



# MusicAPI

## AUTHORIZATION

Bearer Token

---

### Token

<token>

---

# Admin Music Call

## AUTHORIZATION

Bearer Token

---

### Token

<token>

---

## POST Admin Add Music

http://localhost:5000/music/

if user is Admin i.e user.isAdmin==true  
then user can add music in the library



## Bearer Token

This request is using Bearer Token from folder [Admin Music Call](#)

## BODY raw

```
{
  "music_id": "0002",
  "music_link": "https://www.youtube.com/watch?v=bafqYXSG094",
  "title": "OneMusicCall",
  "singer": "TwoSingerCall",
  "album": "ThreeAlbumCall",
  "release_date": "2022-10-17T03:25:31.642Z"
}
```

## Example Request

[Admin Add Music](#)

```
var request = require('request');
var options = {
  'method': 'POST',
  'url': 'http://localhost:5000/music/',
  'headers': {
  },
  body: '{\r\n      "music_id": "0002",\r\n      "music_link": "https://www.youtube.com/watch?v=bafqYXSG094",\r\n      "title": "OneMusicC'
  .
```

[View More](#)

## DEL Admin Delete A specific Music Id



```
http://localhost:5000/music/delete/634ccb2b4718a42cdc453ae0
```

StartFragment

if user is Admin i.e user.isAdmin==true

then user can delete a music in the library using music's objectID i.e music.\_id not music\_id

EndFragment

## AUTHORIZATION

Bearer Token

---

This request is using Bearer Token from folder [Admin Music Call](#)

Example Request

Admin Delete A specific Music Id

```
var request = require('request');
var options = {
  'method': 'DELETE',
  'url': 'http://localhost:5000/music/delete/634ccb2b4718a42cdc453ae0',
  'headers': {
  }
};
request(options, function (error, response) {
  if (error) throw new Error(error);
});
```

[View More](#)



## AUTHORIZATION

Bearer Token

---

### Token

<token>

---

## GET Get Music List

http://localhost:5000/music/

StartFragment

Can view a list of songs available in the library

EndFragment

## AUTHORIZATION

Bearer Token

---

This request is using Bearer Token from folder [Normal/Admin User Calls](#)

### Example Request

```
var request = require('request');
var options = {
  'method': 'GET',
  'url': 'http://localhost:5000/music/'
};
```

Get Music List



```
request(options, function (error, response) {
```

[View More](#)

## GET Get Specific Music Meta Data using music ObjectID

```
http://localhost:5000/music/634d25dd9d17066445c7e56b
```

StartFragment

Admin/Normal user can get meta data of a music using music's ObjectID

## AUTHORIZATION

Bearer Token

This request is using Bearer Token from folder [Normal/Admin User Calls](#)

Example Request

Get Specific Music Meta Data using music ObjectID

```
var request = require('request');
var options = {
  'method': 'GET',
  'url': 'http://localhost:5000/music/634d25dd9d17066445c7e56b',
  'headers': {
  }
};
request(options, function (error, response) {
```

## PUT Like A Music

```
http://localhost:5000/user/like/634d25dd9d17066445c7e56b
```

StartFragment

Admin/Normal user can like a music using music's ObjectID

EndFragment

## AUTHORIZATION

Bearer Token

### Token

<token>

Example Request

Like A Music

```
var request = require('request');
var options = {
  'method': 'PUT',
  'url': 'http://localhost:5000/user/like/634d25dd9d17066445c7e56b',
  'headers': {
  }
};
request(options, function (error, response) {
  if (error) throw new Error(error);
});
```

[View More](#)



## PUT Dislike A music

```
http://localhost:5000/user/dislike/634d25dd9d17066445c7e56b
```

StartFragment

Admin/Normal user can dislike a music using music's ObjectID

EndFragment

## AUTHORIZATION

Bearer Token

### Token

<token>

Example Request

Dislike A music

```
var request = require('request');
var options = {
  'method': 'PUT',
  'url': 'http://localhost:5000/user/dislike/634d25dd9d17066445c7e56b',
  'headers': {
  }
};
request(options, function (error, response) {
  if (error) throw new Error(error);
});
```

[View More](#)



http://localhost:5000/user/likes

StartFragment

Admin/Normal user can get list of liked musics

EndFragme

## AUTHORIZATION

Bearer Token

### Token

<token>

Example Request

Get All Liked Musics

```
var request = require('request');
var options = {
  'method': 'GET',
  'url': 'http://localhost:5000/user/likes',
  'headers': {
  }
};
request(options, function (error, response) {
  if (error) throw new Error(error);
});
```

View More





http://localhost:5000/

getting started

## AUTHORIZATION

### Bearer Token

---

This request is using Bearer Token from collection [MusicAPI](#)

#### Example Request

Start

```
var request = require('request');
var options = {
  'method': 'GET',
  'url': 'http://localhost:5000/',
  'headers': {
  }
};
request(options, function (error, response) {
  if (error) throw new Error(error);
});
```

[View More](#)

## POST Login

http://localhost:5000/user/login



## AUTHORIZATION

Bearer Token

This request is using Bearer Token from collection [MusicAPI](#)

## BODY raw

```
{
  "username": "test0002",
  "password": "pass0001"
}
```

## Example Request

Login

```
var request = require('request');
var options = {
  'method': 'POST',
  'url': 'http://localhost:5000/user/login',
  'headers': {
  },
  body: '{\r\n  "username": "test0002",\r\n  "password": "pass0001"\r\n}'
}
```

[View More](#)

## POST Signup

http://localhost:5000/user/signup



by default isAdmin is false and hence not necessary to create User with Normal access

## AUTHORIZATION

Bearer Token

This request is using Bearer Token from collection [MusicAPI](#)

## BODY raw

```
{
  "name" : "testcalss",
  "username": "Neeraj3073",
  "password": "pass0001",
  "isAdmin": true
}
```

Example Request

Signup

```
var request = require('request');
var options = {
  'method': 'POST',
  'url': 'http://localhost:5000/user/signup',
  'headers': {
  },
  body: '{\r\n    "name" : "testcalss",\r\n    "username": "Neeraj3073",\r\n    "password": "pass0001",\r\n    "isAdmin": true\r\n}'
```