Request format: POST

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

Request parameters

Parameter	Type	Mandatory	Description
<i>Parameters in the request body, format is x-www-form-urlencoded:</i>			
userName	AN..30	Yes	Login issued to the Store upon integration with the Payment Gateway.
password	AN..30	Yes	Password issued to the Store upon integration with the Payment Gateway.

Parameter	Type	Mandatory	Description
mdOrder	ANS..36	Yes	Order number in the Payment Gateway system.
qrHeight	String	No	The height of the QR code in pixels. Specify if <code>renderedQR</code> is required. Minimum value: 10. Maximum value: 1000
qrWidth	String	No	The width of QR code. Specify if <code>renderedQR</code> is required. Minimum value: 10. Maximum value: 1000.
qrFormat	ANS*	No	The possible values are: matrix - returns a matrix of zeros and ones; image - returns the image in base64.

Response parameters

Parameter	Type	Mandatory	Description
errorCode	ANS..3	No	Error code.
errorMessage	AN..512	No	Error description.
payload	String	No	The content of the QR code registered in the SBP. Present if value of <code>qrStatus</code> is <code>STARTED</code> .
qrId	String	No	ID of the QR code.
qrStatus	String	No	Status of the QR code request. The available values are: <ul style="list-style-type: none"> <code>STARTED</code> - QR code generated; <code>CONFIRMED</code> - the order is accepted for payment; <code>REJECTED</code> - payment declined; <code>REJECTED_BY_USER</code> - payment by QR code was rejected by the Merchant; <code>ACCEPTED</code> - order has been paid.

Request format

```
POST https://web.rbsuat.com/ab/rest/sbp/c2b/qr/dynamic/get.do

userName=userName&password=password&mdOrder=04888d6f-7920-7531-8332-8de901efddd0&qrFormat=image
```

Response format

```
{
  "qrId": "54d14bae6f7f4a73929308e9afa5915d",
  "payload":
  "https://qr.nspk.ru/54d14bae6f7f4a73929308e9afa5915d?type=02&bank=600000000015&sum=13000&cur=RUB&crc=464g",
  "qrStatus": "STARTED"
}
```

2.4.2.3 Request for status of payment by QR code (REST)

The requests used to get the status of a request in the SBP is `status.do`.

URL (test): <https://web.rbsuat.com/ab/rest/sbp/c2b/qr/status.do>

URL (prod): <https://pay.alfabank.ru/payment/rest/sbp/c2b/qr/status.do>

Request format: POST

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

Request parameters

Parameter	Type	Mandatory	Description
<i>Parameters in the request body, format is x-www-form-urlencoded:</i>			
userName	AN..30	Yes	Login issued to the Store upon integration with the Payment Gateway.
password	AN..30	Yes	Password issued to the Store upon integration with the Payment Gateway.
mdOrder	ANS..36	Yes	Order number in the Payment Gateway system.
qrId	String	Yes	ID of the QR code.

Response parameters

Parameter	Type	Mandatory	Description
errorCode	ANS..3	No	Error code.
errorMessage	AN..512	No	Error description.
qrStatus	String	No	Status of the QR code request. The available values are: <ul style="list-style-type: none"> STARTED - QR code generated; CONFIRMED - the order is accepted for payment; REJECTED - payment declined; REJECTED_BY_USER - payment by QR code was rejected by the Merchant; ACCEPTED - order has been paid.
qrType	String	No	QR code type: <ul style="list-style-type: none"> STATIC - static; DYNAMIC - dynamic. Currently, the value returned can be only DYNAMIC.
transactionState	String	No	Order status: <ul style="list-style-type: none"> CREATED - order is created; DECLINED - order is declined; DEPOSITED – order is paid;

Request format

```
POST https://web.rbsuat.com/ab/rest/sbp/c2b/qr/status.do

userName=userName&password=password&mdOrder=04888d6f-7920-7531-8332-8de901efddd0&qrId=3946c0c02d1042f7b7e63cc0f1b52a95
```

Response format

```
{
  "qrType": "DYNAMIC",
  "qrStatus": "ACCEPTED",
  "transactionState": "DEPOSITED"
}
```

2.4.2.4 Request for order status (REST)

The request used for getting the status of a registered order is `getOrderStatusExtended.do`.

URL (test): <https://web.rbsuat.com/ab/rest/sbp/c2b/qr/getOrderStatusExtended.do>

URL (prod): <https://pay.alfabank.ru/payment/rest/sbp/c2b/qr/getOrderStatusExtended.do>

Request parameters:

Name	Type	Mandatory	Description
userName	AN..30	yes	The Store login received on the connection
password	AN..30	yes	The Store password received on the connection
orderId	ANS36	yes*	Identifier of the order in the payment system. It is unique within the system.
orderNumber	AN..32	yes*	Order identifier in the Store system.
language	A2	no	Language in the ISO 639-1 encoding. If this parameter is not specified, it is considered that the language is Russian. The error message will be returned in that language.

Either `orderId` or `orderNumber` must be present in the request. If both parameters are present in the request, `orderId` has priority.

Several sets of response parameters exist. The exact sets of parameters to be returned depend on the version of `getOrderStatusExtended` specified in the merchant settings.

Name	Type	Mandatory	Description	Version of <code>getOrderStatusExtended</code>
orderNumber	AN..32	yes	Order identifier in the Store system.	All versions.
orderStatus	N2	no	The order status in the payment system is defined by the value of this parameter. The list of possible values is given below. It is missing if the order has not been found. <ul style="list-style-type: none"> 0 - the order is registered, but not paid; 1 - the pre-authorized amount is put on hold (for two-phase payments); 	All versions.

Name	Type	Mandatory	Description	Version of getOrderStatusExtended
			<ul style="list-style-type: none"> • 2 - a full authorization of the order amount has been performed; • 3 - the authorization was cancelled; • 4 - a refund operation has been processed for the transaction; • 5 - an authorization through ACS of the issuing bank has been initiated; • 6 - the authorization was declined 	
actionCode	N3	yes	Response code.	All versions.
actionCodeDescription	AN..512	yes	Error description in the language passed in the 'language' parameter of the request.	All versions.
errorCode	ANS..3	no	<p>Error code. The following values are available.</p> <ul style="list-style-type: none"> • 0 - the request has been processed without system errors.; • 1 - Expected [orderId] or [orderNumber]; • 5 - Access denied; • 5 - The user must change his password; • 6 - Order not found; • 7 - System error. 	All versions.
errorMessage	AN..512	no	Error description in the language passed in the 'language' parameter of the request.	All versions.
amount	N..20	yes	Payment amount in minimum currency units (cents or kopeks)	All versions.
currency	N3	no	ISO 4217 code of the payment currency. If the code is not specified, the default value is 810 (Russian rubles).	All versions.
date	ANS	yes	Order registration date.	All versions.
depositedDate	N	no	Date of order payment in Unix-time format (POSIX-time).	10 and higher.

Name	Type	Mandatory	Description	Version of getOrderStatusExtended
orderDescription	AN..512	no	Order description passed upon its registration.	All versions.
ip	ANS..39	yes	IP address of the user who paid for the order. IPv6 is supported in all requests.	All versions.
authRefNum	AN..24	no	Registration number of the payment authorization that has been assigned to it on the payment registration.	02 and higher.
refundedDate	ANS	no	Date and time of the refund.	13 and more.
paymentWay	AS..14	yes	The way of order completion (a payment with entering card data, a payment using a binding, etc.). This parameter specifies the payment method. Possible values: <ul style="list-style-type: none"> CARD - payment with entering of card details; CARD_BINDING - payment by binding; CARD_MOTO - payment through call center; CARD_PRESENT - payment by cardPresent; ALFA_ALFACLICK – for a payment with "Alfa-click" (through the PayByClik system); UPOP – for a payment through the UPOP system, for the holders of the China Union Pay cards. FILE_BINDING - payment by file; SMS_BINDING - payment by SMS; P2P - transfer from card to card; P2P_BINDING - transfer by binding; PAYPAL - payment from PayPal; 	09 and higher.

Name	Type	Mandatory	Description	Version of getOrderStatusExtended
			<ul style="list-style-type: none"> • MTS - payment from MTS account; • APPLE_PAY - Apple Pay; • APPLE_PAY_BINDING - payment by Apple Pay binding; • ANDROID_PAY - Android Pay; • ANDROID_PAY_BINDING - payment by Android Pay binding; • GOOGLE_PAY_CARD - Google Pay, non-tokenized; • GOOGLE_PAY_CARD_BINDING - payment by binding with non-tokenized Google Pay card; • GOOGLE_PAY_TOKENIZED - Google Pay, tokenized; • GOOGLE_PAY_TOKENIZED_BINDING - payment by binding with tokenized Google Pay card; • SAMSUNG_PAY - Samsung Pay; • SAMSUNG_PAY_BINDING - payment by Samsung Pay binding; • IPOS - iPOS payment. 	
avsCode	A1	no	<p>AVS Response Code - AVS response code (verification of card holder's registration address and postal code). The possible values are:</p> <ul style="list-style-type: none"> • A - Both address and postal code match; • B - Address matches, postal code does not match; • C - Postal code matches, address does not match; • D - Neither address nor postal code match; • E - Verification requested but not available; 	19 and higher

Name	Type	Mandatory	Description	Version of getOrderStatusExtended
			<ul style="list-style-type: none"> F - Transaction improperly requests verification. 	
<p>The <code>merchantOrderParams</code> element is present in a response if an order contains additional parameters of the Merchant. Each additional order parameter is contained in a separate <code>merchantOrderParams</code> element.</p>				
<code>name</code>	AN..20	no	Name of the additional parameter	All versions.
<code>value</code>	AN..10 24	no	Value of the additional parameter	All versions.
<p><code>cardAuthInfo</code> element. Contains a structure consisting of a list of <code>secureAuthInfo</code> element and the following parameters:</p>				
<code>pan</code>	NS..19	no	Masked number of the card that has been used for the payment.	All versions.
<code>expiration</code>	N6	no	Card expiration date in YYYYMM format. This parameter is to be specified only after the order has been paid.	All versions.
<code>cardholderName</code>	A..64	no	Name of the cardholder. This parameter is to be specified only after the order has been paid.	All versions.
<code>approvalCode</code>	AN6	no	Payment authorization code. This field has a fixed length (six characters) and can contain digits and Latin letters. This parameter is to be specified only after the order has been paid.	All versions.
<code>chargeback</code>	A..5	no	Whether the funds were forcibly returned to the buyer by the bank. The available values are: <ul style="list-style-type: none"> true false 	06 and higher.
<code>paymentSystem</code>	N..10	yes	The payment system name. Available values: <ul style="list-style-type: none"> VISA; MASTERCARD; AMEX; JCB; CUP; MIR. 	08 and higher.

Name	Type	Mandatory	Description	Version of getOrderStatusExtended
product	AN..255	yes	Additional details on corporate cards. These details are filled in by the technical support service through the administrative console. If such details are missing, an empty value is returned.	08 and higher.
productCategory	string	yes	Additional details about corporate cards. These details are filled in by the technical support service through the administrative console. If such details are missing, an empty value is returned. The possible values are: DEBIT, CREDIT, PREPAID, NON_MASTERCARD, CHARGE, DIFFERED_DEBIT.	17
The <code>secureAuthInfo</code> element contains the <code>eci</code> element and the <code>threeDSInfo</code> element, which is a list of parameters <code>cavv</code> and <code>xid</code>):				
eci	N..4	no	Electronic Commerce Indicator. The indicator is specified only after an order has been paid and in case the corresponding permission is present.	All versions.
cavv	ANS..200	no	The value for the cardholder and card authentication check. The indicator is specified only after an order has been paid and in case the corresponding permission is present.	All versions.
xid	ANS..80	no	Electronic Commerce Indicator of the transaction. The indicator is specified only after an order has been paid and in case the corresponding permission is present.	All versions.
The <code>bindingInfo</code> contains the following parameters.				
clientId	AN..255	no	Identifier of the Customer in the Store system passed on the order registration. This parameter is present only if a Store has the permission to create bindings.	All versions.
bindingId	AN..255	no	Identifier of a binding created on an order payment or used to pay for an order. This parameter is present only if a Store has the permission to create bindings.	All versions.
authDateTime	ANS	no	Date and time of authorization.	02 and higher.
terminalId	AN..10	no	Terminal ID.	02 and higher.

Name	Type	Mandatory	Description	Version of getOrderStatusExtended
The <code>paymentAmountInfo</code> contains the following parameters.				
<code>approvedAmount</code>	N..20	no	The amount put on hold on the card (is used only for two-phase payments).	03 and higher.
<code>depositedAmount</code>	N..20	no	The amount authorized to be debited from the card.	03 and higher.
<code>refundedAmount</code>	N..20	no	The refund amount.	03 and higher.
<code>paymentState</code>	A..10	no	The order status.	03 and higher.
<code>feeAmount</code>	N..20	no	Fee amount.	11 and higher.
<code>totalAmount</code>	N..20	no	Order amount + fee (if applicable to the order).	18 and higher.
The <code>bankInfo</code> contains the following parameters.				
<code>bankName</code>	AN..20 0	no	Issuing bank name.	03 and higher.
<code>bankCountryCode</code>	AN..4	no	Country codes of the issuing bank.	03 and higher.
<code>bankCountryName</code>	AN..16 0	no	Name of the country of the issuing bank passed in the 'language' parameter of the request or in the language of the user who has called the method if the language has not been specified in the request.	03 and higher.
The <code>refunds</code> block contains information about refunds. Is added in response to <code>getOrderStatusExtended</code> request Version 05 and higher, and is only present if there is a refund for an order.				
Parameters of the block:				
Name	Type	Mandatory	Description	Version of getOrderStatusExtended
<code>referenceNumber</code>	N12	no	Transaction reference number assigned by the Payment Gateway upon transaction completion.	05 and higher
<code>actionCode</code>	N..5	no	Response codes - a digital indication of the result that resulted from the user's access to the system.	05 and higher
<code>amount</code>	N..12	no	The refund amount in minimum currency units.	05 and higher
<code>date</code>	ANS	no	Date of order registration in Unix-time format (POSIX-time).	05 and higher

POST request example

orderId=b9054496-c65a-4975-9418-1051d101f1b9&language=ru&merchantOrderNumber=0784sse49d0s134567890

Response example:

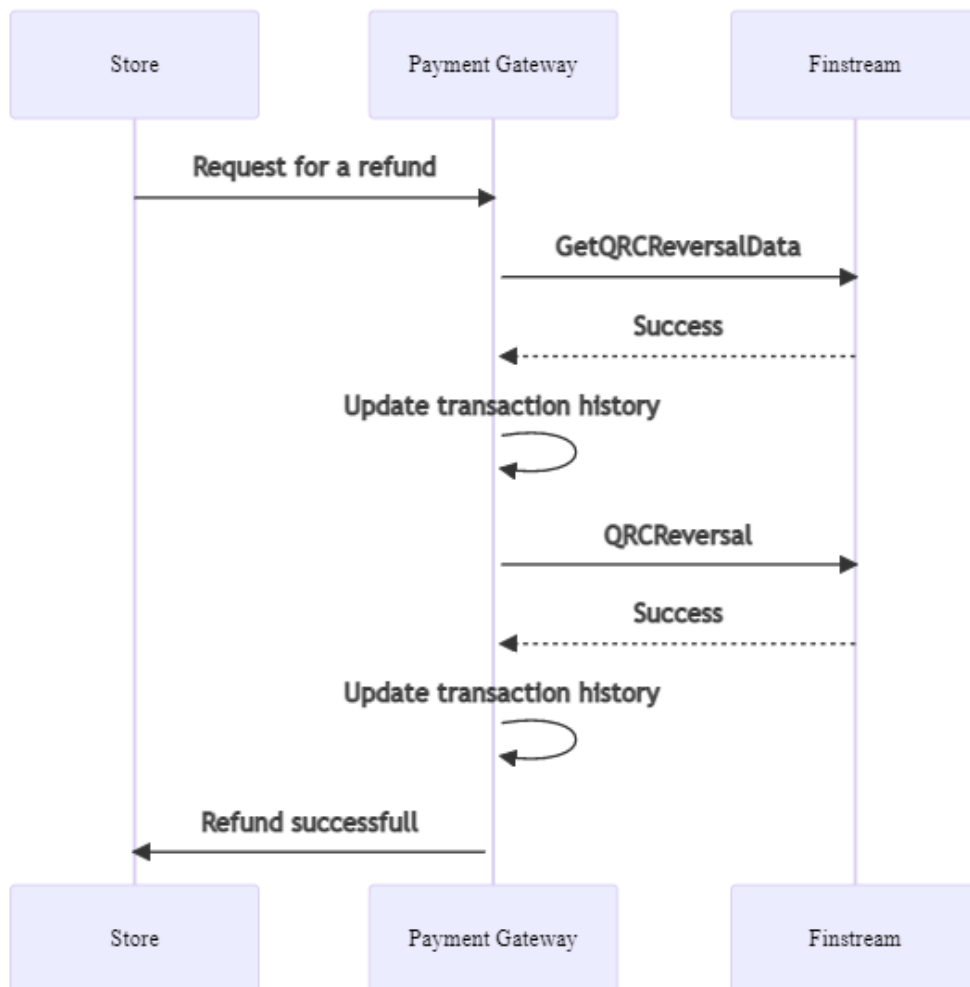
```
{ "errorCode": "0", "errorMessage": "Успешно", "orderNumber": "0784sse49d0s134567890", "orderStatus": 6, "actionCode": -2007, "actionCodeDescription": "Время сессии истекло", "amount": 33000, "currency": "810", "date": 1383819429914, "orderDescription": "", "merchantOrderParams": [{"name": "email", "value": "yap"}], "attributes": [{"name": "mdOrder", "value": "b9054496-c65a-4975-9418-1051d101f1b9"}], "cardAuthInfo": {"expiration": "201912", "cardholderName": "Ivan", "secureAuthInfo": {"eci": 6, "threeDSInfo": {"xid": "MDAwMDAwMDEzODM4MTk0MzAzMjM="}}, "pan": "411111**1111"}, "terminalId": "333333" }
```

3 Refunds

Note: Refunds are possible for Merchants who have been assigned the appropriate permissions in the Payment Gateway.

3.1 Refunds via API

3.1.1 Scenario for a refund via API



Description:

1. The Merchant sends a request for a refund to the Payment Gateway. Specification of request is available in the following sections:
 - "[3.1.2.1. Request for refund \(SOAP\)](#)";
 - "[3.1.2.2. Request for refund \(REST\)](#)".
2. The Payment Gateway sends a request to the Finstream system checking the possibility of refund.
3. After receiving a response, the transaction history is updated in the Payment Gateway.

4. The case a response indicates that the refund is possible, the Payment Gateway sends to Finstream a request for a refund.
5. After receiving a response, the transaction history is updated in the Payment Gateway.
6. The Payment Gateway returns the result of processing of refund operation.

If an error is received at Step 2 or Step 4, then after updating the transaction history, the Payment Gateway sends to the Merchant's system an `errorCode=7` - "System error" .

3.1.2 Refund API description

3.1.2.1 Request for refund (SOAP)

The request used for making a refund is `refundOrder`.

WSDL (test): <https://web.rbsuat.com/ab/webservices/merchant-ws?wsdl>

WSDL (prod): <https://pay.alfabank.ru/payment/webservices/merchant-ws?wsdl>

Upon this request, the funds for the specified order are to be returned to the payer. The request will end with an error if the funds have not been debited for this order. The system permits multiple refunds, but the total refunded amount must not exceed the originally debited amount.

When making a refund for payment of housing and communal services, only a full refund is possible.

To process a refund operation, it is necessary to have the corresponding permissions in the system.

Request parameters:

Name	Type	Mandatory	Description
<code>orderId</code>	ANS36	yes	Identifier of the order in the payment system. It is unique within the system.
<code>refundAmount</code>	N..12	yes	The refund amount in order currency. May be less than or equal to the balance in the order.
<code>language</code>	A2	no	Language in the ISO 639-1 encoding. If this parameter is not specified, it is considered that the language is Russian. The error message will be returned in that language.

Response parameters:

Name	Type	Mandatory	Description
<code>errorCode</code>	N3	no	Error code.
<code>errorMessage</code>	AN..512	no	Error description in the language passed in the 'language' parameter of the request.

Error codes (`errorCode` field):

Value	Description
0	Request processing took place without system errors
5	Access denied
5	The user must change the password
5	[<code>orderId</code>] is empty

Value	Description
5	Invalid amount
6	Unknown order ID
7	Payment must be in the correct state
7	Refund amount exceeds deposited amount
7	System error

Request example:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:mer="http://engine.paymentgate.ru/webservices/merchant">
  <soapenv:Header/>
  <soapenv:Body>
    <mer:refundOrder>
      <order language="ru" orderId="4302d369-a5e8-4432-a5e5-
42acfab52c86" refundAmount="20000">
        <!--Zero or more repetitions:-->
        <params name=" " value=" "/>
      </order>
    </mer:refundOrder>
  </soapenv:Body>
</soapenv:Envelope>
```

Response example:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:refundOrderResponse
xmlns:ns1="http://engine.paymentgate.ru/webservices/merchant">
      <return errorCode="7" errorMessage="Refund amount exceeds
deposited amount"/>
    </ns1:refundOrderResponse>
  </soap:Body>
</soap:Envelope>
```

3.1.2.2 Request for refund (REST)

The request used for making a refund is `refund.do`.

URL (test): <https://web.rbsuat.com/ab/rest/refund.do>

URL (prod): <https://pay.alfabank.ru/payment/rest/refund.do>

Upon this request, the funds for the specified order are to be returned to the payer. The request will end with an error if the funds have not been debited for this order. The system permits multiple refunds, but the total refunded amount must not exceed the originally debited amount.

When making a refund for payment of housing and communal services, only a full refund is possible.

To process a refund operation, it is necessary to have the corresponding permissions in the system.

Request parameters:

Name	Type	Mandatory	Description						
userName	AN..30	yes	The Store login received on the connection						
password	AN..30	yes	The Store password received on the connection						
orderId	ANS36	yes	Identifier of the order in the payment system. It is unique within the system.						
amount	N..12	yes	Payment amount in minimum currency units (cents or kopeks)						
language	A2	no	Language in the ISO 639-1 encoding. If this parameter is not specified, it is considered that the language is Russian. The error message will be returned in that language.						
jsonParams	See description	no	<p>The fields for additional information to be retained: {"param": value, "param2":value2}. These fields can be passed to the bank's processing system for subsequent display in the registers.* This functionality can be enabled by agreement with the bank during the integration period.</p> <p>Data type</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>ANS..255 bytes</td> </tr> <tr> <td>value</td> <td>ANS..1024 bytes</td> </tr> </tbody> </table> <p>It is forbidden to pass reserved names in the parameter (if they are passed, the order may be rejected):</p> <ul style="list-style-type: none"> • sbrf_spasibo:amount_bonus • sbrf_sbermiles:amount_bonus • loyaltyId 	Name	Type	name	ANS..255 bytes	value	ANS..1024 bytes
Name	Type								
name	ANS..255 bytes								
value	ANS..1024 bytes								

Response parameters:

Name	Type	Mandatory	Description
errorCode	N3	No	Error code.
errorMessage	AN..512	No	Error description.

Error codes (errorCode field):

Classification of error codes:

Value	Description
0	Request processing took place without system errors
5	Incorrect value of a request parameter
6	Unregistered OrderId
7	System error

Meaning of the error codes:

Value	Description
0	Request processing took place without system errors
5	Access denied
5	The user must change his password
5	[orderId] is empty
6	Unknown order ID
7	Payment must be in the correct state
7	Incorrect deposit amount (less than one ruble)
7	System error

POST request example

```
https://web.rbsuat.com/ab/rest/refund.do?userName=loginapi&password=login*%3F1&amount=500&currency=810&language=ru&orderId=5e 97e3fd-1d20-4b4b-a542-f5995f5e8208
```

Response example:

```
{"errorCode":0}
```

3.2 Refunds via Merchant Portal

Refunds for payments via SBP can be made on Merchant Portal.

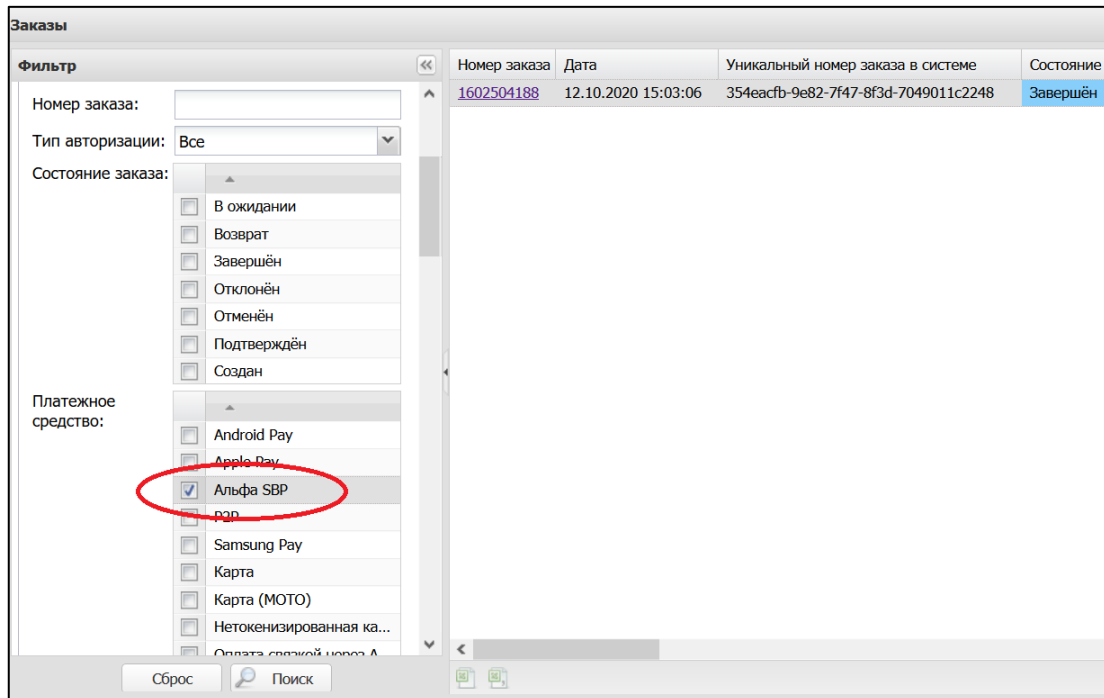
The description of the refund process through the Merchant Portal is presented in the document "[Instructions for working with the console \(MPortal\)](#)" (see "6.3.3.2 Refunds").

The procedure for looking up orders paid via SBP is covered in "[4. Looking up orders paid via SBP](#)" below.

4 Looking up orders paid via SBP

4.1 Looking up orders

To look up orders paid through SBP in the Merchant Portal, on the order filter panel, select "Alfa SBP" in the "Payment way" field:



For details on order filter, see the document "[Instructions for working with the console \(MPortal\)](#)". "6.1 Search for orders using a filter").

4.2 Order history

SBP operations are recorded in the order history.

For details on viewing order history in Merchant Portal, see the document "[Instructions for working with the console \(MPortal\)](#)". "6.3.3.1 History").

5 Fiscalization of transactions

A fiscal receipt is issued for successful payments.

In case of a successful refund, a fiscal receipt for a refund is issued.

6 Dummy mode on the test bench

The fp-dummy stub emulates the behavior of the SBP system.

Payment on test payment page:

1. Choose payment via SBP;
2. A QR code will be displayed on the screen;
3. After a few seconds, the result of the payment will be shown on the screen.

The result of payment depends on the amount of the order:

- the amount is less than 500 rubles - payment is successful.
- the amount is greater than 500 rubles - payment is not successful.

If a non-standard payment page is used, the button for payment via SBP may not be available. In this case, you should contact the bank.