

Documenting the MiniWall development process

Phase A: Set up your MiniWall project and install necessary libraries

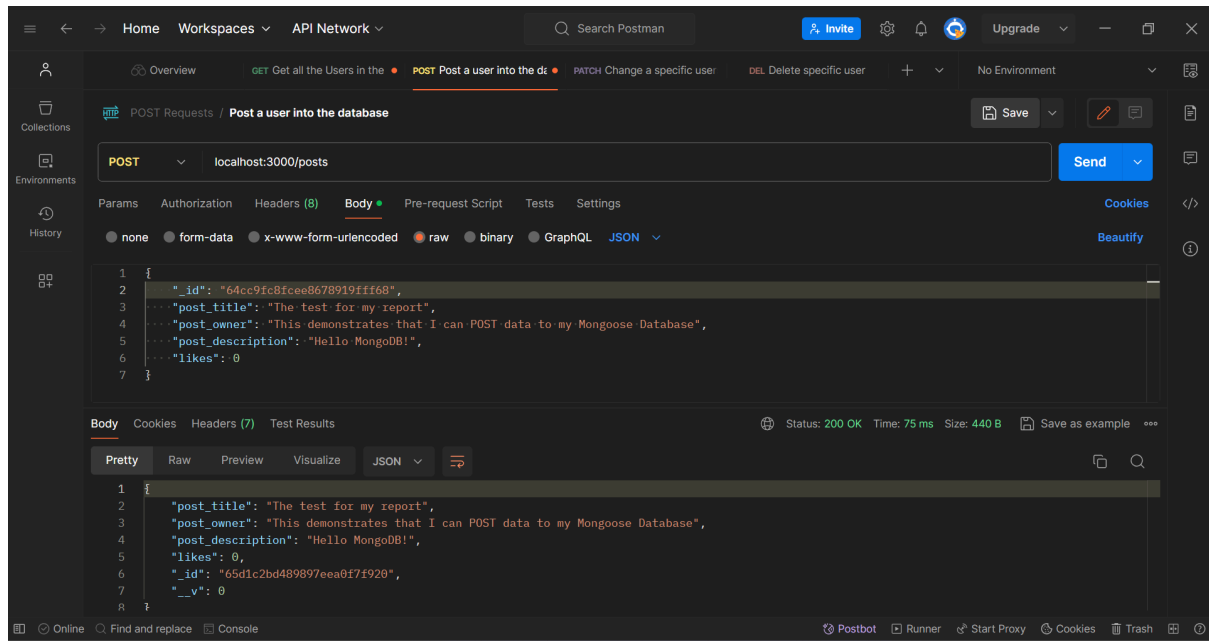
I began by initialising my npm project and installing all the required packets using the following commands (as documented in my Commands.md file):

```
npm init

npm install express nodemon mongoose dotenv body-parser

npm install joi bcryptjs jsonwebtoken
```

After this I followed the first lab (University of London, 2024a) and the second in order to apply the teachings to my own project (University of London, 2024b). Below I demonstrate through Postman and my browser that my URL endpoints are available and facilitate the CRUD API requirements (create, read, update and delete) - screenshots of which are displayed below:



Home Workspaces API Network Search Postman Invite Upgrade

Overview GET Get all the Users in the database and display them POST Post a user into the database PATCH Change a specific user's data DELETE Delete specific user's data No Environment

GET localhost:3000/posts Save Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body Cookies Headers (7) Test Results Status: 200 OK Time: 24 ms Size: 508 B Save as example

Pretty Raw Preview Visualize JSON

```
1 {
2   "_id": "65ccd3e2a42929988253f1b0",
3   "post_title": "SENDING ONE",
4   "post_owner": "LEGEND",
5   "post_description": "SENDING IT!",
6   "likes": 0,
7   "v": 0
8 }
```

Online Find and replace Console Postbot Runner Start Proxy Cookies Trash

Home Workspaces API Network Search Postman Invite Upgrade

Overview GET Get all the Users in the database and display them POST Post a user into the database PATCH Change a specific user's data DELETE Delete specific user's data No Environment

PATCH localhost:3000/posts/65d1c2bd489897eea0f7f920 Save Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies Beautify

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

```
1 {
2   "post_title": "This has been changed to demonstrate that PATCH functions",
3   "post_owner": "Thomas Gardner",
4   "post_description": "Hello MongoDB! Change this record for me",
5   "likes": 5
6 }
```

Body Cookies Headers (7) Test Results Status: 200 OK Time: 26 ms Size: 327 B Save as example

Pretty Raw Preview Visualize JSON

```
1 {
2   "acknowledged": true,
3   "modifiedCount": 1,
4   "upsertedId": null,
5   "upsertedCount": 0,
6   "matchedCount": 1
7 }
```

Online Find and replace Console Postbot Runner Start Proxy Cookies Trash

HomeWorkspacesAPI Network

Search Postman

Invite

Upgrade

Overview

GET Get all the Users In the

POST Post a user into the dt

PATCH Change a specific us

DEL Delete specific user

No Environment

DELETE Requests / Delete specific user

Save

Send

DELETE

localhost:3000/posts/65d1c2bd489897eea0f7f920

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body

Cookies

Headers (7)

Test Results

Status: 200 OKTime: 29 msSize: 273 BSave as example

Pretty

Raw

Preview

Visualize

JSON

```
1 {
2   "acknowledged": true,
3   "deletedCount": 1
4 }
```

Online

Find and replace

Console

Postbot

Runner

Start Proxy

Cookies

Trash

Course: Cloud computing

CSM020-2024-JAN: CSM

CSM020 - Cloud Compu

Cloud Computing Final C

localhost:3000

localhost:3000

Star

Up

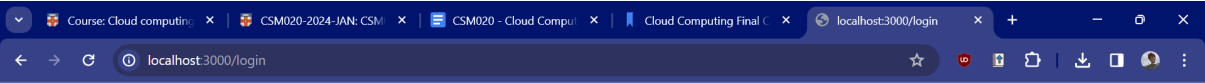
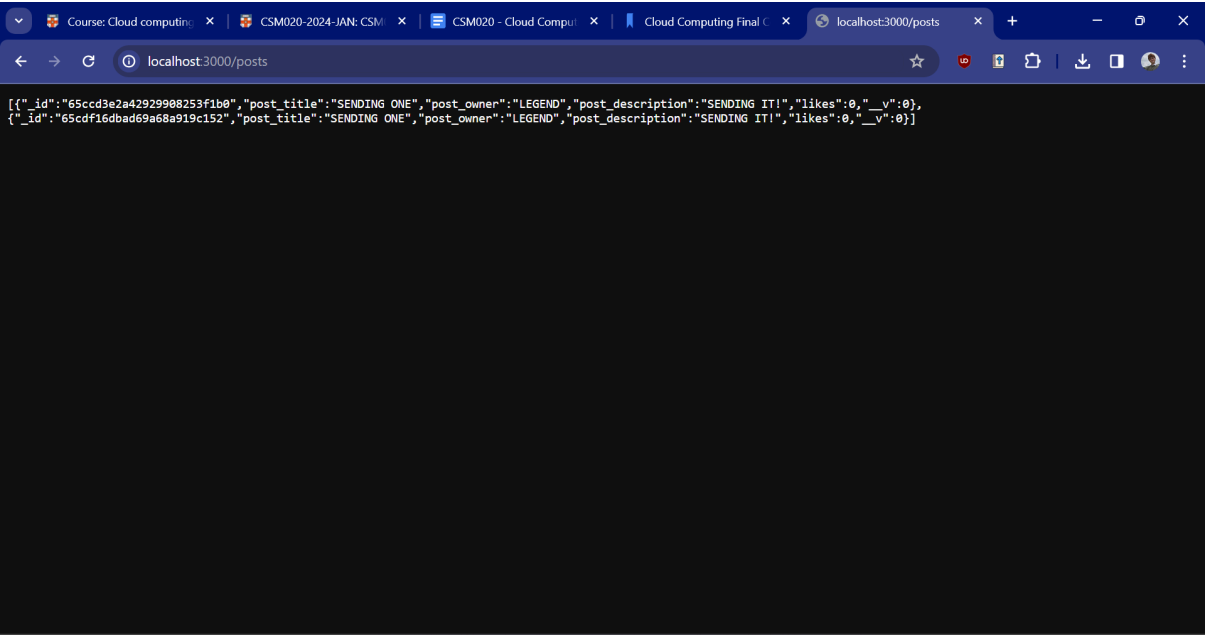
Download

Print

Share

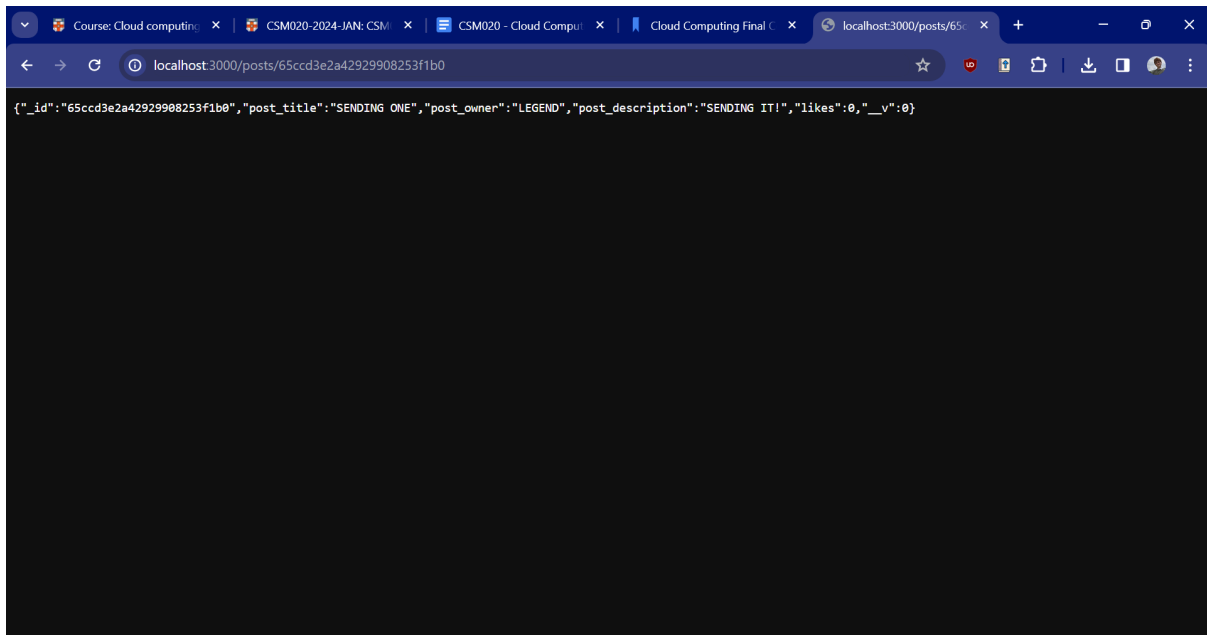
More

You are on your home page!

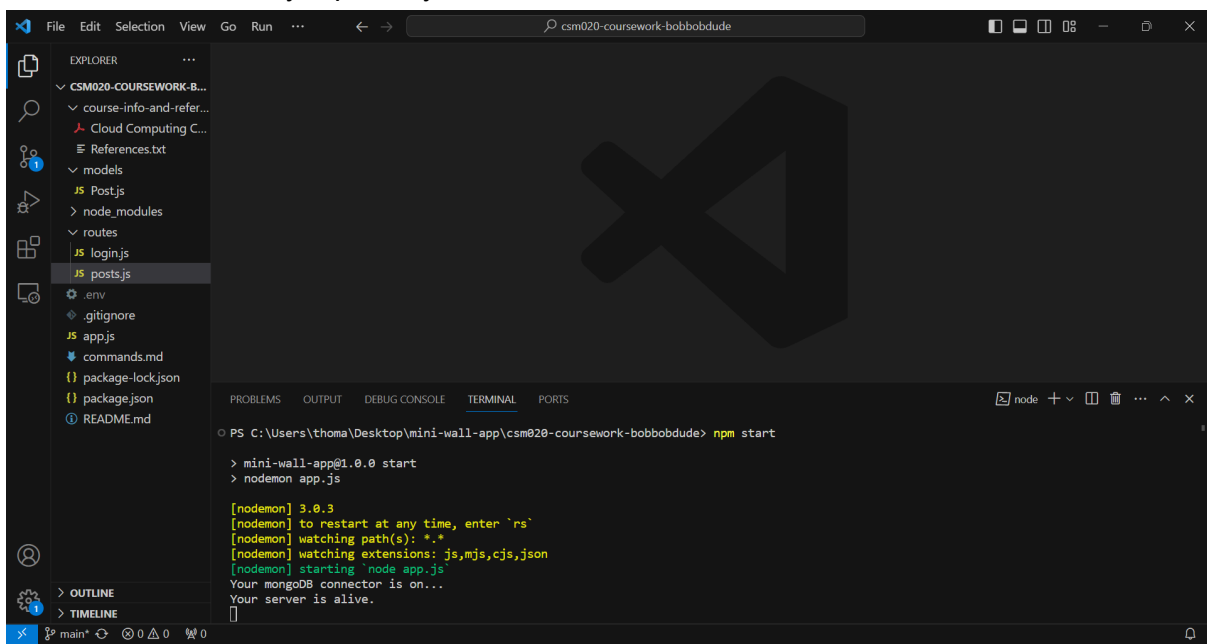


The temp login page!





The file structure of my repository can be seen below:



First we have a course-info-and-reference folder for the Coursework brief and References that I will update throughout the project.

Next we have the models folder which will contain all the different data schemas I will implement using MongoDB.

Then we have routes which allows users to navigate to different locations on the server and handle the data they input (such as passwords and posts) so it can be processed appropriately.

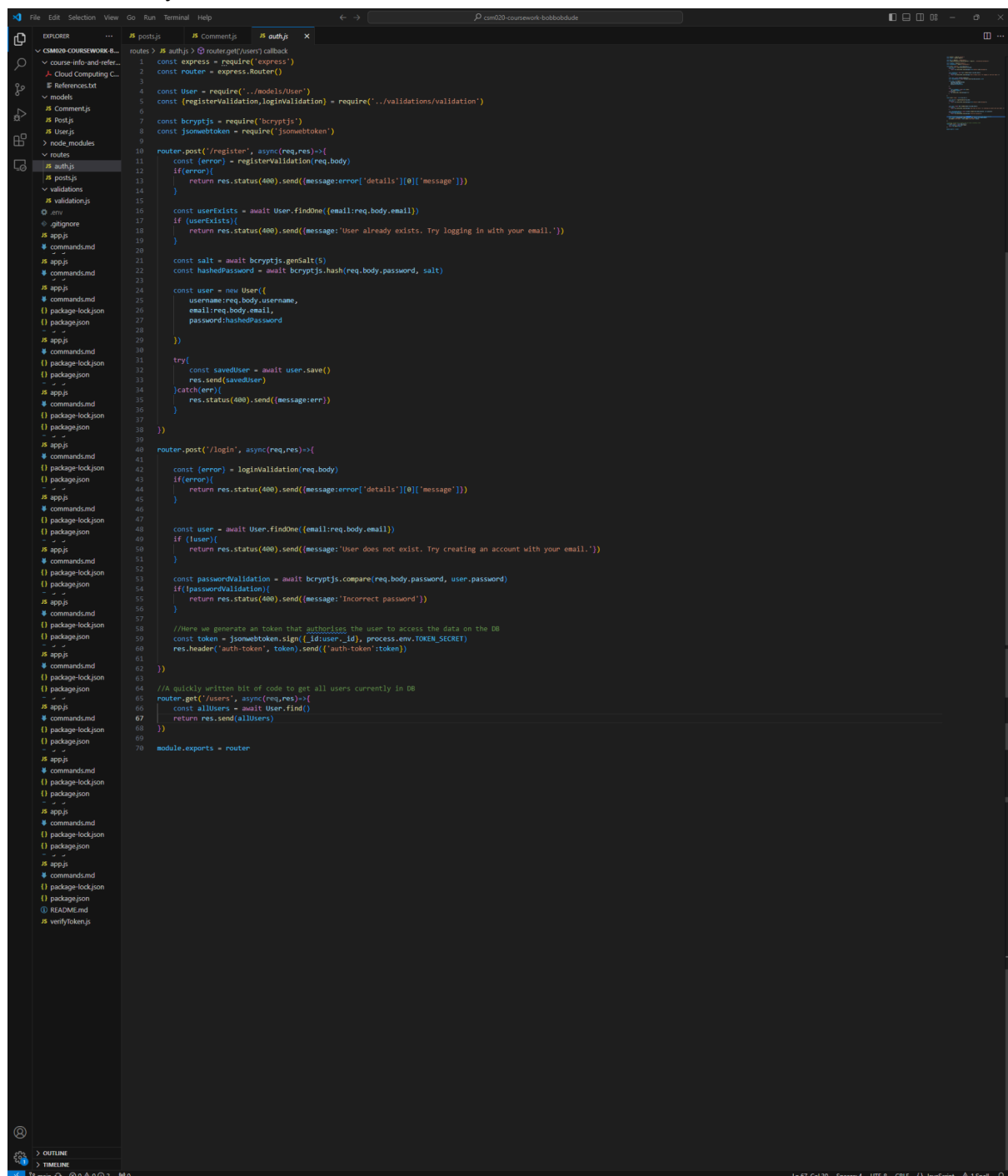
In the root we have the app.js file which contains the processes required to start and configure the server as well as other configuration files. The gitignore is essential as it

ensures that I do not upload the .env file which contains password information for my database as well as other sensitive information.

Phase B: Enforcing authentication/verification functionalities

I have created the MongoDB database to store both users and posts, and the only means of creating, reading, updating and deleting is through the use of an authentication token using JSON Web Tokens (JWT) sent in the header. In addition to this all user input from login credentials to posts is verified before being inserted in the server.

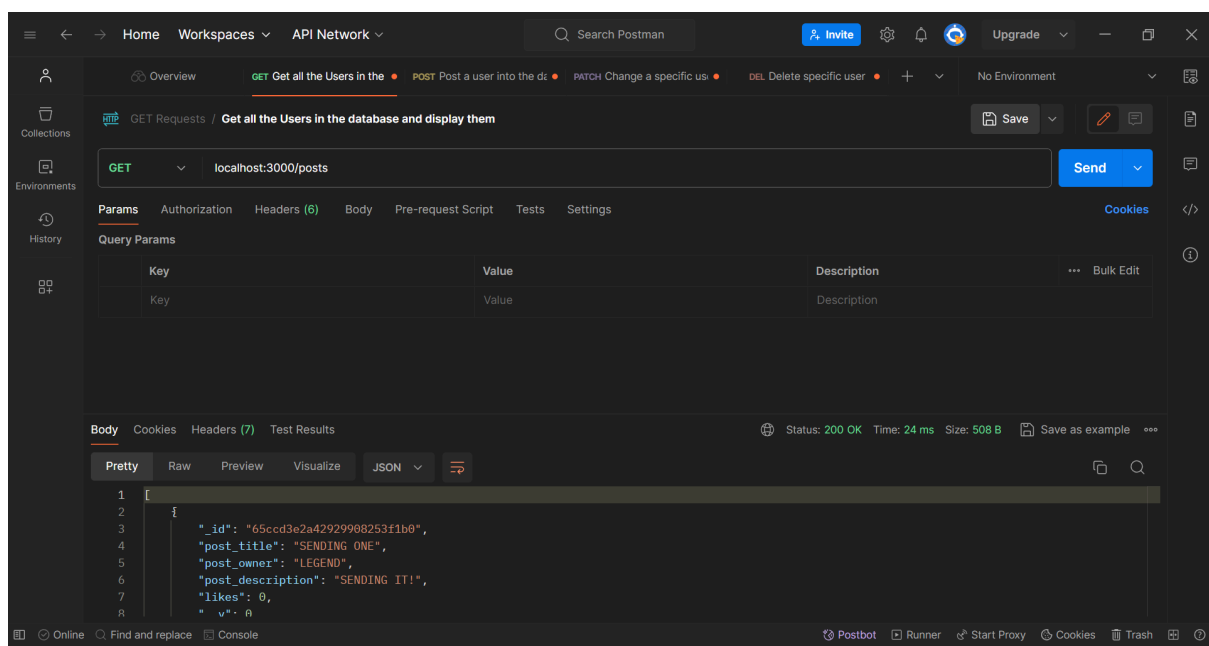
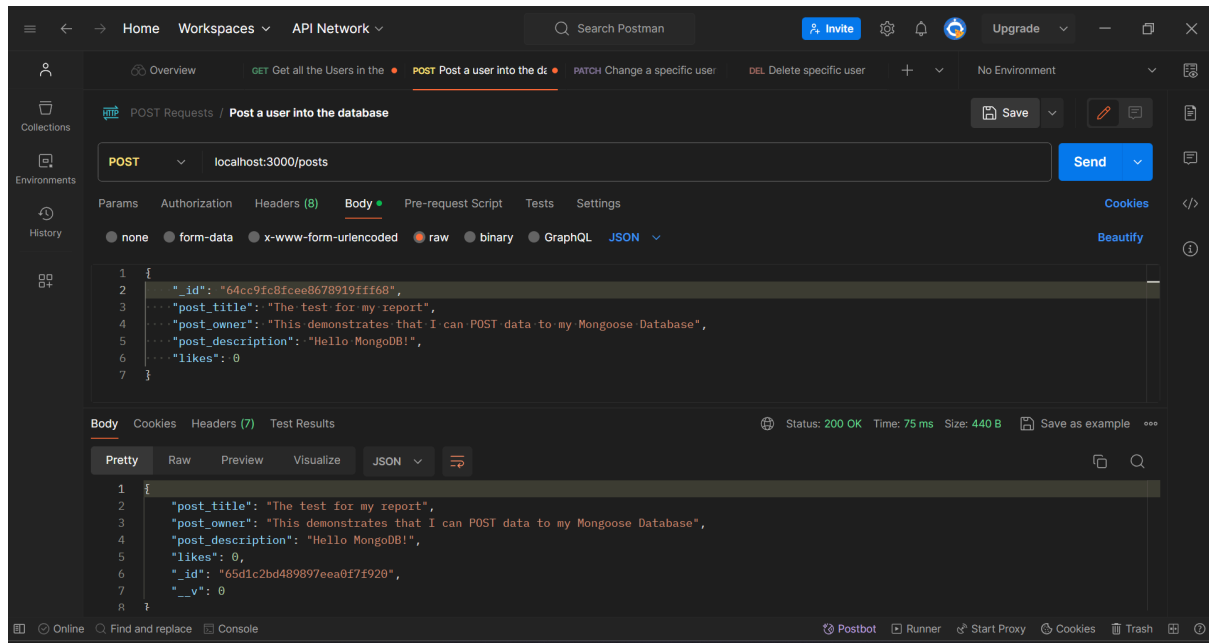
You can see my auth file below:

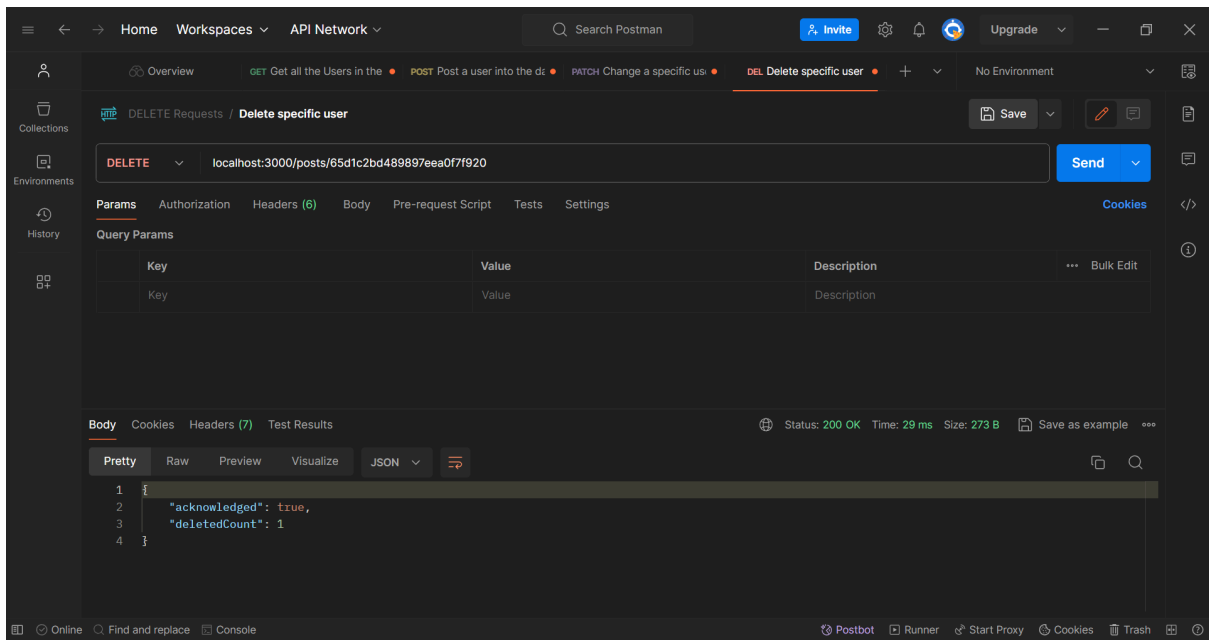
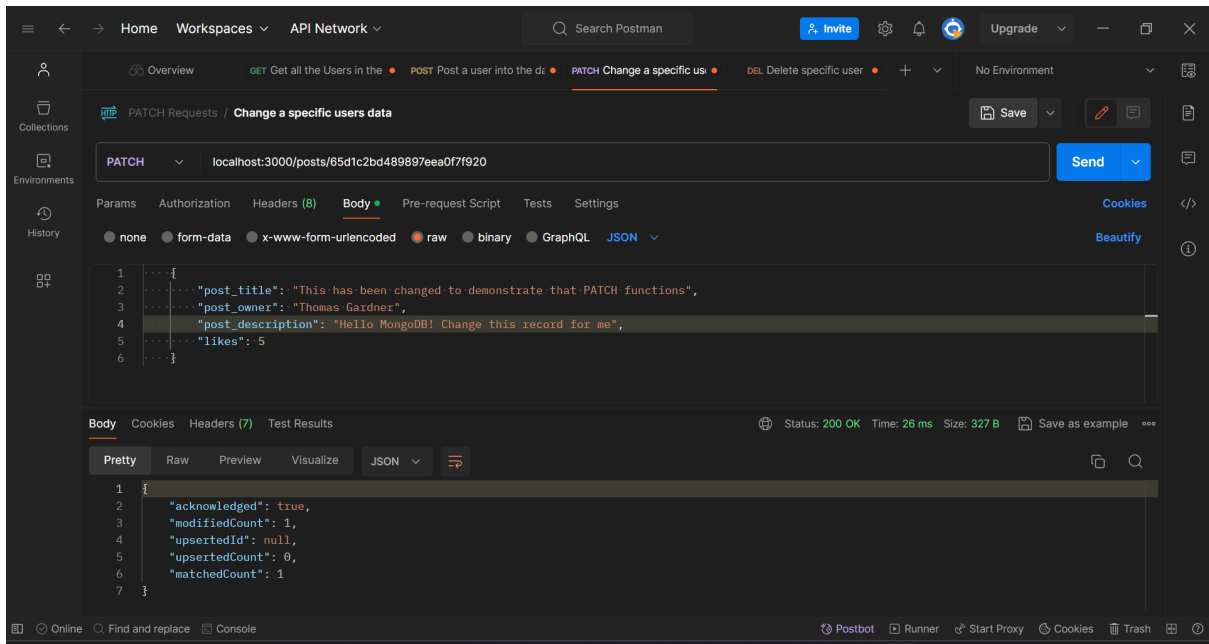


```
1 const express = require('express')
2 const router = express.Router()
3
4 const User = require('../models/User')
5 const {registerValidation, loginValidation} = require('../validations/validation')
6
7 const bcryptjs = require('bcryptjs')
8 const jwt = require('jsonwebtoken')
9
10 router.post('/register', async(req,res)=>{
11   const {error} = registerValidation(req.body)
12   if(error){
13     return res.status(400).send({message:error['details'][0]['message']})
14   }
15
16   const userExists = await User.findOne({email:req.body.email})
17   if (userExists){
18     return res.status(400).send({message:'User already exists. Try logging in with your email.'})
19   }
20
21   const salt = await bcryptjs.genSalt(10)
22   const hashedPassword = await bcryptjs.hash(req.body.password, salt)
23
24   const user = new User({
25     username:req.body.username,
26     email:req.body.email,
27     password:hashedPassword
28   })
29
30   try{
31     const savedUser = await user.save()
32     res.send(savedUser)
33   }catch(err){
34     res.status(400).send({message:err})
35   }
36 })
37
38 router.post('/login', async(req,res)=>{
39
40   const {error} = loginValidation(req.body)
41   if(error){
42     return res.status(400).send({message:error['details'][0]['message']})
43   }
44
45   const user = await User.findOne({email:req.body.email})
46   if (!user){
47     return res.status(400).send({message:'User does not exist. Try creating an account with your email.'})
48   }
49
50   const passwordValidation = await bcryptjs.compare(req.body.password, user.password)
51   if(!passwordValidation){
52     return res.status(400).send({message:'Incorrect password'})
53   }
54
55   //Here we generate an token that authorizes the user to access the data on the DB
56   const token = jwt.sign({id:user._id}, process.env.TOKEN_SECRET)
57   res.header('auth-token', token).send({auth-token:token})
58
59 })
60
61 //A quickly written bit of code to get all users currently in DB
62 router.get('/users', async(req,res)=>{
63   const allUsers = await User.find()
64   return res.send(allUsers)
65 })
66
67 module.exports = router
```

Phase C: Development of the MiniWall RESTful API

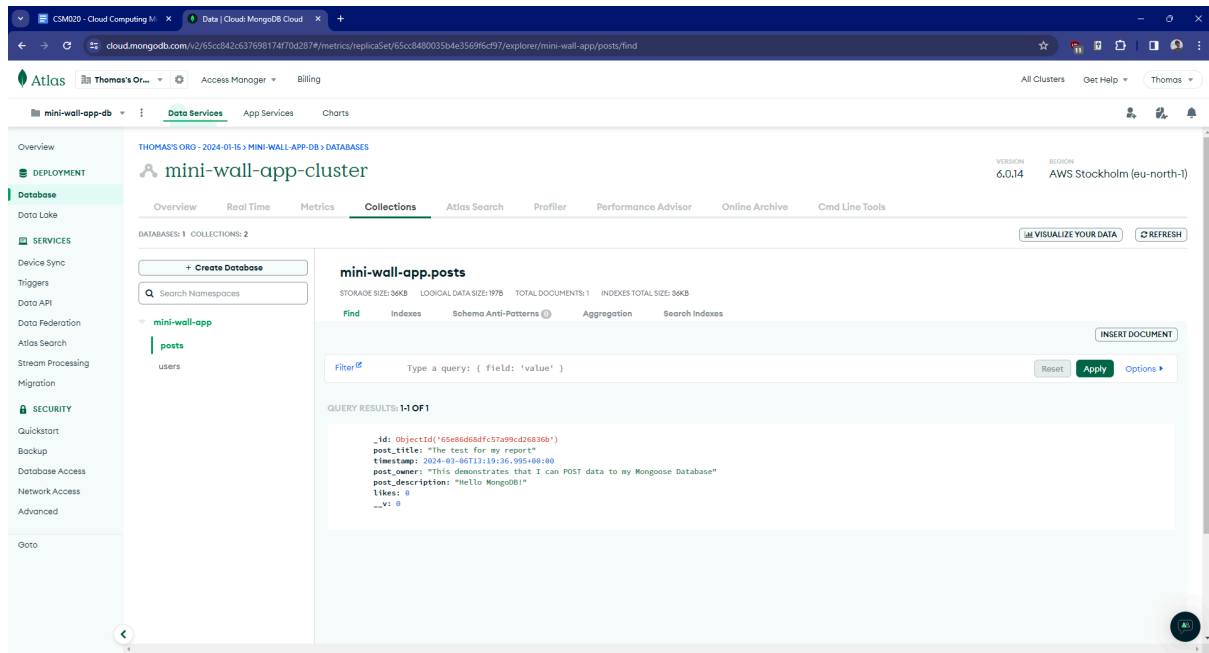
As demonstrated above in Phase A I have allowed for CRUD functionalities, but for completeness I will also provide screenshots here from Postman:





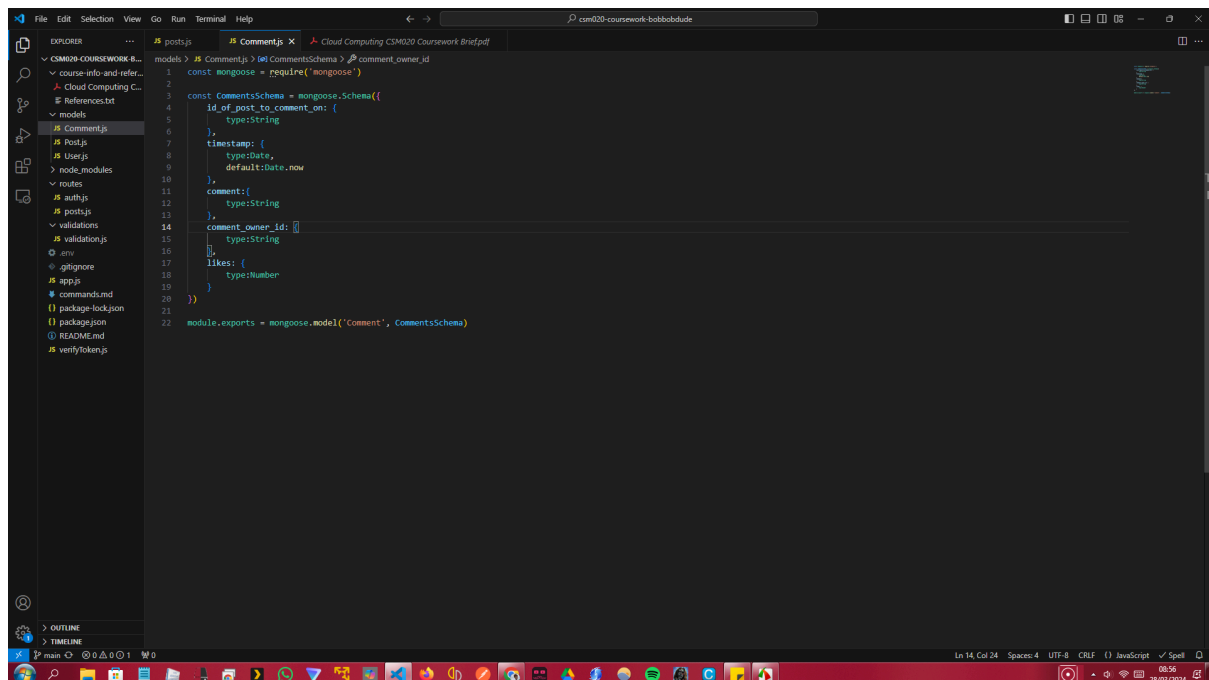
As you can see I can **Create** new posts and users, **Read** a list of all the posts or users, **Update** specific posts, and also **Delete** specific posts and users.

I also adapted the data schema of a post to include timestamp information as shown by this MongoDB screenshot:

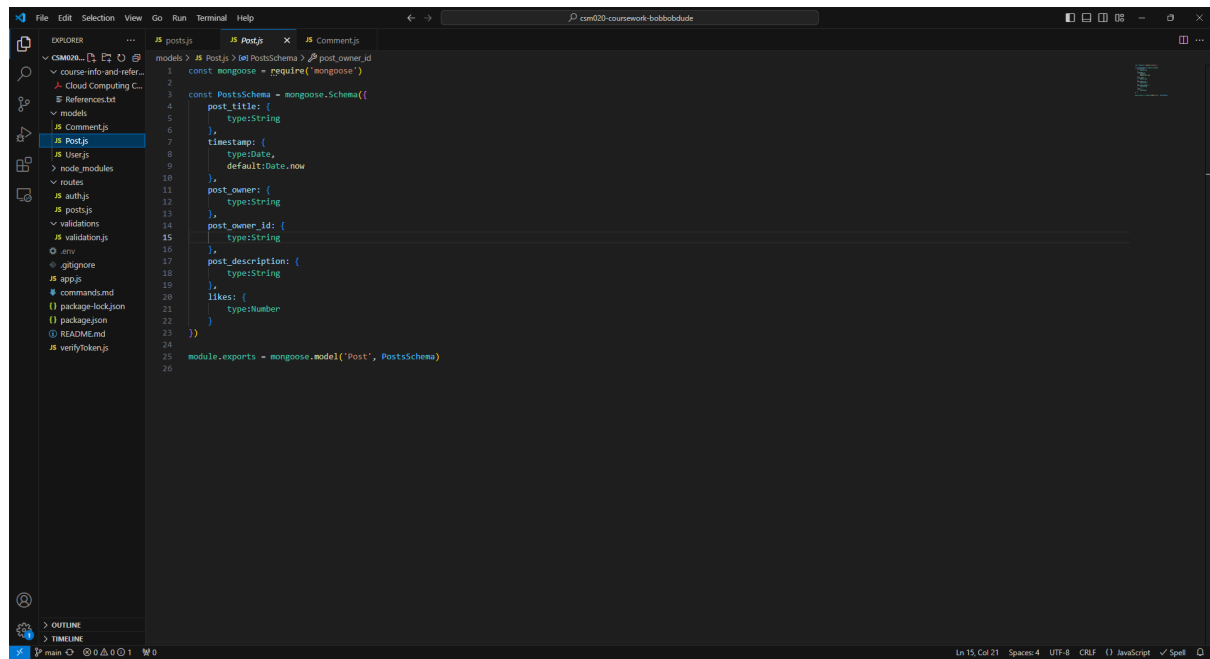


I will now demonstrate the data models used in my implementation.

Below I have my Comment.js data model which contains the original ID of the post to comment on stored as the data type “String”. This allows us to track which comment pertains to which original post. The other data types are self explanatory:

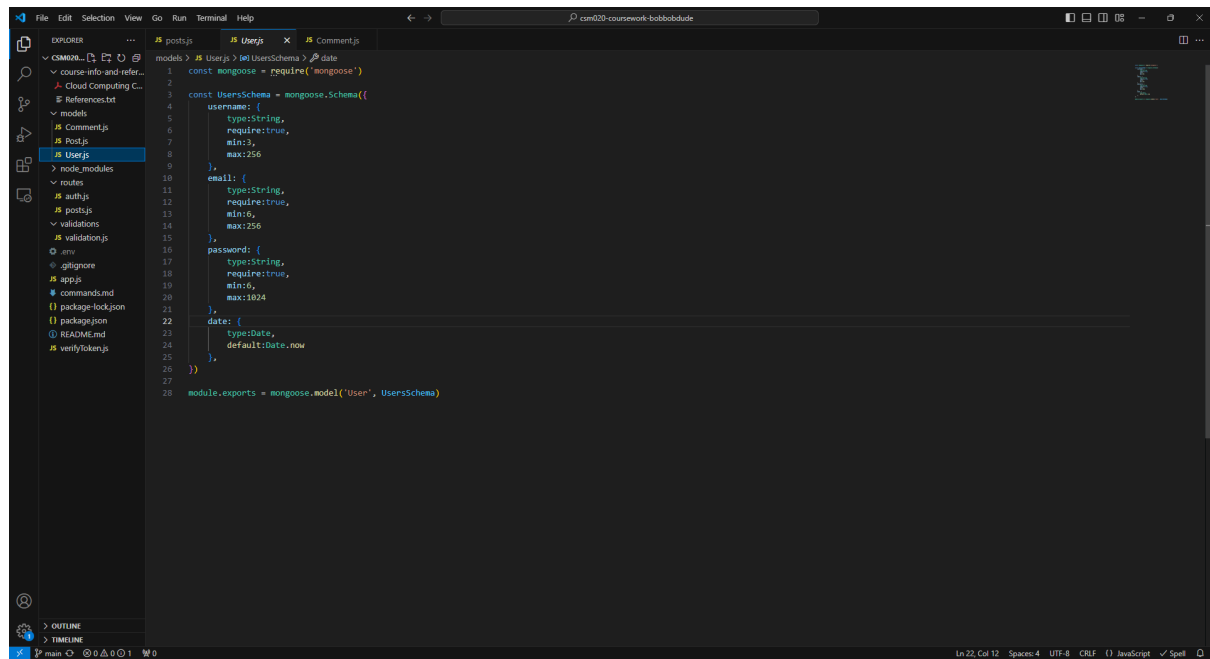


Here is my Post data structure:



```
1 const mongoose = require('mongoose')
2
3 const PostsSchema = mongoose.Schema({
4   post_title: {
5     type: String
6   },
7   timestamp: {
8     type: Date,
9     default: Date.now
10  },
11   post_owner: {
12     type: String
13   },
14   post_owner_id: {
15     type: String
16   },
17   post_description: {
18     type: String
19   },
20   likes: {
21     type: Number
22   }
23 })
24
25 module.exports = mongoose.model('Post', PostsSchema)
26
```

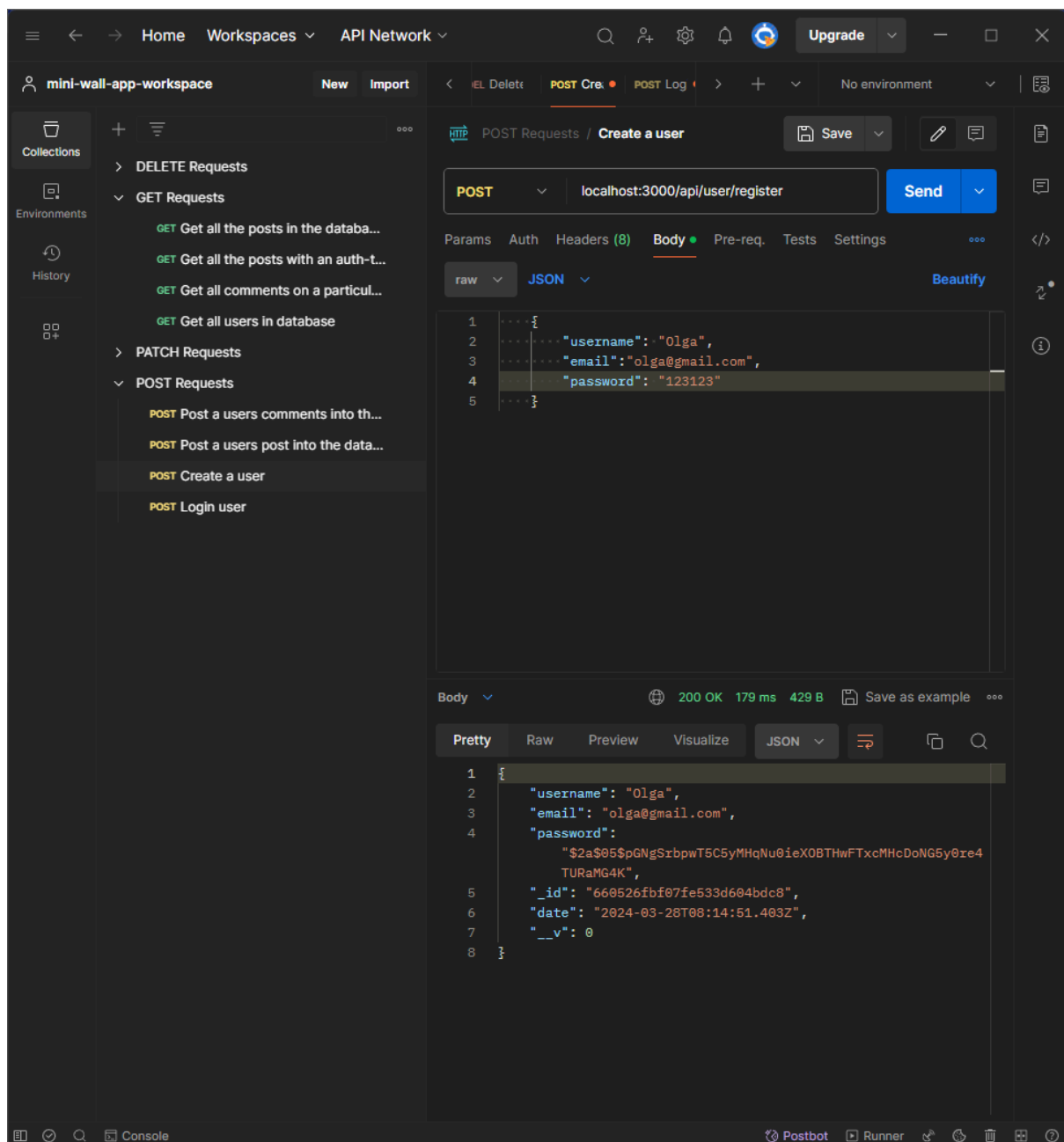
And finally my User data structure:



```
1 const mongoose = require('mongoose')
2
3 const UsersSchema = mongoose.Schema({
4   username: {
5     type: String,
6     required: true,
7     min: 3,
8     max: 256
9   },
10   email: {
11     type: String,
12     required: true,
13     min: 6,
14     max: 256
15   },
16   password: {
17     type: String,
18     required: true,
19     min: 6,
20     max: 1024
21   },
22   date: {
23     type: Date,
24     default: Date.now
25   }
26 })
27
28 module.exports = mongoose.model('User', UsersSchema)
29
```

Phase D: Development of the MiniWall testing cases

TC 1. Olga, Nick and Mary register in the application and access the API.



mini-wall-app-workspace

NewImport

Collections

DELETE Requests

GET Requests

POST Requests

PATCH Requests

POST Create a user

Login user

POST Requests / Create a user

POSTlocalhost:3000/api/user/register

Send

ParamsAuthHeaders (8)BodyPre-req. Tests Settings

rawJSON

1{2 "username": "Nick",3 "email": "nick@gmail.com",4 "password": "123123"5 }

Body200 OK62 ms429 BSave as example

PrettyRawPreviewVisualizeJSON

1{2 "username": "Nick",3 "email": "nick@gmail.com",4 "password": "\$2a\$06\$RL7/I3hdUD1NJwd3Jtt/Ru/0c8MKCp0Qjw0ys69ZiN/X/ef6bnGGW",5 "_id": "66062767f07fe533d604bdcb",6 "date": "2024-03-28T08:16:39.606Z",7 "__v": 08 }

ConsolePostbotRunner

mini-wall-app-workspace

NewImport

HTTP DELETE POST Create POST Log

No environment

Collections

DELETE Requests

GET Requests

GET Get all the posts in the databa...

GET Get all the posts with an auth-t...

GET Get all comments on a particul...

GET Get all users in database

PATCH Requests

POST Requests

POST Post a users comments into th...

POST Post a users post into the data...

POST Create a user

POST Login user

POST Requests / Create a user

POSTlocalhost:3000/api/user/register

Send

ParamsAuthHeaders (8)BodyPre-req. Tests Settings

rawJSON

1{2"username": "Mary",3"email": "mary@gmail.com",4"password": "123123"5}

Body200 OK 47 ms 429 BSave as example

PrettyRawPreviewVisualizeJSON

1{2"username": "Mary",3"email": "mary@gmail.com",4"password":5"\$2a\$05\$7bdqWRcUPJouJfckCSZ5RuW3Ysh7GxzPhDyPW2V5PG66MKZUmESiy",6"_id": "66052792f07fe533d604bdce",7"date": "2024-03-28T08:17:22.413Z",8"__v": 0}

Console

PostbotRunner

Google Docs Online Document | CSM20 - Cloud Computing M | Data | Cloud MongoDB Cloud | Extract Text from an Image | O | X | +

cloud.mongodb.com/v2/65cc842c637698174f70d287f/metrics/replicaSet/65cc840035b4e35696c977/explorer/mini-wall-app/users/find

Bookings | MS Comp Sci UOL | Lego | Torrent Stuff | Friends 1994 free st... | Friends 1994 free st... | FMovies | Watch M... | Watch Movies And... | Dhokla.net - Stream... | T2chillito | Proteus | Interactive Investor | Codewars - Achieve... | Alla bokmärken

Atlas | Thomas's Or... | Access Manager | Billing | All Clusters | Get Help | Thomas

mini-wall-app-db | Data Services | App Services | Charts

Overview | DEPLOYMENT | Database | Data Lake | SERVICES | Device Sync | Triggers | Data API | Data Federation | Atlas Search | Stream Processing | Migration | SECURITY | Quickstart | Backup | Database Access | Network Access | Advanced | Ooto

+ Create Database | Search Namespaces | mini-wall-app | comments | posts | users

mini-wall-app.users | STORAGE SIZE: 36KB | LOGICAL DATA SIZE: 648B | TOTAL DOCUMENTS: 3 | INDEXES TOTAL SIZE: 36KB | Find | Indexes | Schema Anti-Patterns | Aggregation | Search Indexes

Filter | Type a query: { field: 'value' } | Reset | Apply | Options

INSERT DOCUMENT

```
{
  "_id": ObjectId("669526fb97fe53d694bdc8"),
  "username": "oliga",
  "email": "oliga@gmail.com",
  "password": "52a5955p2G5r5pw75CSyW9hRubiex0EThwTxcMRcDoN5y8re4TURuRGAK",
  "date": 2024-03-28T08:14:15Z,
  "age": 40,
  "v": 0
}
```

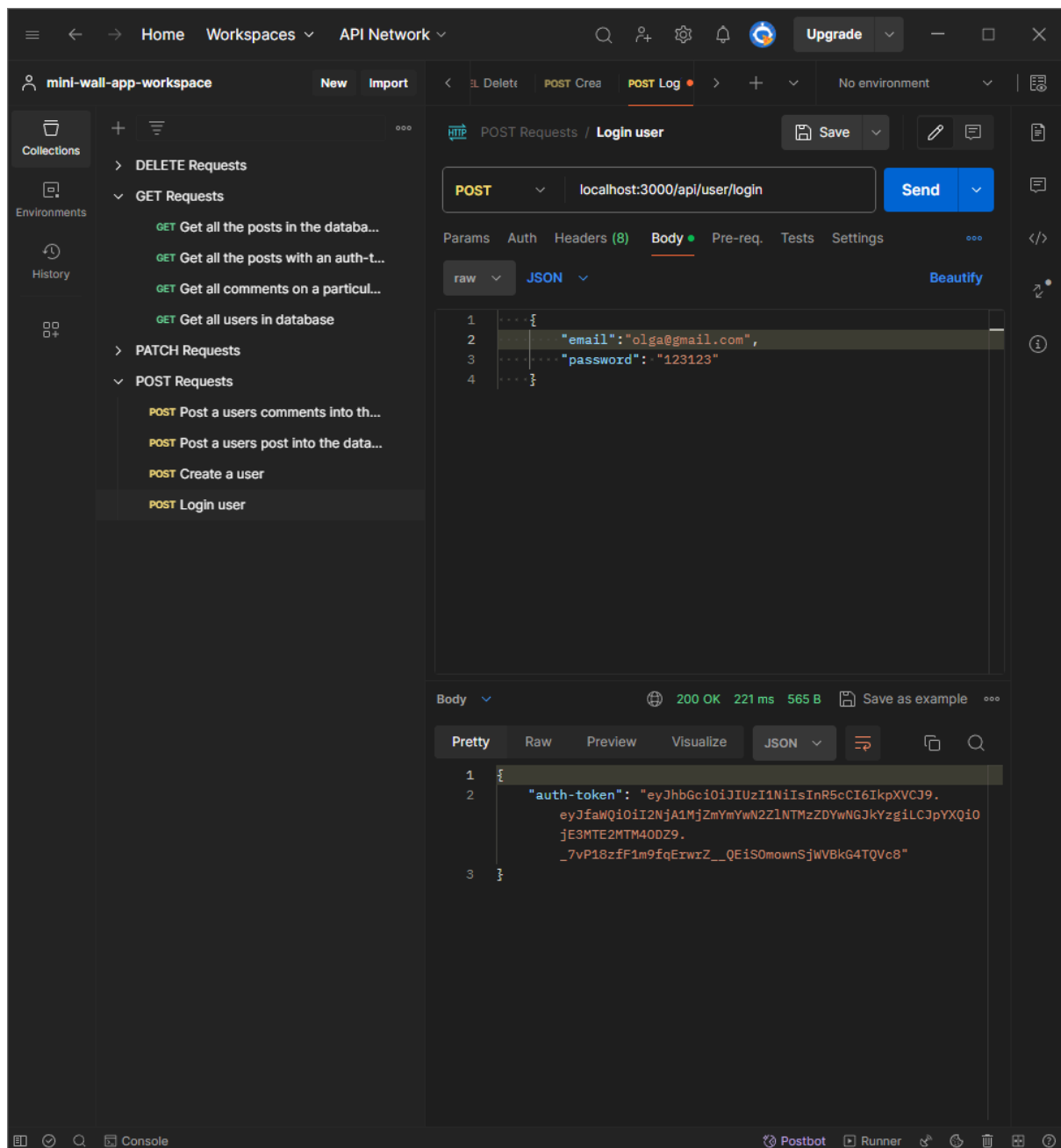
```
{
  "_id": ObjectId("66952767f97fe53d694bdc9"),
  "username": "Blick",
  "email": "nlic@gmail.com",
  "password": "52a5955817713hd0Lkhd13t1/Ru/Oc8MKCp0Qjw0ys69Z1N/X/ef6bnGM",
  "date": 2024-03-28T08:16:39Z,
  "age": 40,
  "v": 0
}
```

```
{
  "_id": ObjectId("66952792f97fe53d694bdce"),
  "username": "Mory",
  "email": "mory@gmail.com",
  "password": "52a59557bdc0P3ou2FkcCSZSRuW3Ygh7GxzPhdyPM2VSPG66MKZueE5y",
  "date": 2024-03-28T08:17:22Z,
  "age": 40,
  "v": 0
}
```

System Status: All Good
©2024 MongoDB, Inc. | Status | Terms | Privacy | Atlas Blog | Contact Sales

https://cloud.mongodb.com/v2/65cc842c637698174f70d287f/migration

TC 2. Olga, Nick and Mary will use the oAuth v2 authorisation service to get their tokens.



mini-wall-app-workspace

NewImport

Collections

DELETE Requests

GET Requests

GET Get all the posts in the databa...

GET Get all the posts with an auth-t...

GET Get all comments on a particul...

GET Get all users in database

PATCH Requests

POST Requests

POST Post a users comments into th...

POST Post a users post into the data...

POST Create a user

POST Login user

POST Requests / Login user

POSTlocalhost:3000/api/user/login

Send

ParamsAuthHeaders (8)BodyPre-req. Tests Settings

rawJSON

1{2"email": "nick@gmail.com",3"password": "123123"4}

Body200 OK36 ms565 BSave as example

PrettyRawPreviewVisualizeJSON

1{2"auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2NjA1Mjc2N2YwN2ZlNTMzZDYwNGJkY2IiLCJpYXQiOiIjE3MTE2MTM6MDJ9.eyJXssUsILF_H4zH2NzpC7cmccA0oZ04a1ThwxRP9zFU8"3}

ConsolePostbotRunner

mini-wall-app-workspace

NewImport

Collections

DELETE Requests

GET Requests

GET Get all the posts in the databa...

GET Get all the posts with an auth-t...

GET Get all comments on a particul...

GET Get all users in database

PATCH Requests

POST Requests

POST Post a users comments into th...

POST Post a users post into the data...

POST Create a user

POST Login user

POST Requests / Login user

POSTlocalhost:3000/api/user/login

Send

ParamsAuthHeaders (8)BodyPre-req. Tests Settings

rawJSON

1{2"email": "mary@gmail.com",3"password": "123123"4}

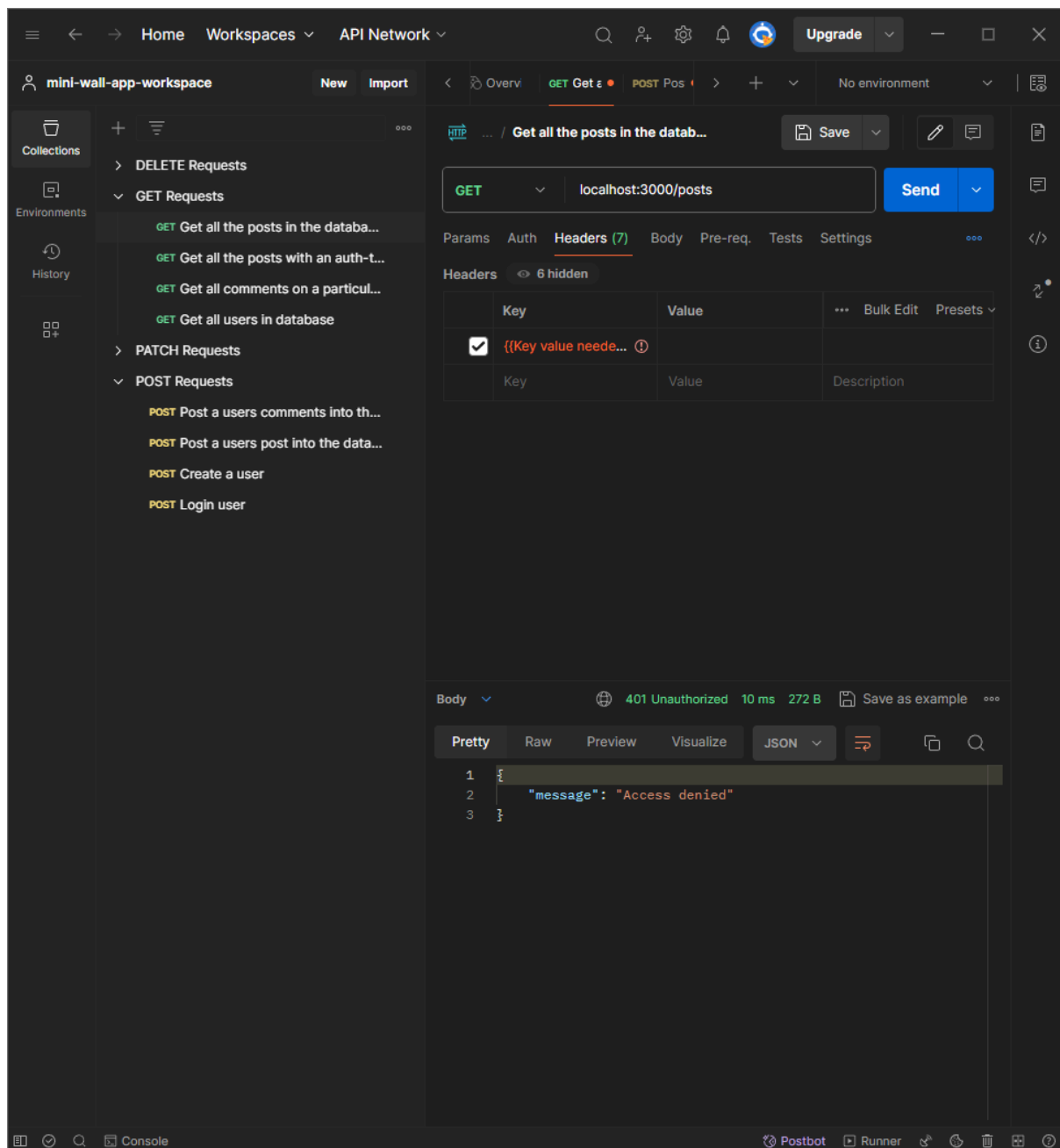
Body200 OK 24 ms 565 BSave as example

PrettyRawPreviewVisualizeJSON

1{2"auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2NjA1Mjc5MmYwN2ZlNTMzZDYwNGJkY2UiLCJpYXQiOiIyMTE2MTM5MTh9.DsZT0_sCq9u4zb1YohiDxvWw5Fgdv7l55YIukH06NYI"3}

ConsolePostbotRunner

TC 3. Olga calls the API (any endpoint) without using a token.
This call should be unsuccessful as the user is unauthorised.

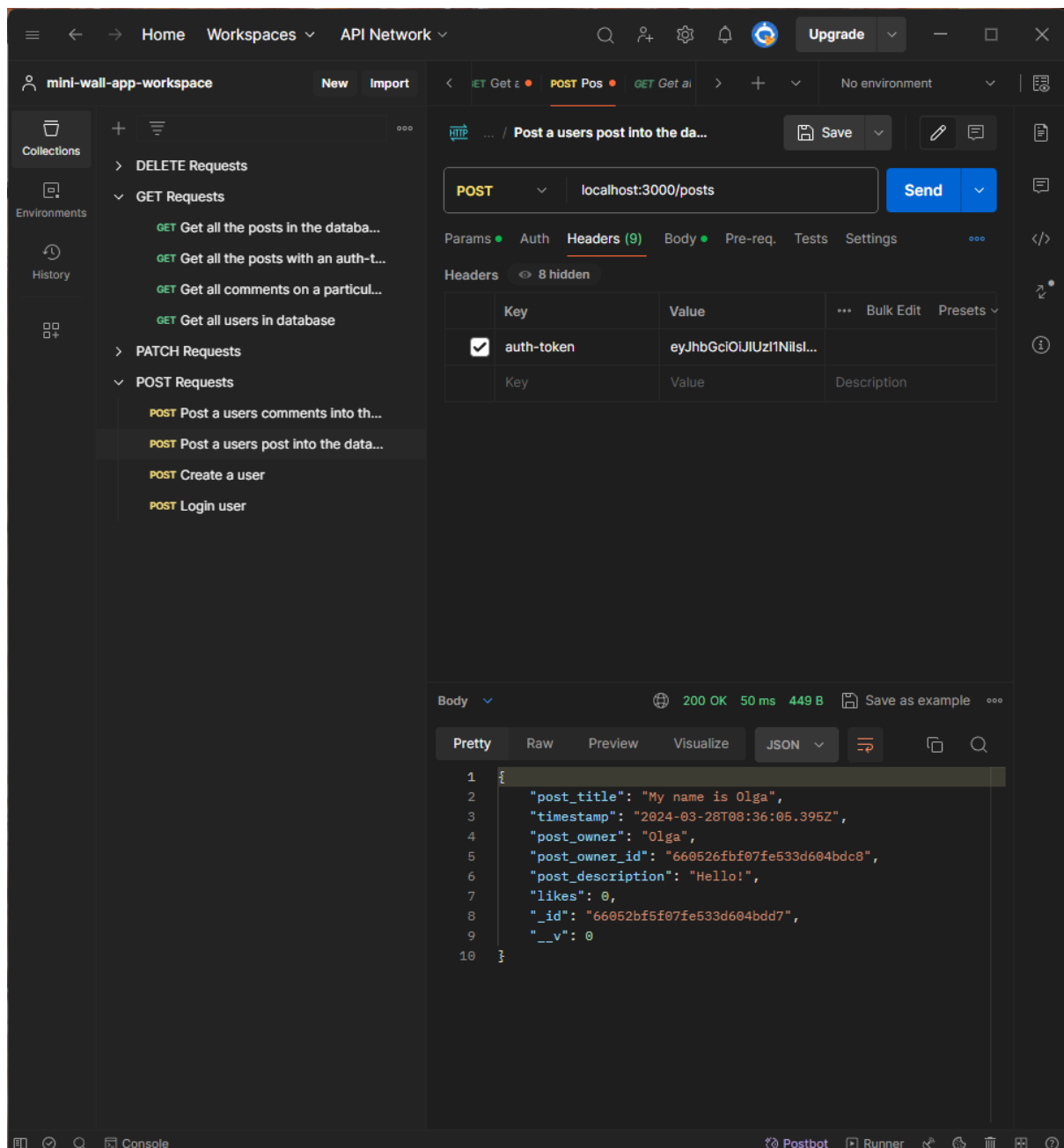


TC 4. Olga posts a text using her token.

I had to regenerate the auth tokens for this due to not saving the ones above:

Olga auth token -

```
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiJ1IiwiaWF0IjoiMTY5MjYwN2ZINTMzZDYwNjYyZgiLCJpYXQiOiJlbnR5cCI6IkpXVCJ9.XHnJ_jzB8ufHYAsnXfu3_TGsQEnl6muSOFDgVQ3kplU"
```

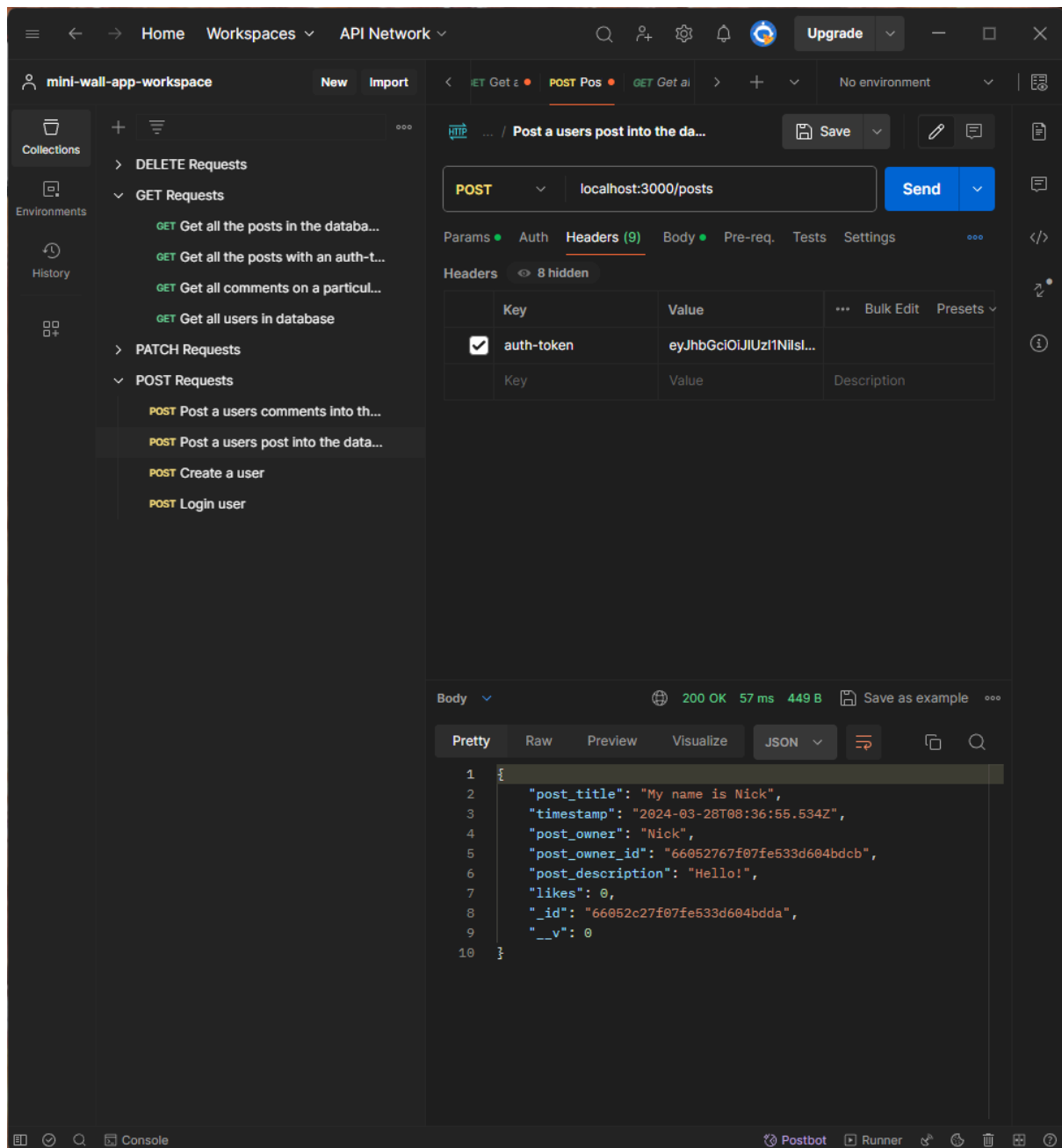


TC 5. Nick posts a text using his token.

Nick auth token -

```
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiJ1IiwiaWF0IjoiMTY5MjYwN2ZINTMzZDYwNjYyZgiLCJpYXQiOiJlbnR5cCI6IkpXVCJ9.XHnJ_jzB8ufHYAsnXfu3_TGsQEnl6muSOFDgVQ3kplU"
```

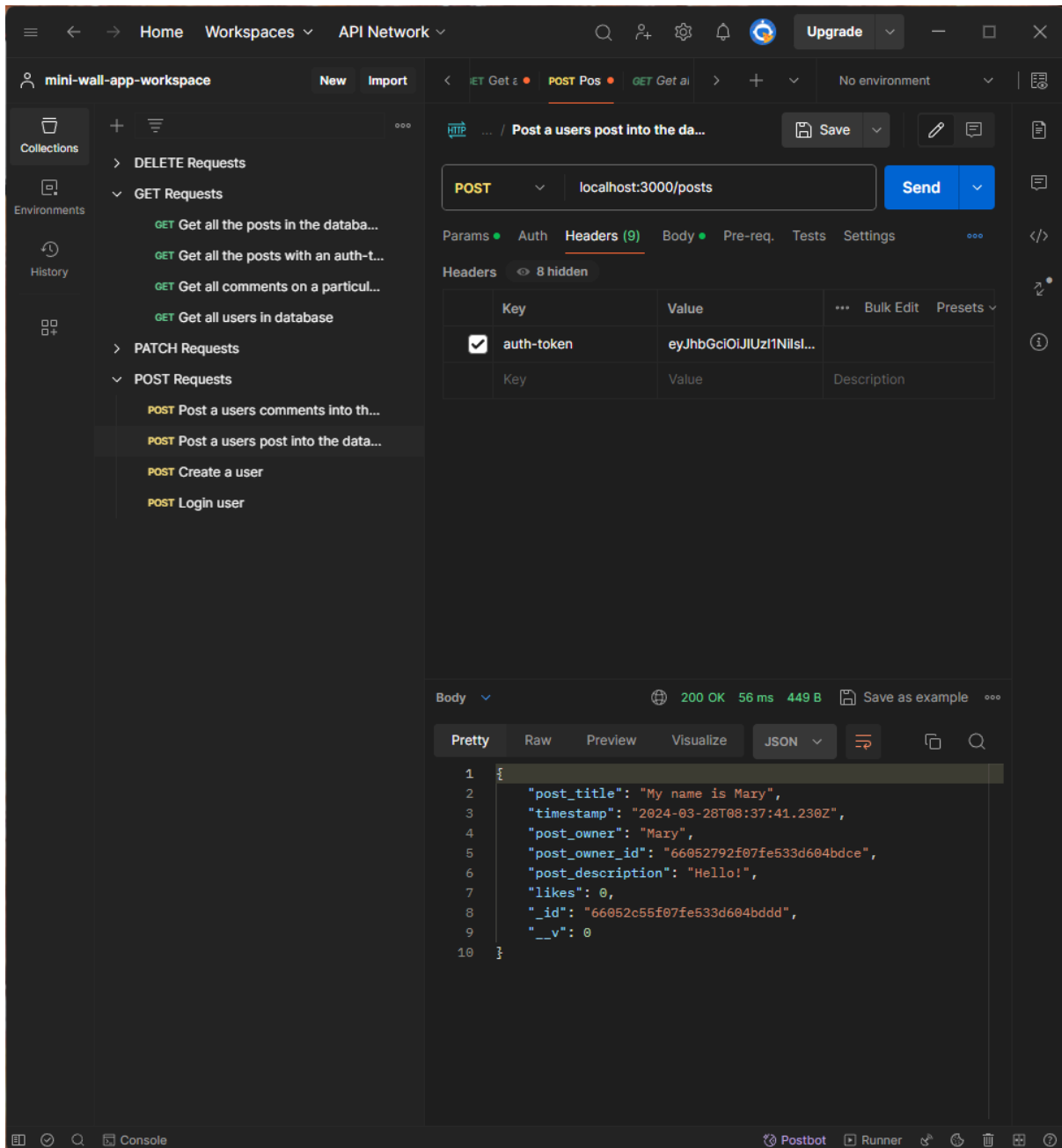
GJKY2IiLCJpYXQiOiE3MTE2MTQyMjN9.O2Q3xw9lj7ZWEobdhc3Pa0vhi4xuEy3LWqorJtyfP
Wk"



TC 6. Mary posts a text using her token.

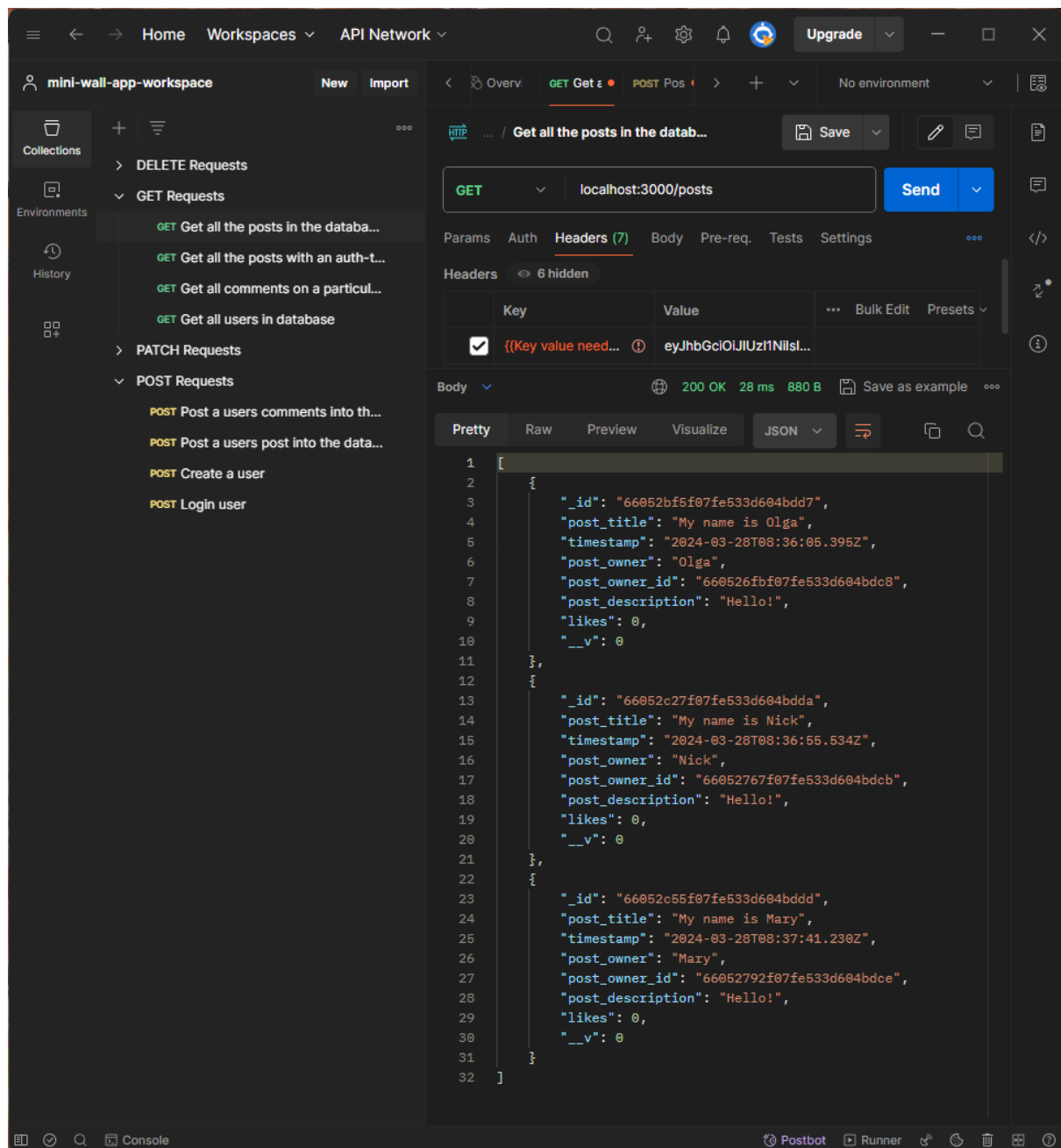
Mary auth token -

"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2NjA1Mjc5MmYwN2ZlNTMzZDYwNjE3MTE2MTQyMjN9.lew9ljza1215F3e-F_Zlc2nDp8ZRbweDEbA2gkl0
_1M"



TC 7. Nick and Olga browse available posts in chronological order in the MiniWall; there should be three posts available. Note that we do not have any likes yet.

Browsing with Olga's auth token:



Browsing with Nick's auth token:

mini-wall-app-workspace

NewImport

Collections

Environments

History

DELETE Requests

GET Requests

PATCH Requests

POST Requests

GET Get all the posts in the databa...

GET Get all the posts with an auth-t...

GET Get all comments on a particul...

GET Get all users in database

POST Post a users comments into th...

POST Post a users post into the data...

POST Create a user

POST Login user

Overview

GET Get all the posts in the databa...

POST Post a users post into the data...

localhost:3000/posts

Send

Params

Auth

Headers (7)

Body

Pre-req.

Tests

Settings

Key	Value	Description
<input checked="" type="checkbox"/>	{(Key value need...}	eyJhbGciOiJIUzI1NiIs...

Body

200 OK20 ms880 B

Save as example

Pretty

Raw

Preview

Visualize

JSON

```
1  [
2    {
3      "_id": "66052bf5f07fe533d604bdd7",
4      "post_title": "My name is Olga",
5      "timestamp": "2024-03-28T08:36:05.395Z",
6      "post_owner": "Olga",
7      "post_owner_id": "660526fbf07fe533d604bdc8",
8      "post_description": "Hello!",
9      "likes": 0,
10     "__v": 0
11   },
12   {
13     "_id": "66052c27f07fe533d604bdda",
14     "post_title": "My name is Nick",
15     "timestamp": "2024-03-28T08:36:55.534Z",
16     "post_owner": "Nick",
17     "post_owner_id": "66052767f07fe533d604bdc8",
18     "post_description": "Hello!",
19     "likes": 0,
20     "__v": 0
21   },
22   {
23     "_id": "66052c55f07fe533d604bddb",
24     "post_title": "My name is Mary",
25     "timestamp": "2024-03-28T08:37:41.230Z",
26     "post_owner": "Mary",
27     "post_owner_id": "66052792f07fe533d604bdce",
28     "post_description": "Hello!",
29     "likes": 0,
30     "__v": 0
31   }
32 ]
```

Console

Postbot

Runner

TC 8. Nick and Olga comment on Mary's post in a round robin fashion (one after the other).

Nick then Olga comments on Mary's post:

The screenshot shows the Postman interface with a workspace named "mini-wall-app-workspace". The left sidebar displays a collection of requests under "POST Requests", with "Post a users comments into th..." selected. The main panel shows the details of this request:

- Method:** POST
- URL:** localhost:3000/posts/66052c55f07fe533d6...
- Headers:** 10 headers are listed, including a key-value pair: `{{Key value neede...}}` with value `eyJhbGciOiJIUz11NiIsI...`.
- Body:** The response is displayed in JSON format, showing a successful creation of a comment.

The response body is as follows:

```
1 {
2   "id_of_post_to_comment_on": "66052c55f07fe533d604bddd",
3   "timestamp": "2024-03-28T09:20:56.048Z",
4   "comment": "Nick's comment on Mary's post",
5   "comment_owner_id": "66052767f07fe533d604bdcb",
6   "likes": 0,
7   "_id": "66053678f07fe533d604bde3",
8   "__v": 0
9 }
```

The status bar at the bottom indicates a 200 OK response with a response time of 307 ms and a body size of 469 B.

mini-wall-app-workspace

NewImport

Collections

Environments

History

DELETE Requests

GET Requests

PATCH Requests

POST Requests

POST Post a users comments into th...

POST Post a users post into the data...

POST Create a user

POST Login user

POST localhost:3000/posts/66052c55f07fe533d6...

Send

ParamsAuthHeaders (10)BodyPre-req. Tests Settings

Headers 8 hidden

	Key	Value	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	{(Key value neede...	eyJhbGciOiJIUzI1NiIs...			
<input type="checkbox"/>					
	Key	Value	Description		

Body 200 OK 298 ms 469 B Save as example

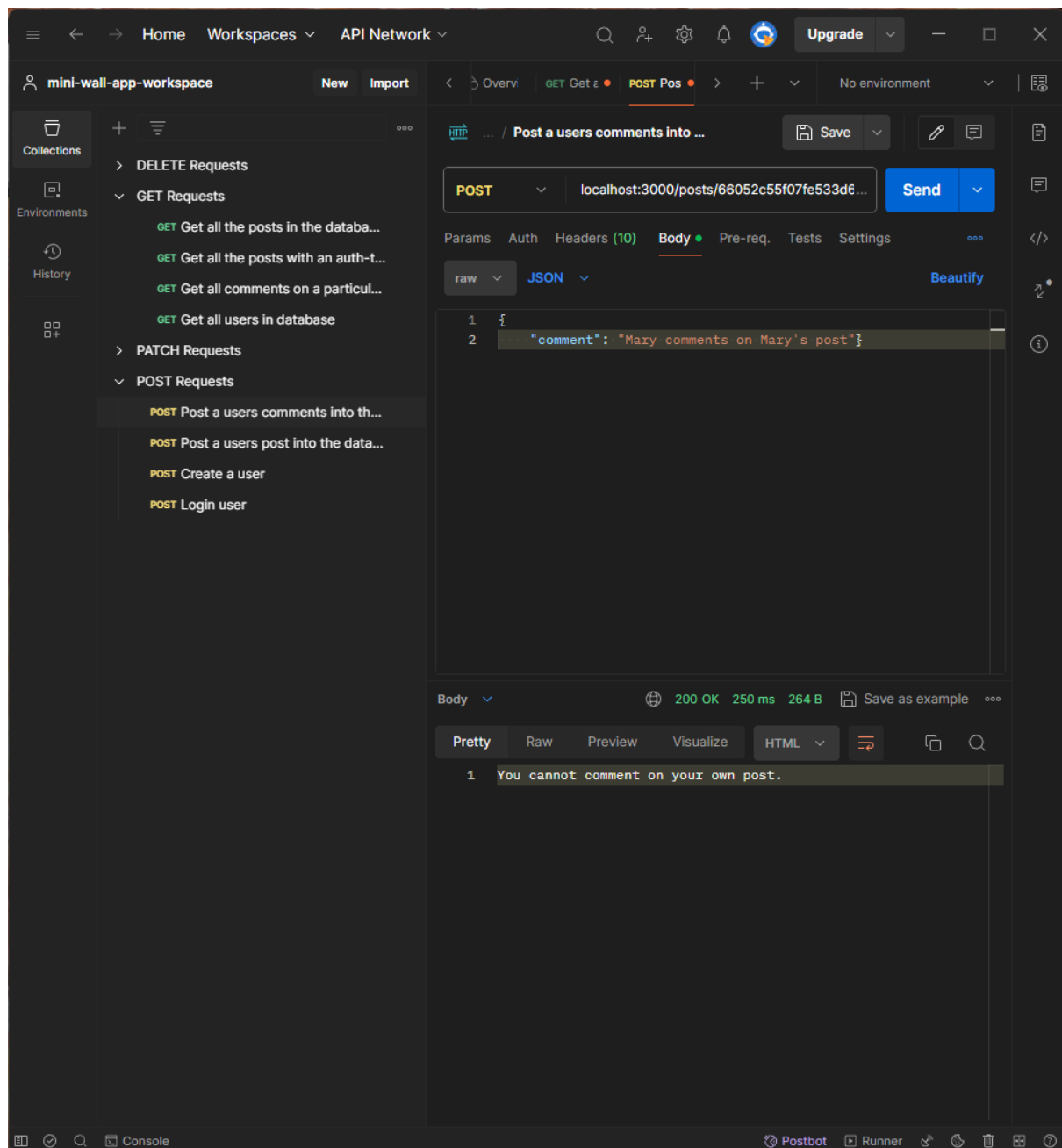
PrettyRawPreviewVisualizeJSON

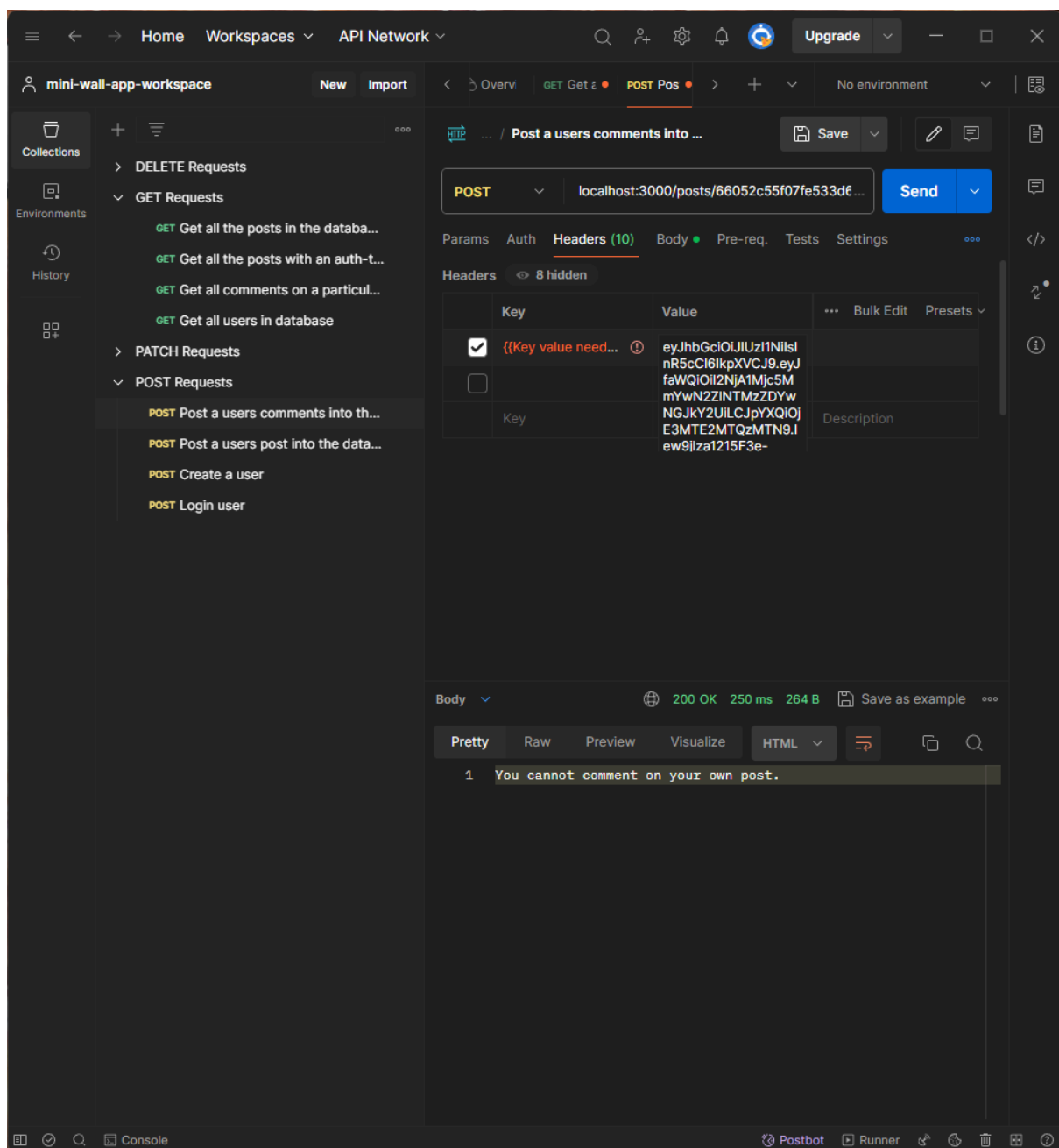
```
1 {
2   "id_of_post_to_comment_on": "66052c55f07fe533d604bddd",
3   "timestamp": "2024-03-28T09:24:43.935Z",
4   "comment": "Olga's comment on Mary's post",
5   "comment_owner_id": "660526fbf07fe533d604bdc8",
6   "likes": 0,
7   "_id": "6605375bf07fe533d604bde7",
8   "__v": 0
9 }
```

Console

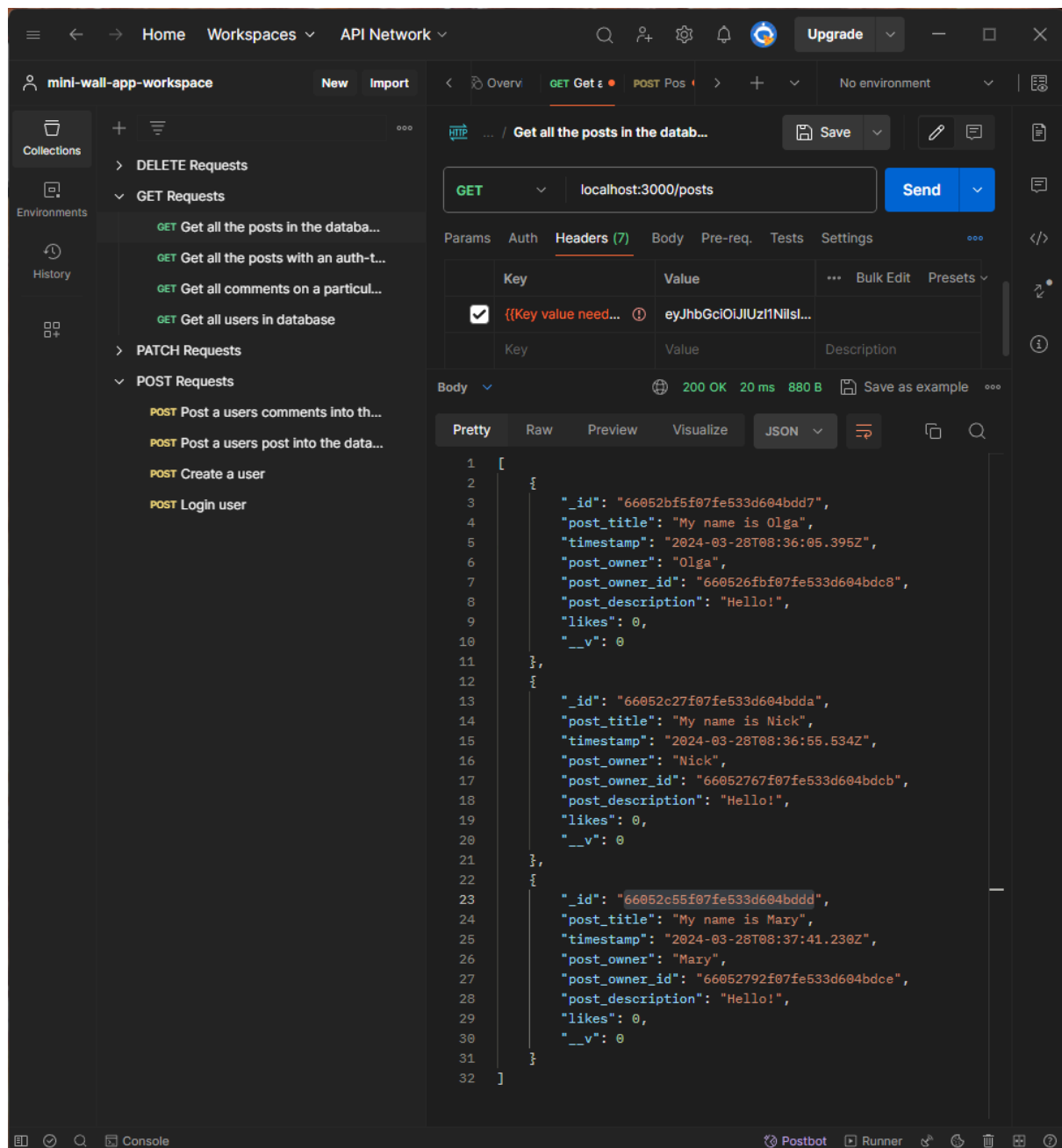
PostbotRunner

TC 9. Mary comments on her post. This call should be unsuccessful; an owner cannot comment on owned posts.

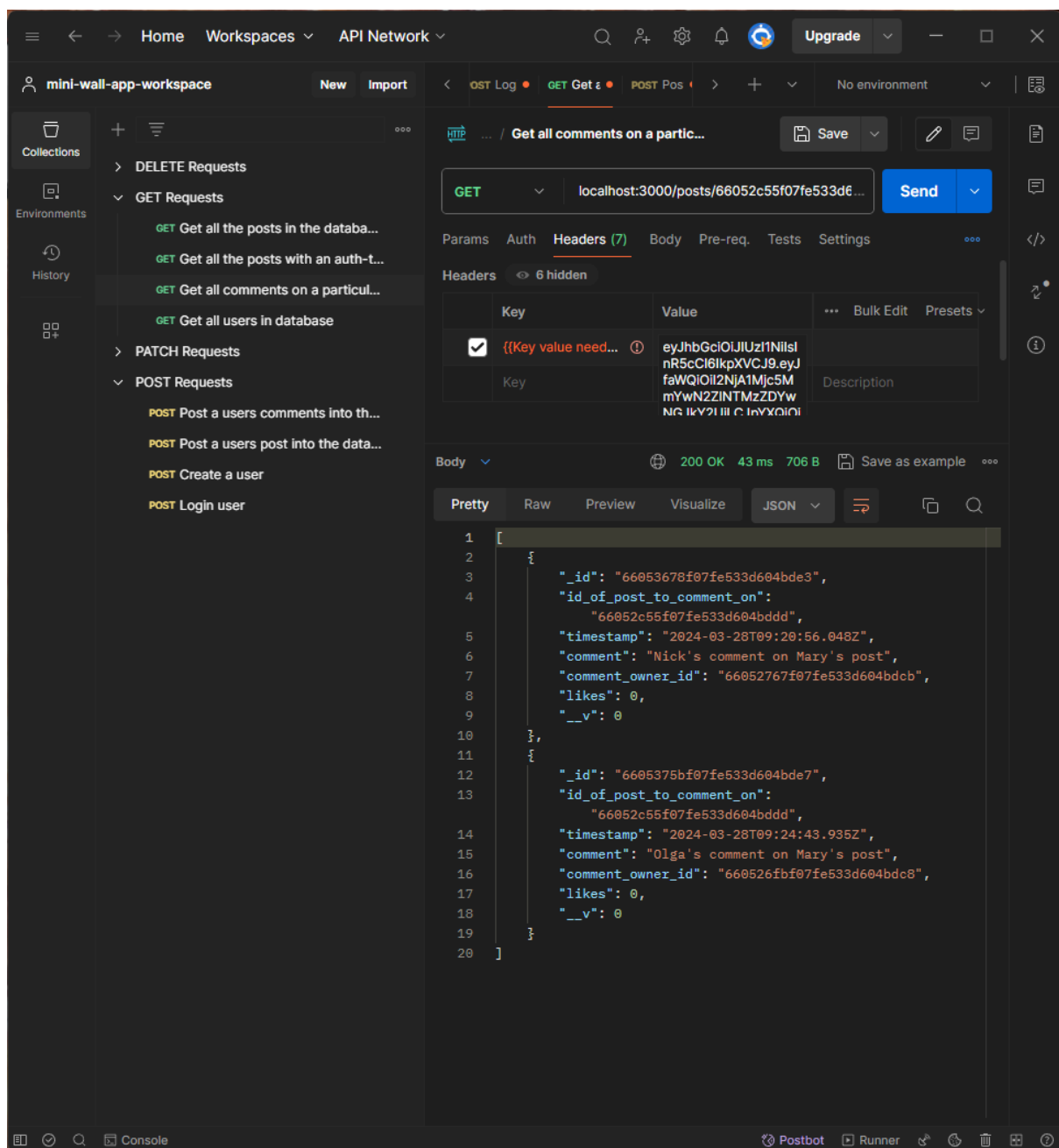




TC 10. Mary can see posts in a chronological order (newest posts are on the top as there are no likes yet).



TC 11. Mary can see the comments for her posts.



TC 12. Nick and Olga like Mary's posts.

Olga likes Mary's post and then Nick:

mini-wall-app-workspace

NewImport

Collections

Environments

History

DELETE Requests

GET Requests

PATCH Requests

POST Requests

PATCH Requests / Like a post

localhost:3000/posts/66052c55f07fe533d6...

Send

ParamsAuthHeaders (8)BodyPre-req. TestsSettings

Headers7 hidden

	Key	Value	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	{(Key value neede...	eyJhbGciOiJIUz11NiIs...			
	Key	Value		Description	

Body

200 OK58 ms529 B

Save as example

PrettyRawPreviewVisualizeJSON

```
1 {
2   "_id": "66052c55f07fe533d604bddd",
3   "post_title": "My name is Mary",
4   "timestamp": "2024-03-28T08:37:41.230Z",
5   "post_owner": "Mary",
6   "post_owner_id": "66052792f07fe533d604bdce",
7   "post_description": "Hello!",
8   "likes": 2,
9   "__v": 2,
10  "users_who_have_liked": [
11    "66052767f07fe533d604bdcb",
12    "660526fbf07fe533d604bdc8"
13  ]
14 }
```

PostbotRunner

mini-wall-app-workspace

NewImport

Collections

Environments

History

DELETE Requests

GET Requests

PATCH Requests

POST Requests

PATCH Requests / Like a post

localhost:3000/posts/66052c55f07fe533d6...

Send

ParamsAuthHeaders (8)BodyPre-req. TestsSettings

Headers 7 hidden

	Key	Value	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	{(Key value neede...	eyJhbGciOiJIUzI1NiIs...			
	Key	Value		Description	

Body 200 OK 118 ms 502 B Save as example

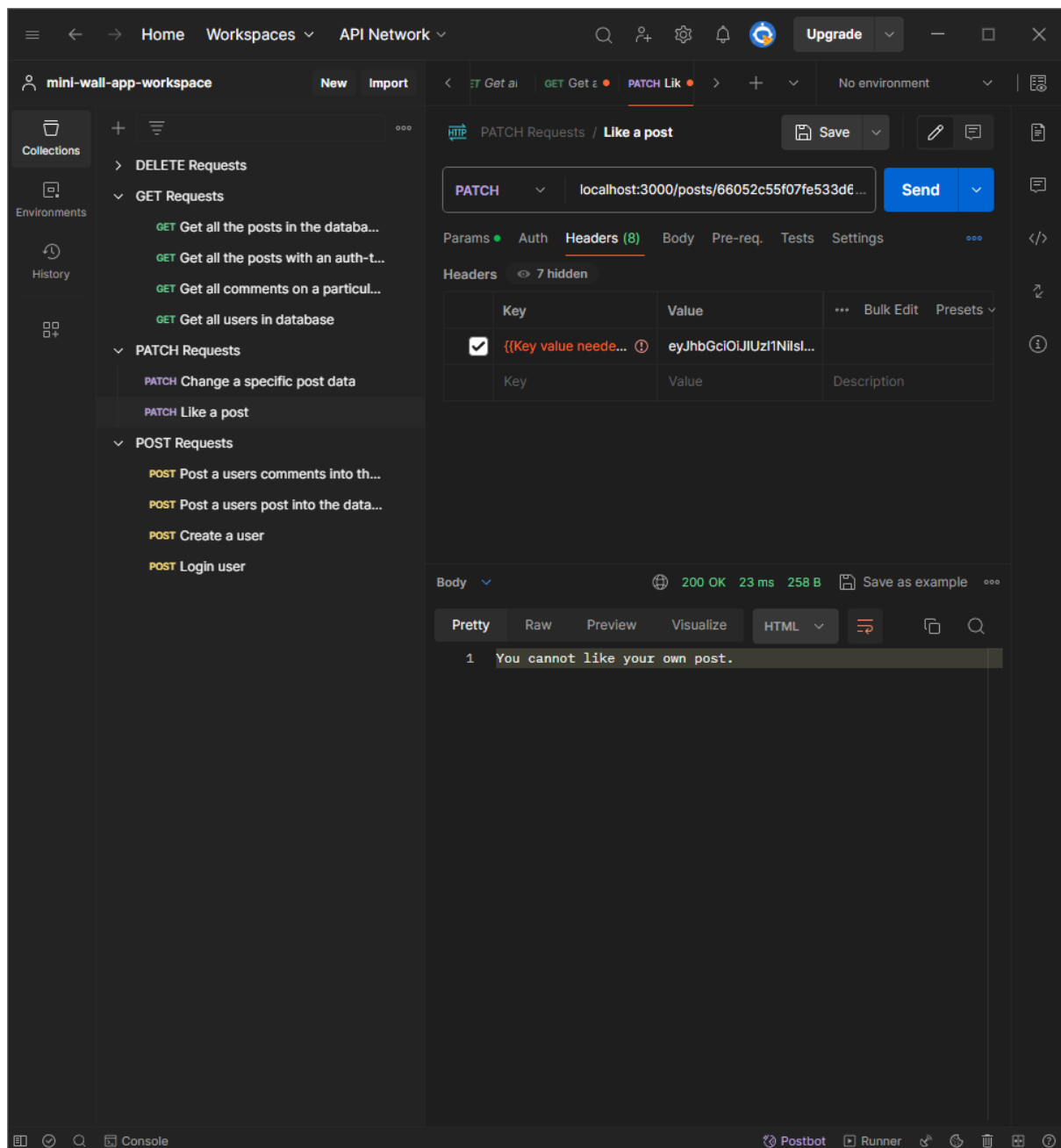
PrettyRawPreviewVisualizeJSON

```
1 {
2   "users_who_have_liked": [
3     "66052767f07fe533d604bdc"
4   ],
5   "_id": "66052c55f07fe533d604bddd",
6   "post_title": "My name is Mary",
7   "timestamp": "2024-03-28T08:37:41.230Z",
8   "post_owner": "Mary",
9   "post_owner_id": "66052792f07fe533d604bdce",
10  "post_description": "Hello!",
11  "likes": 1,
12  "__v": 1
13 }
```

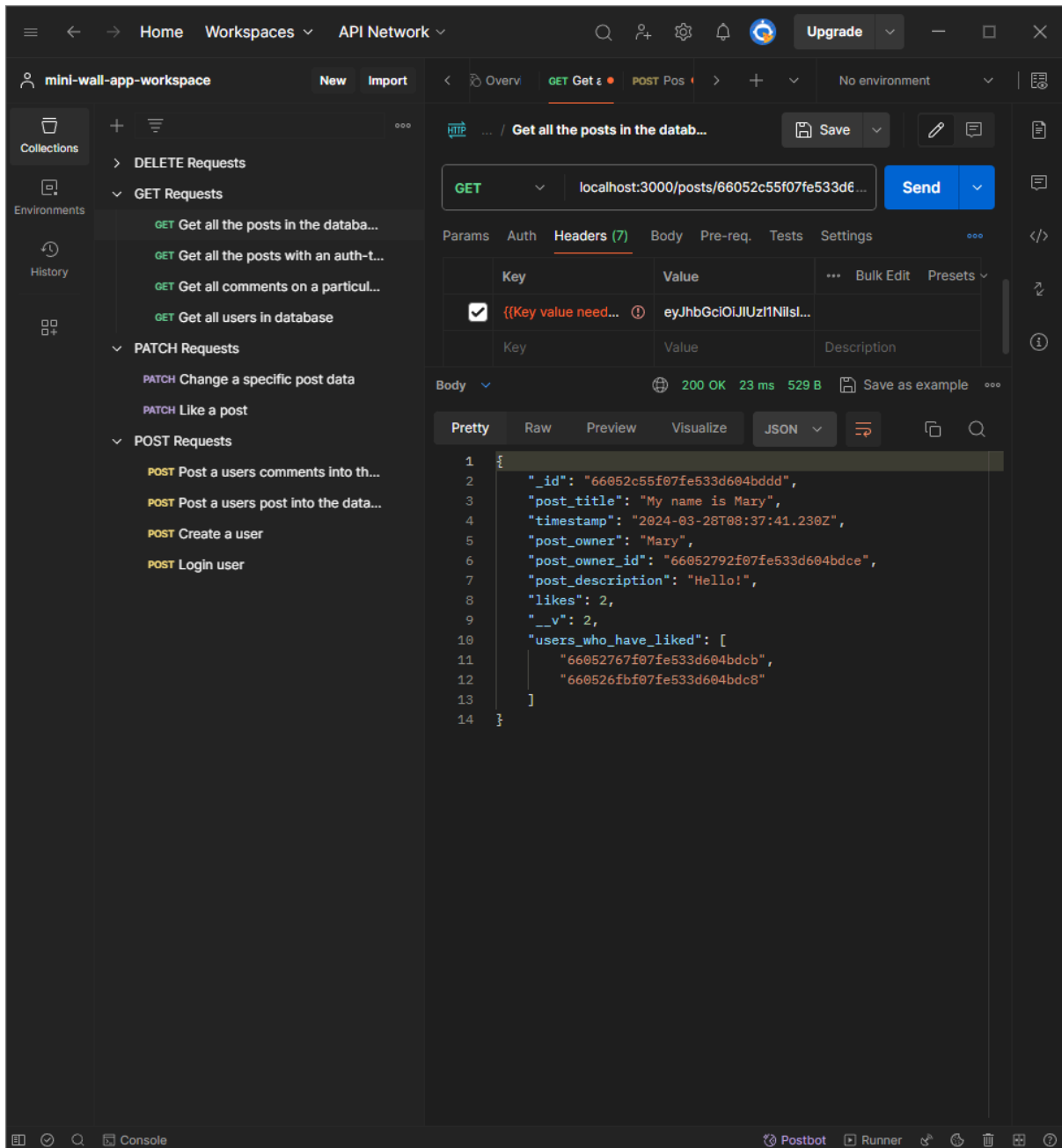
Console

PostbotRunner

TC 13. Mary likes her posts. This call should be unsuccessful; the owner cannot like their posts.



TC 14. Mary can see that there are two likes in her posts.



TC 15. Nick can see the list of posts, since Mary's post has two likes it is shown at the top.

Not implemented.

Phase E: Deploy your MiniWall project into a GCP VM using Docker

Following the video (2024a) I created a GCP VM with 15GB of disk space, using an Ubuntu boot image and allowing HTTPS traffic so I can connect to the VM.

I then installed Docker and followed the video to attempt to deploy my project on docker (University of London, 2024e):

```
ssh.cloud.google.com/v2/ssh/projects/cloud-class-410616/zones/us-central1-a/instances/instance-20240328-111238?authuser=0&hl=en_US&proje...
ssh.cloud.google.com/v2/ssh/projects/cloud-class-410616/zones/us-central1-a/instances/instance-20240328-111238?authuser=0&hl=en_US&p...
SSH-in-browser
UPLOAD FILE
DOWNLOAD FILE

thomasrobertgardner1056@instance-20240328-111238:~$ sudo apt get install docker.io
E: Invalid operation get
thomasrobertgardner1056@instance-20240328-111238:~$ sudo apt-get install docker.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libatasmart4 libblockdev-fs2 libblockdev-loop2 libblockdev-part-err2 libblockdev-part2 libblockdev-swap2
  libblockdev-utils2 libblockdev2 libbim-glib4 libbim-proxy libmm-glib0 libnspr4 libnss3 libnuma1
  libparted-fs-resize0 libqmi-glib5 libqmi-proxy libudisks2-0 libxmlb2 usb-modeswitch usb-modeswitch-data
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base libidn11 pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap docker-doc rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base docker.io libidn11 pigz runc ubuntu-fan
0 upgraded, 9 newly installed, 0 to remove and 29 not upgraded.
Need to get 63.3 MB of archives.
After this operation, 267 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us-central1.gce.archive.ubuntu.com/ubuntu focal/universe amd64 pigz amd64 2.4-1 [57.4 kB]
Get:2 http://us-central1.gce.archive.ubuntu.com/ubuntu focal/main amd64 bridge-utils amd64 1.6-2ubuntu1 [30.5 kB]
Get:3 http://us-central1.gce.archive.ubuntu.com/ubuntu focal-updates/main amd64 runc amd64 1.1.7-0ubuntu1~20.04.2 [383
6 kB]
Get:4 http://us-central1.gce.archive.ubuntu.com/ubuntu focal-updates/main amd64 containerd amd64 1.7.2-0ubuntu1~20.04.
1 [32.5 MB]
Get:5 http://us-central1.gce.archive.ubuntu.com/ubuntu focal-updates/main amd64 dns-root-data all 2023112702-ubuntu0.2
0.04.1 [5308 B]
Get:6 http://us-central1.gce.archive.ubuntu.com/ubuntu focal/main amd64 libidn11 amd64 1.33-2.2ubuntu2 [46.2 kB]
Get:7 http://us-central1.gce.archive.ubuntu.com/ubuntu focal-updates/main amd64 dnsmasq-base amd64 2.90-0ubuntu0.20.04
.1 [350 kB]
Get:8 http://us-central1.gce.archive.ubuntu.com/ubuntu focal-updates/universe amd64 docker.io amd64 24.0.5-0ubuntu1~20
.04.1 [26.4 MB]
Get:9 http://us-central1.gce.archive.ubuntu.com/ubuntu focal-updates/main amd64 ubuntu-fan all 0.12.13ubuntu0.1 [34.4
kB]
Fetched 63.3 MB in 2s (30.8 MB/s)
Preconfiguring packages ...
Selecting previously unselected package pigz.
(Reading database ... 62275 files and directories currently installed.)
Preparing to unpack .../0-pigz_2.4-1_amd64.deb ...
Unpacking pigz (2.4-1) ...
Selecting previously unselected package bridge-utils.
Preparing to unpack .../1-bridge-utils_1.6-2ubuntu1_amd64.deb ...
Unpacking bridge-utils (1.6-2ubuntu1) ...
Selecting previously unselected package runc.
Preparing to unpack .../2-runc_1.1.7-0ubuntu1~20.04.2_amd64.deb ...
Unpacking runc (1.1.7-0ubuntu1~20.04.2) ...
Selecting previously unselected package containerd.
Preparing to unpack .../3-containerd_1.7.2-0ubuntu1~20.04.1_amd64.deb ...
Unpacking containerd (1.7.2-0ubuntu1~20.04.1) ...
Selecting previously unselected package dns-root-data.
Preparing to unpack .../4-dns-root-data_2023112702-ubuntu0.20.04.1_all.deb ...
Unpacking dns-root-data (2023112702-ubuntu0.20.04.1) ...
Selecting previously unselected package libidn11:amd64.
Preparing to unpack .../5-libidn11_1.33-2.2ubuntu2_amd64.deb ...
Unpacking libidn11:amd64 (1.33-2.2ubuntu2) ...
Selecting previously unselected package dnsmasq-base.
Preparing to unpack .../6-dnsmasq-base_2.90-0ubuntu0.20.04.1_amd64.deb ...
Unpacking dnsmasq-base (2.90-0ubuntu0.20.04.1) ...
Selecting previously unselected package docker.io.
```

```
ssh.cloud.google.com/v2/ssh/projects/cloud-class-410616/zones/us-central1-a/instances/instance-20240328-111238?authuser=0&hl=en_US&proje...  
ssh.cloud.google.com/v2/ssh/projects/cloud-class-410616/zones/us-central1-a/instances/instance-20240328-111238?authuser=0&hl=en_US&p...  
SSH-in-browser  
UPLOAD FILE  
DOWNLOAD FILE  
thomasrobertgardner1056@instance-20240328-111238:~$ sudo add user docker user  
sudo: add: command not found  
thomasrobertgardner1056@instance-20240328-111238:~$ sudo adduser docker-user  
Adding user `docker-user' ...  
Adding new group `docker-user' (1004) ...  
Adding new user `docker-user' (1003) with group `docker-user' ...  
Creating home directory `/home/docker-user' ...  
Copying files from `/etc/skel' ...  
New password:  
Retype new password:  
passwd: password updated successfully  
Changing the user information for docker-user  
Enter the new value, or press ENTER for the default  
Full Name []:  
Room Number []:  
Work Phone []:  
Home Phone []:  
Other []:  
Is the information correct? [Y/n]  
thomasrobertgardner1056@instance-20240328-111238:~$ sudo usermod -aG sudo docker-user  
thomasrobertgardner1056@instance-20240328-111238:~$ sudo usermod -aG docker docker-user  
thomasrobertgardner1056@instance-20240328-111238:~$ su - docker-user  
Password:  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
docker-user@instance-20240328-111238:~$ pwd  
/home/docker-user  
docker-user@instance-20240328-111238:~$
```

```
ssh.cloud.google.com/v2/ssh/projects/cloud-class-410616/zones/us-central1-a/instances/instance-20240328-111238?authuser=0&hl=en_US&proje...
ssh.cloud.google.com/v2/ssh/projects/cloud-class-410616/zones/us-central1-a/instances/instance-20240328-111238?authuser=0&hl=en_US&p...
SSH-in-browser
[+] UPLOAD FILE [v] DOWNLOAD FILE [i] [k] [s]

docker-user@instance-20240328-111238:~$ docker run
"docker run" requires at least 1 argument.
See 'docker run --help'.

Usage: docker run [OPTIONS] IMAGE [COMMAND] [ARG...]

Create and run a new container from an image
docker-user@instance-20240328-111238:~$ docker run --name mini-python -it python
Unable to find image 'python:latest' locally
latest: Pulling from library/python
71215d55680c: Pull complete
3cb8f9c23302: Pull complete
5f899db30843: Pull complete
567db630df8d: Pull complete
d68cd2123173: Pull complete
63941d09e532: Pull complete
097431623722: Pull complete
09527fa4de8d: Pull complete
Digest: sha256:19973e1796237522ed1fcc1357c766770b47dc15854eafdda055b65953fe5ec1
Status: Downloaded newer image for python:latest
Python 3.12.2 (main, Mar 12 2024, 11:02:14) [GCC 12.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
KeyboardInterrupt
>>> exit
Use exit() or Ctrl-D (i.e. EOF) to exit
>>>
docker-user@instance-20240328-111238:~$ su docker-user
Password:
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

docker-user@instance-20240328-111238:~$ ls
docker-user@instance-20240328-111238:~$ git clone --branch master https://bobbobdude:ghp_IJswAO3v7gl19sFymwaBPbOZnVTwv
13wuF15@github.com:bobbobdude/gcp-coursework.git
Cloning into 'gcp-coursework'...
fatal: Remote branch master not found in upstream origin
docker-user@instance-20240328-111238:~$ git clone --branch master https://bobbobdude:ghp_IJswAO3v7gl19sFymwaBPbOZnVTwv
13wuF15@github.com:bobbobdude/gcp-coursework.git
Cloning into 'gcp-coursework'...
fatal: Remote branch master not found in upstream origin
docker-user@instance-20240328-111238:~$ git clone https://github.com/University-of-London/csm020-coursework-bobbobdude
Cloning into 'csm020-coursework-bobbobdude'...
Username for 'https://github.com': bobbobdude
Password for 'https://bobbobdude@github.com':
remote: Enumerating objects: 2355, done.
remote: Counting objects: 100% (2355/2355), done.
remote: Compressing objects: 100% (2009/2009), done.
remote: Total 2355 (delta 296), reused 2309 (delta 259), pack-reused 0
Receiving objects: 100% (2355/2355), 3.59 MiB | 12.85 MiB/s, done.
Resolving deltas: 100% (296/296), done.
docker-user@instance-20240328-111238:~$ ls
csm020-coursework-bobbobdude
docker-user@instance-20240328-111238:~$
```

ssh.cloud.google.com/v2/ssh/projects/cloud-class-410616/zones/us-central1-a/instances/instance-20240328-111238?authuser=0&hl=en_US&proje...
ssh.cloud.google.com/v2/ssh/projects/cloud-class-410616/zones/us-central1-a/instances/instance-20240328-111238?authuser=0&hl=en_US&p...
SSH-in-browser
UPLOAD FILE
DOWNLOAD FILE
GNU nano 4.8
Dockerfile
FROM alpine
RUN apk add --update nodejs npm
COPY ./src
WORKDIR /src
EXPOSE 3000
ENTRYPOINT ["node", "./app.js"]
[Wrote 6 lines]
Get Help
Exit
Write Out
Read File
Where Is
Replace
Cut Text
Paste Text
Justify
To Spell
Cur Pos
Go To Line
Undo
Redo

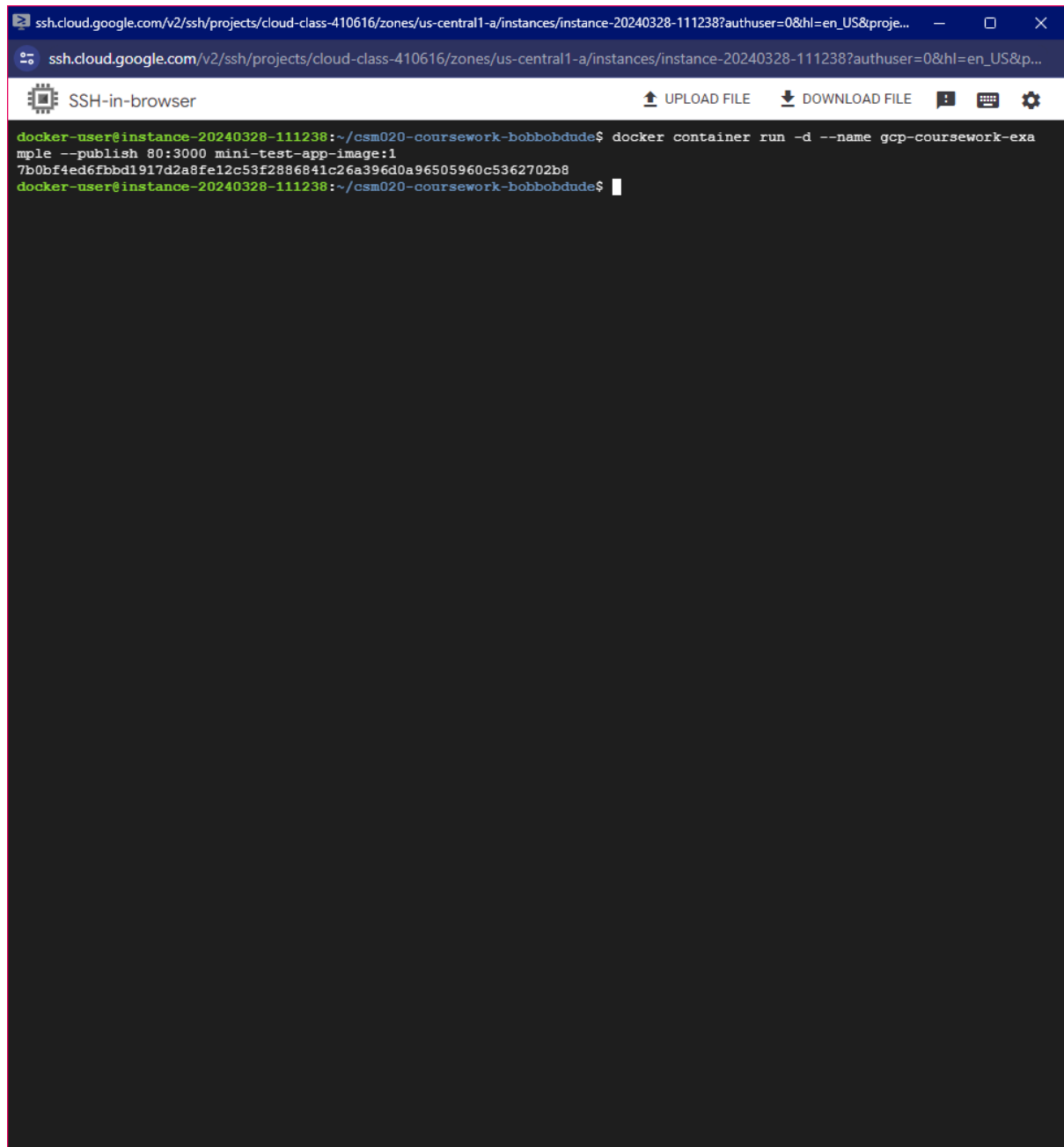
```
ssh.cloud.google.com/v2/ssh/projects/cloud-class-410616/zones/us-central1-a/instances/instance-20240328-111238?authuser=0&hl=en_US&proje...
ssh.cloud.google.com/v2/ssh/projects/cloud-class-410616/zones/us-central1-a/instances/instance-20240328-111238?authuser=0&hl=en_US&p...
SSH-in-browser
[+] UPLOAD FILE [v] DOWNLOAD FILE [i] [k] [g]

"docker image build" requires exactly 1 argument.
See 'docker image build --help'.

Usage: docker image build [OPTIONS] PATH | URL | -

Build an image from a Dockerfile
docker-user@instance-20240328-111238:~/csm020-coursework-bobboobdude$ docker image build -t mini-test-app-image:1 .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/

Sending build context to Docker daemon 20.09MB
Step 1/6 : FROM alpine
latest: Pulling from library/alpine
4abcf2066143: Pull complete
Digest: sha256:c5b1261d6d3e43071626931fc004f70149baeba2c8ec672bd4f27761f8e1ad6b
Status: Downloaded newer image for alpine:latest
--> 05455a08881e
Step 2/6 : RUN apk add --update nodejs npm
--> Running in a0c648ff250e
fetch https://dl-cdn.alpinelinux.org/alpine/v3.19/main/x86_64/APKINDEX.tar.gz
fetch https://dl-cdn.alpinelinux.org/alpine/v3.19/community/x86_64/APKINDEX.tar.gz
(1/12) Installing ca-certificates (20240226-r0)
(2/12) Installing libgcc (13.2.1_git20231014-r0)
(3/12) Installing libstdc++ (13.2.1_git20231014-r0)
(4/12) Installing ada-libc (2.7.4-r0)
(5/12) Installing libbase64 (0.5.0-r0)
(6/12) Installing brotli-libs (1.1.0-r1)
(7/12) Installing c-ares (1.27.0-r0)
(8/12) Installing icu-data-en (74.1-r0)
Executing icu-data-en-74.1-r0.post-install
*
* If you need ICU with non-English locales and legacy charset support, install
* package icu-data-full.
*
(9/12) Installing icu-libs (74.1-r0)
(10/12) Installing nghttp2-libs (1.58.0-r0)
(11/12) Installing nodejs (20.11.1-r0)
(12/12) Installing npm (10.2.5-r0)
Executing busybox-1.36.1-r15.trigger
Executing ca-certificates-20240226-r0.trigger
OK: 74 MiB in 27 packages
Removing intermediate container a0c648ff250e
--> 7009fbc01adb
Step 3/6 : COPY . /src
--> 1766ec99d8a9
Step 4/6 : WORKDIR /src
--> Running in 4c439029aad7
Removing intermediate container 4c439029aad7
--> 8059f5b4496d
Step 5/6 : EXPOSE 3000
--> Running in c23be903f5a3
Removing intermediate container c23be903f5a3
--> 142a30180dd2
Step 6/6 : ENTRYPOINT ["node", "./app.js"]
--> Running in 9c0743dec2e7
Removing intermediate container 9c0743dec2e7
--> 31cbclbc42fc
Successfully built 31cbclbc42fc
Successfully tagged mini-test-app-image:1
docker-user@instance-20240328-111238:~/csm020-coursework-bobboobdude$
```

```
ssh.cloud.google.com/v2/ssh/projects/cloud-class-410616/zones/us-central1-a/instances/instance-20240328-111238?authuser=0&hl=en_US&proj...  
SSH-in-browser  
UPLOAD FILE  
DOWNLOAD FILE  
docker-user@instance-20240328-111238:~/csm020-coursework-bobbohdude$ docker container run -d --name gcp-coursework-example --publish 80:3000 mini-test-app-image:1  
7b0bf4ed6fbbd1917d2a8fe12c53f2886841c26a396d0a96505960c5362702b8  
docker-user@instance-20240328-111238:~/csm020-coursework-bobbohdude$
```

And unfortunately this was as far as I got.

References

University of London (2024a). *CSM020-2024-JAN: 1.7 Lecture 6: Introduction to Google Cloud Platform (GCP) services and virtual machines* | MyLondon. [online] London.ac.uk. Available at: <https://learn.london.ac.uk/mod/page/view.php?id=146102&forceview=1> [Accessed 28 Mar. 2024].

University of London (2024b). *CSM020-2024-JAN: 2.8 Lab 2, Task 1: Building the MiniFilm application using MongoDB and NodeJS*. [online] London.ac.uk. Available at: <https://learn.london.ac.uk/mod/page/view.php?id=146127> [Accessed 14 Feb. 2024].

University of London (2024c). *CSM020-2024-JAN: 3.8 Lab 3, Part 1: Building the MiniPost REST microservice*. [online] London.ac.uk. Available at: <https://learn.london.ac.uk/mod/page/view.php?id=146146> [Accessed 14 Feb. 2024].

University of London (2024d). *CSM020-2024-JAN: 4.6 Lab 4, Part 1: MiniFilm REST Verification and Authentication*. [online] London.ac.uk. Available at: <https://learn.london.ac.uk/mod/page/view.php?id=146162> [Accessed 4 Mar. 2024].

University of London (2024e). *CSM020-2024-JAN: 5.6 Lab 5, Part 3: Pushing code to GitHub and Docker | MyLondon*. [online] London.ac.uk. Available at: <https://learn.london.ac.uk/mod/page/view.php?id=146179&forceview=1> [Accessed 28 Mar. 2024].

