**Payment of the order with the input of the card parameters on the merchant's side - AF**

Alfa Bank

**Table of contents**

[1. Please have a look before reading the documentation 5](#_Toc127951168)

[2. Glossary 6](#_Toc127951169)

[3. Algorithm of connection to the payment gateway 7](#_Toc127951170)

[4. General description of the methods of interaction with the payment gateway 8](#_Toc127951171)

[4.1. Possibility of card payment by installments 8](#_Toc127951172)

[4.2. Payment mechanism (one-phase and two-phase) 8](#_Toc127951173)

[4.3. Patterns of interaction with the payment gateway 8](#_Toc127951174)

[4.4. Interfaces for interaction with the payment gateway 8](#_Toc127951175)

[4.5. Connection coordinates 9](#_Toc127951176)

[5. Description of interaction patterns 11](#_Toc127951177)

[5.1. Payment via internal MPI 11](#_Toc127951178)

[5.1.1. One-phase payment with return to the gateway after ACS authentication 11](#_Toc127951179)

[5.1.2. One-phase payment with return to the gateway after ACS authentication (simplified redirection of the client to ACS) 13](#_Toc127951180)

[5.1.3. Two-phase payment with return to the gateway after ACS authentication 16](#_Toc127951181)

[5.1.4. Two-phase payment with return to the gateway after ACS authentication (simplified redirection of the client to ACS) 19](#_Toc127951182)

[5.1.5. One-phase payment with return to the store after ACS authentication 22](#_Toc127951183)

[5.1.6. Two-phase payment with return to the store after ACS authentication 25](#_Toc127951184)

[5.2. Payment via external MPI 29](#_Toc127951185)

[5.2.1. One-phase payment 29](#_Toc127951186)

[5.2.2. Two-phase payment 31](#_Toc127951187)

[6. Request Specifications 35](#_Toc127951188)

[6.1. Web-Service interface 35](#_Toc127951189)

[6.1.1. Request for order registration (WS) 35](#_Toc127951190)

[6.1.2. Request for order registration with pre-authorization (WS) 39](#_Toc127951191)

[6.1.3. Request to verify the card's enrolment in 3DS (WS) 43](#_Toc127951192)

[6.1.4. Order Payment Requests (WS) 44](#_Toc127951193)

[6.1.5. Finish3DS request (WS) 50](#_Toc127951194)

[6.1.6. Order Status Request (WS) 51](#_Toc127951195)

[6.1.7. Advanced Order Status Request (WS) 52](#_Toc127951196)

[6.1.8. Request for Order Payment Completion (WS) 57](#_Toc127951197)

[6.1.9. Request for order payment cancellation (WS) 58](#_Toc127951198)

[6.1.10. Request for a refund of the order payment (WS) 59](#_Toc127951199)

[6.1.11. Card Verification Request (WS) 60](#_Toc127951200)

[6.2. REST Interface 63](#_Toc127951201)

[6.2.1. Request for order registration (REST) 63](#_Toc127951202)

[6.2.2. Request for order registration with pre-authorization (REST) 67](#_Toc127951203)

[6.2.3. Request to verify the card's enrolment in 3DS (REST) 71](#_Toc127951204)

[6.2.4. Order Payment Requests (REST) 72](#_Toc127951205)

[6.2.5. Finish3DS Request (REST) 76](#_Toc127951206)

[6.2.6. Order Status Request (REST) 78](#_Toc127951207)

[6.2.7. Advanced Order Status Request (REST) 79](#_Toc127951208)

[6.2.8. Request for Order Payment Completion (REST) 83](#_Toc127951209)

[6.2.9. Request for order payment cancellation (REST) 84](#_Toc127951210)

[6.2.10. Request for a refund of the order payment (REST) 85](#_Toc127951211)

[6.2.11. Card Verification Request (REST) 86](#_Toc127951212)

[6.3. Proxy interface for payment by installments (REST) 89](#_Toc127951213)

[6.3.1. Request to check the possibility of payment by installments (REST) 89](#_Toc127951214)

[6.3.2. Request for information about installment payment options (REST) 90](#_Toc127951215)

[6.3.3. Request for information about the installment plan status (REST) 94](#_Toc127951216)

[7. Test cards 96](#_Toc127951217)

[8. Appendix 1. Response codes - decryption of actionCode (processing response) 101](#_Toc127951218)

* Please have a look before reading the documentation
* Glossary
* Algorithm of connection to the payment gateway
* General description of the methods of interaction with the payment gateway

**о** Possibility of card payment by installments

**о** Payment mechanism (one-phase and two-phase)

**о** Patterns of interaction with the payment gateway

**о** Interfaces for interaction with the payment gateway

**о** Connection coordinates

* Description of interaction patterns

**о** Payment via internal MPI

* One-phase payment with return to the gateway after ACS authentication
* Cancellation of order payment
* Refund of order payment funds
* Verification of the card's enrolment in 3DS
* One-phase payment with return to the gateway after ACS authentication (simplified redirection of the client to ACS)
* Cancellation of order payment
* Refund of order payment funds
* Verification of the card's enrolment in 3DS
* Two-phase payment with return to the gateway after ACS authentication
* Cancellation of order payment
* Refund of order payment funds
* Verification of the card's enrolment in 3DS
* Two-phase payment with return to the gateway after ACS authentication (simplified redirection of the client to ACS)
* Cancellation of order payment
* Refund of order payment funds
* Verification of the card's enrolment in 3DS
* One-phase payment with return to the store after ACS authentication
* Cancellation of order payment
* Refund of order payment funds
* Verification of the card's enrolment in 3DS
* Two-phase payment with return to the store after ACS authentication
* Cancellation of order payment
* Refund of order payment funds
* Verification of the card's enrolment in 3DS

**о** Payment via external MPI

* One-phase payment
* Cancellation of order payment
* Refund of order payment funds
* Verification of the card's enrolment in 3DS
* Two-phase payment
* Cancellation of order payment
* Refund of order payment funds
* Verification of the card's enrolment in 3DS

• Request Specifications

**о** Web-Service Interface

* Request for order registration (WS)
* Request for order registration with pre-authorization (WS)
* Request to verify the card's enrolment in 3DS (WS)
* Order Payment Requests (WS)
* Request for order payment when using an external MPI (WS)
* Request for order payment when using an internal MPI (WS)
* Finish3DS request (WS)
* Order Status Request (WS)
* Advanced Order Status Request (WS)
* Request for Order Payment Completion (WS)
* Request for order payment cancellation (WS)
* Request for a refund of the order payment (WS)
* Card Verification Request (WS)

**о** REST Interface

* Request for order registration (REST)
* Request for order registration with pre-authorization (REST)
* Request to verify the card's enrolment in 3DS (REST)
* Order Payment Requests (REST)
* Request for order payment when using an external MPI (REST)
* Request for order payment when using an internal MPI (REST)
* Finish3DS Request (REST)
* Order Status Request (REST)
* Advanced Order Status Request (REST)
* Request for Order Payment Completion (REST)
* Request for order payment cancellation (REST)
* Request for a refund of the order payment (REST)
* Card Verification Request (REST)

**о** Proxy interface for payment by installments (REST)

* Request to check the possibility of payment by installments (REST)
* Request for information about installment payment options (REST)
* Request for information about the installment plan status (REST)
* Test cards
* Appendix 1. Response codes - decryption of actionCode (processing response)

1. Please have a look before reading the documentation

|  |
| --- |
| Any functions or functionality of the system not described in this document may be used only at your own risk. |

The document describes the payment of the order on the side of the store.

1. Glossary

* *3-D Secure* is a Visa IPS technology that allows additional user authentication by means of the issuing bank.
* *ACS* – Access Control Server, a 3-D Secure infrastructure element that provides payer validation on the issuing bank's side.
* *Merchant Plugin Interface (MPI)* is a 3-D Secure and SecureCode technology component that can be placed on the PS side or on the store side
* *SecureCode* is a MasterCard IPS technology that allows additional user authentication by means of the issuing bank. Technologically, it is equivalent to 3-D Secure. In the text below, mentioning 3-D Secure also implies SecureCode.
* *Bank Card* is a card of the international payment systems VISA and MasterCard.
* *Acquiring bank* is a bank that implements and operates a payment gateway.
* *Issuing bank* is a bank that issued the client's bank card.
* *Refund* is a partial or full repayment of funds to the buyer's card in case of their refusal to receive the goods (services) or its return. The refund operation shall be performed after the funds are debited from the buyer's account.
* *Two-phase payment* is an operation to pay for goods/services made via the Internet using bank cards, requiring additional confirmation. The two-phase mechanism of operation enables to separate the process of checking the solvency of a bank card (authorization) and withdrawing money (financial confirmation). At the first phase of a two-phase payment, the solvency of the bank card is checked and funds are blocked on the client's account.
* *Order* is an elementary essence of the system, describes an order in an online store or its equivalent. Any order has an amount.
* *Store (Merchant)* is a retailer and service outlet (RSO) that sells goods or provides services through a website.
* *IPS* is an international payment system (for example, Visa or MasterCard)
* *One-phase payment* is an operation to pay for goods/services made via the Internet using bank cards, which does not require additional confirmation.
* *Cancellation of the payment operation (Reversal)* is a removal of the block from the funds on the buyer's card. This function is available for a limited time; the exact terms must be specified in the bank.
* *Payment form* is an HTML page that is used by the client to enter payment details.
* *Payment details* are the details used by the user to pay for the order. Usually, this is the card number, expiration date, CVC.
* *Acquiring Bank's Payment Gateway (PG)* is an automatic system that allows a *store* to accept and the Client to send payments via the Internet using bank cards.
* *Payer* is an individual making a payment using their card for the merchant's services in the merchant's online store.
* *Binding* is a correspondence between the Payer and the payment details of the card (card number, card expiration date).

1. Algorithm of connection to the payment gateway

In order to connect to the payment gateway, the store must:

1. Get logins and passwords for access to the test server from Bank employees:
2. Login with the "-api" suffix – to connect via the software interface (API);
3. Login with the "-operator" suffix – to work in the Personal Area via the web interface.
   1. Publish one's own payment page.
   2. Perform integration according to this document.
   3. Check the solution operation using test cards (listed in the [Test Cards](#_bookmark77) section) via:
4. REST interface \ Web-Service interface;
5. personal area in the administrative console. The following is necessary for the check:
6. to carry out several successfully and unsuccessfully paid orders;
7. to check the display of the correct status on the page with the payment result and compare it with the status in the PA (personal area);
8. to complete the payment (in case of using the two-phase work pattern), cancel and refund the order.
   1. After testing, it is necessary to contact the bank and inform about the readiness to work in the production environment. It is advisable to provide the address of the test resource through which the Bank's employees could make verification payments.
   2. After successful verification of the integration and the payment page (if it was created by the store), subject to the concluded contract, the store shall be provided with the details for connecting to the production.
   3. After receiving logins and passwords to the production server, it is necessary to make verification payments on this card: to pay for the order, then cancel it and/or refund it.
9. General description of the methods of interaction with the payment gateway

Payment by bank card for goods and services via the Internet can be carried out with the transfer of full card details to the payment gateway, using bindings as well as using external payment systems.

* 1. Possibility of card payment by installments

It is possible to pay by installments through the MasterCard and VISA installment systems. To be able to accept payment by installments, contact the bank's representatives. This document also provides information on how to accept payment by installments using the payment gateway API.

* 1. Payment mechanism (one-phase and two-phase)

A store can use a one-phase or two-phase payment acceptance mechanism:

* *One-phase payment* is an operation to pay for goods/services made via the Internet using bank cards that does not require additional confirmation (one request initiates blocking and debiting funds from the card immediately).
* *Two-phase payment* is an operation to pay for goods/services made via the Internet using bank cards, requiring additional confirmation. The two-phase mechanism of operation enables to divide the process into checking the solvency of a bank card (authorization) and debiting funds (financial confirmation). The first request initiates checking the solvency of the bank card and blocking funds on the client's account, the second request initiates debiting.
  1. Patterns of interaction with the payment gateway
  2. Interfaces for interaction with the payment gateway

The store can use one of the interfaces to interact with the payment gateway – the interface on the WebServices or REST.

To authorize the store's access to the payment gateway system, any request from the store must contain the store's name and password received when registering the store in the system. Detailed information about the authorization of requests is provided below:

##### Implementation of interaction through the Web-Service interface:

The description (WSDL) of the service is located on the test server, which is available without restrictions.

The name and password values are passed in the format described in the WS-Security specification, userName token authorization type. The header with such authorization will look something like this:

<wsse:Security xmlns:wsse= http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss- wssecurity-secext-1.0.xsd xmlns:wsu= http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-

%20wssecurity-utility-1.0.xsd >

<wsse:UsernameToken wsu:Id= UsernameToken-87 >

<wsse:Username>aa</wsse:Username>

<wsse:Password Type= http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss- username-

token-profile-1.0#PasswordText >123456</wsse:Password>

</wsse:UsernameToken>

</wsse:Security>

If errorCode = 0 in the response to the request, it means that the request was processed by the payment gateway without system errors. At the same time, errorCode does not display the order status.

To get the order status, use the getOrderStatus or getOrderStatusExtended request (one of these methods must be implemented when integrating the RSO with the payment gateway).

##### Implementation of interaction through the REST interface

|  |
| --- |
| Backward compatibility of the API is not violated if:   * a new optional parameter appears in the request; * a new parameter appears in the response; * the order of the parameters in the response changes. |

Interactions are implemented as HTTP requests by GET or POST methods to specific URLs. Parameters are passed as parameters of GET or POST requests, the values of which must be compatible with the URL (i.e. url encoded).

The result of processing the request is returned as a JSON object.   
For example: {"ErrorCode":"12","ErrorMessage":"Empty amount"}

The name and password values are passed in the following parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Compulsoriness** | **Description** |
| userName | AN..30 | yes | The store's login received when connecting |
| password | AN..30 | yes | The store's password received when connecting |

The request content type must be application/x-www-form-urlencoded. All text fields must be Unicode (UTF-8) encoded.

Special characters in the REST request must be escaped using URL encoding. For example, the password qwe?rt%y should be passed as qwe%0Frt%25y. For more information about escaping in requests and the symbol table, see <https://wm-school.ru/html/html_url_acsii.html>

If errorCode = 0 in the response to the request, it means that the request was processed by the payment gateway without system errors. At the same time, errorCode does not display the order status.

To get the order status, use the getOrderStatus.do or getOrderStatusExtende.do request (one of these methods must be implemented when integrating the RSO with the payment gateway).

The connection coordinates for the WebService and REST interfaces are presented in the next section.

* 1. Connection coordinates

When registering the seller, the representative is provided with a login and password that can be used in the personal area, and also needs to be used in the protocols.

The description of the test service (WSDL) is located at [https://tws.egopay.ru/ab/webservices/merchant-ws?wsdl .](https://web.rbsuat.com/ab/webservices/merchant-ws?wsdl)

URLs for accessing REST methods are shown in the table below.

|  |  |
| --- | --- |
| **Method name** | **URL** |
| Order registration | https://tws.egopay.ru/ab/rest/register.do |
| Order registration with pre-authorization | https://tws.egopay.ru/ab/rest/registerPreAuth.do |
| Request for order payment completion | https://tws.egopay.ru/ab/rest/deposit.do |
| Order payment request | https://tws.egopay.ru/ab/rest/paymentorder.do |
| Finish3ds method | https://tws.egopay.ru/ab/rest/finish3dsPayment.do |
| Request for order payment cancellation | https://tws.egopay.ru/ab/rest/reverse.do |
| Request for a refund of the order payment | https://tws.egopay.ru/ab/rest/refund.do |
| Getting the order status | https://tws.egopay.ru/ab/rest/getOrderStatus.do |
| Getting an extended order status | https://tws.egopay.ru/ab/rest/getOrderStatusExtended.do |
| Request to verify the card's enrolment in 3DS | https://tws.egopay.ru/ab/rest/verifyEnrollment.do |

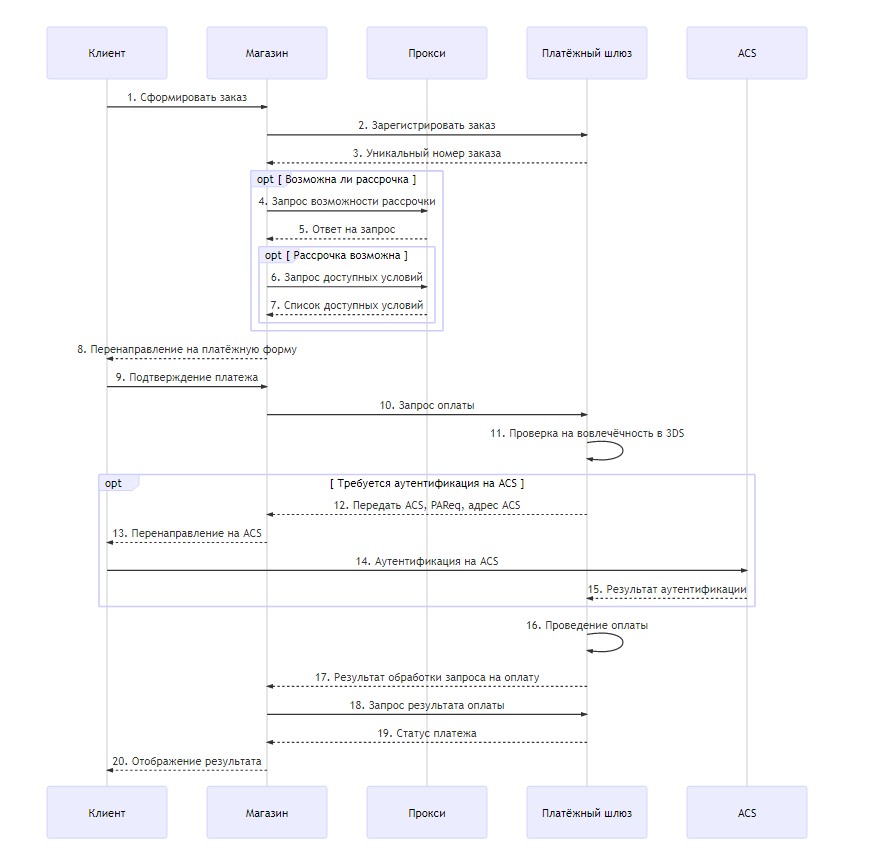
Also, through the REST interface, by agreement with the bank, methods related to payment by installments are available (see the table below).

|  |
| --- |
| These requests are not available via the WSDL interface. |

|  |  |
| --- | --- |
| **Method name** | **URL** |
| Request to verify the possibility of payment by installments | [https://tws.egopay.ru/ab/installment-](https://web.rbsuat.com/ab/installment-proxy/binranges/check)   [proxy/binranges/check](https://web.rbsuat.com/ab/installment-proxy/binranges/check) |
| Request for information about installment payment options | [https://tws.egopay.ru/ab/installment-](https://web.rbsuat.com/ab/installment-proxy/installments/get)   [proxy/installments/get](https://web.rbsuat.com/ab/installment-proxy/installments/get) |
| Request for information about the status of the installment plan | [https://tws.egopay.ru/ab/installment-](https://web.rbsuat.com/ab/installment-proxy/installments/getInstallmentStatus)   [proxy/installments/getInstallmentStatus](https://web.rbsuat.com/ab/installment-proxy/installments/getInstallmentStatus) |

1. Description of interaction patterns
   1. Payment via internal MPI
      1. One-phase payment with return to the gateway after ACS authentication

##### Order payment scenario



|  |  |
| --- | --- |
| **Клиент** | **Client** |
| **Магазин** | **Store** |
| **Прокси** | **Proxy** |
| **Платёжный шлюз** | **Payment Gateway** |
| **ACS** | **ACS** |
| 1. Сформировать заказ | 1. Generate an order |
| 2. Зарегистрировать заказ | 2. Register an order |
| 3. Уникальный номер заказа | 3. Unique order number |
| [ Возможна ли рассрочка ] | [Whether installment is possible] |
| 4. Запрос возможности рассрочки | 4. Request for the possibility of installments |
| 5. Ответ на запрос | 5. Response to the request |
| [ Рассрочка возможна ] | [Installments are possible] |
| 6. Запрос доступных условий | 6. Request for available conditions |
| 7. Список доступных условий | 7. List of available conditions |
| 8. Перенаправление на платёжную форму | 8. Redirection to the payment form |
| 9. Подтверждение платежа | 9. Payment confirmation |
| 10. Запрос оплаты | 10. Request for payment |
| 11. Проверка на вовлечёчность в 3DS | 11. Verification of 3DS enrolment |
| [ Требуется аутентификация на ACS ] | [ ACS authentication required ] |
| 12. Передать ACS, PAReq. адрес ACS | 12. Transfer ACS, PAReq, ACS address |
| 13. Перенаправление на ACS | 13. Redirection to ACS |
| 14. Аутентификация на ACS | 14. ACS authentication |
| 15. Результат аутентификации | 15. Authentication result |
| 16. Проведение оплаты | 16. Making the payment |
| 17. Результат обработки запроса на оплату | 17. Result of processing the payment request |
| 18. Запрос результата оплаты | 18. Request for payment result |
| 19. Статус платежа | 19. Payment status |
| 20. Отображение результата | 20. Displaying the result |

##### Description

|  |  |
| --- | --- |
| 1 | 1. The client generates an order on the RSO (online store) website and confirms the decision to pay for the goods with a bank card. |
| 2 | 1. The store sends a request to register the order in the payment gateway. For registration, parameters such as the debit amount, the debit currency, the order number in the store's system, as well as the client's return URL are used. The request specification is presented in the following sections:    * [Request for order registration (SOAP);](#_bookmark46)    * [Request for order registration (REST).](#_bookmark60) |
| 3 | 1. In case of successful order registration, the response returns a unique order number in the payment gateway (in the orderID parameter). |
| 4 | In order to check whether the client's card number is included in the installment program, the merchant sends a [check request](#_bookmark74) to the payment gateway proxy, specifying the client's card BIN. |
|  | This opportunity is provided by agreement with the bank. |
| 5 | * If the request returns true, the merchant can make the next request (get) to find out what installment options are provided to the client. * If the request returns false, the client has no option to pay by installments, and the next request is skipped. |
| 6 | The merchant makes a get [request in the proxy of the payment gateway](#_bookmark75) to get information about the installment options for the client. |
| 7 | The payment gateway proxy returns a response with installment options. Subsequently, to make payment by installments, one will need to pass the received parameters as additional in the payment request (paymentOrder). |
| 8 | 1. The online store transfers the cardholder's browser to its payment page to enter card data. |
| 9 | 1. The cardholder enters the card details on the payment page and transmits the data to the online store. |
| 10 | 1. After receiving the payment details, the store sends a request to the payment gateway to pay for the order. The request transmits parameters such as the order number in the payment gateway (received in step 9) and card details. The request specification is presented in the following sections:    * [Order payment request (SOAP);](#_bookmark51)    * [Order payment request (REST).](#_bookmark65) |
| 11 | 1. After receiving the payment details, the system verifies the enrolment of the card in 3-D Secure, including using Directory Server.   In the event that the use of 3-D Secure technology is required for making a payment, the next step of the scenario is performed.  If the payment has to go through SSL, then the scenario execution continues from **Step 16**. |
| 12 | 1. The verification result is sent to the store (card enrolment + URL of the Issuing Bank's ACS server where the client needs to be authenticated). |
| 13 | 1. The store's system sends a redirection to the issuer's ACS to the client's browser. |
| 14 | 1. The Client authenticates through ACS of the Issuing bank. |
| 15 | 1. The Issuing bank's ACS redirects the client to the payment gateway by transmitting PARes. If the client fails to authenticate through ACS, the payment gateway will reject the payment. If authentication is successful, **Step 12** of the scenario is executed. |
| 16 | 1. The payment gateway authorizes the payment. |
| 17 | 1. The result of processing the order payment request is returned to the store. |
| 18 | 1. The store's system requests the order payment status from the payment gateway (by the unique order ID in the payment gateway, which was received when the order was registered in the orderID parameter). The specification of the order status request is presented in the sections:  * [Advanced Order Status Request (SOAP);](#_bookmark54) * [Advanced Order Status Request (REST).](#_bookmark68) |
| 19 | 1. The payment gateway returns the payment status to the store. |
| 20 | 1. The store displays a page to the client with the result of the order payment. |

* + - 1. Cancellation of order payment

Cancellation of order payment is available to stores if they have the appropriate rights (in agreement with the bank). For one-phase payments, payment cancellation is possible for orders in the **Completed**/**Deposited** state.

Cancellation of order payment is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, through REST/SOAP interfaces. The request specification is presented in the sections below.
  + [Request for order payment cancellation (SOAP).](#_bookmark56)
  + [Request for order payment cancellation (REST)](#_bookmark70).

If the cancellation operation is successful, the order will be transferred from the **Completed**/**Deposited** state to the **Canceled**/**Reversed** state.

* + - 1. Refund of order payment funds

A full or partial refund for paid orders (in the **Completed**/**Deposited** state) is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, via SOAP/REST interfaces. The request specification is presented in the sections below.
  + [Request for a refund of the order payment (SOAP).](#_bookmark57)
  + [Request for a refund of the order payment (REST).](#_bookmark71)

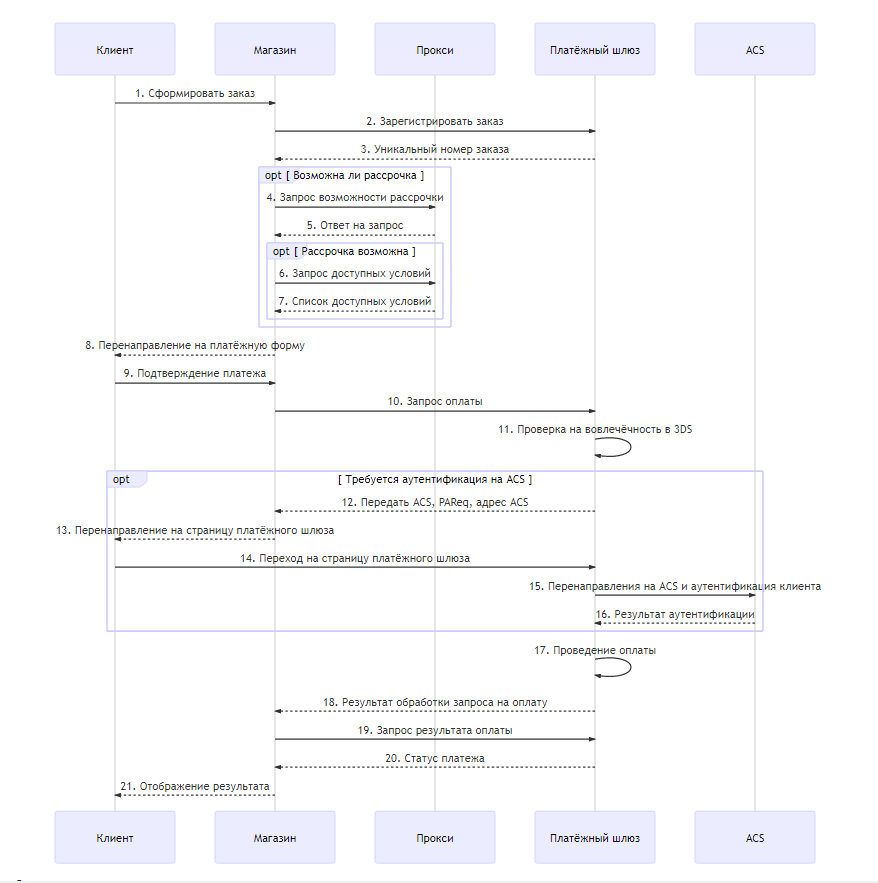
After IPS receives a refund request sent by one of the above methods, IPS returns the specified amount to the client's account.

* + - 1. Verification of the card's enrolment in 3DS

The system allows the store to independently verify the enrolment of the bank card in 3-D Secure, if necessary. This can be done using the API, via SOAP/REST interfaces. The request specification is presented in the following sections.

* [Request to verify the card's enrolment in 3DS (SOAP).](#_bookmark48)
* [Request to verify the card's enrolment in 3DS (REST)](#_bookmark62).
  + 1. One-phase payment with return to the gateway after ACS authentication (simplified redirection of the client to ACS)

##### Order payment scenario



|  |  |
| --- | --- |
| **Клиент** | **Client** |
| **Магазин** | **Store** |
| **Прокси** | **Proxy** |
| **Платёжный шлюз** | **Payment Gateway** |
| **ACS** | **ACS** |
| 1. Сформировать заказ | 1. Generate an order |
| 2. Зарегистрировать заказ | 2. Register an order |
| 3. Уникальный номер заказа | 3. Unique order number |
| [ Возможна ли рассрочка ] | [Whether installment is possible] |
| 4. Запрос возможности рассрочки | 4. Request for the possibility of installments |
| 5. Ответ на запрос | 5. Response to the request |
| [ Рассрочка возможна ] | [Installments are possible] |
| 6. Запрос доступных условий | 6. Request for available conditions |
| 7. Список доступных условий | 7. List of available conditions |
| 8. Перенаправление на платёжную форму | 8. Redirection to the payment form |
| 9. Подтверждение платежа | 9. Payment confirmation |
| 10. Запрос оплаты | 10. Request for payment |
| 11. Проверка на вовлечёчность в 3DS | 11. Verification of 3DS enrolment |
| [ Требуется аутентификация на ACS ] | [ ACS authentication required ] |
| 12. Передать ACS, PAReq. адрес ACS | 12. Transfer ACS, PAReq, ACS address |
| 13. Перенаправление на страницу платежного шлюза | 13. Redirection to the payment gateway page |
| 14. Переход на страницу платежного шлюза | 14. Shift to the payment gateway page |
| 15. Перенаправления на ACS и аутентификация клиента | 15. Redirections to ACS and client authentication |
| 16. Результат аутентификации | 16. Authentication result |
| 17. Проведение оплаты | 17. Making the payment |
| 18. Результат обработки запроса на оплату | 18. Result of processing the payment request |
| 19. Запрос результата оплаты | 19. Request for payment result |
| 20. Статус платежа | 20. Payment status |
| 21. Отображение результата | 21. Displaying the result |

##### Description

|  |  |
| --- | --- |
| 1 | 1. The client generates an order on the RSO (online store) website and confirms the decision to pay for the goods with a bank card. |
| 2 | 1. The store sends a request to register the order in the payment gateway. For registration, parameters such as the debit amount, the debit currency, the order number in the store's system, as well as the client's return URL are used. The request specification is presented in the following sections:    * [Request for order registration (SOAP);](#_bookmark46)    * [Request for order registration (REST).](#_bookmark60) |
| 3 | 1. In case of successful order registration, the response returns a unique order number in the payment gateway (in the orderID parameter). |
| 4 | In order to check whether the client's card number is included in the installment program, the merchant sends a [check request](#_bookmark74) to the payment gateway proxy, specifying the client's card BIN. |
|  | This opportunity is provided by agreement with the bank. |
| 5 | * If the request returns true, the merchant can make the next request (get) to find out what installment options are provided to the client. * If the request returns false, the client has no option to pay by installments, and the next request is skipped. |
| 6 | The merchant makes a get [request in the proxy of the payment gateway](#_bookmark75) to get information about the installment options for the client. |
| 7 | The payment gateway proxy returns a response with installment options. Subsequently, to make payment by installments, one will need to pass the received parameters as additional in the payment request (paymentOrder). |
| 8 | 1. The online store transfers the cardholder's browser to its payment page to enter card data. |
| 9 | 1. The cardholder enters the card details on the payment page and transmits the data to the online store. |
| 10 | 1. After receiving the payment details, the store sends a request to the payment gateway to pay for the order. The request transmits parameters such as the order number in the payment gateway (received in step 9) and card details. The request specification is presented in the following sections:    * [Order payment request (SOAP);](#_bookmark51)    * [Order payment request (REST).](#_bookmark65) |
| 11 | 1. After receiving the payment details, the system verifies the enrolment of the card in 3-D Secure, including using Directory Server.  In the event that the use of 3-D Secure technology is required for making a payment, the next step of the scenario is performed.  If the payment has to go through SSL, then the scenario execution continues from **Step 16**. |
| 12 | 1. The verification result is sent to the store (card enrolment + URL of the Issuing Bank's ACS server where the client needs to be authenticated). |
| 13 | The store's system sends a redirection to the payment gateway page to the client's browser using the following link: https://tws.egopay.ru/ab/acsRedirect.do?orderId =<order number> where <order number> is the unique order number of the client. |
|  | The link contains the address of the test server of the payment gateway. |
| 14 | 1. The client goes to the payment gateway page. |
| 15 | 1. Without additional actions on the client's part, the payment gateway redirects the client to the Issuing bank's ACS, where the client is authenticated. |
| 16 | 1. The Issuing bank's ACS redirects the client to the payment gateway by transmitting PARes. If the client fails to authenticate through ACS, the payment gateway will reject the payment. If authentication is successful, the following scenario is executed. |
| 17 | 1. The payment gateway authorizes the payment. |
| 18 | 1. The result of processing the order payment request is returned to the store. |
| 19 | 1. The store's system requests the order payment status from the payment gateway (by the unique order ID in the payment gateway, which was received when the order was registered in the orderID parameter). The specification of the order status request is presented in the sections: |
| * [Advanced Order Status Request (SOAP);](#_bookmark54) * [Advanced Order Status Request (REST).](#_bookmark68) |
| 20 | 1. The payment gateway returns the payment status to the store. |
| 21 | 1. The store displays a page to the client with the result of the order payment. |

* + - 1. Cancellation of order payment

Cancellation of order payment is available to stores if they have the appropriate rights (in agreement with the bank). For one-phase payments, payment cancellation is possible for orders in the **Completed**/**Deposited** state.

Cancellation of order payment is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, through REST/SOAP interfaces. The request specification is presented in the sections below.
  + [Request for order payment cancellation (SOAP).](#_bookmark56)
  + [Request for order payment cancellation (REST)](#_bookmark70).

If the cancellation operation is successful, the order will be transferred from the **Completed**/**Deposited** state **to the Canceled**/**Reversed** state.

* + - 1. Refund of order payment funds

A full or partial refund for paid orders (in the **Completed**/**Deposited** state) is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, via SOAP/REST interfaces. The request specification is presented in the sections below.
  + [Request for a refund of the order payment (SOAP).](#_bookmark57)
  + [Request for a refund of the order payment (REST).](#_bookmark71)

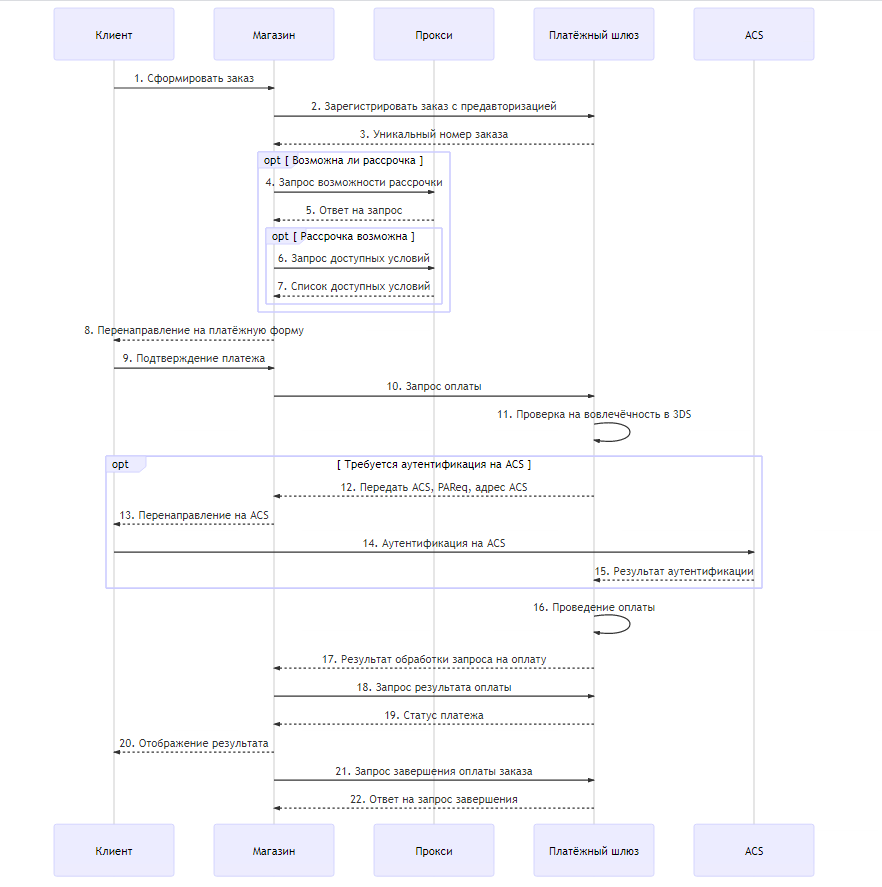
After IPS receives a refund request sent by one of the above methods, IPS returns the specified amount to the client's account.

* + - 1. Verification of the card's enrolment in 3DS

The system allows the store to independently verify the enrolment of the bank card in 3-D Secure, if necessary. This can be done using the API, via SOAP/REST interfaces. The request specification is presented in the following sections.

* [Request to verify the card's enrolment in 3DS (SOAP).](#_bookmark48)
* [Request to verify the card's enrolment in 3DS (REST)](#_bookmark62).
  + 1. Two-phase payment with return to the gateway after ACS authentication

##### Order payment scenario



|  |  |
| --- | --- |
| **Клиент** | **Client** |
| **Магазин** | **Store** |
| **Прокси** | **Proxy** |
| **Платёжный шлюз** | **Payment Gateway** |
| **ACS** | **ACS** |
| 1. Сформировать заказ | 1. Generate an order |
| 2. Зарегистрировать заказ с предавторизацией | 2. Register the order with pre-authorization |
| 3. Уникальный номер заказа | 3. Unique order number |
| [ Возможна ли рассрочка ] | [Whether installment is possible] |
| 4. Запрос возможности рассрочки | 4. Request for the possibility of installments |
| 5. Ответ на запрос | 5. Response to the request |
| [ Рассрочка возможна ] | [Installments are possible] |
| 6. Запрос доступных условий | 6. Request for available conditions |
| 7. Список доступных условий | 7. List of available conditions |
| 8. Перенаправление на платёжную форму | 8. Redirection to the payment form |
| 9. Подтверждение платежа | 9. Payment confirmation |
| 10. Запрос оплаты | 10. Request for payment |
| 11. Проверка на вовлечёчность в 3DS | 11. Verification of 3DS enrolment |
| [ Требуется аутентификация на ACS ] | [ ACS authentication required ] |
| 12. Передать ACS, PAReq. адрес ACS | 12. Transfer ACS, PAReq, ACS address |
| 13. Перенаправление на ACS | 13. Redirection to ACS |
| 14. Аутентификация на ACS | 14. ACS authentication |
| 15. Результат аутентификации | 15. Authentication result |
| 16. Проведение оплаты | 16. Making the payment |
| 17. Результат обработки запроса на оплату | 17. Result of processing the payment request |
| 18. Запрос результата оплаты | 18. Request for payment result |
| 19. Статус платежа | 19. Payment status |
| 20. Отображение результата | 20. Displaying the result |
| 21. Запрос завершения оплаты заказа | 21. Request for order payment completion |
| 22. Ответ на запрос завершения | 22. Response to completion request |

##### Description

|  |  |
| --- | --- |
| 1 | The client generates an order on the RSO (online store) website and confirms the decision to pay for the goods with a bank card. |
| 2 | The store sends a request to register the order in the payment gateway. For registration, parameters such as the debit amount, the debit currency, the order number in the store's system, as well as the client's return URL are used. The request specification is presented in the following sections:   * [Request for order registration with pre-authorization (SOAP);](#_bookmark47) * [Request for order registration with pre-authorization (REST).](#_bookmark61) |
| 3 | In case of successful order registration, the response returns a unique order number in the payment gateway (in the orderID parameter). |
| 4 | In order to check whether the client's card number is included in the installment program, the merchant sends a [check request](#_bookmark74) to the payment gateway proxy, specifying the client's card BIN. |
|  | This opportunity is provided by agreement with the bank. |
| 5 | * If the request returns true, the merchant can make the next request (get) to find out what installment options are provided to the client. * If the request returns false, the client has no option to pay by installments, and the next request is skipped. |
| 6 | The merchant makes a get [request in the proxy of the payment gateway](#_bookmark75) to get information about the installment options for the client. |
| 7 | The payment gateway proxy returns a response with installment options. Subsequently, to make payment by installments, one will need to pass the received parameters as additional in the payment request (paymentOrder). |
| 8 | The online store transfers the cardholder's browser to its payment page to enter card data. |
| 9 | The cardholder enters the card details on the payment page and transmits the data to the online store. |
| 10 | After receiving the payment details, the store sends a request to the payment gateway to pay for the order. The request transmits parameters such as the order number in the payment gateway (received in step 9) and card details. The request specification is presented in the following sections:   * [Order payment request (SOAP);](#_bookmark51) * [Order payment request (REST).](#_bookmark65) |
| 11 | After receiving the payment details, the system verifies the enrolment of the card in 3-D Secure, including using Directory Server.  If the payment requires the use of 3-D Secure technology, the next step of the scenario is performed.  If the payment has to go through SSL, then the scenario execution continues from **Step 16**. |
| 12 | The verification result is sent to the store (card enrolment + URL of the Issuing Bank's ACS server where the client needs to be authenticated). |
| 13 | The store's system sends a redirection to the issuer's ACS to the client's browser. |
| 14 | The Client authenticates through ACS of the Issuing bank. |
| 15 | The Issuing bank's ACS redirects the client to the payment gateway by transmitting PARes. If the client fails to authenticate through ACS, the payment gateway will reject the payment. If authentication is successful, **Step 16** of the scenario is executed. |
| 16 | The payment gateway authorizes the payment. |
| 17 | The result of processing the order payment request is returned to the store. |
| 18 | The store's system requests the order payment status from the payment gateway (by the unique order ID in the payment gateway, which was received when the order was registered in the orderID parameter). The specification of the advanced order status request is presented in the following sections:   * [Advanced Order Status Request (SOAP);](#_bookmark54) * [Advanced Order Status Request (REST).](#_bookmark68) |
| 19 | The payment gateway returns the payment status to the store. |
| 20 | The store displays a page to the client with the result of the order payment. |
| 21 | To debit funds from the client's account, the store must send a payment completion request to the payment gateway. The request specification is presented in the following sections:   * [Request for order payment completion (SOAP);](#_bookmark55) * [Request for order payment completion (REST)](#_bookmark69). |
| 22 | The payment gateway returns the result of processing the request. The order status is not refundable. To receive an order, one must send a corresponding request to the gateway, as described in step 14. |

* + - 1. Cancellation of order payment

Cancellation of order payment is available to stores if they have the appropriate rights (in agreement with the bank). For one-phase payments, payment cancellation is possible for orders in the **Completed**/**Deposited** state.

Cancellation of order payment is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, through REST/SOAP interfaces. The request specification is presented in the sections below.
  + [Request for order payment cancellation (SOAP).](#_bookmark56)
  + [Request for order payment cancellation (REST)](#_bookmark70).

If the cancellation operation is successful, the order will be transferred from the **Completed**/**Deposited** state **to the Canceled**/**Reversed** state.

* + - 1. Refund of order payment funds

A full or partial refund for paid orders (in the **Completed**/**Deposited** state) is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, via SOAP/REST interfaces. The request specification is presented in the sections below.
  + [Request for a refund of the order payment (SOAP).](#_bookmark57)
  + [Request for a refund of the order payment (REST).](#_bookmark71)

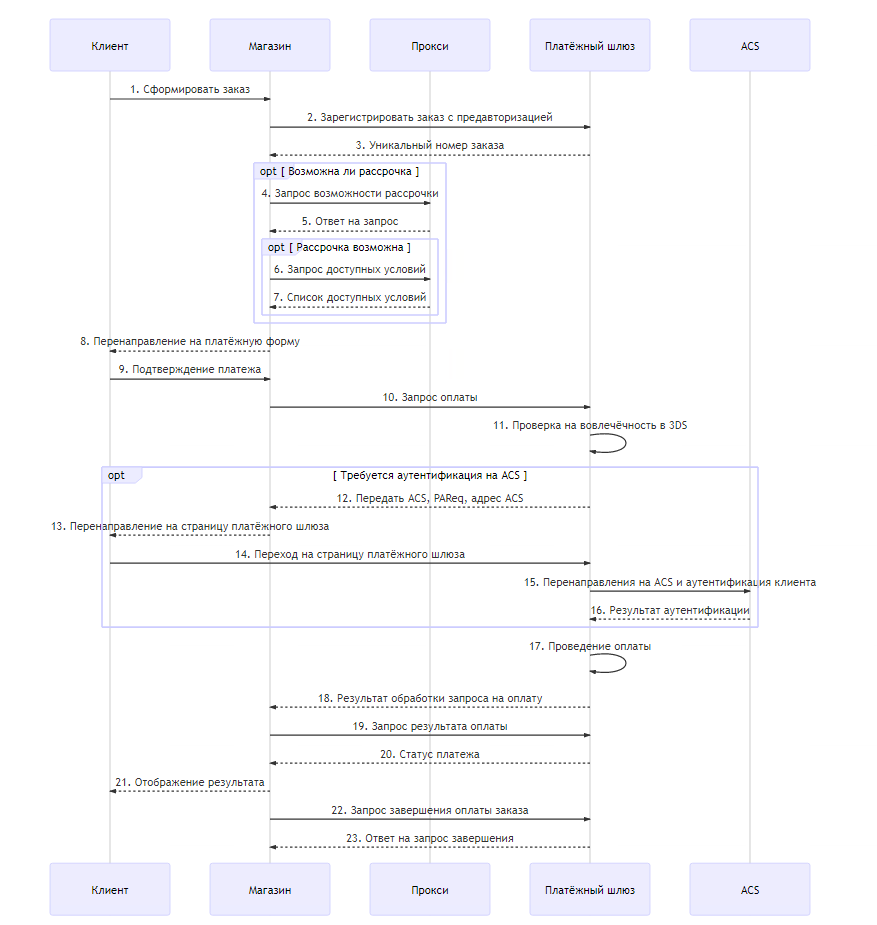
After IPS receives a refund request sent by one of the above methods, IPS returns the specified amount to the client's account.

* + - 1. Verification of the card's enrolment in 3DS

The system allows the store to independently verify the enrolment of the bank card in 3-D Secure, if necessary. This can be done using the API, via SOAP/REST interfaces. The request specification is presented in the following sections.

* [Request to verify the card's enrolment in 3DS (SOAP).](#_bookmark48)
* [Request to verify the card's enrolment in 3DS (REST)](#_bookmark62).
  + 1. Two-phase payment with return to the gateway after ACS authentication (simplified redirection of the client to ACS)

##### Order payment scenario



|  |  |
| --- | --- |
| **Клиент** | **Client** |
| **Магазин** | **Store** |
| **Прокси** | **Proxy** |
| **Платёжный шлюз** | **Payment Gateway** |
| **ACS** | **ACS** |
| 1. Сформировать заказ | 1. Generate an order |
| 2. Зарегистрировать заказ с предавторизацией | 2. Register the order with pre-authorization |
| 3. Уникальный номер заказа | 3. Unique order number |
| [ Возможна ли рассрочка ] | [Whether installment is possible] |
| 4. Запрос возможности рассрочки | 4. Request for the possibility of installments |
| 5. Ответ на запрос | 5. Response to the request |
| [ Рассрочка возможна ] | [Installments are possible] |
| 6. Запрос доступных условий | 6. Request for available conditions |
| 7. Список доступных условий | 7. List of available conditions |
| 8. Перенаправление на платёжную форму | 8. Redirection to the payment form |
| 9. Подтверждение платежа | 9. Payment confirmation |
| 10. Запрос оплаты | 10. Request for payment |
| 11. Проверка на вовлечёчность в 3DS | 11. Verification of 3DS enrolment |
| [ Требуется аутентификация на ACS ] | [ ACS authentication required ] |
| 12. Передать ACS, PAReq. адрес ACS | 12. Transfer ACS, PAReq, ACS address |
| 13. Перенаправление на страницу платежного шлюза | 13. Redirection to the payment gateway page |
| 14. Переход на страницу платежного шлюза | 14. Shift to the payment gateway page |
| 15. Перенаправление на ACS и аутентификация клиента | 15. Redirection to ACS and client authentication |
| 16. Результат аутентификации | 16. Authentication result |
| 17. Проведение оплаты | 17. Making the payment |
| 18. Результат обработки запроса на оплату | 18. Result of processing the payment request |
| 19. Запрос результата оплаты | 19. Request for payment result |
| 20. Статус платежа | 20. Payment status |
| 21. Отображение результата | 21. Displaying the result |
| 22. Запрос завершения оплаты заказа | 22. Request for order payment completion |
| 23. Ответ на запрос завершения | 23. Response to completion request |

##### Description

|  |  |
| --- | --- |
| 1 | The client generates an order on the RSO (online store) website and confirms the decision to pay for the goods with a bank card. |
| 2 | The store sends a request to register the order in the payment gateway. For registration, parameters such as the debit amount, the debit currency, the order number in the store's system, as well as the client's return URL are used. The request specification is presented in the following sections:   * [Request for order registration with pre-authorization (SOAP);](#_bookmark47) * [Request for order registration with pre-authorization (REST).](#_bookmark61) |
| 3 | In case of successful order registration, the response returns a unique order number in the payment gateway (in the orderID parameter). |
| 4 | In order to check whether the client's card number is included in the installment program, the merchant sends a [check request](#_bookmark74) to the payment gateway proxy, specifying the client's card BIN. |
|  | This opportunity is provided by agreement with the bank. |
| 5 | * If the request returns true, the merchant can make the next request (get) to find out what installment options are provided to the client. * If the request returns false, the client has no option to pay by installments, and the next request is skipped. |
| 6 | The merchant makes a get [request in the proxy of the payment gateway](#_bookmark75) to get information about the installment options for the client. |
| 7 | The payment gateway proxy returns a response with installment options. Subsequently, to make payment by installments, one will need to pass the received parameters as additional in the payment request (paymentOrder). |
| 8 | The online store transfers the cardholder's browser to its payment page to enter card data. |
| 9 | The cardholder enters the card details on the payment page and transmits the data to the online store. |
| 10 | After receiving the payment details, the store sends a request to the payment gateway to pay for the order. The request transmits parameters such as the order number in the payment gateway (received in step 9) and card details. The request specification is presented in the following sections:   * [Order payment request (SOAP);](#_bookmark51) * [Order payment request (REST).](#_bookmark65) |
| 11 | After receiving the payment details, the system verifies the enrolment of the card in 3-D Secure, including using Directory Server.  If the payment requires the use of 3-D Secure technology, the next step of the scenario is performed.  If the payment has to go through SSL, then the scenario execution continues from **Step 17**. |
| 12 | The verification result is sent to the store (card enrolment + URL of the Issuing Bank's ACS server where the client needs to be authenticated). |
| 13 | The store's system sends a redirection to the payment gateway page to the client's browser using the following link: https://tws.egopay.ru/ab/acsRedirect.do?orderId =<order number> where <order number> is the unique order number of the client. |
|  | The example contains the address of the test server of the payment gateway. |
| 14 | The client goes to the payment gateway page. |
| 15 | Without additional actions on the part of the client, the payment gateway redirects the client to the Issuing bank's ACS page, where the client is authenticated. |
| 16 | The Issuing bank's ACS redirects the client to the payment gateway by transmitting PARes. If the client fails to authenticate through ACS, the payment gateway will reject the payment. If authentication is successful, the following scenario is executed. |
| 17 | The payment gateway authorizes the payment. |
| 18 | The result of processing the order payment request is returned to the store. |
| 19 | The store's system requests the order payment status from the payment gateway (by the unique order ID in the payment gateway, which was received when the order was registered in the orderID parameter). The specification of the order status request is presented in the sections:   * [Advanced Order Status Request (SOAP);](#_bookmark54) * [Advanced Order Status Request (REST).](#_bookmark68) |
| 20 | The payment gateway returns the payment status to the store. |
| 21 | The store displays a page to the client with the result of the order payment. |
| 22 | To debit funds from the client's account, the store must send a payment completion request to the payment gateway. The request specification is presented in the following sections:   * [Request for order payment completion (SOAP);](#_bookmark55) * [Request for order payment completion (REST)](#_bookmark69). |
| 23 | The payment gateway returns the result of processing the request. The order status is not refundable. To receive an order, one must send a corresponding request to the gateway, as described in step 14. |

* + 1. 1. Cancellation of order payment

Cancellation of order payment is available to stores if they have the appropriate rights (in agreement with the bank). For one-phase payments, payment cancellation is possible for orders in the **Completed**/**Deposited** state.

Cancellation of order payment is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, through REST/SOAP interfaces. The request specification is presented in the sections below.
  + [Request for order payment cancellation (SOAP).](#_bookmark56)
  + [Request for order payment cancellation (REST)](#_bookmark70).

If the cancellation operation is successful, the order will be transferred from the **Completed**/**Deposited** state **to the Canceled**/**Reversed** state.

* + - 1. Refund of order payment funds

A full or partial refund for paid orders (in the **Completed**/**Deposited** state) is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, via SOAP/REST interfaces. The request specification is presented in the sections below.
  + [Request for a refund of the order payment (SOAP).](#_bookmark57)
  + [Request for a refund of the order payment (REST).](#_bookmark71)

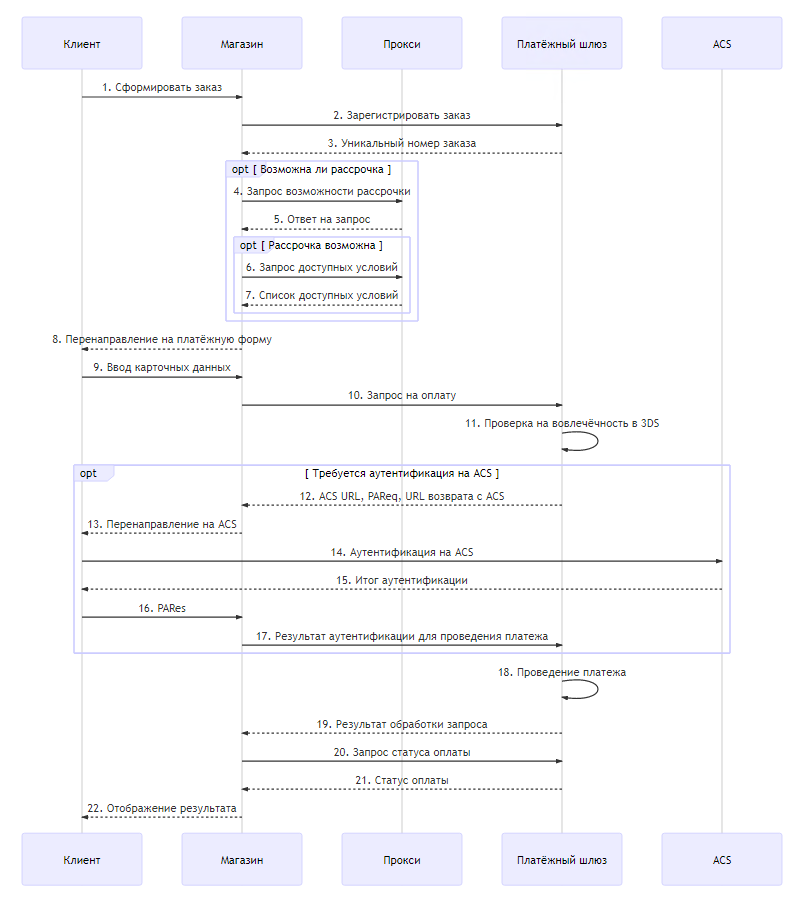
After IPS receives a refund request sent by one of the above methods, IPS returns the specified amount to the client's account.

* + - 1. Verification of the card's enrolment in 3DS

The system allows the store to independently verify the enrolment of the bank card in 3-D Secure, if necessary. This can be done using the API, via SOAP/REST interfaces. The request specification is presented in the following sections.

* [Request to verify the card's enrolment in 3DS (SOAP).](#_bookmark48)
* [Request to verify the card's enrolment in 3DS (REST)](#_bookmark62).
  + 1. One-phase payment with return to the store after ACS authentication

##### Order payment scenario



|  |  |
| --- | --- |
| **Клиент** | **Client** |
| **Магазин** | **Store** |
| **Прокси** | **Proxy** |
| **Платёжный шлюз** | **Payment Gateway** |
| **ACS** | **ACS** |
| 1. Сформировать заказ | 1. Generate an order |
| 2. Зарегистрировать заказ с предавторизацией | 2. Register the order with pre-authorization |
| 3. Уникальный номер заказа | 3. Unique order number |
| [ Возможна ли рассрочка ] | [Whether installment is possible] |
| 4. Запрос возможности рассрочки | 4. Request for the possibility of installments |
| 5. Ответ на запрос | 5. Response to the request |
| [ Рассрочка возможна ] | [Installments are possible] |
| 6. Запрос доступных условий | 6. Request for available conditions |
| 7. Список доступных условий | 7. List of available conditions |
| 8. Перенаправление на платёжную форму | 8. Redirection to the payment form |
| 9. Ввод карточных данных | 9. Entering card data |
| 10. Запрос оплаты | 10. Request for payment |
| 11. Проверка на вовлечёчность в 3DS | 11. Verification of 3DS enrolment |
| [ Требуется аутентификация на ACS ] | [ ACS authentication required ] |
| 12. ACS URL, PAReq. URL возврата c ACS | 12. ACS URL, PAReq, URL for return from ACS |
| 13. Перенаправление на ACS | 13. Redirection to ACS |
| 14. Аутентификация на ACS | 14. ACS authentication |
| 15. Итог аутентификации | 15. Authentication result |
| 16. PARes | 16. PARes |
| 17. Результат аутентификации для проведения платежа | 17. Authentication result for making a payment |
| 18. Проведение оплаты | 18. Making the payment |
| 19. Результат обработки запроса | 19. Result of processing the request |
| 20. Запрос статуса запроса | 20. Request status request |
| 21. Статус платежа | 21. Payment status |
| 22. Отображение результата | 22. Displaying the result |

##### Description

|  |  |
| --- | --- |
| 1 | The client generates an order on the RSO (online store) website and confirms the decision to pay for the goods with a bank card. |
| 2 | The store sends a request to register the order in the payment gateway. For registration, parameters such as the debit amount, the debit currency, the order number in the store's system, as well as the client's return URL are used. The request specification is presented in the following sections:   * [Request for order registration (SOAP);](#_bookmark46) * [Request for order registration (REST).](#_bookmark60) |
| 3 | In case of successful order registration, the response returns a unique order number in the payment gateway (in the orderID parameter). |
| 4 | In order to check whether the client's card number is included in the installment program, the merchant sends a [check request](#_bookmark74) to the payment gateway proxy, specifying the client's card BIN. |
| This opportunity is provided by agreement with the bank. |
| 5 | * If the request returns true, the merchant can make the next request (get) to find out what installment options are provided to the client. * If the request returns false, the client has no option to pay by installments, and the next request is skipped. |
| 6 | The merchant makes a get [request in the proxy of the payment gateway](#_bookmark75) to get information about the installment options for the client. |
| 7 | The payment gateway proxy returns a response with installment options. Subsequently, to make payment by installments, one will need to pass the received parameters as additional in the payment request (paymentOrder). |
| 8 | The online store transfers the cardholder's browser to its payment page to enter card data. |
| 9 | The cardholder enters the card details on the payment page and transmits the data to the online store. |
| 10 | After receiving the payment details, the store sends a request to the payment gateway to pay for the order. The request transmits parameters such as the order number in the payment gateway (received in step 9) and card details. The request specification is presented in the following sections:   * [Request for payment via internal MPI (SOAP);](#_bookmark51) * [Request for payment via internal MPI (REST).](#_bookmark65) |
| 11 | After receiving the payment details, the system verifies the enrolment of the card in 3-D Secure, including using Directory Server.  If the payment requires the use of 3-D Secure technology, the next step of the scenario is performed.  If the payment has to go through SSL, then the scenario execution continues from **Step 18.** |
| 12 | The result of verification for the need to use 3DS technology is sent to the store (card enrolment + URL of the Issuing Bank's ACS server where the client needs to be authenticated). |
| 13 | The store's system sends a redirection to the issuer's ACS to the client's browser. |
| 14 | The Client authenticates through ACS of the Issuing bank. |
| 15 | The Issuing Bank's ACS sends a redirection to the store to the client's browser. |
| 16 | The PARes received from ACS is transferred to the store. |
| 17 | The store, using the finish3DS request, transmits to the gateway the received PARes required for making the payment. The request specification is presented in the following sections:   * [finish3DS request (SOAP);](#_bookmark52) * [finish3DS request (REST).](#_bookmark66) |
| 18 | The payment gateway authorizes the payment. |
| 19 | The result of processing the request is returned to the store. |
| 20 | The store's system requests the order payment status from the payment gateway (by the unique order ID in the payment gateway, which was received when the order was registered in the orderID parameter). The specification of the order status request is presented in the sections:   * [Advanced Order Status Request (SOAP);](#_bookmark54) * [Advanced Order Status Request (REST).](#_bookmark68) |
| 21 | The payment gateway returns the payment status to the store. |
| 22 | The store displays a page to the client with the result of the order payment. |

* + - 1. Cancellation of order payment

Cancellation of order payment is available to stores if they have the appropriate rights (in agreement with the bank). For one-phase payments, payment cancellation is possible for orders in the **Completed**/**Deposited** state.

Cancellation of order payment is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, through REST/SOAP interfaces. The request specification is presented in the sections below.
  + [Request for order payment cancellation (SOAP).](#_bookmark56)
  + [Request for order payment cancellation (REST)](#_bookmark70).

If the cancellation operation is successful, the order will be transferred from the **Completed**/**Deposited** state **to the Canceled**/**Reversed** state.

* + - 1. Refund of order payment funds

A full or partial refund for paid orders (in the **Completed**/**Deposited** state) is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, via SOAP/REST interfaces. The request specification is presented in the sections below.
  + [Request for a refund of the order payment (SOAP).](#_bookmark57)
  + [Request for a refund of the order payment (REST).](#_bookmark71)

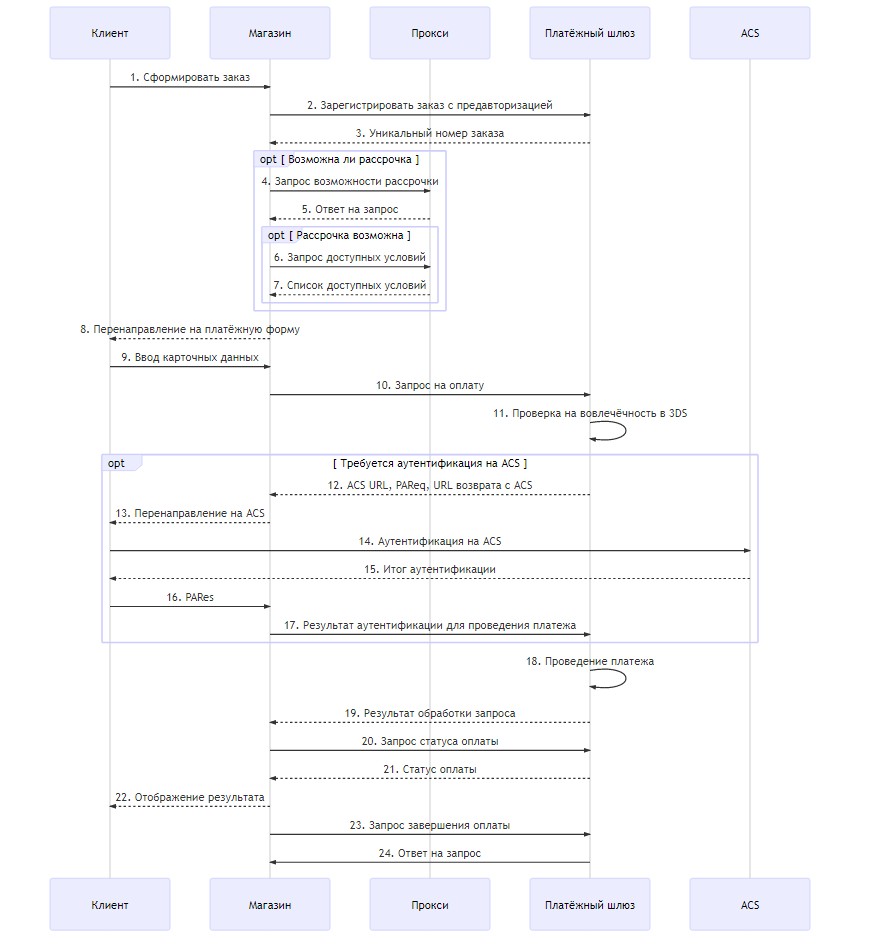
After IPS receives a refund request sent by one of the above methods, IPS returns the specified amount to the client's account.

* + - 1. Verification of the card's enrolment in 3DS

The system allows the store to independently verify the enrolment of the bank card in 3-D Secure, if necessary. This can be done using the API, via SOAP/REST interfaces. The request specification is presented in the following sections.

* [Request to verify the card's enrolment in 3DS (SOAP).](#_bookmark48)
* [Request to verify the card's enrolment in 3DS (REST)](#_bookmark62).
  + 1. Two-phase payment with return to the store after ACS authentication

##### Order payment scenario



|  |  |
| --- | --- |
| **Клиент** | **Client** |
| **Магазин** | **Store** |
| **Прокси** | **Proxy** |
| **Платёжный шлюз** | **Payment Gateway** |
| **ACS** | **ACS** |
| 1. Сформировать заказ | 1. Generate an order |
| 2. Зарегистрировать заказ с предавторизацией | 2. Register the order with pre-authorization |
| 3. Уникальный номер заказа | 3. Unique order number |
| [ Возможна ли рассрочка ] | [Whether installment is possible] |
| 4. Запрос возможности рассрочки | 4. Request for the possibility of installments |
| 5. Ответ на запрос | 5. Response to the request |
| [ Рассрочка возможна ] | [Installments are possible] |
| 6. Запрос доступных условий | 6. Request for available conditions |
| 7. Список доступных условий | 7. List of available conditions |
| 8. Перенаправление на платёжную форму | 8. Redirection to the payment form |
| 9. Ввод карточных данных | 9. Entering card data |
| 10. Запрос оплаты | 10. Request for payment |
| 11. Проверка на вовлечёчность в 3DS | 11. Verification of 3DS enrolment |
| [ Требуется аутентификация на ACS ] | [ ACS authentication required ] |
| 12. ACS URL, PAReq, URL возврата c ACS | 12. ACS URL, PAReq, URL return from ACS |
| 13. Перенаправление на ACS | 13. Redirection to ACS |
| 14. Аутентификация на ACS | 14. ACS authentication |
| 15. Итог аутентификации | 15. Authentication result |
| 16. PARes | 16. PARes |
| 17. Результат аутентификации для проведения платежа | 17. Authentication result for making a payment |
| 18. Проведение платежа | 18. Making the payment |
| 19. Результат обработки запроса | 19. Result of processing the request |
| 20. Запрос статуса запроса | 20. Request status request |
| 21. Статус платежа | 21. Payment status |
| 22. Отображение результата | 22. Displaying the result |
| 23. Запрос завершения оплаты | 23. Request for payment completion |
| 24. Ответ на запрос | 24. Response to the request |

##### Description:

|  |  |
| --- | --- |
| 1 | The client generates an order on the RSO (online store) website and confirms the decision to pay for the goods with a bank card. |
| 2 | The store sends a request to the payment gateway to register the order with pre-authorization. For registration, parameters such as the debit amount, the debit currency, the order number in the store's system, as well as the client's return URL are used. The request specification is presented in the following sections:   * [Request for order registration with pre-authorization (SOAP);](#_bookmark47) * [Request for order registration with pre-authorization (REST).](#_bookmark61) |
| 3 | In case of successful order registration, the response returns a unique order number in the payment gateway (in the orderID parameter). |
| 4 | In order to check whether the client's card number is included in the installment program, the merchant sends a [check request](#_bookmark74) to the payment gateway proxy, specifying the client's card BIN. |
|  | This opportunity is provided by agreement with the bank. |
| 5 | * If the request returns true, the merchant can make the next request (get) to find out what installment options are provided to the client. * If the request returns false, the client has no option to pay by installments, and the next request is skipped. |
| 6 | The merchant makes a get [request in the proxy of the payment gateway](#_bookmark75) to get information about the installment options for the client. |
| 7 | The payment gateway proxy returns a response with installment options. Subsequently, to make payment by installments, one will need to pass the received parameters as additional in the payment request (paymentOrder). |
| 8 | The online store transfers the cardholder's browser to its payment page to enter card data. |
| 9 | The cardholder enters the card details on the payment page and transmits the data to the online store. |
| 10 | After receiving the payment details, the store sends a request to the payment gateway to pay for the order. The request transmits parameters such as the order number in the payment gateway (received in step 9) and card details. The request specification is presented in the following sections:   * [Order payment request (SOAP);](#_bookmark51) * [Order payment request (REST).](#_bookmark65) |
| 11 | After receiving the payment details, the system verifies the enrolment of the card in 3-D Secure, including using Directory Server.  If the payment requires the use of 3-D Secure technology, the next step of the scenario is performed.  If the payment has to go through SSL, then the scenario execution continues from **Step 18**. |
| 12 | The result of verification for the need to use 3DS technology is sent to the store (card enrolment + URL of the Issuing Bank's ACS server where the client needs to be authenticated). |
| 13 | The store's system sends a redirection to the issuer's ACS to the client's browser. |
| 14 | The Client authenticates through ACS of the Issuing bank. |
| 15 | The Issuing Bank's ACS sends a redirection to the store to the client's browser. |
| 16 | The PARes received from ACS is transferred to the store. |
| 17 | The store, using the finish3DS request, transmits to the gateway the received PARes required for making the payment. The request specification is presented in the following sections:   * [finish3DS request (SOAP);](#_bookmark52) * [finish3DS request (REST)](#_bookmark66). |
| 18 | The payment gateway authorizes the payment. |
| 19 | The result of processing the request is returned to the store. |
| 20 | The store's system requests the order payment status from the payment gateway (by the unique orderID in the payment gateway, which was received when the order was registered in the orderID parameter). The specification of the advanced order status request is presented in the following sections:   * [Advanced Order Status Request (SOAP);](#_bookmark54) * [Advanced Order Status Request (REST).](#_bookmark68) |
| 21 | The payment gateway returns the payment status to the store. |
| 22 | The store displays a page to the client with the result of the order payment. |
| 23 | To debit funds from the client's account, the store must send a payment completion request to the payment gateway. The request specification is presented in the following sections:   * [Request for order payment completion (SOAP);](#_bookmark55) * [Request for order payment completion (REST)](#_bookmark69). |
| 24 | The payment gateway returns the result of processing the request. The order status is not refundable. To receive an order, one must send a corresponding request to the gateway, as described in step |

* + - 1. Cancellation of order payment

Cancellation of order payment is available to stores if they have the appropriate rights (in agreement with the bank). For one-phase payments, payment cancellation is possible for orders in the **Completed**/**Deposited** state.

Cancellation of order payment is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, through REST/SOAP interfaces. The request specification is presented in the sections below.
  + [Request for order payment cancellation (SOAP).](#_bookmark56)
  + [Request for order payment cancellation (REST)](#_bookmark70).

If the cancellation operation is successful, the order will be transferred from the **Completed**/**Deposited** state **to the Canceled**/**Reversed** state.

* + - 1. Refund of order payment funds

A full or partial refund for paid orders (in the **Completed**/**Deposited** state) is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, via SOAP/REST interfaces. The request specification is presented in the sections below.
  + [Request for a refund of the order payment (SOAP).](#_bookmark57)
  + [Request for a refund of the order payment (REST).](#_bookmark71)

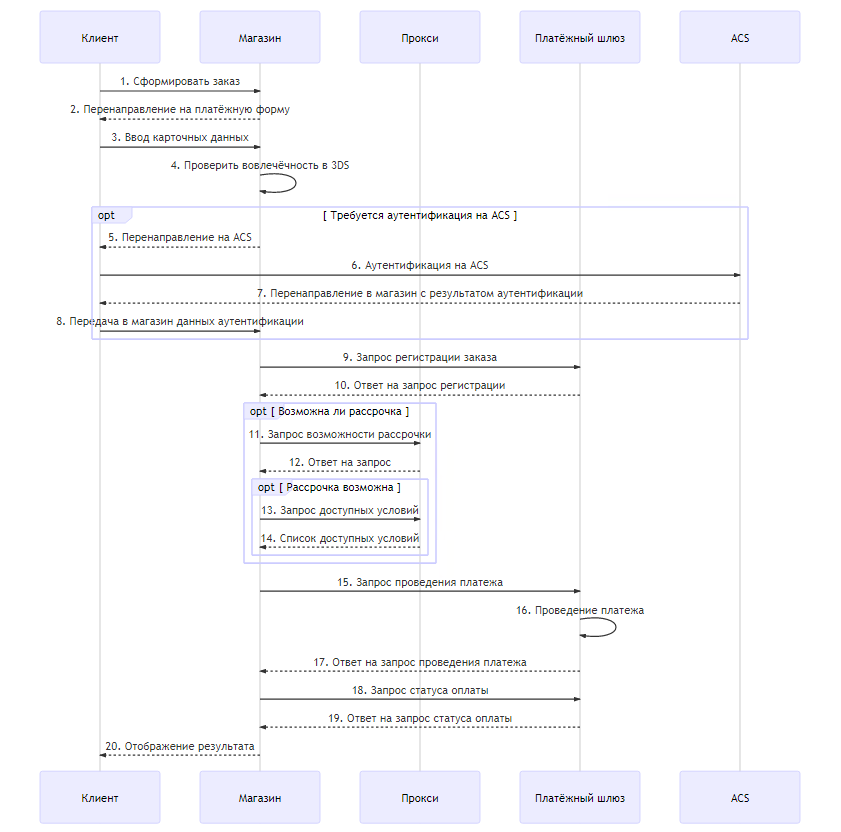
After IPS receives a refund request sent by one of the above methods, IPS returns the specified amount to the client's account.

* + - 1. Verification of the card's enrolment in 3DS

The system allows the store to independently verify the enrolment of the bank card in 3-D Secure, if necessary. This can be done using the API, via SOAP/REST interfaces. The request specification is presented in the following sections.

* [Request to verify the card's enrolment in 3DS (SOAP).](#_bookmark48)
* [Request to verify the card's enrolment in 3DS (REST)](#_bookmark62).
  1. Payment via external MPI
     1. One-phase payment

##### Order payment scenario



|  |  |
| --- | --- |
| **Клиент** | **Client** |
| **Магазин** | **Store** |
| **Прокси** | **Proxy** |
| **Платёжный шлюз** | **Payment Gateway** |
| **ACS** | **ACS** |
| 1. Сформировать заказ | 1. Generate an order |
| 2. Перенаправление на платёжную форму | 2. Redirection to the payment form |
| 3. Ввод карточных данных | 3. Entering card data |
| 4. Проверить вовлечёчность в 3DS | 4. Verify the 3DS enrolment |
| [ Требуется аутентификация на ACS ] | [ ACS authentication required ] |
| 5. Перенаправление на ACS | 5. Redirection to ACS |
| 6. Аутентификация на ACS | 6. ACS authentication |
| 7. Перенаправление в магазин с результатом аутентификации | 7. Redirection to the store with the authentication result |
| 8. Передача в магазин данных аутентификации | 8. Transfer of authentication data to the store |
| 9. Запрос регистрации заказа | 9. Request for order registration |
| 10. Ответ на запрос регистрации | 10. Response to the registration request |
| [ Возможна ли рассрочка ] | [Whether installment is possible] |
| 11. Запрос возможности рассрочки | 11. Request the possibility of installments |
| 12. Ответ на запрос | 12. Response to the request |
| [ Рассрочка возможна ] | [Installments are possible] |
| 13. Запрос доступных условий | 13. Request for available conditions |
| 14. Список доступных условий | 14. List of available conditions |
| 15. Запрос проведения платежа | 15. Request to make a payment |
| 16. Проведение платежа | 16. Making the payment |
| 17. Ответ на запрос проведения платежа | 17. Response to the request to make a payment |
| 18. Запрос статуса оплаты | 18. Payment status request |
| 19. Ответ на запрос статуса оплаты | 19. Response to the payment status request |
| 20. Отображение результата | 20. Displaying the result |

##### Description:

|  |  |
| --- | --- |
| 1 | The client generates an order on the RSO (online store) website and confirms the decision to pay for the goods with a bank card. |
| 2 | The online store redirects the client's browser to its payment page in order to enter card details. |
| 3 | The cardholder enters the card details on the payment page and transmits the data to the online store. |
| 4 | After receiving the payment details, the store verifies the enrolment of the card in 3-D Secure, including using Directory Server.  If the payment requires the use of 3-D Secure technology, **Step 5** is performed.  If the payment has to go through SSL, the scenario execution continues from **Step 9**. |
| 5 | The store's system sends a redirection to the Issuing Bank's ACS to the client's browser. |
| 6 | The Client authenticates through ACS of the Issuing bank. |
| 7 | After authentication of the cardholder, the Issuing Bank's ACS redirects the client to the store, transmitting the encrypted parameters of the authorization result. |
| 8 | The client sends authentication data to the store. |
| 9 | The store sends a request for order registration to the payment gateway. The request specification is presented in the following sections:   * [Request for order registration (SOAP);](#_bookmark46) * [Request for order registration (REST).](#_bookmark60) |
| 10 | In case of successful order registration, the response returns a unique order number in the payment gateway (in the orderID parameter). |
| 11 | In order to check whether the client's card number is included in the installment program, the merchant sends a [check request](#_bookmark74) to the payment gateway proxy, specifying the client's card BIN. |
|  | This opportunity is provided by agreement with the bank. |
| 12 | * If the request returns true, the merchant can make the next request (get) to find out what installment options are provided to the client. * If the request returns false, the client has no option to pay by installments, and the next request is skipped. |
| 13 | The merchant makes a get [request in the proxy of the payment gateway](#_bookmark75) to get information about the installment options for the client. |
| 14 | The payment gateway proxy returns a response with installment options. Subsequently, to make payment by installments, one will need to pass the received parameters as additional in the payment request (paymentOrder). |
| 15 | The store sends a request for order payment to the payment gateway. The request transmits parameters such as the order number in the payment gateway (received in step 9) and card details. If 3-D Secure technology was used during payment, the parameters of the ACS authorization result (cavv, xid, eci) should also be transmitted. The request specification is presented in the following sections:   * [Order payment request (SOAP);](#_bookmark50) * [Order payment request (REST).](#_bookmark64) |
| 16 | The payment gateway makes the payment. |
| 17 | The payment gateway returns the result of processing the request to the store. |
| 18 | The store sends a request to the payment gateway to receive the payment status of the order. The request specification is presented in the following sections:   * [Advanced Order Status Request (SOAP);](#_bookmark54) * [Advanced Order Status Request (REST).](#_bookmark68) |
| 19 | The payment gateway returns the payment status to the store. |
| 20 | The store displays a page to the client with the result of the order payment. |

* + - 1. Cancellation of order payment

Cancellation of order payment is available to stores if they have the appropriate rights (in agreement with the bank). For one-phase payments, payment cancellation is possible for orders in the **Completed**/**Deposited** state.

Cancellation of order payment is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, through REST/SOAP interfaces. The request specification is presented in the sections below.
  + [Request for order payment cancellation (SOAP).](#_bookmark56)
  + [Request for order payment cancellation (REST)](#_bookmark70).

If the cancellation operation is successful, the order will be transferred from the **Completed**/**Deposited** state **to the Canceled**/**Reversed** state.

* + - 1. Refund of order payment funds

A full or partial refund for paid orders (in the **Completed**/**Deposited** state) is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, via SOAP/REST interfaces. The request specification is presented in the sections below.
  + [Request for a refund of the order payment (SOAP).](#_bookmark57)
  + [Request for a refund of the order payment (REST).](#_bookmark71)

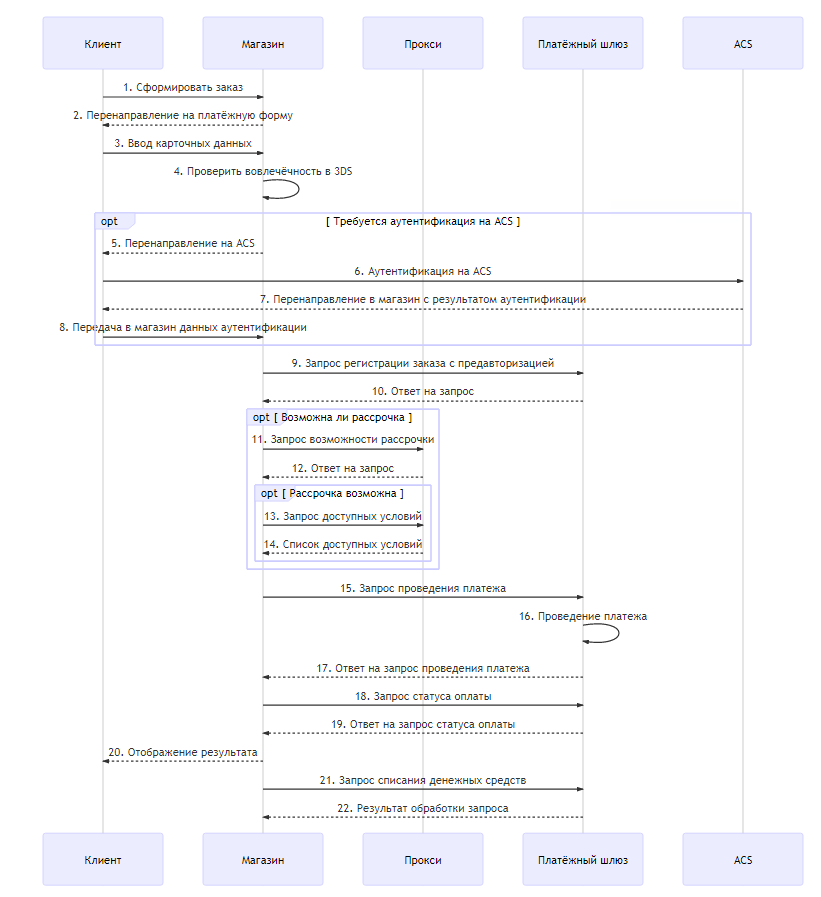
After IPS receives a refund request sent by one of the above methods, IPS returns the specified amount to the client's account.

* + - 1. Verification of the card's enrolment in 3DS

The system allows the store to independently verify the enrolment of the bank card in 3-D Secure, if necessary. This can be done using the API, via SOAP/REST interfaces. The request specification is presented in the following sections.

* [Request to verify the card's enrolment in 3DS (SOAP).](#_bookmark48)
* [Request to verify the card's enrolment in 3DS (REST)](#_bookmark62).
  + 1. Two-phase payment

##### Order payment scenario



|  |  |
| --- | --- |
| **Клиент** | **Client** |
| **Магазин** | **Store** |
| **Прокси** | **Proxy** |
| **Платёжный шлюз** | **Payment Gateway** |
| **ACS** | **ACS** |
| 1. Сформировать заказ | 1. Generate an order |
| 2. Перенаправление на платёжную форму | 2. Redirection to the payment form |
| 3. Ввод карточных данных | 3. Entering card data |
| 4. Проверить вовлечёчность в 3DS | 4. Verify the 3DS enrolment |
| [ Требуется аутентификация на ACS ] | [ ACS authentication required ] |
| 5. Перенаправление на ACS | 5. Redirection to ACS |
| 6. Аутентификация на ACS | 6. ACS authentication |
| 7. Перенаправление в магазин с результатом аутентификации | 7. Redirection to the store with the authentication result |
| 8. Передача в магазин данных аутентификации | 8. Transfer of authentication data to the store |
| 9. Запрос регистрации заказа с предавторизацией | 9. Request for order registration with pre-authorization |
| 10. Ответ на запрос | 10. Response to the request |
| [ Возможна ли рассрочка ] | [Whether installment is possible] |
| 11. Запрос возможности рассрочки | 11. Request the possibility of installments |
| 12. Ответ на запрос | 12. Response to the request |
| [ Рассрочка возможна ] | [Installments are possible] |
| 13. Запрос доступных условий | 13. Request for available conditions |
| 14. Список доступных условий | 14. List of available conditions |
| 15. Запрос проведения платежа | 15. Request to make a payment |
| 16. Проведение платежа | 16. Making the payment |
| 17. Ответ на запрос проведения платежа | 17. Response to the request to make a payment |
| 18. Запрос статуса оплаты | 18. Payment status request |
| 19. Ответ на запрос статуса оплаты | 19. Response to the payment status request |
| 20. Отображение результата | 20. Displaying the result |
| 21. Запрос списания денежных средств | 21. Request to debit funds |
| 22. Результат обработки запроса | 22. Result of processing the request |

##### Description:

|  |  |
| --- | --- |
| 1 | The client generates an order on the RSO (online store) website and confirms the decision to pay for the goods with a bank card. |
| 2 | The online store redirects the client's browser to its payment page in order to enter card details. |
| 3 | The cardholder enters the card details on the payment page and transmits the data to the online store. |
| 4 | After receiving the payment details, the store verifies the enrolment of the card in 3-D Secure, including using Directory Server.  If the payment requires the use of 3-D Secure technology, the next **Step 5** is performed.  If the payment has to go through SSL, the scenario execution continues from **Step 9**. |
| 5 | The store's system sends a redirection to the Issuing Bank's ACS to the client's browser. |
| 6 | The Client authenticates through ACS of the Issuing bank. |
| 7 | After authentication of the cardholder, the Issuing Bank's ACS redirects the client' to the store, transmitting the encrypted parameters of the authorization result. |
| 8 | The client transmits the authentication data to the store. |
| 9 | The store sends a request to the payment gateway to register the order with pre-authorization. The request specification is presented in the following sections:   * [Request for order registration with pre-authorization (SOAP);](#_bookmark47) * [Request for order registration with pre-authorization (REST).](#_bookmark61) |
| 10 | In case of successful order registration, the response returns a unique order number in the payment gateway (in the orderID parameter). |
| 11 | In order to check whether the client's card number is included in the installment program, the merchant sends a [check request](#_bookmark74) to the payment gateway proxy, specifying the client's card BIN. |
| This opportunity is provided by agreement with the bank. |
| 12 | * If the request returns true, the merchant can make the next request (get) to find out what installment options are provided to the client. * If the request returns false, the client has no option to pay by installments, and the next request is skipped. |
| 13 | The merchant makes a get [request in the proxy of the payment gateway](#_bookmark75) to get information about the installment options for the client. |
| 14 | The payment gateway proxy returns a response with installment options. Subsequently, to make payment by installments, one will need to pass the received parameters as additional in the payment request (paymentOrder). |
| 15 | The store sends a request for order payment to the payment gateway. The request transmits parameters such as the order number in the payment gateway (received in step 9) and card details. If 3-D Secure technology was used during payment, the parameters of the ACS authorization result (cavv, xid, eci) should also be transmitted. The request specification is presented in the following sections:   * [Order payment request (SOAP);](#_bookmark50) * [Order payment request (REST).](#_bookmark64) |
| 16 | The payment gateway makes the payment. |
| 17 | The payment gateway returns the result of processing the request to the store. |
| 18 | The store's system requests the order payment status from the payment gateway (by the unique order ID in the payment gateway, which was received when the order was registered in the orderID parameter).   * [Advanced Order Status Request (SOAP);](#_bookmark54) * [Advanced Order Status Request (REST).](#_bookmark68) |
| 19 | The payment gateway returns the payment status to the store. |
| 20 | The store displays a page to the client with the result of the order payment. |
| 21 | To debit funds from the client's account, the store must send a payment completion request to the payment gateway. The request specification is presented in the following sections:   * [Request for order payment completion (SOAP);](#_bookmark55) * [Request for order payment completion (REST)](#_bookmark69). |
| 22 | The payment gateway returns the result of processing the request. The order status is not refundable. To receive an order, one must send a corresponding request to the gateway, as described in step 18. |

* + - 1. Cancellation of order payment

Cancellation of order payment is available to stores if they have the appropriate rights (in agreement with the bank). For one-phase payments, payment cancellation is possible for orders in the **Completed**/**Deposited** state.

Cancellation of order payment is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, through REST/SOAP interfaces. The request specification is presented in the sections below.
  + [Request for order payment cancellation (SOAP).](#_bookmark56)
  + [Request for order payment cancellation (REST)](#_bookmark70).

If the cancellation operation is successful, the order will be transferred from the **Completed**/**Deposited** state **to the Canceled**/**Reversed** state.

* + - 1. Refund of order payment funds

A full or partial refund for paid orders (in the **Completed**/**Deposited** state) is carried out by standard means (see the list below).

* Through the administrative console (a description is provided in the document "[Instructions](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308) [for working with the console](https://rbs-develop.paymentgate.ru/wiki/pages/viewpage.action?pageId=12976308)", section "Working with orders").
* Using the API, via SOAP/REST interfaces. The request specification is presented in the sections below.
  + [Request for a refund of the order payment (SOAP).](#_bookmark57)
  + [Request for a refund of the order payment (REST).](#_bookmark71)

After IPS receives a refund request sent by one of the above methods, IPS returns the specified amount to the client's account.

* + - 1. Verification of the card's enrolment in 3DS

The system allows the store to independently verify the enrolment of the bank card in 3-D Secure, if necessary. This can be done using the API, via SOAP/REST interfaces. The request specification is presented in the following sections.

* [Request to verify the card's enrolment in 3DS (SOAP).](#_bookmark48)
* [Request to verify the card's enrolment in 3DS (REST)](#_bookmark62).

1. Request Specifications
   1. Web-Service interface
      1. Request for order registration (WS)

The Request for order registration is called registerOrder, it is described in the WSDL of the service.

##### Request parameters:

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| merchantOrderNumber | ANS..32 | yes | The order number (identifier) in the store's system is unique for each store within the system |
| description | ANS..598 | no | Description of the order in any form. |
|  |  |  | To enable sending this field to processing, contact technical support. |
| amount | N..12 | yes | Payment amount in minimum currency units |
| currency | N3 | no | The payment currency code is ISO 4217. If not specified, the default value is used. |
| language | A2 | no | The language is encoded in ISO 639-1. If not specified, the default language specified in the store settings will be used. |
| pageView | ANS..20 | no | The value of this parameter determines which pages of the payment interface should be loaded for the client. Possible values are:   * DESKTOP – for pages designed to be displayed on PC screens (pages with names payment\_<locale>.html and errors\_<locale>.html will be searched for in the archive of the payment interface pages); * MOBILE – for pages designed to be displayed on mobile devices (pages with names mobile\_payment\_<locale>.html and mobile\_errors\_<locale>.html will be searched for in the archive of the payment interface pages); * If the store has added arbitrary prefixes to their page file names, value of the desired prefix should be passed in the pageView parameter for the corresponding pages to be loaded. For example, if iphone is passed in the parameter, pages with names iphone\_payment\_<locale>.html and iphone\_error\_<locale>.html will be searched for in the archive of the payment interface pages.   Where:  locale – l anguage of the page encoded in ISO 639-1. For example, ru for Russian or en for English.  If the parameter is missing or does not match the format, the default value is pageView=DESKTOP. |
| sessionTimeoutSecs | N...9 | no | The order lifetime in seconds.  If the parameter is not specified, the value specified in the merchant's settings or the default time (1200 seconds = 20 minutes) will be used.  If the request contains the expirationDate parameter, the value of the sessionTimeoutSecs parameter is ignored. |
| bindingId | AN..255 | no | ID of the binding created earlier. It can only be used if the store has permission to work with bindings. If this parameter is passed in this request, it means that:   1. This order can only be paid using a binding; 2. The payer will be redirected to the payment page, where only CVC input is required. |
| expirationDate | ANS | no | Date and time of the order lifetime expiration. Format: yyyy-MM-dd'T'HH:mm:ss.  If this parameter is not present in the request, then sessionTimeoutSecs is used to determine the time of the order lifetime expiration. |
| returnUrl | ANS..512 | yes | The address to which the user must be redirected in case of successful 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>**. |
|  |  |  | See also the **3DS payment** block – for one-phase and two-phase payments. returnUrl is used in the same way as in step 11 in the specified patterns. |
| failUrl | ANS..512 | no | The address to which the user must be redirected in case of 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>**. |
| See also the **3DS payment** block – for one-phase and two-phase payments. failUrl is used in the same way as in step 11 in the specified patterns. |
| dynamicCallbackUrl | Unable to render include or excerpt- include. Could not retrieve page. | no | Unable to render include or excerpt-include. Could not retrieve page. |
| params | See description | no | Tag containing attributes to pass additional parameters of the merchant.  The fields for additional information and its subsequent storage. To pass N parameters, the request must contain N params tags, where the name attribute contains the name, and the value attribute contains its value: |
|  |  | |  |  |  |  | | --- | --- | --- | --- | | **Name** | **Type** | **Mandatory** | **Description** | | name | ANS..255 bytes | yes | Name of the additional parameter | | value | ANS..1024 bytes | yes | Value of the additional parameter |   These fields can be passed to the Bank processing system to be subsequently displayed in the registries.\*  This functionality can be enabled upon additional agreement with the bank during the integration period. |
| In case of payments for housing and communal services, the following additional parameters must be passed:   * payment\_doc\_id – the identifier of the payment document; * order\_num – the order number. |
| If sending notifications to a buyer is set up for a seller, the email address of a buyer must be passed in this tag in the email parameter. |
| clientId | AN..255 | no | The client's number (ID) in the store's system. It is used to implement the functionality of bindings. It can be present if creating bindings is allowed for the store. |
| Specifying this parameter when processing payments with the use of bindings is mandatory. Otherwise, a payment will be failed. |
| merchantLogin | AN..255 | no | To register an order on behalf of a child merchant, specify the merchant login in this parameter. |
| features | ANS..255 | no | A container for feature parameter, where the following values can be specified.  AUTO\_PAYMENT – if the request for order registration initiates auto-payments.  VERIFY – 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. |
| **Specifics of passing the VERIFY value** 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. |
| FORCE\_TDS – use of 3-D Secure is enforced for the payment. If the card does not support 3-D Secure, the transaction will fail.  FORCE\_SSL – use of SSL is enforced for the payment (no 3-D Secure).  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.  Example of use is given below. |
| <**features**>  <**feature**>AUTO\_PAYMENT</**feature**>  </**features**> |
| email | ANS..40 | no | Buyer's email address. |
| phone | ANS.12 | no | Buyer'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:  • +79998887766;  • 9998887766. |

**\*** By default, the following fields are passed to the bank's 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 | The order number in the payment system. It is unique within the system. It is missing if the order registration failed due to an error detailed in ErrorCode. |
| formUrl | AN..512 | no | URL of the payment form to which the client's browser is to be redirected. It is not returned if the order registration failed due to an error detailed in ErrorCode. |
| errorCode | N3 | no | Error code. |
| errorMessage | AN..512 | no | Error description in the language passed in the language parameter in the request. |

**Error codes (**ErrorCode **field):**

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 1 | Invalid order number |
| 1 | Order with given order number is processed already |
| 3 | Currency is not defined |
| 4 | Empty amount |
| 4 | The order number cannot be empty |
| 4 | The return URL cannot be empty |
| 5 | A parameter value has been specified incorrectly |
| 5 | Access denied |
| 5 | The user must change password |
| 7 | System error |
| 13 | Using both values, Features FORCE\_TDS/FORCE\_SSL and AUTO\_PAYMENT, is not allowed |
| 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="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<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="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<**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>

* + 1. Request for order registration with pre-authorization (WS)

|  |
| --- |
| The ability to make two-phase payments is available on a separate request. |

The order pre-authorization request is called registerOrderPreAuth.

##### Request parameters:

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| merchantOrderNumber | ANS..32 | yes | The order number (identifier) in the store's system is unique for each store within the system |
| description | ANS..598 | no | Description of the order in any form. |
|  |  |  | To enable sending this field to processing, contact technical support. |
| amount | N..12 | yes | Payment amount in kopecks (or cents) |
| currency | N3 | no | The payment currency code is ISO 4217. If not specified, the default value is used. |
| language | A2 | no | The language is encoded in ISO 639-1. If not specified, the default language specified in the store settings will be used. |
| pageView | ANS..20 | no | The value of this parameter determines which pages of the payment interface should be loaded for the client. Possible values are:   * DESKTOP – for pages designed to be displayed on PC screens (pages with names payment\_<locale>.html and errors\_<locale>.html will be searched for in the archive of the payment interface pages); * MOBILE – for pages designed to be displayed on mobile devices (pages with names mobile\_payment\_<locale>.html and mobile\_errors\_<locale>.html will be searched for in the archive of the payment interface pages); * If the store has added arbitrary prefixes to their page file names, value of the desired prefix should be passed in the pageView parameter for the corresponding pages to be loaded. For example, if iphone is passed in the parameter, pages with names iphone\_payment\_<locale>.html and iphone\_error\_<locale>.html will be searched for in the archive of the payment interface pages.   Where:  locale – language of the page encoded in ISO 639-  1. For example, ru for Russian or en for English.  If the parameter is missing or does not match the format, the default value is pageView=DESKTOP. |
| sessionTimeoutSecs | N...9 | no | The order lifetime in seconds.  If the parameter is not specified, the value specified in the merchant's settings or the default time (1200 seconds = 20 minutes) will be used.  If the request contains the expirationDate parameter, the value of the sessionTimeoutSecs parameter is ignored. |
| bindingId | AN..255 | no | ID of the binding created earlier. It can only be used if the store has permission to work with bindings. If this parameter is passed in this request, it means that:   1. This order can only be paid using a binding; 2. The payer will be redirected to the payment page, where only CVC input is required. |
| expirationDate | ANS | no | Date and time of the order lifetime expiration. Format:  yyyy-MM-dd'T'HH:mm:ss.  If this parameter is not present in the request, then sessionTimeoutSecs is used to determine the time of the order lifetime expiration. |
| returnUrl | ANS..512 | yes | The address to which the user must be redirected in case of successful 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>**. |
|  |  |  | See also the **3DS payment** block - for one-phase and two-phase payments. returnUrl is used in the same way as in step 11 in the specified patterns. |
| failUrl | ANS..512 | no | The address to which the user must be redirected in case of 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>**. |
| See also the **3DS payment** block - for one-phase and two-phase payments. failUrl is used in the same way as in step 11 in the specified patterns. |
| dynamicCallbackUrl | Unable to render include or excerpt- include. Could not retrieve page. | no | Unable to render include or excerpt-include. Could not retrieve page. |
| params |  | no | Tag containing attributes to pass additional parameters of the merchant.  The fields for additional information and its subsequent storage. To pass N parameters, the request must contain N params tags, where the name attribute contains the name, and the value attribute contains its value:   |  |  |  |  | | --- | --- | --- | --- | | **Name** | **Type** | **Mandatory** | **Description** | | name | ANS..20 | yes | Name of the additional parameter | | value | ANS..1024 | yes | Value of the additional parameter |   These fields can be passed to the Bank processing system to be subsequently displayed in the registries.\*  This functionality can be enabled upon additional agreement with the bank during the integration period.If sending notifications to a buyer is set up for a seller, the email address of a buyer must be passed in this tag in the email parameter. |
| clientId | AN..255 | no | The client's number (ID) in the store's system. It is used to implement the functionality of bindings. It can be present if creating bindings is allowed for the store. |
| Specifying this parameter when processing payments with the use of bindings is mandatory. Otherwise, a payment will be failed. |
| merchantLogin | AN..255 | no | To register an order on behalf of a child merchant, specify the merchant login in this parameter. |
| features | ANS..255 | no | A container for the feature parameter, where the following values can be passed.  AUTO\_PAYMENT – if the request for order registration initiates auto-payments.  VERIFY – 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. |
| **Specifics of passing the VERIFY value** 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. |
| Example of use: |
| <features>  <feature>AUTO\_PAYMENT</feature>  </features> |
| autocompletionDate | ANS..19 | no | The date and time of autocompletion of the two-phase payment is in the following format. |
| 2017-12-29T13:02:51 |
|  |
| To enable this functionality, contact technical support. |
| email | ANS..40 | no | Buyer's email address. |
| phone | ANS.12 | no | Buyer'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:  • +79998887766;  • 9998887766. |

**\*** By default the following fields are passed to the bank's 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 | The order number in the payment system. It is unique within the system. It is missing if the order registration  failed due to an error detailed in ErrorCode. |
| formUrl | AN..512 | no | URL of the payment form to which the client's browser is to be redirected. It is not returned if the order registration failed due to an error detailed in ErrorCode. |
| errorCode | N3 | no | Error code. |
| errorMessage | AN..512 | no | Error description in the language passed in the language parameter in the request. |

**Error codes (**ErrorCode **field):**

| **Meaning** | **Description** |
| --- | --- |
| 0 | Request processing took place without system errors |
| 1 | Invalid order number |
| 1 | Order with given order number is processed already |
| 3 | Currency is not defined |
| 4 | Empty amount |
| 4 | The order number cannot be empty |
| 4 | The return URL cannot be empty |
| 5 | Payments with pre-authorization are not allowed |
| 5 | The value of one of the parameters is incorrect |
| 5 | Access denied |
| 5 | The user must change password |
| 7 | System error |
| 13 | Using both values, Features FORCE\_TDS/FORCE\_SSL and AUTO\_PAYMENT, is not allowed |
| 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="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<soapenv:Header/>

<soapenv:Body>

<mer:registerOrderPreAuth>

<order merchantOrderNumber="asuaakdfadsfasdfasdd5" description=" " amount="10000" currency=" " language="ru" pageView="DESKTOP" bindingId=" " sessionTimeoutSecs=" ">

<returnUrl>http://yoursite.com</returnUrl>

<params name="param1" value="paramValue1"/>

<params name="param2" value="paramValue2"/>

<clientId>7777</clientId>

<merchantLogin> </merchantLogin>

<features>

<feature>AUTO\_PAYMENT</feature>

</features>

</order>

</mer:registerOrderPreAuth>

</soapenv:Body>

</soapenv:Envelope>

##### Response example:

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<ns1:registerOrderPreAuthResponse xmlns:ns1="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<**return** orderId="5e5dc6bd-dee3-4c96-849a-09f3f575f4b6" errorCode="0" errorMessage="Success">

<formUrl> https://server/application\_context/payment\_ru.html?mdOrder=5e5dc6bd-dee3- 4c96-849a-09f3f575f4b6 </formUrl>

</**return**>

</ns1:registerOrderPreAuthResponse>

</soap:Body>

</soap:Envelope>

* + 1. Request to verify the card's enrolment in 3DS (WS)

The verifyEnrollment request is used to verify the card's involvement in 3DS.

##### Request parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| pan | N..19 | yes | Payment card number. |

##### Response parameters:

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| errorCode | N3 | no | Error code. |
| errorMessage | AN..512 | no | Error description. |
| isEnrolled | A1 | no | A sign of the card's enrolment in 3DS. Possible values: Y, N, U. |
| emitterName | AN..160 | no | Name of the Issuing bank. |
| emitterCountryCode | AN..4 | no | Country code of the issuing bank. |

##### Error Codes (ErrorCode field):

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 1 | The card number is not specified |
| 1 | The card number must be a number containing from 13 to 19 digits |
| 5 | Access denied |
| 5 | The user must change password |
| 6 | No information was found for the specified card number |
| 7 | System error |

**Request example:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mer="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<soapenv:Header/>

<soapenv:Body>

<mer:verifyEnrollment>

<pan>4111111111111111</pan>

</mer:verifyEnrollment>

</soapenv:Body>

</soapenv:Envelope>

##### Response example:

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<ns1:verifyEnrollmentResponse xmlns:ns1="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<**return** isEnrolled="Y" emitterName="TEST CARD" emitterCountryCode="RU" errorCode="0"/>

</ns1:verifyEnrollmentResponse>

</soap:Body>

</soap:Envelope>

* + 1. Order Payment Requests (WS)

6.1.4.1 Request for order payment when using an external MPI (WS)

##### Request parameters:

| **Name** | **Type** | | **Mandatory** | **Description** |
| --- | --- | --- | --- | --- |
| orderId | ANS36 | | yes | Order number received during order registration |
| pan | N..19 | | yes | Payment card number. |
| cvc | | | N..3 | yes | CVC |
| year | | | N..4 | yes | Year |
| month | | | N..2 | yes | Month |
| cardholderName | | | A..512 | yes | Cardholder's name |
| language | | | A..2 | yes | Response language |
| ip | | | ANS..39 | yes | The IP address of the user who paid for the order. |
|  | | |  |  | IPv6 is supported in all requests. |
| email | | | ANS..\* | no | The payer's email address. If the merchant's settings provide for sending notifications to clients about order transactions, then this parameter must be present. |
| params | | |  | no | An additional tag with attributes for transmitting additional merchant parameters \*.  In the case of using an external MPI, IPS expects that the "eci" parameter will be received in each request for order payment. If the "eci" value differs from those used for SSL authorizations, it is also necessary to send the "xid" and "cavv" parameters.  When paying with MIR cards with 3-D Secure support, if ECI=01 or ECI=02, the CAVV value must be transmitted on a mandatory basis.  The authenticationTypeIndicator parameter (payment authentication type) is required when making a payment via an external MPI using the 3DS 2.0 protocol.  If 3DS 1 or SSL is used in the external MPI, the authenticationTypeIndicator parameter is optional and will be determined automatically based on the ECI value.  Possible values are:   |  |  |  | | --- | --- | --- | | **Meaning** | **Decoding** | **Compulsoriness/ automatic detection rules** | | 0 | SSL | ECI = 07 | |  | SSL authentication |  | | 1 | THREE\_DS1\_FULL 3DS 1 Authentication | ECI = 02, 05 | | 2 | THREE\_DS1\_ATTEMPT  3DS 1 authentication attempt | ECI = 01,06 | | 3 | THREE\_DS2\_FULL  Strong authentication (SCA) | Mandatory for 3DS 2.0. | | 4 | THREE DS2 FRICTIONLE  SS  Risk-based authentication (RBA) | Mandatory for 3DS 2.0. | | 5 | THREE\_DS2\_ATTEMPT  Authentication attempt (3DS 2.0) | Mandatory for 3DS 2.0. |   If you want to accept payment by installments, and executed a [get request](#_bookmark75) to obtain information about the available installment options, you should pass the contents of the installmentsProposals block received in response to this request as additional parameters. |

\* The fields for additional information and its subsequent storage. These fields can be passed to the Bank processing system to be subsequently displayed in the registries.

This functionality can be enabled upon additional agreement with the bank during the integration period. To pass N parameters, the request must contain N params tags, where the name attribute contains the name, and the value attribute contains its value.

Also, the assignment of the list of additional parameters includes storing and transmitting to SmartVista the cavv, xid, eci parameters in the case of using an external MPI and having the appropriate permission in the IPS system to use it.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| name | AN..20 | yes | Name of the additional parameter |
| value | AN..1024 | yes | Value of the additional parameter |

##### Response parameters:

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| errorCode | N..1 | yes | Error code |
| errorMessage | AN..512 | no | Error message |
| info | AN..512 | no | The result of the payment attempt. The possible values are shown below:   * Your payment is proceeded, redirecting... * The operation was rejected. Check the entered data, the sufficiency of funds on the card and repeat the operation. Redirecting... * Sorry, the payment cannot be made. Redirecting... * The operation was rejected. Contact the store. Redirecting... * The operation was rejected. Contact the bank that issued the card. Redirecting... * The operation is impossible. Cardholder authentication failed. Redirecting... * No connection with the bank. Repeat later. Redirecting... * Data entry timeout. Redirecting... * No response was received from the bank. Repeat later. Redirecting... |

If **errorCode** = 0 and **info** = Your order is processed, redirecting, the payment was successful. In other cases with an error, see the **errorMessage**.

Error Codes (ErrorCode field):

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 5 | Access denied |
| 5 | The user must change password |
| 5 | Payment attempts have been exhausted or the session time has ended |
| 5 | Data validation error |
| 7 | System error |

##### Request example:

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mer="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<soapenv:Header>

<soapenv:Body>

<mer:paymentOrder>

<order orderId="f0c87e00-7b26-4573-942f-3d40596a6d43" year="2015" month="12" pan="4111111111111111" cvc="123" cardholderName="A Moriarty" language="ru" ip="?">

<!-Zero or more repetitions:->

<params name="xid" value="MDAwMDAwMDEzMzkyMjg5ODExNTc="/>

<params name="eci" value="05"/>

<params name="cavv" value="AAABCpEChRM5IomAKFAAAAAAAAA="/>

<params name="authenticationTypeIndicator" value="1"/>

</order>

</mer:paymentOrder>

</soapenv:Body>

</soapenv:Envelope>

##### Response example:

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<ns1:paymentOrderResponse xmlns:ns1="https://tws.egopay.ru/ab/webservices/merchant">

<**return** errorCode="0" info="Your payment is proceeded, redirecting..."/>

</ns1:paymentOrderResponse>

</soap:Body>

</soap:Envelope>

6.1.4.2 Request for order payment when using an internal MPI (WS)

##### Request parameters:

| **Name** | | **Type** | | **Mandatory** | | **Description** | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| orderId | | ANS36 | | yes | | Order number received during order registration | |
| pan | | N..19 | | yes | | Payment card number. | |
| cvc | | N..3 | | yes | | CVC | |
| year | | N..4 | | yes | | Year | |
| month | | N..2 | | yes | | Month | |
| cardholderName | | A..512 | | yes | | Cardholder | |
| language | | A..2 | | yes | | Response language | |
| ip | | ANS..39 | | yes | | The IP address of the user who paid for the order. | |
|  | |  | |  | | IPv6 is supported in all requests. | |
| email | | ANS..\* | | no | | The payer's email address. If the merchant's settings provide for sending notifications to clients about order transactions, then this parameter must be present. | |
| params | |  | | no | | An additional tag with attributes for transmitting additional merchant parameters \*.  If you want to accept payment by installments, and executed a [get request](#_bookmark75) to obtain information about the available installment options, you should pass the contents of the installmentsProposals block received in response to this request as additional parameters. | |

\* The fields for additional information and its subsequent storage. These fields can be passed to the Bank processing system to be subsequently displayed in the registries.\*\*

This functionality can be enabled upon additional agreement with the bank during the integration period. To pass N parameters, the request must contain N Params tags, where the name attribute contains the name, and the value attribute contains the value:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| name | AN..20 | yes | Name of the additional parameter |
| value | AN..1024 | yes | Value of the additional parameter |

\*\* *By default, the orderNumber and its description fields are passed to the bank's processing system (no more than 99 characters; %, +, carriage return \r, and line feed \n cannot be used)*

##### Response parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| errorCode | N..1 | yes | Error code |
| errorMessage | AN..512 | no | Error message |
| info | AN..512 | no | The result of the payment attempt. The possible values are shown below:   * Your payment is proceeded, redirecting... * The operation was rejected. Check the entered data, the sufficiency of funds on the card and repeat the operation. Redirecting... * Sorry, the payment cannot be made. Redirecting... * The operation was rejected. Contact the store. Redirecting... * The operation was rejected. Contact the bank that issued the card. Redirecting... * The operation is impossible. Cardholder authentication failed. Redirecting... * No connection with the bank. Repeat later. Redirecting... * Data entry timeout. Redirecting... * No response was received from the bank. Repeat later. Redirecting... |
| redirect | AN..512 | no | User redirection address |
| acsUrl | AN..512 | no | It is not used for payments that do not require additional authentication through the Issuing Bank's ACS. |
| paReq | AN..512 | no | It is not used for payments that do not require additional authentication through the Issuing Bank's ACS. |

Error Codes (ErrorCode field):

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 5 | Access denied |
| 5 | The user must change password |
| 5 | Payment attempts have been exhausted or the session time has ended |
| 5 | Data validation error |
| 7 | System error |

##### Request example:

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mer="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<soapenv:Header>

<soapenv:Body>

<mer:paymentOrder>

<order orderId="09808814-eb60-4cea-86bb-ed849dbe7ae9" year="2015" month="12" pan="4111111111111111" cvc="123" cardholderName="A Moriarty" language="ru" ip="?">

<!-Zero or more repetitions:->

<params name="name" value="value"/>

</order>

</mer:paymentOrder>

</soapenv:Body>

</soapenv:Envelope>

##### Example of a response in the case of a payment that does not require additional authentication through the Issuing Bank's ACS:

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<ns1:paymentOrderResponse xmlns:ns1="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<**return** info="Ваш платёж обработан, происходит переадресация..." redirect="www.ya.ru?MDORDER=6d295b5b-84c3-44cd-adea- c0eaca122465&amp;ANSWER=%3C%3Fxml+version%3D%221.0%22+encoding%3D%22UTF- 8%22%3F%3E%0A%3CPSApiResult+primaryRC%3D%220%22+secondaryRC%3D%220%22%2F%3 E&amp;STATE=payment\_deposited&amp;ACTION\_CODE=0&amp;AUTH\_CODE=2" errorCode="0"/>

</ns1:paymentOrderResponse>

</soap:Body>

</soap:Envelope>

##### Example of a response in the case of a 3DS payment that requires additional authentication through the Issuing Bank's ACS:

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<ns1:paymentOrderResponse xmlns:ns1="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<**return** errorCode="0" info="Your payment is proceeded, redirecting..." redirect="https://tws.egopay.ru:443/ab/finish3ds.do" acsUrl="https://tws.egopay.ru/acs/auth/start.do" paReq="eJxVUk1zgjAQ/SsM95KEr1pnjUOLnXqgYxUvvVHYAVQ+DFDUX99EUeshM/t2N2/3vQS mh2Kn/aJo8qqc6MyguoZlXCV5mU70dfj+NNKnHMJMIPorjDuBHAJsmihFLU8metGkBtM5LLwl7 jkMTFwSGSaQK5RXRJxFZcshivev809uWzazHCADhALF3OfMtGzHfR4BuWAoowJ5iE27yqoayBl CXHVlK47ctS0gVwCd2PGsbesxIX3fG2lVpTs04qoAokpA7jssOhU1kuqQJzzwvf5yZqdPf0uDc HsM/C8WnNIJENUBSdQiNykzGaWOxujYpmNqAznnISrUDtx1XUqloguCWg3xHkr/UyC9FNLqq4w rAjzUVYmyQ/p3iyHBJr4ZodWDALmBygO5K3r7UB7HrXRvnXxvhdufxGKUdH34kiUb15mZ3k+3W Srnz01qXi79Yw67DFQAiKIhw6OS4cFl9PAR/gAOWr9V"/>

</ns1:paymentOrderResponse>

</soap:Body>

</soap:Envelope>

##### If the payment requires the use of 3-D Secure technology, then after receiving a response to the payment request, the client must be redirected to ACS. There are two ways of redirection: normal and simplified (see the table below).

|  |  |
| --- | --- |
| **Redirection method** | **Description** |
| Normal | To redirect to ACS, the merchant, for its part, must send the client to the address specified in the acsUrl parameter with the request body MD=mdorder&PAReq=pareq&TermUrl=redirect, where:   * mdorder - unique order number in the payment gateway system, * pareq - parameter from the response to the payment request, * redirect - parameter from the response to the payment request (the address of return from ACS) or the store's address, if the pattern of return from ACS to the store is used.   The request must be in POST format.  Depending on the connection pattern used, the client will be redirected to the payment gateway or to the store after ACS authentication. The payment completion processes for each of these cases are described below. |
| Simplified | In order for the client to get to the ACS page, the seller redirects them to the payment gateway page of the following type:  https://tws.egopay.ru/ab/acsRedirect.do?orderId=<order number>  where: <order number> is the client's unique order number.  Further, without additional actions on the part of the client, the payment gateway redirects the client to the Issuing bank's ACS page, where the client is authenticated. |

* + 1. Finish3DS request (WS)

The issuer's ACS authenticates the cardholder and redirects the client back to the store. The PARes received from ACS is transmitted to the store, after which the store transmits it to the gateway using the Finish3DS method.

The finish3DS method is described below:

##### Request parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| md | AN..30 | yes | Order number received during order registration |
| paRes |  | yes | Payer Authentication Response – response to payer identification |

##### Response parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| returnUrl | AN..512 | yes | URL of the payment form to which the client's browser is to be redirected. |
| errorCode | N3 | no | Error code. It may be missing if the result did not lead to an error. |
| errorMessage | AN..512 | no | Error description in the language passed in the Language parameter in the request. |

If errorCode = 0 and info ="Your order is processed, redirecting", the payment was successful. In other cases with an error, see the errorMessage.

**Error codes (**ErrorCode **field):**

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 5 | Access denied |
| 5 | The user must change password |
| 5 | 3DS operation completion error |
| 7 | Order not found |
| 7 | System error |

##### Request example:

<soapenv:Envelope [xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"](http://schemas.xmlsoap.org/soap/envelope/) xmlns:mer="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<soapenv:Header/>

<soapenv:Body>

<mer:finishThreeDs>

<request language="ru" md="8b5b7ee5-eb5a-4cf4-81ec- 7153f7ca2864"

paRes="eJzFV2mzokoS/Ssd/T4at1kEkRe0EcW+WCCLgH5jBwVRWfXXD2r37ft6eiZ6ZuLFEG FYnKjMypOZp4pinPyaJLydRN01WTEwaZogSz4V8dfPi/k8JKg5+kahYfBGYPjyLcRD+i2g8GW EzRdBvEA/r5gNsJLmZZAuyZBOF28EkU4GwWQaYAk6vS5CFMXxJUlGk0GfXJuiPq2wL+gXnEG+ v05rX6M8OLUrJogurKKvCJxaoCiDfHtlquSq8CuKQV4DBvlhsekeo2ZiMBbxCvJgeP0EVOeFu cFnI7xnN3iAXxnkMYOJgzZZ4ShGojRKf8KwPwn8T2zJIE+cOT/cgaruJt8kSVJLmkE+YsyUrW tyim6rBTFnkPc3JhnP9SmZZkzE3scM8iO8c3BaoR8ebHom3xPKOP6KaYvql2E9caZpg7ZrVjs G+TZioqDvVwAAlp3HxN6y2ESR/LwE35+J7nMKk0TFCiWnoKb/pxUos/patHn1CPWvAIM8QkGe hV0xdpGdpsWuyaexKk/N1895257/RJBhGL4M8y/1NUPwiQiC0sg0IW6K7I/PL6skVk5p/R+Zc cGpPhVRUBb3oJ26AiZtXsef3mP7lRvHenjCEEvg3iZXbxFGnN4eCDrHyMkn8munH5j9zio/B3 ttgrcmD7DHAj85WjFWkiaPjkg+bS3l6+c/flMZfJElTfvfRPM9ko8evvtzg7JLVuNlPjOtnlV wfrbnbntQ7I+7AtOSQfn63e41k0Hew//G7VXIDwl7TfT1ejePNAdVXfkIMMqH7mYGy9QkoRea 8wW61M+L1ElSyOUdhpU05W4jqZPbaGjuylHf5GicEpcYW/MiskkxU96WNYClI6CaFt1TqSuhc c55kyRpTwjnO1gISSgKYVaLqhTPhO5uRL2qXdYQva6F5f46nkA+HrDRxjxiUft+3Sv7NVaLy6 8fivSNpZbcXqz8SWx80AavEZdc2yKdumXaB6CicLXDcSC7ZGBQWJApKticRPue7IpZGUKASpx 9kWwlnPOmwLLmFkABjNwdqGymuyzYOaB0HWiBgc92vGuamjDE/M5X870k3qE5DNwLXwuDbtqu OQoO2LxsocPJ7DnmsFuI0yhkCZ+fMgN5OOgHgML7doRiPWHKE4P3d2wweWENwVEC2FZgc8i5L hwFHhgvv7XDonoeVeJx58OR54H2whsHYKzq8oIFwfJpC0aoRFKJrn21D+fmkO2XA2/uVK3eK3 kf6eDJGUzUUAAVSQW1xAJNouIiuPJmhlHbQ9WrQSOL9sHLN3FII/TZ3/ce0LqIHi6XAz8/9dv FMOvUdqeIF53ofW52OFLJkXBF6tAfeXFuoWh6ksvFYLhdejdLEVAz0z97DpSacHG7DbkPBzeM jGWHEZHkIuOMdJPh6GRX1wov+9LwlSopS733sLRRSbmw3VReJrcSZJAFQDpkgfLkJptTLSzUY NmdIOrH+TynTYyV0qKmulHTE7zcxZjcqVAGj/zEymCdoZSAfze33tpH8zBV02wGznzWWhIG1d 3eBRMC4plnLoeyiYu3nWfl0V2AcErkEx8hvxXKLbSEgR+etrww5EPgm21U0X2s/NyPwzMuYTB FCCZu6T/VS3zVSwBKst+3vnkpWQ0lAjwrKulYI8Ea33nQynKUuDna7DBQxlFFtl4MsIxabtUj iVwdlcZJii5q8tpUa9l3Z50XLbzrQhENAqRjYfYzyUM20FyeO4Kul0lP35x+F9TN0t8Q93hrE FbjswN93VmdvSGMHm8NTdvUlaGug9Amp+4rubo7UhSBjI4wifdnZf5Sqqf7Q6rFD6lOXS4hSi IPLf3/lKrO7+469lepfsOGzeF/kurmweubVKWnVD09n6SY/swXTHynbtMBz7GFObkxeXs5HDt xKxLHICgCez2zyIUbHO9YN508O/GCCLWv3tf2gII28JZYIndcfkuWlJ/x0sbvgMDVoWZh55o6 VFVP6Vh1qcsxXrctqElNuXp0YuiLGTKb20SrHDvtepkt3LVzG22yrSh5wuaunBlBt7vswRBzo y1elE48NN0hb2a+P/YD29VkMyg8MAFbE1Lx5MY+Ocd8Znosa9uJgYBzHyPKblPnIJMxJRZs/Y xoPDs88iXbUAgc9pSxzl6niXSrm9aJuBhUYx/89uzJsUUVgSicBeVv2jptjjN5ctw280rpz95 90uYjLgsKrAMmbjLyq/6c6sUCrvVZtG9LdGZut70slfUyu3DD4ZAP/Poc4K7VlEHXaFK7JLxy U/ioI+PsZayQe7W5sQf91PmeJFcmoWVidQlQOxExkG6Q3Rx4iaCChVonB3J5ktD8uuAiIxyv3 GwA0NrwSrmWjRmJmbfDWRdpmZsd8UU4boPzvsFdD8HP0WZBVa0p3/xiZ/6eTFnyIVOX+y7T7f Z61zjaUqieygI1dehY48wt5w/75le7198jmZdksWm/HeEBEJNEb/ApWQF/Ys47Ngi8YHzbaR8

+RPdYOqYDf/N03F76/MDVtAphvNHB0FTyItyYRRUM4amouLm9GNtzFkb8/hanqHK7pTdIC2l3 GncU7WEIV2eeye1Do2yI4JCxjXZc70o5Lo1aUo+EROxbMVibdyMcJHLcgNCi7rsk2MeVpQWUV uKqOrZ6ZZDO7OaPSMu3i5zTT+s8a0aaGCipQFhRMQr/++n4yO2Pk0VgkcHkIAD/8mSBd7RPp6 ovAtea407QOq4jqKejJ1b5FVPNppD3FiFnKXaXziiYB3mejm2na5q0pQrphEVwvNcnhRaoI8n Z2tq1D0VpuMt83h3PzgzX8hbvacINCxbZnLPM8DrRh2CnD2k6coG4KQ+zPdbJTlljBrsVGyXa 5bgx6woxZzfa2WvndjXG8sL81cmC/PgeRN6/EX98PT6vnc9L8uOe9PHy/A9JE/bD"/>

</mer:finishThreeDs>

</soapenv:Body>

</soapenv:Envelope>

##### Response example:

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<ns1:finishThreeDsResponse xmlns:ns1="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<**return** errorCode="0" returnUrl="http://ya.ru?orderId=8b5b7ee5- eb5a-4cf4-81ec-7153f7ca2864"/>

</ns1:finishThreeDsResponse>

</soap:Body>

</soap:Envelope>

* + 1. Order Status Request (WS)

The getOrderStatus request is used to request the status of a registered order.

The order status must be determined by the value of the OrderStatus parameter. The AuthCode field is deprecated.

##### Request parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| orderId | ANS36 | yes | The order number in the payment system. It is unique within the system. |
| language | A2 | no | The language is encoded in ISO 639-1. If not specified, it is assumed that the language is Russian. The error message will be returned in this language. |

##### Response parameters:

| **Name** | | **Type** | | **Mandatory** | | **Description** | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| orderStatus | | N2 | | no | | The value of this parameter determines the status of the order in the payment system.  The list of possible values is given in the table below. It is missing if the order has not been found. | |
| errorCode | | N3 | | no | | Error code. | |
| errorMessage | | AN..512 | | no | | Error description in the language passed in the Language parameter in the request. | |
| orderNumber | | AN..32 | | yes | | Order number (ID) in the store's system | |
| pan | | N..19 | | no | | The masked number of the card that was used for payment. It is specified only after payment of the order. | |
| expiration | | N6 | | no | | Card expiration date in the YYYYMM format. It is specified only after payment of the order. | |
| cardholderName | | A..64 | | no | | Cardholder's name. It is specified only after payment of the order. | |
| amount | | N..20 | | yes | | Payment amount in kopecks (or cents) | |
| currency | | N3 | | no | | The payment currency code is ISO 4217. If not specified, the default value is used. | |
| approvalCode | | AN6 | | no | | IPS authorization code. This field has a fixed length (6 characters) and can contain digits and Latin letters. | |
| authCode | | N3 | | no | | This field is deprecated. Its value is always "2", regardless of the order status and the authorization code of the processing system. | |
| ip | | ANS..39 | | no | | The IP address of the user who paid for the order. | |
| IPv6 is supported in all requests. | |
| date | | ANS | | yes | | Order registration date | |
| orderDesctiption | | AN..512 | | yes | | Description of the order transmitted during its registration | |
| actionCodeDesctiption | | AN..512 | | yes | | Decryption of the response code in the language transmitted in the Language parameter in the request. | |
| clientId | | AN..255 | | no | | The client's number (ID) in the store's system, transmitted during the order registration. Present only if the store is allowed to create bindings. | |
| bindingId | | AN..255 | | no | | ID of the binding created when paying for the order or used for payment. Present only if the store is allowed to create bindings. | |

**The** orderStatus **field can have the following values:**

| **Status number** | **Description** |
| --- | --- |
| 0 | The order is registered, but not paid for |
| 1 | The pre-authorized amount is reserved (for two-phase payments) |
| 2 | Full authorization of the order amount has been carried out |
| 3 | Authorization canceled |
| 4 | A refund operation was performed on the transaction |
| 5 | Authorization initiated through the issuing bank's ACS |
| 6 | Authorization denied |

**Error codes (**ErrorCode **field):**

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 2 | Payment rejected |
| 5 | Access denied |
| 5 | The user must change password |
| 5 | [orderID] not specified |
| 6 | Unknown order number |
| 7 | System error |

**Request example:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mer="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<soapenv:Header/>

<soapenv:Body>

<mer:getOrderStatus>

<order orderId="b1221b79-5703-42c9-a4b1-ed0d0f36493e" language="ru"/>

</mer:getOrderStatus>

</soapenv:Body>

</soapenv:Envelope>

##### Response example:

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<ns1:getOrderStatusResponse xmlns:ns1="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<**return** orderStatus="2" errorCode="0" orderNumber="456789012345678" pan="411111\*\*1111" expiration="201512" cardholderName="Ivan" amount="15000" currency="810" approvalCode="123456" uthCode="2" ip="212.5.125.194" date="2013-09-03T16:48:52.459+04:00"

orderDescription=" " actionCodeDescription="Request successfully processed"

clientId="666" bindingId="df0b1801-a754-4576-b174-c2485d98bc9b"/>

</ns1:getOrderStatusResponse>

</soap:Body>

</soap:Envelope>

* + 1. Advanced Order Status Request (WS)

The getOrderStatusExtended request is used to request the status of a registered order.

##### Request parameters:

| **Name** | | **Type** | | **Mandatory** | | **Description** | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| orderId | | ANS36 | | yes\* | | The order number in the payment system. It is unique within the system | |
| language | | A2 | | no | | The language is encoded in ISO 639-1. If not specified, it is assumed that the language is Russian. The error message will be returned in this language. | |
| merchantOrderNumber | | ANS..32 | | yes\* | | The order number (ID) in the store's system. | |

\* The request must pass either the orderId parameter or the merchantOrderNumber parameter. If both parameters are passed in the request, orderId has a higher priority.

There are several sets of response parameters. Exactly what kind of sets of parameters will be returned depends on the version of getOrderStatusExtended specified in the seller's settings.

| **Name** | **Type** | | **Mandatory** | | **Description** | | **getOrderStatusExtended version** | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| orderNumber | AN..32 | | yes | | The order number (ID) in the store's system. | | All versions. | |
| orderStatus | N2 | | no | | The value of this parameter determines the order status in the payment system. A list of possible values is given in the list below.  It is missing if the order has not been found.   * **0** - The order is registered, but not paid for; * **1** - The pre-authorized amount is reserved (for two-phase payments); * **2** - A full authorization of the order amount has been carried out; * **3** - Authorization canceled; * **4** - A refund operation was performed on the transaction; * **5** - An authorization has been initiated through the issuing bank's ACS; * **6** - Authorization denied. | | All versions. | |
| actionCode | N3 | | yes | | Response code. | | All versions. | |
| actionCodeDescription | AN..512 | | yes | | Decryption of the response code in the language transmitted in the Language parameter in the request. | | All versions. | |
| errorCode | N3 | | no | | Error code. The following options are possible.   * **0** - Request processing took place without system errors; * **1** - [orderId] or [orderNumber] expected; * **5** - Access denied; * **5** - The user must change password; * **6** - Order not found; * **7** - System error. | | All versions. | |
| errorMessage | AN..512 | | no | | Error description in the language passed in the Language parameter in the request. | | All versions. | |
| amount | N..20 | | yes | | Payment amount in kopecks (or cents) | | All versions. | |
| currency | N3 | | no | | The payment currency code is ISO 4217. If not specified, it is considered equal to 810 (Russian rubles). | | All versions. | |
| date | ANS | | yes | | Order registration date. | | All versions. | |
| depositedDate | N | | no | | The date of the order payment in UNIX-time format (POSIX-time). | | All versions. | |
| orderDescription | AN..512 | | no | | Description of the order transmitted during its registration | | All versions. | |
| ip | ANS.. 39 | | yes | | The IP address of the user who paid for the order. | | All versions. | |
| IPv6 is supported in all requests. | |
| authRefNum | AN..24 | | no | | The payment authorization account number that is assigned when registering the payment. | | 02 and above. | |
| refundedDate | ANS | | no | | Refund date and time. | | 13 and above. | |
| paymentWay | AS..14 | | yes | | The method of making the payment (payment with the input of card data, payment by bindings, etc.).  The payment method is passed in this parameter. It can have the following values:   * CARD – payment with card data entry; * CARD\_BINDING – payment by binding; * CARD\_MOTO – payment via a call center; * CARD\_PRESENT – payment as cardPresent; * ALFA\_ALFACLICK – for payment using Alfa-Click (via the PayByClik system); * UPOP – for payment via the UPOP system, available for China UnionPay cardholders; * FILE\_BINDING – payment via file; * SMS\_BINDING – payment via sms; * P2P – transfer from card to card; * P2P\_BINDING – transfer by binding; * PAYPAL – payment from a PayPal account; * MTS – payment from an MTS account; * APPLE\_PAY – Apple Pay; * APPLE\_PAY\_BINDING – Apple Pay binding payment; * ANDROID\_PAY – Android Pay; * ANDROID\_PAY\_BINDING – payment with a binding of Android Pay; * GOOGLE\_PAY\_CARD – Google Pay non-tokenized; * GOOGLE\_PAY\_CARD\_BINDING – payment with a binding of a non-tokenized GooglePay card; * GOOGLE\_PAY\_TOKENIZED – Google Pay tokenized; * GOOGLE\_PAY\_TOKENIZED\_BINDING – payment with a binding of a tokenized GooglePay card; * SAMSUNG\_PAY – Samsung Pay; * SAMSUNG\_PAY\_BINDING – payment with a Samsung Pay binding; * IPOS – iPOs payment. | | 09 and above. | |
| avsCode | A1 | | no | | AVS Response Code – the code of the AVS verification response (checking the address and postal code of the cardholder). Possible values are:   * A – postal code and address match; * B – address matches, postal code does not match; * C – postal code matches, address does not match; * D – postal code and address do not match; * E – data validation is requested, but the result is unsuccessful; * F – incorrect format of the AVS/AVV verification request. | | 19 and above | |
| *The* merchantOrderParams *element is present in the response if the order contains additional parameters of the seller. Each additional order parameter is presented in a separate* merchantOrderParams *element.* | | | | | | | | |
| name | AN..20 | | no | | Name of the additional parameter | | All versions. | |
| value | AN..1024 | | no | | Value of the additional parameter | | All versions. | |
| *The* cardAuthInfo *element contains a structure consisting of a list of the* secureAuthInfo *element and the following parameters:* | | | | | | | | |
| pan | N..19 | | no | | The masked number of the card that was used for payment. It is specified only after payment of the order. | | All versions. | |
| expiration | N6 | | no | | Card expiration date in the YYYYMM format. It is specified only after payment of the order. | | All versions. | |
| cardholderName | A..64 | | no | | Cardholder's name. It is specified only after payment of the order. | | All versions. | |
| approvalCode | AN6 | | no | | Payment authorization code. This field has a fixed length (6 characters) and can contain digits and Latin letters. It is specified only after payment of the order. | | All versions. | |
| chargeback | A..5 | | no | | Whether the funds were forcibly returned to the buyer by the bank. The following values are possible.   * **true** ; * **false**. | | 06 and above. | |
| paymentSystem | N..10 | | yes | | Payment system name. The following options are available. VISA;MASTERCARD;AMEX;  * **JCB**; * **CUP**; * **MIR**. | | 08 and above. | |
| product | AN..255 | | yes | | Additional information about corporate cards. This information is filled in by the technical support in the management console. If there is no such information, an empty value is returned. | | 08 and above. | |
| productCategory | string | | yes | | Additional information about the category of corporate cards. This information is filled in by the technical support in the management console. If there is no such information, an empty value is returned.  Possible values: DEBIT, CREDIT, PREPAID, NON\_MASTERCARD, CHARGE, DIFFERED\_DEBIT. | | 17 | |
| *The* secureAuthInfo *element (the element consists of an* eci *element and a* threeDSInfo *element, which is a list of the* cavv *and* xid *parameters):* | | | | | | | | |
| eci | N..4 | | no | | Electronic commercial indicator. It is specified only after the order payment and in case of appropriate permission. | | All versions. | |
| cavv | ANS..200 | | no | | The value of checking the authentication of the cardholder. It is specified only after the order payment and in case of appropriate permission. | | All versions. | |
| xid | ANS..80 | | no | | Electronic commercial transaction ID. It is specified only after the order payment and in case of appropriate permission. | | All versions. | |
| *The* bindingInfo *element consists of the following parameters:* | | | | | | | | |
| clientId | AN..255 | | no | | The client's number (ID) in the store's system, transmitted during the order registration. Present only if the store is allowed to create bindings. | | All versions. | |
| bindingId | AN..255 | | no | | ID of the binding created when paying for the order or used for payment. Present only if the store is allowed to create bindings. | | All versions. | |
| authDateTime | ANS | | no | | Date/time of authorization. | | 02 and above. | |
| terminalId | | AN..10 | | no | | Terminal ID. | | 02 and above. | |
| *The* paymentAmountInfo *element consists of the following parameters:* | | | | | | | | | |
| approvedAmount | | N..20 | | no | | The amount reserved on the card (used only for two-phase payments). | | 03 and above. | |
| depositedAmount | | N..20 | | no | | The amount confirmed for debiting from the card. | | 03 and above. | |
| refundedAmount | | N..20 | | no | | Refund amount. | | 03 and above. | |
| paymentState | | A..10 | | no | | Order status. | | 03 and above. | |
| feeAmount | | N..20 | | no | | Fee amount. | | 11 and above. | |
| totalAmount | | N..20 | | no | | Order amount + fee (fee if it was used in the order). | | 18 and above. | |
| *The* bankInfo *element consists of the following parameters:* | | | | | | | | | |
| bankName | | AN..200 | | no | | Name of the Issuing bank. | | 03 and above. | |
| bankCountryCode | | AN..4 | | no | | Country code of the issuing bank. | | 03 and above. | |
| bankCountryName | | AN..160 | | no | | Name of the issuing bank's country in the language passed in the language parameter in the request, or in the language of the user who called the method, if the language is not specified in the request. | | 03 and above. | |

**Request example:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mer="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<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="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<**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="Success">

<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>

* + 1. Request for Order Payment Completion (WS)

The depositOrder request is used to request the debiting of a pre-authorized order. This operation can be performed if there are appropriate rights in the system.

##### Request parameters:

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| orderId | ANS36 | yes | The order number in the payment system. It is unique within the system. |
| depositAmount | N..12 | yes | The amount of the debit in the order currency. It may be less than or equal to the amount of pre-authorization. It cannot be less than 1 ruble. |
| language | A2 | no | The language is encoded in ISO 639-1. If not specified, it is assumed that the language is Russian. The error message will be returned in this language. |

##### Response parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| errorCode | N3 | no | Error code. |
| errorMessage | AN..512 | no | Error description in the language passed in the Language parameter in the request. |

**Attention!!!** If the "amount" parameter is not specified, the completion will occur for the entire pre-authorized amount.

Error Codes (ErrorCode field):

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 5 | The deposit amount exceeds the reserved amount |
| 5 | The deposit amount must be greater than zero or at least one ruble |
| 5 | Access denied |
| 5 | The user must change password |
| 6 | Invalid order number |
| 6 | [orderId] not set |
| 7 | The payment must be in the correct state |
| 7 | System error |

**Request example:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mer="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<soapenv:Header/>

<soapenv:Body>

<mer:depositOrder>

<order language="ru" orderId="4302d369-a5e8-4432-a5e5- 42acfab52c86" depositAmount="30000">

<!- Zero or more repetitions: ->

<params name="?" value="?"/>

</order>

</mer:depositOrder>

</soapenv:Body>

</soapenv:Envelope>

##### Response example:

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<ns1:depositOrderResponse xmlns:ns1="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<**return** errorCode="6" errorMessage="Invalid order number"/>

</ns1:depositOrderResponse>

</soap:Body>

</soap:Envelope>

* + 1. Request for order payment cancellation (WS)

The ReverseOrder request is used to request cancellation of an order. The cancellation function is available within a limited time after payment, the exact terms must be specified in the Bank.

The payment cancellation operation can be performed only once. If it ends with an error, the repeated payment cancellation operation will fail.

This function is available to stores in agreement with the Bank. To perform the cancellation operation, the user must have the appropriate rights.

##### Request parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| orderId | ANS36 | yes | The order number in the payment system. It is unique within the system. |
| language | A2 | no | The language is encoded in ISO 639-1. If not specified, it is assumed that the language is Russian. The error message will be returned in this language. |

##### Response parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| errorCode | N3 | no | Error code. |
| errorMessage | AN..512 | no | Error description in the language passed in the Language parameter in the request. |

Error Codes (ErrorCode field):

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 5 | Access denied |
| 5 | The user must change password |
| 5 | [orderId] not set |
| 6 | Invalid order number |
| 7 | Invalid operation for the current order status |
| 7 | System error |

**Request example:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mer="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<soapenv:Header/>

<soapenv:Body>

<mer:reverseOrder>

<order language="ru" orderId="f88a2bbf-2021-4ccc-8783- 8a13068a89f9">

<!- Zero or more repetitions: ->

<params name=" " value=" "/>

</order>

</mer:reverseOrder>

</soapenv:Body>

</soapenv:Envelope>

##### Response example:

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<ns1:reverseOrderResponse xmlns:ns1="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<**return** errorCode="7" errorMessage="Invalid operation for the current order status"/>

</ns1:reverseOrderResponse>

</soap:Body>

</soap:Envelope>

* + 1. Request for a refund of the order payment (WS)

The refundOrder request is used for the refund.

Upon this request, the funds for the specified order will be returned to the payer. The request will end with an error if the funds for this order have not been debited. The system enables to return funds more than once, but for a total amount not exceeding the initial debit amount.

|  |
| --- |
| When processing a refund for housing and utilities services, only a full refund is available. |

To process the refund operation, one must have the appropriate rights in the system.

##### Request parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| orderId | ANS36 | yes | The order number in the payment system. It is unique within the system. |
| refundAmount | N..12 | yes | The refund amount in the order currency. May be less than or equal to the remnant in the order. |
| language | A2 | no | The language is encoded in ISO 639-1. If not specified, it is assumed that the language is Russian. The error message will be returned in this language. |

##### Response parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| errorCode | N3 | no | Error code. |
| errorMessage | AN..512 | no | Error description in the language passed in the Language parameter in the request. |

Error Codes (ErrorCode field):

|  |  |  |  |
| --- | --- | --- | --- |
| **Meaning** | | **Description** | |
| 0 | | Request processing took place without system errors | |
| 5 | | Access denied | |
| 5 | | The user must change password | |
| 5 | | [orderId] not set | |
| 5 | | Incorrect amount | |
| 6 | | Invalid order number | |
| 7 | | The payment must be in the correct state | |
| 7 | | The refund amount exceeds the amount debited | |
| 7 | | System error | |

**Request example:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mer="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<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="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<**return** errorCode="7" errorMessage="The refund amount exceeds the amount debited"/>

</ns1:refundOrderResponse>

</soap:Body>

</soap:Envelope>

* + 1. Card Verification Request (WS)

To make a card verification request, the verifyCard method is used.

The method performs verify operations. If the verify operation is successful, the order will be transferred to the CANCELED status.

##### Request parameters:

| **Name** | | **Type** | | **Mandatory** | **Description** | |
| --- | --- | --- | --- | --- | --- | --- |
| userName | | AN..30 | | no (either a login and password, or a token needs to be specified) | The store's login received when connecting. If an open token (token parameter) is used instead of login and password authentication, the userName parameter does not need to be passed. | |
| password | | AN..30 | | no (either a login and password, or a token needs to be specified) | | The store's password received when connecting. If an open token (token parameter) is used for authentication during registration instead of a login and password, the password parameter does not need to be passed. |
| token | | ANS..30 | | no (either a login and password, or a token needs to be specified) | | A public key that can be used to register an order. If login and password are used for authentication during order registration, the token parameter does not need to be passed. |
| amount | | N…3 | | Yes | | The amount of the debit in the order currency. Maximum value for card verification: 100 |
| currency | | N..3 | | No | | The payment currency code is ISO 4217. If not specified, the default value is used. |
| pan | | N12…19 | | Yes | | Card number. |
| cvc | | N...3 | | Yes | | The CVC/CVV2 code is on the back of the card. |
| expiry | | N6 | | Yes | | The year and month of expiration of the card in the following format:  YYYYMM. |
| cardholderName | | A..200 | | No | | Cardholder's name. It is specified only after payment of the order. |
| backUrl | | AN..512 | | Yes | | The address to which the user must be redirected in case of successful payment. The address must be specified in full, including the protocol used (for example, [https://test.ru](https://test.ru/) instead of [test.ru](http://test.ru/)). Otherwise, the user will be redirected to the address of the following type: http://<payment\_gateway\_address>/<merchant\_address> |
| failUrl | | AN…512 | | No | | The address to which the user must be redirected in case of failed payment. The address must be specified in full, including the protocol used (for example, [**https://test.ru**](https://test.ru/) instead of [**test.ru**](http://test.ru/)). Otherwise, the user will be redirected to the address of the following  type: **http://<payment\_gateway\_address>/<merchant\_address>**. |
| description | | ANS…598 | | No | | Description of the order in any form. |
| email | | ANS..40 | | No | | Buyer's email address. |
| language | | A2 | | No | | The language is encoded in ISO 639-1. If not specified, the default language specified in the store settings will be used. |
| threeDSServerTransId | | ANS..36 | | No | | Transaction ID created on the 3DS server. |
| This parameter is used when authenticating a client using the 3DS protocol of 2.0 version. |
| threeDSVer2FinishUrl | | AN..512 | | No | | Client forwarding address after authentication to the ACS server. |
| This parameter is used when authenticating a client using the 3DS protocol of 2.0 version. |
| This parameter should be passed to the gateway both in the first payment/transfer request and in the second one. |
| threeDSVer2MdOrder | |  | | No | | The order number that was registered in the first part of the request as part of the 3DS 2.0 operation. |

##### Response parameters:

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| errorCode | N..1 | Yes | Error code. |
| errorMessage | AN..512 | No | Error description in the language passed in the language parameter in the request. |
| orderId | ANS36 | No | The order number in the payment system. It is unique within the system.  It is missing if the order registration failed due to an error detailed in errorCode. |
| orderNumber | ANS..36 | No | The order number (ID) in the store's system. |
| authCode | AN6 | No | Payment authorization code, can contain digits and Latin letters. |
| actionCode | N..5 | No | Response codes are a digital designation of the result that the user's access to the system led to. |
| actionCodeDescription | AN..512 | No | Decryption of the response code in the language passed in the language request parameter. |
| time |  | No | Time of the operation. |
| eci | N..4 | No | Electronic commercial indicator. |
| amount | N..12 | No | Payment amount in minimum currency units. |
| currency | N..3 | No | The payment currency code is ISO 4217. If not specified, the default value is used. |
| rrn | AN12 | No | Unique identification number of the operation assigned after performing this operation. |
| acsUrl | AN..512 | No | The redirect address to the ACS server address. |
| termUrl | AN..512 | No | The payer's redirect link after authentication. |
| paReq | AN..512 | No | Request for payer authentication. |
| userMessage | ANS...512 | No | Message to the user with a description of the result code. |

##### Request example:

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:mer="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<soapenv:Header>

<wsse:Security xmlns:wsse="http://docs.oasis- open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-

%20wssecurity-utility-1.0.xsd">

<wsse:UsernameToken wsu:Id="UsernameToken-87">

<wsse:Username>with\_verify-api</wsse:Username>

<wsse:Password Type="http://docs.oasis- open.org/wss/2004/01/oasis-200401-wss-username-token-profile- 1.0#PasswordText">testPwd</wsse:Password>

</wsse:UsernameToken>

</wsse:Security>

</soapenv:Header>

<soapenv:Body>

<mer:verifyCard>

<request amount="100" currency="170" description="test" pan="5555555555555599" cvc="123" expiry="202412" cardHolderName="Test Cardholder" email="test@test.ru" language="es" backUrl="httsp://google.ru" failUrl="httsp://google.ru"/>

</mer:verifyCard>

</soapenv:Body>

</soapenv:Envelope>

##### Response example:

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<ns1:verifyCardResponse xmlns:ns1="https://ecom.alfabank.ru/api/soap/merchant-ws/">

<**return** orderNumber="148021" orderId="14ca4142-d56c-752c-8771- 3f6700adf340" rrn="111111111115" authCode="123456" actionCode="0" actionCodeDescription="" time="2020-09-04T15:00:57.186+03:00" eci="07" amount="100" currency="170" errorCode="0" errorMessage="Éxito"/>

</ns1:verifyCardResponse>

</soap:Body>

</soap:Envelope>

* 1. REST Interface
     1. Request for order registration (REST)

To register an order, the register.do request is used (see the section "Connection coordinates").

##### Request parameters:

| **Name** | | **Type** | **Mandatory** | **Description** | |
| --- | --- | --- | --- | --- | --- |
| userName | | AN..30 | no (either a login and password, or a token needs to be specified) | AN..30 | |
| password | AN..30 | | no (either a login and password, or a token needs to be specified) | AN..30 |
| token | AN..30 | | no (either a login and password, or a token needs to be specified) | ANS..30 |
| orderNumber | AN..32 | | yes | The order number (identifier) in the store's system is unique for each store within the system |
| amount | N..12 | | yes | Payment amount in kopecks (or cents) |
| currency | N3 | | no | The payment currency code is ISO 4217. If not specified, the default value is used. |
| returnUrl | ANS..512 | | yes | The address to which the user must be redirected in case of successful 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>**. |
| See also the **3DS payment** block - for one-phase and two-phase payments. returnUrl is used in the same way as in step 11 in the specified patterns. |
| failUrl | ANS..512 | | no | The address to which the user must be redirected in case of 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>**. |
| See also the **3DS payment** block - for one-phase and two-phase payments. failUrl is used in the same way as in step 11 in the specified patterns. |
| dynamicCallbackUrl | Unable to render include or excerpt- include. Could not retrieve page. | | no | Unable to render include or excerpt-include. Could not retrieve page. |
| description | ANS..598 | | no | Description of the order in any form. |
| To enable sending this field to processing, contact technical support. |
| language | A2 | | no | The language is encoded in ISO 639-1. If not specified, the default language specified in the store settings will be used. |
| pageView | ANS..20 | | no | The value of this parameter determines which pages of the payment interface should be loaded for the client. Possible values are:   * DESKTOP – for pages designed to be displayed on PC screens (pages with names payment\_<locale>.html and errors\_<locale>.html will be searched for in the archive of the payment interface pages); * MOBILE – for pages designed to be displayed on mobile devices (pages with names mobile\_payment\_<locale>.html and mobile\_errors\_<locale>.html will be searched for in the archive of the payment interface pages); * If the store has added arbitrary prefixes to their page file names, value of the desired prefix should be passed in the pageView parameter for the corresponding pages to be loaded. For example, if iphone is passed in the parameter, pages with names   iphone\_payment\_<locale>.html and  iphone\_error\_<locale>.html names will be searched for in the archive of the payment interface pages.  Where:  locale – the l anguage of the page encoded in ISO 639-1. For example, ru for Russian or en for English.  If the parameter is missing or does not match the format, the default value is pageView=DESKTOP. |
| clientId | AN..255 | | no | The client's number (ID) in the store's system. It is used to implement the functionality of bindings. It can be present if creating bindings is allowed for the store. |
| Specifying this parameter when processing payments with the use of bindings is mandatory. Otherwise, a payment will be failed. |
| merchantLogin | AN..255 | | no | To register an order on behalf of a child merchant, specify the merchant login in this parameter. |
| jsonParams | See description | | no | A block for passing additional merchant parameters. The fields for additional information and its subsequent storage are passed as:  {"<name1>":"<value1>",...,"<nameN>":"<valueN>"},  These fields can be passed to the Bank processing system to be subsequently displayed in the registries.\*  This functionality can be enabled upon additional agreement with the bank during the integration period.  If sending notifications to a buyer is set up for a seller, the email address of a buyer must be passed in this block in the email parameter.  Data type   |  |  | | --- | --- | | **Name** | **Type** | | name | ANS..255 bytes | | value | ANS..1024 bytes | |
| In case of payments for housing and utilities services, the following additional parameters must be passed:   * payment\_doc\_id – the identifier of the payment document; * order\_num – the order number. |
| sessionTimeoutSecs | N...9 | | no | The order lifetime in seconds.  If the parameter is not specified, the value specified in the merchant's settings or the default time (1200 seconds = 20 minutes) will be used.  If the request contains the expirationDate parameter, the value of the sessionTimeoutSecs parameter is ignored. |
| expirationDate | ANS | | no | Date and time of the order lifetime expiration. Format:  yyyy-MM-ddTHH:mm:ss.  If this parameter is not present in the request, then sessionTimeoutSecs is used to determine the time of the order lifetime expiration. |
| bindingId | AN..255 | | no | ID of the binding created earlier. It can only be used if the store has permission to work with bindings. If this parameter is passed in this request, it means that:   1. This order can only be paid using a binding; 2. The payer will be redirected to the payment page, where only CVC input is required. |
| features | ANS..255 | | no | The following values can be used.  AUTO\_PAYMENT – The payment is made without authentication of the cardholder (without CVC and 3D-Secure). In order to make such payments, the seller must have the appropriate permits.  VERIFY – 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. |
|  |  | |  | **Specifics of passing the VERIFY value** 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. |
|  |  | |  | FORCE\_TDS – use of 3-D Secure is enforced for the payment. If the card does not support 3-D Secure, the transaction will fail.  FORCE\_SSL – use of SSL is enforced for the payment (no 3-D Secure).  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. |
| email | ANS..40 | | no | Buyer's email address. |
| phone | ANS.12 | | no | Buyer'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:  • +79998887766;  • 9998887766. |

**\*** By default the following fields are passed to the bank's 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 | The order number in the payment system. It is unique within the system. It is missing if the order registration failed due to an error detailed in ErrorCode. |
| formUrl | AN..512 | No | URL of the payment form to which the client's browser is to be redirected. It is not returned if the order registration failed due to an error detailed in ErrorCode. |
| errorCode | N3 | No | Error code. |
| errorMessage | AN..512 | no | Error description in the language passed in the language parameter in the request. |

**Error codes (**ErrorCode **field):**

| **Meaning** | **Description** |
| --- | --- |
| 0 | Request processing took place without system errors |
| 1 | Order with given order number is processed already |
| 1 | An order with this number was registered, but was not paid for |
| 1 | Invalid order number |
| 3 | Currency is not defined |
| 4 | The order number cannot be empty |
| 4 | The merchant's name cannot be empty |
| 4 | Empty amount |
| 4 | The return URL cannot be empty |
| 4 | The password cannot be empty |
| 5 | The seller's login is incorrect |
| 5 | Incorrect amount |
| 5 | Incorrect 'Language' parameter |
| 5 | Access denied |
| 5 | The user must change password |
| 5 | Access denied |
| 5 | [jsonParams] is invalid |
| 7 | System error |
| 13 | Using both values, Features FORCE\_TDS/FORCE\_SSL and AUTO\_PAYMENT, is not allowed |
| 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 |

##### Example of a POST request:

amount=100&currency=810&language=ru&orderNumber=87654321&returnUrl=htt p://yoursite.com&pageView=DESKTOP&jsonParams={"param1":"value1","param2":"value2"}&expirationDate=2014-09- 08T14:14:14&merchantLogin=merch\_child&features=AUTO\_PAYMENT

##### Response example:

{"orderId":"70906e55-7114-41d6-8332-

4609dc6590f4","formUrl":"https://server/application\_context//merchants

/test/payment\_ru.html?mdOrder=70906e55-7114-41d6-8332-4609dc6590f4"}

* + 1. Request for order registration with pre-authorization (REST)

To request registration of an order with pre-authorization, the registerPreAuth.do request is used (see the section "Connection coordinates").

|  |
| --- |
| The ability to make two-phase payments is available on a separate request. |

##### Request parameters:

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| userName | AN..30 | no (either a login and password, or a token needs to be specified) | AN..30 |
| password | AN..30 | no (either a login and password, or a token needs to be specified) | AN..30 |
| token | AN..30 | no (either a login and password, or a token needs to be specified) | ANS..30 |
| orderNumber | AN..32 | yes | The order number (identifier) in the store's system is unique for each store within the system |
| amount | N..12 | yes | Payment amount in kopecks (or cents) |
| currency | N3 | no | The payment currency code is ISO 4217. If not specified, the default value is used. |
| returnUrl | ANS..512 | yes | The address to which the user must be redirected in case of successful 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>**. |
|  |  |  | See also the **3DS payment** block - for one-phase and two-phase payments. returnUrl is used in the same way as in step 11 in the specified patterns. |
| failUrl | ANS..512 | no | The address to which the user must be redirected in case of 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>**. |
|  |  |  | See also the **3DS payment** block - for one-phase and two-phase payments. failUrl is used in the same way as in step 11 in the specified patterns. |
| dynamicCallbackUrl | Unable to render include or excerpt- include. Could not retrieve page. | no | Unable to render include or excerpt-include. Could not retrieve page. |
| description | ANS..598 | no | Description of the order in any form. |
| To enable sending this field to processing, contact technical support. |
| language | A2 | no | The language is encoded in ISO 639-1. If not specified, the default language specified in the store settings will be used. |
| pageView | ANS..20 | no | The value of this parameter determines which pages of the payment interface should be loaded for the client. Possible values are: |
|  |  |  | * DESKTOP – for pages designed to be displayed on PC screens (pages with names payment\_<locale>.html and errors\_<locale>.html will be searched for in the archive of the payment interface pages); * MOBILE – for pages designed to be displayed on mobile devices (pages with names mobile\_payment\_<locale>.html and mobile\_errors\_<locale>.html will be searched for in the archive of the payment interface pages); * If the store has added arbitrary prefixes to their page file names, value of the desired prefix should be passed in the pageView parameter for the corresponding pages to be loaded. For example, if iphone is passed in the parameter, pages with names iphone\_payment\_<locale>.html and iphone\_error\_<locale>.html will be searched for in the archive of the payment interface pages.   Where:  locale – the l anguage of the page encoded in ISO 639-1. For example, ru for Russian or en for English.  If the parameter is missing or does not match the format, the default value is pageView=DESKTOP. |
| clientId | AN..255 | no | The client's number (ID) in the store's system. It is used to implement the functionality of bindings. It can be present if creating bindings is allowed for the store. |
| Specifying this parameter when processing payments with the use of bindings is mandatory. Otherwise, a payment will be failed. |
| merchantLogin | AN..255 | no | To register an order on behalf of a child merchant, specify the merchant login in this parameter. |
| jsonParams | AN..1024 | no | A block for passing additional merchant parameters. The fields for additional information and its subsequent storage are passed as:  {"<name1>":"<value1>",...,"<nameN>":"<va lueN>"},  These fields can be passed to the Bank processing system to be subsequently displayed in the registries.\*  This functionality can be enabled upon additional agreement with the bank during the integration period.  If sending notifications to a buyer is set up for a seller, the email address of a buyer must be passed in this block in the email parameter. |
| sessionTimeoutSecs | N...9 | no | The order lifetime in seconds.  If the parameter is not specified, the value specified in the merchant's settings or the default time (1200 seconds = 20 minutes) will be used.  If the request contains the expirationDate parameter, the value of the sessionTimeoutSecs parameter is ignored. |
| expirationDate | ANS | no | Date and time of the order lifetime expiration. Format:  yyyy-MM-ddTHH:mm:ss.  If this parameter is not present in the request, then sessionTimeoutSecs is used to determine the time of the order lifetime expiration. |
| bindingId | AN..255 | no | ID of the binding created earlier. It can only be used if the store has permission to work with bindings. If this parameter is passed in this request, it means that:   1. This order can only be paid using a binding; 2. The payer will be redirected to the payment page, where only CVC input is required. |
| features | ANS..255 | no | The following values can be used.  AUTO\_PAYMENT – if the request for order registration initiates auto-payments.  VERIFY – 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. |
|  |  |  | **Specifics of passing the VERIFY value**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. |
| autocompletionDate | ANS..19 | no | The date and time of autocompletion of the two-phase payment is in the following format.  2017-12-29T13:02:51 |
|  |  |  | To enable this functionality, contact technical support. |
| email | ANS..40 | no | Buyer's email address. |
| phone | ANS.12 | no | Buyer'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:  • +79998887766;  • 9998887766. |

**\*** By default the following fields are passed to the bank's 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 | The order number in the payment system. It is unique within the system. It is missing if the order registration failed due to an error detailed in ErrorCode. |
| formUrl | AN..512 | No | URL of the payment form to which the client's browser is to be redirected. It is not returned if the order registration failed due to an error detailed in ErrorCode. |
| errorCode | N3 | No | Error code. |
| errorMessage | AN..512 | no | Error description in the language passed in the language parameter in the request. |

**Error codes (**ErrorCode **field):**

| **Meaning** | **Description** |
| --- | --- |
| 0 | Request processing took place without system errors |
| 1 | Order with given order number is processed already |
| 1 | An order with this number was registered, but was not paid for |
| 1 | Invalid order number |
| 3 | Currency is not defined |
| 4 | The order number cannot be empty |
| 4 | The merchant's name cannot be empty |
| 4 | Empty amount |
| 4 | The return URL cannot be empty |
| 4 | The password cannot be empty |
| 5 | Incorrect amount |
| 5 | Incorrect 'Language' parameter |
| 5 | The seller's login is incorrect |
| 5 | Access denied |
| 5 | The user must change password |
| 5 | Access denied |
| 5 | [jsonParams] is invalid |
| 7 | System error |
| 13 | Using both values, Features FORCE\_TDS/FORCE\_SSL and AUTO\_PAYMENT, is not allowed |
| 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 |

##### Example of a POST request:

amount=100&currency=810&language=ru&orderNumber=87654321&returnUrl=htt p://yoursite.com&pageView=MOBILE&jsonParams={"param1":"value1","param2 ":"value2"}&merchantLogin=merch\_child&features=AUTO\_PAYMENT

##### Response example:

{"orderId":"61351fbd-ac25-484f-b930- 4d0ce4101ab7","formUrl":"https://server/application\_context/merchants/ test/mobile\_payment\_ru.html?mdOrder=61351fbd-ac25-484f-b930- 4d0ce4101ab7"}

* + 1. Request to verify the card's enrolment in 3DS (REST)

The verifyEnrollment.do request is used to verify the card's involvement in 3DS.

##### Request parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Compulsoriness** | **Description** |
| userName | AN..30 | yes | User login (API) |
| password | AN..30 | yes | User password (API) |
| pan | N..19 | yes | Payment card number. |

##### Response parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Compulsoriness** | **Description** |
| errorCode | N3 | no | Error code. |
| errorMessage | AN..512 | no | Error description. |
| enrolled | A1 | no | A sign of the card's enrolment in 3DS. Possible values: Y, N, U. |
| emitterName | AN..160 | no | Name of the Issuing bank. |
| emitterCountryCode | AN..4 | no | Country code of the issuing bank. |

##### Error Codes (ErrorCode field):

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 1 | The card number is not specified |
| 1 | The card number must be a number containing from 13 to 19 digits |
| 5 | The user must change password. |
| 5 | Access denied |
| 6 | No information was found for the specified card number. |
| 7 | System error. |

##### Example of a POST request:

pan=4111111111111111

##### Response example:

{"errorCode":"0","errorMessage":"Успешно","emitterName":"TEST

CARD","emitterCountryCode":"RU","enrolled":"Y"}

* + 1. Order Payment Requests (REST)
       1. Request for order payment when using an external MPI (REST)

##### Request parameters:

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| userName | AN..30 | yes | The store's login received when connecting |
| password | AN..30 | yes | The store's password received when connecting |
| MDORDER | ANS36 | yes | Order number received during order registration |
| $PAN | N..19 | yes | Payment card number. |
| $CVC | N..3 | yes | CVC |
| YYYY | N..4 | yes | Year |
| MM | N..2 | yes | Month |
| TEXT | A..512 | yes | Cardholder's name |
| language | A..2 | yes | Response language |
| ip | ANS.. 39 | yes | The IP address of the user who paid for the order. |
| IPv6 is supported in all requests. |
| email | ANS..  \* | no | The payer's email address. If the merchant's settings provide for sending notifications to clients about order transactions, then this parameter must be present. |
| bindingNotNeeded | boolean | no | Possible values are:   * true – prohibiting the creation of a binding after the payment (the client ID transmitted during order registration will be removed from the order details after the payment); * false – as a result of a successful payment, a binding can be created (if the appropriate conditions are met). This value is the default value. |
| json Params | AN..1024 | no | The fields for additional information and its subsequent storage, such as  {"param":"value1","param2":"value2"}.  These fields can be passed to the Bank processing system to be subsequently displayed in the registries.\* This functionality can be enabled upon additional agreement with the bank during the integration period.  In the case of using an external MPI, IPS expects that the "eci" parameter will be received in each request for order payment. If the "eci" value differs from those used for SSL authorizations, it is also necessary to send the "xid" and "cavv" parameters. |
|  |  |  | When paying with MIR cards with 3-D Secure support, if ECI=01 or ECI=02, the CAVV value must be transmitted on a mandatory basis. |
|  |  |  | The authenticationTypeIndicator parameter (payment authentication type) is required when making a payment via an external MPI using the 3DS 2.0 protocol.  If 3DS 1 or SSL is used in the external MPI, the authenticationTypeIndicator parameter is optional and will be determined automatically based on the ECI value.  Possible values are:   |  |  |  | | --- | --- | --- | | **Meaning** | **Decoding** | **Compulsoriness/ automatic detection rules** | | 0 | SSL | ECI = 07 | |  | SSL authentication |  | | 1 | THREE\_DS1\_FULL | ECI = 02, 05 | |  | 3DS 1 authentication |  | | 2 | THREE\_DS1\_ATTEMPT | ECI = 01, 06 | |  | 3DS 1 authentication attempt |  | | 3 | THREE\_DS2\_FULL | Mandatory for 3DS 2.0. | |  | Strong authentication (SCA) |  | | 4 | THREE\_DS2\_FRICTIONLESS | Mandatory for 3DS 2.0. | |  | Risk-based authentication (RBA) |  | | 5 | THREE\_DS2\_ATTEMPT  Authentication attempt (3DS 2.0) | Mandatory for 3DS 2.0. | |
|  |  |  | Authentication attempt (3DS 2.0)  If you want to accept payment by installments, and executed a [get request](#_bookmark75) to obtain information about the available installment options, you should pass the contents of the installmentsProposals block received from that request as additional parameters. Example is given below.  jsonParams={"installmentsCount":2,"firstInstallmentAmount":56 0,"subSeqInstallmentAmount":560,"transactionAmount":70000,"to talAmount":1120,"currencyCode":"RUR","annualRate":54,"interes tRate":100,"installmentFee":0} |

\* *By default, the* orderNumber *and its* description *fields are passed to the bank's processing system (no more than 99 characters; %, +, carriage return \r, and line feed \n cannot be used)*

##### Response parameters:

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| errorCode | N..1 | yes | Error code |
| error | AN..512 | no | Error message |
| info | AN..512 | no | The result of the payment attempt. The possible values are shown below:   * Your payment is proceeded, redirecting... * The operation was rejected. Check the entered data, the sufficiency of funds on the card and repeat the operation. Redirecting... * Sorry, the payment cannot be made. Redirecting... * The operation was rejected. Contact the store. Redirecting... * The operation was rejected. Contact the bank that issued the card. Redirecting... * The operation is impossible. Cardholder authentication failed. Redirecting... * No connection with the bank. Repeat later. Redirecting... * Data entry timeout. Redirecting... * No response was received from the bank. Repeat later. Redirecting... |

If **errorCode** = 0 and **info** = Your order is processed, redirecting, the payment was successful. In other cases with an error, see the error message in the **error** field.

Error codes (success field):

| **Meaning** | **Description** |
| --- | --- |
| 0 | Request processing took place without system errors |
| 5 | All payment attempts have been used |
| 5 | System or internal error |

**Request example:**

MDORDER=0d4b02cb-5147-4232-9012- 4d38c743ahr6&$PAN=5555555555555599&$CVC=123&YYYY=2015&MM=12&TEXT=Card

Holder&language=ru&jsonParams={"eci":"05","xid":"MDAwMDAwMDEzMzkyMjg5ODExN Tc=","cavv":"AAABCpEChRM5IomAKFAAAAAAAAA=","authenticationTypeIndicator":" 1"}

##### Response example:

{"info":"Your payment is proceeded, redirecting...", "redirect":"finish.html?orderId=00d4b02cb-5147-4232-9012- 4d38c743ahr6","success":0}

* + - 1. Request for order payment when using an internal MPI (REST)

The paymentorder.do method is used to request payment.

Only POST is supported.

##### Request parameters:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | | **Type** | **Mandatory** | **Description** | |
| userName | | AN..30 | yes | The store's login received when connecting | |
| password | | AN..30 | yes | The store's password received when connecting | |
| MDORDER | | ANS36 | yes | Order number received during order registration | |
| $PAN | | N..19 | yes | Payment card number. | |
| $CVC | | N..3 | yes | CVC | |
| YYYY | N..4 | | yes | Year |
| MM | N..2 | | yes | Month |
| TEXT | A..512 | | yes | Cardholder |
| language | A..2 | | yes | Response language |
| ip | ANS..39 | | yes | The IP address of the user who paid for the order. |
| IPv6 is supported in all requests. |
| email | ANS..\* | | no | The payer's email address. If the merchant's settings provide for sending notifications to clients about order transactions, then this parameter must be present. |
| bindingNotNeeded | boolean | | no | Possible values are:   * true – prohibiting the creation of a binding after the payment (the client ID transmitted during order registration will be removed from the order details after the payment); * false – as a result of a successful payment, a binding can be created (if the appropriate conditions are met). This value is the default value. |
| json Params | AN..1024 | | no | The fields for additional information and its subsequent storage,  such as {"param":"value","param2":"value2"}. These fields can be passed to the Bank processing system to be subsequently displayed in the registries.  By default, the orderNumber and its description fields are passed to the bank's processing system (no more than 99 characters; %, +, carriage return \r, and line feed \n cannot be used).  This functionality can be enabled upon additional agreement with the bank during the integration period.  If you want to accept payment by installments, and executed a [get request](#_bookmark75) to obtain information about the available installment options, you should pass the contents of the installmentsProposals block received from that request as additional parameters. Example is given below.  jsonParams={"installmentsCount":2,"firstInstallmentAmount":56 0,"subSeqInstallmentAmount":560,"transactionAmount":70000,"to talAmount":1120,"currencyCode":"RUR","annualRate":54,"interes tRate":100,"installmentFee":0} |

##### Response parameters:

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| errorCode | N..1 | yes | Error code |
| error | AN..512 | no | Error message |
| info | AN..512 | no | The result of the payment attempt. The possible values are shown below:   * Your payment is proceeded, redirecting... * The operation was rejected. Check the entered data, the sufficiency of funds on the card and repeat the operation. Redirecting... * Sorry, the payment cannot be made. Redirecting... * The operation was rejected. Contact the store. Redirecting... * The operation was rejected. Contact the bank that issued the card. Redirecting... * The operation is impossible. Cardholder authentication failed. Redirecting... * No connection with the bank. Repeat later. Redirecting... * Data entry timeout. Redirecting... * No response was received from the bank. Repeat later. Redirecting... |
| redirect | AN..512 | no | Return address after payment |
| termUrl | AN..512 | no | It is not used for payments that do not require additional authentication through the Issuing Bank's ACS. |
| acsUrl | AN..512 | no | It is not used for payments that do not require additional authentication through the Issuing Bank's ACS. |
| paReq | AN..512 | no | It is not used for payments that do not require additional authentication through the Issuing Bank's ACS. |

Error Codes (ErrorCode field):

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 5 | All payment attempts have been used |
| 5 | System or internal error |

##### Request example:

MDORDER=0d4b02cb-5147-4232-9012- 4d38c743ahr6&PAN=5555555555555599&$CVC=123&YYYY=2015&MM=12&TEXT=Card

Holder&language=ru

##### Example of a response in the case of a payment that does not require additional authentication through the Issuing Bank's ACS:

{"redirect":"www.ya.ru?MDORDER=8cfb5c89-c2db-49e4-80c5 - a0569c897cbf&ANSWER=%3C%3Fxml+version%3D%221.0%22+encoding%3D%22UTF- 8%22%3F%3E%0A%3CPSApiResult+primaryRC%3D%220%22+secondaryRC%3D%220%22% 2F%3E&STATE=payment\_deposited&ACTION\_CODE=0&AUTH\_CODE=2","info":"Your payment is proceeded, redirecting...","errorCode":0}

##### Example of a response in the case of a 3DS payment that requires additional authentication through the Issuing Bank's ACS:

{"info":"Your payment is proceeded, redirecting...", "acsUrl":"https://tws.egopay.ru/acs/auth/start.do", "paReq":"eJxVUdFygjAQ/BWG9xKioNQ54tCirQ+oo/QDKNwAVYIGENuvbyJQ60Nmdu8um 80ezK/FUbugqPKS\nuzo1TF1DHpdJzlNX/wiXT44+ZxBmAtHfY9wIZBBgVUUpanni6kWVG lRnsPV2eGbQKzEpZIyADFRe\nEXEW8ZpBFJ9fVmtmjaYT0wTSUyhQrHw2BdIB4FGBbI8X5 Is8zep1zhHIrQhx2fBafLOJNQYyEGjE\nkWV1fZoR0ratkZZlekRDNEBUB8jdwbZRqJJK1 zxhge+13Vn8bMLD9yZM7eDLs4Lw4AJRE5BENbKR\nSS3TNm2NOjPTmVnS+60OUaEsMGrKr vpQR+GkXvEee/9rILMUMurhHwMDvJ5KjnJC5veHgdxdv76r\nFONaxuSveRIvnyfeWxjsN k4QHXJaFZ/V+NIuVba3IaWYy4ioTTtJRYAoGdKvjfQrlehh1b8el7SS\n", "termUrl":"https://tws.egopay.ru/:443/ab/rest/finish3ds.do", "errorCode":0}

##### If the payment requires the use of 3-D Secure technology, then after receiving a response to the payment request, the client must be redirected to ACS. There are two ways of redirection: normal and simplified (see the table below).

| **Redirection method** | **Description** |
| --- | --- |
| Normal | To redirect to ACS, the merchant, for its part, must send the client to the address specified in the acsUrl parameter with the request body MD=mdorder&PAReq=pareq&TermUrl=redirect, where:   * mdorder - unique order number in the payment gateway system, * pareq - parameter from the response to the payment request, * redirect - parameter from the response to the payment request (the address of return from ACS) or the store's address, if the pattern of return from ACS to the store is used.   The request must be in POST format.  Depending on the connection pattern used, the client will be redirected to the payment gateway or to the store after ACS authentication. The payment completion processes for each of these cases are described below. |
| Simplified | In order for the client to get to the ACS page, the seller redirects them to the payment gateway page of the following type: https://tws.egopay.ru/ab/acsRedirect.do?orderId =<order number>  where: <order number> is the client's unique order number.  Then, without additional actions on the part of the client, the payment gateway redirects the client to the Issuing bank's ACS page, where the client is authenticated. |

* + 1. Finish3DS Request (REST)

With this pattern, the issuer's ACS authenticates the cardholder and redirects the client back to the store. The PARes received from ACS is transmitted to the store, after which the store transmits it to the gateway using the finish3dsPayment.do method.

The finish3dsPayment.do method is described below:

##### Request parameters:

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| userName | AN..30 | yes | The store's login received when connecting. |
| password | AN..30 | yes | The store's password received when connecting. |
| mdOrder | AN..30 | yes | Order number received during order registration |
| paRes |  | yes | Payer Authentication Response - response to payer identification |

##### Response parameters

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| redirect | AN..512 | no | URL of the payment form to which the client's browser is to be redirected. |
| errorCode | N3 | no | Error code. It may be missing if the result did not lead to an error. |
| error | AN..512 | no | Error description in the language passed in the Language parameter in the request. |

If **errorCode** = 0, the payment was successful. In other cases with an error, see the **error** message.

Error Codes (ErrorCode field):

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 5 | Access denied |
| 5 | The user must change password |
| 5 | 3DS operation completion error |
| 7 | Order not found |
| 7 | System error |

##### Request example:

mdOrder=906bf262-bd53-4ac7-983c- 07127954681b&paRes=eJzFV2uTokoS%2FSsTcz8aPYCC6A3HiOKNCshT8BtCyRuUN%2Fz 6RXu6p%2B9s78bsbtxYIgyLE5VJnso8%0D%0AlVUbIywhZHToNSXcbiRYVW4Av0T%2B969 ZFXzDvm43R6DB6on4qwW%2Bdq%2FwhVi58AVfXvAXlySWL95l%0D%0AQULU8yD0HwYtLKu oyLfYN%2FTbfIO8vU7OSy9083q7cb07JcpbfE4uUXSD%2FHjdZLAUmS25QV4HG%2BSn%0D

%0AxbF5jKopxD7ytxIDutcfOypGMigGi8mj2Mlj8n2DPGZsfLeG2zmK4SiBEl%2Bw5Z%2F o8k9siuaJb24P%0D%0AdyArmsn3nEDRRxQfsc20HCXMvWG7xBcb5P1tA%2FtbkcOH1QZ5H 2%2BQn%2BHd3HyLfniw6Zl8T%2BjGsLeb%0D%0AOso%2BDeuJb6rarZtq62yQH6ON57btF gBAUQt%2FlELlvNechTSAt2ei%2B5yygV60RYkpqOn%2FaQXSoCij%0D%0AOsweof4V2CC

PUJBnYrcbPQry6WMl%2FNJnaV59%2FxrW9e1PBOm67lu3%2BFaUATKfiCDoGpkm%2BFUU% 2FPH1%0D%0A1Qr6Yn4t%2FiMz2s2LPPLcNBrdeqoKCdZh4X95j%2B0zN4b28IQhGku%2FT

K5ePAzPXx4IusCIySfyudMP%0D%0AzH7nK78GW1buSxW6j4JGfnG03WjwCh8VAb%2BYmvj 96x%2B%2FqQwmCmBV%2FzfRvEXy0cObP8tNG7hlvTHg%0D%0AZld%2BdR0XC5bRZ0wYBGu MX0TS9ze715kb5D38H9xeE%2FlhwV4nhqxy1zVg8vNYF6J7bIQ75sC31BX1%0D%0AvfnyZ IdDwMh25uRqdQpmB%2FFgIVlIKGMHfIHB%2FQYfHIkQuagyA0JJOEij%2FEj267KDx4zB6 cpEayhj%0D%0A6JGjXBat9Pu51hZ3S7RqlaN3A4Pn%2BpAplzUteVK%2FNnftIfCCarw7p nsiLt3N6B2MtOSdkF%2BC7x%2BS%0D%0A9IPlHg6vrGwCXTNu7b6OaFjW0XWqlmkfkESRL gyaBsE9AJ1IgUDcgWPO6SN0oll6kQDK0%2Fqd18XL%0D%0AglFZilJNILGgp0ewowLZooB jgNQyJA10TOAwlqru2c5nHHsXnnlulNSuo1%2FxA9vJqm6pPWuA46ut%0D%0AZNACdfNpb LjM16hE4TZjsKjESJ0cA1QazV7iigkTn5g0vmOdyrAHCSQ8wEyWCiXasqSeZYDy6rcw%0D

%0AKFQOvYxLHFvqGQbsX%2FHKABi1sxhWk8DqaQt6SfT4FD3Yu%2FayULvgvOoY1dnti7M Ytp4MnpzBRA0F%0D%0AksjvQMFTYM%2BTfuSWjBpgpBln7c6tBE6PT%2BHRv6yR9c0%2Bt yewb7x1d7%2FHzCJvzWU3a3a1I3J3GW9t%0D%0AehYnJExwiyPjNmG4hYai11xIl51iNdd RTTlAzlT7djIkvrosh6ELbamzLp6yajDc4y2knxEW7BIj%0D%0AKC3tcj%2Bnii1mME3l9 oRdqx0hRLp1FVZwSEEgUQDwceCKT26COuVCQxWKclhOThaLcK1iFH%2BNCrLp%0D%0A9zK cp46PCc1OEsBjfXyx024SD8G%2Fm1uYeqLGUzbVqqPVZ655tttZ5siqEsCf60yHkqDOucE 5aaE3%0D%0AspI0LeQT7yXGZFNT0tiO6Z62DNuFnWurtZetW1%2F8tR67Z1xsp3ISmLhd% 2Fylf3Gu%2BWCDC87m21XtK%0D%0A7VHcnQdRxicF4h7mzknSghDFB2M%2FiztSSXaIefI BFpArc5cQSGns1nOCXEcFUVbZQbCtWXPylqdy%0D%0AKXIKDq59pLYz%2FoQcJXV1a%2FB 1sYLtejBaxy2qlX3ER99UcK2yqW5dOlqjH3GlndfKfn8sMmV3cC86%0D%0AMVVfShdNQpI 40hvsJN5flfmpVPPxIdXop1SnKucREQpdvf5%2FSlVmnFHG%2FirVH1h3jP8nqR4fvH5I% 0D%0AlX9K9SSHkxSvv%2FIFE9%2Bp2mTA0FSkTm5URl91ScOZHJ64buTqh5lGLC03GbFm6

jwOd0fYwt6NB71D%0D%0AQe2eVhgUGjoc4Iq0A4Y%2F2g1g6eKy17BbQcZZ1pIylt2LtPc PdQ0KYi%2BWpzVU5OUMmS10vBaTZl%2Fe%0D%0AZ0vrYAy9TtQZKUzYwhICxW2c%2Bxl0P

t3r3F1suLhq4rCa2XbfdlRTEFUnMkAFVIHz0ZMb9eTsM4F6%0D%0AoihdhwoCbq2PiM6xC EEgYKLP6vIN2TNU91gvQZdY16DygDLO8hq%2FmrKq5fhdISs9tuvbSfA1MnI5%0D%0A9sa Kf9PWqdO0yhC9WS0ysb2dxkmbj7g0iaUMMHETkM%2Fqc8oXBejaptC2TtGZapqtwKfFKrj TXRyH%0D%0AHXO4uXNLq1K3qfZ8vcJP6TGyUUOYU%2Fc%2BQ8bsOFCxnDf2iRcyFd8HXHZ

3UR1yGLgeEWcBTpDdgeWu%0D%0AgDGxynk0LJe0p1z6kp51QNKOjJgeBGVGYOoQ32RuLdC zZL689KZ7O1dz64TMb95xSWa1Kgx25Ki%2F%0D%0AJ1Nq9uiop%2F2bTE1pPbWOJqHtBp%

2F5eXPwhjM77Bt8tQo%2B273%2BHsm8S1btJ8nisgF6%2BdldWewXrBNj%0D%0Aduob%2B JsP28rS0dEpwxd24SXzU5G1pjaUDlCnlLPt%2FV7XFB2%2BVNG7tM%2FpS2UkeKdTUd4J3 HrG32v7%0D%0AeIgx2tnPdT%2FLmVpC1%2BMSdk17UOvMUwaubWoMoeZJo2p8CfXrCuHJS 3XjclU9hXToTYe9DCACqpjl%0D%0APnKUgIpMkT7g0SGIjYt%2F5cANMjiXFDDDiOvcSpX VsrQKkpArgPX22WLQEsbKFQTUVTbat675WPOf%0D%0AHYelkE6lJQD%2BZceRZX%2BJHI2

BnN0DWd%2Bdg%2BqCCnmEEGrEqUrSoNJhQeAXWshEyYMhvuOQYbYGGuOL%0D%0AuhK6Mm4 oOrGcS6OD7KU0ETskYXSWu5SB2xP3qh1UJmKsBncxEd602JvfrYrq2LkLrd0Mk8hjnB%2F G%0D%0A7naxylHVT7rQOJ15iTUuXhZuMtic3BxPzTK7rDX0s46D%2FDwnIu9nx5%2Bnyud 19Hk7ftyfPt6a%2FwFG%0D%0ADPms%0D%0A

##### Response example:

{"redirect":"http://ya.ru?orderId=906bf262-bd53-4ac7-983c- 07127954681b","errorCode":0}

* + 1. Order Status Request (REST)

The getOrderStatus request.do is used to get the current order status. The order status must be determined by the value of the OrderStatus parameter.

The AuthCode field is deprecated.

##### Request parameters:

| **Name** | **Type** | **Compulsoriness** | **Description** |
| --- | --- | --- | --- |
| userName | AN..30 | yes | The store's login received when connecting |
| password | AN..30 | yes | The store's password received when connecting |
| orderId | ANS36 | yes | The order number in the payment system. It is unique within the system. |
| language | A2 | no | The language is encoded in ISO 639-1. If not specified, it is assumed that the language is Russian. The error message will be returned in this language. |

##### Response parameters:

| **Name** | **Type** | **Compulsoriness** | **Description** |
| --- | --- | --- | --- |
| orderStatus | N2 | No | The value of this parameter determines the status of the order in the payment system.  The list of possible values is given in the table below. It is missing if the order has not been found. |
| errorCode | N3 | No | Error code. |
| errorMessage | AN..512 | No | Error description in the language passed in the Language parameter in the request. |
| orderNumber | AN..32 | Yes | Order number (ID) in the store's system |
| Pan | N..19 | no | The masked number of the card that was used for payment. It is specified only after payment of the order. |
| expiration | N6 | no | Card expiration date in the YYYYMM format. It is specified only after payment of the order. |
| cardholderName | A..64 | no | Cardholder's name. It is specified only after payment of the order. |
| Amount | N..20 | yes | Payment amount in kopecks (or cents) |
| currency | N3 | no | The payment currency code is ISO 4217. If not specified, it is considered equal to 810 (Russian rubles). |
| approvalCode | AN6 | no | IPS authorization code. This field has a fixed length (6 characters) and can contain digits and Latin letters. |
| authCode | N3 | no | This field is deprecated. Its value is always "2", regardless of the order status and the authorization code of the processing system. |
| Ip | ANS..39 | no | The IP address of the user who paid for the order. |
|  |  |  | IPv6 is supported in all requests. |
| *BindingInfo element:* |  |  |  |
| clientId | AN..255 | no | The client's number (ID) in the store's system, transmitted during the order registration. Present only if the store is allowed to create bindings. |
| bindingId | AN..255 | no | ID of the binding created when paying for the order or used for payment.  Present only if the store is allowed to create bindings. |

**The** orderStatus **field can have the following values:**

|  |  |
| --- | --- |
| **Status number** | **Description** |
| 0 | The order is registered, but not paid for |
| 1 | The pre-authorized amount is reserved (for two-phase payments) |
| 2 | Full authorization of the order amount has been carried out |
| 3 | Authorization canceled |
| 4 | A refund operation was performed on the transaction |
| 5 | Authorization initiated through the issuing bank's ACS |
| 6 | Authorization denied |

**Error codes (**ErrorCode **field):**

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 2 | The order was rejected due to an error in the payment details |
| 5 | Access denied |
| 5 | The user must change password |
| 5 | [orderID] not specified |
| 6 | Invalid order number |
| 7 | System error |

##### Example of a POST request:

orderId=b8d70aa7-bfb3-4f94-b7bb-aec7273e1fce&language=ru

##### Response example:

{"expiration":"201512","cardholderName":"tr tr","depositAmount":789789,"currency":"810","approvalCode":"123456","a uthCode":2,"clientId":"666","bindingId":"07a90a5d-cc60-4d1b-a9e6- ffd15974a74f","ErrorCode":"0","ErrorMessage":"Success","OrderStatus":2

,"OrderNumber":"23asdafaf","Pan":"411111\*\*1111","Amount":789789}

* + 1. Advanced Order Status Request (REST)

The getOrderStatusExtended.do. request is used to request the status of a registered order.

##### Request parameters:

| **Name** | **Type** | **Compulsoriness** | **Description** |
| --- | --- | --- | --- |
| userName | AN..30 | yes | The store's login received when connecting |
| password | AN..30 | yes | The store's password received when connecting |
| orderId | ANS36 | yes\* | The order number in the payment system. It is unique within the system. |
| orderNumber | AN..32 | yes\* | The order number (ID) in the store's system. |
| language | A2 | no | The language is encoded in ISO 639-1. If not specified, it is assumed that the language is Russian. The error message will be returned in this language. |

\* The request must contain either orderId or orderNumber. If both parameters are present in the request, the OrderID is prioritized.

There are several sets of response parameters. Exactly what kind of sets of parameters will be returned depends on the version of getOrderStatusExtended specified in the seller's settings.

| **Name** | | **Type** | | **Mandatory** | **Description** | | **getOrderStatusExtended version** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| orderNumber | | AN..32 | | yes | The order number (ID) in the store's system. | | All versions. |
| orderStatus | | N2 | | no | The value of this parameter determines the order status in the payment system. A list of possible values is given in the list below.  It is missing if the order has not been found.   * **0** - The order is registered, but not paid for; * **1** - The pre-authorized amount is reserved (for two-phase payments); * **2** - A full authorization of the order amount has been carried out; * **3** - Authorization canceled; * **4** - A refund operation was performed on the transaction; * **5** - An authorization has been initiated through the issuing bank's ACS; * **6** - Authorization denied. | | All versions. |
| actionCode | | N3 | | yes | Response code. | | All versions. |
| actionCodeDescription | | AN..512 | | yes | Decryption of the response code in the language transmitted in the Language parameter in the request. | | All versions. |
| errorCode | | ANS..3 | | no | Error code. The following options are possible.   * **0** - Request processing took place without system errors; * **1** - [orderId] or [orderNumber] expected; * **5** - Access denied; * **5** - The user must change password; * **6** - Order not found; * **7** - System error. | | All versions. |
| errorMessage | | AN..512 | | no | Error description in the language passed in the Language parameter in the request. | | All versions. |
| amount | | N..20 | | yes | Payment amount in kopecks (or cents) | | All versions. |
| currency | | N3 | | no | The payment currency code is ISO 4217. If not specified, it is considered equal to 810 (Russian rubles). | | All versions. |
| date | | ANS | | yes | Order registration date. | | All versions. |
| depositedDate | | N | | no | The date of the order payment in UNIX-time format (POSIX-time). | | All versions. |
| orderDescription | | AN..512 | | no | Description of the order transmitted during its registration | | All versions. |
| ip | | ANS.. 39 | | yes | The IP address of the user who paid for the order. | | All versions. |
|  | |  | |  | IPv6 is supported in all requests. | |  |
| authRefNum | | AN..24 | | no | The payment authorization account number that is assigned when registering the payment. | | 02 and above. |
| refundedDate | | ANS | | no | Refund date and time. | | 13 and above. |
| paymentWay | | AS..14 | | yes | The method of making the payment (payment with the input of card data, payment by bindings, etc.).  The payment method is passed in this parameter. It can have the following values:   * CARD – payment with card data entry; * CARD\_BINDING – payment by binding; * CARD\_MOTO – payment via a call center; * CARD\_PRESENT – payment as cardPresent; * ALFA\_ALFACLICK – for payment using Alfa-Click (via the PayByClik system); * UPOP – for payment via the UPOP system, available for China UnionPay cardholders; * FILE\_BINDING – payment via file; * SMS\_BINDING – payment via sms; * P2P – transfer from card to card; * P2P\_BINDING – transfer by binding; * PAYPAL – payment from a PayPal account; * MTS – payment from an MTS account; * APPLE\_PAY – Apple Pay; * APPLE\_PAY\_BINDING – Apple Pay binding payment; * ANDROID\_PAY – Android Pay; * ANDROID\_PAY\_BINDING – payment with a binding of Android Pay; * GOOGLE\_PAY\_CARD – Google Pay non-tokenized; * GOOGLE\_PAY\_CARD\_BINDING – payment with a binding of a non-tokenized GooglePay card; * GOOGLE\_PAY\_TOKENIZED – Google Pay tokenized; * GOOGLE\_PAY\_TOKENIZED\_BINDING – payment with a binding of a tokenized GooglePay card; * SAMSUNG\_PAY – Samsung Pay; * SAMSUNG\_PAY\_BINDING – payment with a Samsung Pay binding; * IPOS – iPOs payment. | | 09 and above. |
|  | |  | |  |
|  |  | |  | |  |
| avsCode | A1 | | no | | AVS Response Code – the code of the AVS verification response (checking the address and postal code of the cardholder). Possible values are:   * A – postal code and address match; * B – address matches, postal code does not match; * C – postal code matches, address does not match; * D – postal code and address do not match; * E – data validation is requested, but the result is unsuccessful; * F – incorrect format of the AVS/AVV verification request. | | 19 and above |
| *The* merchantOrderParams *element is present in the response if the order contains additional parameters of the seller. Each additional order parameter is presented in a separate* merchantOrderParams *element.* | | | | | | | |
| name | AN..20 | | no | | Name of the additional parameter | | All versions. |
| value | AN..1024 | | no | | Value of the additional parameter | | All versions. |
| *The* cardAuthInfo *element contains a structure consisting of a list of the* secureAuthInfo *element and the following parameters:* | | | | | | | |
| pan | NS..19 | | no | | The masked number of the card that was used for payment. | | All versions. |
| expiration | N6 | | no | | Card expiration date in the YYYYMM format. It is specified only after payment of the order. | | All versions. |
| cardholderName | A..64 | | no | | Cardholder's name. It is specified only after payment of the order. | | All versions. |
| approvalCode | AN6 | | no | | Payment authorization code. This field has a fixed length (6 characters) and can contain digits and Latin letters. It is specified only after payment of the order. | | All versions. |
| chargeback | A..5 | | no | | Whether the funds were forcibly returned to the buyer by the bank. The following values are possible.   * **true** ; * **false**. | | 06 and above. |
| paymentSystem | N..10 | | yes | | Payment system name. The following options are available. VISA;MASTERCARD;AMEX;  * **JCB**; * **CUP**; * **MIR**. | | 08 and above. |
| product | AN..255 | | yes | | Additional information about corporate cards. This information is filled in by the technical support in the management console. If there is no such information, an empty value is returned. | | 08 and above. |
| productCategory | string | | yes | | Additional information about the category of corporate cards. This information is filled in by the technical support in the management console. If there is no such information, an empty value is returned.  Possible values: DEBIT, CREDIT, PREPAID, NON\_MASTERCARD, CHARGE, DIFFERED\_DEBIT. | | 17 |
| *The* secureAuthInfo *element (the element consists of an* eci *element and a*  threeDSInfo *element, which is a list of the* cavv *and* xid *parameters):* | | | | | | | |
| eci | N..4 | | no | | Electronic commercial indicator. It is specified only after the order payment and in case of appropriate permission. | | All versions. |
| cavv | ANS..200 | | no | | The value of checking the authentication of the cardholder. It is specified only after the order payment and in case of appropriate permission. | | All versions. |
| xid | ANS..80 | | no | | Electronic commercial transaction ID. It is specified only after the order payment and in case of appropriate permission. | | All versions. |
| *The* bindingInfo *element consists of the following parameters:* | | | | | | | |
| clientId | AN..255 | | no | | The client's number (ID) in the store's system, transmitted during the order registration. Present only if the store is allowed to create bindings. | All versions. | |
| bindingId | AN..255 | | no | | ID of the binding created when paying for the order or used for payment. Present only if the store is allowed to create bindings. | All versions. | |
| authDateTime | ANS | | no | | Date/time of authorization. | 02 and above. | |
| terminalId | AN..10 | | no | | Terminal ID. | 02 and above. | |
| *The* paymentAmountInfo *element consists of the following parameters:* | | | | | | | |
| approvedAmount | N..20 | | no | | The amount reserved on the card (used only for two-phase payments). | 03 and above. | |
| depositedAmount | N..20 | | no | | The amount confirmed for debiting from the card. | 03 and above. | |
| refundedAmount | N..20 | | no | | Refund amount. | 03 and above. | |
| paymentState | A..10 | | no | | Order status. | 03 and above. | |
| feeAmount | N..20 | | no | | Fee amount. | 11 and above. | |
| totalAmount | N..20 | | no | | Order amount + fee (fee if it was used in the order). | 18 and above. | |
| *The* bankInfo *element consists of the following parameters:* | | | | | | | |
| bankName | AN..200 | | no | | Name of the Issuing bank. | 03 and above. | |
| bankCountryCode | AN..4 | | no | | Country code of the issuing bank. | 03 and above. | |
| bankCountryName | AN..160 | | no | | Name of the issuing bank's country in the language passed in the language parameter in the request, or in the language of the user who called the method, if the language is not specified in the request. | 03 and above. | |
| The refunds block contains information on refunds. It is added in response to the getOrderStatusExtended request of 05 version and higher and is present only if there is a refund on the order.  The block parameters are as follows:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Name** | **Type** | **Mandatory** | **Description** | **getOrderStatusExtended version** | | referenceNumber | N12 | no | The reference number of the transaction assigned by the payment gateway after its completion. | 05 and above | | actionCode | N..5 | no | Response codes are a digital designation of the result that the user's access to the system led to. | 05 and above | | amount | N..12 | no | Refund amount in minimum currency units. | 05 and above | | date | ANS | no | Date of order registration in UNIX-time format (POSIX-time). | 05 and above | | | | | | | | |

##### Example of a POST request:

orderId=b9054496-c65a-4975-9418-

1051d101f1b9&language=ru&merchantOrderNumber=0784sse49d0s134567890

##### Response example:

{"errorCode":"0","errorMessage":"Успешно","orderNumber":"0784sse49d0s1 34567890","orderStatus":6,"actionCode":- 2007,"actionCodeDescription":"Session time has expired","amount":33000,"currency":"810","date":1383819429914,"orderDe scription":" ","merchantOrderParams":[{"name":"email","value":"yap"}],"attributes": [{"name":"mdOrder","value":"b9054496-c65a-4975-9418- 1051d101f1b9"}],"cardAuthInfo":{"expiration":"201912","cardholderName"

:"Ivan","secureAuthInfo":{"eci":6,"threeDSInfo":{"xid":"MDAwMDAwMDEzOD M4MTk0MzAzMjM="}},"pan":"411111\*\*1111"},"terminalId":"333333"}

* + 1. Request for Order Payment Completion (REST)

The deposit.do request is used to request the completion of a pre-authorized order.

This operation can be performed if there are appropriate rights in the system.

##### Request parameters:

| **Name** | **Type** | **Compulsoriness** | **Description** |
| --- | --- | --- | --- |
| userName | AN..30 | yes | The store's login received when connecting |
| password | AN..30 | yes | The store's password received when connecting |
| orderId | ANS36 | yes | The order number in the payment system. It is unique within the system. |
| amount | N..12 | yes | Payment amount in kopecks (or cents) |
| language | A2 | no | The language is encoded in ISO 639-1. If not specified, it is assumed that the language is Russian. The error message will be returned in this language. |
| If the amount parameter is set to zero, the completion is done for the whole pre-authorized amount. | | | |

##### Response parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Compulsoriness** | **Description** |
| errorCode | N3 | No | Error code. |
| errorMessage | AN..512 | No | Description of the error in the language. |

**Error codes (**ErrorCode **field):**

Classification:

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

Decoding:

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 5 | Access denied |
| 5 | The user must change password |
| 5 | Incorrect amount |
| 5 | The deposit amount must be equal to zero or at least one ruble |
| 6 | Invalid order number |
| 7 | The payment must be in the correct state |
| 7 | System error |

##### Example of a POST request:

amount=100&currency=810&language=ru&orderId=e5b59d3d-746b-4828-9da4- 06f126e01b68

##### Response example:

{"errorCode":0}

* + 1. Request for order payment cancellation (REST)

The reverse.do request is used to cancel the order payment. The cancellation function is available within a limited time after payment, the exact terms must be specified in the Bank.

The payment cancellation operation can be performed only once. If it ends with an error, the repeated payment cancellation operation will fail.

This function is available to stores in agreement with the Bank. To perform the cancellation operation, the user must have the appropriate rights.

##### Request parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Compulsoriness** | **Description** |
| userName | AN..30 | yes | The store's login received when connecting |
| password | AN..30 | yes | The store's password received when connecting |
| orderId | ANS36 | yes | The order number in the payment system. It is unique within the system. |
| language | A2 | no | The language is encoded in ISO 639-1. The error description is returned in this language. If the parameter is not available, the default language specified in the merchant's settings is used. |
| jsonParams | See description | no | The fields for additional information and its subsequent storage, such as {"param":value,"param2":value 2}. These fields can be passed to the Bank processing system to be subsequently displayed in the registries.\* This functionality can be enabled upon additional agreement with the bank during the integration period.  Data type   |  |  | | --- | --- | | **Name** | **Type** | | name | ANS..255 bytes | | value | ANS..1024 bytes | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Response parameters:** | |  |  |
| **Name** | **Type** | **Compulsoriness** | **Description** |
| errorCode | N3 | No | Error code. |
| errorMessage | AN..512 | No | Description of the error in the language. |

##### Error Codes (ErrorCode field):

Classification:

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

Decoding:

|  |  |
| --- | --- |
| **Meaning** | **Description** |
| 0 | Request processing took place without system errors |
| 5 | Access denied |
| 5 | The user must change password |
| 5 | [orderId] not set |
| 6 | Invalid order number |
| 7 | The operation is not possible for the current payment status |
| 7 | Reversal is impossible. Reason: incorrect internal values, check the amount of hold, deposit |
| 7 | System error |

##### Example of a POST request:

language=ru&orderId=9231a838-ac68-4a3e-bddb-d9781433d852

##### Response example:

{"errorCode":"0","errorMessage":"Успешно"}

* + 1. Request for a refund of the order payment (REST)

The refund.do request is used for the refund.

Upon this request, the funds for the specified order will be returned to the payer. The request will end with an error if the funds for this order have not been debited. The system enables to return funds more than once, but for a total amount not exceeding the initial debit amount.

|  |
| --- |
| When processing a refund for housing and utilities services, only a full refund is available. |

To process the refund operation, one must have the appropriate rights in the system.

##### Request parameters:

| **Name** | **Type** | **Compulsoriness** | **Description** |
| --- | --- | --- | --- |
| userName | AN..30 | yes | The store's login received when connecting |
| password | AN..30 | yes | The store's password received when connecting |
| orderId | ANS36 | yes | The order number in the payment system. It is unique within the system. |
| amount | N..12 | yes | Payment amount in kopecks (or cents) |
| language | A2 | no | The language is encoded in ISO 639-1. If not specified, it is assumed that the language is Russian. The error message will be returned in this language. |
| jsonParams | See description | no | The fields for additional information and its subsequent storage, such as {"param":value,"param2":value 2}. These fields can be passed to the Bank processing system to be subsequently displayed in the registries.\* This functionality can be enabled upon additional agreement with the bank during the integration period.  Data type   |  |  | | --- | --- | | **Name** | **Type** | | name | ANS..255 bytes | | value | ANS..1024 bytes | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Response parameters:** | |  |  |
| **Name** | **Type** | **Compulsoriness** | **Description** |
| errorCode | N3 | No | Error code. |
| errorMessage | AN..512 | No | Description of the error in the language. |

**Error codes (**ErrorCode **field):**

Classification:

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

Decoding:

| **Meaning** | **Description** |
| --- | --- |
| 0 | Request processing took place without system errors |
| 5 | Access denied |
| 5 | The user must change password |
| 5 | [orderId] not set |
| 6 | Invalid order number |
| 7 | The payment must be in the correct state |
| 7 | Incorrect deposit amount (less than one ruble) |
| 7 | System error |

##### Example of a POST request:

amount=500&currency=810&language=ru&orderId=5e97e3fd-1d20-4b4b-a542- f5995f5e8208

##### Response example:

{"errorCode":0}

* + 1. Card Verification Request (REST)

The verifyCard.do method is used to request a card verification.

The method performs verify operations. If the verify operation is successful, the order will be transferred to the CANCELED status.

##### Request parameters:

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| userName | AN..30 | no (either a login and password, or a token needs to be specified) | The store's login received when connecting. If an open token (token parameter) is used instead of login and password authentication, the userName parameter does not need to be passed. |
| password | AN..30 | no (either a login and password, or a token needs to be specified) | The store's password received when connecting. If an open token (token parameter) is used for authentication during registration instead of a login and password, the password parameter does not need to be passed. |
| token | ANS..30 | no (either a login and password, or a token needs to be specified) | A public key that can be used to register an order. If login and password are used for authentication during order registration, the token parameter does not need to be passed. |
| amount | N…3 | Yes | The amount of the debit in the order currency. Maximum value for card verification: 100 |
| currency | N..3 | No | The payment currency code is ISO 4217. If not specified, the default value is used. |
| pan | N12…19 | Yes | Card number. |
| cvc | N...3 | Yes | The CVC/CVV2 code is on the back of the card. |
| expiry | N6 | Yes | The year and month of expiration of the card in the following format:  YYYYMM. |
| cardholderName | A..200 | No | Cardholder's name. It is specified only after payment of the order. |
| backUrl | AN..512 | Yes | The address to which the user must be redirected in case of successful payment. The address must be specified in full, including the protocol used  (for example, [https://test.ru](https://test.ru/) instead of [test.ru](http://test.ru/)). Otherwise, the user will be redirected to the address of the following type: http://<payment\_gateway\_address>/<merchant\_address> |
| failUrl | AN…512 | No | The address to which the user must be redirected in case of failed payment. The address must be specified in full, including the protocol used  (for example, [**https://test.ru**](https://test.ru/) instead of [**test.ru**](http://test.ru/)). Otherwise, the user will be redirected to the address of the following  type: **http://<payment\_gateway\_address>/<merchant\_address>**. |
| description | ANS…598 | No | Description of the order in any form. |
| email | ANS..40 | No | Buyer's email address. |
| language | A2 | No | The language is encoded in ISO 639-1. If not specified, the default language specified in the store settings will be used. |
| threeDSServerTransId | ANS..36 | No | Transaction ID created on the 3DS server. |
|  |  |  | This parameter is used when authenticating a client using the 3DS protocol of 2.0 version. |
| threeDSVer2FinishUrl | AN..512 | No | Client forwarding address after authentication to the ACS server. |
|  |  |  | This parameter is used when authenticating a client using the 3DS protocol of 2.0 version. |
|  |  |  | This parameter should be passed to the gateway both in the first payment/transfer request and in the second one. |
| threeDSVer2MdOrder |  | No | The order number that was registered in the first part of the request as part of the 3DS 2.0 operation. |

##### Response parameters:

| **Name** | | **Type** | | **Mandatory** | | **Description** | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| errorCode | | N..1 | | Yes | | Error code. | |
| errorMessage | | AN..512 | | No | | Error description in the language passed in the language parameter in the request. | |
| orderId | | ANS36 | | No | | The order number in the payment system. It is unique within the system.  It is missing if the order registration failed due to an error detailed in ErrorCode. | |
| orderNumber | | ANS..36 | | No | | The order number (ID) in the store's system. | |
| authCode | | AN6 | | No | | Payment authorization code, can contain digits and Latin letters. | |
| actionCode | | N..5 | | No | | Response codes are a digital designation of the result that the user's access to the system led to. | |
| actionCodeDescription | | AN..512 | | No | | Decryption of the response code in the language passed in the language request parameter. | |
| time | |  | | No | | Time of the operation. | |
| eci | | N..4 | | No | | Electronic commercial indicator. | |
| amount | | N..12 | | No | | Payment amount in minimum currency units. | |
| currency | | N..3 | | No | | The payment currency code is ISO 4217. If not specified, the default value is used. | |
| rrn | | AN12 | | No | | Unique identification number of the operation assigned after performing this operation. | |
| acsUrl | | AN..512 | | No | | The redirect address to the ACS server address. | |
| termUrl | | AN..512 | | No | | The payer's redirect link after authentication. | |
| paReq | | AN..512 | | No | | Request for payer authentication. | |
| userMessage | | ANS...512 | | No | | Message to the user with a description of the result code. | |

##### Request example:

curl --request POST \

--url https://all.test.com/rest/verifyCard.do \

--header 'content-type: application/x-www-form-urlencoded' \

--data userName=IK24-test-api \

--data password=IK24-test-api \

--data pan=4111111111111111 \

--data cvc=662 \

--data expiry=202412

##### Response example:

{

"errorCode": "0", "errorMessage": "Success",

"orderId": "cfc238ca-68f9-745c-ba7e-eb9100af79e0", "orderNumber": "12017",

"rrn": "111111111115",

"authCode": "123456",

"actionCode": 0, "actionCodeDescription": "", "time": 1595284781180,

"eci": "07",

"amount": 0,

"currency": "170"

}

* 1. Proxy interface for payment by installments (REST)
     1. Request to check the possibility of payment by installments (REST)

The check request is used to check the possibility of paying by installments using the card's BIN code (see [Connection coordinates](#_bookmark8)).

##### Request parameters

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Mandatory** | **Description** |
| bin | N15..19 | It is mandatory to forward bin or a pair of bindingId and modOrder | The client's bank card number. |
| bindingId | AN..255 | It is mandatory to forward bin or  a pair of bindingId and  modOrder | ID of the binding created earlier. It can only be used if the store has permission to work with bindings. If this parameter is passed in this request, it means that:   1. This order can only be paid using a binding; 2. The payer will be redirected to the payment page, where only CVC input is required. |
| mdOrder | ANS..36 | It is mandatory to forward bin or  a pair of bindingId and  modOrder | The order number in the payment gateway system. |

##### Response parameters

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| success | boolean | yes | Request processed successfully or unsuccessfully:   * true - the request was processed successfully; * false - an error occurred while processing the request. |
| checkResult | boolean | no | The result of checking the card's BIN code is the possibility of payment by installments:   * true - this card can be used to make an installment payment; * false - no ability to pay by installments. |
| description | ANS..512 | yes | A message in response to a request. If the request contains an error, the error description will be returned in this parameter. |
| paymentSystemIndicator | A..4 | no | The payment system used by the client, the following options are available:   * VISA – VISA payment system;   MC – MasterCard payment system. |

**Request example**

{

"bin": "411111111111"

}

##### Response example

{

"success": **true**,

"description": "Bin present in the allowed ranges.", "checkResult": **true**

"paymentSystemIndicator": "VISA"

}

* + 1. Request for information about installment payment options (REST)

The get request is used to get information about options of payment by installments.

##### Request parameters

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| currencyCode | A3 | yes | The payment currency code is ISO 4217. |
| transactionAmount | N..12 | yes | The initial payment amount (excluding installments) in minimum currency units. |
| primaryAccountNumber | N..19 | yes, if bindingId is not specified | Client's card number |
| mdOrder | ANS..36 | yes | The order number in the payment gateway system. |
| bindingId | AN..255 | no | ID of the binding created earlier. It can only be used if the store has permission to work with bindings. |
|  |  |  | If bindingId is passed, then the primaryAccountNumber parameter passing is optional. |

##### Response parameters

| **Name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| success | boolean | yes | Request processed successfully or unsuccessfully:   * true - the request was processed successfully; * false - an error occurred while processing the request. |
| description | ANS..512 | yes | A message in response to a request. If the request contains an error, the error description will be returned in this parameter. |
| receiptFreeText | ANS..512 | no | Information message of the issuer to the client, preferred to display after the installment payment. |
| mcInstallmentsProposals – a block of installment payment parameters for MasterCard system users. The response may contain several such blocks. If an invalid request was sent (for example, the mdOrder is specified incorrectly), the block will return empty.  The content of the block is presented below. | | | |
| installmentsCount | N..2 | Yes if  the mcInstallmentsProposals block  returned not empty | Maximum number of contributions. |
| firstInstallmentAmount | N..12 | Yes if  the mcInstallmentsProposals block  returned not empty | The amount of the first payment in minimum currency units. |
| subSeqInstallmentAmount | N..12 | Yes if  the mcInstallmentsProposals block  returned not empty | The amount of subsequent installment payments. |
| transactionAmount | N..12 | Yes if  the mcInstallmentsProposals block  returned not empty | The initial payment amount (excluding installments) in minimum currency units. |
| totalAmount | N..12 | Yes if  the mcInstallmentsProposals block  returned not empty | The total payment amount taking into account the surcharge for installments. |
| currencyCode | A3 | Yes if  the mcInstallmentsProposals block  returned not empty | The payment currency code is ISO 4217. |
| annualRate | N..3 | Yes if  the mcInstallmentsProposals block  returned not empty | The annual rate charged for borrowing and expressed as one percentage number, which represents the actual annual cost of funds during the term of payment of the contribution. This includes any handling fees or additional costs associated with the operation. |
| interestRate | N..3 | Yes if  the mcInstallmentsProposals block  returned not empty | Annual interest rate (loan charge as a percentage of the loan amount). |
| installmentFee | N..3 | Yes if  the mcInstallmentsProposals block  returned not empty | The amount of commission interest for installments.  Expressed as a percentage number.{{ |
| visaIstallmentsProposals - a block of installment payment parameters for VISA system users. The response may contain several such blocks. If an invalid request was sent (for example, the mdOrder is specified incorrectly), the block will return empty.  The content of the block is presented below. | | | |
| installmentFrequency | ANS...9 | Yes if  the visaInstallmentsProposals block returned not empty | Frequency of installment payments. List of possible values:   * WEEKLY – weekly; * BI-WEEKLY – biweekly; * MONTHLY – monthly. |
| installmentsCount | N..2 | Yes if  the visaInstallmentsProposals block returned not empty | Maximum number of contributions. |
| lastInstallmentAmount | N...12 | Yes if  the visaInstallmentsProposals block returned not empty | The amount of the last payment in minimum currency units. |
| regularInstallmentAmount | N..12 | Yes if  the visaInstallmentsProposals block returned not empty | The amount of the planned installment payment. |
| transactionAmount | N..12 | Yes if  the visaInstallmentsProposals block returned not empty | The initial payment amount (excluding installments) in minimum currency units. |
| totalPlanCost | N..12 | Yes if  the visaInstallmentsProposals block returned not empty | The total amount of the installment plan. Includes the transaction amount taking into account all handling fees within the installment plan. |
| totalFixedFeeAmount | N...12 | Yes if  the visaInstallmentsProposals block returned not empty | The amount of the final fixed installment fee. |
| totalVariableFeeAmount | N...12 | Yes if  the visaInstallmentsProposals block returned not empty | The amount of the final fee on interest within the installment plan. |
| currencyCode | A3 | Yes if  the visaInstallmentsProposals block returned not empty | The payment currency code is ISO 4217. |
| termsAndConditions | ANS...512 | Yes if  the visaInstallmentsProposals block returned not empty | Installment terms and conditions. |
| annualRate | N..3 | Yes if  the visaInstallmentsProposals block returned not empty | The annual rate charged for borrowing and expressed as one percentage number, which represents the actual annual cost of funds during the term of payment of the contribution. This includes any handling fees or additional costs associated with the operation. |
| visaIstallmentsProposals.installmentFees - a block of fees for installments under the VISA program.  The content of the block is presented below. | | | |
| feeAmount | N..12 | Yes if  the visaInstallmentsProposals block returned not empty | The amount of fee (for the corresponding feeType) charged from the total amount. |
| feeRate | N..3 | Yes if  the visaInstallmentsProposals block returned not empty | The percentage of fee (for the corresponding feeType) charged from the total amount. |
| feeType | ANS...20 | Yes if  the visaInstallmentsProposals block returned not empty | Type of fee charged.  Possible list of values (does not imply further use):   * CONSUMER (charged from the client); * ISSUER\_TO\_ACQUIRER (charged from the issuer in favor of the acquirer); * ACQUIRER\_TO\_ISSUER (charged from the acquirer in favor of the issuer); * ISSUER\_TO\_VISA (paid by the issuer in favor of VISA); * ACQUIRER\_TO\_VISA (charged from the acquirer in favor of VISA). |

##### Request example

{

"currencyCode": "USD", "transactionAmount": 70000,

"primaryAccountNumber": "5570309355581111", "mdOrder": "e7f6930f-e7a2-78a4-9d60-307e0007a509"

}

##### Response example

{

"success": **true**,

"description": "Installments calculated successfully.", "receiptFreeText": "YOU CHOOSE THE INSTALMENT PROGRAMME.",

"installmentsProposals": [{ "installmentsCount": 2,

"firstInstallmentAmount": 35056,

"subSeqInstallmentAmount": 35056,

"transactionAmount": 70000,

"totalAmount": 70112, "currencyCode": "USD", "annualRate": 98,

"interestRate": 100,

"installmentFee": 0

}, {

"installmentsCount": 4,

"firstInstallmentAmount": 17564,

"subSeqInstallmentAmount": 17564,

"transactionAmount": 70000,

"totalAmount": 70256, "currencyCode": "USD", "annualRate": 149,

"interestRate": 150,

"installmentFee": 0

}, {

"installmentsCount": 6,

"firstInstallmentAmount": 11743,

"subSeqInstallmentAmount": 11743,

"transactionAmount": 70000,

"totalAmount": 70458, "currencyCode": "USD", "annualRate": 200,

"interestRate": 200,

"installmentFee": 0

}, {

"installmentsCount": 9,

"firstInstallmentAmount": 11977,

"subSeqInstallmentAmount": 7777,

"transactionAmount": 70000,

"totalAmount": 74193, "currencyCode": "USD", "annualRate": 1439,

"interestRate": 0,

"installmentFee": 4200

}, {

"installmentsCount": 12,

"firstInstallmentAmount": 6212,

"subSeqInstallmentAmount": 6212,

"transactionAmount": 70000,

"totalAmount": 74544, "currencyCode": "USD", "annualRate": 1155,

"interestRate": 0,

"installmentFee": 4550

}

]

}

* + 1. Request for information about the installment plan status (REST)

The getInstallmentStatus request is used to get the installment status.

##### Request parameters

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Type** | **Mandatory** | **Description** |
| mdOrder | ANS..36 | Yes | The order number in the payment gateway system. |

Response parameters

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Type** | **Mandatory** | **Description** |
| success | A..5 | Yes | Indicates the request success. The following values are available:   * true – the request was processed successfully; * false – the request failed. |
| description | ANS..598 | No | Description of the returned status. |
| installmentStatus | A..9 | No | Installment statuses. The following options are possible:   * REQUESTED – a request for the provision of installments has been executed; * DECLINED – the provision of installments is rejected; * PROCESSED – the request for the provision of installments has been processed; * REVERSED – funds for the order are returned to the client; |

**Request example**

{

"mdOrder": "1234567890"

}

##### Response example

{

"description": "Request processed", "installmentStatus": "PROCESSED", "success": **true**

}

1. Test cards

As the Cardholder name (name of the cardholder), specify at least 2 words in the English layout.

For all cards enroled in 3-D Secure (**veres=y, pares=y or a**) ACS password: 12345678.

|  |
| --- |
| The test environment is not intended for load testing - if it is necessary to conduct such testing, the RSO should contact the Bank. |

Test cards:

**pan: 4111 1111 1111 1111**

**exp date: 2024/12 cvv2: 123**

**3dsecure: veres=y, pares=y**

**pan: 5100 0000 0000 0008**

**exp date: 2024/12 cvv2: 123**

**3dsecure: veres=y, pares=y**

**pan: 6011 0000 0000 0004**

**exp date: 2024/12 cvv2: 123**

**3dsecure: veres=y, pares=y**

**pan: 6390 0200 0000 000003**

**exp date: 2024/12**

**cvv2: 123(optional parameter) 3dsecure: veres=y, pares=a**

**pan: 5555 5555 5555 5599**

**exp date: 2024/12 cvv2: 123 3dsecure: veres=n**

**pan: 4444 0000 0000 1111**

**exp date: 2024/12 cvv2: 123 3dsecure: veres=n**

**pan: 2200 0000 0000 0004**

**exp date: 2024/12**

**cvc: 123**

**3dsecure: veres=n**

**pan: 2200 0000 0000 0012**

**exp date: 2024/12 cvc: 123**

**3dsecure: veres=y pares=n**

**pan: 2200 0000 0000 0020**

**exp date: 2024/12 cvc: 123 3dsecure: veres=u**

**pan: 2200 0000 0000 0038**

**exp date: 2024/12 cvc: 123**

**3dsecure: veres=y pares=u**

**pan: 2200 0000 0000 0046**

**exp date: 2024/12 cvc: 123**

**3dsecure: veres=y pares=a**

**pan: 2200 0000 0000 0053**

**exp date: 2024/12 cvc: 123**

**3dsecure: veres=y pares=y**

**pan: 2200 0000 0000 0053**

**exp date: 2024/12 cvv2: 123**

**3dsecure: VeRes=Y, PaRes=Y**

**pan: 2200 0000 0000 0046**

**exp date: 2024/12 cvv2: 123**

**3dsecure: VeRes=Y, PaRes=A**

**pan: 2200 0000 0000 0012**

**exp date: 2024/12 cvv2: 123**

**3dsecure: VeRes=Y, PaRes=N**

**pan: 2200 0000 0000 0038**

**exp date: 2024/12 cvv2: 123**

**3dsecure: veres=y, pares=u**

**pan: 2200 0000 0000 0020**

**exp date: 2024/12 cvv2:** 123 **3dsecure: VeRes=U**

**pan: 2200 0000 0000 0004**

**exp date: 2024/12 cvv2: 123 3dsecure: VeRes=N**

Cards returning errors:

**pan: 5555 5555 5555 5557**

**exp date: 2024/12 cvv2: 123**

**3dsecure: veres=y, pares=u**

**pan: 4444 3333 2222 1111**

**exp date: 2024/12 cvv2: 123**

**3dsecure: veres=y, pares=u**

Declined. PaRes status is U (-2011)

**pan: 4000 0000 0000 0002**

**exp date: 2024/12 cvv2: 123 3dsecure: veres=u**

**pan: 5555 5555 4444 4442**

**exp date: 2024/12 cvv2: 123 3dsecure: veres=u**

Declined. VeRes status is U (-2016)

**pan: 4444 4444 4444 4422**

**exp date: 2024/12 cvv2: 123**

Invalid message format (913)

**pan: 4444 4444 4444 4455**

**exp date: 2024/12 cvv2: 123**

Card limitations exceeded (902)

**pan: 4444 4444 4444 3333**

**exp date: 2024/12 cvv2: 123**

Limit exceeded (123)

**pan: 4444 4444 4444 6666**

**exp date: 2024/12 cvv2: 123**

BLOCKED\_BY\_LIMIT (-20010)

**pan: 4444 4444 1111 1111**

**exp date: 2024/12 cvv2: 123**

Network refused transaction (5)

**pan: 4444 4444 9999 9999**

**exp date: 2024/12 cvv2: 123**

TDSEC\_COMM\_ERROR (151017)

**pan: 5432 5432 5432 5430**

**exp date: 2024/12 cvv2: 521**

INSUFFICIENT\_FUNDS (116)

Below are the cards that support the 3-D Secure protocol of 2.0 version

**PAN**: **2201382000000021**

**ECI**: **02**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **C - additional authentication with CReq/CRes is required**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **Y - authentication/verification was successful** . **TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **1 - yes**.

**AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0 - the code must be entered** .

**PAN**: **2201382000000039**

**ECI**: **02**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **C - additional authentication with CReq/CRes is required**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **Y - authentication/verification was successful**. **TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **1 - yes**.

**AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0 - the code must be entered** .

**PAN**: **4175003806480112**

**ECI**: **06**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **N - not authenticated/verified, transaction cancellation**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **N - not authenticated/verified, transaction cancellation**.

**TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **1 - yes**. **AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0 - the code must be entered** .

**PAN**: **5100000000000123**

**ECI**: **05**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **Y - authentication/verification was successful**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **Y - authentication/verification was successful**. **TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **0 - no** .

**AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0 - N/A (3-D Secure method is disabled)** .

**PAN**: **5555555555555599 ECI**: **N/A**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **C - additional authentication with CReq/CRes is required**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **Y - authentication/verification was successful**. **TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **0 - no** .

**AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0 - N/A (3-D Secure method is disabled)** .

**PAN**: **5100000000000180**

**ECI**: **05**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **Y - authentication/verification was successful**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **Y - authentication/verification was successful**. **TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **1 - yes**.

**AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0 - N/A (3-D Secure method is disabled)** .

**PAN**: **5213243734062662**

**ECI**: **02**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **C - additional authentication with CReq/CRes is required**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **Y - authentication/verification was successful**. **TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **1 - yes**.

**AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0 - the code must be entered** .

**PAN**: **4111111111111111**

**ECI**: **05**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **Y - authentication/verification was successful**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **Y - authentication/verification was successful**. **TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **1 - yes**.

**AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0**

**- N/A (3-D Secure method is disabled)**.

**PAN**: **5100000000000008**

**ECI**: **05**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **Y - authentication/verification was successful**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **Y - authentication/verification was successful**. **TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **1 - yes**.

**AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0 - N/A (3-D Secure method is disabled)** .

**PAN**: **2201382000000013**

**ECI**: **02**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **C - additional authentication with CReq/CRes is required**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **Y - authentication/verification was successful**. **TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **1 - yes**.

**AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0 - the code must be entered**.

**PAN**: **2201382000000047**

**ECI**: **02**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **C - additional authentication with CReq/CRes is required**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **Y - authentication/verification was successful**. **TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **1 - yes**.

**AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0 - the code must be entered**.

**PAN**: **2201382000000054**

**ECI**: **02**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **C - additional authentication with CReq/CRes is required**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **Y - authentication/verification was successful**. **TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **1 - yes**.

**AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0 - the code must be entered**.

|  |
| --- |
| payment with this card will be rejected due to insufficient funds on the card |

**PAN**: **5559573111111183**

**ECI**: **05**

**CVC**: **123**

**Eхp date**: **2024/12**

**ARES\_TRANS\_STATUS** (authentication request): **Y - authentication/verification was successful**.

**RREQ\_TRANS\_STATUS** (response-confirmation of the request receipt with the results from the 3DS Requestor / 3DS Server): **Y - authentication/verification was successful**. **TDS\_METHOD\_ENABLED** (is the 3-D Secured authentication method enabled): **1 - yes**.

**AUTO\_ACS\_CHALLENGE** (redirect to the finish page without checking the SMS code): **0 - N/A (3-D Secure method is disabled)**.

1. Appendix 1. Response codes - decryption of actionCode (processing response)

The response code is a digital designation of the result that the user's operation in the system led to. The following codes are defined in the system.

| **Action code** | **error\_i d** | **error\_message** | **Description** | **Messages recommended for the Client** |
| --- | --- | --- | --- | --- |
| -20010 | -20010 | Blocked by limit | The transaction was rejected due to the fact that the payment amount exceeded the limits set by the Issuing Bank | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons.  The bank's phone number must be indicated on the back of the card. |
| -9000 | -9000 | Started | Transaction start status | *When this actionCode occurs, the message is not displayed to the Client.* |
| -3003 | -3003 | Unknown | Unknown | Payment error. Try again later. |
| -2102 | -2102 | Blocked by passenger name | Blocked by passenger name | Payment error. Contact the store's representative for the reasons. |
| -2101 | -2101 | Blocked by e-mail | Blocked by e-mail | Payment error. Contact the store's representative for the reasons. |
| -2025 | -2025 | Rejected. RReq status is not "Y" | Rejected. RReq status is not "Y" | Payment error. Contact the store's representative for the reasons. |
| -2024 | -2024 | Rejected. Partial 3D Secure (v.2) is prohibited | Rejected. Partial 3D Secure (v.2) is prohibited | Payment error. Contact the store's representative for the reasons. |
| -2023 | -2023 | "U" in ARes | "U" in ARes | Payment error. Contact the store's representative for the reasons. |
| -2020 | -2020 | Invalid ECI received | Invalid ECI received. The code is set if the ECI that came to PaRes does not match the acceptable value for this IPS. The rule only works for MasterCard (01.02) and  Visa (05.06), where the values in parentheses are valid for IPS. | Payment error. Try again later. |
| -2019 | -2019 | Rejected as per iReq in PARes | PARes from the issuer contains iReq, as a result of which the payment was rejected | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons.  The bank's phone number must be indicated on the back of the card. |
| -2018 | -2018 | Rejected. DS connection timeout | The Visa or MasterCard Directory server is either unavailable or a communication error has occurred in response to the card  enrolment (VeReq) request. This is an error in the interaction of the payment gateway and the IPS servers due to  technical problems on the side of the latter. | Payment error. Try again later. |
| -2017 | -2017 | Rejected. PARes status is not "Y" | Rejected. PARes status is not "Y" | Payment error. Contact the store's representative for the reasons. |
| -2016 | -2016 | U in VERes | The issuing bank could not determine whether the card is 3dsecure. | Payment error. Try again later. If this error occurs again, contact your bank for the reasons. The bank's phone number must be indicated on the back of the card. |
|  |  |  |  |
| -2015 | -2015 | Rejected as per iReq in VERes | VERes from DS contains iReq, as a result of which the payment was rejected. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| -2013 | -2013 | Payment attempts exhausted | Payment attempts exhausted. | Payment error. Try again later. |
| -2012 | -2012 | The operation is not supported | This operation is not supported. | Payment error. Try again later. |
| -2011 | -2011 | U in PARes | The issuing bank was unable to authorize the card's 3dsecure | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons.  The bank's phone number must be indicated on the back of the card. |
| -2010 | -2010 | XID mismatch | XID mismatch. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| -2008 | -2008 | Invalid wallet | Invalid wallet. | *This actionCode is no longer used.* |
| -2007 | 2007 | Session time has expired | The period for entering card data from the moment of payment registration has expired (the default timeout is 20 minutes; the session timeout can be specified during order registration or in the seller's settings). | Payment error. Try again later. |
| -2006 | 2006 | 3DS authorization failed | It means that the issuer rejected the authentication (3DS authorization failed) | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons.  The bank's phone number must be indicated on the back of the card. |
| -2005 | 2005 | 3DS signature verification failed | It means that we could not verify the issuer's signature, that is, PARes was readable, but signed incorrectly. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| -2003 | -2003 | Blocked by port | Blocking by port. | *This actionCode is no longer used.* |
| -2002 | 2002 | Blocked by amount | The transaction was rejected due to the fact that the payment amount exceeded the established limits.  Note: this means the Acquiring Bank's limits on the daily turnover of the Store, or the Store's limits on turnover | Payment error. Contact the store's representative for the reasons. |
|  |  |  | on one card, or the Store's limit on one operation |  |
| -2001 | 2001 | Blocked by IP | The transaction was rejected because the Client's IP address was blacklisted. | Payment error. Contact the store's representative for the reasons. |
| -2000 | 2000 | Blocked by card number | The transaction was rejected because the card was blacklisted. | Payment error. Contact the store's representative for the reasons. |
| -102 | -102 | Payment declined by the payment agent | Payment declined by the payment agent. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| -100 | -100 | There were no attempts to pay. | There were no attempts to pay. | *When this actionCode occurs, the message is not displayed to the Client.* |
| -1 | -1 | Processing is not available | The waiting period for a response from the processing system has expired. | Payment error. Try again later. |
| 0 | 0 | Request successfully processed | The payment was successful. | Successful transaction. |
| 1 | 1 | Declined. Proof of identity required | Proof of identity is required to complete the transaction successfully. In the case of an Internet transaction (as in our case), it is impossible, therefore it is considered as declined. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons.  The bank's phone number must be indicated on the back of the card. |
| 4 | 4 | Restriction of card usage | Restriction of card usage. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 5 | 5 | Processing the transaction is rejected by the network. | Processing the transaction is rejected by the network. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 15 | 15 | IPS could not identify the card issuer | IPS could not identify the card issuer. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 53 | 53 | Invalid account | The card does not exist in processing systems. | The operation was rejected. Contact the store. |
| 58 | 58 | Restrictions on the terminal | Transaction on this card is prohibited in the device. | Payment error. Try again later. If this error occurs again, contact the store's representative to find out what to do next. |
| 62 | 62 | Blocked card | The card is blocked. It is necessary to contact the issuing bank. | Check that the card data is entered correctly. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 100 | 100 | Card restriction. | Card restriction (the issuing bank has banned Internet transactions on the card). | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 101 | 101 | Incorrect card expiration date. | Incorrect card expiration date. | Check that the card data is entered correctly. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 103 | 103 | Contact the issuing bank. | No connection with the Issuing Bank. The outlet needs to contact the issuing bank. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 104 | 104 | Restriction on the card account | An attempt to perform an operation on an account that has restrictions on its use. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 106 | 106 | The allowed number of attempts to enter the PIN has been exceeded.  The card is probably temporarily blocked. | The allowed number of attempts to enter the PIN has been exceeded.  The card is probably temporarily blocked. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 109 | 109 | Incorrect merchant/terminal ID | The merchant/terminal ID is specified incorrectly or the ACC is blocked at the level of  processing. | Payment error. Try again later. If this error occurs again, contact the store's representative to find out what to do next. |
| 111 | 111 | Invalid card number | Invalid card number. | Check that the card data is entered correctly. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 116 | 116 | Insufficient funds. | The transaction amount exceeds the available balance of the selected account. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 119 | 119 | Decline. SECURITY\_VIOLATIO  N from SV | The transaction is illegal. | The operation was rejected. Contact the bank that issued the card. |
| 119  (the source code was 93, mapped as **119**) | 119 | Terminal limit | The merchant's acquiring limit has been used up. | The operation was rejected. Contact the bank that issued the card. |
| 120 | 120 | Transaction was rejected by the issuing bank. | Refusal to conduct an operation - the transaction is not authorized by the issuer. The response code of the payment network is 57 or 93. Reasons for refusal should be  clarified with the issuer. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 121 | 121 | Available limit exceeded. | An attempt has been made to execute a transaction for an amount exceeding the daily limit set by the issuing bank. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 123 | 123 | Available limit exceeded. | The limit on the number of transactions was exceeded: the client has completed the maximum allowed number of transactions within the limit cycle and is trying to conduct another one. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 125 | 125 | Invalid card number. | Invalid card number. Such an error can mean a number of things:  An attempt to return an amount greater than the hold, an attempt to return a zero amount. For AmEx, the card expiration date is incorrect | Check that the card data is entered correctly. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 208 | 208 | The card is lost | The card is lost. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 209 | 209 | Card restriction. | Card limits are exceeded. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 261 | 261 | Available limit exceeded | An attempt has been made to execute a transaction exceeding the limit set by the acquiring bank. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 400 | 400 | Reversal processed. | Reversal processed. | The payment has been cancelled. |
| 433 | 433 | Suspicious reversal | Suspicious reversal | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 434 | 434 | The response was received after the reversal | The response was received after the reversal | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 435 | 435 | There is no such response code from the network | There is no such response code from the network | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 902 | 902 | Card restriction. | Card restriction (The cardholder is trying to execute a transaction that is not allowed for them). | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 903 | 903 | Available limit exceeded. | An attempt has been made to execute a transaction for an amount exceeding the limit set by the issuing bank. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 904 | 904 | Invalid message format | Incorrect message format from the point of view of the issuing bank. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 907 | 907 | No connection with the bank. | There is no connection with the Bank that issued your card. Authorization in stand-in mode is not allowed for this card number (this mode means that the issuer cannot contact the payment network and therefore the transaction is possible offline with subsequent unloading to the back office, or it will be rejected). | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 909 | 909 | It is impossible to perform the operation. | It is impossible to perform the operation (A general error in the functioning of the system. It shall be recorded by the payment network or the issuing bank). | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 910 | 910 | No connection with the bank. | The issuing bank is unavailable. | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 913 | 913 | Invalid message format. | Incorrect message format (Incorrect transaction format  in terms of the network). | Payment error. Try again later. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 914 | 914 | The original transaction was not found | Transaction not found (when completion or reversal or refund is sent). | *When this actionCode occurs, the message is not displayed to the Client*. |
| 999 | 999 | Suspicion of fraud. | There is no beginning of transaction authorization.  Rejected by fraud or 3dsec error. | Payment error. Try again later. If this error occurs again, contact the representative of store to find out what to do next. |
|  |  |  | After receiving this response code, further attempts to make a payment are rejected. |  |
| 1001 | 1001 | Empty | Empty (It is set at the time of registration of the transaction, i.e. when no card data has been entered for the transaction yet). | *When this actionCode occurs, the message is not displayed to the Client.* |
| 1111 | 1111 | The number of card requests is exceeded. | Response code for orders rejected due to exceeding the number of attempts to pay by card. | The operation was rejected. Contact the store. |
| 2001 | 2001 | Fraud | Fraudulent transaction (according to the processing system or the payment  network). | Payment error. Try again later. If this error occurs again, contact the representative of store to find out what to do next. |
| 2002 | 2002 | Invalid operation | Invalid operation. | Payment error. Try again later. If this error occurs again, contact the representative of store to find out what to do next. |
| 2003 | 2003 | SSL is forbidden | SSL (Not 3d-  Secure/SecureCode) transactions are forbidden to the Store. | Payment error. Try again later. If this error occurs again, contact the store's representative to find out what to do next. |
| 2004 | 2004 | SSL without CVC is forbidden | Payment via SSL without entering CVC2 is forbidden. | Payment error. Try again later. If this error occurs again, contact the representative of store to find out what to do next. |
| 2005 | 2005 | The order does not comply with the 3DS rule | The payment does not meet the conditions of the 3ds verification rule. | Payment error. Try again later. If this error occurs again, contact the representative of store to find out what to do next. |
| 2006 | 2006 | One-phase payments are forbidden | One-phase payments are forbidden. | Payment error. Try again later. If this error occurs again, contact the representative of store to find out what to do next. |
| 2007 | 2007 | The order has already been paid | The order has already been paid. | The order has already been paid.. |
| 2008 | 2008 | The transaction has not been completed yet | The transaction has not been completed yet. | *When this actionCode occurs, the message is not displayed to the Client.* |
| 2009 | 2009 | The refund amount exceeds the payment amount | The refund amount exceeds the payment amount. | The refund amount exceeds the payment amount. |
| 2014 | 2014 | 3DS rule execution error | 3DS rule execution error | Payment error. Try again later. If this error occurs again, contact the store's representative to find out what to do next. |
| 2015 | 2015 | Error in executing the terminal selection rule | Error in executing the terminal selection rule (the rule is incorrect). | Payment error. Contact the store's representative to find out what to do next. |
| 2016 | 2016 | 3DS is forbidden | The merchant does not have a 3-D Secure authorization required to make the payment. | Payment error. Contact the store's representative to find out what to do next. |
| 2023 | 2023 | Thread limit is exceeded | The queue of requests to be processed has exceeded the allowed limit. | Payment error. Try again later. |
| 4005 | 4005 | Rejected by the seller | The order was rejected by the seller. | Rejected by the seller. |
| 9001 | 9001 | Internal IPS  error | Internal IPS failure code. | Payment error. Try again later. |
| 71015 | 1015 | The information entered is incorrect | Incorrect card parameters have been entered. | Check that the card data is entered correctly. If this error occurs again, contact your Bank to clarify the reasons. The bank's phone number must be indicated on the back of the card. |
| 151017 | 1017 | Decline. 3DSec comm error | 3-D Secure - communication error. | Payment error. Try again later. If this error occurs again, contact the store's representative to find out what to do next. |
| 151018 | 018 | Decline. Processing timeout | Processing timeout. Failed to send. | Payment error. Try again later. If this error occurs again, contact the store's representative to find out what to do next. |
| 151019 | 1019 | Decline. Processing timeout | Processing timeout. Sending is success, but no response was received from the bank. | Payment error. Try again later. If this error occurs again, contact the store's representative to find out what to do next. |
| 341014 | 1014 | Decline. General Error | IPS failure code. | Payment error. Try again later. If this error occurs again, contact the representative of store to find out what to do next. |