Assessment: MapBox

Position: Backend Engineer – Maps API

Task: 🦋 Butterfly critique

Task 1: Allow a user to rate butterflies on a scale of 0 through 5

Task 2: Allow retrieval of a list of a user's rated butterflies, sorted by rating

Implementation

butterfly-homework\src\

(1) Index.js → Developed two APIs

- **a.** addRatings (POST)
- **b.** userRatings (GET)

(2) validators.js → Implemented two validators

- **a.** validateRatings () → check the type of the input
- **b.** validateRatingRange () → validate the range of the input

butterfly-homework\test

(3) index.test.js → implemented testing functionality for APIs

(4) validators.test.js → implemented testing functionality for implemented validators in validators.js

JSON File Name: butterflies.db.json

Existing JSON:

The existing JSON contains butterflies entity and users entity. Here, we will add one more entity(table) that is "ratings" to fulfill the given tasks.

Reason: To allow user to rate from 0 to 5 ratings on the butterflies and retrieve the list of rated butterfliers of a particular user in a sorting manner.

Important keys of the ratings json:

```
"ratings": [    {

                "user_id": "OOWzUaHLsK",

                "butterfly_id": "GI9_EuH8s1",

                "rate": 4

                }
```

Tool used to test the APIs: Postman

Node version: v14.15.3

Npm version: 6.14.9

Task 1:

Assumption: User has already logged-In and selected existing butterflies to give rating

File: index.js

API: app.post ('/addRatings', async (req, res) ➔ {}).

HTTP Method used: POST

→Here, addRatings API is used to add the data to the database (JSON) in "ratings".

→To Accomplish this API,

Validations are performed before adding the data to the database.

(1) validate the type of the data

a. User_id: string
b. Butterfly_id: string
c. Rate: number

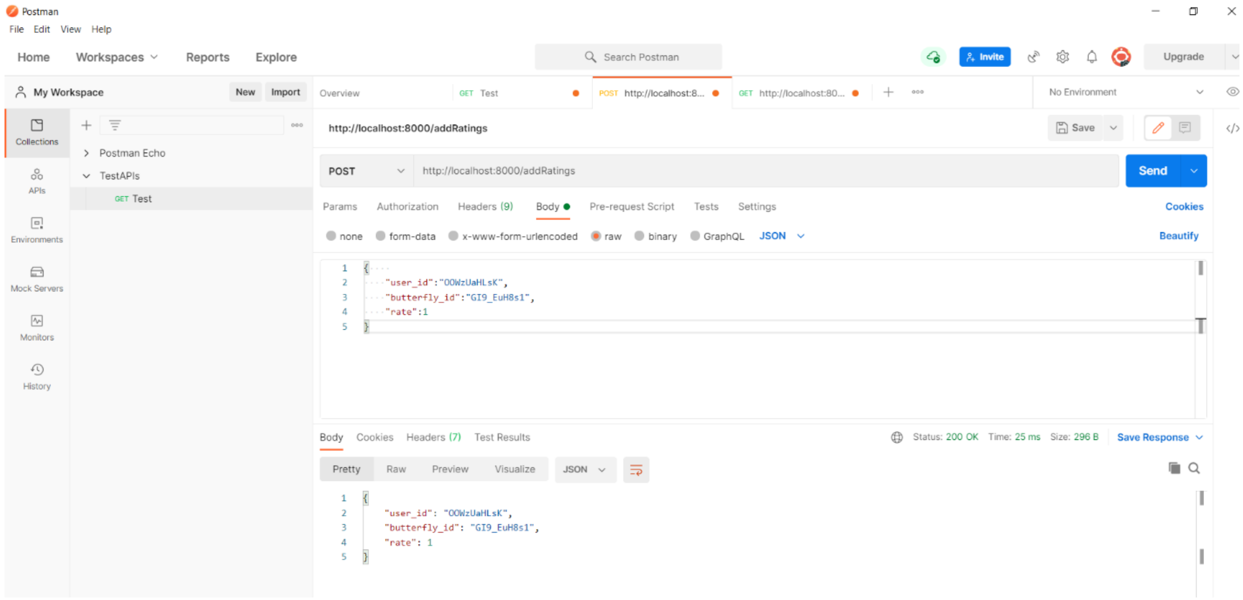
Function created: **validateRatings ()**

(2) Validate the range of the rate (0 to 5)

Function created: **validateRatingRange ()**

Screenshots of the API

Postman API Testing



http://localhost:8000/addRatings

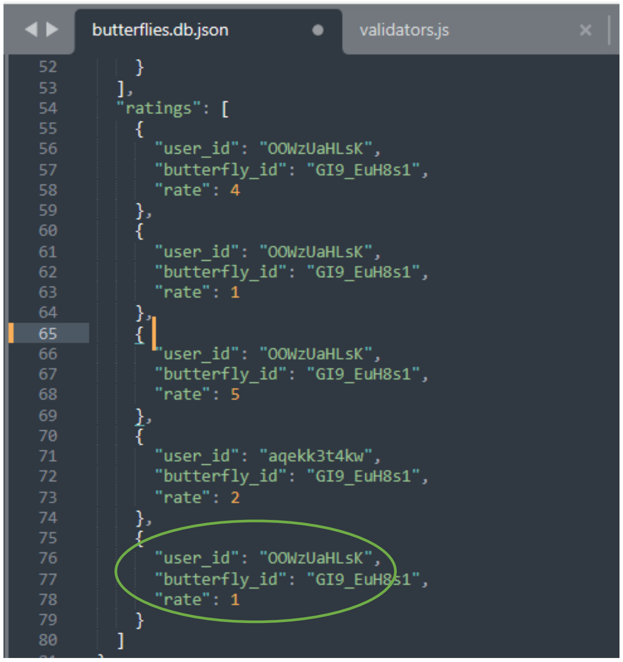
POST http://localhost:8000/addRatings

```
{
    "user_id":"OOWzUaHLsK",
    "butterfly_id":"GI9_EuH8s1",
    "rate":1
}
```

Status: 200 OK Time: 25 ms Size: 296 B

```
{
    "user_id": "OOWzUaHLsK",
    "butterfly_id": "GI9_EuH8s1",
    "rate": 1
}
```

Updated butterflies.db.json



```
      }
    ],
    "ratings": [
      {
        "user_id": "OOWzUaHLsK",
        "butterfly_id": "GI9_EuH8s1",
        "rate": 4
      },
      {
        "user_id": "OOWzUaHLsK",
        "butterfly_id": "GI9_EuH8s1",
        "rate": 1
      },
      {
        "user_id": "OOWzUaHLsK",
        "butterfly_id": "GI9_EuH8s1",
        "rate": 5
      },
      {
        "user_id": "aqekk3t4kw",
        "butterfly_id": "GI9_EuH8s1",
        "rate": 2
      },
      {
        "user_id": "OOWzUaHLsK",
        "butterfly_id": "GI9_EuH8s1",
        "rate": 1
      }
    ]
```

Task 2:

File: index.js

API: app.get (‘/userRatings/:user_id’, async (req, res) ➔ {}).

HTTP Method used: GET

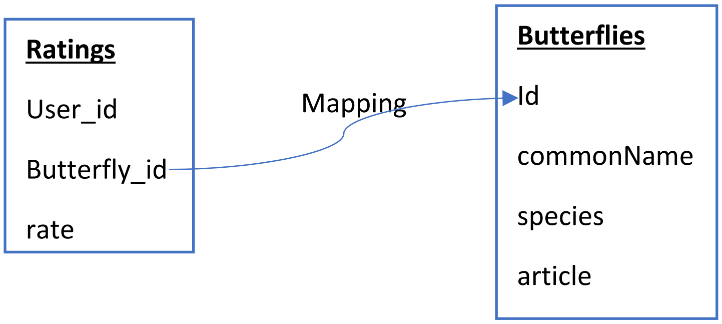
→Here, we will get the list of the butterflies in a sorting order based on ratings of the user.

Logic:

→ We will fetch the data of a particular user from ratings on user_id provided in the URL.

→After that, we will sort the ratings in a descending order.

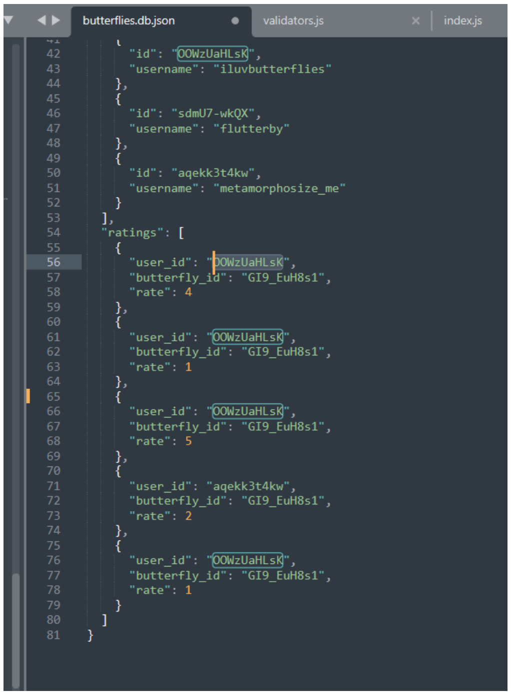
→ Mapping of ratings and butterflies to get the butterfly data.



→Once we map both the ids from the tables, we can get the details of the butterflies.

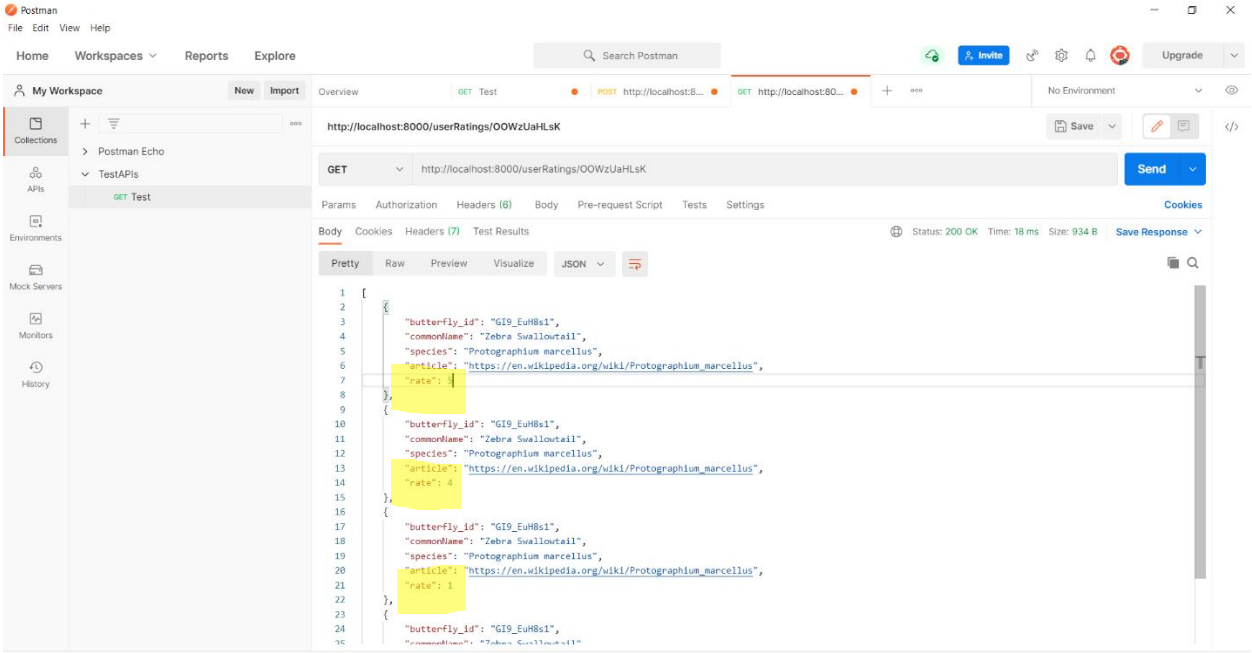
Screenshots

Unsorted JSON Data in “ratings”



```
    {
      "id": "OOWzUaHLsK",
      "username": "iluvbutterflies"
    },
    {
      "id": "sdmU7-wkQX",
      "username": "flutterby"
    },
    {
      "id": "aqekk3t4kw",
      "username": "metamorphosize_me"
    }
  ],
  "ratings": [
    {
      "user_id": "OOWzUaHLsK",
      "butterfly_id": "GI9_EuH8s1",
      "rate": 4
    },
    {
      "user_id": "OOWzUaHLsK",
      "butterfly_id": "GI9_EuH8s1",
      "rate": 1
    },
    {
      "user_id": "OOWzUaHLsK",
      "butterfly_id": "GI9_EuH8s1",
      "rate": 5
    },
    {
      "user_id": "aqekk3t4kw",
      "butterfly_id": "GI9_EuH8s1",
      "rate": 2
    },
    {
      "user_id": "OOWzUaHLsK",
      "butterfly_id": "GI9_EuH8s1",
      "rate": 1
    }
  ]
}
```

Postman API Testing to display the butterflies in a sorting order



http://localhost:8000/userRatings/OOWzUaHLsK

GET http://localhost:8000/userRatings/OOWzUaHLsK

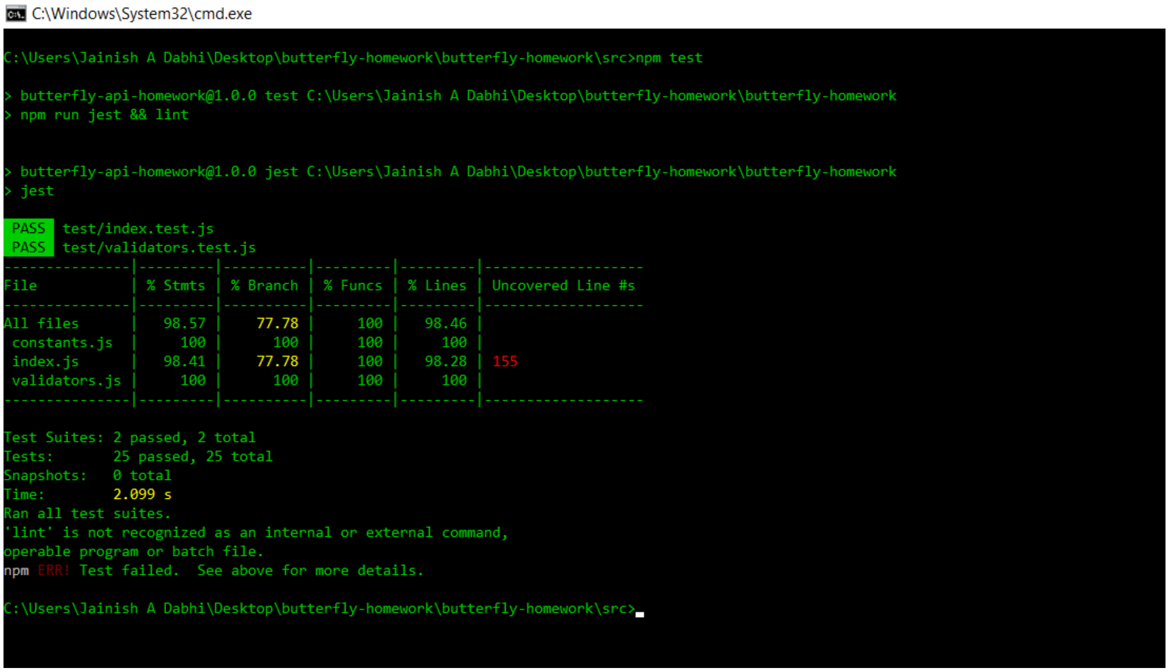
Status: 200 OK Time: 18 ms Size: 934 B

```
[
    {
        "butterfly_id": "GI9_EuH8s1",
        "commonName": "Zebra Swallowtail",
        "species": "Protographium marcellus",
        "article": "https://en.wikipedia.org/wiki/Protographium_marcellus",
        "rate": 5
    },
    {
        "butterfly_id": "GI9_EuH8s1",
        "commonName": "Zebra Swallowtail",
        "species": "Protographium marcellus",
        "article": "https://en.wikipedia.org/wiki/Protographium_marcellus",
        "rate": 4
    },
    {
        "butterfly_id": "GI9_EuH8s1",
        "commonName": "Zebra Swallowtail",
        "species": "Protographium marcellus",
        "article": "https://en.wikipedia.org/wiki/Protographium_marcellus",
        "rate": 1
    },
    {
        "butterfly_id": "GI9_EuH8s1",
        "commonName": "Zebra Swallowtail"
```

Testing

→Implemented testing methods to validate the test cases for all the APIs and Validators.

Screenshots of Testing



C:\Windows\System32\cmd.exe

```
C:\Users\Jainish A Dabhi\Desktop\butterfly-homework\butterfly-homework\src>npm test

> butterfly-api-homework@1.0.0 test C:\Users\Jainish A Dabhi\Desktop\butterfly-homework\butterfly-homework
> npm run jest && lint


> butterfly-api-homework@1.0.0 jest C:\Users\Jainish A Dabhi\Desktop\butterfly-homework\butterfly-homework
> jest

 PASS  test/index.test.js
 PASS  test/validators.test.js
---------------|---------|----------|---------|---------|-------------------
File           | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
---------------|---------|----------|---------|---------|-------------------
All files      |   98.57 |    77.78 |     100 |   98.46 |
 constants.js  |     100 |      100 |     100 |     100 |
 index.js      |   98.41 |    77.78 |     100 |   98.28 | 155
 validators.js |     100 |      100 |     100 |     100 |
---------------|---------|----------|---------|---------|-------------------

Test Suites: 2 passed, 2 total
Tests:       25 passed, 25 total
Snapshots:   0 total
Time:        2.099 s
Ran all test suites.
'lint' is not recognized as an internal or external command,
operable program or batch file.
npm ERR! Test failed.  See above for more details.

C:\Users\Jainish A Dabhi\Desktop\butterfly-homework\butterfly-homework\src>
```