Интернет заказ

**Функциональная спецификация API модуля**

**A-TRADE**

Оглавление

[Цель документа 2](#_Toc121849468)

[Описание 2](#_Toc121849469)

[Аутентификация 2](#_Toc121849470)

[Загрузка прайс-листа 3](#_Toc121849471)

[Получение резерва 6](#_Toc121849472)

[Подтверждение резерва 7](#_Toc121849473)

[Получение заказов для формирования накладной 8](#_Toc121849474)

[Получение сформированной накладной 9](#_Toc121849475)

[Получение справочника складов для отгрузки 11](#_Toc121849476)

# Цель документа

Для оперативного создания и изменения заказов, а также получения информации об их состоянии необходимо настроить взаимодействие между учетными системами A-TRADE (AT) и поставщиков. В предлагаемом документе описан порядок настроек таких систем с помощью методов API.

# Описание

Обмен данными производится посредством взаимодействия веб-сервисов с помощью XML или JSON. Пользователь выбирает формат возвращаемых данных при вызове функции API. Сервис работает с помощью REST, через который реализован удаленный вызов процедур для предоставления информации о состоянии заказов, предоставления прайс-листов. Работа сервиса осуществляется по протоколу HTTP.

Структура запроса:

<http://domain/ltapi/format/method>

где:

domain - адрес сервера,

format - тип передаваемых данных: json или xml, в случае если будет передан иной формат, сервер вернёт ошибку http code 500

method - название метода вызова

например:

<http://domain/ltapi/json/method>

<http://domain/ltapi/xml/method>

Данные передаются методом POST

Коды ответа REST

При любых ответах сервер вернёт http код 200, за исключением:

400 Bad Request («неправильный, некорректный запрос»)

401 Unauthorized («не авторизован»)

500 Internal Server Error («внутренняя ошибка сервера»)

# Аутентификация

Имя метода: auth

Тип запроса: POST

***Входные параметры:*** *логин + пароль*

Вход осуществляется с помощью выданного Поставщику логина и пароля

http://komus-lt-test.a-trade.io/ltapi/xml/auth?login=#AuthUser#&password=#AuthPass#

где:

#AuthUser# = логин

#AuthPass# = пароль

***Выходные параметры (json/xml)***

В ответ сервер возвращает уникальный идентификатор сессии token-ключ, который в дальнейшем необходимо передавать в каждом запросе через заголовок X-Token.

Срок действия token-ключа равен 2 часам с последнего запроса. По истечении времени сессии, сервер вернёт код 401

status:401 = Unauthorized. Необходимо получить новый токен

*Пример структуры возвращаемых данных XML*

<?xml version="1.0" encoding="utf-8"?>

<Response>

<token>e3cd229f9b5c4c5ea96cfca5dc92c797</token>

</Response>

*Пример структуры возвращаемых данных JSON*

{"token":" e3cd229f9b5c4c5ea96cfca5dc92c797"}

# Загрузка прайс-листа

Имя метода: SetPrice

Тип запроса: POST

Метод для предоставления цен и остатков товаров.

Выгружается полный справочник без разбиения на пакеты.

***Входные параметры (json/xml)***

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Заполнение |
| articul | Артикул поставщика | **Обязательно для заполнения** |
| ean | Штрих-код | **Обязательно для заполнения** |
| name | Наименование товара |  |
| cost | Цена поставки | **Обязательно для заполнения** |
| discount | Цена со скидкой |  |
| rest | Свободный остаток | **Обязательно для заполнения** |
| val | Валюта цены | USD, если не заполнено, то рубли |

*Пример структуры запроса XML*

<?xml version="1.0" encoding="utf-8"?>

<Request>

<PriceLine>

<articul>GBZ-350В</articul>

<ean>GBZ-350В</ean>

<name>Поломоечная машина KEDI GBZ-350В аккумуляторная</name>

<cost>96600</cost>

<discount>95600</discount>

<rest>7</rest>

<val></val>

</PriceLine>

<PriceLine>

<articul>GBZ-350В</articul>

<ean>GBZ-350В</ean>

<name>Поломоечная машина KEDI GBZ-350В аккумуляторная</name>

<cost>96600</cost>

<discount>95600</discount>

<rest>7</rest>

<val></val>

</PriceLine>

<PriceLine>

<articul>GBZ-380A</articul>

<ean>GBZ-380A</ean>

<name>Поломоечная машина KEDI GBZ-380A</name>

<cost>34597.5</cost>

<discount>32597.5</discount>

<rest>мало</rest>

<val></val>

</PriceLine>

<PriceLine>

<articul>GBZ-520ВT</articul>

<ean>GBZ-520ВT</ean>

<name>Поломоечная машина KEDI GBZ-520BT аккумуляторная</name>

<cost>189600</cost>

<discount></discount>

<rest>3</rest>

<val></val>

</PriceLine>

<PriceLine>

<articul>KEDI 530B-P21/22/23</articul>

<ean>KEDI 530B-P21/22/23</ean>

<name>KEDI 530B-P21/22/23||Скребок для поломоечной машины KEDI 530B-P21/22/23(д GBZ-530,520ВТ)3шт/комп</name>

<cost>53.34</cost>

<discount></discount>

<rest>20</rest>

<val>USD</val>

</PriceLine>

<PriceLine>

<articul>8.501.0506</articul>

<ean>8.501.0506</ean>

<name>Поломоечная машина LAVOR Professional Crystal Clean</name>

<cost>77700</cost>

<discount></discount>

<rest>2</rest>

<val></val>

</PriceLine>

</Request>

*Пример структуры запроса JSON*

{

"Price": [

{"articul":"GBZ-350В",

"ean":"GBZ-350В",

"name":"Поломоечная машина KEDI GBZ-350В аккумуляторная",

"cost":"96600",

"discount":"95600",

"rest":"7",

"val":""

},

{"articul":"GBZ-350В",

"ean":"GBZ-350В",

"name":"Поломоечная машина KEDI GBZ-350В аккумуляторная",

"cost":"96600",

"discount":"95600",

"rest":"7",

"val":""

},

{"articul":"GBZ-380A",

"ean":"GBZ-380A",

"name":"Поломоечная машина KEDI GBZ-380A",

"cost":"34597.5",

"discount":"32597.5",

"rest":"мало",

"val":""

},

{"articul":"GBZ-520ВT",

"ean":"GBZ-520ВT",

"name":"Поломоечная машина KEDI GBZ-520BT аккумуляторная",

"cost":"189600",

"discount":"",

"rest":"3",

"val":""

},

{"articul":"KEDI 530B-P21/22/23",

"ean":"KEDI 530B-P21/22/23",

"name":"KEDI 530B-P21/22/23||Скребок для поломоечной машины KEDI 530B-P21/22/23(д GBZ-530,520ВТ)3шт/комп",

"cost":"53.34",

"discount":"",

"rest":"20",

"val":"USD"

},

{"articul":"8.501.0506",

"ean":"8.501.0506",

"name":"Поломоечная машина LAVOR Professional Crystal Clean",

"cost":"77700",

"discount":"",

"rest":"2",

"val":""

}

]

}

***Выходные параметры***

Result – результат запроса (success/error)

Message – в случае ошибки, содержит сообщение об ошибке

*Пример структуры возвращаемых данных XML*

<?xml version="1.0" encoding="utf-8"?>

<Response>

<success>true</success>

<message>Обработка завершена</message>

</Response>

*Пример структуры возвращаемых данных JSON*

{

"success":"true",

"message:"Обработка завершена"

}

# Получение резерва

Имя метода: GetReserve

Тип запроса: POST

Метод для получения заявок на резерв.

***Входные параметры: пусто***

***Выходные параметры***

|  |  |
| --- | --- |
| Параметр | Описание |
| articul | Артикул поставщика |
| name | Наименование товара |
| rquantity | Количество запрашиваемое |
| cost | Цена поставки запрашиваемая |
| warehouse | Код склада поставки ТС |
| reserved | Уже подтверждено |
| dlvrdate | Плановая дата поставки |
| unit | Единица измерения Комус |

*Пример структуры возвращаемых данных XML*

<?xml version="1.0" encoding="utf-8"?>

<Response>

<success>true</success>

<count>1</count>

<Reserve>

<ReserveLine>

<articul>58234</articul>

<name>Драм-картридж Panasonic KX-FA84A/A7/E чер. для KX-FLM653/663/513/543</name>

<rquantity>5</rquantity>

<reserved></reserved>

<cost>110.10</cost>

<warehouse>0D0</warehouse>

< dlvrdate>2021-12-26</dlvrdate>

< unit>ШТ</unit>

</ReserveLine>

</Reserve>

</Response>

<?xml version="1.0" encoding="utf-8"?>

<Response>

<success>true</success>

<count>0</count>

</Response>

*Пример структуры возвращаемых данных JSON*

{

"success":"true",

"count":1,

"Reserve":[

{

"articul":"58234",

"name":"Драм-картридж Panasonic KX-FA84A/A7/E чер. для KX-FLM653/663/513/543",

"rquantity":"5",

"reserved":"",

"cost":"110.10",

"warehouse":"0D0",

"dlvrdate":"2021-12-26",

"unit":"ШТ"

}

]

}

{

"success":"true",

"count":0

}

# Подтверждение резерва

Имя метода: SetReserve

Тип запроса: POST

Метод для подтверждения заявок на резерв. После его выполнения изменение заказа методами API невозможно.

***Входные параметры (json/xml)***

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Заполнение |
| articul | Артикул поставщика | **Обязательно для заполнения** |
| cost | Цена поставки подтвержденная | **Если не заполнено,** Цена поставки запрашиваемая |
| quantity | Количество подтвержденное | **Если не заполнено,** Количество запрашиваемое |
| dlvrdate | Дата прихода на ЦС ТС |  |

*Пример структуры запроса XML*

<?xml version="1.0" encoding="utf-8"?>

<Request>

<Reserve>

<ReserveLine>

<articul>58234</articul>

<cost>120.10</cost>

<dlvrdate>2021-12-26</dlvrdate>

<quantity>3</quantity>

</ReserveLine>

</Reserve>

</Request>

*Пример структуры запроса JSON*

{

"Reserve":[

{

"articul":"58234",

"cost":"120.10",

"dlvrdate":"2021-12-26",

"quantity":"3"

}

]

}

***Выходные параметры***

Result – результат запроса (0 – успешно, 1 – ошибка)

Message – в случае ошибки, содержит сообщение об ошибке

*Пример структуры возвращаемых данных XML*

<?xml version="1.0" encoding="utf-8"?>

<Response>

<Result>0</Result>

<message>Новых 1 поз. Скрипт постпроцесса OK.</message>

</Response>

<?xml version="1.0" encoding="utf-8"?>

<Response>

<Result>1</Result>

<message>Нет данных!</message>

</Response>

*Пример структуры возвращаемых данных JSON*

{

"Result":0,

"message":"Новых 1 поз. Скрипт постпроцесса OK."

}

{

"Result":1,

"message":"Нет данных!"

}

# Получение заказов для формирования накладной

Имя метода: GetOrder

Тип запроса: POST

Метод для получения актуального списка товаров для формирования накладной

***Входные параметры: пусто***

***Выходные параметры***

|  |  |
| --- | --- |
| Параметр | Описание |
| articul | Артикул поставщика |
| name | Наименование товара |
| cost | Цена поставки подтвержденная |
| quantity | Количество подтвержденное |
| dlvrdate | Дата прихода на ЦС ТС |
| unit | Единица измерения Комус |

*Пример структуры возвращаемых данных XML*

<?xml version="1.0" encoding="utf-8"?>

<Response>

<success>true</success>

<count>3</count>

<Orders>

<OrdersLine>

<articul>58234</articul>

<name>Драм-картридж Panasonic KX-FA84A/A7/E чер. для KX-FLM653/663/513/543</name>

<quantity>3</quantity>

<cost>120.10</cost>

<dlvrdate>2021-12-26</dlvrdate>

<unit>ШТ</unit>

</OrdersLine>

</Orders>

</Response>

*Пример структуры возвращаемых данных JSON*

{

"success":"true",

"count":3,

"Orders":

[

{

"articul":"58234",

"name":" Драм-картридж Panasonic KX-FA84A/A7/E чер. для KX-FLM653/663/513/543",

"quantity":"3",

"cost":"120.10",

"dlvrdate":"2021-12-26",

"unit":"ШТ"

}

]

}

# Получение сформированной накладной

Имя метода: GetFinalOrder

Тип запроса: POST

Метод для получения сформированных накладных

***Входные параметры (json/xml)***

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Заполнение |
| key | Уникальный ID накладной в системе AT | **Если не заполнено,** Накладные за текущую дату |
| date | Дата накладной/Диапазон дат | **Если не заполнено,** Накладные за текущую дату |
| number | Номер накладной в системе ТС | **Если не заполнено,** Накладные за текущую дату |
| warehouse | Код склада поставки ТС | **Если не заполнено,** Накладные за текущую дату по всем складам поставки ТС |

*Пример структуры запроса XML*

<?xml version="1.0" encoding="utf-8"?>

<Request>

<Orders>

<key></key>

<number></number>

<date>2021-12-25</date>

<warehouse>0D0</warehouse>

</Orders>

</Request>

*Пример структуры запроса JSON*

{

"Orders":

{

"key":"",

"number":"",

"date": "2021-12-25",

"warehouse": ""

}

}

***Выходные параметры***

OrderHeader – заголовок заказа, структура:

|  |  |
| --- | --- |
| Параметр | Описание |
| key | Уникальный ID накладной в системе AT |
| number | Номер накладной в системе ТС |
| docsum | Сумма накладной |
| amount | Количество позиций накладной |
| dlvrdate | Дата прихода на ЦС ТС |
| warehouse | Код склада поставки ТС |

OrderItems – строки заказа, структура:

|  |  |
| --- | --- |
| Параметр | Описание |
| articul | Артикул поставщика |
| name | Наименование товара |
| quantity | Количество подтвержденное |
| cost | Цена поставки подтвержденная |
| sum | Сумма поставки подтвержденная |
| unit | Единица измерения Комус |

*Пример структуры возвращаемых данных XML*

<?xml version="1.0" encoding="utf-8"?>

<Response>

<Orders>

<OrdersLine>

<OrderHeader>

<key>816270</key>

<number>Zkomus-lt-test21-00008658</number>

<warehouse>0D0</warehouse>

<date>2021-12-25</date>

<dlvrdate></dlvrdate>

<docsum>390.30</docsum>

<amount>1</amount>

</OrderHeader>

<OrderItems>

<OrderItemsLine>

<articul>58234</articul>

<name>Драм-картридж Panasonic KX-FA84A/A7/E чер. для KX-FLM653/663/513/543</name>

<quantity>3</quantity>

<cost>130.10</cost>

<sum>390.30</sum>

<unit>ШТ</unit>

</OrderItemsLine>

</OrderItems>

</OrdersLine>

</Orders>

</Response>

*Пример структуры возвращаемых данных JSON*

{

"Orders":[

{

"OrderHeader":

{

"key":816270,

"number":"Zkomus-lt-test21-00008658",

"warehouse":"0D0",

"date":"2021-12-25",

"dlvrdate":"",

"docsum":"390.30",

"amount":3

},

"OrderItems":[

{

"rkod":"20000042",

"articul":"58234",

"name":" Драм-картридж Panasonic KX-FA84A/A7/E чер. для KX-FLM653/663/513/543",

"quantity":14,

"cost":"130.10",

"sum":"1821.40"",

"unit":"ШТ"

}

]

}

]

}

# Получение справочника складов для отгрузки

Имя метода: getWarehouses

Тип запроса: POST

Метод для получения справочника складов ТС для отгрузки.

***Входные параметры: пусто***

***Выходные параметры***

|  |  |
| --- | --- |
| Параметр | Описание |
| warehouse | Код склада ТС |
| warehouse\_name | Наименование склада |
| address | Адрес склада |

*Пример структуры возвращаемых данных XML*

<?xml version="1.0" encoding="utf-8"?>

<Response>

<success>true</success>

<count>3</count>

<Warehouses>

<WarehousesLine>

<warehouse>0Q0</warehouse>

<warehouse\_name>Реутов</warehouse\_name>

<address>г.Реутов, ул. Никольская, д.2</address>

</WarehousesLine>

<WarehousesLine>

<warehouse>0QR</warehouse>

<warehouse\_name>Реутов</warehouse\_name>

<address>г.Реутов, ул. Никольская, д.2</address>

</WarehousesLine>

<WarehousesLine>

<warehouse>0D0</warehouse>

<warehouse\_name>Домодедово</warehouse\_name>

<address>г.Домодедово, ул. Станционная 24 Б</address>

</WarehousesLine>

</Warehouses>

</Response>

*Пример структуры возвращаемых данных JSON*

{

"success":"true",

"count":3,

"Warehouses":[

{warehouse: '0Q0',

warehouse\_name: 'Реутов',

address: 'г.Реутов, ул. Никольская, д.2'},

{warehouse: 0QR,

warehouse\_name: Реутов,

address: ‘г.Реутов, ул. Никольская, д.2’},

{warehouse: 0D0,

warehouse\_name: Домодедово,

address: ' г.Домодедово, ул. Станционная 24 Б '}

]

}