

**pikassa**

# **Документация Merchant API**

**ВЕРСИЯ 1.8**

## Содержание

Требования	4
Схема проведения операции	4
Создание счета на оплату	5
Параметры запроса:	5
Пример POST формы	7
Пример ссылки на оплату	7
Электронная цифровая подпись	7
Пример формирования подписи на .Net	8
Пример формирования подписи на php	9
Пример формирования подписи на Python	10
Примеры ответов	10
Возврат денежных средств по счету	11
Параметры запроса	11
Пример POST формы	12
Пример ответа	12
Отправка уведомлений об изменении статуса счета	12
Параметры запроса	13
Пример запроса	14
Параметры ответа	14
Пример ответа	14
Подтверждение платежа	14
Параметры запроса	15
Отмена предавторизации	15
Параметры запроса	15
Пример POST формы	16
Пример ответа	16
Аннулирование счета	17
Параметры запроса	17
Пример POST формы	17

Пример ответа	18
<b>Отправка данных в налоговую в соответствии с ФЗ-54</b>	<b>18</b>
Структура параметров запроса	29
Описание подпараметра client	30
Описание подпараметра company	31
Описание подпараметра agent_info	32
Описание подпараметра supplier_info	33
Описание подпараметра items	33
Описание подпараметра payments	38
Описание подпараметра vats	39
Описание подпараметра additional_user_props	40

# Требования

Для подключения необходимо:

1. Зарегистрировать личный кабинет Pikassa по кнопке «Подключиться» на странице <https://pikassa.io/>;
2. В личном кабинете создать магазин в разделе «Магазины» и заполнить его общие данные;
3. Сгенерировать в личном кабинете в технических настройках магазина секретный ключ (secretPhrase) для формирования электронной цифровой подписи;
4. Заполнить в личном кабинете в технических настройках поле Result URL для получения уведомлений о проведенных платежах.

## Схема проведения операции

1. Магазин отправляет в Pikassa запрос на создание счета;
2. Pikassa обрабатывает запрос;
3. Если счет создан корректно, Pikassa отправляет запрос на проведение операции в банк-эквайер и перенаправляет плательщика на платежную страницу банка-эквайера;
4. Плательщик вводит данные банковской карты;
5. Банк-эквайер отправляет запрос на списание денежных средств в банк-эмитент (банк, выпустивший карту плательщика);
6. Банк-эмитент отправляет ответ в банк-эквайер;
7. Pikassa получает ответ об успешной/неуспешной оплате счета от банка-эквайера;
8. Происходит переадресация плательщика на страницу магазина\*
  - 8.1. В случае успешной оплаты счета плательщик перенаправляется на страницу, указанную в параметре PIMPAY\_SUCCESS\_URL;
  - 8.2. В случае неуспешной оплаты происходит редирект на страницу, указанную в параметре PIMPAY\_FAIL\_URL;
9. Pikassa отправляет магазину уведомление об успешной/неуспешной оплате счета;
10. Магазин возвращает ответ-подтверждение о проведенной операции.

---

\* В адрес, переданный в запросе на создание счета, добавляется параметр "external Id", в котором передается идентификатор счета в системе магазина. Например, <http://bestshop.ru/success?externalId=e5ebc0d4-f90b-409b-874c> Параметр external Id эквивалентен параметру PIMPAY\_EXTERNAL\_ID для формы создания счета .

# Создание счета на оплату

Используйте POST запрос:

<https://kassa.pimpay.ru/merchant-api/api/v1/CreateInvoice>

`contentType: "application/x-www-form-urlencoded";`

## Параметры запроса:

<code>PIMPAY_SHOP_ID</code>	обязательное, числовое, больше нуля	Идентификатор магазина в Pikassa
<code>PIMPAY_EXTERNAL_ID</code>	обязательное, текстовое	Уникальный внутренний идентификатор счета в системе магазина, максимум - 100 символов, a-z,A-Z,0-9, "-"
<code>PIMPAY_AMOUNT</code>	обязательное, числовое с плавающей запятой (1-15000), не более 2-х знаков после запятой	Сумма счета, разделитель дробной части - "." (точка)
<code>PIMPAY_DESC</code>	обязательное, текстовое	Описание платежа, максимум 1000 символов
<code>PIMPAY_INVOICE_EXPIRATION_DATE</code>	необязательное, текстовое	Дата и время действия счета  Если в запросе указать данный параметр, то произвести оплату по данному счету можно будет только до момента наступления указанных даты и времени. Если по истечении заданного времени и даты оплата не была совершена, счет переходит в состояние 6 (Cancelled), и при попытке его оплатить пользователь будет перенаправлен на страницу ошибки ( <code>PIMPAY_FAIL_URL</code> )  Формат даты: <code>yyyy-MM-dd HH:mm:ss.fffz</code> Пример: <code>2018-04-28 17:42:30.220+03:00</code>
<code>PIMPAY_CUSTOM_DATA</code>	необязательное, текстовое	Различная служебная информация, поле будет возвращено в уведомлении об оплате счета с результатом платежа, максимум 1000 символов
<code>PIMPAY_CUSTOMER_EMAIL</code>	необязательное, текстовое	Email плательщика, максимум 320 символов
<code>PIMPAY_CUSTOMER_PHONE</code>	необязательное, текстовое	Телефон плательщика. Пример: <code>79038887767/+79038887767</code> , максимум 20 символов, 0-9, "+"

<code>PIMPAY_INVOICE_DELIVERY_METHOD</code>	необязательное, текстовое	<p>Метод доставки счета клиенту:</p> <p><b>BROWSER</b> – моментальная оплата счета в браузере. Платежная страница отобразится в браузере.</p> <p><b>EMAIL</b> – отправить счет на адрес электронной почты. Адрес передается параметром <code>PIMPAY_CUSTOMER_EMAIL</code>, в случае отсутствия данных в этом поле запрос считается невалидным.</p> <p><b>SMS</b> – отправить счет SMS-сообщением на телефон Телефон передается параметром <code>PIMPAY_CUSTOMER_PHONE</code>, в случае отсутствия данных в этом поле запрос считается невалидным.</p> <p><b>URL</b> – при использовании данного способа доставки в ответе возвращается параметр <code>redirectUrl</code>, содержащий ссылку на оплату. Пример ссылки смотрите ниже.</p> <p>Если параметр отсутствует в запросе, то такой запрос считается по умолчанию <b>BROWSER</b>.</p>
<code>PIMPAY_SUCCESS_URL</code>	обязательное, текстовое	URL-адрес страницы магазина, на которую будет переправлен плательщик в случае успешно оплаченного счета, максимум 100 символов.
<code>PIMPAY_FAIL_URL</code>	обязательное, текстовое	URL-адрес страницы магазина, на которую будет переправлен плательщик в случае неуспешно оплаченного счета, максимум 100 символов.
<code>PIMPAY_OFD_DATA</code>	необязательное, текстовое	Данные для фискализации. В зависимости от настроек магазина отправляются тому или иному провайдеру фискальных данных.
<code>PIMPAY_INVOICE_CURRENCY</code>	необязательное, текстовое	<p>Валюта счета. Поддерживаются следующие значения:</p> <p>RUB – рубль EUR – евро USD – доллар</p> <p>Если параметр отсутствует в запросе, то валютой счета по умолчанию является рубль (<b>RUB</b>).</p>
<code>PIMPAY_PREAUTH</code>	необязательное, логическое	Для проведения двухэтапных платежей. Допустимые значения "0" или "1"
<code>PIMPAY_SIGN</code>	обязательное	Электронная цифровая подпись

## Пример POST формы

```
<form action="https://kassa.pimpay.ru/merchant-api/api/v1/CreateInvoice" method="POST">
  <input type="hidden" name="PIMPAY_SHOP_ID" value="1">
  <input type="hidden" name="PIMPAY_EXTERNAL_ID" value="e5ebc0d4-f90b-409b-874c-c72991da">
  <input type="hidden" name="PIMPAY_AMOUNT" value="3500.90">
  <input type="hidden" name="PIMPAY_DESC" value="Назначение (описание) платежа">
  <input type="hidden" name="PIMPAY_CUSTOM_DATA" value="Служебная информация">
  <input type="hidden" name="PIMPAY_CURRENCY" value="RUB">
  <input type="hidden" name="PIMPAY_CUSTOMER_EMAIL" value="client@e-mail.ru">
  <input type="hidden" name="PIMPAY_CUSTOMER_PHONE" value="79997778899">
  <input type="hidden" name="PIMPAY_INVOICE_DELIVERY_METHOD" value="BROWSER">
  <input type="hidden" name="PIMPAY_SUCCESS_URL" value="http://yoursite.com/success">
  <input type="hidden" name="PIMPAY_FAIL_URL" value="http://yoursite.com/fail">
  <input type="hidden" name="PIMPAY_SIGN" value="вычисленное значение подписи">
  <input type="submit" value="Оплатить">
</form>
```

## Пример ссылки на оплату

```
{
  "success": true,
  "externalId": "b9720b69-1d48-4d6f-8180-fc6f789bd412",
  "message": null,
  "redirectUrl": "https://pikassa.io/portal2/pay/i/c5a11a3b-90c3-4073-88c1-78355154d8db"
}
```

## Электронная цифровая подпись

Электронная цифровая подпись отвечает за безопасность операции.

Для формирования подписи используйте **secretPhrase** (уникальный ключ, который генерируется в технических настройках магазина в личном кабинете) и все параметры, которые передаете при создании счета.

**Электронную цифровую подпись необходимо вычислять и передавать при создании каждого счета и проверять при получении результата оплаты счета.**

Для параметров **PIMPAY\_A = 111**, **PIMPAY\_B = "VALUE"**, **PIMPAY\_C = "Чай"** и секретной фразы **"secretPhrase"**:

1. Параметры сортируются по имени параметра в прямом алфавитном порядке.

2. Производится URL-кодирование значений параметров (кодировка UTF8)
3. Создается строка для формирования подписи: имена параметров объединяются с URL-кодированными и приведенными к верхнему регистру значениями через символ "=". Получившиеся пары конкатенируются через символ "&".

Пример:

```
"PIMPAY_A=111&PIMPAY_B=VALUE&PIMPAY_C=%D0%A7%D0%B0%D0%B9"
```

4. К полученной строке добавляется секретная фраза.

Пример:

```
"PIMPAY_A=111&PIMPAY_B=Value&PIMPAY_C=%D0%A7%D0%B0%D0%B9secretPhrase"
```

5. По полученной строке считается md5-хэш (кодировка UTF8)
6. Конвертируем результат в base64 строку

#### Пример формирования подписи на .Net

```
public IDictionary<string, string> AddSignature(IDictionary<string, string> queryParams,
string secretPhrase)
{
    using (var hasher = MD5.Create())
    {
        var encoding = Encoding.UTF8;
        var encodedParams = queryParams.OrderBy(p => p.Key).Select(p => $"{p.Key}
={HttpUtility.UrlEncode(p.Value, encoding)}");
        var queryString = string.Join("&", encodedParams).ToUpper();
        var toSignBytes = encoding.GetBytes(queryString + secretPhrase);
        var signBytes = hasher.ComputeHash(toSignBytes);
        var signStr = Convert.ToBase64String(signBytes);
        queryParams.Add("PIMPAY_SIGN", signStr);
    }

    return queryParams;
}
```



## Пример формирования подписи на php

```
<?php
const URL = 'https://kassa.pimpay.ru/merchant-api/api/v1/CreateInvoice';
$secret = 'VerySecret_Phrase!234';
$params = array(
    'PIMPAY_SHOP_ID'      => '1',
    'PIMPAY_EXTERNAL_ID' => uniqid(),
    'PIMPAY_AMOUNT'     => '12000.35',
    'PIMPAY_DESC'       => 'Описание счета',
    'PIMPAY_CUSTOM_DATA' => '{"name": \"value\"}',
    'PIMPAY_CUSTOMER_EMAIL' => 'asd@asd.ru',
    'PIMPAY_CUSTOMER_PHONE' => '+79345557788',
    'PIMPAY_SUCCESS_URL' => 'http://google.com/success',
    'PIMPAY_FAIL_URL'    => 'http://google.com/fail'
);
uksort($params, 'strcasecmp');
$toSignArray = Array();
foreach ($params as $key => $value) {
    $toSignArray[] = $key . '=' . urlencode($value);
}

//объединяем через &
$toSign = implode('&', $toSignArray);
//приводим к верхнему регистру
$toSign = strtoupper($toSign);
//добавляем secret
$toSign = $toSign . $secret;
//приводим к UTF-8
$toSign = mb_convert_encoding($toSign, "UTF-8");
$sign = base64_encode(md5($toSign, true));
$params['PIMPAY_SIGN'] = $sign

;
?>
<html>
<head>
<meta charset="utf-8" />
</head>
<body>
<form action="<?= URL ?>" method="post" enctype="application/x-www-form-urlencoded">
<? foreach ($params as $key => $value) { ?>
<?= $key ?>: <input type="text" value="<?= $value ?>" name="<?= $key ?>" /><br/>
<? } ?>
<button type="submit">Submit</button>
</form>
</body>
</html>
```

## Пример формирования подписи на Python

```
# -*- coding: UTF-8 -*-

import urllib.parse
import base64
import hashlib
import urllib

params = {
    "PIMPAY_SHOP_ID": "1",
    "PIMPAY_EXTERNAL_ID": "123",
    "PIMPAY_AMOUNT": "10",
    "PIMPAY_DESC": "Payment",
    "PIMPAY_CUSTOMER_EMAIL": "test@test.com",
    "PIMPAY_CUSTOMER_PHONE": "+71231231212",
    "PIMPAY_SUCCESS_URL": "https://google.com",
    "PIMPAY_FAIL_URL": "https://ya.ru"
}

secret = '12345678'

def calc_sign(params: dict, secret: str):
    to_sign = ''
    for key in sorted(params.keys(), key=str.lower):
        if len(to_sign) != 0:
            to_sign += '&'
        to_sign += '{}={}'.format(key, urllib.parse.quote_plus(str(params[key])))
    to_sign = to_sign.upper() + secret
    hasher = hashlib.md5(to_sign.encode('utf-8'))
    sign = base64.b64encode(hasher.digest())
    return sign

params['PIMPAY_SIGN'] = calc_sign(params, secret)
```

При неправильном создании счета на оплату возвращается JSON response с кодом статуса 400/500 и описанием ошибки.

## Примеры ответов

Для всех типов, кроме счетов с методом доставки **BROWSER**.

в случае успешного выполнения (status code = 200)

```
{
  success: true,
  externalId: "e5ebc0d4-f90b-409b-874c-c72991da"
}
```

в случае ошибки (status code = 400/500)

```
{
  success: false,
  externalId: "e5ebc0d4-f90b-409b-874c-c72991da",
  message: "Field 'description' has invalid format (exceeded the maximum length (1000))"
}
```

Для счетов с методом доставки **BROWSER** возвращается **status code** (400/500) и **plain text** в **body** с описанием причины в случае ошибки.

## Возврат денежных средств по счету

Используйте POST запрос:

```
https://kassa.pimpay.ru/merchant-api/api/v1/RefundInvoice
contentType: "application/x-www-form-urlencoded";
```

### Параметры запроса

<b>PIMPAY_SHOP_ID</b>	обязательное, числовое, больше нуля	Идентификатор магазина в Pikassa
<b>PIMPAY_EXTERNAL_ID</b>	обязательное, текстовое	Уникальный внутренний идентификатор счета в системе магазина, максимум - 100 символов, a-z,A-Z,0-9, “-“
<b>PIMPAY_AMOUNT</b>	обязательное, числовое с плавающей запятой (1-15000), не более 2-х знаков после запятой	Сумма счета, разделитель дробной части - "." (точка)
<b>PIMPAY_REASON</b>	обязательное, текстовое, не более 1000 символов	Описание причины возврата
<b>PIMPAY_SIGN</b>	обязательное	Электронная цифровая подпись

## Пример POST формы

```
<form action="https://kassa.pimpay.ru/merchant-api/api/v1/RefundInvoice" method="POST">
  <input type="hidden" name="PIMPAY_SHOP_ID" value="1">
  <input type="hidden" name="PIMPAY_EXTERNAL_ID" value="e5ebc0d4-f90b-409b-874c-c72991da">
  <input type="hidden" name="PIMPAY_AMOUNT" value="176.80">
  <input type="hidden" name="PIMPAY_REASON" value="Причина возврата">
  <input type="hidden" name="PIMPAY_SIGN" value="вычисленное значение подписи">
  <input type="submit" value="Вернуть">
</form>
```

## Пример ответа

в случае успешного выполнения (status code = 200)

```
{
  success: true,
  externalId: "e5ebc0d4-f90b-409b-874c-c72991da"
}
```

в случае ошибки (status code = 400/500)

```
{
  success: false,
  externalId: "e5ebc0d4-f90b-409b-874c-c72991da",
  message: "Сообщение об ошибке"
}
```

## Отправка уведомлений об изменении статуса счета

При регистрации магазина указывается URL адрес обратного вызова магазина (POST-запрос), по которому будет осуществляться нотификация о статусе совершенных операций.

POST запрос по URL, указанному в настройках магазина:

```
contentType: "application/x-www-form-urlencoded";
```

## Параметры запроса

<b>PIMPAY_EXTERNAL_ID</b>	обязательное, текстовое	Уникальный внутренний идентификатор счета в системе магазина, максимум - 100 символов, a-z,A-Z,0-9,“-“  Должен начинаться и заканчиваться на букву/цифру
<b>PIMPAY_AMOUNT</b>	обязательное, числовое с плавающей запятой (1-15000), не более 2-х знаков после запятой	Сумма счета, разделитель дробной части - "." (точка)
<b>PIMPAY_FINAL_AMOUNT</b>	обязательное, числовое с плавающей запятой (1-15000), не более 2-х знаков после запятой	Финальная сумма, которая была оплачена по данному счету. Разделитель дробной части - "." (точка)
<b>PIMPAY_INVOICE_CURRENCY</b>	обязательное, текстовое	Валюта счета
<b>PIMPAY_DESC</b>	обязательное, текстовое	Описание платежа, максимум 1000 символов
<b>PIMPAY_CUSTOM_DATA</b>	необязательное, текстовое	Различная служебная информация, поле будет возвращено в уведомлении об оплате счета с результатом платежа, максимум 1000 символов.*
<b>PIMPAY_STATUS_CODE</b>	обязательное, числовое	Возвращает текущий статус счета 1 – Payed (Оплачен) 2 – Failed (Ошибка оплаты) 3 – PartlyRefunded (Частично возврат) 4 – Refunded (Полный возврат) 5 – RefundFailed (Ошибка возврата) 6 – Cancelled (Отменен)
<b>PIMPAY_STATUS_TIME</b>	обязательно, текстовое	Дата обновления статуса в формате "YYYY-MM-DDThh:mm:ss±hh:mm" (ISO 8601), где: YYYY - год (строго 4 цифры) MM - месяц (строго 2 цифры) DD - день (строго 2 цифры) hh - час (в 24 часовом формате, строго 2 цифры) mm - день (строго 2 цифры) ss - день (строго 2 цифры) ±hh:mm - часовой пояс Например: 2005-08-09T18:31:42+03:30 (9 августа 2005 года 18 часов 31 минута 42 секунды UTC+03 часа 30 минут)"
<b>PIMPAY_STATUS_REASON</b>	обязательное, текстовое	Описание причины перехода в текущий статус
<b>PIMPAY_INVOICE_ID</b>	обязательное, числовое	Идентификатор счета в системе kassa.pimpay
<b>PIMPAY_SIGN</b>	обязательное	Электронная подпись

## Пример запроса

```
PIMPAY_EXTERNAL_ID=e5ebc0d4-f90b-409b-874c&PIMPAY_AMOUNT=3500.90&PIMPAY_DESC=Назначение  
(описание) платежа&PIMPAY_CUSTOM_DATA=Служебная  
информация&PIMPAY_STATUS_CODE=1&PIMPAY_STATUS_TIME=2018-05-08  
17:17:07.689+03:00&PIMPAY_STATUS_REASON=Списание с карты  
Visa&PIMPAY_INVOICE_ID=3343&PIMPAY_SIGN=вычисленная_подпись
```

## Параметры ответа

В ответ ожидается JSON с параметрами:

Success	Результат обработки (true/false)
externalId	Уникальный внутренний идентификатор счета в системе магазина, максимум - 100 символов, a-z,A-Z,0-9, “-“ Должен начинаться и заканчиваться на цифру Эквивалентен параметру PIMPAY_EXTERNAL_ID для формы создания счета.

## Пример ответа

```
{  
  success: true,  
  externalId: "e5ebc0d4-f90b-409b-874c"  
}
```

Если при отправке уведомления о статусе операции Pikassa не получит ответ от магазина либо получит ответ со статусом `success: false`, то Pikassa будет отправлять уведомление 10 раз, пока не получит ответ со статусом `success: true`.

## Подтверждение платежа

Метод позволяет списать ранее захолдированную сумму с карты плательщика. Используется только в двухэтапных платежах.

---

\* В случае если счет был оплачен через «Почту России», в служебной информации будет возвращен трек-номер посылки.  
Пример ответа: {"postalId": "14086731020428"}

Используйте POST запрос:

```
https://kassa.pimpay.ru/merchant-api/api/v1/AuthInvoice  
contentType: "application/x-www-form-urlencoded";
```

### Параметры запроса

<b>PIMPAY_SHOP_ID</b>	обязательное, числовое, больше нуля	Идентификатор магазина в Pikassa
<b>PIMPAY_EXTERNAL_ID</b>	обязательное, текстовое	Уникальный внутренний идентификатор счета в системе магазина, максимум - 100 символов, a-z,A-Z,0-9, "-"
<b>PIMPAY_AMOUNT</b>	обязательное, числовое с плавающей запятой (1-15000), не более 2-х знаков после запятой	Сумма счета, разделитель дробной части - "." (точка)
<b>PIMPAY_REASON</b>	обязательное, текстовое, не более 1000 символов	Описание причины возврата
<b>PIMPAY_SIGN</b>	обязательное	Электронная цифровая подпись

## Отмена предавторизации

Метод позволяет разблокировать ранее захолдированную сумму на карте плательщика. Используется только в двухэтапных платежах.

Используйте POST запрос:

```
https://kassa.pimpay.ru/merchant-api/api/v1/CancelInvoice  
contentType: "application/x-www-form-urlencoded";
```

### Параметры запроса

<b>PIMPAY_SHOP_ID</b>	обязательное, числовое, больше нуля	Идентификатор магазина в Pikassa
<b>PIMPAY_EXTERNAL_ID</b>	обязательное, текстовое	Уникальный внутренний идентификатор счета в системе магазина, максимум - 100 символов, a-z,A-Z,0-9, "-"
<b>PIMPAY_AMOUNT</b>	обязательное, числовое с плавающей запятой (1-15000), не более 2-х знаков после запятой	Сумма счета, разделитель дробной части - "." (точка)

<b>PIMPAY_REASON</b>	обязательное, текстовое, не более 1000 символов	Описание причины расходирования
<b>PIMPAY_SIGN</b>	обязательное	Электронная цифровая подпись

### Пример POST формы

```
<form action="https://kassa.pimpay.ru/merchant-api/api/v1/CancelInvoice" contentType:
"application/x-www-form-urlencoded"; method="POST">
  <input type="hidden" name="PIMPAY_SHOP_ID" value="1">
  <input type="hidden" name="PIMPAY_EXTERNAL_ID" value="e5ebc0d4-f90b-409b-874c-c72991da">
  <input type="hidden" name="PIMPAY_AMOUNT" value="176.80">
  <input type="hidden" name="PIMPAY_REASON" value="Причина отмены предавторизации">
  <input type="hidden" name="PIMPAY_SIGN" value="вычисленное значение подписи">
  <input type="submit" value="Отменить">
</form>
```

### Пример ответа

в случае успешного выполнения (status code = 200)

```
{
  success: true,
  externalId: "e5ebc0d4-f90b-409b-874c-c72991da"
}
```

в случае ошибки (status code = 400/500)

```
{
  success: false,
  externalId: "e5ebc0d4-f90b-409b-874c-c72991da",
  message: "Сообщение об ошибке"
}
```



## Аннулирование счета

Метод позволяет отменить возможность оплаты счета. Доступен только для тех счетов, по которым еще не было переходов на платежную страницу. В обратном случае запрос будет отклонен.

Аннулирование недоступно для счетов с методом доставки «BROWSER», так как он подразумевает моментальный переход на страницу оплаты.

Используйте POST запрос:

```
https://kassa.pimpay.ru/merchant-api/api/v1/CancelInvoice  
contentType: "application/x-www-form-urlencoded";
```

### Параметры запроса

<code>PIMPAY_SHOP_ID</code>	обязательное, числовое, больше нуля	Идентификатор магазина в Pikassa
<code>PIMPAY_EXTERNAL_ID</code>	обязательное, текстовое	Уникальный внутренний идентификатор счета в системе магазина, максимум - 100 символов, a-z,A-Z,0-9, “-“
<code>PIMPAY_REASON</code>	обязательное, текстовое, не более 1000 символов	Описание причины
<code>PIMPAY_SIGN</code>	обязательное	Электронная цифровая подпись

### Пример POST формы

```
<form action="https://kassa.pimpay.ru/merchant-api/api/v1/CancelInvoice" method="POST">  
<input type="hidden" name="PIMPAY_SHOP_ID" value="1">  
<input type="hidden" name="PIMPAY_EXTERNAL_ID" value="836277C3-F70E-41D7-  
B748-225DBF065762">  
<input type="hidden" name="PIMPAY_REASON" value="Причина аннулирования счета">  
<input type="hidden" name="PIMPAY_SIGN" value="вычисленное значение подписи">  
<input type="submit" value="Вернуть">  
</form>
```

## Пример ответа

в случае успешного выполнения (status code = 200)

```
{
  success: true,
  externalId: "e5ebc0d4-f90b-409b-874c-c72991da"
}
```

в случае ошибки (status code = 400/500)

```
{
  success: false,
  externalId: "e5ebc0d4-f90b-409b-874c-c72991da",
  message: "Сообщение об ошибке"
}
```

## Отправка данных в налоговую в соответствии с Ф3-54

Для отправки фискальных данных используется протокол АТОЛ; поддерживаемый провайдер данных – АТОЛ.

Для подключения возможности передавать фискальные данные необходимо сообщить данные аккаунта АТОЛ менеджеру Pikassa.

**ВАЖНО:** Параметры, которые не должны быть указаны в JSON схеме: `"timestamp"`, `"external_id"`, `"service"`.

## JSON схема запроса

```
{
  "type": "object",
  "properties": {
    "receipt": {
      "type": "object",
      "properties": {
        "client": {
          "anyOf": [
            {
              "type": "object",
              "properties": {
                "email": {
                  "type": "string",
                  "format": "email",
                  "pattern": "^[^\\s]+$",
                  "maxLength": 64
                },
                "phone": {
                  "type": "string",
                  "maxLength": 64
                }
              }
            },
            {
              "type": "object",
              "properties": {
                "email": {
                  "type": "string",
                  "format": "email",
                  "pattern": "^[^\\s]+$",
                  "maxLength": 64
                },
                "phone": {
                  "type": "string",
                  "pattern": "^[^\\s]+$",
                  "maxLength": 64
                }
              }
            }
          ]
        }
      }
    }
  }
}
```

```

    },
    "company": {
      "type": "object",
      "properties": {
        "email": {
          "type": "string",
          "maxLength": 64
        },
        "sno": {
          "enum": [
            "osn",
            "usn_income",
            "usn_income_outcome",
            "envd",
            "esn",
            "patent"
          ]
        },
        "inn": {
          "type": "string",
          "maxLength": 12
        },
        "payment_address": {
          "type": "string",
          "maxLength": 256
        }
      },
      "required": [
        "email",
        "inn",
        "payment_address"
      ]
    },
    "agent_info": {
      "type": "object",
      "properties": {
        "type": {
          "type": "string",
          "enum": [
            "bank_paying_agent",
            "bank_paying_subagent",
            "paying_agent",
            "paying_subagent",
            "attorney",

```

```

"paying_agent": {
  "type": "object",
  "properties": {
    "operation": {
      "type": "string"
    },
    "phones": {
      "type": "array",
      "items": {
        "type": "string"
      }
    }
  }
},
"receive_payments_operator": {
  "type": "object",
  "properties": {
    "phones": {
      "type": "array",
      "items": {
        "type": "string"
      }
    }
  }
},
"money_transfer_operator": {
  "type": "object",
  "properties": {
    "phones": {
      "type": "array",
      "items": {
        "type": "string"
      }
    },
    "name": {
      "type": "string"
    },
    "address": {
      "type": "string"
    },
    "inn": {
      "type": "string"
    }
  }
}

```

```

"supplier_info": {
  "type": "object",
  "properties": {
    "phones": {
      "type": "array",
      "items": {
        "type": "string"
      }
    }
  }
},
"items": {
  "type": "array",
  "minItems": 1,
  "items": {
    "type": "object",
    "properties": {
      "name": {
        "type": "string",
        "maxLength": 128
      },
      "price": {
        "$ref": "#/definitions/number_price"
      },
      "quantity": {
        "$ref": "#/definitions/number_three_format"
      },
      "sum": {
        "$ref": "#/definitions/number_two_format"
      },
      "measurement_unit": {
        "type": "string",
        "maxLength": 16
      },
      "payment_method": {
        "enum": [
          "full_prepayment",
          "prepayment",
          "advance",
          "full_payment",
          "partial_payment",
          "credit",
          "credit_payment"
        ]
      }
    }
  }
}

```

```

"payment_object": {
  "enum": [
    "commodity",
    "excise",
    "job",
    "service",
    "gambling_bet",
    "gambling_prize",
    "lottery",
    "lottery_prize",
    "intellectual_activity",
    "payment",
    "agent_commission",
    "composite",
    "another",
    "property_right",
    "non-operating_gain",
    "insurance_premium",
    "sales_tax",
    "resort_fee"
  ]
},
"vat": {
  "type": "object",
  "properties": {
    "type": {
      "enum": [
        "none",
        "vat0",
        "vat10",
        "vat18",
        "vat20",
        "vat110",
        "vat118",
        "vat120"
      ]
    },
    "sum": {
      "$ref": "#/definitions/number_two_format"
    }
  }
},

```

```

"agent_info": {
  "type": "object",
  "properties": {
    "type": {
      "enum": [
        "bank_paying_agent",
        "bank_paying_subagent",
        "paying_agent",
        "paying_subagent",
        "attorney",
        "commission_agent",
        "another"
      ]
    },
    "paying_agent": {
      "type": "object",
      "properties": {
        "operation": {
          "type": "string",
          "maxLength": 24
        },
        "phones": {
          "type": "array",
          "items": {
            "type": "string"
          }
        }
      }
    },
    "receive_payments_operator": {
      "type": "object",
      "properties": {
        "phones": {
          "type": "array",
          "items": {
            "type": "string"
          }
        }
      }
    },
    "money_transfer_operator": {
      "type": "object",
      "properties": {
        "phones": {

```



```

        },
        "name": {
            "type": "string"
        },
        "address": {
            "type": "string"
        },
        "inn": {
            "type": "string"
        }
    }
}
},
"supplier_info": {
    "type": "object",
    "properties": {
        "phones": {
            "type": "array",
            "items": {
                "type": "string"
            }
        },
        "name": {
            "type": "string"
        },
        "inn": {
            "type": "string"
        }
    }
},
"user_data": {
    "type": "string",
    "maxLength": 64
}
},
"required": [
    "name",
    "price",
    "quantity",
    "sum",
    "payment_method",
    "payment_object"
]

```

```

"payments": {
  "type": "array",
  "minItems": 1,
  "maxItems": 10,
  "items": {
    "type": "object",
    "properties": {
      "type": {
        "$ref": "#/definitions/type_format"
      },
      "sum": {
        "$ref": "#/definitions/number_two_format"
      }
    },
    "required": [
      "type",
      "sum"
    ]
  }
},
"vats": {
  "type": "array",
  "minItems": 1,
  "maxItems": 6,
  "items": {
    "type": "object",
    "properties": {
      "type": {
        "enum": [
          "none",
          "vat0",
          "vat10",
          "vat18",
          "vat20",
          "vat110",
          "vat118",
          "vat120"
        ]
      },
      "sum": {
        "$ref": "#/definitions/number_two_format"
      }
    }
  }
}

```

```

    },
    "additional_check_props": {
      "type": "string",
      "maxLength": 16
    },
    "cashier": {
      "type": "string",
      "maxLength": 64
    },
    "additional_user_props": {
      "type": "object",
      "properties": {
        "name": {
          "type": "string",
          "maxLength": 64
        },
        "value": {
          "type": "number",
          "maxLength": 256
        }
      },
      "required": [
        "name",
        "value"
      ]
    }
  ],
  "required": [
    "client",
    "company",
    "items",
    "total",
    "payments"
  ]
}
},
"definitions": {
  "number_two_format": {
    "type": "number",
    "minimum": 0,
    "maximum": 100000000,
    "multipleOf": 0.01
  },
}

```

```
    "number_price": {
      "type": "number",
      "minimum": 0,
      "maximum": 42946973,
      "multipleOf": 0.01
    },
    "number_three_format": {
      "type": "number",
      "minimum": 0,
      "maximum": 100000,
      "multipleOf": 0.001
    },
    "type_format": {
      "type": "number",
      "enum": [
        0,
        1,
        2,
        3,
        4,
        5,
        6,
        7,
        8,
        9
      ]
    }
  },
  "required": [
    "receipt"
  ]
}
```

## Структура параметров запроса

receipt

1. <u>client</u>	4.1. phones	5.9.4.4.inn
1.1. email		5.10. supplier_info
1.2. phone	5. <u>items</u>	5.10.1.phones
	5.1. name	5.10.2.name
2. <u>company</u>	5.2. price	5.10.3.inn
2.1. email	5.3. quantity	5.11. user_data
2.2. sno	5.4. sum	
2.3. inn	5.5. measurement_unit	6. <u>payments</u>
2.4. payment_address	5.6. payment_method	6.1.type
	5.7. payment_object	6.2.sum
3. <u>agent_info</u>	5.8. vat	
3.1. type	5.8.1.type	7. <u>vats</u>
3.2. paying_agent	5.8.2.sum	7.1.type
3.2.1.operation	5.9. agent_info	7.2.sum
3.2.2.phones	5.9.1.type	
3.3. receive_payments_operator	5.9.2.paying_agent	8.total
3.3.1.phones	5.9.2.1.operation	
3.4. money_transfer_operator	5.9.2.2.phones	9.additional_check_props
3.4.1.phones	5.9.3.receive_payments_operator	
3.4.2.name	5.9.3.1.phones	10.cashier
3.4.3.address	5.9.4.money_transfer_operator	
3.4.4.inn	5.9.4.1.phones	11. <u>additional_user_props</u>
	5.9.4.2.name	11.1.name
4. <u>supplier_info</u>	5.9.4.3.address	11.2.value

## Описание основных параметров запроса

Название поля	Тип поля	Обязательные поля	Ограничения	Тег ФФД
receipt	object	Да	Чек	-
client	object	Да	Атрибуты клиента.	-
company	object	Да	Атрибуты компании.	-

<b>agent_info</b>	object	Нет	Атрибуты агента.	
<b>items</b>	array of objects	Да	Атрибуты позиций. Ограничение по количеству от 1 до 100.	-
<b>payments</b>	array of objects	Да	Оплаты. Ограничение по количеству от 1 до 10.	-
<b>vats</b>	array of objects	Нет	Атрибуты налогов на чек. Ограничение по количеству от 1 до 6. Необходимо передать либо сумму налога на позицию, либо сумму налога на чек. Если будет переданы и сумма налога на позицию и сумма налога на чек, сервис учтет только сумму налога на чек.	
<b>total</b>	number	Да	Итоговая сумма чека в рублях с заданным в CMS округлением: целая часть не более 8 знаков; дробная часть не более 2 знаков. Сумму чека можно округлить, но не более, чем на 99 копеек. При регистрации в ККТ происходит расчёт фактической суммы: суммирование значений sum позиций.	1020 Сумма расчета, указанного в чеке (БСО)
<b>cashier</b>	string	Нет	ФИО кассира. Максимальная длина строки - 64 символа.	1021 Кассир
<b>additional_check_props</b>	object	Нет	Дополнительный реквизит пользователя.	1084 Дополнительный реквизит пользователя.

## Описание подпараметра client

Название поля	Тип поля	Обязательные поля	Ограничения	Тег ФФД
<b>client</b>	object	Да	Атрибуты клиента.	-
<b>email</b>	string	В запросе обязательно должно быть заполнено хотя бы одно из полей: email или phone. Если заполнены оба поля, ОФД отправит электронный чек только на email.	Электронная почта покупателя. Максимальная длина строки - 64 символа. В запросе обязательно должно быть заполнено хотя бы одно из полей: email или phone.	1008 Телефон или электронный адрес покупателя

<b>phone</b>	string	В запросе обязательно должно быть заполнено хотя бы одно из полей: email или phone. Если заполнены оба поля, ОФД отправит электронный чек только на email.	Телефон покупателя.	1008 Телефон или электронный адрес покупателя
--------------	--------	--	---------------------	--

### Описание подпараметра company

Название поля	Тип поля	Обязательные поля	Ограничения	Тег ФФД
<b>company</b>	object	Да	Атрибуты компании.	-
<b>email</b>	string	Да	Электронная почта отправителя чека. Максимальная длина строки - 64 символа	1117 Адрес электронной почты отправителя чека
<b>sno</b>	enum (string)	Поле необязательно, если у организации один тип налогообложения	Система налогообложения. Перечисление со значениями: «osp» - общая СН; «usn_income» - упрощенная СН (доходы); «usn_income_outcome» - упрощенная СН (доходы минус расходы); «envd» - единый налог на вмененный доход; «esn» - единый сельскохозяйственный налог; «patent» - патентная СН	1055 Применяемая система налогообложения
<b>inn</b>	string	Да	ИНН организации. Используется для предотвращения ошибочных регистраций чеков на ККТ зарегистрированных с другим ИНН (сравнивается со значением в ФН). Допустимое количество символов 10 или 12.	1018 ИНН пользователя
<b>payment_address</b>	string	Да	Место расчетов. Максимальная длина строки - 256 символов.	1187 Место расчетов

## Описание подпараметра agent\_info

Название поля	Тип поля	Обязательные поля	Ограничения	Тег ФФД
<b>agent_info</b>	object	Нет	Атрибуты агента.	-
<b>type</b>	enum (string)	Нет Если передан объект «agent_info», в нём обязательно должно быть передано поле «type».	Признак агента (ограничен агентами, введенными в ККТ при фискализации).	1057 Признак агента
<b>paying_agent</b>	object	Нет	Атрибуты платежного агента.	-
<b>operation</b>	string	Нет	Наименование операции. Максимальная длина строки - 24 символа.	1044 Операция платежного агента
<b>phones</b>	array of strings	Нет	Телефоны платежного агента. Максимальная длина одной строки массива - 19 символов	1073 Телефон платежного агента
<b>receive_payments_operator</b>	object	Нет	Атрибуты оператора по приему платежей.	-
<b>phones</b>	array of strings	Нет	Телефоны оператора по приему платежей. Максимальная длина одной строки массива - 19 символов.	1074 Телефон оператора по приему платежей
<b>money_transfer_operator</b>	object	Нет	Атрибуты оператора перевода.	-
<b>phones</b>	array of strings	Нет	Телефоны оператора перевода. Максимальная длина одной строки массива - 19 символов.	1075 Телефон оператора перевода
<b>name</b>	string	Нет	Наименование оператора перевода. Максимальная длина строки - 64 символа	1026 Наименование оператора перевода
<b>address</b>	string	Нет	Адрес оператора перевода. Максимальная длина строки - 256 символов	1005 Адрес оператора перевода
<b>inn</b>	string	Нет	ИНН оператора перевода. Максимальная длина строки - 12 символов.	1016 ИНН оператора перевода

### Возможные значения type:

«**bank\_paying\_agent**» – банковский платежный агент. Оказание услуг покупателю (клиенту) пользователем, являющимся банковским платежным агентом.



«**bank\_paying\_subagent**» – банковский платежный субагент. Оказание услуг покупателю (клиенту) пользователем, являющимся банковским платежным субагентом.

«**paying\_agent**» – платежный агент. Оказание услуг покупателю (клиенту) пользователем, являющимся платежным агентом.

«**paying\_subagent**» – платежный субагент. Оказание услуг покупателю (клиенту) пользователем, являющимся платежным субагентом.

«**attorney**» – поверенный. Осуществление расчета с покупателем (клиентом) пользователем, являющимся поверенным.

«**commission\_agent**» – комиссионер. Осуществление расчета с покупателем (клиентом) пользователем, являющимся комиссионером.

«**another**» – другой тип агента. Осуществление расчета с покупателем (клиентом) пользователем, являющимся агентом и не являющимся банковским платежным агентом (субагентом), платежным агентом (субагентом), поверенным, комиссионером.

#### Описание подпараметра supplier\_info

Название поля	Тип поля	Обязательные поля	Ограничения	Тег ФФД
<b>supplier_info</b>	object	Нет. Поле обязательно, если передан «agent_info»	Атрибуты поставщика.	-
<b>phones</b>	array of strings	Нет	Телефоны поставщика. Максимальная длина одной строки массива - 19 символов.	1171 Телефон поставщика

#### Описание подпараметра items

Название поля	Тип поля	Обязательные поля	Ограничения	Тег ФФД
<b>items</b>	array of objects	Да	Атрибуты позиций. Ограничение по количеству от 1 до 100.	-

<code>name</code>	string	Да	<p>Наименование товара.  Максимальная длина строки - 128 символов.  Если поле <code>payment_object</code> имеет значение <code>"nonoperating_gain"</code> для данного предмета расчета, то поле <code>"name"</code> должно принимать значение от 1 до 25.  Если поле <code>"payment_object"</code> имеет значение <code>"insurance_premium"</code> для данного предмета расчета, то поле <code>"items"-&gt;"name"</code> должно принимать значение от 26 до 31.  Описание значений поля <code>"name"</code> согласно ФФД приведено в Приложении 1.</p>	1030 Наименование предмета расчета
<code>price</code>	number	Да	<p>Цена в рублях:  целая часть не более 8 знаков;  дробная часть не более 2 знаков.  Максимальное значение цены - 42 949 672.95.  При этом произведение цены и количество/веса (<code>price*quantity</code>) позиции должно быть не больше максимального значения цены позиции.</p>	1079 Цена за единицу предмета расчета с учетом скидок и наценок
<code>quantity</code>	number	Да	<p>Количество/вес:  целая часть не более 5 знаков;  дробная часть не более 3 знаков.  Максимальное значение - 99 999.999</p>	1023 Количество предмета расчета
<code>sum</code>	number	Да	<p>Сумма в рублях:  целая часть не более 8 знаков;  дробная часть не более 2 знаков.</p>	1043 Стоимость предмета расчета с учетом скидок и наценок
<code>measurement_unit</code>	string	Нет	<p>Единица измерения товара, работы, услуги, платежа, выплаты, иного предмета расчета. Максимальная длина строки - 16 символов.</p>	1197 Единица измерения предмета расчета
<code>payment_method</code>	enum (string)	Нет Если признак не передан, по умолчанию используется значение <code>«full_prepayment»</code> .	<p>Признак способа расчёта.</p>	1214 Признак способа расчета

<code>payment_object</code>	enum (string)	Нет Если признак не передан, по умолчанию используется значение «commodity».	Признак предмета расчёта. Возможные значения указаны ниже.	1212 Признак предмета расчета
<code>vat</code>	object	Да	Атрибуты налога на позицию. Необходимо передать либо сумму налога на позицию, либо сумму налога на чек. Если будут переданы и сумма налога на позицию и сумма налога на чек, сервис учтет только сумму налога на чек	-
<code>type</code>	enum (string)	Да	Устанавливает номер налога в ККТ.	1199 Ставка НДС
<code>sum</code>	number	Нет	Сумма налога позиции в рублях: целая часть не более 8 знаков; дробная часть не более 2 знаков.	1200 Сумма НДС за предмет расчета
<code>agent_info</code>	object	Нет	Атрибуты агента. Если объект не передан, по умолчанию флаг агента не устанавливается.	-
<code>type</code>	enum (string)	Нет Если передан объект «agent_info», в нём обязательно должно быть передано поле «type».	Признак агента по предмету расчёта (ограничен агентами, введенными в ККТ при фискализации).	1222 Признак агента по предмету расчета
<code>paying_agent</code>	object	Нет	Атрибуты платежного агента.	-
<code>operation</code>	string	Нет	Наименование операции. Максимальная длина строки - 24 символа.	1044 Операция платежного агента
<code>phones</code>	array of strings	Нет	Телефоны платежного агента.	1073 Телефон платежного агента
<code>receive_payments_operator</code>	object	Нет	Атрибуты оператора по приему платежей.	-
<code>phones</code>	array of strings	Нет	Телефоны оператора по приему платежей.	1074 Телефон оператора по приему платежей
<code>money_transfer_operator</code>	object	Нет	Атрибуты оператора перевода.	-
<code>phones</code>	array of strings	Нет	Телефоны оператора перевода.	1075 Телефон оператора перевода

<b>name</b>	string	Нет	Наименование оператора перевода.	1026 Наименование оператора перевода
<b>address</b>	string	Нет	Адрес оператора перевода.	1005 Адрес оператора перевода
<b>inn</b>	string	Нет	ИНН оператора перевода.	1016 ИНН оператора перевода
<b>supplier_info</b>	object	Нет. Поле обязательно, если передан «agent_info»	Атрибуты поставщика.	-
<b>phones</b>	array of strings	Нет	Телефоны поставщика.	1171 Телефон поставщика
<b>name</b>	string	Нет	Наименование поставщика	1225 Наименование поставщика
<b>inn</b>	string	Нет	ИНН поставщика	1226 ИНН поставщика
<b>user_data</b>	string	Нет	Дополнительный реквизит предмета расчета. Максимальная длина строки - 64 символа.	1191 Дополнительный реквизит предмета расчета

#### Возможные значения payment\_method:

«full\_prepayment» – предоплата 100%. Полная предварительная оплата до момента передачи предмета расчета.

«prepayment» – предоплата. Частичная предварительная оплата до момента передачи предмета расчета.

«advance» – аванс.

«full\_payment» – полный расчет. Полная оплата, в том числе с учетом аванса (предварительной оплаты) в момент передачи предмета расчета

«partial\_payment» – частичный расчет и кредит. Частичная оплата предмета расчета в момент его передачи с последующей оплатой в кредит.

«credit» – передача в кредит. Передача предмета расчета без его оплаты в момент его передачи с последующей оплатой в кредит.

«credit\_payment» – оплата кредита. Оплата предмета расчета после его передачи с оплатой в кредит (оплата кредита).

#### Возможные значения type:

«**bank\_paying\_agent**» – банковский платежный агент. Оказание услуг покупателю (клиенту) пользователем, являющимся банковским платежным агентом.

«**bank\_paying\_subagent**» – банковский платежный субагент. Оказание услуг покупателю (клиенту) пользователем, являющимся банковским платежным субагентом.

«**paying\_agent**» – платежный агент. Оказание услуг покупателю (клиенту) пользователем, являющимся платежным агентом.

«**paying\_subagent**» – платежный субагент. Оказание услуг покупателю (клиенту) пользователем, являющимся платежным субагентом.

«**attorney**» – поверенный. Осуществление расчета с покупателем (клиентом) пользователем, являющимся поверенным.

«**commission\_agent**» – комиссионер. Осуществление расчета с покупателем (клиентом) пользователем, являющимся комиссионером.

«**another**» – другой тип агента. Осуществление расчета с покупателем (клиентом) пользователем, являющимся агентом и не являющимся банковским платежным агентом (субагентом), платежным агентом (субагентом), поверенным, комиссионером.

#### Возможные значения payment\_object:

«**commodity**» – товар.

«**excise**» – подакцизный товар.

«**job**» – работа.

«**service**» – услуга.

«**gambling\_bet**» – ставка азартной игры.

«**gambling\_prize**» – выигрыш азартной игры.

«**lottery**» – лотерейный билет.

«**lottery\_prize**» – выигрыш лотереи.

«**intellectual\_activity**» – предоставление результатов интеллектуальной деятельности.

«**payment**» – платеж.

«**agent\_commission**» – агентское вознаграждение.

«**composite**» – составной предмет расчета.

«another» – иной предмет расчета.

«property\_right» – имущественное право. О передаче имущественных прав.

«non-operating\_gain» – внереализационный доход.

«insurance\_premium» – страховые взносы.

«sales\_tax» – торговый сбор.

«resort\_fee» – курортный сбор.

### Описание подпараметра payments

Название поля	Тип поля	Обязательные поля	Ограничения	Тег ФФД
payments	array of objects	Да	Оплаты. Ограничение по количеству от 1 до 10.	-
type	enum (number)	Да	Вид оплаты. Возможные значения: «1» - электронный; «2» - предварительная оплата (аванс); «3» - постоплата (кредит); «4» - иная форма оплаты (встречное предоставление); «5» - «9» - расширенные виды оплаты. Для каждого фискального типа оплаты можно указать расширенный вид оплаты.	1081 Сумма по чеку электронными;
sum	number	Да	Сумма к оплате в рублях: целая часть не более 8 знаков; дробная часть не более 2 знаков.	1215 Сумма по чеку предоплатой (зачет аванса и (или) предыдущих платежей); 1216 Сумма по чеку постоплатой (кредит); 1217 Сумма по чеку встречным представлением.

## Описание подпараметра vats

Название поля	Тип поля	Обязательные поля	Ограничения	Тег ФФД
<b>vats</b>	array of objects	Нет	Атрибуты налогов на чек. Ограничение по количеству от 1 до 6. Необходимо передать либо сумму налога на позицию, либо сумму налога на чек. Если будет переданы и сумма налога на позицию и сумма налога на чек, сервис учтет только сумму налога на чек.	
<b>type</b>	enum (string)	Нет Если передан объект «vats», в нём обязательно должно быть переданы поля «type» и «sum».	Устанавливает номер налога в ККТ.	1105 Сумма расчета по чеку без НДС; 1104 Сумма расчета по чеку с НДС по ставке 0%; 1103 Сумма НДС чека по ставке 10%; 1102 Сумма НДС чека по ставке 18%; 1107 Сумма НДС чека по расч. ставке 10/110; 1106 Сумма НДС чека по расч. ставке 18/118.
<b>sum</b>	number		Сумма налога позиции в рублях: целая часть не более 8 знаков; дробная часть не более 2 знаков	1081 Сумма по чеку электронными;

### Возможные значения type:

«none» - без НДС; «vat0» - НДС по ставке 0%; «vat10» - НДС чека по ставке 10%; «vat18» - НДС чека по ставке 18%; «vat110» - НДС чека по расчетной ставке 10/110; «vat118» - НДС чека по расчетной ставке 18/118; «vat20» - НДС чека по ставке 20%; «vat120» - НДС чека по расчетной ставке 20/120.

С 01.02.2019 00:00 при отправке ставки vat18 или vat118 в чеках приход и расход сервис будет возвращать ошибку IncomingValidationException с текстом "Передана некорректная ставка налога. С 01.02.2019 ставки НДС 18 и 18/118 не могут использоваться в чеках sell (приход) и buy (расход)".

## Описание подпараметра additional\_user\_props

Название поля	Тип поля	Обязательные поля	Ограничения	Тег ФФД
<b>additional_user_props</b>	object	Нет	Дополнительный реквизит пользователя.	1084 Дополнительный реквизит пользователя.
<b>name</b>	string	Нет Если передан объект «additional_user_props», в нём обязательно должно быть передано поле «name»	Наименование дополнительного реквизита пользователя. Максимальная длина строки - 64 символа.	1085 Наименование дополнительного реквизита пользователя
<b>value</b>	string	Нет Если передан объект «additional_user_props», в нём обязательно должно быть передано поле «name»	Значение дополнительного реквизита пользователя. Максимальная длина строки - 256 символов.	1086 Значение дополнительного реквизита пользователя

С последней версией протокола АТОЛ можно ознакомиться [на официальном сайте АТОЛ.](#)



## Пример JSON

```
{
  "receipt": {
    "client": {
      "email": "kkt@kkt.ru"
    },
    "company": {
      "email": "chek@romashka.ru",
      "sno": "osn",
      "inn": "1234567891",
      "payment_address": "http://magazin.ru/"
    },
    "items": [
      {
        "name": "колбаса Клинский Брауншвейгская с/к в/с ",
        "price": 1000.0,
        "quantity": 0.3,
        "sum": 300.0,
        "measurement_unit": "кг",
        "payment_method": "full_payment",
        "payment_object": "commodity",
        "vat": {
          "type": "vat18"
        }
      },
      {
        "name": "яйцо Окское куриное С0 белое",
        "price": 100.0,
        "quantity": 1.0,
        "sum": 100.0,
        "measurement_unit": "Упаковка 10 шт.",
        "payment_method": "full_payment",
        "payment_object": "commodity",
        "vat": {
          "type": "vat10"
        }
      }
    ],
    "payments": [
      {
        "type": 1,
        "sum": 400.0
      }
    ],
    "vats": [
      {
        "type": "vat18",
        "sum": 45.76
      },
      {
        "type": "vat10",
        "sum": 9.09
      }
    ],
    "total": 400.0
  }
}
```

## История изменений

Версия	Изменения	Дата
1.5	<ul style="list-style-type: none"><li>Добавлен метод аннулирования счета</li></ul>	01.09.2018
1.5.1	Изменился формат ответов для следующих операций: <ul style="list-style-type: none"><li>создание счета на оплату (кроме счетов с методом доставки <b>BROWSER</b>);</li><li>возврат денежных средств по счету;</li><li>аннулирование счета</li></ul>	08.10.2018
1.6	<ul style="list-style-type: none"><li>добавлены параметры <code>PIMPAY_INVOICE_CURRENCY</code>, <code>PIMPAY_OFD_DATA</code> в запрос <code>CreateInvoice</code>;</li><li>расширено уведомление об оплате счета (<code>PIMPAY_CURRENCY</code>, <code>PIMPAY_FINAL_AMOUNT</code>);</li><li>добавлен пример формирования электронной подписи на Python;</li><li>добавлены данные для отправки чеков в налоговую в соответствии с ФЗ-54</li></ul>	04.12.2018
1.7	<ul style="list-style-type: none"><li>изменены данные для отправки чеков в налоговую в соответствии с ФЗ-54.</li></ul>	30.01.2019
1.7.1	<ul style="list-style-type: none"><li>добавлена служебная информация в <code>PIMPAY_CUSTOM_DATA</code>, которая будет отправляться в уведомлении о проведении платежа в случае оплаты счета через "Почту России";</li><li>добавлен пример ответа в параметре <code>PIMPAY_CUSTOM_DATA</code>, содержащий служебную информацию в обратном вызове магазина в случае оплаты счета через "Почту России"</li></ul>	20.02.2019
1.8	Добавлены <ul style="list-style-type: none"><li>метод подтверждения платежа, используемый при двухстадийной оплате;</li><li>метод расходирования платежа, используемый при двухстадийной оплате</li></ul>	04.10.2019