Library Mgt

How to run the application

There are 2 ways to run this application.

1 way

In the terminal,

1. mvn clean install
2. docker-compose build
3. docker-compose up -d

2 way

1. Open the project in intellij
2. Do mvn clean install the project
3. In the docker-compose.yml , you will find the postgres database,
4. Run that service
5. Then on intellij, run the application, and follow the below things

then run below things in postman,

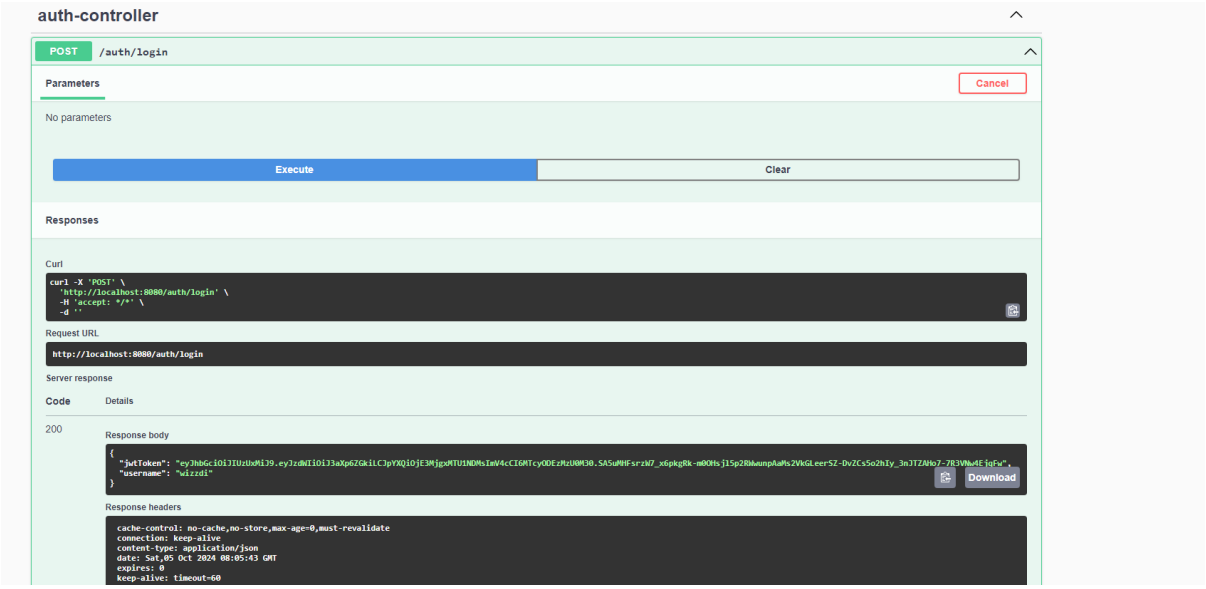
Here is the API list where all Post api and one get api to fetch the data.

Login api:

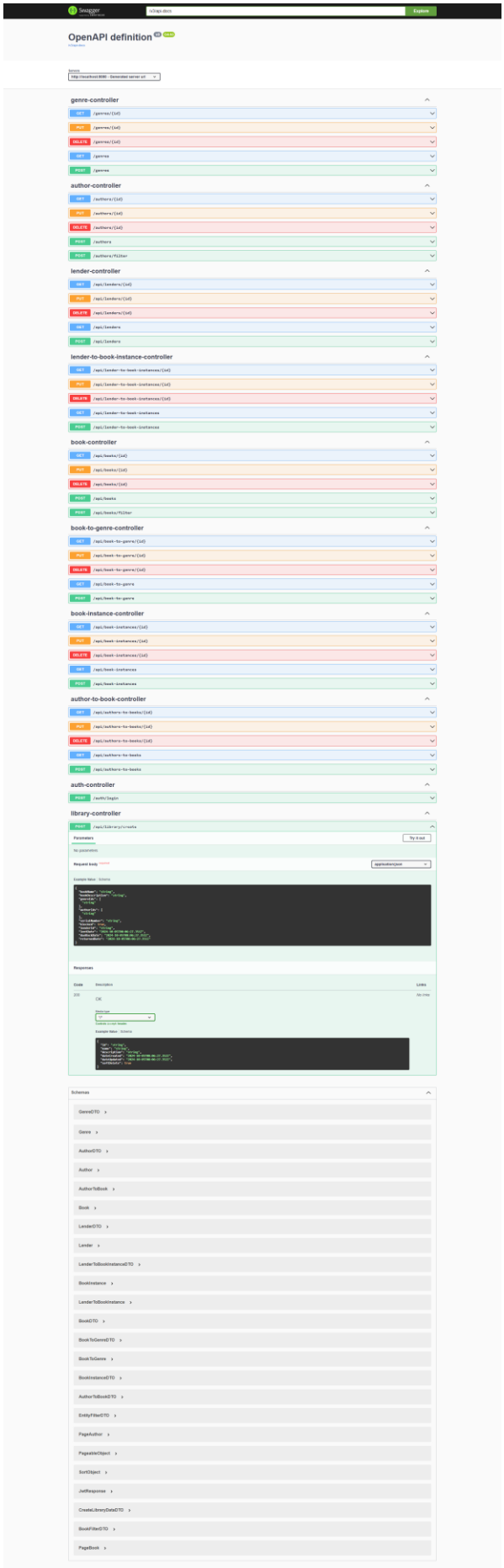
POST : - http://localhost:8080/auth/login

Default username : wizzdi

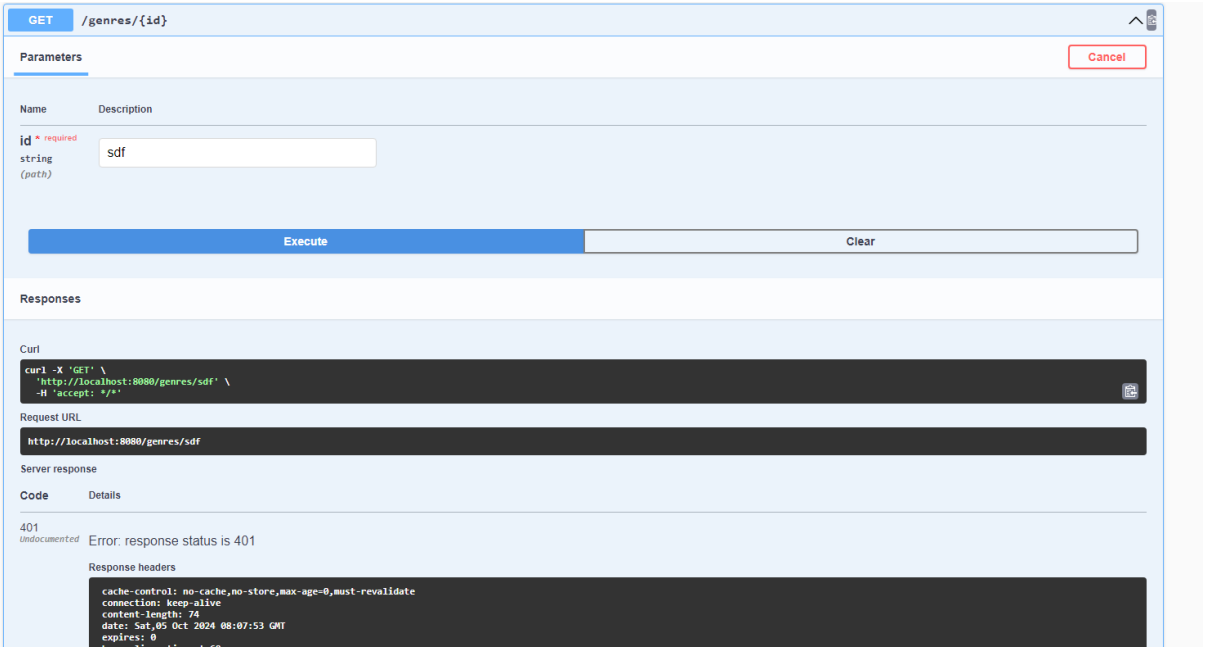
Default password: wizzdi



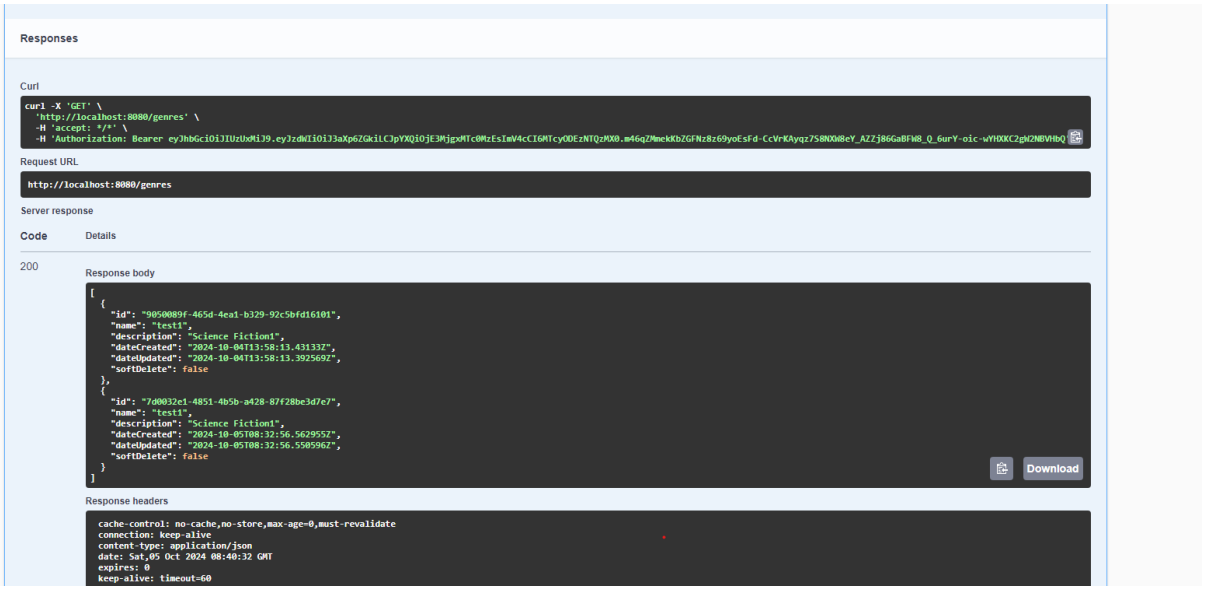
Swagger All Crud Api’s screenshot


Swagger
Explore
OpenAPI definition
genre-controller
author-controller
lender-controller
lender-to-book-instance-controller
book-controller
book-to-genre-controller
book-instance-controller
author-to-book-controller
auth-controller
library-controller
Schemas

ACCESS DENIED ERROR IF NOT PASSED TOKEN



ACCESS PASSED AFTER PASSING TOKEN



NOTE: --
REST YOU CAN ACCESS ALL API's on swagger. I am not putting all apis here.

*****Library Api which contain all things*****

POST:

http://localhost:8080/api/library/create

Request:


```
{
 "bookName": "string",
 "bookDescription": "string",
 "genreIds": [
  "string"
 ],
 "authorIds": [
  "string"
 ],
 "serialNumber": "string",
 "blocked": true,
 "lenderId": "string",
 "lentDate": "2024-10-05T08:37:36.675Z",
 "dueBackDate": "2024-10-05T08:37:36.675Z",
 "returnedDate": "2024-10-05T08:37:36.675Z"
}
```


Old Apis*******

1. Author

POST - http://localhost:8080/authors

```
{
  "email": "author1@example.com",
  "name": "John1",
  "address": "123 Author St.",
  "phoneNumber": "123-456-7890",
  "gender": "MALE",
  "birthDate": "1980-01-01T00:00:00Z",
  "socialSecurityNumber": "123-45-6789"
}
```

Response:


```
{
  "id": "46e69a0e-1e40-4f60-8773-8a5faaa1bfe2",
  "name": "John1",
  "description": null,
  "dateCreated": "2024-09-30T19:58:23.112867+05:30",
  "dateUpdated": "2024-09-30T19:58:23.1088666+05:30",
  "softDelete": false,
  "email": "author1@example.com",
  "address": "123 Author St.",
  "phoneNumber": "123-456-7890",
  "gender": "MALE",
  "birthDate": "1980-01-01T00:00:00Z",
  "socialSecurityNumber": "123-45-6789",
  "books": null
}
```

2. Books

POST- http://localhost:8080/api/books

Request:

{

"name": "Sample Book",

"description": "A description of the sample book.",

"genreIds": ["d1a41fe2-3c59-4c9e-809d-1e33dcf113dc", "bda82300-403c-4cc3-b451-10142743f03e"],

"authorIds": ["7d08f1dc-c4ad-4003-b326-f93b6e53d083", "46e69a0e-1e40-4f60-8773-8a5faaa1bfe2"]

}

Response: -

{

"id": "3dc5df8d-b01b-4614-9b7b-5c2a510c6ac6",

"name": "Sample Book",

"description": "A description of the sample book.",

"dateCreated": "2024-09-30T20:02:20.5128873+05:30",

"dateUpdated": "2024-09-30T20:02:20.5248862+05:30",

"softDelete": false,

"genres": [

{

"id": null,

"name": null,

"description": null,

"dateCreated": "2024-09-30T20:02:20.5188863+05:30",

```
      "dateUpdated": "2024-09-30T20:02:20.5188863+05:30",
      "softDelete": false,
      "book": null,
      "genre": null
    },
    {
      "id": null,
      "name": null,
      "description": null,
      "dateCreated": "2024-09-30T20:02:20.5188863+05:30",
      "dateUpdated": "2024-09-30T20:02:20.5188863+05:30",
      "softDelete": false,
      "book": null,
      "genre": null
    }
  ],
  "authors": [
    {
      "id": null,
      "name": null,
      "description": null,
      "dateCreated": "2024-09-30T20:02:20.5188863+05:30",
      "dateUpdated": "2024-09-30T20:02:20.5188863+05:30",
      "softDelete": false,
      "author": null,
      "book": null,
      "mainAuthor": false
    },
```

```
    {
      "id": null,
      "name": null,
      "description": null,
      "dateCreated": "2024-09-30T20:02:20.5188863+05:30",
      "dateUpdated": "2024-09-30T20:02:20.5188863+05:30",
      "softDelete": false,
      "author": null,
      "book": null,
      "mainAuthor": false
    }
  ]
}
```


3. Genre
   POST - http://localhost:8080/genres

   Request:

```
{
    "description": "Science Fiction1",
    "name": "test1"
}
```

Response:


```
{
  "id": "bda82300-403c-4cc3-b451-10142743f03e",
  "name": "test1",
  "description": "Science Fiction1",
  "dateCreated": "2024-09-30T19:57:53.4070251+05:30",
```

```
  "dateUpdated": "2024-09-30T19:57:53.4060254+05:30",
  "softDelete": false
}
```


4. Filter Api :-

POST - http://localhost:8080/authors/filter

Request:

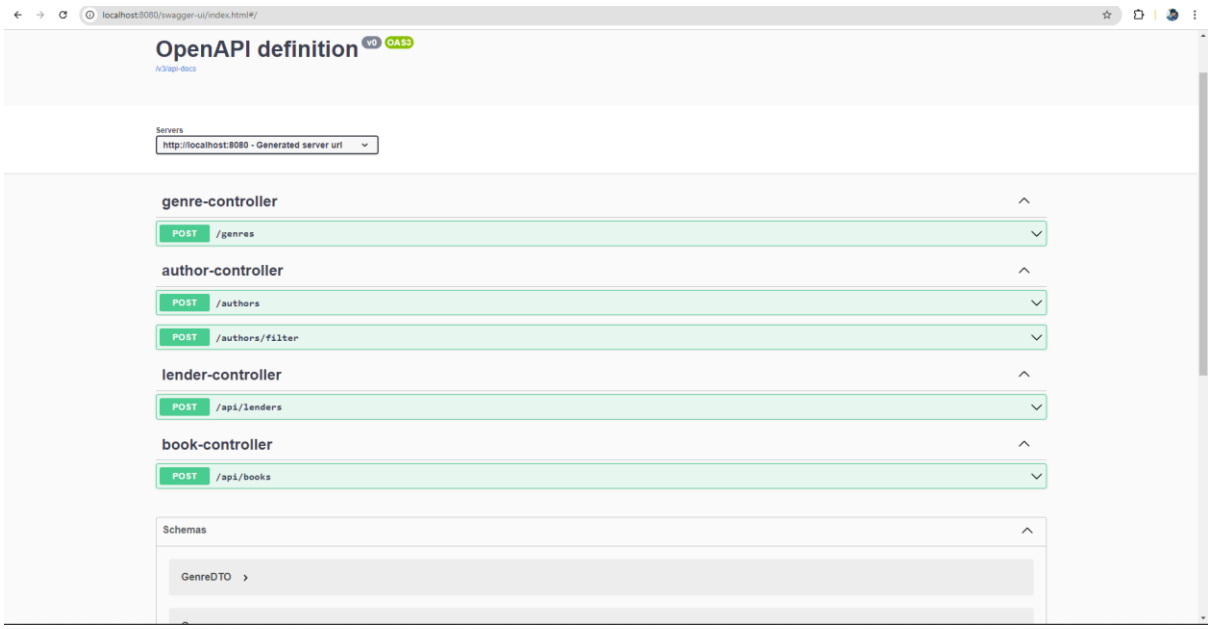

```
{
  "namePattern": "John",              // Pattern to filter names
  "referencedIds": ["46e69a0e-1e40-4f60-8773-8a5faaa1bfe2"],
  "startDate": "2024-09-30T00:00:00Z", // Start date for filtering
  "endDate": "2024-10-01T23:59:59Z"    // End date for filtering
}
```


Response: -

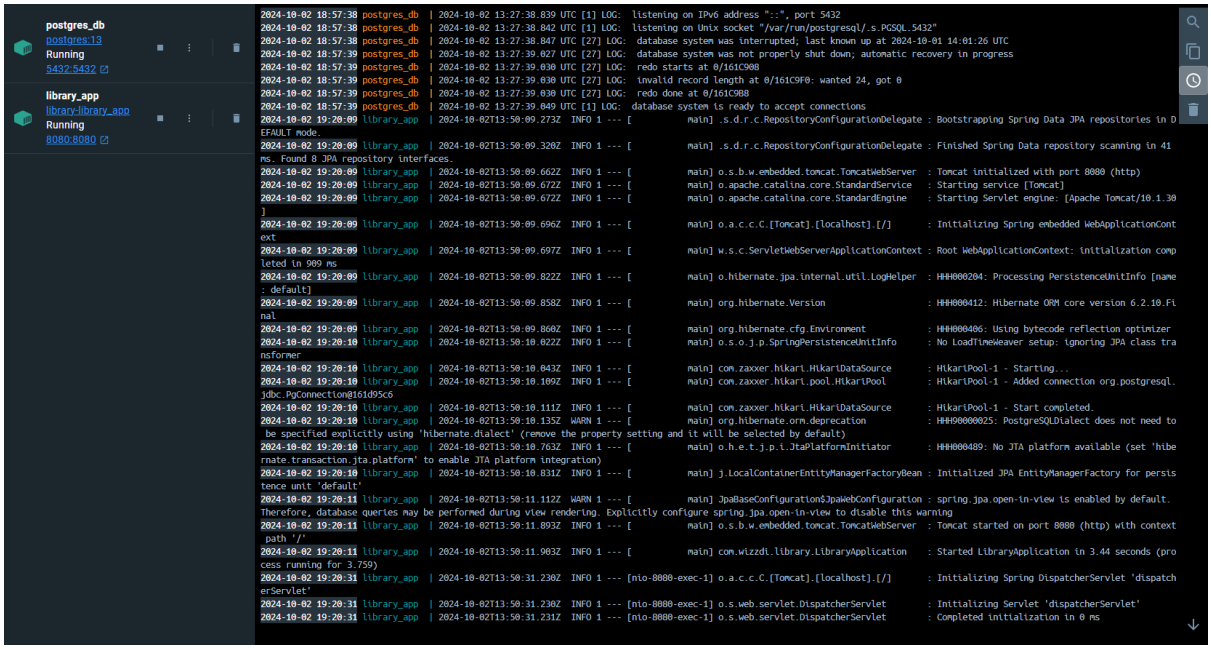

```
[
  {
    "id": "46e69a0e-1e40-4f60-8773-8a5faaa1bfe2",
    "name": "John1",
    "description": null,
    "dateCreated": "2024-09-30T14:28:23.112867Z",
    "dateUpdated": "2024-09-30T14:28:23.108867Z",
    "softDelete": false,
```

```
    "email": "author1@example.com",
    "address": "123 Author St.",
    "phoneNumber": "123-456-7890",
    "gender": "MALE",
    "birthDate": "1980-01-01T00:00:00Z",
    "socialSecurityNumber": "123-45-6789",
    "books": []
  }
]
```

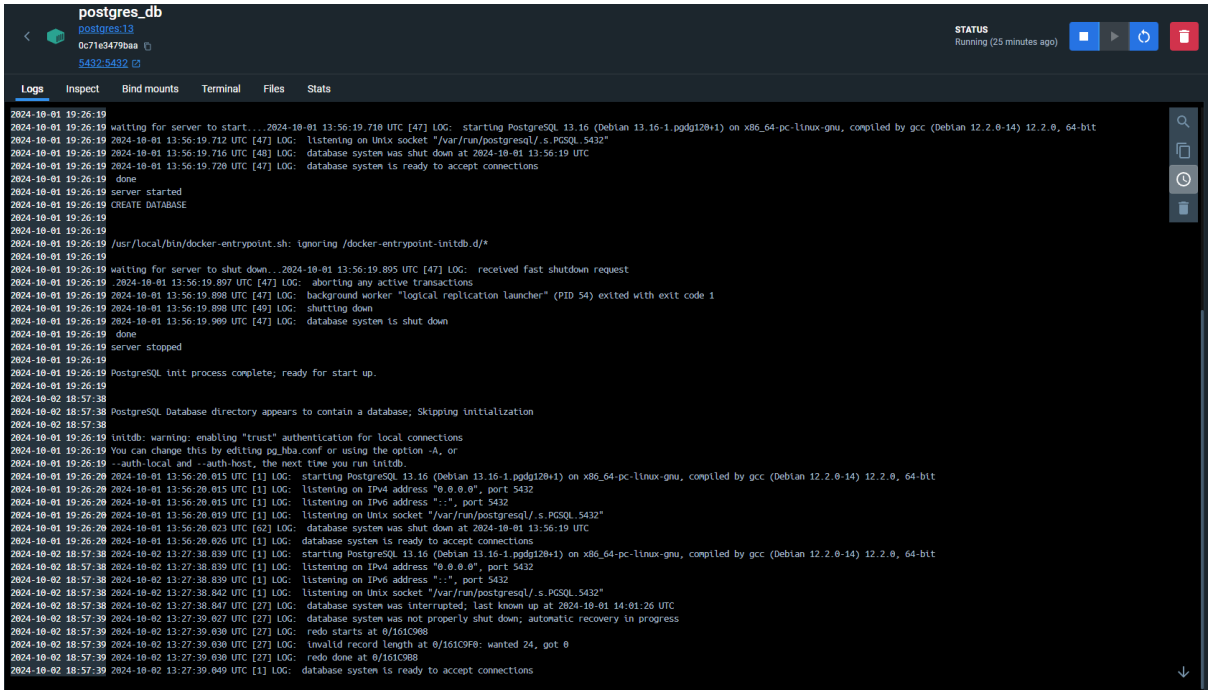
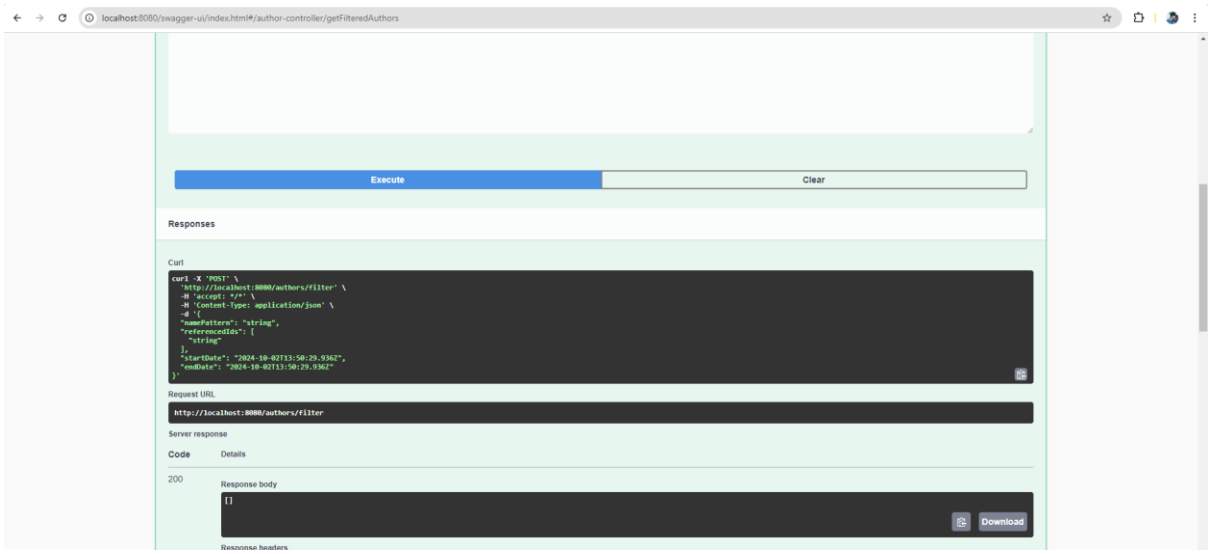

Swagger Url - http://localhost:8080/swagger-ui/index.html#/



Docker library app running screenshot



Database running screenshot



Swagger running screenshot