

FRIENDSHIP API

Version 1.0.0

Friendship is a platform designed to be a global marketplace of unique sport and gaming videos. The platform helps federations and digital content suppliers turn their videos into a successful business. Friendship allows them to sell the content and reach out to the buyers willing to find the content they need.

*Currently, Friendship data feed available with one method: through **REST API**.*


For the beginning, you need to sign up in friendship platform, create a project in SpringBME and become a partner. After your partner request approval, you will be able to book events from friendship platform and retrieve them from your website or via API methods.

Welcome to the FRIENDSHIP platform

get started by choosing one of the methods described below

Don't have a betting website?


Create your betting and gaming website with all the products and services that are in the SpringBME, including Friendship

 If you become a partner and create your website using SpringBME, you can book your events without worrying about integrations. You just book events, and they automatically appear in your website.

[Go to SpringBME to create your betting website →](#)

Already have an existing betting website

Request to become a partner to be able to book unlimited streams, scouting, odds feed and statistics data.

 You should become a partner to be able to book events and integrate with the Friendship API. After becoming an approved partner you will get all your booked data from our API.

[Go to SpringBME to become a Friendship partner →](#)

REST API DOCUMENTATION

Host	https://vstreamapi.feedconstruct.com
prefix	/api

Headers

You can find your credentials in the account settings section

x-partner-id	Find in account settings
x-api-key	Find in account settings

All requests start from the URL /API. So, for example, my games endpoint URL is **https://vstreamapi.feedconstruct.com/api/mygames**. All requests take **JSON as request** data and return **JSON as response** data.

All listed actions (unless the contrary is specified) can take the following parameters for **pagination** and **ordering**:

- *offset* (0 by default)
- *limit* (20 by default)
- *order_by* ("id" by default)
- *order_direction* ("asc" by default)
- *without_pagination=1* (0 by default)

More detailed information about data filtering is available [here](#).

Regions

This endpoint data can be necessary for **events Filtration by regions**.

1. Regions list retrieval

```
METHOD GET
URL /regions
URL PARAMS ids
           exclude_ids
```

2. Single Region retrieval

```
METHOD GET
URL /regions/{id}
URL PARAMS -
```

Countries

This endpoint data can be necessary for **events Booking by country**.

1. Countries list retrieval

```
METHOD GET
URL /countries
URL PARAMS ids
           q
           exclude_ids
```

2. Single Country retrieval

```
METHOD GET
URL /countries/{id}
URL PARAMS -
```

Sports

1. Sports list retrieval

```
METHOD GET
URL /sports
URL PARAMS ids
            exclude_ids
```

2. Single League retrieval

```
METHOD GET
URL /sports/{id}
URL PARAMS -
```

Leagues

1. Leagues list retrieval

```
METHOD GET
URL /leagues
URL PARAMS ids
            sport_ids
            exclude_ids
            exclude_sport_ids
```

2. Single League retrieval

```
METHOD GET
URL /leagues/{id}
URL PARAMS -
```

MyGames

This endpoint provides bought games list.

1. Games list retrieval

@important

There is also `type` URL param which sorts out all games, bought games, and finished games.

type=1 => All not started and started games

type=2 => All booked games which are finished or cancelled

type=3 => All booked games which are not finished

```
METHOD GET
URL /mygames /mygames/{id}
URL PARAMS type*
ids
sport_ids
league_ids
team_ids
region_ids
content_providers-product_ids
exclude_ids
exclude_sport_ids
exclude_league_ids
exclude_team_ids
exclude_region_ids
content_providers-exclude_product_ids
```

For more details [see response example](#).

2. Streaming view

Currently, available protocols are *mse*, *hls*, *rtmp*.

mse is the most stable, has the highest quality, lowest delay, is available on most platforms. [Learn more here](#)

The protocol can be indicated adding URL param in the API request, i.e. **protocol=hls**.

If the protocol is not indicated, then **mse** is used by default.

You can get either an *iframe* or the direct URL of the stream.

- For *Iframe*

```
METHOD GET
URL /streaming/{creation_id}
URL PARAMS signature *
```

- For stream URL

```
METHOD GET
URL /streaming/{creation_id}/outputs
URL PARAMS signature *
```

For more details [see response example](#).

@important

Notice that URL requires *creation_id*, make sure that you take *creation_id* from MyGames, not just *id*.

`signature` URL param is required. You can generate it using [this tutorial](#).

Any Endpoint Data Filters depending retrieved fields

Name	Type	Example
<i>ids</i> <i>exclude_ids</i>	<i>comma</i> <i>separated</i> <i>string</i>	<i>1,2</i>
<i>sport_ids</i> <i>exclude_sport_ids</i>	<i>comma</i> <i>separated</i> <i>string</i>	<i>1,2</i>
<i>league_ids</i> <i>exclude_league_ids</i>	<i>comma</i> <i>separated</i> <i>string</i>	<i>1,2</i>
<i>match_statuses</i> <i>exclude_match_statuses</i>	<i>comma</i> <i>separated</i> <i>string</i> <i>0 => Not</i> <i>Started., 1 =></i> <i>Started.,</i> <i>2 => Finished.,</i> <i>3 => Canceled.</i>	<i>0,1,2,3</i>
<i>team_ids</i> <i>exclude_team_ids</i>	<i>comma</i> <i>separated</i> <i>string</i>	<i>1,2</i>
<i>region_ids</i> <i>exclude_region_ids</i>	<i>comma</i> <i>separated</i> <i>string</i>	<i>1,2</i>
<i>content_providers-product_ids</i> <i>content_providers-exclude_product_ids</i>	<i>comma</i> <i>separated</i> <i>string</i>	<i>1,2</i>
<i>start</i>	<i>timestamp</i>	<i>1540828060</i>
<i>end</i>	<i>timestamp</i>	<i>1540828060</i>
<i>q</i> <i>Search for all</i> <i>keywords(team_1_name,</i> <i>team_2_name ...)</i>	<i>string</i>	
<i>limit</i>	<i>integ</i> <i>er</i>	<i>0</i>
<i>offset</i>	<i>integ</i> <i>er</i>	<i>20</i>
<i>without_pagination</i>	<i>boolean</i>	<i>true false</i>

Booking

The booking process is available both in web interface and in API.

This endpoint allows the Client user to buy events. Via this endpoint you will be able to book events which will appear in MyGames.

1. Booking Events Cart calculation

```
METHOD PUT
URL /book/events
PAYLOAD {
  "items": [{
    "event_id": 18923,
    "contents": [{
      "id": 3625,
      "product_id": 3,
      "countries": ["GER"]
    }]
  }]
}
```

2. Book Events

```
METHOD POST
URL /book/events
PAYLOAD {
  "items": [
    {
      "event_id": 18923,
      "contents": [{
        "id": 3625,
        "product_id": 3,
        "countries": ["GER"]
      }]
    }
  ]
}
```

Code Examples

Authorization signature generation

Example in PHP

```
<?php

$id = 2; // client's id in your website, integer value
$partnerId = 0; // x-partner-id
$api_key = '*****'; // x-api-key
$time = time();
$validminutes = 5;

$md5raw = md5($id . $api_key . $time . $validminutes . $partnerId);

$params = json_encode([
    'id' => $id,
    'time' => $time,
    'token' => base64_encode($md5raw),
    'validminutes' => $validminutes,
    'partner_id' => $partnerId
]);

$signature = base64_encode($params);

echo $signature;
```

Response Examples

1. My games retrieval data structure

Json response

```
{
  "status": "Success.",
  "message": "Successfully processed.",
  "data": [
    {
      "id": 1131254063,
      "creation_id": 15856995,
      "match_status": 0,
      "team_1_name": "Boston Celtics (Rose)",
      "team_2_name": "Philadelphia 76ers (Milla)",
      "sport_name": "E-Basketball",
      "league_name": "NBA Regular Cup (Season 2720)",
      "region_name": "North America",
      "team_1_id": 537343,
      "team_2_id": 531284,
      "sport_id": 73,
      "league_id": 18270806,
      "region_id": 5,
      "comment": null,
      "start_date": 1576788300,
      "approved": 1,
      "created_at": "2019-12-18 17:22:01",
      "updated_at": "2019-12-19 14:01:52",
      "product_ids": "[3]",
      "sport_alias": "EBasketball",
      "status_name": "Not Started.",
      "is_live": false,
      "content_providers": [
        {
          "id": 419315066,
          "event_id": 1131254063,
          "match_status": 0,
          "content_providing_user_id": 99,
          "default_price": 1,
          "product_id": 3,
          "organization_id": 1432,
          "upcoming_status": 1,
          "countries_price": {
            "ABW": "1",
            "AFG": "1",
            "AGO": "1",
            "AIA": "1"
          },
          "regions_price": {
            "002": "1",
            "009": "1",
            "019": "1",
            "142": "1"
          }
        }
      ]
    }
  ]
}
```

```

    "150": "1"
  },
  "blocked_countries": [],
  "product_content_id": null,
  "on_air_status": 0,
  "options": null,
  "product": {
    "id": 3,
    "name": "Video Streaming"
  },
  "booking": {
    "id": 1518420,
    "organization_id": 19070,
    "event_id": 1131254063,
    "product_event_map_id": 419315066,
    "product_id": 3,
    "autobooking_rule_id": 135,
    "is_blocked": 0,
    "bought_countries": [
      "ARM",
      "NLD",
      "RUS"
    ],
    "amount": 3,
    "autobooking_rule": {
      "id": 135,
      "name": "All matches"
    }
  }
},
]
},
],
"total_count": 1
}

```

2. Output stream URLs

Json response

```

{
  "status": "Success.",
  "message": "Successfully processed.",
  "data": {
    "hls-apple": {
      "url": "https://streaming.betconstruct.tv/",
      "stream_name": "15862007/index.m3u8?token=token"
    },
    "hls": {
      "url": "https://streaming.betconstruct.tv/",
      "stream_name": "15862007/video.m3u8?token=token"
    },
    "rtmp": {
      "url": "rtmp://streaming.betconstruct.tv:1935/static",

```

```
        "stream_name": "15862007?token=token"  
    },  
    "mse": {  
        "url": "wss://streaming.betconstruct.tv/clock/mse_id",  
        "stream_name": "15862007?token=token"  
    }  
}
```