

---

# TicketsCloud

*Выпуск v1-beta*

июн. 21, 2022



<b>1</b>	<b>Простая интеграция для распространителей</b>	<b>3</b>
1.1	Базовые понятия	3
1.1.1	HTTP	3
1.1.2	Авторизация по ключу	4
1.1.3	DSL схемы ответа	5
1.1.4	Сервера	5
1.2	Работа с мероприятиями	5
1.2.1	/v1/services/simple/events	5
1.2.2	/v1/resources/events/:id/tickets	7
1.2.3	VEVENT	8
1.2.4	Media	8
1.2.5	Места	8
1.3	Работа с заказом	8
1.3.1	Жизненный цикл	8
1.3.2	Действия	10
1.4	Платёжные системы	12
1.4.1	api/BASE	12
1.4.2	Оплата через виджет	12
1.5	Возврат билетов	12
1.5.1	Создание запроса на возврат	12
1.5.2	Подтверждение/отклонение запроса на возврат	13
1.5.3	Список запросов на возврат	13
1.6	Города	13
1.7	Страны	14
1.8	Примеры	14
1.8.1	fields-schema	14
1.8.2	/v1/services/simple/events	16
1.8.3	Возврат билетов	21
1.8.4	Получение информации по билетам	23
1.8.5	Пример использования	23
1.8.6	Получение юр. информации об организаторе	24
1.8.7	Создание и изменение заказа	26
1.8.8	Проведение заказа	30
1.8.9	Ручное подтверждение заказа	33
1.8.10	Флаг <code>all_or_nothing</code>	36
1.8.11	Отмена заказа	38

1.8.12	Использование промокодов . . . . .	40
<b>2</b>	<b>Глоссарий</b>	<b>45</b>
	Алфавитный указатель	47

Содержание:



---

## Простая интеграция для распространителей

---

Для того, чтобы продать *билет* через TicketsCloud, нужно пройти следующие шаги.

1. Зарегистрировать кампанию на сайте → <https://ticketscloud.org>.
2. Получить *API-ключ* от менеджеров TicketsCloud → *авторизация*
3. Заключить *сделку* с *организатором* на *мероприятие* → работа со сделками
4. Получить информацию о мероприятии → *работа с мероприятиями*
5. **Создать и провести заказ** → *работа с заказом*
  - a) *Создание заказа*
  - b) *Бронирование билетов*
  - c) *Передача сервису информации о покупателе*
  - d) *Указание платёжной системы*
  - e) *Ожидание оплаты*
  - f) *Подтверждение оплаты*

Содержание:

### 1.1 Базовые понятия

В этом разделе описаны общие положения API TicketsCloud.

#### 1.1.1 HTTP

В TicketsCloud используется стандартный JSON REST API.

### Методы

В TicketsCloud используются следующие методы:

- **GET**  
Получение объектов.  
В ресурсах используется, как для коллекций (например: GET /v1/resources/events), так и для конкретных объектов (например: GET /v1/resources/events/:id).
- **POST**  
Создание нового объекта. Применяется к коллекциям. В ответе возвращается созданный объект.
- **PATCH**  
Изменение созданного объекта. Применяется к объекту. В ответе возвращается изменённый объект.
- **DELETE**  
Удаление или действие отмены объекта. Применяется к объекту. Ответ с пустым телом и кодом 200.

### Заголовки HTTP-запросов

- **Accept**  
Крайне желательно указывать этот заголовок как *application/json*.
- **Content-Type**  
Крайне желательно указывать этот заголовок как *application/json*.
- **Authorization**  
Заголовок авторизации.
- **X-Partner**  
Id авторизованного партнёра.

### Коды ответа

Коды ответа стандартные и описаны в спецификации протокола HTTP (**RFC 2616**).

### Тело запроса и ответа

Если не указано иного, в теле запросов и ответов используется **JSON**. При работе с коллекциями, возвращается массив объектов, при работе с конкретным объектом – сам объект.

#### 1.1.2 Авторизация по ключу

Для авторизации по ключу, значение *заголовка «Authorization»* имеет префикс **key** (key<пробел>).

Пример заголовка: Authorization: key 0123456789abcdef0123456789abcdef



### 1.1.3 DSL схемы ответа

Для того, чтобы ограничить или расширить ответ, применяется «схема». Она, схема, указывается в query параметре `fields-schema` и имеет следующую структуру: `field_1,field_2,reference{ref_field_1,ref_fields_2}`. Схему можно применять как для GET-запросов, так и для POST/PATCH, точно так же передавая её в URL.

Не рекомендуется делать очень грубокие схемы для запросов к коллекциям, потому что в таком случае запрос будет строиться дольше и может быть отменён по таймауту.

*Пример использования.*

### 1.1.4 Сервера

Наш продакшен сервер доступен по адресу <https://api.ticketcloud.org>. Не рекомендуется использовать его для отладки интеграции.

Для этих целей есть стейджинг сервер, доступный по адресу <https://stage.ticketcloud.org>. Данные на этом сервере каждую ночь сбрасываются.

## 1.2 Работа с мероприятиями

В этом разделе описывается упрощённая, но достаточная в большинстве случаев интеграции, модель работы с мероприятиями.

### 1.2.1 /v1/services/simple/events

Запрос GET `/v1/services/simple/events` позволяет простым способом всю необходимую получить информацию по всем мероприятиям, доступным для продажи *авторизованному* партнёру.

*Пример запроса.*

#### Основные поля

- id** id мероприятия
- created\_at** дата создания
- updated\_at** дата последнего изменения
- lifetime** *vevent*, время проведения мероприятия
- status** текущий статус мероприятия (`public` — публичное мероприятие, можно продавать билеты)
- title**
  - text** название мероприятия
  - desc** описание мероприятия
- media** *media* логотипы в различных размерах:
  - cover\_original**
  - cover**

**cover\_small**

**org** информация об организаторе мероприятия

- id** id организатора
- name** название организатора
- desc** краткое описание
- media** логотипы (*media*)
- contact** контактная информация

**venue** место проведения

- id**
- address** адрес
- country** *страна*
- city** *город*
- name** название
- desc** краткое описание
- point** координата (GeoJSON's point)

**map** схема зала

- id**

**sets** билетные категории

- id**
- pos** int, порядковый номер категории (для сортировки)
- name** название категории
- amount** общее количество билетов в сети
- amount\_vacant** количество билетов, доступных для продажи
- price\_org** номинальная цена билета
- price\_extra** сервисный сбор
- price** общая цена билета
- with\_seats** наличие посадочных мест в категории
- seats** *места*
- sector** сектор
- rules** правила
  - id**
  - cal** *vevent*, время действия правила
  - current** true, если правило текущее
  - price\_org** номинальная цена
  - price\_extra** сервисный сбор
  - price** конечная цена

**partner** устаревшее поле, будет удалено в будущем

### Фильтры

Фильтры добавляются в виде параметров URL.

**org** id организатора;

**venue** id площадки;

**country** страна;

**city** город;

**lifetime\_\_range** фильтр по времени проведения мероприятия, указывается двумя ISO-датами через запятую;

**price\_\_range** фильтр по стоимости, указывается двумя числами, через запятую;

### Паджинация

**page\_size** количество мероприятий на странице;

**page** номер страницы;

### Сортировка

Сортировка производится указанием параметра **sort** с соответствующим значением. Каждое значение может начинаться с символа **-** для изменения порядка сортировки.

Может быть указано несколько элементов сортировки одновременно, через запятую:

```
GET /v1/services/simple/events?sort=-price,start
```

**start**

**end**

**price**

## 1.2.2 /v1/resources/events/:id/tickets

Получение списка билетов мероприятия, для категорий с рассадкой.

### Поля

**id** id билета

**status** одно из **vacant**, **reserved** или **sold**

**set** билетная категория

**reserved\_till** если статус **reserved**, то это время окончания конца резервирования

**seat** Информация о месте проведения

**row** ряд

**number** место

**sector** сектор карты

Пример запроса.

## Фильтры

Фильтры добавляются в виде параметров URL.

**status** статус билета или список статусов через запятую;

### 1.2.3 VEVENT

Поле типа vevent, это строка в формате VEVENT из [RFC 2445](#). В настоящий момент поддерживается только два значения: DTSTART и DTEND. Значения указываются только в [UTC](#)

Пример:

```
BEGIN:VEVENT\r\n
DTSTART;VALUE=DATE-TIME:20160124T160000Z\r\n
DTEND;VALUE=DATE-TIME:20160124T173000Z\r\n
END:VEVENT\r\n
```

### 1.2.4 Media

Различные медиа-данные имеют общий формат.

**id**

**author** id создателя

**content\_type** тип файла (например, «image/jpeg»)

**length** размер в байтах

**md5hash** хеш md5 от содержимого

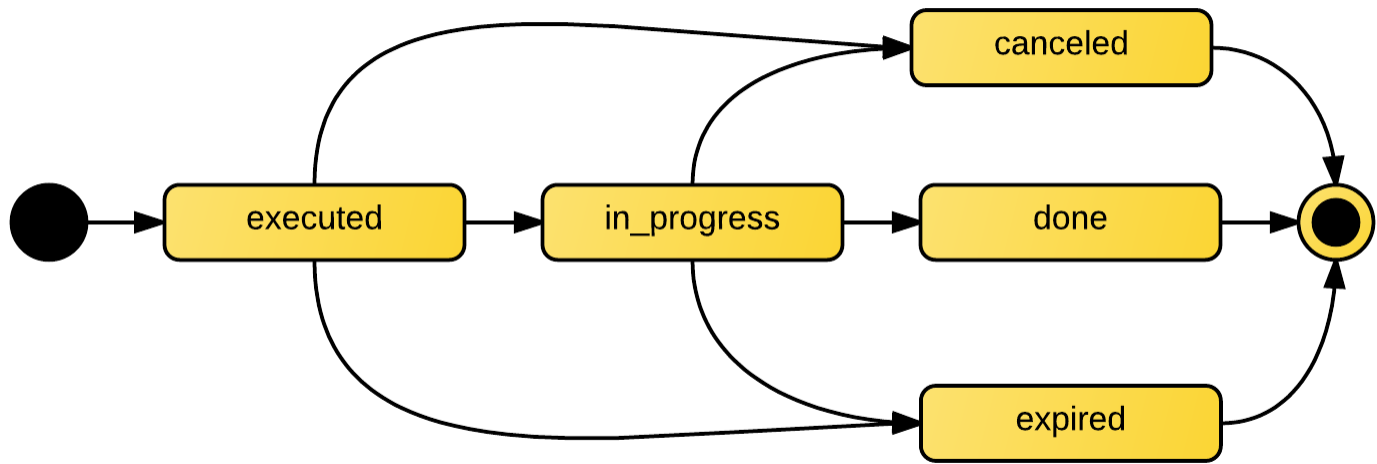
**url** полный урл до файла

### 1.2.5 Места

## 1.3 Работа с заказом

### 1.3.1 Жизненный цикл

### Диаграмма состояний (статусов) заказа



#### executed

Заказ всегда создаётся в статусе **executed**. Через 10 минут, заказ в этом состоянии переходит в *expired*.

С заказом в состоянии **executed** возможны следующие *действия*:

- забронировать билеты;
- добавить информацию о покупателе;
- выбрать платёжную систему;
- отменить;
- перевести в состояние *in\_progress*;

#### in\_progress

Это состояние ожидания подтверждения оплаты. Время бронирования увеличивается с учётом *выбранной платёжной системы*.

Возможные *действия*:

- убрать часть билетов из заказа;
- отменить;
- перевести в состояние *done*;

#### done

Заказ в этом статусе уже оплачен. Инициатор перехода в этот статус зависит от *выбранной платёжной системы*.

### canceled

Заказ отенён. Ничего сделать с ним нельзя.

### expired

Заказ просрочен, забронированные билеты снова вернулись в продажу.

## 1.3.2 Действия

Все действия с заказом, кроме его создания, делаются по запросу `PATCH /v1/resources/orders/:id`. В один запрос одновременно может быть добавлено несколько действий.

Все запросы на изменение конкретного заказа, должны делаться синхронно. В случае получения запроса до конца обработки предыдущего, будет возвращена ошибка `409 Conflict`. Это ограничение касается только работы с одним заказом, а работать одновременно с несколькими заказами можно.

### Создание

Заказ создаётся запросом `POST /v1/resources/orders`. Возможно добавление всех действий, в статусе *executed*.

### Бронирование билетов

За бронирование билетов отвечают три поля:

- *tickets*
- *random*
- *all\_or\_nothing*

#### Поле tickets

В поле **tickets** передаются все *id билетов*, которые должны быть забронированы текущим *заказом*. Если заказ изменяется (покупатель решил добавить ещё один билет), то в обязательном порядке передаются *все* билеты, которые должны быть в заказе (в т.ч. те, что уже забронированы). Для удаления конкретного билета из заказа, нужно передать все билеты, кроме удаляемого.

Нельзя использовать в одном запросе с **random**.

- *Пример запросов.*

#### Поле random

Поле **random** предназначено для бронирования случайных билетов из указанных *категорий*. Оно нужно для того, чтобы добавлять в заказ билеты без мест. Имеет вид объекта, где ключ является *id категории*, а значение — количество билетов.

Так же, как и в **tickets**, всегда нужно передавать желаемое состояние. Т.е. если пользователь удалил один билет из категории, то передать надо **random** со всеми категориями и количествами, только в одной из категорий будет на один билет меньше.

В ответе от сервера всегда будет список забронированных билетов в поле `tickets`.

Нельзя использовать в одном запросе с `tickets`.

- *Пример запросов.*

### Поле `all_or_nothing`

Если поле `all_or_nothing` `true`, то бронируются либо все билеты, либо ни одного. При изменении заказа с одновременным разбронированием и бронированием билетов, в случае неудачи с бронированием хотя бы одного билета, разбронирования не происходит, т.е. список забронированных билетов не изменяется.

Если значение не указано, или `false`, то билеты, которые не удалось забронировать пропускаются и отсутствуют поле `tickets` в ответе.

Можно использовать, как с `tickets`, так и с `random`.

- *Пример запросов.*

### Добавление информации о покупателе

Информация о покупателе содержится в поле `customer`. В ней есть следующие поля:

**email** email пользователя

**first\_name** имя

**last\_name** фамилия

**mobile** телефон

В зависимости от *выбранной платёжной системой*, некоторых из этих полей могут быть обязательными для завершения заказа.

### Выбор платёжной системы

*Платёжные системы* настраиваются в поле `payment` со следующими подполями:

**system** тип платёжной системы, передаётся либо `id`, либо строкой из *списка*;

**send\_tickets** отсылать ли письмо о покупке (если не установлено, зависит от *платёжной системы*);

- *Пример работы с заказом*

### Смена статуса

Смена статуса (состояния заказа) осуществляется путём установки поля `status`. При попытке перевести заказ в состояние, не соответствующее *диаграмме состояний*, сервер вернёт ответ с кодом 400 `BadRequest`.

Нельзя вручную перевести заказ в статус `expired`. Возможность перехода из `in_progress` в `done` определяется *выбранной платёжной системой* и может быть ограничена.

- *Пример запросов.*

## Отмена

Отмена заказа, это *установка статуса canceled*. Очень важно отменять заказ, как только это возможно, чтобы билеты вернулись в продажу.

- *Пример запросов.*

## Частичная оплата

В статусе заказа *in\_progress*, можно часть билетов убрать из заказа. Например, если пользователь оплатил только часть билетов.

Для разбронирования, нужно сделать запрос с *полем tickets* с оставшимися билетами.

## 1.4 Платёжные системы

### 1.4.1 api/BASE

*Платёжная система api/BASE* используется для проведения заказа по api. В этом случае деньги за заказ каким-либо образом берёт сам продавец.

- *Пример работы с заказом без платрона*

### 1.4.2 Оплата через виджет

Для оплаты любого заказа можно открыть виджет сразу на втором шаге со всеми доступными способами оплаты. Чтобы это сделать необходимо сформировать URL-виджета обычного виджета, которые доступны в личном кабинете и добавить GET-параметр *order* с id созданного *заказа*.

#### Структура:

`https://ticketscloud.org/v1/widgets/common?token=r:token&event=:event&order=:order`

**token** можно получить в ЛК в коде вставки любого виджета

**event** id мероприятия

**order** id заказа

## 1.5 Возврат билетов

Когда заказ полностью оформлен, билеты можно вернуть в продажу.

### 1.5.1 Создание запроса на возврат

Нужно сделать запрос POST `/v1/resources/refund_requests` со следующими данными:

**order** id заказа;

**culprit** *виновник* (инициатор) возврата;



**tickets** список id билетов, которые должны остаться в заказе (конфликтует с **tickets\_refund**);

**tickets\_refund** список id билетов, которые нужно вернуть в заказе (конфликтует с **tickets**);

Будет создан запрос на возврат в статусе **new**. Для одного заказа может быть создано несколько запросов.

### Виновник возврата

Возврат может быть по различным причинам. Виновник возврата указывается в поле **culprit** при создании запроса на возврат.

**user** виноват в возврате пользователь (например, сдал билет);

**org** виноват организатор (например, отмена мероприятия);

## 1.5.2 Подтверждение/отклонение запроса на возврат

Запрос PATCH /v1/resources/refund\_requests/:id:

**status** approved или rejected.

## 1.5.3 Список запросов на возврат

По запросу GET /v1/resources/refund\_requests возвращается список запросов на возврат билетов.

### Фильтры

**status** фильтр по статусу запроса;

**event** запросы по конкретному мероприятию;

**org** запросы по конкретному организатору (актуально для агентов);

**vendor** по конкретному продавцу (для организаторов);

### Сортировка

**created\_at** по времени создания запроса (по умолчанию);

**value** по стоимости возвращаемых билетов;

## 1.6 Города

```
{
  "country": "RU",
  "id": 524901,
  "name": {
    "be": "Горад Масква",
    "default": "Moscow",
```

(continues on next page)

(продолжение с предыдущей страницы)

```
    "en": "Moscow",
    "fr": "Moscou",
    "ru": "Москва",
    "zh": ""
  },
  "timezone": "Europe/Moscow"
}
```

## 1.7 Страны

```
{
  "id": "RU",
  "name": {
    "be": "Расійская Федэрацыя",
    "default": "Russia",
    "en": "Russia",
    "fr": "Russie",
    "ru": "Россия",
    "zh": ""
  }
}
```

## 1.8 Примеры

### 1.8.1 fields-schema

*Основное описание.*

Запрос к заказу без схемы, возвращает все поля в максимально простом виде:

**Request:**

```
GET /v1/resources/orders/56a69feaf06c5a21c3938057 HTTP/1.1
Host: api.ticketscloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json
```

**Response:**

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a69feaf06c5a21c3938057",
  "created_at": "2016-01-25T22:21:30.737000+00:00",
  "customer": {
    "email": "volkov@alex.com",
    "first_name": "Алексей",
    "lang": null,
```

(continues on next page)

(продолжение с предыдущей страницы)

```

      "last_name": "Волков",
      "mobile": "89167683232",
      "user": "56a6a01841226dbf3b346180"
    },
    "deal": null,
    "event": "568a22b5f06c5a42975fb913",
    "number": 497291895,
    "org": "56810047f06c5a6ac62f4e1d",
    "payment": {
      "failure_url": "http://ticketscloud.org/",
      "redirect_url": "https://www.goodservice.ru/ps/test/start_payment.php?payment_id=23324826",
      "success_url": "http://ticketscloud.org/?tcordersuccess=497291895",
      "system": "545b544a5d645a463e779d53"
    },
    "promocodes": [],
    "reserved_till": "2016-01-25T22:52:21.062000+00:00",
    "rules": {
      "568a22b6f06c5a42985fb914": "56a61828f06c5a059b937fdc"
    },
    "status": "in_progress",
    "tickets": [
      "568a22b6f06c5a42e8847c55"
    ],
    "updated_at": "2016-01-25T22:22:21.063000+00:00",
    "value": "100.00",
    "value_extra": "0.00",
    "values": {
      "extra": "0.00",
      "full": "100.00",
      "nominal": "100.00"
    },
    "vendor": "56810047f06c5a6ac62f4e1d"
  }
}

```

Запрос со схемой, раскрывающий информацию о билетах в заказе и название мероприятия:

#### Request:

```

GET /v1/resources/orders/56a69feaf06c5a21c3938057?fields-schema=id,tickets%7Bid,serial,number,seat
↪%7Brow,number%7D%7D,event%7Btitle%7Btext%7D%7D HTTP/1.1
Host: ticketscloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

```

#### Response:

```

HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a69feaf06c5a21c3938057",
  "event": {
    "title": {
      "text": "test"
    }
  }
}

```

(continues on next page)

(продолжение с предыдущей страницы)

```

    },
    "tickets": [
      {
        "id": "568a22b6f06c5a42e8847c55",
        "number": 118398,
        "seat": {},
        "serial": "AEY"
      }
    ]
  }
}

```

## 1.8.2 /v1/services/simple/events

Request:

```

GET /v1/services/simple/events HTTP/1.1
Host: api.ticketcloud.org
Accept: */*
Authorization: key 047bdb8bcee44d3693371920aaf9135c

```

Response:

```

HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

[
  {
    "id": "568a22b5f06c5a42975fb913",
    "age_rating": 0,
    "created_at": "2016-01-04T07:43:49.793000+00:00",
    "deal": null,
    "lifetime": "BEGIN:VEVENT\r\nDTSTART;VALUE=DATE-TIME:20160127T210000Z\r\nDTEND;VALUE=DATE-TIME:20160128T205900Z\r\nEND:VEVENT\r\n",
    "map": null,
    "media": {},
    "org": {
      "id": "56810047f06c5a6ac62f4e1d",
      "contact": {
        "address": "Россия г. Москва, ул. Жуковского д. 17 к. 2, 6UYDD",
        "email": "vsuharnikov+6uydd@gmail.com",
        "name": "",
        "phones": [
          "8-916-768-32-32"
        ],
        "www": "v_test_6UYDD.com"
      },
      "desc": "description",
      "media": {},
      "name": "v_test_6UYDD brand",
      "tags": []
    },
    "partner": {
      "id": "56810047f06c5a6ac62f4e1d",
      "contact": {

```

(continues on next page)

(продолжение с предыдущей страницы)

```

        "address": "Россия г. Москва, ул. Жуковского д. 17 к. 2, 6UYDD",
        "email": "vsuharnikov+6uydd@gmail.com",
        "name": "",
        "phones": [
            "8-916-768-32-32"
        ],
        "www": "v_test_6UYDD.com"
    },
    "desc": "description",
    "media": {},
    "name": "v_test_6UYDD brand",
    "tags": []
},
"sets": [
    {
        "id": "568a22b6f06c5a42985fb914",
        "amount": 3,
        "amount_vacant": 1,
        "name": "fdfd",
        "pos": 0,
        "price": "100.00",
        "price_extra": "0.00",
        "price_org": "100.00",
        "rules": [
            {
                "id": "56a61828f06c5a059b937fdc",
                "cal": "BEGIN:VEVENT\r\nDTSTART;VALUE=DATE-TIME:20160102T210000Z\r\nDTEND;
↪VALUE=DATE-TIME:20160128T205900Z\r\nEND:VEVENT\r\n",
                "current": true,
                "price": "100.00",
                "price_extra": "0.00",
                "price_org": "100.00"
            }
        ],
        "seats": null,
        "sector": null
    }
],
"status": "public",
"tags": [
    "ВЫСТАВКИ"
],
"ticket_template": {
    "fan_cover_url": null,
    "name": null,
    "text_color": null
},
"tickets_amount": 3,
"tickets_amount_vacant": 1,
"title": {
    "desc": "test",
    "text": "test"
},
"updated_at": "2016-01-25T13:33:04.583000+00:00",
"venue": {
    "id": "554111c09cb538793e6a3c37",

```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "address": "Пресненский вал, дом 6, строение 1",
    "city": {
      "id": 524901,
      "country": "RU",
      "name": {
        "be": "Горад Масква",
        "default": "Moscow",
        "en": "Moscow",
        "fr": "Moscou",
        "ru": "Москва",
        "zh": ""
      },
      "timezone": "Europe/Moscow"
    },
    "country": {
      "id": "RU",
      "name": {
        "be": "Расійская Федэрацыя",
        "default": "Russia",
        "en": "Russia",
        "fr": "Russie",
        "ru": "Россия",
        "zh": ""
      }
    },
    "desc": null,
    "name": "16 тонн",
    "point": {
      "coordinates": [
        37.564342000000001,
        55.764308000000001
      ],
      "type": "Point"
    }
  },
  {
    "id": "56a6253df06c5a059a93802e",
    "age_rating": 0,
    "created_at": "2016-01-25T13:38:05.007000+00:00",
    "deal": null,
    "lifetime": "BEGIN:VEVENT\r\nDTSTART;VALUE=DATE-TIME:20160512T200000Z\r\nDTEND;VALUE=DATE-
    ↪TIME:20160513T195900Z\r\nEND:VEVENT\r\n",
    "map": {
      "id": "54d79ee69cb538749c32c221",
      "desc": null,
      "name": "default",
      "sectors": [
        {
          "id": "54d7a0409cb538783b7bf8d5",
          "desc": null,
          "name": "Партер"
        },
        {
          "id": "54d7a0409cb538783b7bf8d6",
          "desc": null,

```

(continues on next page)

(продолжение с предыдущей страницы)

```

        "name": "Балкон"
    }
  ],
  "svg": {
    "map": {
      "id": "54d7a0409cb538783b7bf8d7",
      "author": null,
      "content_type": "image/svg+xml",
      "length": null,
      "md5hash": "9efaa9cf8af50f95a3ddf205f7bcebe0",
      "url": "https://s3-eu-west-1.amazonaws.com:443/media.ticketscloud/production/
↪map/2015-02/54d79ee69cb538749c32c221-54d79ee69cb538749c32c220.svg"
    },
    "mapz": {
      "id": "561c43a79cb5380fdcab402d",
      "author": null,
      "content_type": "image/svg+xml",
      "length": null,
      "md5hash": "c77114baa0912d621044e8ad17f8aede",
      "url": "https://s3-eu-west-1.amazonaws.com:443/media.ticketscloud/production/
↪map/2015-10/54d79ee69cb538749c32c221-54d79ee69cb538749c32c220.svgz"
    },
    "source": {
      "id": "54d79ee69cb538749c32c220",
      "author": null,
      "content_type": "image/svg+xml",
      "length": null,
      "md5hash": "d5bc921d747322b599f78987fa492c0b",
      "url": "https://s3-eu-west-1.amazonaws.com:443/media.ticketscloud/production/
↪maps/2015-02/54d79ee69cb538749c32c220.svg"
    }
  }
},
"media": {},
"org": {
  "id": "56810047f06c5a6ac62f4e1d",
  "contact": {
    "address": "Россия г. Москва, ул. Жуковского д. 17 к. 2, 6UYDD",
    "email": "vsuharnikov+6uydd@gmail.com",
    "name": "",
    "phones": [
      "8-916-768-32-32"
    ],
    "www": "v_test_6UYDD.com"
  },
  "desc": "description",
  "media": {},
  "name": "v_test_6UYDD brand",
  "tags": []
},
"partner": {
  "id": "56810047f06c5a6ac62f4e1d",
  "contact": {
    "address": "Россия г. Москва, ул. Жуковского д. 17 к. 2, 6UYDD",
    "email": "vsuharnikov+6uydd@gmail.com",
    "name": "",

```

(continues on next page)

(продолжение с предыдущей страницы)

```

        "phones": [
            "8-916-768-32-32"
        ],
        "www": "v_test_6UYDD.com"
    },
    "desc": "description",
    "media": {},
    "name": "v_test_6UYDD brand",
    "tags": []
},
"sets": [
    {
        "id": "56a6254bf06c5a059b93800c",
        "amount": 3,
        "amount_vacant": 3,
        "name": "Паптер",
        "pos": 0,
        "price": "100.00",
        "price_extra": "0.00",
        "price_org": "100.00",
        "rules": [
            {
                "id": "56a6254bf06c5a059b93800b",
                "cal": "BEGIN:VEVENT\r\nDTSTART;VALUE=DATE-TIME:20160123T200000Z\r\nDTEND;
↪VALUE=DATE-TIME:20160513T195900Z\r\nEND:VEVENT\r\n",
                "current": true,
                "price": "100.00",
                "price_extra": "0.00",
                "price_org": "100.00"
            }
        ],
        "seats": {
            "1": [
                [
                    1,
                    1
                ],
                [
                    3,
                    3
                ],
                [
                    6,
                    6
                ]
            ]
        ]
    },
    {
        "sector": "54d7a0409cb538783b7bf8d5"
    }
],
"status": "public",
"tags": [
    "экскурсии"
],
"ticket_template": {
    "fan_cover_url": null,

```

(continues on next page)



(продолжение с предыдущей страницы)

```

        "name": null,
        "text_color": null
    },
    "tickets_amount": 3,
    "tickets_amount_vacant": 3,
    "title": {
        "desc": null,
        "text": "With seats"
    },
    "updated_at": "2016-01-25T13:38:19.844000+00:00",
    "venue": {
        "id": "54d49b9df06c5a0dbde10e7f",
        "address": "443010, г. Самара, ул. Фрунзе, д. 141",
        "city": {
            "id": 499099,
            "country": "RU",
            "name": {
                "be": "Горад Самара",
                "default": "Samara",
                "en": "Samara",
                "fr": "Samara",
                "ru": "Самара",
                "zh": ""
            },
            "timezone": "Europe/Samara"
        },
        "country": {
            "id": "RU",
            "name": {
                "be": "Расійская Федэрацыя",
                "default": "Russia",
                "en": "Russia",
                "fr": "Russie",
                "ru": "Россия",
                "zh": ""
            },
            "timezone": ""
        },
        "desc": null,
        "name": "Самарская Государственная Филармония",
        "point": {
            "coordinates": [
                50.094984999999995,
                53.191517999999999
            ],
            "type": "Point"
        }
    }
}
]

```

### 1.8.3 Возврат билетов

Для возврата нам потребуется *заказ* в статусе *done*, его *id*, список *id* билетов из этого заказа, которые надо вернуть и виновник.

Допустим покупатель решил вернуть два билета из заказа.

Создаем *запрос на возврат*:

### Request:

```
POST /v1/resources/refund_requests HTTP/1.1
Host: api.ticketcloud.org
Accept: application/json
Authorization: key 968cd37341029c26d173b1113f0002cd
Content-Type: application/json

{
  "culprit": "user",
  "order": "5703d7542d709509727f929c",
  "tickets_refund": [
    "56ba2b6f9cb53851972a28db",
    "56ba2b6f9cb53851972a2883"
  ]
}
```

### Response:

```
HTTP/1.1 201 Created
Content-Type: application/json; charset=UTF-8
X-Partner: 54bcf9269cb53859759321c8

{
  "created_at": "2016-04-05T15:34:02.502481+00:00",
  "culprit": "user",
  "id": "5703daea2d709509727f92a5",
  "order": "5703d7542d709509727f929c",
  "org": "54bcf9269cb538597597b5c8",
  "payment_system": "545b544a5d645a463e779d53",
  "status": "new",
  "tickets_refund": [
    "56ba2b6f9cb53851972a28db",
    "56ba2b6f9cb53851972a2883"
  ],
  "tickets": [],
  "updated_at": "2016-04-05T15:34:02.502481+00:00",
  "value": "0.00",
  "vendor": "54bcf9269c3248597597b5c8"
}
```

После успешного создания запроса на возврат мы можем подтвердить его. Для этого нам понадобится только *id запроса на возврат*.

### Request:

```
PATCH /v1/resources/refund_requests/5703daea2d709509727f92a5 HTTP/1.1
Host: api.ticketcloud.org
Accept: application/json
Authorization: key 968cd37341029c26d173b1113f0002cd
Content-Type: application/json

{
  "status": "approved"
}
```

### Response:

```

HTTP/1.1 201 Created
Content-Type: application/json; charset=UTF-8
X-Partner: 54bcf9269cb53859759321c8

{
  "created_at": "2016-04-05T15:41:10.130000+00:00",
  "culprit": "user",
  "id": "5703daea2d709509727f92a5",
  "order": "5703d7542d709509727f929c",
  "org": "54bcf9269cb538597597b5c8",
  "payment_system": "545b544a5d645a463e779d53",
  "status": "approved",
  "tickets_refund": [
    "56ba2b6f9cb53851972a28db",
    "56ba2b6f9cb53851972a2883"
  ],
  "tickets": [],
  "updated_at": "2016-04-05T15:41:10.130000+00:00",
  "value": "0.00",
  "vendor": "54bcf9269c3248597597b5c8"
}

```

После успешного подтверждения *запроса на возврат* деньги будут возвращены в соответствии с логикой системы.

### 1.8.4 Получение информации по билетам

Для получения информации по билетам связанным с конкретным заказом нужно сделать запрос GET `/v1/resources/orders/:id/tickets`

На выходе получаем массив объектов со следующими полями:

- id** id билета
- barcode** штрихкод билета
- number** номер билета
- serial** серия билета
- seat** место (включает в себя подобъект с рядом, номером места и id сектора)
- set** id квоты в которой находится билет

### 1.8.5 Пример использования

Имеем:

**API-ключ** `113211f9c3082bbdb29dae7cda1eaf242b`

**ID заказа** `5829c30ad152860018a378b2`

Получаем информацию по билетам заказа:

**Request:**

```
GET /v1/resources/orders/5829c30ad152860018a378b2/tickets HTTP/1.1
Host: api.ticketcloud.org
Accept: */*
Authorization: key 113211f9c3082bbdb29dae7cda1eaf242b
```

### Response:

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: fa310da89cb5387ecf647810

[
  {
    "id": "582cc243235e3431252b1572",
    "barcode": "60018146012573969",
    "number": 151553,
    "seat": {
      "number": 27,
      "row": 12,
      "sector": "5592c3459cb538014c3a7a36"
    },
    "serial": "AKT",
    "set": "582cc78e515e3500142b128d"
  },
  {
    "id": "582cc291515234560152b1571",
    "barcode": "2572487563832285",
    "number": 151552,
    "seat": {
      "number": 26,
      "row": 12,
      "sector": "5592c3459cb538014c3a7a36"
    },
    "serial": "AKT",
    "set": "582cc78e515e3500142b128d"
  }
]
```

## 1.8.6 Получение юр. информации об организаторе

Для получения юридической информации об организаторе мероприятия необходимо сделать запрос на `GET /v1/resources/partners/:partner_id?fields-schema=legal{type,bank,detail,who}`

### Пример использования

Мы распространитель и у нас есть один оплаченный заказ. Нам необходимо получить информацию об организаторе. Имеем: :API-ключ: `0205be31ef3846d4a163ce71aa6d16b6` :ID заказа: `584e9ad50944c10017e3a9df`

Получаем id партнера организатор:

### Request:

```
GET /v1/resources/orders/584e9ad50944c10017e3a9df?fields-schema=org HTTP/1.1
Accept: */*
Authorization: key 0205be31ef3846d4a163ce71aa6d16b6
```

**Response:**

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: 584e8df10944c1001ae3a9e3

{
  "org": "584e8bc20944c10017e3a9d3"
}
```

Имея идентификатор партнера организатора можем получить его юридическую информацию:  
**Request:**

```
GET /v1/resources/partners/584e8bc20944c10017e3a9d3?fields-schema=legal%7Btype,bank,detail,who%7D HTTP/1.1
Accept: */*
Authorization: key 0205be31ef3846d4a163ce71aa6d16b6
```

**Response:**

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: 584e8df10944c1001ae3a9e3

{
  "legal": {
    "bank": {
      "bik": "432424234",
      "ks": "42343242342342242424",
      "name": "БанкЪ",
      "rs": "43242423424242334224"
    },
    "detail": {
      "address": "Супер орг",
      "inn": "423423643275",
      "name": "ИП Супер орг",
      "nds": false,
      "ogrnip": "423424234234322",
      "taxes": "osn",
      "type": "ru/ltd"
    },
    "type": "ru/ip",
    "who": {
      "name": "",
      "position": "",
      "reason": "на основании устава"
    }
  }
}
```

### 1.8.7 Создание и изменение заказа

Запрос на создание заказа одного конкретного билета:

Request:

```
POST /v1/resources/orders HTTP/1.1
Host: api.ticketcloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "event": "56a6253df06c5a059a93802e",
  "tickets": [
    "56a6253df06c5a059a9380aa"
  ]
}
```

Response:

```
HTTP/1.1 201 Created
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a6ae06f06c5a21c393805d",
  "created_at": "2016-01-25T23:21:42.154000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "56a6253df06c5a059a93802e",
  "number": 497291898,
  "org": "56810047f06c5a6ac62f4e1d",
  "payment": {
    "failure_url": null,
    "success_url": null,
    "system": null
  },
  "promocodes": [],
  "reserved_till": "2016-01-25T23:31:42.165000+00:00",
  "rules": {
    "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
  },
  "status": "executed",
  "tickets": [
    "56a6253df06c5a059a9380aa"
  ],
  "updated_at": "2016-01-25T23:21:42.166000+00:00",
  "value": "100.00",
  "value_extra": "0.00",
  "values": {
    "extra": "0.00",
    "full": "100.00",
    "nominal": "100.00"
  },
  "vendor": "56810047f06c5a6ac62f4e1d"
}
```

(continues on next page)

(продолжение с предыдущей страницы)

}

Запрос на добавление билета с заданным id в заказ:

**Request:**

```

PATCH /v1/resources/orders/56a6ae06f06c5a21c393805d HTTP/1.1
Host: api.ticketscloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "tickets": [
    "56a6253df06c5a059a9380aa",
    "56a6253df06c5a059a9380a4"
  ]
}
```

**Response:**

```

HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a6ae06f06c5a21c393805d",
  "created_at": "2016-01-25T23:21:42.154000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "56a6253df06c5a059a93802e",
  "number": 497291898,
  "org": "56810047f06c5a6ac62f4e1d",
  "payment": {
    "failure_url": null,
    "success_url": null,
    "system": null
  },
  "promocodes": [],
  "reserved_till": "2016-01-25T23:31:42.165000+00:00",
  "rules": {
    "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
  },
  "status": "executed",
  "tickets": [
    "56a6253df06c5a059a9380a4",
    "56a6253df06c5a059a9380aa"
  ],
  "updated_at": "2016-01-25T23:27:56.776000+00:00",
  "value": "200.00",
  "value_extra": "0.00",
  "values": {
    "extra": "0.00",
    "full": "200.00",
    "nominal": "200.00"
  }
}
```

(continues on next page)

(продолжение с предыдущей страницы)

```
  },
  "vendor": "56810047f06c5a6ac62f4e1d"
}
```

Запрос на создание заказа (Заказываем два случайных билета из заданного сета):

#### Request:

```
POST /v1/resources/orders HTTP/1.1
Host: api.ticketcloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "event": "56a6253df06c5a059a93802e",
  "random": {
    "56a6254bf06c5a059b93800c": 1
  }
}
```

#### Response:

```
HTTP/1.1 201 Created
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a6a93ef06c5a21c3938059",
  "created_at": "2016-01-25T23:01:18.612000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "56a6253df06c5a059a93802e",
  "number": 497291896,
  "org": "56810047f06c5a6ac62f4e1d",
  "payment": {
    "failure_url": null,
    "success_url": null,
    "system": null
  },
  "promocodes": [],
  "reserved_till": "2016-01-25T23:11:18.625000+00:00",
  "rules": {
    "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
  },
  "status": "executed",
  "tickets": [
    "56a6253df06c5a059a9380a0"
  ],
  "updated_at": "2016-01-25T23:01:18.625000+00:00",
  "value": "100.00",
  "value_extra": "0.00",
  "values": {
    "extra": "0.00",
    "full": "100.00",

```

(continues on next page)



(продолжение с предыдущей страницы)

```

    "nominal": "100.00"
  },
  "vendor": "56810047f06c5a6ac62f4e1d"
}

```

Запрос на изменение ранее созданного заказа (добавление еще одного случайного билета в заказ):

**Request:**

```

PATCH /v1/resources/orders/56a6a93ef06c5a21c3938059 HTTP/1.1
Host: api.ticketscloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "random": {
    "56a6254bf06c5a059b93800c": 2
  }
}

```

**Response:**

```

HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a6a93ef06c5a21c3938059",
  "created_at": "2016-01-25T23:01:18.612000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "56a6253df06c5a059a93802e",
  "number": 497291896,
  "org": "56810047f06c5a6ac62f4e1d",
  "payment": {
    "failure_url": null,
    "success_url": null,
    "system": null
  },
  "promocodes": [],
  "reserved_till": "2016-01-25T23:11:18.625000+00:00",
  "rules": {
    "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
  },
  "status": "executed",
  "tickets": [
    "56a6253df06c5a059a9380a4",
    "56a6253df06c5a059a9380a0"
  ],
  "updated_at": "2016-01-25T23:05:38.937000+00:00",
  "value": "200.00",
  "value_extra": "0.00",
  "values": {
    "extra": "0.00",

```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "full": "200.00",
    "nominal": "200.00"
  },
  "vendor": "56810047f06c5a6ac62f4e1d"
}

```

### 1.8.8 Проведение заказа

Запрос на создание заказа с одним случайным билетом из заданного сета:

**Request:**

```

POST /v1/resources/orders HTTP/1.1
Host: api.ticketscloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "event": "56a6253df06c5a059a93802e",
  "random": {
    "56a6254bf06c5a059b93800c": 1
  }
}

```

**Response:**

```

HTTP/1.1 201 Created
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a73b7ff06c5a21c393806b",
  "created_at": "2016-01-26T09:25:19.378000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "56a6253df06c5a059a93802e",
  "number": 497291903,
  "org": "56810047f06c5a6ac62f4e1d",
  "payment": {
    "failure_url": null,
    "success_url": null,
    "system": null
  },
  "promocodes": [],
  "reserved_till": "2016-01-26T09:35:19.392000+00:00",
  "rules": {
    "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
  },
  "status": "executed",
  "tickets": [
    "56a6253df06c5a059a9380a0"
  ],
}

```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "updated_at": "2016-01-26T09:25:19.393000+00:00",
    "value": "100.00",
    "value_extra": "0.00",
    "values": {
      "extra": "0.00",
      "full": "100.00",
      "nominal": "100.00"
    },
    "vendor": "56810047f06c5a6ac62f4e1d"
  }
}

```

Запрос на указание платежной системы:

**Request:**

```

PATCH /v1/resources/orders/56a73b7ff06c5a21c393806b HTTP/1.1
Host: api.ticketscloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "payment": {
    "system": "api/BASE"
  }
}

```

**Response:**

```

HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a73b7ff06c5a21c393806b",
  "created_at": "2016-01-26T09:25:19.378000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "56a6253df06c5a059a93802e",
  "number": 497291903,
  "org": "56810047f06c5a6ac62f4e1d",
  "payment": {
    "failure_url": null,
    "success_url": null,
    "system": null
  },
  "promocodes": [],
  "reserved_till": "2016-01-26T09:35:19.392000+00:00",
  "rules": {
    "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
  },
  "status": "executed",
  "tickets": [
    "56a6253df06c5a059a9380a0"
  ],
}

```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "updated_at": "2016-01-26T09:25:19.393000+00:00",
    "value": "100.00",
    "value_extra": "0.00",
    "values": {
      "extra": "0.00",
      "full": "100.00",
      "nominal": "100.00"
    },
    "vendor": "56810047f06c5a6ac62f4e1d"
  }
}

```

Запрос на изменение статуса заказа в in\_progress:

**Request:**

```

PATCH /v1/resources/orders/56a73b7ff06c5a21c393806b HTTP/1.1
Host: api.ticketscloud.org
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "status": "in_progress"
}

```

**Response:**

```

HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "created_at": "2016-01-26T09:25:19.378000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "56a6253df06c5a059a93802e",
  "id": "56a73b7ff06c5a21c393806b",
  "number": 497291903,
  "org": "56810047f06c5a6ac62f4e1d",
  "payment": {
    "failure_url": null,
    "redirect_url": "https://www.goodservice.ru/payment_params.php?
    ↪customer=b6cbe4bd8a338159bdf0f4b059652f7023330898",
    "success_url": null,
    "system": "545b544a5d645a463e779d53"
  },
  "promocodes": [],
  "reserved_till": "2016-01-26T09:56:06.619000+00:00",
  "rules": {
    "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
  },
  "status": "in_progress",
  "tickets": [
    "56a6253df06c5a059a9380a0"
  ],
  "updated_at": "2016-01-26T09:26:06.620000+00:00",
}

```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "value": "100.00",
    "value_extra": "0.00",
    "values": {
      "extra": "0.00",
      "full": "100.00",
      "nominal": "100.00"
    },
    "vendor": "56810047f06c5a6ac62f4e1d"
  }
}

```

### 1.8.9 Ручное подтверждение заказа

Отправляем запрос на создание заказа:

**Request:**

```

POST /v1/resources/orders HTTP/1.1
Host: api.ticketscloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "event": "56a6253df06c5a059a93802e",
  "random": {
    "56a6254bf06c5a059b93800c": 1
  }
}

```

**Response:**

```

HTTP/1.1 201 Created
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a76c72f06c5a77b5938010",
  "created_at": "2016-01-26T12:54:10.744000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "56a6253df06c5a059a93802e",
  "number": 497291907,
  "org": "56810047f06c5a6ac62f4e1d",
  "payment": {
    "failure_url": null,
    "success_url": null,
    "system": null
  },
  "promocodes": [],
  "reserved_till": "2016-01-26T13:04:10.757000+00:00",
  "rules": {
    "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
  },
}

```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "status": "executed",
    "tickets": [
      "56a6253df06c5a059a9380a0"
    ],
    "updated_at": "2016-01-26T12:54:10.758000+00:00",
    "value": "100.00",
    "value_extra": "0.00",
    "values": {
      "extra": "0.00",
      "full": "100.00",
      "nominal": "100.00"
    },
    "vendor": "56810047f06c5a6ac62f4e1d"
  }
}

```

Устанавливаем статус in\_progress и платежную систему api/BASE:

#### Request:

```

PATCH /v1/resources/orders/56a76c72f06c5a77b5938010 HTTP/1.1
Host: api.ticketcloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "payment": {
    "system": "api/BASE"
  },
  "status": "in_progress"
}

```

#### Response:

```

HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a76c72f06c5a77b5938010",
  "created_at": "2016-01-26T12:54:10.744000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "56a6253df06c5a059a93802e",
  "number": 497291907,
  "org": "56810047f06c5a6ac62f4e1d",
  "payment": {
    "failure_url": null,
    "success_url": null,
    "system": "554b31d49cb538549d038eb4"
  },
  "promocodes": [],
  "reserved_till": "2016-01-27T00:57:27.789000+00:00",
  "rules": {
    "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
  }
}

```

(continues on next page)

(продолжение с предыдущей страницы)

```

    },
    "status": "in_progress",
    "tickets": [
        "56a6253df06c5a059a9380a0"
    ],
    "updated_at": "2016-01-26T12:57:27.789000+00:00",
    "value": "100.00",
    "value_extra": "0.00",
    "values": {
        "extra": "0.00",
        "full": "100.00",
        "nominal": "100.00"
    },
    "vendor": "56810047f06c5a6ac62f4e1d"
}

```

Делаем запрос на перевод заказа в статус done и тем самым подтверждаем его оплату:

**Request:**

```

PATCH /v1/resources/orders/56a76c72f06c5a77b5938010 HTTP/1.1
Host: api.ticketscloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
    "status": "done"
}

```

**Response:**

```

HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
    "id": "56a76c72f06c5a77b5938010",
    "created_at": "2016-01-26T12:54:10.744000+00:00",
    "customer": {
        "lang": "ru"
    },
    "deal": null,
    "event": "56a6253df06c5a059a93802e",
    "number": 497291907,
    "org": "56810047f06c5a6ac62f4e1d",
    "payment": {
        "failure_url": null,
        "success_url": null,
        "system": "554b31d49cb538549d038eb4"
    },
    "promocodes": [],
    "reserved_till": "2016-01-27T00:57:27.789000+00:00",
    "rules": {
        "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
    },
    "status": "done",
}

```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "tickets": [
      "56a6253df06c5a059a9380a0"
    ],
    "updated_at": "2016-01-26T12:57:27.789000+00:00",
    "value": "100.00",
    "value_extra": "0.00",
    "values": {
      "extra": "0.00",
      "full": "100.00",
      "nominal": "100.00"
    },
    "vendor": "56810047f06c5a6ac62f4e1d"
  }
}

```

### 1.8.10 Флаг all\_or\_nothing

В сети находятся 3 свободных билета.

Отправляем запрос на создание заказа с 2-мя билетами:

**Request:**

```

POST /v1/resources/orders HTTP/1.1
Host: api.ticketscloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "all_or_nothing": "true",
  "event": "56a6253df06c5a059a93802e",
  "random": {
    "56a6254bf06c5a059b93800c": 2
  }
}

```

**Response:**

```

HTTP/1.1 201 Created
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a774a8f06c5a28629386b8",
  "created_at": "2016-01-26T13:29:12.624000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "56a6253df06c5a059a93802e",
  "number": 497291912,
  "org": "56810047f06c5a6ac62f4e1d",
  "payment": {
    "failure_url": null,
    "success_url": null,
  }
}

```

(continues on next page)



(продолжение с предыдущей страницы)

```

    "system": null
  },
  "promocodes": [],
  "reserved_till": "2016-01-26T13:39:12.642000+00:00",
  "rules": {
    "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
  },
  "status": "executed",
  "tickets": [
    "56a6253df06c5a059a9380ae",
    "56a6253df06c5a059a9380b0"
  ],
  "updated_at": "2016-01-26T13:29:12.642000+00:00",
  "value": "200.00",
  "value_extra": "0.00",
  "values": {
    "extra": "0.00",
    "full": "200.00",
    "nominal": "200.00"
  },
  "vendor": "56810047f06c5a6ac62f4e1d"
}

```

Нам удалось забронировать два билета.

Отправляем еще один запрос на создание заказа с двумя билетами:

#### Request:

```

POST /v1/resources/orders HTTP/1.1
Host: api.ticketscloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "all_or_nothing": "true",
  "event": "56a6253df06c5a059a93802e",
  "random": {
    "56a6254bf06c5a059b93800c": 1
  }
}

```

#### Response:

```

HTTP/1.1 201 Created
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a7752af06c5a28629386ba",
  "created_at": "2016-01-26T13:31:22.838000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "56a6253df06c5a059a93802e",

```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "number": 497291913,
    "org": "56810047f06c5a6ac62f4e1d",
    "payment": {
      "failure_url": null,
      "success_url": null,
      "system": null
    },
    "promocodes": [],
    "reserved_till": "2016-01-26T13:41:22.844000+00:00",
    "rules": {
      "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
    },
    "status": "executed",
    "tickets": [],
    "updated_at": "2016-01-26T13:31:22.844000+00:00",
    "value": "0.00",
    "value_extra": "0.00",
    "values": {
      "extra": "0.00",
      "full": "0.00",
      "nominal": "0.00"
    },
    "vendor": "56810047f06c5a6ac62f4e1d"
  }
}

```

Второй запрос не вернул ни одного id билета в поле tickets так как остался всего один свободный билет в сети и в запросе присутствовал активный флаг `all_or_nothing`.

### 1.8.11 Отмена заказа

Отправляем запрос на создание заказа:

**Request:**

```

POST /v1/resources/orders HTTP/1.1
Host: api.ticketscloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "event": "56a6253df06c5a059a93802e",
  "random": {
    "56a6254bf06c5a059b93800c": 1
  }
}

```

**Response:**

```

HTTP/1.1 201 Created
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a7717ff06c5a28629386b3",

```

(continues on next page)

(продолжение с предыдущей страницы)

```

"created_at": "2016-01-26T13:15:43.289000+00:00",
"customer": {
  "lang": "ru"
},
"deal": null,
"event": "56a6253df06c5a059a93802e",
"number": 497291910,
"org": "56810047f06c5a6ac62f4e1d",
"payment": {
  "failure_url": null,
  "success_url": null,
  "system": null
},
"promocodes": [],
"reserved_till": "2016-01-26T13:25:43.302000+00:00",
"rules": {
  "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
},
"status": "executed",
"tickets": [
  "56a6253df06c5a059a9380ac"
],
"updated_at": "2016-01-26T13:15:43.302000+00:00",
"value": "100.00",
"value_extra": "0.00",
"values": {
  "extra": "0.00",
  "full": "100.00",
  "nominal": "100.00"
},
"vendor": "56810047f06c5a6ac62f4e1d"
}

```

Отправляем запрос на изменение статуса на canceled и тем самым отменяем заказ:

#### Request:

```

PATCH /v1/resources/orders/56a7717ff06c5a28629386b3 HTTP/1.1
Host: api.ticketscloud.org
Accept: application/json
Authorization: key 047bdb8bcee44d3693371920aaf9135c
Content-Type: application/json

{
  "status": "canceled"
}

```

#### Response:

```

HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
X-Partner: 56810047f06c5a6ac62f4e1d

{
  "id": "56a7717ff06c5a28629386b3",
  "created_at": "2016-01-26T13:15:43.289000+00:00",
  "customer": {

```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "lang": "ru"
  },
  "deal": null,
  "event": "56a6253df06c5a059a93802e",
  "number": 497291910,
  "org": "56810047f06c5a6ac62f4e1d",
  "payment": {
    "failure_url": null,
    "success_url": null,
    "system": null
  },
  "promocodes": [],
  "reserved_till": "2016-01-26T13:25:43.302000+00:00",
  "rules": {
    "56a6254bf06c5a059b93800c": "56a6254bf06c5a059b93800b"
  },
  "status": "canceled",
  "tickets": [
    "56a6253df06c5a059a9380ac"
  ],
  "updated_at": "2016-01-26T13:15:43.302000+00:00",
  "value": "100.00",
  "value_extra": "0.00",
  "values": {
    "extra": "0.00",
    "full": "100.00",
    "nominal": "100.00"
  },
  "vendor": "56810047f06c5a6ac62f4e1d"
}

```

### 1.8.12 Использование промокодов

Делаем запрос на создание заказа:

**Request:**

```

POST /v1/resources/orders HTTP/1.1
Accept: application/json
Accept-Encoding: gzip, deflate
Authorization: key 769313917cb94710860b507323134796
Connection: keep-alive
Content-Length: 80
Content-Type: application/json
Host: stage.ticketscloud.org
User-Agent: HTTPie/0.9.3

{
  "event": "570fb4f62d70951b3a2b0b5c",
  "random": {
    "570fb5082d70951b372b0b7b": 3
  }
}

```

**Response:**

```

HTTP/1.1 201 Created
Cache-Control: private, max-age=0, no-cache, no-store
Connection: keep-alive
Content-Length: 860
Content-Type: application/json; charset=UTF-8
Date: Thu, 14 Apr 2016 15:26:45 GMT
Server: nginx/1.6.2
X-Partner: 570faa682d70951b372b0b56

{
  "created_at": "2016-04-14T15:26:45.664000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "570fb4f62d70951b3a2b0b5c",
  "id": "570fb6b52d70951b392b0f35",
  "number": 220994,
  "org": "570faa682d70951b372b0b56",
  "payment": {
    "system": null
  },
  "promocodes": [],
  "reserved_till": "2016-04-14T15:36:45.688000+00:00",
  "rules": {
    "570fb5082d70951b372b0b7b": "570fb5082d70951b372b0b7a"
  },
  "salespoint": {
    "blanks": {
      "rejected": {},
      "used": {}
    }
  },
  "status": "executed",
  "tickets": [
    "570fb5082d709579c102b556",
    "570fb5082d709579c102b554",
    "570fb5082d709579c102b555"
  ],
  "updated_at": "2016-04-14T15:26:45.688000+00:00",
  "value": "300.00",
  "value_extra": "0.00",
  "values": {
    "discount": "0.00",
    "extra": "0.00",
    "full": "300.00",
    "nominal": "300.00",
    "sets": [
      {
        "d": "0.00",
        "id": "570fb5082d70951b372b0b7b",
        "p": "100.00",
        "ps": []
      }
    ]
  },
  "vendor": "570faa682d70951b372b0b56"
}

```

Применяем ранее созданный промокод на 30% «PROMO» к заказу

### Request

```
PATCH /v1/resources/orders/570fb6b52d70951b392b0f35 HTTP/1.1
Accept: application/json
Accept-Encoding: gzip, deflate
Authorization: key 769313917cb94710860b507323134796
Connection: keep-alive
Content-Length: 25
Content-Type: application/json
Host: stage.ticketscloud.org
User-Agent: HTTPie/0.9.3

{
  "promocodes": [
    "PROMO"
  ]
}
```

### Response

```
HTTP/1.1 200 OK
Cache-Control: private, max-age=0, no-cache, no-store
Connection: keep-alive
Content-Encoding: gzip
Content-Type: application/json; charset=UTF-8
Date: Thu, 14 Apr 2016 15:35:26 GMT
Server: nginx/1.6.2
Transfer-Encoding: chunked
X-Partner: 570faa682d70951b372b0b56

{
  "created_at": "2016-04-14T15:26:45.664000+00:00",
  "customer": {
    "lang": "ru"
  },
  "deal": null,
  "event": "570fb4f62d70951b3a2b0b5c",
  "id": "570fb6b52d70951b392b0f35",
  "number": 220994,
  "org": "570faa682d70951b372b0b56",
  "payment": {
    "system": null
  },
  "promocodes": [
    "570fb8952d70951b382b0b65"
  ],
  "reserved_till": "2016-04-14T15:36:45.688000+00:00",
  "rules": {
    "570fb5082d70951b372b0b7b": "570fb5082d70951b372b0b7a"
  },
  "salespoint": {
    "blanks": {
      "rejected": {},
      "used": {}
    }
  },
}
```

(continues on next page)

(продолжение с предыдущей страницы)

```
"status": "executed",
"tickets": [
  "570fb5082d709579c102b556",
  "570fb5082d709579c102b554",
  "570fb5082d709579c102b555"
],
"updated_at": "2016-04-14T15:35:26.046000+00:00",
"value": "210.00",
"value_extra": "0.00",
"values": {
  "discount": "90.00",
  "extra": "0.00",
  "full": "210.00",
  "nominal": "210.00",
  "sets": [
    {
      "d": "30.00",
      "id": "570fb5082d70951b372b0b7b",
      "p": "70.00",
      "ps": [
        "570fb8952d70951b382b0b65"
      ]
    }
  ]
},
"vendor": "570faa682d70951b372b0b56"
}
```

В результате успешного применения промокода цены в разделе «values» изменились и в массиве promocodes появился новый элемент.





**TicketsCloud** Платформа TicketsCloud.

**API-ключ** Ключ, выданный менеджером *TicketsCloud* для авторизации.

**Билет** Раздел «билеты».

**Билетная категория** TicketSet

**Заказ** Сущность, отображающая покупку пользователем *билетов* → *работа с заказом*

**Сделка** Совокупность условий по продаже *билетов* на *мероприятие*. См. раздел «сделки».

**Мероприятие** Раздел «мероприятия».

**Организатор** *Партнёр*, организатор *мероприятия*.

**Партнёр** Основное действующее лицо в системе.

**Платёжная система** Информация в *заказе* о способе оплаты → *платёжные системы*

**Распространитель** *Партнёр*, распространитель *билетов*.



## Символы

Билет, [45](#)  
Билетная категория, [45](#)  
Мероприятие, [45](#)  
Организатор, [45](#)  
Партнёр, [45](#)  
Платёжная система, [45](#)  
Распространитель, [45](#)  
Сделка, [45](#)  
Заказ, [45](#)

## А

API-ключ, [45](#)

## Р

RFC

RFC 2445, [8](#)  
RFC 2616, [4](#)

## Т

TicketsCloud, [45](#)