OTOT TASK B

Dai Tianle A0196715U

Link to TASK **B1, B2.1-2.3**: https://github.com/ddx-510/ddNotes

Link to TASK **B2.4**: https://github.com/ddx-510/OTOT_B2.4_Serverless

TASK B1

To run the API locally, first clone this repo In the root folder, enter

```
npm install
npm run dev
```

The server will be set up.

`NODE_ENV=testing` determines the current mode. If you want to test on the local Postgres server, please set up the info inside the **.env** file and set `NODE_ENV=development`

The current sample .env file is as follows:

```
PG_USER=ddx
PG_PASSWORD=110119
PG_HOST=localhost
PG_PORT=5432
PG_DATABASE=dd_note
PORT=8080
NODE_ENV=testing
testURL=postgres://ebcngfzavaeurg:8742049fe1e4df1504acba5eb3b55145639f80dc
94d2cf41ac9a0b7c04b59191@ec2-3-213-66-35.compute-
1.amazonaws.com:5432/d2ecbgmihk2hla
```

TASK B1, B2.1 TEST API

To test the API using this Postman collection
https://www.getpostman.com/collections/32177646aa13a2be42ff

Some sample screenshots:

http://localhost:3000/api/notes/show/1

Save

GET http://localhost:3000/api/notes/show/1 Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookie

Query Params

KEY	VALUE	DESCRIPTION	•••	Bulk Ec
Key	Value	Description		

Body Cookies Headers (8) Test Results Status: 200 OK Time: 40 ms Size: 391 B Save Response

Pretty Raw Preview Visualize JSON


```
{
    "allNotes": [
        {
            "notes_id": 0,
            "user_id": 1,
            "title": "test1",
            "body": "asdfad"
        },
        {
            "notes_id": 1,
            "user_id": 1,
            "title": "asad",
            "body": "aas"
        }
```


POST http://localhost:3000/api/auth/login Send

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookie

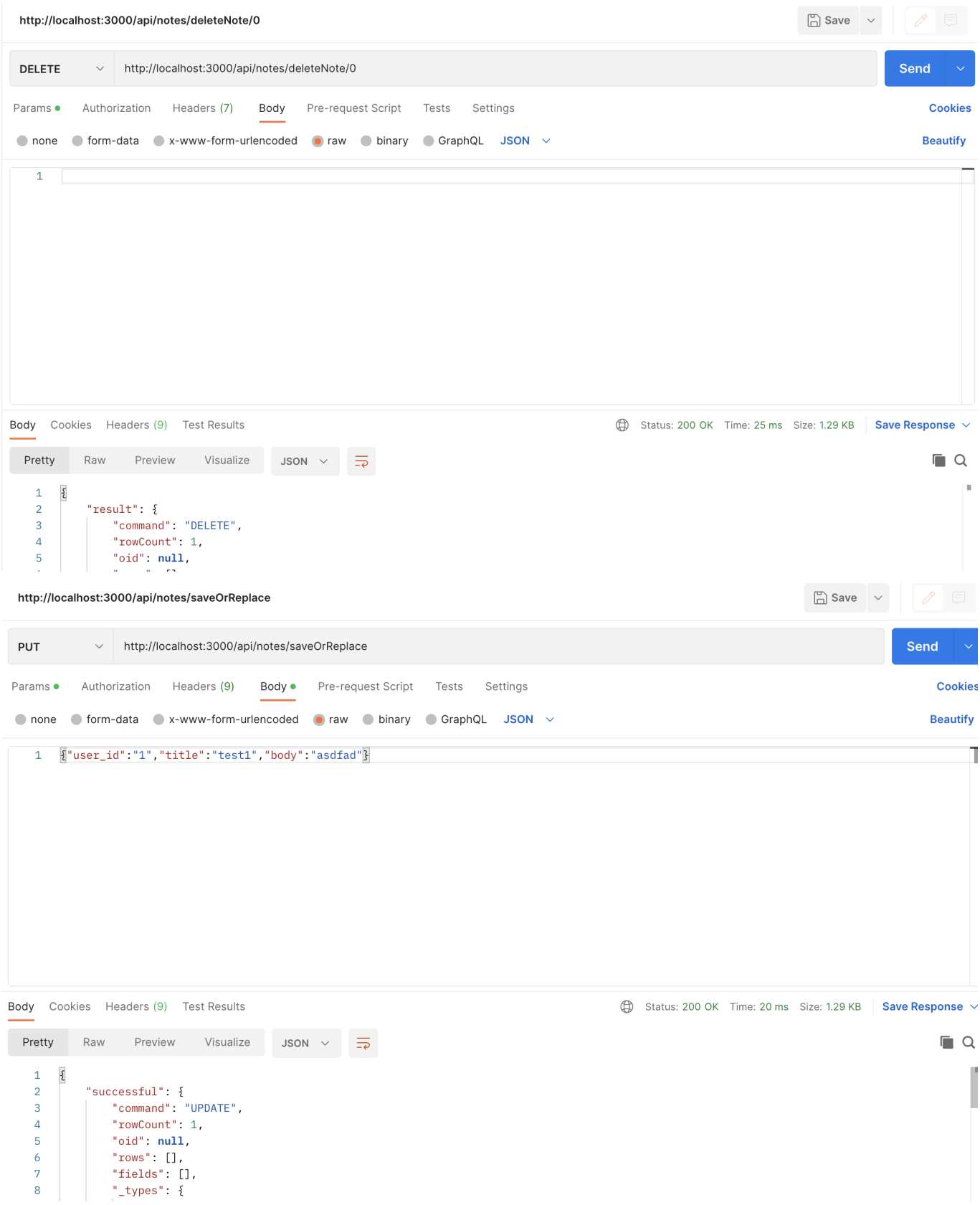
none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify


```
{"email":"ddxtest01@gmail.com","password":"12345"}
```


Body Cookies Headers (8) Test Results Status: 200 OK Time: 21 ms Size: 358 B Save Response

Pretty Raw Preview Visualize JSON


```
{
    "name": "ddx",
    "id": 1,
    "token": "A JWT token to keep the user logged in.",
    "msg": "Successful login"
}
```



CI and CD

To run test locally:

```
npm install
npm test
```

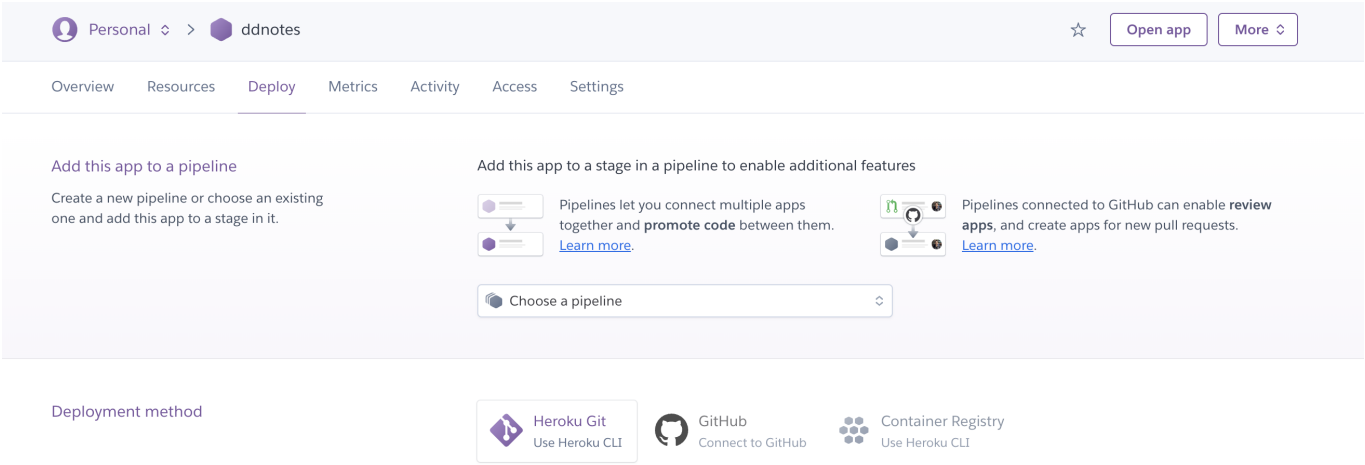
This will test the api on edge cases as well.

The CI was implemented using Github action, The workflows can be accessed here.

The CD was also implemented using the same workflow After the build and test are passed on push, It will trigger the deploy to heroku.

Note that this is not achieved by heroku git repo linkage, the whole deploy process was triggered by the workflows

Proof:



TASK B2.3 Front end

Access the app at https://ddnotes.herokuapp.com/

Login with the test account

```
email: ddxtest1@gmail.com
password:12345
```

TASK B2.4

Introduction

The serverless function is deployed on google cloud.

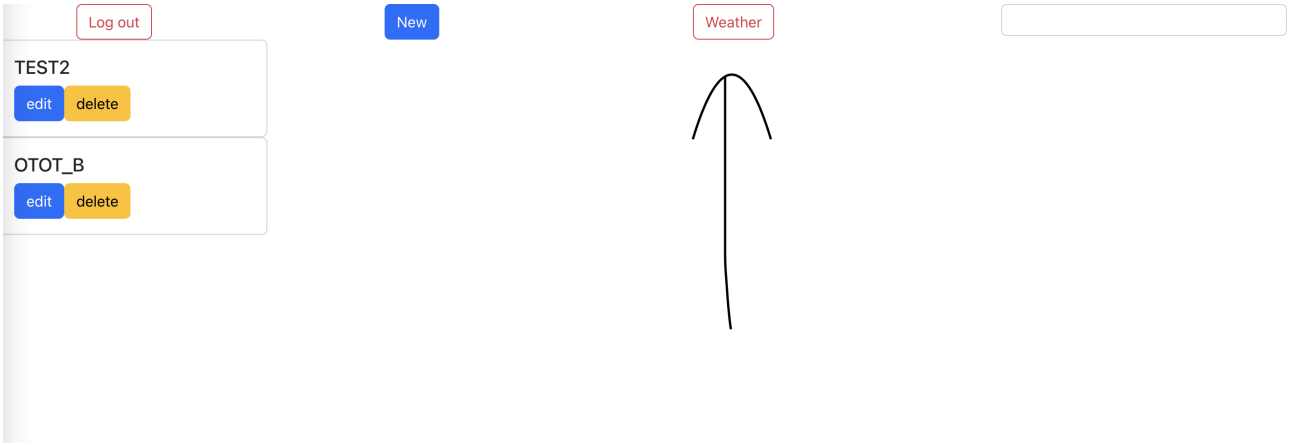
It is using the SG GOV API to get the weather data of all regions in singapore. The serverless function takes in a input name of the area of residence, and it will return the weather forecast of the next two hours of this region

This is the sample request to the serverless function with input name Clementi:

https://us-central1-otot-b-ddnote.cloudfunctions.net/weather?name=Clementi

On the front-end

Click the button:



And you will see the weather report when you are writing notes 😃