

# FuturePulse API

## Contents

Introduction .....	6
Authentication .....	6
Login.....	6
Artist.....	7
Search for artists .....	7
Get artist information .....	7
Get tracks of an artist.....	8
Get albums of an artist .....	8
Get popularity of an artist.....	8
Get insights of an artist.....	8
Get streams of an artist .....	8
Get artist venue ranking .....	9
Get metrics per platform .....	9
Artists under monitoring (tracking) .....	10
Get venue popularities for artist.....	11
Get venue cities .....	11
Tracks .....	12
Get tracks .....	12
Get track by id.....	12
Get tracks by ISRCs.....	12
Get track by ISRC with audio features .....	13
Get track information by ISRC (outer id) .....	13
Get recognition of a track .....	13
Get track audio features .....	13
Get top tracks.....	14
Search tracks .....	17
Search tracks with audio genres predictions .....	17
Get popularity of a track.....	18
Get popularity sources.....	19
Predicted Tags based on an audio file .....	19

Gigs.....	20
Get gigs .....	20
Get gigs by artist .....	20
Get countries by type.....	20
Genres .....	22
Get genres.....	22
Get genre by id.....	22
Get genres ranking.....	23
Audio descriptors for a track .....	23
Playlists .....	24
Get Playlists.....	24
Get Playlist by id.....	24
Get Playlists of an artist .....	24
Get Playlists audio features .....	24
Get Playlists of a track endpoint .....	24
Get tracks of an artist appearing in playlists.....	25
Charts .....	26
Get charts.....	26
Get chart by id.....	26
Get countries of charts endpoints .....	26
Appearances of an artist in a chart endpoint .....	26
Events.....	27
Get events .....	27
Get event by id .....	27
Get events of an artist .....	27
Other .....	28
Lexicology.....	28
Dashboard.....	28
Annex .....	29
GET /artists.....	29
GET /artists/<:id> .....	30
GET /artists/<artist_id>/tracks .....	30
GET /artists_venue_ranking.....	31

GET /artists/{artist_id}/trackings .....	32
GET /artists/{artist_id}/streams.....	33
GET /artists/{artist_id}/albums.....	33
GET /artists/<artist_id>/insights.....	34
GET /artists/spotify_artist_id/playlists .....	36
GET /artists/spotify_artist_id/playlists/tracks.....	37
GET /tracks.....	38
GET /tracks/id .....	39
GET /tracks_by_isrc.....	39
GET /track/isrc/{isrc}.....	40
GET /tracks/{isrc}/recognition .....	42
GET /catalog/track_entries .....	47
POST /musimotion/search.....	49
POST /popularity/musimap/toptracks.....	53
POST /catalog/search.....	54
GET /popularity.....	57
GET /popularity/sources .....	58
GET /popularity/spotify/countries.....	60
GET /artists/{artist_id}/venue_popularities .....	60
GET /venue_cities .....	61
GET /gigs .....	62
GET /artists/{artist_id}/gigs .....	62
GET /musimotion .....	63
GET /genres.....	72
GET /genres/id/popularity .....	73
GET /genres_ranking.....	73
GET /mergedtrack/isrc/{isrc} .....	75
POST /musimotion/live.....	79
GET /playlists.....	79
GET /playlist/id.....	81
GET /tracks/{spotify_track_id}/playlists .....	85
GET /playlists/{playlist_id}/audio_features .....	87
GET /charts.....	97

GET /charts/id .....	98
GET /charts/countries .....	99
GET /charts/{chart_id}/artists/{spotify_artist_id} .....	99
GET /events .....	100
GET /events/id .....	101
GET /events/artists/{artist_id}/events .....	101
GET /sources/facebook_page/<fb_id>/facebook_mentions .....	102
GET /sources/facebook_page/<fb_id>/facebook_fan_count .....	103
GET /sources/spotify_artist/<spotify_artist_id>/spotify_artist_followers .....	103
GET /sources/spotify_artist/<spotify_artist_id>/spotify_artist_popularity .....	104
GET /sources/youtube_channel/<youtube_channel_id>/youtube_channel_subscribers .....	105
GET /sources/youtube_channel/<youtube_channel_id>/youtube_channel_views .....	106
GET /sources/soundcloud_artist/<soundcloud_artist_id>/soundcloud_artist_followers_count .....	107
GET /sources/lastfm_artist/<lastfm_user_name>/lastfm_artist_listener_count .....	107
GET /sources/lastfm_artist/<lastfm_user_name>/lastfm_artist_play_count .....	108
GET /sources/deezer_artist/<deezer_artist_id>/deezer_artist_fans .....	109
GET /spotify_artists/{spotify_artist_id}/streams .....	109
GET /lexicology .....	113
GET /dashboard .....	114

## Introduction

This document describes the web services that are available through the API Gateway. The API Gateway offers a uniform way to access the services provided by the project partners (ATC, Musimap, IRCAM, CERTH, BMAT). The API Gateway's URLs (starting with <https://api.futurepulse.eu/>) do not refer to ports that could impose access limitations on certain environments. The endpoints are presented by functionality following a simple artist search example workflow.

We should note that the services are both available from http (e.g. <http://api.futurepulse.eu>). This setting allows testing/ evaluation from an application that has no SSL configuration (https).

## Authentication

The API Gateway provides access to its services through JWT token authentication. The user has to register to the FuturePulse application (<https://app.futurepulse.eu/login>) with an email and password. The user credentials (email/password) will be used to access the API Gateway. Then he will use the token to call the API Gateway services.

## Login

A bearer token has to be generated to be used on the API Gateway calls. A post request is issued to <https://api.futurepulse.eu/api/login> with the following parameters:

Headers	<i>Content-Type: application/json</i>
Body	<pre>{   "username":   "theEmailRegisteredOnTheWebApp",   "password": "   thePasswordRegisteredOnTheWebApp" }</pre>

Upon success, we will get a response. We should copy the Authorization header part (excluding the "Bearer" part):

Bearer

```
eyJhbGciOiJSUzI1NiJ9.eyJzdWiiOiJnaW90ZXN0MUBhdGMuZ3liLCJpc3MiOiJGdXR1cmVQdWxzZSBBdXRoIFNlcnZlciIsImdyb3VwIjoiaUwVjB3JkEXhYmVslwieweqwoxNTcyMjY1fQ.JyOu4TDkwRvASTWj7IokbSFJnL1LsQVYB8TGiMcF4PSm5iV4YDAun0nRMV1DwfYZ51xSGviF_1fmGY_UgMi7QkVBhtQy5aJJHZmdEVZIAj2RukrqTZhPzBbWiBIIR4IzicA-pG93gjuSgkHPEtLIB9mnSfqKehl5M1k5LiWe2c1QGz0pCd9Hzioy0ja3xVJGC5r0PKi5spZ68mk7wGUGpYAg6DsBkk89ZICSQk4uPRY5Aw5K2qxmDtxeuwrtoqP0nsR078VQaRDCRIQwPNcn0gWBWjb2xYUUTTvwUVEPqpxqFycVks5xC9CeolazDujaa92PTC3G1h1KZAMhghTB7XM2FRQT-s3i_k0R_r4viQguAITO4ub8rWvS_PQXmmtkl3BrLfX0hVgzNdHI4wzJJqdjzoqNtnPtNO8HBRJ66bSZYTQjFUZBYT214E8F3NDmtBfe9ngNWjHxztryMAPbRpJzgpDwohhMHKGuE2sG_ibtYCP3_rymkrNZNnnnrC60xbR3-
```

***n0DCjyBtlZeY23muCf\_JVnVqHudmpILU003qFa6vzeM8QQ3IDQirPiivNzebB16Y4mOKu459L9EEIGfTRyokpM2nKiUQoiHHvrB5WZ2XxQPV5t89pk-UoM1jmg-4Vv6ZpQ2uDyA8lb8tv89oDofFdre-VRJCPXoVnejO4Y***

This token in bold (starting with *eyJhG...*) will be set on the Authorization header of the subsequent API Gateway calls.

## Artist

We present a set of services that relate to the artist entity. These include common functionalities such as search, information, related tracks, popularity, and insights. The actual output of the services is presented on the Annex.

### Search for artists

A common scenario would be to search for artists according to certain criteria and then request subsequent information (e.g. tracks, timelines) regarding this artist. For this purpose, the /artists search GET service is used:

[https://api.futurepulse.eu/artists?q=&sort=popularity&order=DESC&page=1&num\\_per\\_page=10](https://api.futurepulse.eu/artists?q=&sort=popularity&order=DESC&page=1&num_per_page=10)

Parameters:

Parameter	Description	Default	Example
page	Page number	1	
num_per_page	Number of artists per page	10	num_per_page=5
q	Query to search for artists in the collection	-	q=rasmus
country	Filter per country using <a href="#">ISO 3166-1 alpha-2 country codes</a>	-	country=SE
genre	Filter artists by genre (spotify genres)	-	genre=swedish%20pop
sort	Sort by the latest popularity, spotify_popularity, name	-	sort=spotify_popularity
order	Order of sorting	asc	asc or desc

### Get artist information

Using the artist id from the previous query, we can request for more artist information. The specific GET service has the form /artists/<:id> or <https://api.futurepulse.eu/artists/1313>. Information about the response data is provided on the Annex.

## Get tracks of an artist

We can fetch the tracks of an artist by calling the `/artists/<artist_id>/tracks` service:

[https://api.futurepulse.eu/artists/1313/tracks?num\\_per\\_page=10&page=1&sort=album\\_title&order=ASC](https://api.futurepulse.eu/artists/1313/tracks?num_per_page=10&page=1&sort=album_title&order=ASC)

You can rank artists tracks by using `sort/order` parameters. The options for `sort` are `title`, `album_title`, `label_name`, `duration` and `release_date` (default if no sort is used).

The `order` parameter can be `ASC` or `DESC` (default).

To filter by `isrc` use `isrc=....` e.g. `/artists/47/tracks?isrc=SEVJH0300206`.

To filter by release date, use the `released_since=....` e.g. `/artists/47/tracks?released_since=2018-01-01`

The above example gives tracks of artist 47 (the rasmus) released after 2018-01-01. We could have also a `release_until` parameter (can be implemented). To search by title, use the `q` parameter: e.g. `q=dark`. That returns tracks that have dark in their title.

In the same way, `album_query=....` searches in the albums by their title.

## Get albums of an artist

We can get the albums of a certain artist via the GET service `/artists/<:artist_id>/albums`:  
<https://api.futurepulse.eu/artists/47/albums>.

## Get popularity of an artist

We can get daily popularity of an artist, derived from an aggregation of the aforementioned social media and streaming metrics via the GET service `/artists/<:artist_id>/popularity`:

<https://api.futurepulse.eu/artists/47/popularity>.

## Get insights of an artist

The platform offers analytics derived from music streaming services (Deezer, Spotify, SoundCloud, Last.fm). These include information focused on the listeners and fans on these platforms. The artist insights service is `/artists/<artist_id>/insights`: <https://api.futurepulse.eu/artists/1313/insights>.

## Get streams of an artist

A certain artist can have streams in music streaming services (Deezer/Spotify/iTunes). The get streams of a certain artist the service is `/artists/{artist_id}/streams`: <https://api.futurepulse.eu/artists/47/streams>.

Parameters:



Parameter	Description	Default	Example
since	Date since (yyyy-mm-dd)	-	
until	Date until (yyyy-mm-dd)	-	
source	Streaming services: deezer or spotify or itunes	-	source=spotify

### Get artist venue ranking

The GET service returns artist venue ranking information with respect a certain city.

Example: [https://api.futurepulse.eu/artists\\_venue\\_ranking?city=Athens](https://api.futurepulse.eu/artists_venue_ranking?city=Athens)

For the list of supported cities check the /venue\_cities service.

### Get metrics per platform

To retrieve artist metrics per platform, use the following GET service:

*/sources/<:source\_type>/<:source\_id>/<:metric\_name>*.

Path Parameters:

Source type	Metric name	Description
facebook_page	facebook_mentions	Mentions of artist's page at a weekly basis
	facebook_fan_count	Number of likes the artist's page received
spotify_artist	spotify_artist_followers	Number of followers of the artist in spotify
	spotify_artist_popularity	Spotify popularity
youtube_channel	youtube_channel_subscribers	Number of subscribers on artist's YouTube channel
	youtube_channel_views	Total number of views in artist's YouTube channel
twitter_user	twitter_user_followers	Number of followers the artist has in Twitter
soundcloud_artist	soundcloud_artist_followers_count	Number of followers the artist has in Soundcloud

lastfm_artist	lastfm_artist_listener_count	Number of listeners the artist has in Last.fm
	lastfm_artist_play_count	Number of total plays the artist has in Last.fm

- **Facebook**

**GET** /sources/facebook\_page/<facebook\_page\_id>/facebook\_mentions

**GET** /sources/facebook\_page/<facebook\_page\_id>/facebook\_fan\_count

e.g. [https://api.futurepulse.eu/sources/facebook\\_page/72688240730/facebook\\_mentions](https://api.futurepulse.eu/sources/facebook_page/72688240730/facebook_mentions)

- **Spotify**

**GET** /sources/spotify\_artist/<spotify\_artist\_id>/spotify\_artist\_followers

**GET** /sources/spotify\_artist/<spotify\_artist\_id>/spotify\_artist\_popularity

e.g.

[https://api.futurepulse.eu/sources/spotify\\_artist/0muluUHym4N0wB62sD0MPo/spotify\\_artist\\_followers](https://api.futurepulse.eu/sources/spotify_artist/0muluUHym4N0wB62sD0MPo/spotify_artist_followers)

- **YouTube**

**GET** /sources/youtube\_channel/<youtube\_channel\_id>/youtube\_channel\_subscribers

**GET** /sources/youtube\_channel/<youtube\_channel\_id>/youtube\_channel\_views

e.g.

[https://api.futurepulse.eu/sources/youtube\\_channel/UCqCc1Zmtvj\\_11I2zGfgWkSw/youtube\\_channel\\_subscribers](https://api.futurepulse.eu/sources/youtube_channel/UCqCc1Zmtvj_11I2zGfgWkSw/youtube_channel_subscribers)

- **Twitter**

**GET** /sources/twitter\_user/<twitter\_user\_id>/twitter\_user\_followers

e.g. [https://api.futurepulse.eu/sources/twitter\\_user/27222127/twitter\\_user\\_followers](https://api.futurepulse.eu/sources/twitter_user/27222127/twitter_user_followers)

- **SoundCloud**

**GET** /sources/soundcloud\_artist/<soundcloud\_artist\_id>/soundcloud\_artist\_followers\_count

e.g. [https://api.futurepulse.eu/sources/soundcloud\\_artist/207992365/soundcloud\\_artist\\_followers\\_count](https://api.futurepulse.eu/sources/soundcloud_artist/207992365/soundcloud_artist_followers_count)

- **Last.FM**

**GET** /sources/lastfm\_artist/<lastfm\_user\_name>/lastfm\_artist\_listener\_count

**GET** /sources/lastfm\_artist/<lastfm\_user\_name>/lastfm\_artist\_play\_count

e.g. [https://api.futurepulse.eu/sources/lastfm\\_artist/Malena%20Ernman/lastfm\\_artist\\_listener\\_count](https://api.futurepulse.eu/sources/lastfm_artist/Malena%20Ernman/lastfm_artist_listener_count)

To obtain the ids of each platform (e.g. Facebook page id, YouTube channel id, etc.), first use /artist/<id> method. Then for the existing associated accounts call the corresponding API endpoint.

The data returned by the previous calls can be also restricted to a specific time using since / until query parameters e.g. since=2018-10-01&until=2018-10-18.

Artists under monitoring (tracking)

The GET service `artists/{artist_id}/trackings` checks whether an artist is being monitored and for which services.

Example: <https://api.futurepulse.eu/artists/47/trackings>

### Get venue popularities for artist

The service returns the venue popularities of an artist in respect to a set of tracked cities.

Example: [https://api.futurepulse.eu/artists/47/venue\\_popularities](https://api.futurepulse.eu/artists/47/venue_popularities)

### Get venue cities

The service returns the cities for which we track venue values.

Example: [https://api.futurepulse.eu/venue\\_cities](https://api.futurepulse.eu/venue_cities)

## Tracks

Tracks are related to an artist, but they can be fetched separately also (without an artist reference). A set of basic services is offered (add a track, get all tracks, get by id, get track recognition analysis). The actual output of the services is presented on the Annex.

### Get tracks

We must call the GET /tracks service to get a list of all the tracks and search for tracks by their name. An example is the following:

[https://api.futurepulse.eu/tracks?sort=spotify\\_popularity&order=DESC&page=1&num\\_per\\_page=10](https://api.futurepulse.eu/tracks?sort=spotify_popularity&order=DESC&page=1&num_per_page=10).

Parameters:

Parameter	Description	Default	Example
page	Page number	1	
num_per_page	Number of tracks per page	10	num_per_page=5
q	Query to search for tracks in the collection	-	q=letters
isrc	Search track by isrc	-	isrc=TCACQ1655666
sort	Sort by release_date, title, duration, recognition, spotify_popularity	-	sort=release_date
order	Order of sorting	asc	asc or desc

E.g.

[https://api.futurepulse.eu/tracks?sort=release\\_date&order=desc](https://api.futurepulse.eu/tracks?sort=release_date&order=desc)

<https://api.futurepulse.eu/tracks?q=dead%20letters>

<https://api.futurepulse.eu/tracks?isrc=TCACQ1655666>

### Get track by id

Use the GET /tracks/{futurepulse\_id} to get a certain track: <https://api.futurepulse.eu/tracks/10323>.

### Get tracks by ISRCs

This service accepts multiple ISRC codes to return the relevant tracks. It has the form GET /tracks\_by\_isrcs?isrc\_codes=<isrc1>|<isrc2>. Example:

[https://api.futurepulse.eu/tracks\\_by\\_isrcs?isrc\\_codes=GBAHS1600463|GBAHS1400099](https://api.futurepulse.eu/tracks_by_isrcs?isrc_codes=GBAHS1600463|GBAHS1400099).

## Get track by ISRC with audio features

Use the GET `/track/isrc/<isrc>` to get information about a certain track that includes audio features, such as genres, electronic genres, moods, bpm and others:

<https://api.futurepulse.eu/track/isrc/DEM389900024>.

Example response (partial):

```
"BNElectronicGenre":[
  {
    "confidence":"0.9372133165",
    "value":"Industrial music"
  },
  {
    "confidence":"0.0407237454",
    "value":"Electronic rock"
  },
  ...
]
```

## Get track information by ISRC (outer id)

Another service to get track information is the GET `/catalog/track_entries/<isrc>`:

[https://api.futurepulse.eu/catalog/track\\_entries/USQX91500448](https://api.futurepulse.eu/catalog/track_entries/USQX91500448).

## Get recognition of a track

In order to get the recognition level of a track (how well known a track is) use the service endpoint GET `/tracks/<isrc>/recognition`: <https://api.futurepulse.eu/tracks/USRC11701901/recognition>.

Parameters:

Parameter	Description	Default	Example
country	Country	- (global)	SE

Example: <https://api.futurepulse.eu/tracks/USRC11701901/recognition?country=SE>.

## Get track audio features

To get the track audio features, use the service GET `/tracks/{spotify_track_id}/audio_features`:

Example: [https://api.futurepulse.eu/tracks/70eFcWOvIMObDhURTqT4Fv/audio\\_features](https://api.futurepulse.eu/tracks/70eFcWOvIMObDhURTqT4Fv/audio_features).

Response:

```
{
  "_id": "70eFcw0v1MObDhURTqT4Fv",
  "acousticness": 0.124,
  "danceability": 0.64,
  "duration_ms": 197867,
  "energy": 0.648,
  "instrumentalness": 0,
  "key": 5,
  "liveness": 0.0802,
  "loudness": -8.113,
  "mode": 0,
  "speechiness": 0.187,
  "tempo": 92.977,
  "time_signature": 4,
  "valence": 0.548
}
```

## Get top tracks

This service returns the most popular tracks according to certain criteria.

[https://api.futurepulse.eu/popularity/musimap/toptracks?ntop=20&date=2020-10-03&days\\_diff=-30&relative=1&ids=FIHT71700003&ids\\_source=isrc](https://api.futurepulse.eu/popularity/musimap/toptracks?ntop=20&date=2020-10-03&days_diff=-30&relative=1&ids=FIHT71700003&ids_source=isrc)

More specific and returns:

a) The top popular tracks for a source and field on a particular date (using 'date' only).

b) The tracks with biggest increase for a source and field between two dates (using 'date' and 'date\_diff' or 'days\_diff').

- The result is a list of tracks and their popularity values in decreasing (increasing, if reversed=true) order.
- both a) and b) allow for the date(s) to be in the past, present or (up to 20 days in the) future (to retrieve top tracks (or difference) by predicted instead of actual popularity) (the response of a query by 'date' only will include field "predicted=" which is either '1' or 'null'; in a 'date\_diff' query this is not the case).
- both allow to specify reversed=true, which will return the least popular tracks (case a), or those with biggest decrease (case b), respectively.
- a query with date\_diff allows to specify relative=true, in this case the relative instead of absolute increase (decrease) is measured.

Example response (in "data" field of JSON response):

a) GET /popularity/musimap/toptracks?field=popularity&date=2020-01-19

The response contains the outer\_id of the tracks, their popularity values from the requested field on the requested date (in descending/ascending order) and whether the value was crawled ("predicted": true) otherwise "predicted" field will be missing

```
[
  {
    "popularity": 100,
    "outer_id": "fc79d859-bf8b-4ee6-a85a-8e592720af17",
    "isrc": "QZES71982312"
  },
  {
    "popularity": 98.7721234899951,
    "outer_id": "907d8647-e753-4f42-8eb6-f8e93f6a8e09",
    "predicted": true,
    "isrc": "USUM71913350"
  }, ...
]
```

b) GET /popularity/musimap/toptracks?date=2020-02-19&date\_diff=2020-02-12

The response contains for each track in the list: the two dates that have been compared (date1, date2), their popularity values from the requested field on those dates (popularity\_date1, popularity\_date2), the absolute difference of the popularity values between the two dates (popularity\_date2 - popularity\_date1) and the relative difference (in % increase of popularity\_date2 vs. popularity\_date1) and the track's outer\_id. (No information about predicted is given).

```
[
  {
    "date1": "2020-02-12",
    "date2": "2020-02-19",
    "popularity_date1": 51.69,
    "popularity_date2": 90.27,
    "difference": 38.57,
    "difference_relative": 74.62,
    "outer_id": "6ba1d681-512e-49e4-a9d3-792e6a27c1f6",
    "isrc": "USUM71913350"
  }, ...
]
```

Parameters:

Path parameters	Description	Default	Example
-----------------	-------------	---------	---------

source *	source for which to retrieve the top popular tracks (e.g. "musimap" or "spotify")	-	"musimap"
<b>Query parameters</b>	<b>Description</b>	<b>Default</b>	<b>Example</b>
ntop	number of top tracks to be returned	10	
field	field for which to check the popularity values in the specified source to determine the top tracks		"popularity"
date_diff	date against which to check an increase in the popularity in YYYY-MM-DD format; if omitted, only values on 'date' will be checked, no difference		"2019-01-01"
date *	date for which to check the top tracks in YYYY-MM-DD format (default is yesterday)		"2019-01-01"
days_diff	may be used instead of 'date_diff' to specify a relative number of days of offset from 'date' to define the date of comparison (e.g. -7 will define the date of comparison a week ago, 10 will define the comparison date 10 days in the future, i.e. positive or negative integers are allowed)		
relative	if set to true, determine top (bottom) tracks by relative increase (decrease) in % instead of absolute values (only possible in combination with 'date_diff' parameter)		false
reversed	if set to true, gets the least popular tracks (or biggest decrease) instead		false
country	"global" or ISO 3166-1 alpha-2 two letter code of a country (e.g. "us" or "gb")		"global"
ids	(comma-separated) list of ids (of a source specified with ids_source) or ISRCs (specified with ids_source=isrc) to restrict the top tracks search to in order to search the "top tracks"		"FIHT71700003"



	within a subset of tracks (if omitted, "top tracks" will be retrieved from all available tracks)		
ids_source *	(required with ids param) source of the ids provided		"isrc"

## Search tracks

Search tracks with parameters. Example POST <https://api.futurepulse.eu/catalog/search?q=Someplace Nice>

Parameters:

Query parameters	Description	Default	Example
q *	Word contained in the title of the track	-	"bad"
product		-	"musimotion:success"
fill_references			false
scroll	true or false		
scroll_next			
offset	Integer <= 10000. use with care, because of the scoring algorithm, two successive searches can return the results in a different order. If you use it for pagination, you may experience duplicate or missing results. Use scroll instead		0
limit	Integer <= 1000		10

## Search tracks with audio genres predictions

Search tracks with parameters and get audio genres, moods, rhythmic moods, voice family, bpm, key, mode, energy predictions in the response. Example POST: <https://api.futurepulse.eu/musimotion/search?q=love&limit=1>

Parameters:

Query parameters	Description	Default	Example
q *	Word contained in the title of the track	-	"bad"
fill_references			false
scroll	true or false		
scroll_next			
offset	Integer <= 10000. use with care, because of the scoring algorithm, two successive searches can return the results in a different order. If you use it for pagination, you may experience duplicate or missing results. Use scroll instead		0
limit	Integer <= 1000		10

## Get popularity of a track

This endpoint provides:

- popularity measures from different sources (e.g. Spotify, YouTube etc.)
- predictions of those popularity measures on each source 21 days into the future
- a global popularity score for both the past and future popularity of a track (between 0 and 100)

Example:

[https://api.futurepulse.eu/popularity/GBAHS1900714?source=isrc&output=deezer,youtube,bmat&date\\_from=2018-09-01](https://api.futurepulse.eu/popularity/GBAHS1900714?source=isrc&output=deezer,youtube,bmat&date_from=2018-09-01).

Parameters:

Path parameters	Description	Default	Example
id *	isrc code	-	GBAHS1900714
Query parameters	Description	Default	Example
source *	the source of the id given, can be any of the sources available via /popularity/sources endpoint or "isrc"	-	"deezer"
output *	comma separated list of output sources (for a list of possible sources, see /popularity/sources)	-	deezer,spotify,youtube,musimap

date_from *	start date of returned popularity measures	-	"2019-01-01"
-------------	--	---	--------------

### Get popularity sources

Get available popularity sources and their fields. The service returns information that is used in the popularity endpoint service (e.g.: "spotify", "youtube", "deezer", "musimap").

Example: <https://api.futurepulse.eu/popularity/sources>

### Predicted Tags based on an audio file

To get predicted tags such as genres, moods, rhythmic moods, acoustic attributes and voice family based on an audio file you have to use POST <https://api.musimap.io/musimotion/live>.

Parameters:

Parameter	Description	Default	Example
save_file	whether to save the file after the analysis	false	true/ false
<b>Request body schema:</b> multipart/form-data			
media_file*	string<binary>		

## Gigs

### Get gigs

The GET service returns the gigs of an artist according to certain criteria.

Example: [https://api.futurepulse.eu/gigs?page=1&per\\_page=5&artist\\_name=wehbba&venue=tokyo](https://api.futurepulse.eu/gigs?page=1&per_page=5&artist_name=wehbba&venue=tokyo)

Parameter	Description	Default	Example
page	Page number	1	
per_page	Number of gigs per page	20	per_page=5
artist_name	Search gigs by artist name	-	artist_name=wehbba
venue	Search gigs by venue (free text search in venue name and address)	-	venue=tokyo
country	Filter gigs by country		country=SE  One of: [ "AL", "AT", "BE", "BL", "CH", "CY", "CZ", "DE", "DK", "EE", "ES", "FI", "FR", "GR", "HR", "HU", "IE", "IT", "LT", "LU", "LV", "NL", "NO", "PL", "PT", "RO", "RS", "SE", "SI", "SK", "UK"]
since	Filter by date in format: YYYY-mm-dd	-	since=2020-01-01
until	Filter date in format: YYYY-mm-dd	-	since=2020-02-05

### Get gigs by artist

The GET service returns the gigs of an artist. The GET service has the form /artists/{artist\_id}/gigs

Example: <https://api.futurepulse.eu/artists/19646/gigs>

### Get countries by type

The service returns the countries that are supported in the gigs service.

Example: <https://api.futurepulse.eu/countries?type=gigs>

Output (parts of the json response have been removed for presentation purposes):

```
{  
  "countries": [  
    "MN",  
    "AX",  
    "GY",  
    "TG",  
    "MW",  
    "SK",  
    "IN",  
    "NP",  
    ...  
  ]  
}
```

## Genres

Genres in the context of FuturePulse are the different types of music.

### Get genres

To get a list of the available genres, we call GET /genres: <https://api.futurepulse.eu/genres>.

Parameters:

Parameter	Description	Default	Example
page	Page number	1	
num_per_page	Number of tracks per page	10	num_per_page=5
source	Currently there are three sources: pgm_syb, bass_nation and spotify. Bass nation genres focus on electronic music, pgm_syb are a small list of 60 genres used by FuturePulse partners – Playground (PGM) and Soundtrack Your Brand (SYB) – and spotify indicates the genres used by Spotify platform.	-	source=spotify

### Get genre by id

To get a specific genre we have to call GET /genres/<:genre\_id>:  
<https://api.futurepulse.eu/genres/2853/popularity>.

Parameters:

Parameter	Description	Default	Example
page	Page number	1	
num_per_page	Number of playlists per page	20	num_per_page=5
platform	One of itunes, spotify, dezer. It's a required parameter.	-	platform=spotify
type	Type is determined by the release date of the tracks in the playlist. This parameter can get one of the following values: catalogue, mixed or frontline  If no type is specified, then playlists of any type are returned.	-	period=frontline
q	Search playlists by title	-	q=hits

date_to	end date of returned popularity measures (must not be earlier than date_from; may be 21 days in the future; if omitted, will return predicted popularity measures for 21 days into the future)	-	"2019-05-08"
country	"global" or ISO 3166-1 alpha-2 two letter code of a country (e.g. "us" or "gb")	"global"	"gb"

## Get genres ranking

To get the genres ranking per country and other criteria use the GET */genres\_ranking* service:  
[https://api.futurepulse.eu/genres\\_ranking?country=global&source=spotify&date=2020-02-09](https://api.futurepulse.eu/genres_ranking?country=global&source=spotify&date=2020-02-09).

Parameters:

Parameter	Description	Default	Example
country	global, ES, SE, GR, etc.	-	GR
date	yyyy-mm-dd (monthly based)	-	2020-05-17
source	future_pulse or spotify	-	spotify
limit	limit	100	100
rank_by	Rank by popularity or growth	popularity	growth

## Audio descriptors for a track

This service provides a list of audio descriptors (genres) for a certain track.

Example GET <https://api.futurepulse.eu/mergedtrack/isrc/USQX91500448>.

## Playlists

Playlists are lists of songs that can be played either sequentially or in a shuffled order. The FuturePulse offers analytics for playlists in terms of social platform rating and popularity.

### Get Playlists

To get the list of playlists we have to call GET /playlists with some parameters:

[https://api.futurepulse.eu/playlists?sort\\_by=avg\\_popularity&order=DESC&platform=spotify&num\\_per\\_page=10&page=1&type=&genre=&q=&curat](https://api.futurepulse.eu/playlists?sort_by=avg_popularity&order=DESC&platform=spotify&num_per_page=10&page=1&type=&genre=&q=&curat)

### Get Playlist by id

To get a certain playlist we call GET /playlist/<:playlistId>:

<https://api.futurepulse.eu/playlists/37i9dQZF1DXcBWIGoYBM5M>

### Get Playlists of an artist

To get playlists of an artist i.e. playlists in which the specified artist appears we call GET /artists/<:spotify\_artist\_id>/playlists. For example:

[https://api.futurepulse.eu/artists/76ptJV8617638xrpeoUtzl/playlists?sort\\_by=avg\\_popularity&order=DESC&platform=spotify&num\\_per\\_page=10&page=1&type=&genre=&q=&curat](https://api.futurepulse.eu/artists/76ptJV8617638xrpeoUtzl/playlists?sort_by=avg_popularity&order=DESC&platform=spotify&num_per_page=10&page=1&type=&genre=&q=&curat)

### Get Playlists audio features

To get playlists audio features we call GET /playlists/{playlist\_id}/audio\_features. For example:

[https://api.futurepulse.eu/playlists/37i9dQZF1DXbTxeAdrVG2l/audio\\_features](https://api.futurepulse.eu/playlists/37i9dQZF1DXbTxeAdrVG2l/audio_features)

### Get Playlists of a track endpoint

To get playlists of a track we call GET /tracks/{spotify\_track\_id}/playlists. For example:

[https://api.futurepulse.eu/tracks/0z4kyKuFGmUV7WQ7gB1sAp/playlists?platform=spotify&num\\_per\\_page=10&page=1](https://api.futurepulse.eu/tracks/0z4kyKuFGmUV7WQ7gB1sAp/playlists?platform=spotify&num_per_page=10&page=1)



## Get tracks of an artist appearing in playlists

To get the tracks of an artist, appearing in playlists under monitoring we call GET /artists/<:spotify\_artist\_id>/playlists/tracks. For example:

[https://api.futurepulse.eu/artists/76ptJV8617638xrpeoUtzl/playlists?sort\\_by=avg\\_popularity&order=DESC&platform=spotify&num\\_per\\_page=10&page=1&type=&genre=&q=&curat/tracks](https://api.futurepulse.eu/artists/76ptJV8617638xrpeoUtzl/playlists?sort_by=avg_popularity&order=DESC&platform=spotify&num_per_page=10&page=1&type=&genre=&q=&curat/tracks).

## Charts

The music charts are a method of ranking music judging by the popularity during a given period of time. FuturePulse API offers charts related services that refer to Official Charts ([www.officialcharts.com](http://www.officialcharts.com)) categories.

### Get charts

To get the list of charts we have to call GET /charts: <https://api.futurepulse.eu/charts>.

### Get chart by id

To get the details for a specific chart we have to call GET /charts/<:chart\_id>:  
<https://api.futurepulse.eu/charts/5acf4638b40ae300075c0ab3>.

### Get countries of charts endpoints

We can also get a list of the countries of the charts endpoints: GET /charts/countries:  
<https://api.futurepulse.eu/charts/countries>.

### Appearances of an artist in a chart endpoint

The appearances of an artist in a chart endpoint can be fetched through the service: GET /charts/{chart\_id}/artists/{spotify\_artist\_id}:  
<https://api.futurepulse.eu/charts/5acf4638b40ae300075c0ab3/artists/76ptJV8617638xrpeoUtzl>.

## Events

### Get events

Get all the events by criteria from the GET service */events*.

Example: [https://api.futurepulse.eu/events?page=1&num\\_per\\_page=10](https://api.futurepulse.eu/events?page=1&num_per_page=10)

Parameter	Description	Default	Example
page	results page		1
num_per_page	numbers per page		10

### Get event by id

Get a specific event from the GET service */event/{event\_id}*.

Example: <https://api.futurepulse.eu/event/10>

### Get events of an artist

Get the events of an artist from the GET service */artists/{artist\_id}/events*.

Example: <https://api.futurepulse.eu/artists/47/events>.

## Other

In this section we present a set of services that are not strictly correlated to one of the already mentioned entities.

### Lexicology

The service returns a list of descriptions that related to categories like genre, situation, rhythmic\_mood. It accepts a query by name (q) and a category parameter:

[https://api.futurepulse.eu?q=Happiness&category\\_name=mood](https://api.futurepulse.eu?q=Happiness&category_name=mood)

Example response:

```
{
  "timestamp":1604121727588,
  "data":[
    {
      "id":"f05f72ad-e056-cea2-3bd2-e109efa0c02b",
      "category_name":"mood",
      "name":"Happiness",
      "description":"Meaning all moods related to : POSITIVITY / FREEDOM
\nMoments : Festivity / Easy living
\nMoods providing : Unworried feelings / Innocent amusement
\nMusical example : Teen pop"
    }
  ],
  "offset":0,
  "limit":-1,
  "size":1
}
```

### Dashboard

The service returns aggregated information regarding certain types shown on FuturePulse dashboard. The GET service has the form: /dashboard?type=<type>. Example:

<https://api.futurepulse.eu/dashboard?type=genres>

The supported type parameters are genres, artists and genres\_ranking.

## Annex

### GET /artists

Description: Get a list of platform artists.

Response (entries have been removed for presentation purposes):

```
{
  "artists": [
    {
      "beatport_charts": null,
      "beatport_id": null,
      "country": "US",
      "deezer_fans": 1032211,
      "deezer_id": "482",
      "deezer_link": "https://www.deezer.com/artist/482",
      "followed": false,
      "genres": [
        {
          "id": 14,
          "name": "rock",
          "source": "spotify"
        }
      ],
      "id": 1313,
      "image": "https://i.scdn.co/image/42436559ae029809674c8498f2c998e138a7a757",
      "knm_id": "496222",
      "lastfm_id": "Marilyn Manson",
      "lastfm_link": "https://www.last.fm/music/Marilyn%20Manson",
      "name": "Marilyn Manson",
      "playlist_tracks": 363,
      "playlists": 61,
      "popularity": 91,
      "popularity_growth": 4.0,
      "residentadvisor_link": null,
      "sml2f_ratio": 1.873,
      "soundcloud_id": "15801845",
      "soundcloud_link": "http://soundcloud.com/marilynmanson",
      "spotify_analytics": 0,
      "spotify_charts": null,
      "spotify_followers": 2290970,
      "spotify_id": "2VYQTNDsvvKN9wmU5w7xpj",
      "spotify_link": "https://open.spotify.com/artist/2VYQTNDsvvKN9wmU5w7xpj",
      "spotify_monthly_listeners": 3979036,
      "spotify_popularity": 75,
      "trend": null
    }
  ],
  "num_per_page": 10,
  "page": 1,
  "total": 167124
}
```

GET /artists/<:id>

Description: Get information about a specific artist.

E.g. <https://api.futurepulse.eu/artists/47>

Response:

```
{
  "id": 47,
  "apple_music_id": "6929309",
  "band": 1,
  "country": "FI",
  "deezer_id": "84",
  "deezer_link": "https://www.deezer.com/artist/84",
  "discogs_id": "228700",
  "discogs_link": "https://www.discogs.com/artist/228700",
  "fb_id": "165262930168358",
  "fb_link": "https://www.facebook.com/therasmusofficial/",
  "fb_name": "The Rasmus",
  "genres": [
    "funk metal"
  ],
  "image": "https://i.scdn.co/image/6bb2d8d8fa64f811a5fc45d8d0a1eec000a11579",
  "knm_id": "11",
  "lastfm_id": "The Rasmus",
  "lastfm_link": "https://www.last.fm/music/The Rasmus",
  "musicbrainz_id": "012e3432-71d3-4317-9ce5-b60cb6cdc38f",
  "musicbrainz_link": "https://musicbrainz.org/artist/012e3432-71d3-4317-9ce5-b60cb6cdc38f",
  "name": "The Rasmus",
  "soundcloud_id": "11504903",
  "soundcloud_link": "http://soundcloud.com/the-rasmus-official",
  "spotify_id": "76ptJV8617638xrpeoUtzL",
  "spotify_link": "https://open.spotify.com/artist/76ptJV8617638xrpeoUtzL",
  "twitter_id": "113809631",
  "twitter_link": "https://twitter.com/rasmusofficial",
  "wikipedia_link": "https://en.wikipedia.org/wiki/The_Rasmus",
  "youtube_id": "UCDhEg3ByjFIju7bD4jhrctw",
  "youtube_link": "https://www.youtube.com/channel/UCDhEg3ByjFIju7bD4jhrctw",
  "youtube_title": "The Rasmus (Official)"
}
```

GET /artists/<artist\_id>/tracks

Description: Get tracks of a specific artist.

E.g.

[https://api.futurepulse.eu/artists/1313/tracks?num\\_per\\_page=10&page=1&sort=album\\_title&order=ASC](https://api.futurepulse.eu/artists/1313/tracks?num_per_page=10&page=1&sort=album_title&order=ASC)

Response (other tracks have been removed for presentation purposes):

```
{
  "artist": "1313",
  "num_per_page": 10,
  "page": 1,
}
```

```

"total_tracks": 1041,
"tracks": [
  {
    "album_id": 164987,
    "album_title": "24 Hours To Live (Original Motion Picture Soundtrack)",
    "apple_music_id": null,
    "cover": "https://i.scdn.co/image/a9b12debbc58be250639a5b813604db0b04ba8c4",
    "deezer_id": null,
    "duration": 162,
    "id": 242740,
    "isrc": "GB6LT1700774",
    "label_id": 11743,
    "label_name": "Var\u00e8se Sarabande",
    "release_date": "2017-12-29",
    "soundcloud_id": null,
    "spotify_id": "0TL5c9seKjFYSUgbgpMbhi",
    "spotify_popularity": null,
    "title": "God's Gonna Cut You Down"
  }
]
}

```

## GET /artists\_venue\_ranking

Description: Get artist venue ranking information with respect to a certain city.

E.g. [https://api.futurepulse.eu/artists\\_venue\\_ranking?city=Athens](https://api.futurepulse.eu/artists_venue_ranking?city=Athens).

Response (partial for presentation purposes):

```

{
  "artists": [
    {
      "artist_id": "23304",
      "bandsintown_url":
"https://www.bandsintown.com/a/23304?came_from=267&app_id=65cb50b645c2ea2e99e7cf6a36b3287b",
      "city": "Athens",
      "country": "GR",
      "facebook_page": "http://www.facebook.com/40307134837",
      "futurepulse_id": 604156,
      "genres": [
        {
          "id": 415,
          "name": "folk"
        },
        {
          "id": 525,
          "name": "alternative country"
        },
        {
          "id": 559,
          "name": "texas country"
        }
      ]
    }
  ]
}

```

```

    {
      "id": 1243,
      "name": "country dawn"
    },
    {
      "id": 3756,
      "name": "oklahoma country"
    }
  ],
  "id": 604156,
  "image_url": "https://photos.bandsintown.com/large/8595237.jpeg",
  "mbid": "1d10aad6-714a-40a5-8f6e-14be37ac3498",
  "name": "Kelly Willis",
  "score": 0.03505778743102012,
  "since": "2019-07-22",
  "spotify_id": "2WG7Z6CpNehgR1WGgjKF0E",
  "tracker_count": 15227,
  "until": "2020-07-20"
}
, ...
}

```

## GET /artists/{artist\_id}/trackings

Description: Check whether an artist is being monitored and for which services.

E.g. <https://api.futurepulse.eu/artists/47/trackings>.

Response (partial for presentation purposes):

```

{
  "trackings": [
    {
      "id": "76ptJV8617638xrpeoUtz1",
      "job": "track_spotify_artist",
      "mappings": [
      ],
      "next_run": "2020-05-09 06:49:00+00:00",
      "source": "spotify_artist",
      "status": "running"
    },
    {
      "id": "84",
      "job": "track_deezer_artist",
      "mappings": [
      ],
      "next_run": "2020-05-08 22:17:00+00:00",
      "source": "deezer_artist",
      "status": "running"
    },
    {
      "id": "11504903",

```



```

    "job": "track_soundcloud_artist",
    "mappings": [
    ],
    "next_run": "2020-05-08 08:52:00+00:00",
    "source": "soundcloud_artist",
    "status": "running"
  },
  ...]
}

```

## GET /artists/{artist\_id}/streams

Description: Get an analysis about a Spotify artist.

E.g. <https://api.futurepulse.eu/artists/47/streams?source=spotify>.

Response (full):

```

{
  "artist_knm_id": "47",
  "store": "spotify",
  "streams": []
}

```

## GET /artists/{artist\_id}/albums

Description: Get a list of the albums of an artist.

E.g. <https://api.futurepulse.eu/artists/47/albums>.

Response (parts of the json response have been removed for presentation purposes):

```

{
  "albums": [
    {
      "id": 531,
      "label_id": 378,
      "label_name": "Playground Music",
      "release_date": "2018-10-12",
      "spotify_id": "0oH4To3wyqoV04QmjuyQTq",
      "title": "Dark Matters (Bonus Track Edition)",
      "tracks": 13,
      "type": "album",
      "upc": "7332181084276"
    },
    {
      "id": 532,
      "label_id": 378,
      "label_name": "Playground Music",
      "release_date": "2017-10-06",
      "spotify_id": "4Va4qc4gqnwJnGJaBkSook",

```

```

    "title": "Dark Matters",
    "tracks": 10,
    "type": "album",
    "upc": "7332181080643"
  }
],
"artist": "47"
}

```

GET /artists/<artist\_id>/insights

Description: Get insights of a specific artist.

E.g. <https://api.futurepulse.eu/artists/1313/insights>

Response (parts of the json response have been removed for presentation purposes):

```

deezer_artist_fans{
  "growth": {
    "deezer_artist_fans": {
      "current_value": 1032211.0,
      "growth_30": 8352.5,
      "growth_60": 17633.5,
      "percentage_growth_30": 0.8157865564431023,
      "percentage_growth_60": 1.7380140994650481
    },
    "lastfm_artist_listener_count": {
      "current_value": 2267469.0,
      "growth_30": 7132.0,
      "growth_60": 12002.0,
      "percentage_growth_30": 0.31552817124172194,
      "percentage_growth_60": 0.5321292663559254
    },
    "lastfm_artist_play_count": {
      "current_value": 124397900.0,
      "growth_30": 274038.0,
      "growth_60": 626587.5,
      "percentage_growth_30": 0.22077785494621494,
      "percentage_growth_60": 0.5062461464969922
    },
    "soundcloud_artist_followers_count": {
      "current_value": 54827.0,
      "growth_30": 736.5,
      "growth_60": 1465.5,
      "percentage_growth_30": 1.3616069365230494,
      "percentage_growth_60": 2.7463620775278055
    },
    "spotify_artist_followers": {
      "current_value": 2290970.0,
      "growth_30": 45214.5,
      "growth_60": 91516.0,
      "percentage_growth_30": 2.0133313711131957,
      "percentage_growth_60": 4.160850829342191
    }
  },
}

```

```

"spotify_artist_popularity": {
  "current_value": 75.0,
  "growth_30": 2.0,
  "growth_60": 2.0,
  "percentage_growth_30": 2.73972602739726,
  "percentage_growth_60": 2.73972602739726
}
},
"latest_metrics": [
  {
    "change": 0,
    "metric": "spotify_popularity",
    "value": 75
  },
  {
    "change": 1599,
    "metric": "spotify_followers",
    "value": 2290970
  },
  {
    "change": 383,
    "metric": "deezer_fans",
    "value": 1032211
  },
  {
    "change": 20,
    "metric": "soundcloud_followers",
    "value": 54827
  },
  {
    "change": 15697,
    "metric": "lastfm_plays",
    "value": 124397900
  },
  {
    "change": 245,
    "metric": "lastfm_listeners",
    "value": 2267469
  }
],
"spotify_insights": {
  "cities": [
    {
      "city": "Mexico City",
      "country": "MX",
      "listeners": 159991,
      "region": "CMX"
    }
  ],
  "followers": 2124513,
  "global_chart_position": 0,
  "monthly_listeners": 3979036,
  "monthly_listeners_to_followers_ratio": 1.8729167578640376,
  "playlist_listeners": 465875,
  "playlists": [
    {
      "id": "37i9dQZF1DX8vv4uePtB1w",

```

```

    "listeners": 91399,
    "owner": "spotify"
  },
  {
    "id": "37i9dQZF1DX3oM43CtKnRV",
    "listeners": 68317,
    "owner": "spotify"
  }
],
"top_city_listeners": 1526648
}
}

```

GET /artists/spotify\_artist\_id/playlists

Description: Get a playlist of an artist i.e. playlists in which the specified artist appears.

E.g.

[https://api.futurepulse.eu/artists/76ptJV8617638xrpeoUtzl/playlists?sort\\_by=avg\\_popularity&order=DESC&platform=spotify&num\\_per\\_page=10&page=1&type=&genre=&q=&curat.](https://api.futurepulse.eu/artists/76ptJV8617638xrpeoUtzl/playlists?sort_by=avg_popularity&order=DESC&platform=spotify&num_per_page=10&page=1&type=&genre=&q=&curat.)

Response (parts of the json response have been removed for presentation purposes):

```

{
  "artist_id": "76ptJV8617638xrpeoUtzl",
  "playlists": [
    {
      "add_rate": 0.010869565217391304,
      "avg_popularity": 34.15217391304348,
      "cover_url": "https://pl.scdn.co/images/pl/default/a1063a2c9fc00e057f72eeb436f07e95704efc96",
      "curator": "spotify",
      "curator_name": "Spotify",
      "curator_url": "https://open.spotify.com/user/spotify",
      "delete_rate": 0.021505376344086023,
      "description": "Rock-lauluja tuhansien järvien rannoilta, Tehosekoittimesta Maija Vilkkumaahan.",
      "followers": 20050,
      "followers_change": 473,
      "id": "37i9dQZF1DWTREX34Zufzd",
      "name": "Suomirockin klassikot",
      "num_of_tracks": 92,
      "platform": "spotify",
      "popularity_std": 15.873778475844652,
      "tracked_at": "2019-11-12 14:03:00",
      "updated_at": "2019-11-04 11:06:56",
      "url": "https://open.spotify.com/playlist/37i9dQZF1DWTREX34Zufzd"
    },
    {
      "add_rate": 0.10752688172043011,
      "avg_popularity": 52.32258064516129,
      "cover_url": "https://pl.scdn.co/images/pl/default/470f4669988cf29a5b3cd5c83ae7ea426dc0130a",

```

```

    "curator": "spotify",
    "curator_name": "Spotify",
    "curator_url": "https://open.spotify.com/user/spotify",
    "delete_rate": 0.4391891891891892,
    "description": "1990-luvun teinit tietää, jonnet ei muista. Kannessa Tiktak.",
    "followers": 58698,
    "followers_change": 396,
    "id": "37i9dQZF1DWXVBw4Z7cIDC",
    "name": "Jonnet ei muista",
    "num_of_tracks": 93,
    "platform": "spotify",
    "popularity_std": 17.837474922752953,
    "tracked_at": "2019-11-13 11:56:00",
    "updated_at": "2019-09-23 17:34:26",
    "url": "https://open.spotify.com/playlist/37i9dQZF1DWXVBw4Z7cIDC"
  }
],
"total_playlists": 113,
"total_tracks": 185
}

```

GET /artists/spotify\_artist\_id/playlists/tracks

Description: Get the tracks of an artist, appearing in playlists under monitoring, GET /artists/<:spotify\_artist\_id>/playlists/tracks.

E.g.

[https://api.futurepulse.eu/artists/76ptJV8617638xrpeoUtzl/playlists?sort\\_by=avg\\_popularity&order=DESC&platform=spotify&num\\_per\\_page=10&page=1&type=&genre=&q=&curat/tracks](https://api.futurepulse.eu/artists/76ptJV8617638xrpeoUtzl/playlists?sort_by=avg_popularity&order=DESC&platform=spotify&num_per_page=10&page=1&type=&genre=&q=&curat/tracks).

Response (parts of the json response have been removed for presentation purposes):

```

{
  "artist_id": "76ptJV8617638xrpeoUtzl",
  "playlists": [
    {
      "add_rate": 0.010869565217391304,
      "avg_popularity": 34.15217391304348,
      "cover_url": "https://pl.scdn.co/images/pl/default/a1063a2c9fc00e057f72eeb436f07e95704efc96",
      "curator": "spotify",
      "curator_name": "Spotify",
      "curator_url": "https://open.spotify.com/user/spotify",
      "delete_rate": 0.021505376344086023,
      "description": "Rock-lauluja tuhansien järvien rannoilta, Tehosekoittimesta Maija Vilkkumaahan.",
      "followers": 20050,
      "followers_change": 473,
      "id": "37i9dQZF1DWTREX34Zufzd",
      "name": "Suomirockin klassikot",
      "num_of_tracks": 92,
      "platform": "spotify",

```

```

        "popularity_std": 15.873778475844652,
        "tracked_at": "2019-11-12 14:03:00",
        "updated_at": "2019-11-04 11:06:56",
        "url": "https://open.spotify.com/playlist/37i9dQZF1DWTREX34Zufzd"
    },
    {
        "add_rate": 0.0,
        "avg_popularity": 45.27777777777778,
        "cover_url": "https://mosaic.scdn.co/640/ab67616d0000b27325a1033c5bebe9493f56c092ab67616d0000b27369ba1d07a64415bfdfe60dddab67616d0000b273c407644fea15b74c3eb8b9b4ab67616d0000b273fb28f40dec4551052d109cba",
        "curator": "verne_elpais",
        "curator_name": "Verne",
        "curator_url": "https://open.spotify.com/user/verne_elpais",
        "delete_rate": 0.0,
        "description": "",
        "followers": 3649,
        "followers_change": -9,
        "id": "6bdf4k4KArgkpjFUjTuYwc",
        "name": "Hace 15 años estabas escuchando estos temas",
        "num_of_tracks": 18,
        "platform": "spotify",
        "popularity_std": 24.05273270581295,
        "tracked_at": "2019-11-12 15:21:00",
        "updated_at": "2017-12-29 17:27:43",
        "url": "https://open.spotify.com/playlist/6bdf4k4KArgkpjFUjTuYwc"
    }
],
"total_playlists": 113,
"total_tracks": 185
}

```

## GET /tracks

Description: Get a list of tracks.

E.g.

[https://api.futurepulse.eu/tracks?sort=spotify\\_popularity&order=DESC&page=1&num\\_per\\_page=10](https://api.futurepulse.eu/tracks?sort=spotify_popularity&order=DESC&page=1&num_per_page=10)

Response (parts of the json response have been removed for presentation purposes):

```

{
  "num_per_page": 10,
  "page": 1,
  "total": 233189,
  "tracks": [
    {
      "album_id": 148708,
      "album_title": "(What's The Story) Morning Glory? (Deluxe Edition) [Remastered]",
      "artist_id": 149,
      "artist_name": "Oasis",
      "duration": 258,
      "id": 10323,

```

```

    "isrc": "GBQCP1400109",
    "label_id": 763,
    "label_name": "Big Brother Recordings, Ltd.",
    "popularity": 0.0,
    "recognition": 0.0,
    "release_date": "1995-10-02",
    "spotify_id": "5wj4E6IsrVtn8IBJQ0d0C1",
    "spotify_popularity": 75,
    "title": "Wonderwall - Remastered"
  }
]
}

```

## GET /tracks/id

Description: Get a certain track.

E.g. <https://api.futurepulse.eu/tracks/10323>.

Response (full json):

```

{
  "acousticness": 0.000453,
  "album_id": 148708,
  "album_title": "(What's The Story) Morning Glory? (Deluxe Edition) [Remastered]",
  "artist": {
    "id": 149,
    "musicbrainz_id": "39ab1aed-75e0-4140-bd47-540276886b60",
    "name": "Oasis",
    "spotify_id": "2Daxqgr0hkeH0fpeiQq2f4"
  },
  "cover": "https://i.scdn.co/image/a1eccc5f4b2158ed8c021ca5aaec245c87f300eb",
  "danceability": 0.372,
  "duration": 258,
  "energy": 0.865,
  "id": 10323,
  "instrumentalness": 0.0,
  "isrc": "GBQCP1400109",
  "liveness": 0.234,
  "loudness": -4.918,
  "recognition": [],
  "release_date": "1995-10-02",
  "speechiness": 0.0381,
  "spotify_id": "5wj4E6IsrVtn8IBJQ0d0C1",
  "spotify_popularity": 75,
  "tempo": 174.53,
  "title": "Wonderwall - Remastered",
  "valence": 0.478
}

```

## GET /tracks\_by\_isrcs

Description: Get a list of tracks by their ISRCs.

E.g. [https://api.futurepulse.eu/tracks\\_by\\_isrcs?isrc\\_codes=GBAHS1600463|GBAHS1400099](https://api.futurepulse.eu/tracks_by_isrcs?isrc_codes=GBAHS1600463|GBAHS1400099).

Response (full json):

```
[
  {
    "album_id": 90907,
    "album_title": "\u00f7 (Deluxe)",
    "artist_name": "Ed Sheeran",
    "artists": [
      {
        "artist_id": 6965,
        "artist_name": "Ed Sheeran"
      }
    ],
    "duration": 233,
    "followed": false,
    "id": 51876,
    "isrc": "GBAHS1600463",
    "label_id": 1304,
    "label_name": "Atlantic Records UK",
    "popularity": 0.0,
    "recognition": 0.0,
    "release_date": "2017-03-03",
    "spotify_id": "7qiZfU4dY1lW1lzX7mPBI3",
    "spotify_popularity": 89,
    "title": "Shape of You"
  },
  {
    "album_id": 90910,
    "album_title": "x (Deluxe Edition)",
    "artist_name": "Ed Sheeran",
    "artists": [
      {
        "artist_id": 6965,
        "artist_name": "Ed Sheeran"
      }
    ],
    "duration": 281,
    "followed": false,
    "id": 51870,
    "isrc": "GBAHS1400099",
    "label_id": 1304,
    "label_name": "Atlantic Records UK",
    "popularity": 0.0,
    "recognition": 0.0,
    "release_date": "2014-06-21",
    "spotify_id": "34gCuhDGsG4bRPIf9bb02f",
    "spotify_popularity": 85,
    "title": "Thinking Out Loud"
  }
]
```

[GET /track/isrc/{isrc}](#)

Description: Get a certain track.



E.g. <https://api.futurepulse.eu/tracks/10323>.

Response (parts of the json response have been removed for presentation purposes):

```
"success":{
  "data":[
    {
      "content_bpm":"147.9226075472",
      "content_cuetype_fade-in":"170.3854875",
      "content_cuetype_fade-out":"169673.877551",
      "content_duration":"169773.333333",
      "content_key":"Bb",
      "content_mode":"maj",
      "content_tag":{"
        "BNElectronicGenre":[
          {
            "confidence":"0.9372133165",
            "value":"Industrial music"
          },
          {
            "confidence":"0.0407237454",
            "value":"Electronic rock"
          },
          {
            "confidence":"0.0074593435",
            "value":"Hardcore"
          },
          {
            "confidence":"0.0024246769",
            "value":"Hip Hop"
          },
          {
            "confidence":"0.0021190560",
            "value":"Downtempo"
          },
          {
            "confidence":"0.0018245328",
            "value":"UK garage"
          },
          {
            "confidence":"0.0014234093",
            "value":"Video game music"
          },
          {
            "confidence":"0.0013115378",
            "value":"House"
          },
          {
            "confidence":"0.0010337494",
            "value":"Hi-NRG"
          },
          {
            "confidence":"0.0009398881",
            "value":"Jungle"
          }
        ]
      }
    }
  ]
}
```

```

    ,
    {
      "confidence": "0.0000014141",
      "value": "african"
    }
  ],
  "VocalGender": [
    {
      "confidence": "0.8211770030",
      "value": "male"
    },
    {
      "confidence": "0.1461705021",
      "value": "female"
    },
    {
      "confidence": "0.0326524948",
      "value": "instrumental"
    }
  ]
},
"filename": "4025905946051_dem389900024.mp3",
"isrc": [
  "DEM389900024"
],
"md5": "0713f1c49471116d7f589f374856d28a"
}]
}
}

```

## GET /tracks/{isrc}/recognition

Description: Get recognition of a track.

E.g. <https://api.futurepulse.eu/tracks/USRC11701901/recognition>

## Response

```

{
  "charts_positions": [
    {
      "chart_period": "day",
      "positions": [
        {
          "position": 58,
          "since": "2017-08-11",
          "size": 200,
          "until": "2017-08-11"
        },
        {

```

```

        "position": 70,
        "since": "2017-08-12",
        "size": 200,
        "until": "2017-08-12"
    },
...
    {
        "chart_period": "week",
        "positions": [
            {
                "position": 1,
                "since": "2017-08-10",
                "size": 50,
                "until": "2017-08-17"
            },
            {
                "position": 4,
                "since": "2017-08-17",
                "size": 50,
                "until": "2017-08-24"
            },
...
        ]
    },
    {
        "chart_period": "week",
        "positions": [
            {
                "position": 34,
                "since": "2017-08-27",
                "size": 50,
                "until": "2017-09-02"
            },

```

...

```
    ]
  },
  {
    "chart_period": "week",
    "positions": [
      {
        "position": 39,
        "since": "2017-12-27",
        "size": 50,
        "until": "2018-01-03"
      },
      {
        "position": 46,
        "since": "2017-12-30",
        "size": 50,
        "until": "2018-01-06"
      }
    ]
  },
  {
    "chart_period": "week",
    "positions": [
      {
        "position": 50,
        "since": "2018-02-17",
        "size": 100,
        "until": "2018-02-24"
      },
      {
        "position": 50,
        "since": "2018-02-18",
        "size": 100,
```

```

        "until": "2018-02-24"
    }
]
},
{
    "chart_period": "year",
    "positions": [
        {
            "position": 57,
            "since": "2017-01-01",
            "size": 100,
            "until": "2017-12-31"
        },
        {
            "position": 69,
            "since": "2018-01-01",
            "size": 200,
            "until": "2018-12-31"
        }
    ]
}
],
"decay": {
    "2017-08-11": 50.77040396939175,
    "2017-08-12": 50.77040396939175,
    "2017-08-13": 50.77040396939175,
    ...
},
"metrics_date": "2020-10-30",
"p_sp": 82,
"p_yt": 112612226,
"track": {
    "duration": 180,

```

```

    "id": 64603,
    "isrc": "USRC11701901",
    "pilot_related": 1,
    "release_date": "2017-08-11",
    "spotify_id": "7vGuf3Y35N4wmASOKLUVVU",
    "spotify_popularity": 87,
    "title": "Silence"
  },
  "track_metrics": {
    "popularities": [
      [
        {
          "country": "global",
          "outer_id": "7vGuf3Y35N4wmASOKLUVVU",
          "popularity_values": [
            {
              "date": "2020-10-24",
              "popularity": 82
            },
            {
              "date": "2020-10-25",
              "popularity": 82
            }
          ],
          ...
        ],
        "source": "spotify"
      ],
      {
        "country": "global",
        "outer_id": "Tx1sqYc3qas",
        "popularity_values": [
          {
            "comments": 36951,

```

```

        "date": "2020-10-24",
        "dislikes": 26963,
        "likes": 1253917,
        "views": 112293666
    },
    {
        "comments": 36960,
        "date": "2020-10-25",
        "dislikes": 26977,
        "likes": 1254583,
        "views": 112345507
    },
    ...
    ],
    "source": "youtube"
}
]
],
"uid": "test@test.gr"
},
"trec": 87.70243486952663
}

```

## GET /catalog/track\_entries

Description: Get a certain track.

E.g. [https://api.futurepulse.eu/catalog/track\\_entries/USQX91500448](https://api.futurepulse.eu/catalog/track_entries/USQX91500448)

Response

```

{
  "status": 200,
  "timestamp": 1604128215373,
  "data": {

```

```

    "id": "bs6IAHABmVswjcPhLixF",
    "outer_id": "USQX91500448",
    "organization_id": "47f60edf-1a27-4b14-8ae6-18de0356124e",
    "created_at": 1548252301173,
    "updated_at": 1580556889665,
    "medias": [
      {
        "type": "audio",
        "delivery_filepath": "USQX91500448.m4a",
        "analyzes": [
          {
            "analyzer": "musimotion:v1.3",
            "status": "success",
            "status_at": 1580843127410
          }
        ]
      },
      {
        "type": "audio",
        "delivery_filepath": "USQX91500448.m4a"
      }
    ],
    "references": [
      {
        "type": "spotify",
        "id": "0230VLNzXhX0j7CxswUt6D"
      },
      {
        "type": "isrc",
        "id": "USQX91500448"
      }
    ],
    "title": "Cheerleader - Felix Jaehn Remix Radio Edit"
  }
}

```



## POST /musimotion/search

Description: Search for one or more tracks and get audio genres information

E.g. <https://api.futurepulse.eu/musimotion/search?q=love&limit=1>

### Response

```
{
  "status":200,
  "timestamp":1604129283655,
  "data":[
    {
      "score":100.0,
      "item":{
        "id":"Hy6UAHABYNtM0iBy3993",
        "outer_id":"USAT21003370",
        "organization_id":"47f60edf-1a27-4b14-8ae6-18de0356124e",
        "created_at":1548252238641,
        "updated_at":1580557721459,
        "versioned_predicted_lexicology":{
          "v1.3":[
            {
              "id":"f1aea1a6-ca6a-4b20-b0f6-5a8ea50f6cb6",
              "category":"mood",
              "value":83.864474
            },
            {
              "id":"a03a702b-0ec4-300a-d18c-1542c9023315",
              "category":"mood",
              "value":76.77878822119325
            },
            {
              "id":"2154f72e-2004-45b4-8360-9200634c92e8",
              "category":"mood",
              "value":69.36146
            }
          ]
        }
      }
    }
  ]
}
```

```
  },
  {
    "id": "0158d0de-e706-4d42-a727-ee006188c5e1",
    "category": "mood",
    "value": 66.615576
  },
  ...
  {
    "id": "eba5e088-c286-d5e7-80a5-a1eae337b4d9",
    "category": "rhythmic_mood",
    "value": 45.11884
  },
  {
    "id": "6316bc1f-d794-4bc9-6a8c-fbe3868664f6",
    "category": "rhythmic_mood",
    "value": 37.817624
  },
  {
    "id": "095a8c30-3f6e-2a7d-ff6c-e7866676ea98",
    "category": "rhythmic_mood",
    "value": 31.307620000000004
  },
  {
    "id": "044705ca-844f-932a-d15f-4fb8742d9c6b",
    "category": "rhythmic_mood",
    "value": 10.61697
  },
  {
    "id": "39a3ef53-07d7-b274-471b-57fd23478f06",
    "category": "rhythmic_mood",
    "value": 0.0
  },
  {
```

```
"id": "2c967e7e-d9ee-817b-a332-ec8e5984ae50",
"category": "voices_family",
"value": 81.286305
},
{
  "id": "0b897061-b94f-477f-9d72-06491f14a8de",
  "category": "voices_family",
  "value": 10.0
},
{
  "id": "2f0c46c8-03c1-c9c0-cb49-fca8bdb0b84f",
  "category": "voices_family",
  "value": 0.0
},
{
  "id": "094a5811-e347-b1f4-d633-1bad0f06599b",
  "category": "situation",
  "value": 99.0
},
{
  "id": "3cf78511-f764-2f9a-ecc4-b8714a710c1c",
  "category": "situation",
  "value": 98.05714285714286
},
{
  "id": "1775137b-6281-8d53-ecc1-70936238eca9",
  "category": "situation",
  "value": 96.0
},
{
  "id": "78c47f9b-be77-054a-8b53-fbd2a5e24507",
  "category": "situation",
  "value": 93.86666666666666
```

```
},
{
  "id":"2300d119-8ccc-80ad-b245-87588946719d",
  "category":"situation",
  "value":86.4
},
{
  "id":"77fb5952-6200-6ef5-c388-ae6c0de1a992",
  "category":"situation",
  "value":84.48
},
{
  "id":"685c1fb0-523a-f601-5bf0-c1ee80149ea0",
  "category":"situation",
  "value":82.13333333333334
},
{
  "id":"da55f5de-4c57-a9ec-7c23-cfbe8ebae82",
  "category":"situation",
  "value":81.23076923076924
},
{
  "id":"16b0468f-07d0-ec81-56f4-3a1ea15d73b5",
  "category":"situation",
  "value":73.1076923076923
},
{
  "id":"6555605b-cce1-a1b0-ed1a-3a7a06960ddb",
  "category":"situation",
  "value":70.39999999999999
},
{
  "id":"3759b2b1-5ebe-5e8d-7bc6-36230b0fda1d",
```

```

        "category": "situation",
        "value": 66.0
    },
    {
        "id": "0e31c264-e1d0-9f31-ca41-368377b97744",
        "category": "situation",
        "value": 59.4
    }
]
},
"references": [
    {
        "type": "spotify",
        "id": "3Tc6dHggTsaZq1KRhzkeub"
    },
    {
        "type": "isrc",
        "id": "USAT21003370"
    }
],
"title": "Love Love Love"
}
}],
"offset": 0,
"limit": 1,
"size": 1,
"total_size": 2956
}

```

## POST /popularity/musimap/toptracks

Description: Get top tracks popularity

Example: [https://api.futurepulse.eu/popularity/musimap/toptracks?ntop=20&date=2020-10-03&days\\_diff=-30&relative=1&ids=FIHT71700003&ids\\_source=isrc](https://api.futurepulse.eu/popularity/musimap/toptracks?ntop=20&date=2020-10-03&days_diff=-30&relative=1&ids=FIHT71700003&ids_source=isrc)

Response:

```
{
  "status":200,
  "timestamp":1604140783274,
  "data":[
    {
      "popularity_date1":2.83,
      "popularity_date2":7.16,
      "difference":4.33,
      "difference_relative":152.95,
      "date1":"2020-09-03",
      "date2":"2020-10-03",
      "outer_id":"03a619e5-e31c-4e1a-bb18-e0bae7116bdc",
      "isrc":"FIHT71700003"
    }
  ],
  "size":1
}
```

## POST /catalog/search

Description: Search for one or more tracks

E.g. <https://api.futurepulse.eu/catalog/search?q=Someplace Nice>.

Response:

```
{
  "status":200,
  "timestamp":1604127981198,
  "data":[
    {
      "score":100.0,
      "item":{
        "id":"6-8VTnIBa-PBShQxUJpk",
        "outer_id":"EELK11600009",
        "organization_id":"e412e591-6036-48be-9253-728597720bf6",

```

```
"created_at":1590447919198,
"updated_at":1590447919198,
"medias":[
  {
    "type":"audio",
    "delivery_filepath":"20200525_1519/2251_6417138639899_eelk11600009.mp3",
    "analyzes":[
      {
        "analyzer":"musimotion:v1.3",
        "status":"success",
        "status_at":1590511490422
      }
    ]
  },
]
"references":[
  {
    "type":"isrc",
    "id":"EELK11600009"
  },
  {
    "type":"spotify",
    "id":"2GH6W0tAf8U03W7WeWhn0j"
  },
  {
    "type":"certh",
    "id":"2251"
  },
]
"title":"Someplace Nice",
"performers":[
  {
    "role":"owner",
    "name":"Andres Roots"
  }
]
}
```

```

},
{
  "score":93.69,
  "item":{
    "id":"gQ0VTnIBqcCEc_7CIu1S",
    "outer_id":"EELK11600030",
    "organization_id":"e412e591-6036-48be-9253-728597720bf6",
    "created_at":1590447907400,
    "updated_at":1590447907400,
    "medias":[
      {
        "type":"audio",
        "delivery_filepath":"20200525_1519/96545_6417138644855_eelk11600030.mp3",
        "analyzes":[
          {
            "analyzer":"musimotion:v1.3",
            "status":"success",
            "status_at":1590624352171
          }
        ]
      }
    ],
    "references":[
      {
        "type":"isrc",
        "id":"EELK11600030"
      },
      {
        "type":"spotify",
        "id":"6JB76RkZAGQPptqhN1VAwL"
      },
      {
        "type":"certh",
        "id":"96545"
      }
    ],
  }
}

```



```

    "title": "Someplace Nice",
    "performers": [
      {
        "role": "owner",
        "name": "Andres Roots"
      }
    ]
  }
}],
"offset": 0,
"limit": 100,
"size": 2,
"total_size": 2
}

```

## GET /popularity

Description: Get a certain track's popularity.

E.g.

[https://api.futurepulse.eu/popularity/GBAHS1900714?source=isrc&output=deezer,youtube,bmat&date\\_from=2018-09-01](https://api.futurepulse.eu/popularity/GBAHS1900714?source=isrc&output=deezer,youtube,bmat&date_from=2018-09-01).

Response (parts of the json response have been removed for presentation purposes):

```

{
  "status": 200,
  "timestamp": 1604124228699,
  "data": [
    {
      "source": "deezer",
      "country": "global",
      "outer_id": "710164312",
      "popularity_values": [
        {
          "rank": 987091,
          "date": "2019-09-24"
        },
        {
          "rank": 987091,
          "date": "2019-09-25"
        },
        {
          "rank": 987091,
          "date": "2019-09-26"
        }
      ]
    }
  ]
}

```

```

    },
    {
      "rank": 987091,
      "date": "2019-09-27"
    },
    {
      "rank": 987091,
      "date": "2019-09-28"
    },
    ...
  ]
},
{
  "source": "youtube",
  "error": "No reference found"
},
{
  "source": "bmat",
  "country": "global",
  "outer_id": "84bdc0a5-943e-4086-b3d6-01af2d5b1bdf",
  "popularity_values": [
    {
      "playcount": 3,
      "date": "2019-07-13"
    },
    {
      "playcount": 5,
      "date": "2019-07-24"
    },
    {
      "playcount": 2,
      "date": "2019-07-25"
    },
    ...
  ]
}
]
}
}

```

## GET /popularity/sources

Description: Get available sources and their fields for the popularity endpoint.

E.g. <https://api.futurepulse.eu/popularity/sources>.

### Response

```

{
  "status":200,
  "timestamp":1604126209170,
  "data":{

```

```
"bmat":[
  "playcount"],
"deezer":[
  "rank"],
"musimap":[
  "popularity"],
"spotify":[
  "popularity"],
"spotify_analytics":[
  "age_0-17",
  "age_18-22",
  "age_23-27",
  "age_28-34",
  "age_35-44",
  "age_45-59",
  "age_60-150",
  "age_unknown",
  "product_free",
  "product_premium",
  "sex_female",
  "sex_male",
  "sex_neutral",
  "sex_unknown",
  "streams",
  "total_all"],
"spotify_charts":[
  "position_top200",
  "streams"],
"youtube":[
  "comments",
  "dislikes",
  "likes",
  "views"]
```

```
}  
}
```

### GET /popularity/spotify/countries

Description: Get list of countries for which popularity data is available from a particular source (the list contains ISO 3166-1 alpha-2 two letter codes for countries and/or "global")

E.g. <https://api.futurepulse.eu/popularity/sources>.

#### Response

```
{  
  "status":200,  
  "timestamp":1604141623682,  
  "data":[  
    "global"],  
  "size":0  
}
```

### GET /artists/{artist\_id}/venue\_popularities

Description: Get venue popularities for a certain artist.

E.g. [https://api.futurepulse.eu/artists/47/venue\\_popularities](https://api.futurepulse.eu/artists/47/venue_popularities).

Response (parts of the json response have been removed for presentation purposes):

```
{ {  
  "city_popularities": [  
    {  
      "city": "Manchester",  
      "country": "GB",  
      "score": 0.006006071555065963,  
      "since": "2019-07-22",  
      "until": "2020-07-20"  
    },  
    {  
      "city": "Glasgow",  
      "country": "GB",
```

```
"score": 0.005123491745150355,
"since": "2019-07-22",
"until": "2020-07-20"
},
{
"city": "London",
"country": "GB",
"score": 0.004395766107359122,
"since": "2019-07-22",
"until": "2020-07-20"
},
...
}
```

## GET /venue\_cities

Description: Get venue cities

E.g. [https://api.futurepulse.eu/venue\\_cities](https://api.futurepulse.eu/venue_cities).

Response (parts of the json response have been removed for presentation purposes):

```
{
"cities": [
{
"city": "Brussels",
"count": 14738,
"country": "BE"
},
{
"city": "Paris",
"count": 14738,
"country": "FR"
},
{

```

```
    "city": "Berlin",
    "count": 14738,
    "country": "DE"
  },
  ...
  {
    "city": "Utrecht",
    "count": 14738,
    "country": "NL"
  }
}
```

### GET /gigs

Description: Get a list of gigs currently on the platform.

E.g. [https://api.futurepulse.eu/gigs?page=1&per\\_page=5&artist\\_name=wehbba&venue=tokyo](https://api.futurepulse.eu/gigs?page=1&per_page=5&artist_name=wehbba&venue=tokyo)

Output (parts of the json response have been removed for presentation purposes):

```
{
  "artist_id": "278025",
  "count": 0,
  "gigs": [
  ],
  "page": "1",
  "per_page": "5",
  "venue": "tokyo"
}
```

### GET /artists/{artist\_id}/gigs

Description: Get the gigs of an artist.

E.g. <https://api.futurepulse.eu/artists/19646/gigs>

Output (parts of the json response have been removed for presentation purposes):

```
{
  "count": 196,
```

```

"gigs": [
  {
    "_id": "102105286",
    "artist_id": "278025",
    "artists": [
      {
        "id": "278025",
        "name": "Wehbba"
      }
    ],
    "datetime": "2020-03-28T23:00:00",
    "datetime_str": "2020-03-28T23:00:00",
    "description": "Fusion Club",
    "id": "102105286",
    "inserted": "2020-04-17T18:38:21",
    "title": "",
    "url":
    "https://www.bandsintown.com/e/102105286?app_id=65cb50b645c2ea2e99e7cf6a36b3287b&came_from=267
    &utm_medium=api&utm_source=public_api&utm_campaign=event",
    "venue": {
      "city": "M\u00fcnster",
      "country": "Germany",
      "country_code": "DE",
      "latitude": "51.944106",
      "longitude": "7.638996",
      "name": "Fusion Club",
      "region": ""
    }
  },
  ...
]

```

GET /musimotion

Description: Get a track information.

E.g. <https://api.futurepulse.eu/musimotion?reference=USQX91500448>.

Response (parts of the json response have been removed for presentation purposes):

```
{
  "status": 200,
  "timestamp": 1604123398374,
  "data": {
    "seed": {
      "id": "bS6IAHABmVswjcPhLixF",
      "outer_id": "USQX91500448",
      "organization_id": "47f60edf-1a27-4b14-8ae6-18de0356124e",
      "created_at": 1548252301173,
      "updated_at": 1580556889665,
      "references": [
        {
          "type": "spotify",
          "id": "0230VLNzXhX0j7CxswUt6D"
        },
        {
          "type": "isrc",
          "id": "USQX91500448"
        }
      ],
      "title": "Cheerleader - Felix Jaehn Remix Radio Edit"
    },
    "lexicology": {
      "mood": [
        {
          "score": 51.22207100000001,
          "item": {
            "id": "2cc67cfa-8c2d-44f1-b045-7d62608cd88a",
            "name": "Warm-hearted"
          }
        }
      ],
      {
```



```
"score": 50.922847000000004,
"item": {
  "id": "e47037a9-cd12-4936-8c63-4c3795adb120",
  "name": "Innocent"
}
},
{
  "score": 47.49481725654901,
  "item": {
    "id": "a03a702b-0ec4-300a-d18c-1542c9023315",
    "name": "Good vibrations"
  }
},
{
  "score": 42.68182583359682,
  "item": {
    "id": "f05f72ad-e056-cea2-3bd2-e109efa0c02b",
    "name": "Happiness"
  }
},
{
  "score": 42.565614000000004,
  "item": {
    "id": "f1aea1a6-ca6a-4b20-b0f6-5a8ea50f6cb6",
    "name": "Friendly"
  }
},
...
{
  "score": 31.779837999999998,
  "item": {
    "id": "2154f72e-2004-45b4-8360-9200634c92e8",
    "name": "Sentimental"
```

```
    }
  }
],
"acoustic_attribute": [
  {
    "score": 189.2426300048828,
    "item": {
      "id": "63588566-5b72-4d99-a685-1aade8dfe964",
      "name": "Duration"
    }
  },
  {
    "score": 118.0,
    "item": {
      "id": "582ba553-1b91-4425-88e4-d58432c9fee5",
      "name": "Bpm"
    }
  },
  {
    "score": 100.0,
    "item": {
      "id": "06132ef1-2688-4032-9232-01a699193748",
      "name": "Spectral Energy"
    }
  },
  {
    "score": 39.342284241989844,
    "item": {
      "id": "93a6f555-450e-4d1f-b3b4-0631d71f7441",
      "name": "Rhythm Presence"
    }
  },
  {
```

```
"score": 7.0,
"value_label": "E",
"item": {
  "id": "82fe3702-fca7-4493-8499-1f574d446acc",
  "name": "Key"
}
},
{
  "score": 1.0,
  "value_label": "major",
  "item": {
    "id": "2a246dc3-e21f-4806-a91e-e6d621436310",
    "name": "Key Scale"
  }
}
],
"genre": [
  {
    "score": 39.67881903,
    "item": {
      "id": "106089d0-7afd-46b2-bc92-a17b3a29d086",
      "name": "Latin Pop - Latin Dance"
    }
  },
  {
    "score": 28.6923255,
    "item": {
      "id": "20152b88-901e-4c57-867f-1ebe2176be40",
      "name": "Pop"
    }
  },
  {
    "score": 26.7274745,
```

```
"item": {
  "id": "e1e8d1a3-bc09-41bd-8108-954b839d7f94",
  "name": "Folk"
},
{
  "score": 17.7408896,
  "item": {
    "id": "03af43e6-8df1-4f07-9d25-8f6c2854b31b",
    "name": "House"
  }
},
{
  "score": 15.27477,
  "item": {
    "id": "aa90e487-352c-4da4-b3fa-1350ac08db0a",
    "name": "Pop Rock"
  }
},
{
  "score": 15.239704000000001,
  "item": {
    "id": "46d7c399-fae8-4130-b054-3812af530f80",
    "name": "Reggae"
  }
},
],
"rhythmic_mood": [
  {
    "score": 66.75412,
    "item": {
      "id": "eba5e088-c286-d5e7-80a5-a1eae337b4d9",
      "name": "Staccato"
    }
  }
]
```

```

    }
  },
  {
    "score": 18.758357,
    "item": {
      "id": "6316bc1f-d794-4bc9-6a8c-fbe3868664f6",
      "name": "Flowing"
    }
  },
  {
    "score": 17.781968,
    "item": {
      "id": "095a8c30-3f6e-2a7d-ff6c-e7866676ea98",
      "name": "Lyrical"
    }
  }
],
"voices_family": [
  {
    "score": 88.93059,
    "item": {
      "id": "2c967e7e-d9ee-817b-a332-ec8e5984ae50",
      "name": "Male Vocal"
    }
  },
  {
    "score": 0.0,
    "item": {
      "id": "0b897061-b94f-477f-9d72-06491f14a8de",
      "name": "Instrumental (no vocal)"
    }
  },
  {

```

```

    "score": 0.0,
    "item": {
      "id": "2f0c46c8-03c1-c9c0-cb49-fca8bdb0b84f",
      "name": "Female Vocal"
    }
  },
  "situation": [
    {
      "score": 99.0,
      "item": {
        "id": "d6478df6-5cc2-7e96-d4ab-5a024f5a433c",
        "name": "Household chores"
      }
    },
    {
      "score": 90.00000000000001,
      "item": {
        "id": "617e4017-7f57-f662-7f91-edb61d2d576e",
        "name": "Birthday"
      }
    },
    {
      "score": 89.375,
      "item": {
        "id": "3cf78511-f764-2f9a-ecc4-b8714a710c1c",
        "name": "Summer"
      }
    },
    {
      "score": 82.5,
      "item": {
        "id": "b3677017-79f7-11cc-5558-37e999603848",

```

```
    "name": "Family"
  }
},
{
  "score": 80.00000000000001,
  "item": {
    "id": "562a3fcf-5c39-757a-1428-c5390123883e",
    "name": "Party time"
  }
},
{
  "score": 77.64705882352942,
  "item": {
    "id": "98447fe8-312f-4261-8d13-24fb0ba7324f",
    "name": "Carnival"
  }
},
{
  "score": 77.0,
  "item": {
    "id": "825e1065-63f2-b131-ec07-f0024dac472a",
    "name": "Cruise"
  }
},
{
  "score": 73.33333333333331,
  "item": {
    "id": "78c47f9b-be77-054a-8b53-fbd2a5e24507",
    "name": "Just chill"
  }
},
{
  "score": 73.33333333333331,
```

```

    "item": {
      "id": "377a1209-b585-ddf9-c807-2f5a2659e0b9",
      "name": "Wedding Party"
    }
  },
  {
    "score": 73.33333333333331,
    "item": {
      "id": "fdd1bde3-bb22-571a-d501-fe4e8d3a0daa",
      "name": "Bar"
    }
  }
]
}
}
}
}

```

## GET /genres

Description: Get a list of the genres.

E.g. <https://api.futurepulse.eu/genres>

Response (parts of the json response have been removed for presentation purposes):

```

{
  "genres": [
    {
      "description": null,
      "freebase_id": null,
      "genre": "21st century classical",
      "id": 297,
      "source": "spotify",
      "wikidata_id": null,
      "wikipedia_link": null
    },
    {
      "description": null,
      "freebase_id": null,
      "genre": "432hz",
      "id": 2148,
      "source": "spotify",
      "wikidata_id": null,
      "wikipedia_link": null
    }
  ]
}

```



```
    }
  ],
  "num_per_page": 20,
  "page": 1,
  "total": 3418
}
```

## GET /genres/id/popularity

Description: Get a specific genre.

E.g. <https://api.futurepulse.eu/genres/2853/popularity>.

Response (parts of the json response have been removed for presentation purposes):

```
{
  "country": "WRL",
  "resolution": "month",
  "scores": [

  ],
  "statistics": [
    {
      "avg_raw_count": 465.4667,
      "max_raw_count": 14182,
      "min_raw_count": 0,
      "num": 60,
      "resolution": "month",
      "since": "2015-01-01",
      "sum_raw_count": 27928,
      "until": "2015-02-01"
    },
    {
      "avg_raw_count": 452.3833,
      "max_raw_count": 13555,
      "min_raw_count": 0,
      "num": 60,
      "resolution": "month",
      "since": "2015-02-01",
      "sum_raw_count": 27143,
      "until": "2015-03-01"
    }
  ]
}
```

## GET /genres\_ranking

Description: Get the genres ranking per country

E.g. [https://api.futurepulse.eu/genres\\_ranking?country=global&source=spotify&date=2020-02-09](https://api.futurepulse.eu/genres_ranking?country=global&source=spotify&date=2020-02-09).

Response (parts of the json response have been removed for presentation purposes):

```
{
  "country": "global",
  "date": "2020-02-15",
  "genres": [
    {
      "country_normalized_popularity": 0.9282,
      "eukl": 0.6190730235008944,
      "followed": false,
      "growth": 0.0067,
      "id": 401,
      "name": "pop",
      "period_normalized_popularity": 0.0175,
      "popularity": 0.203,
      "prob": 0.11820143568062712,
      "raw_count": 2684.0,
      "relative_growth": 0.0343
    },
    {
      "country_normalized_popularity": 0.6112,
      "eukl": 0.3159947996260154,
      "followed": false,
      "growth": 0.0129,
      "id": 433,
      "name": "rap",
      "period_normalized_popularity": 0.0115,
      "popularity": 0.1337,
      "prob": 0.06033381776544678,
      "raw_count": 1370.0,
      "relative_growth": 0.1067
    },
    ...
  ],
  "growth_span": "3_months",
  "limit_by": "pagerank_09",
  "metrics":
  "raw_count,prob,eukl,pagerank_09,pagerank_09_3_months_dif,pagerank_09_3_months_relative_dif",
  "num_per_page": 100,
  "page": 1,
  "since": "2020-02-15",
  "statistics": {
    "max_scores_all_countries_current_date": {
      "birank_05": 0.08021069862295864,
      "birank_075": 0.051353423124978384,
      "birank_09": 0.026688808072190755,
      "birank_095": 0.021580400342004907,
      "birank_099": 0.015151405985717533,
      "eukl": 0.876111372480566,
      "pagerank_05": 0.1503499980698359,
      "pagerank_075": 0.09603968113851578,
      "pagerank_09": 0.05943977059964959,
      "pagerank_095": 0.04236653932524104,
      "pagerank_099": 0.015173089179082926,
      "prob": 0.3050017488632389,
      "raw_count": 4472.0
    },
  },
}
```

```

"max_scores_all_countries_overall": {
  "birank_05": 0.08902555906392205,
  "birank_075": 0.05701730332976008,
  "birank_09": 0.029577845507835692,
  "birank_095": 0.02637378748199812,
  "birank_099": 0.018492217641108172,
  "eukl": 0.9042259999354403,
  "pagerank_05": 0.17527519687192142,
  "pagerank_075": 0.12477129282086807,
  "pagerank_09": 0.08610341436128409,
  "pagerank_095": 0.06059483511795246,
  "pagerank_099": 0.020614157688431504,
  "prob": 0.33880248833592536,
  "raw_count": 4830.0
},
"max_scores_country_specific_overall": {
  "birank_05": 0.034566122747570864,
  "birank_075": 0.021936841521876988,
  "birank_09": 0.011111379505179493,
  "birank_095": 0.007716867195316601,
  "birank_099": 0.002445912389429213,
  "eukl": 0.6883158497853203,
  "pagerank_05": 0.05808797438255522,
  "pagerank_075": 0.03441865981646764,
  "pagerank_09": 0.01883317221791793,
  "pagerank_095": 0.012408022159698126,
  "pagerank_099": 0.0056866299379599375,
  "prob": 0.13000817661488143,
  "raw_count": 3469.0
}
},
"total_genres": 3636
}

```

GET /mergedtrack/isrc/{isrc}

Description: Get the audio genres for a certain track

E.g. <https://api.futurepulse.eu/mergedtrack/isrc/USQX91500448>.

Response:

```

{
  "BPM": 117.9943478648,
  "Duration": 189.2426300048828,
  "Key": "E",
  "KeyScale": "major",
  "Content_cuetype_fade-in": 0.0,
  "Content_cuetype_fade-out": 178153.061224,

```

```
"VoicesFamily":[
  {
    "confidence":88.93059,
    "value":"Low vocal range"
  },
  {
    "confidence":0.0,
    "value":"Instrumental (no vocal)"
  },
  {
    "confidence":0.0,
    "value":"High vocal range"
  }
],
"Genre":[
  {
    "genre":"dance/edm",
    "score":74.57491171000001,
    "source":"IRCAM"
  },
  {
    "genre":"house",
    "score":51.33808857,
    "source":"IRCAM"
  },
  {
    "genre":"pop",
    "score":44.55883902,
    "source":"IRCAM"
  },
  ...
],
"Mood":[
  {
    "score":51.22207100000001,
    "item":{
```

```

        "id": "2cc67cfa-8c2d-44f1-b045-7d62608cd88a",
        "name": "Warm-hearted"
    }
},
{
    "score": 50.922847000000004,
    "item": {
        "id": "e47037a9-cd12-4936-8c63-4c3795adb120",
        "name": "Innocent"
    }
},
{
    "score": 47.49481725654901,
    "item": {
        "id": "a03a702b-0ec4-300a-d18c-1542c9023315",
        "name": "Good vibrations"
    }
},
...],
"Rhythmic_mood": [
    {
        "score": 66.75412,
        "item": {
            "id": "eba5e088-c286-d5e7-80a5-a1eae337b4d9",
            "name": "Staccato"
        }
    },
    {
        "score": 18.758357,
        "item": {
            "id": "6316bc1f-d794-4bc9-6a8c-fbe3868664f6",
            "name": "Flowing"
        }
    },
    {

```

```
"score":17.781968,
"item":{
  "id":"095a8c30-3f6e-2a7d-ff6c-e7866676ea98",
  "name":"Lyrical"
}
}],
"Situation":[
{
  "score":99.0,
  "item":{
    "id":"d6478df6-5cc2-7e96-d4ab-5a024f5a433c",
    "name":"Household chores"
  }
},
...],
"Acoustic_attribute":[
{
  "score":100.0,
  "item":{
    "id":"06132ef1-2688-4032-9232-01a699193748",
    "name":"Spectral Energy"
  }
},
{
  "score":39.342284241989844,
  "item":{
    "id":"93a6f555-450e-4d1f-b3b4-0631d71f7441",
    "name":"Rhythm Presence"
  }
},
...],
"BNElectronicGenre":[
{
  "confidence":31.80992713,
  "value":"House"
```

```

    },
    {
      "confidence":31.797929200000002,
      "value":"Hip Hop"
    },
    {
      "confidence":11.60927144,
      "value":"IDM"
    },
    ...]
  }

```

### POST /musimotion/live

Description: Submit a file and get its lexicology predictions

E.g. <https://api.musimap.io/musimotion/live>.

Response:

```

{
  "category_name": [
    {
      "value": 99.99,
      "value_label": "string",
      "item": {
        "id": "string",
        "name": "string",
        "category_name": "string",
        "description": "string"
      }
    }
  ]
}

```

### GET /playlists

Description: Get a list if the playlists

E.g.

[https://api.futurepulse.eu/playlists?sort\\_by=avg\\_popularity&order=DESC&platform=spotify&num\\_per\\_page=10&page=1&type=&genre=&q=&curat.](https://api.futurepulse.eu/playlists?sort_by=avg_popularity&order=DESC&platform=spotify&num_per_page=10&page=1&type=&genre=&q=&curat.)

Response (parts of the json response have been removed for presentation purposes):

```

{
  "num_per_page": 10,

```

```

"page": 1,
"playlists": [
  {
    "add_rate": 0.44,
    "avg_popularity": 87.68,
    "cover_url": "https://i.scdn.co/image/ab67706f00000000208af8afd6faca9bd10dc1262",
    "curator": "spotify",
    "curator_name": "Spotify",
    "curator_url": "https://open.spotify.com/user/spotify",
    "delete_rate": 0.8108108108108109,
    "description": "Arizona Zervas is on top of the Hottest 50!",
    "followers": 24699467,
    "followers_change": 349179,
    "genres": [
      {
        "count": 57,
        "genre": "pop"
      },
      {
        "count": 30,
        "genre": "dance pop"
      },
      {
        "count": 28,
        "genre": "post-teen pop"
      },
      {
        "count": 10,
        "genre": "rap"
      },
      {
        "count": 8,
        "genre": "electropop"
      },
      {
        "count": 8,
        "genre": "pop rap"
      },
      {
        "count": 7,
        "genre": "r&b"
      },
      {
        "count": 7,
        "genre": "edm"
      },
      {
        "count": 6,
        "genre": "tropical house"
      },
      {
        "count": 6,
        "genre": "trap music"
      },
      {
        "count": 4,
        "genre": "uk pop"
      }
    ]
  }
]

```



```

    }
  ],
  "id": "37i9dQZF1DXcBWIGoYBM5M",
  "name": "Today's Top Hits",
  "num_of_tracks": 50,
  "platform": "spotify",
  "tracked_at": "2019-11-14 04:52:00",
  "type": "frontline",
  "updated_at": "2019-11-14 00:00:13",
  "url": "https://open.spotify.com/playlist/37i9dQZF1DXcBWIGoYBM5M"
}
],
"total": 1914
}

```

## GET /playlist/id

Description: Get a playlist by id

E.g. <https://api.futurepulse.eu/playlists/37i9dQZF1DXcBWIGoYBM5M>.

Response (parts of the json response have been removed for presentation purposes):

```

{
  "add_rate": 0.44,
  "avg_popularity": 87.68,
  "cover_url": "https://i.scdn.co/image/ab67706f0000000208af8afd6faca9bd10dc1262",
  "curator": "spotify",
  "curator_name": "Spotify",
  "curator_url": "https://open.spotify.com/user/spotify",
  "days_histogram": [
    {
      "count": 0,
      "day": "Monday",
      "week_num": 0
    },
    {
      "count": 0,
      "day": "Tuesday",
      "week_num": 1
    },
    {
      "count": 0,
      "day": "Wednesday",
      "week_num": 2
    },
    {
      "count": 50,
      "day": "Thursday",
      "week_num": 3
    },
    {
      "count": 0,
      "day": "Friday",
      "week_num": 4
    }
  ]
}

```

```

    },
    {
      "count": 0,
      "day": "Saturday",
      "week_num": 5
    },
    {
      "count": 0,
      "day": "Sunday",
      "week_num": 6
    }
  ],
  "days_in_playlist_histogram": [
    {
      "count": 50,
      "days": 0
    }
  ],
  "delete_rate": 0.8108108108108109,
  "description": "Arizona Zervas is on top of the Hottest 50!",
  "followers": 24699467,
  "followers_change": 349179,
  "genres": [
    {
      "count": 57,
      "genre": "pop"
    },
    {
      "count": 30,
      "genre": "dance pop"
    },
    {
      "count": 28,
      "genre": "post-teen pop"
    },
    {
      "count": 10,
      "genre": "rap"
    },
    {
      "count": 8,
      "genre": "electropop"
    },
    {
      "count": 8,
      "genre": "pop rap"
    }
  ],
  "metrics": [
    {
      "add_rate": 1.0,
      "avg_popularity": 83.0,
      "delete_rate": 0.0,
      "deleted_tracks": 0,
      "followers": 23368742,
      "new_tracks": 50,
      "popularity_std": 14.664242223858688,

```

```

    "time": "2019-06-13 04:52:00",
    "tracks": 50
  },
  {
    "add_rate": 1.0,
    "avg_popularity": 77.42,
    "delete_rate": 0.0,
    "deleted_tracks": 6,
    "followers": 23386967,
    "new_tracks": 6,
    "popularity_std": 24.761332758961096,
    "time": "2019-06-15 04:52:00",
    "tracks": 50
  }{
    "add_rate": 0.5,
    "avg_popularity": 83.92,
    "delete_rate": 0.6794871794871795,
    "deleted_tracks": 0,
    "followers": 23929956,
    "new_tracks": 0,
    "popularity_std": 8.36143528349051,
    "time": "2019-08-14 04:52:00",
    "tracks": 50
  },
  {
    "add_rate": 0.44,
    "avg_popularity": 78.66,
    "delete_rate": 0.6744186046511628,
    "deleted_tracks": 4,
    "followers": 23941283,
    "new_tracks": 4,
    "popularity_std": 22.169898511269736,
    "time": "2019-08-16 04:52:00",
    "tracks": 50
  }
],
"name": "Today's Top Hits",
"num_of_tracks": 50,
"platform": "spotify",
"popularity_histogram": [
  {
    "count": 1,
    "popularity": "0"
  },
  {
    "count": 0,
    "popularity": "10"
  },
  {
    "count": 0,
    "popularity": "20"
  },
  {
    "count": 0,
    "popularity": "30"
  },
  {

```

```

    "count": 0,
    "popularity": "40"
  },
  {
    "count": 1,
    "popularity": "50"
  },
  {
    "count": 0,
    "popularity": "60"
  },
  {
    "count": 1,
    "popularity": "70"
  },
  {
    "count": 22,
    "popularity": "80"
  },
  {
    "count": 25,
    "popularity": "90"
  }
],
"popularity_percentile": {
  "p25": 87.0,
  "p5": 75.6,
  "p50": 89.5,
  "p75": 94.0,
  "p95": 97.1
},
"popularity_std": 14.263856421038456,
"release_date_difference": {
  "avg": 78,
  "max": 594,
  "min": 1,
  "p25": 28,
  "p5": 13,
  "p50": 48,
  "p75": 102,
  "p95": 184,
  "std": 90
},
"tracked_at": "2019-11-14 04:52:00",
"tracks": [
  {
    "added_at": "2019-11-14 00:00:13",
    "album_id": "4i3rAwPw7Ln2YrKDusaWyT",
    "artists": [
      {
        "artist_id": "6qqNVTkY8uBg9cP3Jd7DAH",
        "name": "Billie Eilish",
        "url": "https://open.spotify.com/artist/6qqNVTkY8uBg9cP3Jd7DAH"
      }
    ]
  },
  {
    "isrc": "USUM71922577",
    "label": "Darkroom/Interscope Records",

```

```

    "popularity": 0,
    "position": 1,
    "release_date": "2019-11-13 00:00:00",
    "release_date_precision": "day",
    "title": "everything i wanted",
    "track_id": "3ZCTVFBt2Brf31RLEnCKWJ"
  },
  {
    "added_at": "2019-11-14 00:00:13",
    "album_id": "0ix3XtPV1LwmZADsprKxcp",
    "artists": [
      {
        "artist_id": "6M2wZ9GZgrQXHCFfjv46we",
        "name": "Dua Lipa",
        "url": "https://open.spotify.com/artist/6M2wZ9GZgrQXHCFfjv46we"
      }
    ],
    "isrc": "GBAHT1901121",
    "label": "Warner Records",
    "popularity": 93,
    "position": 2,
    "release_date": "2019-10-31 00:00:00",
    "release_date_precision": "day",
    "title": "Don't Start Now",
    "track_id": "6WrI0LAC5M1Rw2MnX2ZvEg"
  }
],
"tracks_age": [
  {
    "age": "within a week",
    "count": 1
  },
  {
    "age": "one week to one month",
    "count": 13
  },
  {
    "age": "one to six months",
    "count": 31
  },
  {
    "age": "more than six months",
    "count": 5
  }
],
"type": "frontline",
"updated_at": "2019-11-14 00:00:13",
"url": "https://open.spotify.com/playlist/37i9dQZF1DXcBWIGoYBM5M"
}

```

[GET /tracks/{spotify\\_track\\_id}/playlists](#)

Description: Get playlists of a track endpoint

E.g.

[https://api.futurepulse.eu/tracks/0z4kyKuFGmUV7WQ7gB1sAp/playlists?platform=spotify&num\\_per\\_page=10&page=1](https://api.futurepulse.eu/tracks/0z4kyKuFGmUV7WQ7gB1sAp/playlists?platform=spotify&num_per_page=10&page=1).

Response (parts of the json response have been removed for presentation purposes):

```
{
  "max_followers": null,
  "max_popularity": null,
  "min_followers": null,
  "min_popularity": null,
  "num_per_page": 10,
  "page": 1,
  "pipeline": {
  },
  "platform": "spotify",
  "playlists": [
    {
      "add_rate": 0.2542372881355932,
      "added_at": "2020-10-28 14:04:37",
      "album_id": "1Rule9WvxRuGh3S7X2Koyk",
      "album_name": "Goliath",
      "artist_ids": [
        "1ZrBGJWLL8NiAjgNifCy90"
      ],
      "avg_popularity": 68.27966101694915,
      "cover_url": "https://i.scdn.co/image/ab67706c0000bebbd9e49c8ce5b2d72d0b773343",
      "curator": "sonymusicentertainment",
      "curator_name": "Filtr Sweden",
      "curator_url": "https://open.spotify.com/user/sonymusicentertainment",
      "delete_rate": 0.8436944937833037,
      "description": "\u00c5rets st\u00f6rsta l\u00e5tar, hetaste svenska musik, bubblare och b\u00e5sta hits idag. Alltid uppdaterad med de senaste hitsen just nu. \u00bb Molly Sand\u00e9n & Newkid",
      "followers": 144817,
      "followers_change": 178,
      "name": "Hits 2020",
      "num_of_tracks": 118,
      "platform": "spotify",
      "playlist_id": "7AnvOrPIQLV8VIASFhN07Z",
      "popularity_std": 24.528048832838447,
      "representative_genres": [
        "pop",
        "swedish pop",
        "dance pop",
        "rap",
        "post-teen pop",
        "pop rap",
        "trap music",
        "hip hop",
        "tropical house",
        "southern hip hop"
      ],
      "tracked_at": "2020-10-29 14:01:00",
      "updated_at": "2020-10-28 14:04:37",
    }
  ]
}
```

```

    "url": "https://open.spotify.com/playlist/7AnvOrPIQLV8VIASFhN07Z"
  },
  ...
],
"playlists_count": 57,
"recommended": false,
"sort_by": {
  "added_at": -1
},
"track_id": "0z4kyKuFGmUV7WQ7gB1sAp"
}

```

## GET /playlists/{playlist\_id}/audio\_features

Description: Get playlist audio features

E.g. [https://api.futurepulse.eu/playlists/37i9dQZF1DXbTxAdrVG2l/audio\\_features](https://api.futurepulse.eu/playlists/37i9dQZF1DXbTxAdrVG2l/audio_features).

Response (parts of the json response have been removed for presentation purposes):

```

{
  "audio_features": [
    {
      "_id": "013Awviz1lIUeC2F0Bz0nh",
      "acousticness": 0.403,
      "danceability": 0.555,
      "duration_ms": 277267,
      "energy": 0.698,
      "instrumentalness": 0,
      "key": 0,
      "liveness": 0.0851,
      "loudness": -8.874,
      "mode": 1,
      "speechiness": 0.0287,
      "tempo": 168.12,
      "time_signature": 4,
      "valence": 0.816
    },
    {
      "_id": "01qCxpvyIm0a0RG3uYxE",
      "acousticness": 0.00338,
      "danceability": 0.457,
      "duration_ms": 235293,
      "energy": 0.892,
      "instrumentalness": 4.46e-06,
      "key": 10,
      "liveness": 0.141,
      "loudness": -5.926,
      "mode": 0,
      "speechiness": 0.0986,
      "tempo": 93.664,
      "time_signature": 4,
      "valence": 0.508
    },
    ...
  ],

```

```

"histograms": {
  "acousticness_histogram": [
    {
      "acousticness": "0.000133",
      "count": 46
    },
    {
      "acousticness": "0.08671969999999998",
      "count": 16
    },
    {
      "acousticness": "0.17330639999999997",
      "count": 3
    },
    {
      "acousticness": "0.25989309999999993",
      "count": 11
    },
    {
      "acousticness": "0.34647979999999995",
      "count": 9
    },
    {
      "acousticness": "0.43306649999999997",
      "count": 2
    },
    {
      "acousticness": "0.5196531999999999",
      "count": 5
    },
    {
      "acousticness": "0.6062398999999999",
      "count": 4
    },
    {
      "acousticness": "0.6928266",
      "count": 1
    },
    {
      "acousticness": "0.7794133",
      "count": 2
    }
  ],
  "danceability_histogram": [
    {
      "count": 4,
      "danceability": "0.243"
    },
    {
      "count": 1,
      "danceability": "0.3103"
    },
    {
      "count": 5,
      "danceability": "0.3776"
    },
    {
      "count": 11,

```



```

    "danceability": "0.44489999999999996"
  },
  {
    "count": 17,
    "danceability": "0.5122"
  },
  {
    "count": 18,
    "danceability": "0.5795"
  },
  {
    "count": 18,
    "danceability": "0.6468"
  },
  {
    "count": 12,
    "danceability": "0.7141"
  },
  {
    "count": 7,
    "danceability": "0.7814"
  },
  {
    "count": 6,
    "danceability": "0.8487"
  }
}],
"energy_histogram": [
  {
    "count": 3,
    "energy": "0.249"
  },
  {
    "count": 10,
    "energy": "0.3212"
  },
  {
    "count": 6,
    "energy": "0.39339999999999997"
  },
  {
    "count": 12,
    "energy": "0.4656"
  },
  {
    "count": 9,
    "energy": "0.5378000000000001"
  },
  {
    "count": 10,
    "energy": "0.61"
  },
  {
    "count": 9,
    "energy": "0.6822"
  },
  {
    "count": 8,

```

```

    "energy": "0.7544"
  },
  {
    "count": 19,
    "energy": "0.8266"
  },
  {
    "count": 13,
    "energy": "0.8988"
  }
}],
"instrumentalness_histogram": [
  {
    "count": 95,
    "instrumentalness": "0.0"
  },
  {
    "count": 1,
    "instrumentalness": "0.022"
  },
  {
    "count": 0,
    "instrumentalness": "0.044"
  },
  {
    "count": 1,
    "instrumentalness": "0.066"
  },
  {
    "count": 0,
    "instrumentalness": "0.088"
  },
  {
    "count": 0,
    "instrumentalness": "0.10999999999999999"
  },
  {
    "count": 0,
    "instrumentalness": "0.132"
  },
  {
    "count": 0,
    "instrumentalness": "0.154"
  },
  {
    "count": 0,
    "instrumentalness": "0.176"
  },
  {
    "count": 2,
    "instrumentalness": "0.19799999999999998"
  }
],
"key_histogram": [
  {
    "count": 25,
    "key": "0.0"
  },
  {

```

```

    "count": 12,
    "key": "1.1"
  },
  {
    "count": 3,
    "key": "2.2"
  },
  {
    "count": 7,
    "key": "3.3000000000000003"
  },
  {
    "count": 15,
    "key": "4.4"
  },
  {
    "count": 8,
    "key": "5.5"
  },
  {
    "count": 3,
    "key": "6.6000000000000005"
  },
  {
    "count": 5,
    "key": "7.700000000000001"
  },
  {
    "count": 7,
    "key": "8.8"
  },
  {
    "count": 14,
    "key": "9.9"
  }
],
"liveness_histogram": [
  {
    "count": 39,
    "liveness": "0.0326"
  },
  {
    "count": 28,
    "liveness": "0.09894"
  },
  {
    "count": 10,
    "liveness": "0.1652799999999998"
  },
  {
    "count": 8,
    "liveness": "0.2316199999999996"
  },
  {
    "count": 5,
    "liveness": "0.29796"
  },
  {

```

```

    "count": 4,
    "liveness": "0.3643"
  },
  {
    "count": 3,
    "liveness": "0.4306399999999997"
  },
  {
    "count": 0,
    "liveness": "0.49698"
  },
  {
    "count": 0,
    "liveness": "0.5633199999999999"
  },
  {
    "count": 2,
    "liveness": "0.6296599999999999"
  }
}],
"loudness_histogram": [
  {
    "count": 3,
    "loudness": "-15.429"
  },
  {
    "count": 3,
    "loudness": "-14.1574"
  },
  {
    "count": 4,
    "loudness": "-12.8858"
  },
  {
    "count": 5,
    "loudness": "-11.6142"
  },
  {
    "count": 10,
    "loudness": "-10.342600000000001"
  },
  {
    "count": 19,
    "loudness": "-9.071"
  },
  {
    "count": 17,
    "loudness": "-7.7994"
  },
  {
    "count": 19,
    "loudness": "-6.527799999999999"
  },
  {
    "count": 11,
    "loudness": "-5.2562"
  },
  {

```

```

    "count": 8,
    "loudness": "-3.984600000000004"
  }],
  "speechiness_histogram": [
    {
      "count": 73,
      "speechiness": "0.0241"
    },
    {
      "count": 11,
      "speechiness": "0.05048999999999999"
    },
    {
      "count": 6,
      "speechiness": "0.07687999999999999"
    },
    {
      "count": 1,
      "speechiness": "0.10326999999999999"
    },
    {
      "count": 3,
      "speechiness": "0.12966"
    },
    {
      "count": 2,
      "speechiness": "0.15605"
    },
    {
      "count": 0,
      "speechiness": "0.18244"
    },
    {
      "count": 1,
      "speechiness": "0.20883"
    },
    {
      "count": 1,
      "speechiness": "0.23521999999999998"
    },
    {
      "count": 1,
      "speechiness": "0.26160999999999995"
    }
  ],
  "tempo_histogram": [
    {
      "count": 3,
      "tempo": "67.427"
    },
    {
      "count": 7,
      "tempo": "79.5043"
    },
    {
      "count": 25,
      "tempo": "91.58160000000001"
    }
  ],

```

```

{
  "count": 19,
  "tempo": "103.6589"
},
{
  "count": 15,
  "tempo": "115.7362"
},
{
  "count": 12,
  "tempo": "127.81349999999999"
},
{
  "count": 9,
  "tempo": "139.8908"
},
{
  "count": 1,
  "tempo": "151.9681"
},
{
  "count": 5,
  "tempo": "164.04539999999997"
},
{
  "count": 3,
  "tempo": "176.12269999999998"
}],
"valence_histogram": [
  {
    "count": 4,
    "valence": "0.0382"
  },
  {
    "count": 7,
    "valence": "0.13088"
  },
  {
    "count": 10,
    "valence": "0.22355999999999998"
  },
  {
    "count": 8,
    "valence": "0.31624"
  },
  {
    "count": 13,
    "valence": "0.40892"
  },
  {
    "count": 13,
    "valence": "0.5015999999999999"
  },
  {
    "count": 11,
    "valence": "0.59428"
  },
},

```

```

    {
      "count": 13,
      "valence": "0.68696"
    },
    {
      "count": 7,
      "valence": "0.77964"
    },
    {
      "count": 13,
      "valence": "0.87232"
    }
  ]
},
"statistics": {
  "acousticness_statistics": {
    "avg": 0.19892206060606063,
    "max": 0.866,
    "min": 0.000133,
    "p25": 0.02905,
    "p5": 0.000756200000000002,
    "p50": 0.101,
    "p75": 0.333,
    "p95": 0.6431999999999999,
    "std": 0.2157161653289775
  },
  "danceability_statistics": {
    "avg": 0.6188585858585859,
    "max": 0.916,
    "min": 0.243,
    "p25": 0.5335000000000001,
    "p5": 0.39960000000000007,
    "p50": 0.628,
    "p75": 0.715,
    "p95": 0.8671,
    "std": 0.14117780250886655
  },
  "energy_statistics": {
    "avg": 0.6701818181818182,
    "max": 0.971,
    "min": 0.249,
    "p25": 0.505,
    "p5": 0.3438,
    "p50": 0.676,
    "p75": 0.86,
    "p95": 0.9315,
    "std": 0.2012648570906618
  },
  "instrumentalness_statistics": {
    "avg": 0.005983712323232324,
    "max": 0.22,
    "min": 0.0,
    "p25": 0.0,
    "p5": 0.0,
    "p50": 1.12e-05,
    "p75": 0.000347,
    "p95": 0.010719999999999964,
    "std": 0.03154661324601844
  }
}

```

```

},
"key_statistics": {
  "avg": 4.757575757575758,
  "max": 11,
  "min": 0,
  "p25": 1.5,
  "p5": 0.0,
  "p50": 5.0,
  "p75": 8.0,
  "p95": 11.0,
  "std": 3.4614497724348205
},
"liveness_statistics": {
  "avg": 0.16278080808080805,
  "max": 0.696,
  "min": 0.0326,
  "p25": 0.08449999999999999,
  "p5": 0.04965,
  "p50": 0.11,
  "p75": 0.212,
  "p95": 0.41839999999999999,
  "std": 0.12735933470586575
},
"loudness_statistics": {
  "avg": -7.6070505050505055,
  "max": -2.713,
  "min": -15.429,
  "p25": -9.018,
  "p5": -13.2286,
  "p50": -7.269,
  "p75": -5.588,
  "p95": -3.7478000000000002,
  "std": 2.733646620529817
},
"speechiness_statistics": {
  "avg": 0.05365454545454547,
  "max": 0.288,
  "min": 0.0241,
  "p25": 0.0292,
  "p5": 0.02559,
  "p50": 0.0347,
  "p75": 0.0524,
  "p95": 0.15369999999999998,
  "std": 0.04659951938176584
},
"tempo_statistics": {
  "avg": 116.33663636363637,
  "max": 188.2,
  "min": 67.427,
  "p25": 99.113,
  "p5": 80.28280000000001,
  "p50": 110.008,
  "p75": 130.67950000000002,
  "p95": 168.4878,
  "std": 25.3577230036211
},
"valence_statistics": {

```



```

    "avg": 0.5544666666666666,
    "max": 0.965,
    "min": 0.0382,
    "p25": 0.361,
    "p5": 0.1557,
    "p50": 0.561,
    "p75": 0.754,
    "p95": 0.9321,
    "std": 0.2452045956693588
  }
}
}

```

## GET /charts

Description: Get the tracks of an artist, appearing in playlists under monitoring, GET /charts.

E.g. <https://api.futurepulse.eu/charts>.

Response (parts of the json response have been removed for presentation purposes):

```

{
  "charts": [
    {
      "country": "GB",
      "genre_specific": "no",
      "id": "5acf4638b40ae300075c0ab3",
      "label": "official_charts",
      "link": "http://www.officialcharts.com/charts/end-of-year-artist-albums-chart/",
      "name": "End of Year Album Chart",
      "period": "year",
      "size": 100,
      "type": "album"
    },
    {
      "country": "GB",
      "genre_specific": "no",
      "id": "5acf4638b40ae300075c0ab2",
      "label": "official_charts",
      "link": "http://www.officialcharts.com/charts/asian-download-chart",
      "name": "Asian Music Chart",
      "period": "week",
      "size": 40,
      "type": "track"
    },
    {
      "country": "GB",
      "genre_specific": "no",
      "id": "5acf4638b40ae300075c0ab5",
      "label": "official_charts",
      "link": "http://www.officialcharts.com/charts/albums-downloads-chart",
      "name": "Official Album Downloads Chart",

```

```

    "period": "week",
    "size": 100,
    "type": "album"
  },
  {
    "country": "GB",
    "genre_specific": "yes",
    "id": "5acf4638b40ae300075c0ab9",
    "label": "official_charts",
    "link": "http://www.officialcharts.com/charts/americana-albums-chart",
    "name": "Official Americana Albums Chart",
    "period": "week",
    "size": 40,
    "type": "album"
  },
  {
    "country": "GB",
    "genre_specific": "no",
    "id": "5acf4638b40ae300075c0aba",
    "label": "official_charts",
    "link": "http://www.officialcharts.com/charts/audio-streaming-chart",
    "name": "Official Audio Streaming Chart",
    "period": "week",
    "size": 100,
    "type": "track"
  }
],
"count": 778
}

```

## GET /charts/id

Description: Get information of a specific chart by id

E.g. <https://api.futurepulse.eu/charts/5acf4638b40ae300075c0ab3>.

Response (full):

```

{
  "count": 2500,
  "country": "GB",
  "genre_specific": "no",
  "genres": [],
  "id": "5acf4638b40ae300075c0ab3",
  "label": "official_charts",
  "link": "http://www.officialcharts.com/charts/end-of-year-artist-albums-chart/",
  "name": "End of Year Album Chart",
  "period": "year",
  "since": "1995-01-01",
  "size": 100,
  "type": "album",
  "unique_tracks": 1314,

```

```
"until": "2018-12-31"
}
```

## GET /charts/countries

Description: Get a list of the countries of charts endpoints.

E.g. <https://api.futurepulse.eu/charts/countries>.

Response (parts of the json response have been removed for presentation purposes):

```
[
  "AD",
  "AE",
  "AR",
  "AT",
  "AU",
  ...,
  "US",
  "UY",
  "VE",
  "VN",
  "VZ",
  "WRL",
  "ZA",
  "ZW"
]
```

## GET /charts/{chart\_id}/artists/{spotify\_artist\_id}

Description: Get the appearances of an artist in a chart endpoint.

E.g.

<https://api.futurepulse.eu/charts/5acf4638b40ae300075c0ab3/artists/76ptJV8617638xrpeoUtzl>.

Response (full):

```
{
  "chart_id": "5acf4638b40ae300075c0ab3",
  "interpolate": true,
  "max_date": null,
  "min_date": null,
  "positions": [
    {
      "position": 66,
      "positions": [
        {
          "position": 66,
          "spotify_album_id": "2QGTxp9WrNfUry0aDF1toY",
          "spotify_track_id": null,

```

```

        "streams": null
      }
    ],
    "since": "2004-01-01T00:00:00",
    "size": 100,
    "tracks": 1
  }
],
"size": 100,
"spotify_artist_id": "76ptJV8617638xrpeoUtz1"
}

```

## GET /events

Description: Get events.

E.g. [https://api.futurepulse.eu/events?num\\_per\\_page=10&page=1](https://api.futurepulse.eu/events?num_per_page=10&page=1).

Response (full):

```

{
  "events": [
    {
      "artist_ids": [
        "4"],
      "description": null,
      "end_date": "2020-03-07T00:00:00+00:00",
      "end_date_timezone": 0,
      "id": 74,
      "name": "TEST",
      "start_date": "2020-03-02T00:00:00+00:00",
      "start_date_timezone": 0,
      "type": "marketing_push",
      "url": ""
    },
    {
      "artist_ids": [
        "4"],
      "description": null,
      "end_date": null,
      "end_date_timezone": 0,
      "id": 60,
      "name": "test",
      "start_date": "2019-11-15T21:00:00+00:00",
      "start_date_timezone": 0,
      "type": "marketing_push",
      "url": ""
    }
  ],
  "num_per_page": 10,
  "page": 1
}

```

## GET /events/id

Description: Get events by id.

E.g. <https://api.futurepulse.eu/events/60>.

Response (full):

```
{
  "artist_ids": [
    "4"],
  "description": null,
  "end_date": null,
  "end_date_timezone": 0,
  "id": 60,
  "name": "test",
  "start_date": "2019-11-15T21:00:00+00:00",
  "start_date_timezone": 0,
  "type": "marketing_push",
  "url": ""
}
```

## GET /events/artists/{artist\_id}/events

Description: Get events of an artist.

E.g. <https://api.futurepulse.eu/artists/47/events>.

Response (full):

```
{
  "events": [
    {
      "artist_ids": [
        "47"],
      "description": null,
      "end_date": null,
      "end_date_timezone": null,
      "id": 73,
      "name": "test",
      "start_date": "2019-11-28T00:00:00+00:00",
      "start_date_timezone": 0,
      "type": "tv_show",
      "url": ""
    },
    ...
    {
      "artist_ids": [
        "47"],
      "description": "test3",
      "end_date": null,
      "end_date_timezone": null,

```

```

    "id": 79,
    "name": "test3",
    "start_date": "2001-02-14T00:00:00+01:00",
    "start_date_timezone": 60,
    "type": "test3",
    "url": null
  }]
}

```

GET /sources/facebook\_page/<fb\_id>/facebook\_mentions

Description: Get the Facebook mentions from a certain page

E.g. [https://api.futurepulse.eu/sources/facebook\\_page/72688240730/facebook\\_mentions](https://api.futurepulse.eu/sources/facebook_page/72688240730/facebook_mentions).

Response (parts of the json response have been removed for presentation purposes):

```

{
  "metric": "facebook_mentions",
  "points": [
    {
      "change": -541,
      "change_zscore": -0.10453755007003555,
      "date": "2019-11-13",
      "interpolated": false,
      "log_value": 3.050379756261458,
      "normalized_value": 1.0466613183081774,
      "timestamp": 1573685100679,
      "value": 1122
    },
    {
      "change": -542,
      "change_zscore": -0.10473024041354531,
      "date": "2019-11-12",
      "interpolated": true,
      "log_value": 3.2211533219547053,
      "normalized_value": 1.5513349129647942,
      "timestamp": 1573516800000,
      "value": 1663
    },
    {
      "change": -542,
      "change_zscore": -0.10473024041354531,
      "date": "2019-11-11",
      "interpolated": true,
      "log_value": 3.343605508104172,
      "normalized_value": 2.0569413608462845,
      "timestamp": 1573430400000,
      "value": 2205
    }
  ],
  "source": "facebook_page",

```

```
"source_id": "72688240730"
}
```

GET /sources/facebook\_page/<fb\_id>/facebook\_fan\_count

Description: Get the Facebook fan counts from a certain page

E.g. [https://api.futurepulse.eu/sources/facebook\\_page/72688240730/facebook\\_fan\\_count](https://api.futurepulse.eu/sources/facebook_page/72688240730/facebook_fan_count).

Response (parts of the json response have been removed for presentation purposes):

```
{
  "metric": "facebook_fan_count",
  "points": [
    {
      "change": -4,
      "change_zscore": -0.2509744187258081,
      "date": "2019-11-13",
      "interpolated": false,
      "log_value": 5.303224145210022,
      "normalized_value": 99.35379644588045,
      "timestamp": 1573685100679,
      "value": 201012
    },
    {
      "change": -5,
      "change_zscore": -0.26681275284279804,
      "date": "2019-11-12",
      "interpolated": true,
      "log_value": 5.303232787241351,
      "normalized_value": 99.41254222352768,
      "timestamp": 1573516800000,
      "value": 201016
    },
    {
      "change": -5,
      "change_zscore": -0.26681275284279804,
      "date": "2019-11-11",
      "interpolated": true,
      "log_value": 5.303243589538688,
      "normalized_value": 99.48597444558672,
      "timestamp": 1573430400000,
      "value": 201021
    }
  ],
  "source": "facebook_page",
  "source_id": "72688240730"
}
```

GET /sources/spotify\_artist/<spotify\_artist\_id>/spotify\_artist\_followers

Description: Get the number of followers of the artist in Spotify.

E.g.

[https://api.futurepulse.eu/sources/spotify\\_artist/0muluUHym4N0wB62sD0MPo/spotify\\_artist\\_followers](https://api.futurepulse.eu/sources/spotify_artist/0muluUHym4N0wB62sD0MPo/spotify_artist_followers).

Response (parts of the json response have been removed for presentation purposes):

```
{
  "metric": "spotify_artist_followers",
  "points": [
    {
      "change": 4,
      "change_zscore": 0.11201225068667792,
      "date": "2019-11-14",
      "interpolated": false,
      "log_value": 4.145569297792989,
      "normalized_value": 100.0,
      "timestamp": 1573723800246,
      "value": 13981
    },
    {
      "change": 4,
      "change_zscore": 0.11201225068667792,
      "date": "2019-11-13",
      "interpolated": true,
      "log_value": 4.145445036137805,
      "normalized_value": 99.78757302177377,
      "timestamp": 1573603200000,
      "value": 13977
    },
    {
      "change": 4,
      "change_zscore": 0.11201225068667792,
      "date": "2019-11-12",
      "interpolated": true,
      "log_value": 4.145320738918325,
      "normalized_value": 99.57514604354753,
      "timestamp": 1573516800000,
      "value": 13973
    }
  ],
  "source": "spotify_artist",
  "source_id": "0muluUHym4N0wB62sD0MPo"
}
```

[GET /sources/spotify\\_artist/<spotify\\_artist\\_id>/spotify\\_artist\\_popularity](https://api.futurepulse.eu/sources/spotify_artist/<spotify_artist_id>/spotify_artist_popularity)

Description: Get the Spotify artist popularity.

E.g.

[https://api.futurepulse.eu/sources/spotify\\_artist/0muluUHym4N0wB62sD0MPo/spotify\\_artist\\_popularity](https://api.futurepulse.eu/sources/spotify_artist/0muluUHym4N0wB62sD0MPo/spotify_artist_popularity).

Response (parts of the json response have been removed for presentation purposes):



```

{
  "metric": "spotify_artist_popularity",
  "points": [
    {
      "change": 1,
      "change_zscore": 2.83754430301092,
      "date": "2019-11-14",
      "interpolated": false,
      "log_value": 1.6901960800285136,
      "normalized_value": 70.58823529411765,
      "timestamp": 1573723800246,
      "value": 48
    },
    {
      "change": 0,
      "change_zscore": -0.04809397123747323,
      "date": "2019-11-13",
      "interpolated": true,
      "log_value": 1.6812412373755872,
      "normalized_value": 64.70588235294117,
      "timestamp": 1573603200000,
      "value": 47
    },
    {
      "change": 0,
      "change_zscore": -0.04809397123747323,
      "date": "2019-11-12",
      "interpolated": true,
      "log_value": 1.6812412373755872,
      "normalized_value": 64.70588235294117,
      "timestamp": 1573516800000,
      "value": 47
    }
  ],
  "source": "spotify_artist",
  "source_id": "0muluUHym4N0wB62sD0MPo"
}

```

[GET /sources/youtube\\_channel/<youtube\\_channel\\_id>/youtube\\_channel\\_subscribers](#)

Description: Get the number of subscribers on artist's YouTube channel.

E.g.

[https://api.futurepulse.eu/sources/spotify\\_artist/0muluUHym4N0wB62sD0MPo/spotify\\_artist\\_popularity](https://api.futurepulse.eu/sources/spotify_artist/0muluUHym4N0wB62sD0MPo/spotify_artist_popularity).

Response (parts of the json response have been removed for presentation purposes):

```

{
  "metric": "youtube_channel_subscribers",
  "points": [
    {
      "change": 1,
      "change_zscore": 0.863662157929476,

```

```

    "date": "2019-11-14",
    "interpolated": false,
    "log_value": 2.6053050461411096,
    "normalized_value": 100.0,
    "timestamp": 1573713960274,
    "value": 402
  },
  {
    "change": 1,
    "change_zscore": 0.863662157929476,
    "date": "2019-11-13",
    "interpolated": true,
    "log_value": 2.60422605308447,
    "normalized_value": 99.49748743718592,
    "timestamp": 1573603200000,
    "value": 401
  },
  "source": "youtube_channel",
  "source_id": "UCqCc1Zmtvj_11I2zGfgWkSw"
}

```

[GET /sources/youtube\\_channel/<youtube\\_channel\\_id>/youtube\\_channel\\_views](#)

Description: Get the total number of view in artist's YouTube channel.

E.g.

[https://api.futurepulse.eu/sources/youtube\\_channel/UCqCc1Zmtvj\\_11I2zGfgWkSw/youtube\\_channel\\_views](https://api.futurepulse.eu/sources/youtube_channel/UCqCc1Zmtvj_11I2zGfgWkSw/youtube_channel_views).

Response (parts of the json response have been removed for presentation purposes):

```

{
  "metric": "youtube_channel_views",
  "points": [
    {
      "change": 135,
      "change_zscore": 0.4928390477916295,
      "date": "2019-11-14",
      "interpolated": false,
      "log_value": 4.7648707467768725,
      "normalized_value": 100.0,
      "timestamp": 1573713960274,
      "value": 58192
    },
    {
      "change": 134,
      "change_zscore": 0.48288475540012504,
      "date": "2019-11-13",
      "interpolated": true,
      "log_value": 4.763862071042256,
      "normalized_value": 99.71532800539822,
      "timestamp": 1573603200000,
      "value": 58057
    }
  ]
}

```

```
],
"source": "youtube_channel",
"source_id": "UCqCc1Zmtvj_11I2zGfgWkSw"
}
```

GET

[/sources/soundcloud\\_artist/<soundcloud\\_artist\\_id>/soundcloud\\_artist\\_followers\\_count](/sources/soundcloud_artist/<soundcloud_artist_id>/soundcloud_artist_followers_count)

Description: Get the number of followers the artist has in SoundCloud.

E.g.

[https://api.futurepulse.eu/sources/soundcloud\\_artist/207992365/soundcloud\\_artist\\_followers\\_count](https://api.futurepulse.eu/sources/soundcloud_artist/207992365/soundcloud_artist_followers_count).

Response (parts of the json response have been removed for presentation purposes):

```
{
  "metric": "soundcloud_artist_followers_count",
  "points": [
    {
      "change": 0,
      "change_zscore": -0.08873565094161139,
      "date": "2019-11-13",
      "interpolated": false,
      "log_value": 1.1760912590556813,
      "normalized_value": 100.0,
      "timestamp": 1573681140333,
      "value": 14
    },
    {
      "change": 0,
      "change_zscore": -0.08873565094161139,
      "date": "2019-11-12",
      "interpolated": true,
      "log_value": 1.1760912590556813,
      "normalized_value": 100.0,
      "timestamp": 1573516800000,
      "value": 14
    }
  ],
  "source": "soundcloud_artist",
  "source_id": "207992365"
}
```

GET [/sources/lastfm\\_artist/<lastfm\\_user\\_name>/lastfm\\_artist\\_listener\\_count](/sources/lastfm_artist/<lastfm_user_name>/lastfm_artist_listener_count)

Description: Get the number of listeners the artist has in Last.fm.

E.g.

[https://api.futurepulse.eu/sources/lastfm\\_artist/Malena%20Ernman/lastfm\\_artist\\_listener\\_count](https://api.futurepulse.eu/sources/lastfm_artist/Malena%20Ernman/lastfm_artist_listener_count).

Response (parts of the json response have been removed for presentation purposes):

```
{
  "metric": "lastfm_artist_listener_count",
  "points": [
    {
      "change": 1,
      "change_zscore": -0.49691250665426606,
      "date": "2019-11-14",
      "interpolated": false,
      "log_value": 4.470381086156399,
      "normalized_value": 100.0,
      "timestamp": 1573770249762,
      "value": 29537
    },
    {
      "change": 3,
      "change_zscore": -0.0400735892463118,
      "date": "2019-11-13",
      "interpolated": false,
      "log_value": 4.470366383000007,
      "normalized_value": 99.93861264579496,
      "timestamp": 1573683849276,
      "value": 29536
    }
  ],
  "source": "lastfm_artist",
  "source_id": "Malena Ernman"
}
```

[GET /sources/lastfm\\_artist/<lastfm\\_user\\_name>/lastfm\\_artist\\_play\\_count](#)

Description: Get the number of total plays the artist has in Last.fm.

E.g. [https://api.futurepulse.eu/sources/lastfm\\_artist/Malena%20Ernman/lastfm\\_artist\\_play\\_count](https://api.futurepulse.eu/sources/lastfm_artist/Malena%20Ernman/lastfm_artist_play_count)

Response (parts of the json response have been removed for presentation purposes):

```
{
  "metric": "lastfm_artist_play_count",
  "points": [
    {
      "change": 19,
      "change_zscore": -0.2488374107965502,
      "date": "2019-11-14",
      "interpolated": false,
      "log_value": 5.616410391091141,
      "normalized_value": 100.0,
      "timestamp": 1573770249762,
      "value": 413437
    },
    {
      "change": 22,
      "change_zscore": -0.16948374829843774,

```

```

    "date": "2019-11-13",
    "interpolated": false,
    "log_value": 5.616390432149855,
    "normalized_value": 99.86962190352021,
    "timestamp": 1573683849276,
    "value": 413418
  }
],
"source": "lastfm_artist",
"source_id": "Malena Ernman"
}

```

GET /sources/deezer\_artist/<deezer\_artist\_id>/deezer\_artist\_fans

Description: Get the artist fans on Deezer.

E.g. [https://api.futurepulse.eu/sources/deezer\\_artist/927/deezer\\_artist\\_fans](https://api.futurepulse.eu/sources/deezer_artist/927/deezer_artist_fans).

Response (parts of the json response have been removed for presentation purposes):

```

{
  "metric": "deezer_artist_fans",
  "points": [
    {
      "change": 720,
      "change_zscore": 2.955329937179468,
      "date": "2019-11-15",
      "interpolated": false,
      "log_value": 6.522446972291714,
      "normalized_value": 100.0,
      "timestamp": 1573783981169,
      "value": 3330020
    },
    {
      "change": 741,
      "change_zscore": 3.0897520196358594,
      "date": "2019-11-14",
      "interpolated": false,
      "log_value": 6.522353061221431,
      "normalized_value": 99.4838154640284,
      "timestamp": 1573697581181,
      "value": 3329300
    }
  ],
  "source": "deezer_artist",
  "source_id": "927"
}

```

GET /spotify\_artists/{spotify\_artist\_id}/streams

Description: Get an analysis about a Spotify artist.

E.g. [https://api.futurepulse.eu/spotify\\_artists/76ptJV8617638xrpeoUtzl/streams](https://api.futurepulse.eu/spotify_artists/76ptJV8617638xrpeoUtzl/streams).

Response (parts of the json response have been removed for presentation purposes):

```
{
  "age_groups": [
    {
      "age_group": "0-17",
      "percentage": 0.04109700168940746,
      "streams": 3911958
    },
    {
      "age_group": "18-22",
      "percentage": 0.23348122319605233,
      "streams": 22224705
    },
    {
      "age_group": "23-27",
      "percentage": 0.32286316093712425,
      "streams": 30732829
    },
    {
      "age_group": "28-34",
      "percentage": 0.22821494097117123,
      "streams": 21723416
    },
    {
      "age_group": "35-44",
      "percentage": 0.10406149891955327,
      "streams": 9905448
    },
    {
      "age_group": "45-59",
      "percentage": 0.055677086465279954,
      "streams": 5299813
    },
    {
      "age_group": "60-150",
      "percentage": 0.011399067054428185,
      "streams": 1085059
    },
    {
      "age_group": "unknown",
      "percentage": 0.0032060207669833397,
      "streams": 305176
    }
  ],
  "artist_id": "76ptJV8617638xrpeoUtz1",
  "average_streams": 54674,
  "countries": [
    {
      "countries": "AD",
      "percentage": 5.242617949316275e-05,
      "streams": 4976
    },
  ],
}
```

```

{
  "countries": "AE",
  "percentage": 0.00016543325369807908,
  "streams": 15702
},
{
  "countries": "AR",
  "percentage": 0.031290074261851826,
  "streams": 2969879
},
{
  "countries": "AT",
  "percentage": 0.007057189586761598,
  "streams": 669829
},
{
  "countries": "AU",
  "percentage": 0.011004598542948357,
  "streams": 1044495
}
],
"days_distribution": [
  {
    "average_streams": 53598.78313253012,
    "average_streams_std": 46985.62075144641,
    "day": "Monday",
    "max_streams": 181840,
    "total_streams": 13346097,
    "total_streams_percentage": 0.14020717271402092
  },
  {
    "average_streams": 56099.709677419356,
    "average_streams_std": 49170.78194477767,
    "day": "Tuesday",
    "max_streams": 180424,
    "total_streams": 13912728,
    "total_streams_percentage": 0.146159904099243
  },
  {
    "average_streams": 57036.02419354839,
    "average_streams_std": 49675.68731410212,
    "day": "Wednesday",
    "max_streams": 175750,
    "total_streams": 14144934,
    "total_streams_percentage": 0.14859933989438462
  },
  {
    "average_streams": 57865.329317269076,
    "average_streams_std": 50329.12714969917,
    "day": "Thursday",
    "max_streams": 178018,
    "total_streams": 14408467,
    "total_streams_percentage": 0.1513678809028041
  },
  {

```

```

{
  "average_streams": 61745.46184738956,
  "average_streams_std": 55196.03835850399,
  "day": "Friday",
  "max_streams": 262661,
  "total_streams": 15374620,
  "total_streams_percentage": 0.16151778319552454
},
{
  "average_streams": 51944.899598393575,
  "average_streams_std": 45180.34799100504,
  "day": "Saturday",
  "max_streams": 182568,
  "total_streams": 12934280,
  "total_streams_percentage": 0.13588083691370642
},
{
  "average_streams": 44446.899598393575,
  "average_streams_std": 38070.561950341886,
  "day": "Sunday",
  "max_streams": 145977,
  "total_streams": 11067278,
  "total_streams_percentage": 0.11626708228031642
}
],
"gender": [
  {
    "gender": "female",
    "percentage": 0.4027627041629987,
    "streams": 38338339
  },
  {
    "gender": "male",
    "percentage": 0.5843525751308951,
    "streams": 55623589
  },
  {
    "gender": "neutral",
    "percentage": 0.005230279940401144,
    "streams": 497862
  },
  {
    "gender": "unknown",
    "percentage": 0.007654440765705033,
    "streams": 728614
  }
],
"months_distribution": [
  {
    "average_streams": 43775.7935483871,
    "average_streams_std": 43283.33438793762,
    "max_streams": 115424,
    "month": "January",
    "total_streams": 6785248,
  }
]

```



```

    "total_streams_percentage": 0.07128229610825285
  },
  {
    "average_streams": 44565.936170212764,
    "average_streams_std": 43067.85650315172,
    "max_streams": 116416,
    "month": "February",
    "total_streams": 6283797,
    "total_streams_percentage": 0.06601431199539809
  }
],
"products": [
  {
    "percentage": 0.2650632260451728,
    "product": "free",
    "streams": 25204151
  },
  {
    "percentage": 0.7349367739548273,
    "product": "premium",
    "streams": 69883166
  }
],
"total_streams": 95188404
}

```

## GET /lexicology

Description: This endpoint is related to the /musimotion endpoints that provide predicted tags for tracks.

Get a list of descriptions that relate to categories of tags like genre, situation, moods, rhythmic\_mood. This endpoint also provides a translation of tag ids to tag names (see example below).

E.g. [https://api.futurepulse.eu?q=Happiness&category\\_name=mood](https://api.futurepulse.eu?q=Happiness&category_name=mood).

Response:

```

{
  "timestamp":1604121727588,
  "data":[
    {
      "id":"f05f72ad-e056-cea2-3bd2-e109efa0c02b",
      "category_name":"mood",
      "name":"Happiness",
      "description":"Meaning all moods related to : POSITIVITY / FREEDOM
\nMoments : Festivity / Easy living
\nMoods providing : Unworried feelings / Innocent amusement
\nMusical example :
Teen pop"
    }
  ],
  "offset":0,

```

```
"limit":-1,  
"size":1  
}
```

## GET /dashboard

Description: The service returns aggregated information regarding certain types shown on FuturePulse dashboard.

E.g. <https://api.futurepulse.eu/dashboard?type=genres>.

Response:

```
{  
  "country": "global",  
  "dashboard_genres": [  
  ],  
  "date": "2020-10-18",  
  "genres": [  
  ],  
  "growth_metric": "pagerank_095_1_month_dif",  
  "growth_span": "1_month",  
  "metrics":  
  "raw_count,prob,eukl,pagerank_095,pagerank_095_1_month_dif,pagerank_095_1_month_relative_dif",  
  "msg": "",  
  "num_per_page": 20,  
  "page": 1,  
  "rank_by": "popularity",  
  "relative_growth_metric": "pagerank_095_1_month_relative_dif",  
  "source": "future_pulse",  
  "statistics": {  
    "max_scores_all_countries_current_date": {  
      "birank_05": 0.07715788946600725,  
      "birank_075": 0.04949261436777887,  
      "birank_09": 0.028843509641404114,  
      "birank_095": 0.02580439777994633,  
      "birank_099": 0.017847190238375735,  
      "eukl": 0.8672123007280113,  
      "pagerank_05": 0.16041489727941355,  
      "pagerank_075": 0.12243071129334834,  
      "pagerank_09": 0.08406984871692941,  
      "pagerank_095": 0.059192448730495936,  
      "pagerank_099": 0.02045644607857667,  
      "prob": 0.29353272784215473,  
      "raw_count": 4161.0  
    },  
    "max_scores_all_countries_overall": {  
      "birank_05": 0.08902555906392205,  
      "birank_075": 0.05701730332976008,  
      "birank_09": 0.029577845507835692,  
      "birank_095": 0.02637378748199812,  
      "birank_099": 0.018492217641108172,  
    }  
  }  
}
```

```
"eukl": 0.9042259999354403,
"pagerank_05": 0.17527519687192142,
"pagerank_075": 0.12577474851273657,
"pagerank_09": 0.08697695201199034,
"pagerank_095": 0.06125028615794812,
"pagerank_099": 0.02157835895931358,
"prob": 0.33880248833592536,
"raw_count": 4830.0
},
"max_scores_country_specific_overall": {
  "birank_05": 0.034566122747570864,
  "birank_075": 0.021936841521876988,
  "birank_09": 0.011111379505179493,
  "birank_095": 0.007716867195316601,
  "birank_099": 0.002445912389429213,
  "eukl": 0.6883158497853203,
  "pagerank_05": 0.05808797438255522,
  "pagerank_075": 0.03441865981646764,
  "pagerank_09": 0.01883317221791793,
  "pagerank_095": 0.012408022159698126,
  "pagerank_099": 0.0056866299379599375,
  "prob": 0.13000817661488143,
  "raw_count": 3469.0
}
},
"uid": "test@test.gr"
}
```