

Individual Project Server

For Hacktiv8 Phase 2 Final Project.

A custom API for a music app.

Genres

GET Get genres

{{BASE_URL}}/genres

Get all song genres registered in the database.

Sub-genres will be collected under the same parent.

GET Get genre by ID and its sub-genres

{{BASE_URL}}/genres/:id

Get a genre by its id as registered on the database, including its sub-genres.

PATH VARIABLES

id

Genre Id

GET Get songs by Genre

{{BASE_URL}}/genres/:id/songs

Get songs under the genre with the specified Id.

PARAMS

limit	Number of records to retrieve per API call. Max 100.
--------------	------------------------------------------------------

offset	Number of entries to offset
---------------	-----------------------------

PATH VARIABLES

id	Genre Id
-----------	----------

Songs

GET Get songs

{{BASE_URL}}/songs

Get songs stored in the database, starting from the most recently added.

PARAMS

limit	Number of records to retrieve per API call. Max 100.
--------------	------------------------------------------------------

offset	Number of entries to offset.
---------------	------------------------------

GET Get song by ID

{{BASE_URL}}/songs/:id

Get song by ID, including associated artists, genres, albums, and play links.

PATH VARIABLES

id	Song Id
-----------	---------

Artists

GET Get artists

{{BASE_URL}}/artists

Get artists in the database, starting from the most recently added.

GET Get artist by ID

{{BASE_URL}}/artists/:id

Get artist information by specified ID.

PATH VARIABLES

id	Artist Id
----	-----------

GET Get artist songs by artist ID

{{BASE_URL}}/artists/:id/songs

Get artist songs by specified ID, including all artists contributing to each song and all play links for each song.

PARAMS

limit	Number of records to retrieve per API call. Max 100.
-------	------------------------------------------------------

offset	Number of entries to offset.
--------	------------------------------

PATH VARIABLES

id	Artist Id
----	-----------

GET Get artist albums by artist ID

{{BASE_URL}}/artists/:id/albums

Get artist's albums by specified ID.

PARAMS

limit	Number of records to retrieve per API call. Max 100.
offset	Number of entries to offset.

PATH VARIABLES

id	Artist Id
-----------	-----------

Albums

GET Get albums

{{BASE_URL}}/albums

Get albums stored in the database, starting from the most recently added.

GET Get albums by ID

{{BASE_URL}}/albums/:id

Get album by specified ID, including each song on the tracklist.

PATH VARIABLES

id	Album ID
-----------	----------

MAIN ENTITY CUD METHODS

POST Add Song

{{BASE_URL}}/songs

HEADERS

access_token

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6ImwidXNlcm5hbWUiOiJKb2h1IERvZSIsInJvbmUiOiJTdGFmZiIsImhdCI6MTY5MzIzOTI0NSwiZXhwIjoxNjkzODQ0MDQ1fQ.8tihmIb3T-fJreO6polGLq3Okj1Eso1fGdK2RFryhcE

Body raw (json)

json

```
{
  "name": "foo",
  "aliases": [],
  "releaseDate": "2020-01-01T00:00:00.000Z",
  "songType": "Original",
  "parentId": "",
  "artists": [
    {
      "id": 1,
      "role": "Singer"
    }
  ],
  "playLinks": [
    {
      "songURL": "https://www.youtube.com/watch?v=RVPXMlIoZxk"
    }
  ]
}
```

PUT Update Song

{{BASE_URL}}/songs/:id

PATH VARIABLES

id

27

Song Id

Body raw (json)

json

```
{
  "name": "foofoobar",
  "aliases": [],
  "releaseDate": "2020-01-01T00:00:00.000Z",
  "songType": "Original",
  "parentId": null
}
```

DELETE Delete Song

{{BASE_URL}}/songs/:id

PATH VARIABLES

id	26
	Song Id

POST Add Artist

{{BASE_URL}}/artists

Body raw (json)

json

```
{
  "name": "foo",
  "aliases": [],
  "imageUrl": null,
  "description": "Original artist",
  "artistLinks": [
    {
      "webURL": "https://en.wikipedia.org/wiki/Main_Page",
      "description": "Wikipedia"
    }
  ]
}
```

```
{
```

PUT Update Artist

{{BASE_URL}}/artists/:id

PATH VARIABLES

id	8
	Artist Id

Body raw (json)

```
json

{
  "name": "foo22",
  "aliases": [],
  "imageUrl": null,
  "description": "An artist."
}
```

DELETE Delete Artist

{{BASE_URL}}/artists/:id

PATH VARIABLES

id	8
	Artist Id

MODULAR ENTITY CUD

POST Add Genre to Song

{{BASE_URL}}/songs/:id/genres/:genreId

PATH VARIABLES

id	1
	Song Id
genreId	2
	Genre Id

DELETE Remove Genre from Song

{{BASE_URL}}/songs/:id/genres/:genreId

PATH VARIABLES

id	1
	Song Id
genreId	2
	Genre Id

POST Add Playlink to Song

{{BASE_URL}}/songs/:id/playlinks

PATH VARIABLES

id	100
	Song Id

Body raw (json)

json


```
{
  "songURL": "https://www.youtube.com/watch?v=5tUoqIcvG34",

  "isInactive": null
}
```

DELETE Delete Playlink from Song

{{BASE_URL}}/songs/:id/playlinks/:playLinkId

PATH VARIABLES

id	8
	Song Id
playLinkId	100
	Play Link id

Body raw (json)

```
json

{
  "songURL": "https://www.youtube.com/watch?v=5tUoqIcvG34",
  "isInactive": null
}
```

POST Add Artist to Song

{{BASE_URL}}/songs/:id/artists/:artistId

PATH VARIABLES

id	1
	Song Id
artistId	700
	Artist Id

Body raw (json)

```
json

{
  "role": "Composer"
}
```

DELETE Delete Artist from Song

{{BASE_URL}}/songs/:id/artists/:artistId

PATH VARIABLES

id	20
	Song Id
artistId	100
	Artist Id

POST Add Artist Link

{{BASE_URL}}/artists/:id/links

PATH VARIABLES

id	1
	Artist Id

Body raw (json)

```
json

{
  "webURL": "https://www.youtube.com",
  ...
}
```

```
      "description": "YouTube",
    },
    "isInactive": false
  }
}
```

DELETE Remove Artist Link

{{BASE_URL}}/artists/:id/links/:linkId

PATH VARIABLES

id	1
	Artist Id
linkId	16
	Link Id

PUT Edit Play Link Status

{{BASE_URL}}/songs/:id/playlinks/:playLinkId

PATH VARIABLES

id	27
	Song Id
playLinkId	45
	Play Link Id

Body raw (json)

```
json

{
  "isInactive": true
}
```

PUT Edit Song Artist Role

{{BASE_URL}}/songs/:id/artists/:artistId

PATH VARIABLES

id	8
	Song Id
artistId	3
	Artist Id

Body raw (json)

```
json

{
  "role": "foo"
}
```

GET Search songs

{{BASE_URL}}/search/songs?title=

PARAMS

title	Substring for which the song titles are to contain.
--------------	-----------------------------------------------------

GET Search albums

{{BASE_URL}}/search/albums?title=

PARAMS

title	Substring for which the album titles are to contain.
--------------	------------------------------------------------------

GET Search artists

{{BASE_URL}}/search/artists?title=

PARAMS

title	Substring for which the artist's names are to contain.
-------	--------------------------------------------------------

POST Login

{{BASE_URL}}/login

Login user.

Body urlencoded

username	John Doe
----------	----------

password	admin
----------	-------

POST Register

{{BASE_URL}}/register

Register new user.

Body urlencoded

email	johnny.doe@mail.com
-------	---------------------

username	Johnny Doe
----------	------------

password	password
----------	----------

POST Get Recent Videos By Artist

POST Get recent videos by Artist

{{BASE_URL}}/youtube/artists

Get recent uploads by the artist by calling the YouTube API.

Body urlencoded

channelUrl <https://www.youtube.com/channel/UCUXfRsEIJ9xO1DT7TbEWksw>

POST Get Popular Songs from VocaDB

{{BASE_URL}}/vocadb/artists

Get top rated VOCALOID songs by calling the VOCALOID database (<https://vocadb.net>)

Body urlencoded

vocadbUrl <https://vocadb.net/Ar/53>