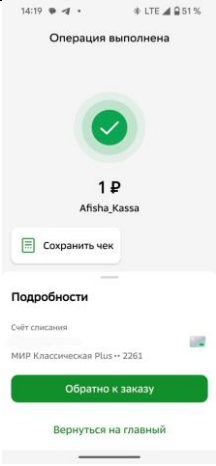
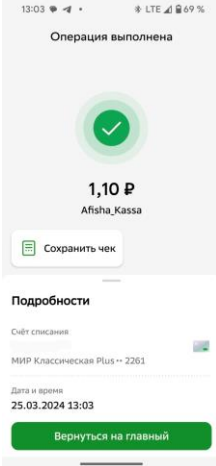


SBP (Fast Payment System) Payment Ecommerce Gateway Specifics

#	Scenario	Action	Result
Order registration			
1	Creating an order for subsequent SBP payment via the Bank's app or for subsequent payment by a previously created subscription	register.do request specifying in jsonParams: "qrType" : "DYNAMIC_QR_SBP", "sbp.scenario" : "C2B" "description": "SBP order registration with a maximum set of fields" (example) "amount": <Order amount in copecks> Description - Optional field when paying via the Bank's app. If specified, the content is displayed to the individual client in SBOL (as well as in other Banks' apps) in the "Purpose of Payment" field and transferred to the partner in the ID1 field register when paying for the order (Add. information_1) If you plan to pay by subscription, the "Description" field must be filled in.	{ "errorCode": "0", "externalParams": { "qrcId": "AD10001RQ100T70E8LEOT61G16J41I09", }, "sbpPayload": "https://qr.nspk.ru/AD10001RQ100T70E8LEOT61G16J41I09?type=02&bank=10000000111&sum=1400&cur=RUB&crc=62E3", "sbolDeepLink": "sberpay://invoicing/v2?bankInvoiceId=a07ec65536924addb531622df735b60a&operationType=Web2App&option=Connect" }, "orderId": "a07ec655-3692-4add-b531-622df735b60a", "formUrl": "https://ift.payecom.ru/pay_ru?orderId=a07ec655-3692-4add-b531-622df735b60a", "errorMessage": "Request processed with no system errors" } SBP dynamic QR type: 02 (type=02)
2	Creating SBP subscription	register.do request specifying in jsonParams: "qrType" : "SUBSCRIPTION_SBP", "sbp.scenario" : "C2B_SUBSCRIPTION", "sbp.subscriptionPurpose" : "To pay for services" (example)	{ "orderId": "06dc6ea3-c932-4b42-b9d9-6273022627b9", "formUrl": "https://ift.payecom.ru/pay_ru?orderId=06dc6ea3-c932-4b42-b9d9-6273022627b9", "errorCode": "0", "errorMessage": "Request processed with no system errors" "externalParams": { "qrcId": "AB1S002LBQ7PDTTE88HQ1SE2PVFVTP04", "sbpPayload": "https://sub.nspk.ru/AB1S002LBQ7PDTTE88HQ1SE2PVFVTP04?type=03&bank=10000000111&crc=3B02", "sbolDeepLink": "sberpay://invoicing/v2?bankInvoiceId=06dc6ea3c9324b42b9d96273022627b9&operationType=Web2App&option=Connect" } } SBP dynamic QR type: 03 (type=03)
3	SBP order creation with the option to create a subscription after payment	"qrType" : "DYNAMIC_QR_SBP", "sbp.scenario" : "C2B_SUBSCRIPTION_WITH_PAYMENT", ", "sbp.subscriptionPurpose" : "To pay for services", "description": "SBP order registration"	The response to register.do is similar to that in item 1 SBP dynamic QR type: 02 (type=02)
4	Displaying the "Back to Order" button in SBOL to go back to the	In the jsonParams block of the register.do request, returnUrl is a required field and it must be filled in.	The returnUrl specifies the address

<p>merchant app/website</p>	<p>If !=", it is transmitted to the NSPK upon SBP QR registration, and in case of successful payment in SBOL (RB of another bank), the "Back to Order" button (return to the store) will be displayed; when clicking this button, you will be redirected to the specified address.</p> <p>If =", then url is substituted from the settings of merch (return address after payment, POS web page)</p> <p>If = and setting addresses are zeroed out, then when registering the SBP QR, the parameter will not be transmitted and the "Back to the Order" button will not be displayed on the SBOL screen.</p> <p>Description from the NSPK website:</p> <p>Contains a link for automatic return of the Payer from the Bank's app to the merchant's app or website. Only ASCII characters are allowed. Format should comply with RFC-3986 specification</p> <p>If filled in, it must contain the domain. Examples:</p> <p>https://pay.mos.ru/mospaynew/newportal/public/result?requestUID=21445743-79b2-408d-8919-b33bfd64030e&packageUID=b5d2e493-5592-4b93-9be1-927c018983c5</p> <p>ozon://ozon.ru/thank_you/46601856-0076?clearBackStack=true</p>	 <p>Address not specified</p> 	
<p>Order payment, order return</p>			
<p>5</p>	<p>Payment using the SBP payment link (SBP QR)</p>	<p>SbpPayload received under item 1 is read in the Payer bank's app, the Client confirms the payment</p>	<p>Order status in response to getOrderStatusExtended.do (gose) orderStatus = 2 actionCode = 0</p>
<p>6</p>	<p>Payment using the SBP payment link (SBP QR) with subsequent creation of a subscription</p>	<p>SbpPayload received under item 3 is read in the payer bank's app and the client confirms the payment. After successful payment, the client creates a subscription</p>	<p>orderStatus = 2 actionCode = 0</p> <p>Subscription parameters in the attributes block are transferred (example) "subscriptionId": "3cbf9ed8cbb9442abdc1e2ff514ebb2e", "memberId": "10000000111</p>
<p>7</p>	<p>Creating an SBP subscription</p>	<p>SbpPayload received under item 2 is read in the Payer bank's app, the Client creates a subscription</p>	<p>orderStatus = 2 actionCode = 0</p> <p>Subscription parameters in the attributes block are transferred (example) "subscriptionId": "3cbf9ed8cbb9442abdc1e2ff514ebb2e", "memberId": "10000000111</p>
<p>8</p>	<p>Subscription fee</p>	<p>After completing item 1, the paymentOrderBySubscription method is called with the following parameters</p> <p>Order Identifier</p> <p>"orderId"</p>	<p>orderStatus = 2 actionCode = 0</p>

		<p>Identifier of the subscription for which the payment transaction is to be performed "subscriptionId": "3cbf9ed8cbb9442abdc1e2ff514break2e",</p> <p>Identifier of the payer bank in which the subscription is created "memberId": "100000000111"</p>	
9	Refund	<p>refund.do</p> <p>"orderId": "amount": _____ (partial refunds are allowed, the amount is specified in copecks)</p>	<p>orderStatus = 4 actionCode = 0</p>

Comments to the SBP OR (SBP payment link) payment process:

1. The partner creates an order using register.do specifying in jsonParams : "QRType" : "DYNAMIC_QR_SBP", "sbp.scenario" : "C2B", "description" field are optionally filled.
2. When processing an order creation request, Sber receives the SBP payment link (SbpPayload) in NSPK and transmits it to the partner in response to register.do;
3. The partner uses SbpPayload in its app/mWEB by transmitting it to the NSPK widget https://sbp.nspk.ru/file/sbp_SDK.zip (or using its widget) or generates and displays a QR code to the client;
4. Individual clients choose their banks in the issuing bank list generated by the widgets or scan the QR code, then go to their banks' app and confirm the payment.
4. After performing the payment transaction via the NSPK circuit, Sber receives a confirmation of payment receipt and assigns the order statuses corresponding to the payment result.
5. Sber sends the partner the order status information in response to getOrderStatusExtended.do (gose) query or to callback

Comments to the subscription creation process:

1. The partner creates an order using register.do specifying in jsonParams : QRType" : "SUBSCRIPTION_SBP", "sbp.scenario" : "C2B_SUBSCRIPTION", "sbp.subscriptionPurpose" : "To pay for services" (example)
2. When processing an order creation request, Sber receives the SBP payment link (SbpPayload) in NSPK and transmits it to the partner in response to register.do;
3. The partner uses SbpPayload in its app/mWEB, transmitting it to the NSPK widget (or using its own widget);
4. Individual clients choose their banks in the issuing bank list generated by the widgets, go to their bank's app and confirm the subscription creation.
4. The issuing bank sends the subscription creation information to the NSPK.
5. The NSPK sends a notification to Sber about the subscription creation by transmitting the SubscriptionId and memberId.
6. Sber saves subscription data in the order and transmits the created subscription information to the partner in response to getOrderStatusExtended.do (gose) or to callback.
7. Partner saves SubscriptionId and memberId binding to the client.

Note:

- According to the NSPK regulations, an individual client in a specific issuing bank can create only one subscription for a specific legal entity.
- If the subscription has already been created, when attempting to re-create the subscription, the transaction will be rejected in the issuing bank, unless the individual client wishes to change the account linked to the subscription. In this case, the subscription will be saved with a new account, but with the same SubscriptionId. The partner will receive a callback (response to gose) with the same SubscriptionId.
- Only individual clients can delete a subscription in their bank app.

Comments to the subscription payment process:

1. The Partner creates an order using the register.do method, indicating in jsonParams : "QRType" : "DYNAMIC_QR_SBP", "sbp.scenario" : "C2B ", mandatory filling out "description".
2. To pay based on the subscription, the partner calls the paymentOrderBySubscription method by indicating the order number orderId (received at the order registration stage), SubscriptionId and memberId in the parameters;
3. Sber sends a request to NSPK for payment by sending SubscriptionId, memberId, and qrcid;
4. NSPK sends a request to the issuing bank for payment by sending SubscriptionId and qrcid. The issuing bank makes a payment by debiting funds from the individual client's account for which the subscription is made. Sber receives the payment status from NSPK and sends it to the partner in gose/callback.

Specifics of the SBP payment process with subsequent creation of the subscription:

1. The Partner creates an order using the register.do method, indicating in jsonParams : "QRType" : "DYNAMIC_QR_SBP", sbp.scenario: C2B_SUBSCRIPTION_WITH_PAYMENT The subscription purpose "sbp.subscriptionPurpose" is required, and "description" is optional.
2. Upon successful payment, the partner will receive a callback / a response to gose order statuses confirming the payment.
3. After a successful payment, if the individual client chooses the action "bind account" in the app of their bank (or the check box "bind account" was selected at the payment confirmation stage), the issuing bank will create the subscription.
4. When the subscription is created, the partner will receive a callback / a response to gose with subscription parameters.
5. The interval between payment and subscription creation depends on the implementation of the subscription creation scenario in the issuing bank and the actions of the individual client, and may take up to several minutes

Getting order status

	Status request	getOrderStatusExtended.do (gose) transmitted parameter - orderId
Examples of responses (basic data provided)		
1	Successful payment via SBP through the bank's application (not a subscription)	<pre> { "errorCode": "0", "merchantOrderParams": [{ "name": "qrType", "value": "DYNAMIC_QR_SBP" }, { "name": "sbp.scenario", "value": "C2B" }], "paymentAmountInfo": { "approvedAmount": 1600402, "depositedAmount": 1600402, "refundedAmount": 0, "paymentState": "DEPOSITED" }, "transactionAttributes": [{ "name": "SbolBankInvoiceId", "value": "bcacc95e80064f2996b7951e3070538f" }], "attributes": [{ "name": "mdOrder", "value": "bcacc95e-8006-4f29-96b7-951e3070538f" }, { "name": "qrcId", "value": "AD10007KKKBGDEH9C8NIQ1QHPPFDNNH54" }], "operations": [{ "date": 1708508624608, "type": "AUTHORIZATION", "amount": 1600402, "referenceNumber": "405200490186", "approvalCode": "158594", "actionCode": 0 }], "orderNumber": "0863622194234", "orderStatus": 2, "actionCode": 0, "errorMessage": "Query processed without system errors", </pre>

		<pre> "amount": 1600402, "currency": "643", "date": 1708508562538, "depositedDate": 1708508624608, orderDescription: Order registration "authRefNum": "405200490186", "authDateTime": 1708508624608, "terminalId": "20183665" } </pre>
2	<p>Successful payment via the bank app with subsequent subscription creation</p>	<pre> { "errorCode": "0", "merchantOrderParams": [{ "name": "qrType", "value": "DYNAMIC_QR_SBP" }, { "name": "sbp.scenario", "value": "C2B_SUBSCRIPTION_WITH_PAYMENT" }, { "name": "sbp.subscriptionPurpose", "value": "information from merchant with details of account binding" }, { "name": "app2app", "value": "true" }], "paymentWay": "SBERPAY_SBP_C2B_DYN" }, "paymentAmountInfo": { "approvedAmount": 16001, "depositedAmount": 16001, "refundedAmount": 0, "paymentState": "DEPOSITED" }, "transactionAttributes": [{ "name": "SbolBankInvoiceId", "value": "bd1CBD4600bf4ab4a7e60c1f37eb9a75" }], "attributes": [{ "name": "mdOrder", "value": "bd1cubicle46-00bf-4ab4-a7e6-0c1f37eb9a75" }, { "name": "qrcId", "value": "BD1P005QT97I6AC49KLRE09MBAE2JS70" }, { "name": "subscriptionId", "value": "d6422960fac848a09d179d43fbc400f9" }, { "name": "memberId", "value": "100000000111" }], "operations": [{ "date": 1707121692503, "type": "AUTHORIZATION", "amount": 16001, "referenceNumber": "403600485418", "approvalCode": "026890", </pre>

		<pre> "actionCode": 0 }], "orderNumber": "7611775153674", "orderStatus": 2, "actionCode": 0, "errorMessage": "Query processed without system errors", "amount": 16001, "currency": "643", "date": 1707121652896, "depositedDate": 1707121692503, orderDescription: Order registration "authRefNum": "403600485418", "authDateTime": 1707121692503, "terminalId": "20183665" } </pre>
3	Successful payment by subscription	<pre> { "orderNumber": "20060583", "orderStatus": 2, "actionCode": 0, "errorCode": "0", "errorMessage": "Query processed without system errors", "amount": 1000 "currency": "643", "date": 1707121733416, "depositedDate": 1707121744397, "orderDescription": "Payment by binding", "authRefNum": "403600485420", "merchantOrderParams": [{ "name": "qrType", "value": "DYNAMIC_QR_SBP" },], "paymentWay": "SBERPAY_SBP_C2B_DYN_SUBS" "authDateTime": 1707121744397, "terminalId": "20288435", "paymentAmountInfo": { "approvedAmount": 1000, "depositedAmount": 1000, "refundedAmount": 0, "paymentState": "DEPOSITED" }, "bankInfo": {}, "transactionAttributes": [{ "name": "SbolBankInvoiceId", "value": "e2df130855ff4283b21fb8d04b9ccfcb" }], "attributes": [{ "name": "mdOrder", "value": "e2df1308-55ff-4283-b21f-b8d04b9ccfcb" }, { "name": "qrcId", "value": "BD10002NN2KCR7M39NABH4O95NIUODU3" }], "operations": [{ "date": 1707121744397, "type": "AUTHORIZATION", </pre>

		<pre> "amount": 1000 "referenceNumber": "403600485420", "approvalCode": "423828", "actionCode": 0 }] } </pre>
4	Subscription created successfully	<pre> { "errorCode": "0", "merchantOrderParams": [{ "name": "qrType", "value": "SUBSCRIPTION_SBP" }, { "name": "sbp.scenario", "value": "C2B_SUBSCRIPTION" }, { "name": "sbp.subscriptionPurpose", "value": "Well, let's check it out" }, { "name": "app2app", "value": "true" }], "bindingInfo": { "clientId": "client_1693389039" }, "payerData": { "email": "test@example.com" }, "transactionAttributes": [{ "name": "SbolBankInvoiceId", "value": "803535bed0fa4c1a9944f2a3541345cc" }], "attributes": [{ "name": "mdOrder", "value": "803535be-d0fa-4c1a-9944-f2a3541345cc" }, { "name": "qrcId", "value": "AB1S005HICG0UB1E90K883B4AR14G3SV" }, { "name": "subscriptionId", "value": "9e53906ee00b478abb6d1be3fc8f23c0" }, { "name": "memberId", "value": "100000000111" }, { "name": "phone", "value": "*****0792" }], "orderNumber": "0545747382784", "orderStatus": 2, "actionCode": 0, "errorMessage": "Query processed without system errors", "amount": 0, "currency": "643", "date": 1712059648595, "orderDescription": "Registering an order with a maximum set of fields", </pre>

5	Payment with creating a subscription, but the subscription has not been created	<pre> "terminalId": "20183665" } { "errorCode": "0", "merchantOrderParams": [{ "name": "qrType", "value": "DYNAMIC_QR_SBP" }, { "name": "sbp.scenario", "value": "C2B_SUBSCRIPTION_WITH_PAYMENT" }, { "name": "sbp.subscriptionPurpose", "value": "information from merchants with details of account binding" }, { "name": "app2app", "value": "true" }], "paymentWay": "SBERPAY_SBP_C2B_DYN" "paymentAmountInfo": { "approvedAmount": 214, "depositedAmount": 214, "refundedAmount": 0, "paymentState": "DEPOSITED" }, "transactionAttributes": [{ "name": "SbolBankInvoiceId", "value": "3c9d876236db4f578b02bc9f59563851" }], "attributes": [{ "name": "mdOrder", "value": "3c9d8762-36db-4f57-8b02-bc9f59563851" }, { "name": "qrcId", "value": "AD1P002EP15654KL8NRA4RBI8CHKD53N" }], "operations": [{ "date": 1709560873230, "type": "AUTHORIZATION", "amount": 214, "referenceNumber": "406400491910", "approvalCode": "850288", "actionCode": 0 }], "orderNumber": "8526268104654", "orderStatus": 2, "actionCode": 0, "errorMessage": "Query processed without system errors", "amount": 214, "currency": "643", "date": 1709560827260, "depositedDate": 1709560873230, orderDescription: Order registration "authRefNum": "406400491910", "authDateTime": 1709560873230, "terminalId": "20183665" </pre>
---	---	---

		} Difference from item 5 - no subscriptionId and memberId have been transferred
6	If the order is not paid for and has expired	{ "orderNumber": "0360288122824", "orderStatus": 6, "actionCode": -2007, "actionCodeDescription": "The waiting period for data entry has expired", "errorCode": "0", "errorMessage": "Query processed without system errors", "amount": 1566, "currency": "643", "date": 1710222871139, orderDescription: Order registration "merchantOrderParams": [{ "name": "qrType", "value": "DYNAMIC_QR_SBP" }, { "name": "sbp.scenario", "value": "C2B" }, "terminalId": "20183665", "transactionAttributes": [{ "name": "SbolBankInvoiceId", "value": "26e7724616c94521a83687b345eccf23" }], "attributes": [{ "name": "mdOrder", "value": "26e77246-16c9-4521-a838-70b345eccf23" }, { "name": "qrId", "value": "AD10007ED5G6V6MD8R48TQ0DJUUP19HH" }] }
7	the order was created to create a subscription, but the subscription has not been created (similar for an order that was created but not paid for and whose life has not yet expired)	{ "errorCode": "0", "merchantOrderParams": [{ "name": "qrType", "value": "SUBSCRIPTION_SBP" }, { "name": "sbp.scenario", "value": "C2B_SUBSCRIPTION" }, { "name": "sbp.subscriptionPurpose", "value": "Check" }, "transactionAttributes": [{ "name": "SbolBankInvoiceId", "value": "bf3e0fe788104aa6af4374294d9657e1" }], "attributes": [{ "name": "mdOrder", "value": "bf3e0fe7-8810-4aa6-af43-74294d9657e1" },], }

		<pre> { "name": "qrcId", "value": "AB1S00316BNKKJDF9E6ALG99QMBNTPP" }], "orderNumber": "7840349671674", "orderStatus": 0, "actionCode": -100, "actionCodeDescription": "No attempts to pay", "errorMessage": "Query processed without system errors", "amount": 0, "currency": "643", "date": 1710405230718, "orderDescription": "Registering the order with a maximum set of fields", "terminalId": "20183665" } </pre>
8	<p>Attempt to pay using a subscription that has been deleted from the issuer's system</p>	<pre> { "errorCode": "0", "merchantOrderParams": [{ "name": "QRType", "value": "DYNAMIC_QR_SBP" }, { "name": "sbp.scenario", "value": "C2B" },], "paymentWay": "SBERPAY_SBP_C2B_DYN_SUBS" "transactionAttributes": [{ "name": "SbolBankInvoiceId", "value": "2de620f35da34a84953653db127cbee3" }], "attributes": [{ "name": "mdOrder", "value": "2de620f3-5da3-4a84-9536-53db127cbee3" }, { "name": "qrcId", "value": "AD100060M4EHP3T94OOP1LO1C1F1EST" }], "operations": [{ "date": 1710506422007, "type": "AUTHORIZATION", "amount": 130, "approvalCode": "000000", "actionCode": -5031 }], "orderNumber": "3544684621264", "orderStatus": 0, "actionCode": -5031, "actionCodeDescription": "Subscription not found.", "errorMessage": "Query processing is completed without system errors", "amount": 130, "currency": "643", "date": 1710506408744, "orderDescription": "Order registration", "authDateTime": 1710506422007, "terminalId": "20288435" } </pre>

Comments on the lifetime of order.

The register.do request uses 2 parameters to determine the order lifetime

1. sessionTimeoutSecs – the order lifetime in seconds;
2. expirationDate – Date of order lifetime expiry (Example:"2023-12-31T23:59:59")

If the query has the expirationDate parameter, the value of the sessionTimeoutSecs parameter will be ignored.

Callback

Notification when creating a subscription	https://ecomtest.sberbank.ru/callbackUrl additionalParams with the SBP subscription parameters: phone Payer's masked phone number in the format *****XXXXX (only if Sber payer, otherwise NULL) subscriptionId: Stored subscription identifier memberId: Sending Bank ID where the subscription is stored extTransactionId SBP C2B transaction ID, if payment was performed qrcId Valid link identifier
---	---

Important for the testing process

1. If you need to view **a new subscription creation**, you should report about this, so that the subscription will be deleted on SBER's side (via SBOL). As according to the NSPK regulations, one Client in a specific issuing bank can create only one subscription for a specific legal entity.
2. If the subscription has already been created, the "Subscription Creation" and "Payment with Subscription" scenarios will result in an account change, without changing the subscriptionId.
3. Deletion/change of the subscription account is possible only in the SBOL interface on the Client's side, and no tools for this, except for SBOL, are provided.