



API de Música: SONGIFY

BALSALOBRE HERRÁEZ, CRISTINA

CALVO TORRES, MARÍA

HERRERA DE MIGUEL, GEMA PATRICIA

MERCHÁN CASTILLO, RAMÓN

Índice

- Temática
- Tecnologías utilizadas
- Peticiones a otro servicio web
 - Formato XML
 - Formato JSON
- Interfaz REST que ofrece operaciones CRUD
 - Formato XML
 - Formato JSON
- Colecciones de la base de datos
- Paginación y filtro de los datos almacenados
- Ejemplo de uso de cliente



Temática (I)

API de música:

/artist

/artist/{id}

/artist/{id}/release

/artist/{id}/release/{id}

Datos recuperados desde MongoDB.



Temática (II)

API de música:

`/artist/{id}/release/{id}/recordings`

`/artist/{id}/release/{id}/recordings/{id}`

`/artist/{id}/recordings`

`/artist/{id}/recordings/{id}`

Peticiones a la API MusicBrainz.

Tecnologías utilizadas



Peticiones a otro servicio web

Formato JSON

```
{
  "type-id": "e431f5f6-b5d2-343d-8b36-72607fffb74b",
  "life-span": {
    "ended": false,
    "end": null,
    "begin": "1996-09"
  },
  "area": {
    "type": null,
    "sort-name": "United Kingdom",
    "name": "United Kingdom",
    "iso-3166-1-codes": [
      "GB"
    ],
    "type-id": null,
    "id": "8a754a16-0027-3a29-b6d7-2b40ea0481ed",
    "disambiguation": ""
  }
}
```

Formato XML

```
<?xml version="1.0" encoding="UTF-8"?>
<metadata xmlns="http://musicbrainz.org/ns/mmd-2.0#">
  <artist id="cc197bad-dc9c-440d-a5b5-d52ba2e14234" type="Group" type-id="e431f5f6-b5d2-343d-8b36-72607fffb74b">
    <name>Coldplay</name>
    <sort-name>Coldplay</sort-name>
    <isni-list>
      <isni>000000011551394X</isni>
    </isni-list>
    <country>GB</country>
    <area id="8a754a16-0027-3a29-b6d7-2b40ea0481ed">
      <name>United Kingdom</name>
      <sort-name>United Kingdom</sort-name>
      <iso-3166-1-code-list>
        <iso-3166-1-code>GB</iso-3166-1-code>
      </iso-3166-1-code-list>
    </area>
    <begin-area id="f03d09b3-39dc-4083-afd6-159e3f0d462f">
      <name>London</name>
    </begin-area>
  </artist>
</metadata>
```

Interfaz REST que ofrece operaciones CRUD (I)

```
paths:
  /artist:
    get:
      tags:
        - artist
      parameters:
        - $ref: "#/components/parameters/limit"
        - $ref: "#/components/parameters/next"
        - $ref: "#/components/parameters/gender"
      summary: GET all the artists of the API
      description: Retreives all the information of the available artists
```

```
/artist/{id}:
  parameters:
    - $ref: "#/components/parameters/id"
  get:
    tags:
      - artist
    summary: GET one artist of the API
    description: Retreives all the information of the specified artist
```

```
post:
  tags:
    - artist
  summary: POST a new artist
  description: Creates a new artist based on the information provided by the user
  requestBody:
    required: true
    content:
      application/json:
        schema:
          $ref: "#/components/schemas/artist"
```

```
put:
  tags:
    - artist
  summary: PUT one artist of the API
  description: Updates all or part of the information of the specified artist
```

```
delete:
  tags:
    - artist
  summary: DELETE one artist of the API and all their releases
  description: Deletes all the information of the specified artist
```

Interfaz REST que ofrece operaciones CRUD (II)

Formato JSON

```
{
  "_id": "6475e94e2814b79e36bab29a",
  "id": "a4ac255f-9775-4c16-8642-7f51502e45dd",
  "type": "Person",
  "name": "Stephen King",
  "gender": "male",
  "area": "United States",
  "begin": "1947-09-21",
  "url_release": "localhost:3000/api/v1/artist/6475e94e2814b79e36bab29a/release"
},
```

Formato XML

```
<?xml version="1.0"?>
<response>
  <results>
    <result>
      <_id>6475e9692814b79e36babb1f</_id>
      <id>03bff10d-3d38-4b47-b19c-bb8c45a6c551</id>
      <title>Bach: Magnificat / Bruckner: Te Deum</title>
      <date>1970</date>
      <id_artist>24f1766e-9635-4d58-a4d4-9413f9f98a4c</id_artist>
    </result>
  </results>
</response>
```


Colecciones de la base de datos

songify.artist

1k 1
DOCUMENTS INDEXES

```
_id: ObjectId('647517df16f1da255775b500')  
id: "24f1766e-9635-4d58-a4d4-9413f9f98a4c"  
type: "Person"  
name: "Johann Sebastian Bach"  
gender: "male"  
area: "Germany"  
begin: "1685-03-21"  
end: "1750-07-28"
```

songify.release

1.5k 1
DOCUMENTS INDEXES

```
_id: ObjectId('647517f616f1da255775b8ea')  
id: "03082446-54ca-4f5f-a0a8-a19fa918be9b"  
title: "Harry Potter ja liekehtivä pikari"  
date: "2013"  
country: "FI"  
language: "fin"  
id_artist: "569c0d90-28dd-413b-83e4-aaa7c27e667b"
```

Paginación de los datos (parámetro next)

```
{
  "_id": "6475e94e2814b79e36bab299",
  "id": "f1bedf1f-4445-4651-9c35-f4a3f3860a13",
  "type": null,
  "name": "sfgfsg",
  "gender": "male",
  "area": "Italy",
  "begin": "1813-10-10",
  "end": "1901-01-27",
  "url_release": "localhost:3000/api/v1/artist/6475e94e2814b79e36bab299/release"
},
"next": "6475e94e2814b79e36bab299"
```

Filtrado de los datos (parámetro gender)

GET ⌵ Send ⌵

```
"_id": "6475e94e2814b79e36bab28b",
"id": "27870d47-bb98-42d1-bf2b-c7e972e6befc",
"type": "Person",
"name": "George Frideric Handel",
"gender": "male",
"area": "United Kingdom",
"begin": "1685-02-23",
"end": "1759-04-14",
"url_release": "localhost:3000/api/v1/artist/6475e94e2814b79e36bab28b/release"

"_id": "6475e94e2814b79e36bab28c",
"id": "ad79836d-9849-44df-8789-180bbc823f3c",
"type": "Person",
"name": "Antonio Vivaldi",
"gender": "male",
"area": "Italy",
"begin": "1678-03-04",
"end": "1741-07-28",
"url_release": "localhost:3000/api/v1/artist/6475e94e2814b79e36bab28c/release"
```

Ejemplo de uso de cliente (I)

```
Enter the URL: http://localhost:3000/api/v1/artist/6475e94e2814b79e36bab28a
Response (JSON): {
  _id: '6475e94e2814b79e36bab28a',
  id: '01809552-4f87-45b0-afff-2c6f0730a3be',
  type: 'Person',
  name: 'Elvis Presley',
  gender: 'male',
  area: 'United States',
  begin: '1935-01-08',
  end: '1977-08-16',
  url_release: 'localhost:3000/api/v1/artist/6475e94e2814b79e36bab28a/release'
}
```

Ejemplo de uso de cliente (II)

```
Enter the HTTP method (GET, DELETE, PUT, or POST), or type "exit" to quit: POST
Enter the URL: http://localhost:3000/api/v1/artist
Enter the JSON payload: {"id": "1", "type": "Person", "name": "Bzrp", "gender": "male", "area": "Argentina", "begin": "1998-08-29"}
Response (JSON): { acknowledged: true, insertedId: '64792fd32245ccc9188d21c5' }
```