

(Diagrama bazei de date)

The top row shows the Spotify web application with the 'Songs list' section displaying three items: 'The trooper 1983', 'Wasted years 1988', and 'The writing on the wall 2021'. The corresponding database table 'Songs' in DBeaver shows three rows of data.

The middle row shows the Spotify web application with the 'Artists' section displaying 'Iron Maiden'. The corresponding database table 'Artists' in DBeaver shows one row of data for Iron Maiden.

The bottom row shows the Spotify web application with the 'Iron Maiden collections' section displaying three items: 'Title: The trooper - Iron Maiden, an 1983', 'Title: Wasted years - Iron Maiden, an 1988', and 'Title: The writing on the wall - Iron Maiden, an 2021'. The corresponding database table 'SongArtist' in DBeaver shows three rows of data.

(Afisarea in pagina web a tutoror melodiilor si a artistilor din DB, dar si a colectiei de melodii pt Iron Maiden)

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'Spotify' database structure, including tables, views, indexes, procedures, and packages. The 'Users' table is selected. The right pane shows the 'Users' table data in a grid view. The table has columns: id, username, password, email, phone, and isAdmin. Two rows of data are visible:

id	username	password	email	phone	isAdmin
1	Albertbut12	\$2b\$12\$gvN6H4KJ	albert@yahoo	1,234,567	1
2	Albertbut123	\$2b\$12\$yJfDCMv0	albert.alex	1,234	false

FastAPI - Swagger UI

localhost:3006/docs#/default/create_users_users.post

Parameters

Name	Description
_username * required string (query)	User_Test
_password * required string (query)	test
_email * required string (query)	test@gmail.com
_phoneNumber * required integer (query)	0712222333
_isAdmin * required string (query)	true

Execute Clear

Responses

200

Response body

```
{
  "access-control-allow-credentials": true,
  "content-length": 2,
  "content-type": "application/json",
  "date": "Sun, 12 Jun 2023 17:06:44 GMT",
  "server": "uicorn"
}
```

Response headers

```
access-control-allow-credentials: true
content-length: 2
content-type: application/json
date: Sun, 12 Jun 2023 17:06:44 GMT
server: uicorn
```

The screenshot shows the DBeaver 22.3.3 interface with the 'Users' table selected. The table structure is as follows:

id_U	username	password	email	phone	isAdmin
5	Albertvui12	\$2b5125qwR6K5kd	albert@yahoo	1,234,843	1
6	Albertvui23	\$2b5125gFDCMV8	albert.aloxa	1,234	false
7	Albertvui34	\$2b5125Z6Bg03Y	test@gmail	712,222,333	true

The left sidebar shows the database structure, including Spotify, SongArtist, and Users tables. The bottom status bar indicates 3 rows fetched.

GET

/collections/{artist_id}/song

Get Artist Song

Parameters

Cancel

Name	Description
artist_id * required	
integer	
(path)	
6	

Execute

Clear

Responses

Curl

```
curl -X 'GET' \  
  'http://localhost:8086/collections/6/song' \  
  -H 'accept: application/json'
```

Request URL

```
http://localhost:8086/collections/6/song
```

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "Title: The trooper - Iron Maiden, an 1983", "Title: Wasted years - Iron Maiden, an 1988", "Title: The writing on the wall - Iron Maiden, an 2021" }</pre></div><div><div>Response headers</div><div><pre>content-length: 146 content-type: application/json date: Sun, 22 Jan 2023 17:28:02 GMT server: uvicorn</pre></div></div></div>

Responses

Code	Description	Links
------	-------------	-------

(Colectia de melodii a artistului cu id-ul 6)

GET /playlist/{playlist_name} Show Playlist

Parameters

Name Description

_playlist * required string (query) Muzica rock

Execute Clear

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8086/playlist/{playlist_name}?_playlist=MuzicaX20rock' \
  -H 'accept: application/json'
```

Request URL

http://localhost:8086/playlist/{playlist_name}?_playlist=MuzicaX20rock

Server response

Code Details

200

Response body

```
"\n{'id = 63cd51bdc0264a2fc35ac5e title = The trooper artist = Iron Maiden ', 'id = 63cd51d1cb0264a2fc35ac5f title = Wasted years artist = Iron Maiden '}]"
```

Response headers

```
content-length: 155
content-type: application/json
date: Sun, 22 Jan 2023 17:32:46 GMT
server: uvicorn
```

Responses

Code Description Links

(Melodiile din playlist-ul „Muzica rock”

POST /playlist Create Playlist

Parameters

Name Description

_playlist * required string (query) new Playlist

_song * required string (query) The trooper

_artist * required string (query) Iron Maiden

Execute Clear

Responses

Curl

```
curl -X 'POST' \
  'http://localhost:8086/playlist?_playlist=newX20Playlist&_song=TheX20trooper&_artist=IronX20Maiden' \
  -H 'accept: application/json'
```

Request URL

http://localhost:8086/playlist?_playlist=newX20Playlist&_song=TheX20trooper&_artist=IronX20Maiden

Server response

Code Details

200

Response body

```
null
```

Response headers

```
access-control-allow-credentials: true
access-control-allow-headers: *
content-length: 4
content-type: application/json
date: Sun, 22 Jan 2023 17:34:32 GMT
server: uvicorn
```

(Crearea unui nou Playlist (new Playlist) in care
Se adauga melodia The trooper cu artistul Iron Maiden)

GET /playlist/{playlist_name} Show Playlist

Parameters

Name Description

_playlist * required string (query) new Playlist

Execute Clear

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8086/playlist/{playlist_name}?_playlist=newX20Playlist' \
  -H 'accept: application/json'
```

Request URL

http://localhost:8086/playlist/{playlist_name}?_playlist=newX20Playlist

Server response

Code Details

200

Response body

```
"\n{'id = 63cd739f65fc32982ba4341a title = The trooper artist = Iron Maiden '}]"
```

Response headers

```
content-length: 79
content-type: application/json
date: Sun, 22 Jan 2023 17:35:32 GMT
server: uvicorn
```

Responses

Code Description Links

(Afisarea playlist-ului „new Playlist” din baza de date noSQL)

GET

/user/{username}

Get User By Username

Cancel

Parameters

Name	Description
<u>_username</u> * required	
string	
(query)	

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8086/user/{username}?_username=User_Test' \
  -H 'accept: application/json'
```

Request URL

```
http://localhost:8086/user/{username}?_username=User_Test
```

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "User": { "username": "User_Test", "email": "test@gmail.com", "isActive": "true", "password": "\$2b\$12\$ZKgc9VY8aSi4wM0KA.0mdt0udePwXc1Viv6dHfmbyn.c1q", "id": "1", "phoneNumber": "712222333" } }</pre></div><div><div>Response headers</div><div><pre>content-length: 183 content-type: application/json date: Sun, 23 Jan 2023 17:31:01 GMT server: ucfcore</pre></div></div></div>

Responses

Code	Description	Links
------	-------------	-------

The screenshot shows the Microsoft Access interface with the 'Database' view selected. The 'Users' table is highlighted in the left-hand navigation pane. The table's data is displayed in the main window, showing columns for Id, Username, Password, Email, Phone, and IsAdmin. The data is as follows:

Id	Username	Password	Email	Phone	IsAdmin
5	Albertvni2	S2b5123qwr6k5kd	albert@yahoo	1,234,843	1
6	Albertbvi123	S2b51235rjFDfCMV8	albert.alexa	1,234	false
8	User_Test	S2b5123Zdj8cgo3Y	test@gmail	712,222,333	true

(Cautarea unui user dupa Username)

POST

/tokenLogin For Access Token

Parameters

No parameters

Request body required

application/x-www-form-urlencoded

grant_typestringpattern: password☒Send empty value

username * requiredstring

password * requiredstring

scopescope☒Send empty value

client_idstring☒Send empty value

client_secretstring☒Send empty value

Execute

Clear

Responses

Curl

```
curl -X 'POST' \
'http://localhost:8086/token' \
-H 'accept: application/json' \
-H 'Content-Type: application/x-www-form-urlencoded' \
-d 'grant_type=User_Test&password=test&scope=&client_id=&client_secret='
```

Request URL

http://localhost:8086/token

Server response

Code	Details
200	<div>Response body<pre>{ "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiI3VGVyK1Rlc3QiLCJleHAiOjE2NzQwMTASNDk9LjpmYWZlbnR5bmFmFR3R6aDQ6Yy" }</pre></div> <div>Download</div>

Response headers

(Verificare login User si generarea unui „acces_token”)

default

POST	/token	Login For Access Token	▼
GET	/user/me	Read Users Me	▼
GET	/users	Users List	▼
POST	/users	Create Users	▼
GET	/users/{user_id}	Get User By Id	▼
DELETE	/users/{user_id}	Delete User	▼
GET	/user/{username}	Get User By Username	▼
GET	/roles	Roles List	▼
POST	/roles	Create Roles	▼
GET	/songs	Songs List	▼
POST	/songs	Create Songs	▼
GET	/songs/{song_id}	User Detail	▼
POST	/songs/{song_id}	Update Song	▼
DELETE	/songs/{song_id}	Delete Song	▼
GET	/artists	Artists List	▼
POST	/artists	Create Artists	▼
GET	/artists/{artist_id}	Get Artist By Id	▼
POST	/artists/{artist_id}	Update Artist	▼
DELETE	/artists/{artist_id}	Delete Artist	▼
GET	/artist/{artist_name}	Get Artist By Name	▼
POST	/collection	Create Collection	▼
GET	/collections/{artist_id}/song	Get Artist Song	▼
POST	/playlist	Create Playlist	▼
GET	/playlist/{playlist_name}	Show Playlist	▼

Schemas

Body_login_for_access_token_token_post	>
Gen	>
HTTPValidationError	>
Token	>
ValidationError	>

New song and artist

Title:

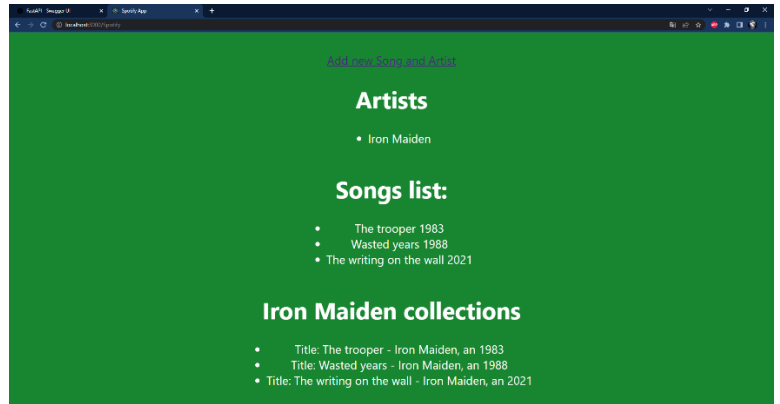
Gen muzical:

An aparitie:

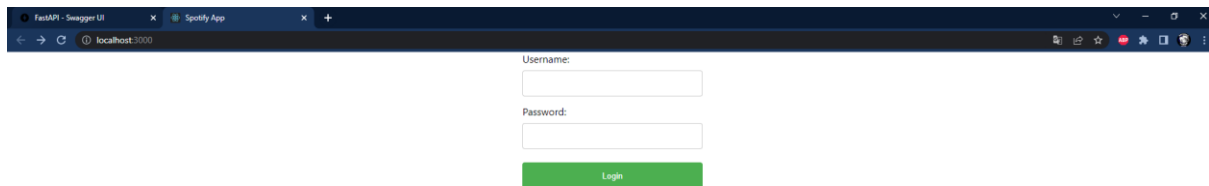
Nume artist:

is Active?

Submit



(pagina de adaugare a unei melodii si a artistului)



(Pagina de Login)