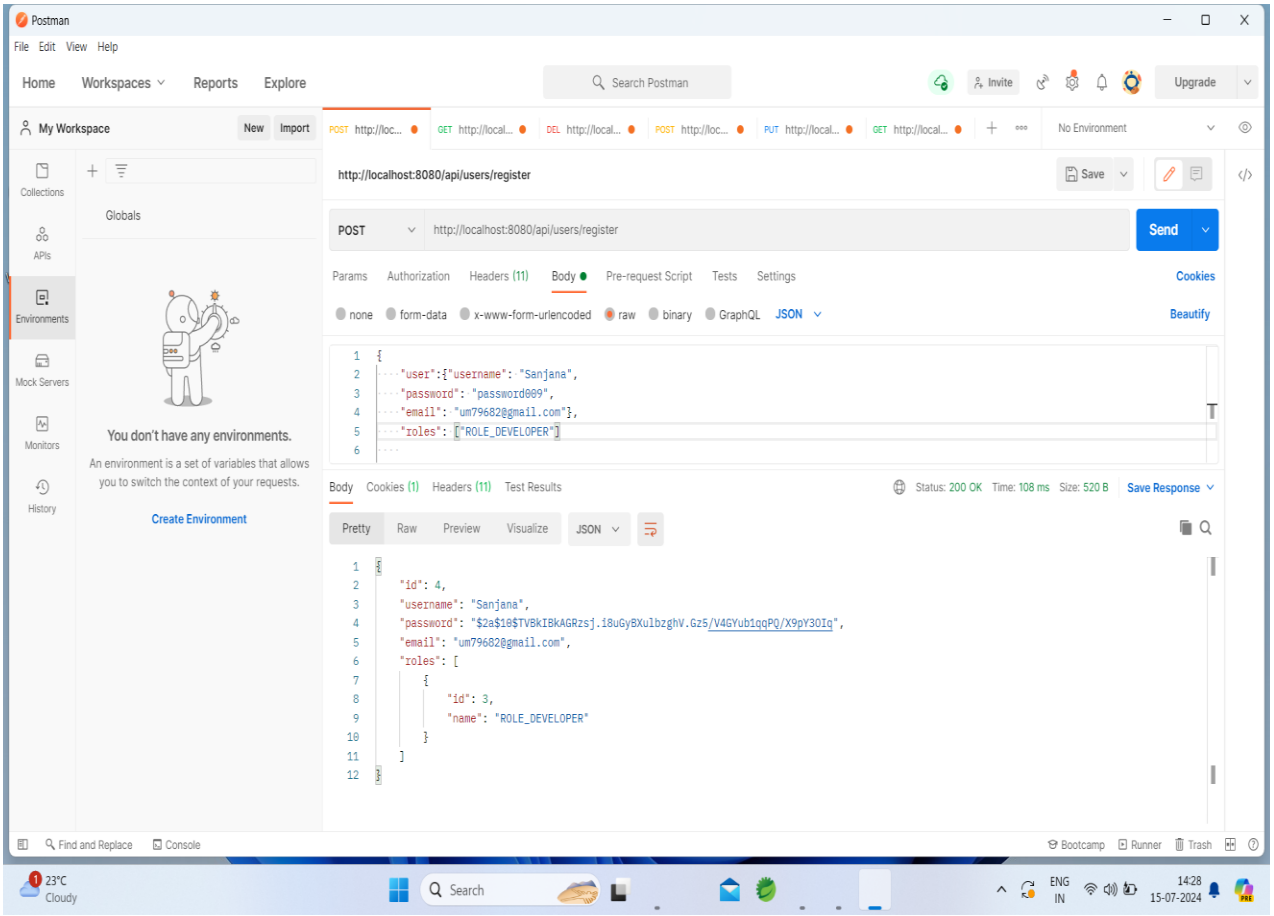
UserService API Using Java And SpringBoot

1. Implement a Spring Boot microservice responsible for user management.
2. User Controller

- End Points : /api/users/register
- To register User
- URL :- http://localhost:8080/api/users/register
- Method : POST
- Request Body as in screenshot
- 

Postman
File Edit View Help
Home Workspaces Reports Explore
Search Postman
Invite
Upgrade
My Workspace
New Import
Collections
APIs
Environments
Mock Servers
Monitors
History
Globals
You don't have any environments.
An environment is a set of variables that allows you to switch the context of your requests.
Create Environment
No Environment
Save
Send
Params Authorization Headers (11) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
Body Cookies (1) Headers (11) Test Results
Save Response
Pretty Raw Preview Visualize JSON
Find and Replace Console
Bootcamp Runner Trash
23°C Cloudy
Search
ENG IN
14:28 15-07-2024

http://localhost:8080/api/users/register

POST http://localhost:8080/api/users/register

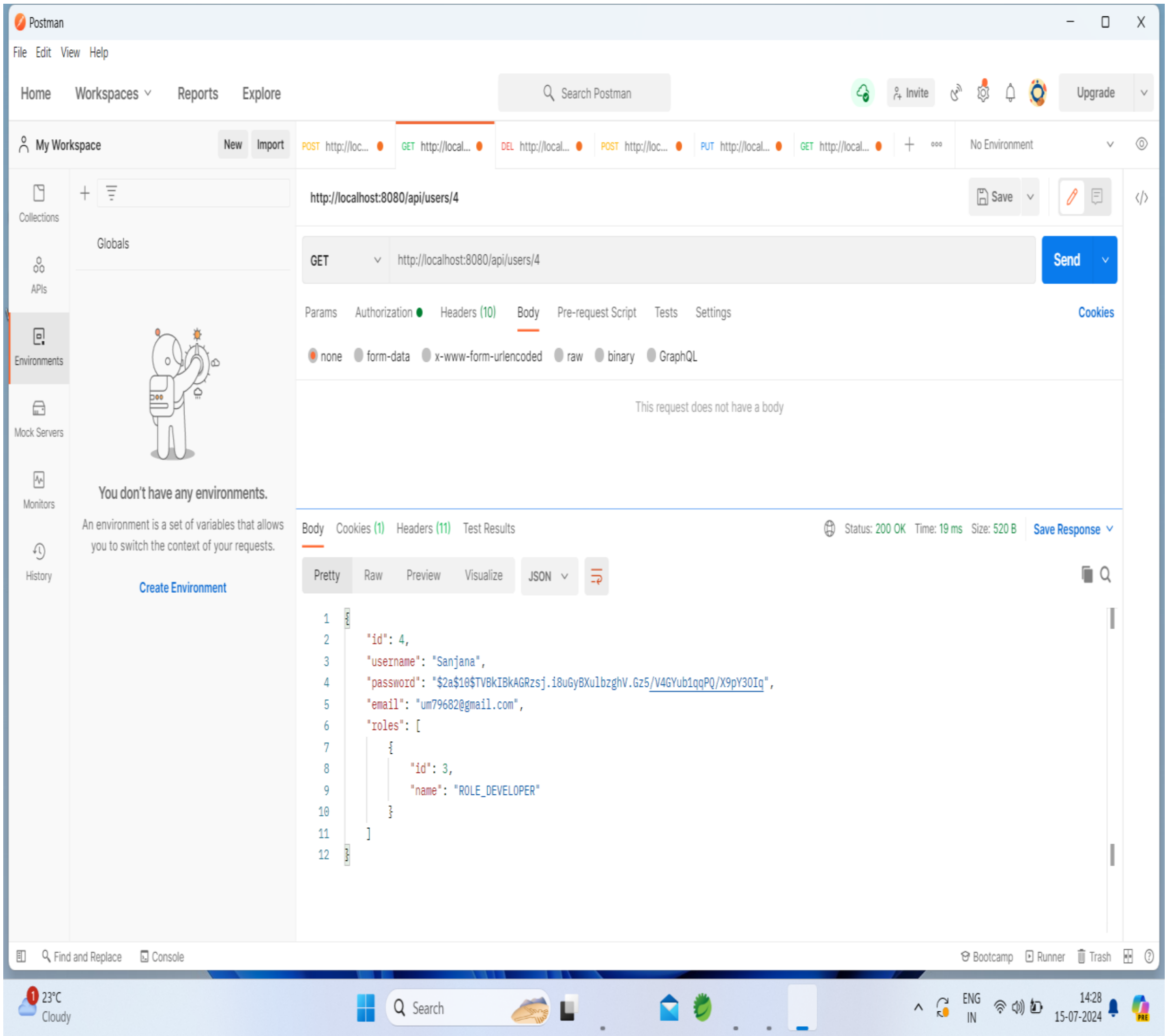
```
{
    "user":{"username": "Sanjana",
    "password": "password009",
    "email": "um79682@gmail.com"},
    "roles": ["ROLE_DEVELOPER"]

```

Status: 200 OK Time: 108 ms Size: 520 B

```
{
    "id": 4,
    "username": "Sanjana",
    "password": "$2a$10$TVBkIBkAGRzsj.i8uGyBXulbzghV.Gz5/V4GYub1qqPQ/X9pY3OIq",
    "email": "um79682@gmail.com",
    "roles": [
        {
            "id": 3,
            "name": "ROLE_DEVELOPER"
        }
    ]
}
```

- End Point : /api/users/{id}
- URL :- http://localhost:8080/api/users/{id}
- To get user
- Method : GET
- Headers : Basic Auth
- Authorization : Username → admin and Password → password
- Response as in screen shot



http://localhost:8080/api/users/4

GET http://localhost:8080/api/users/4

Body: ● none ○ form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

This request does not have a body

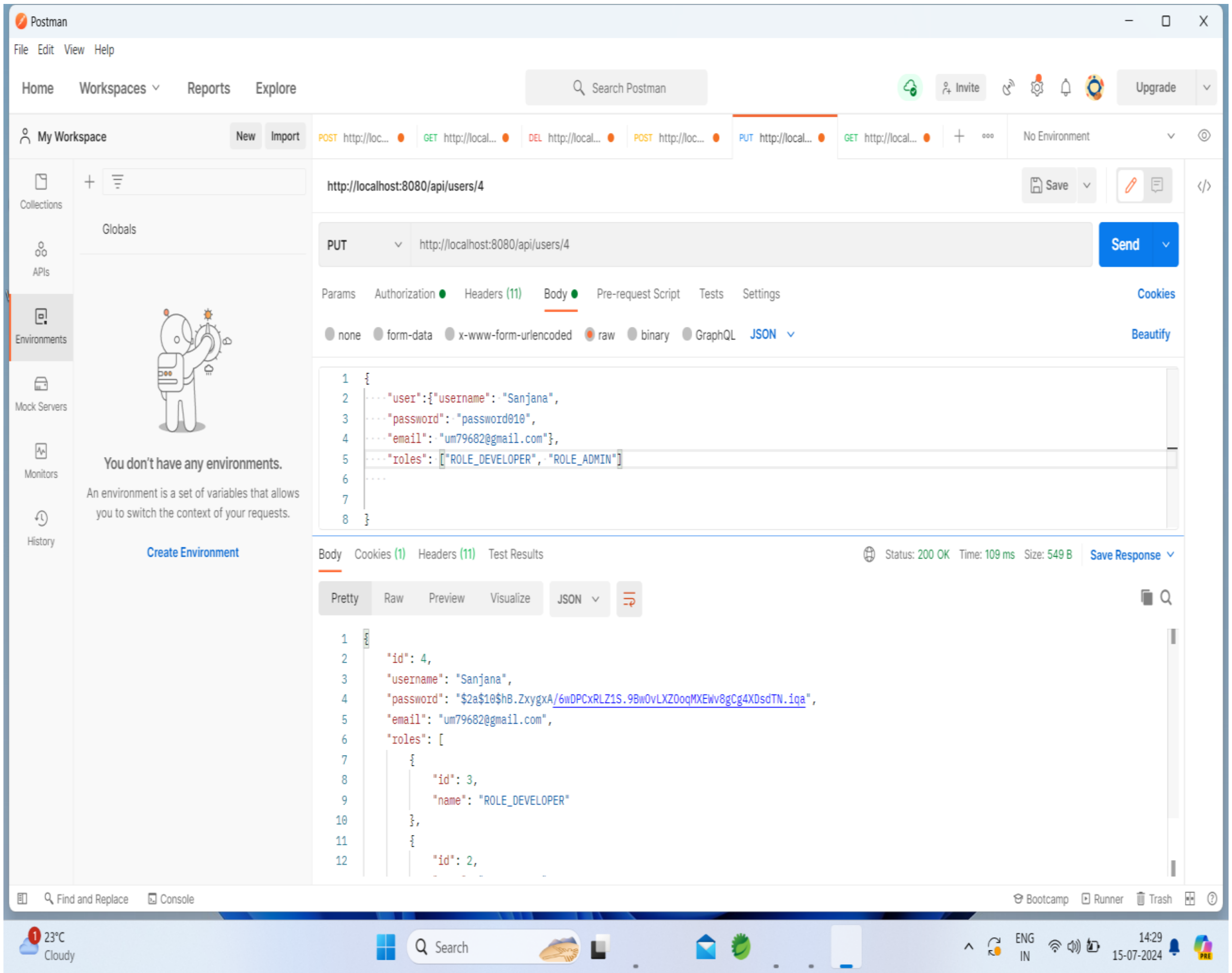
Body Cookies (1) Headers (11) Test Results

Status: 200 OK Time: 19 ms Size: 520 B

```
{
    "id": 4,
    "username": "Sanjana",
    "password": "$2a$10$TVBkIBkAGRzsj.i8uGyBXulbzghV.Gz5/V4GYub1qqPQ/X9pY30Iq",
    "email": "um79682@gmail.com",
    "roles": [
        {
            "id": 3,
            "name": "ROLE_DEVELOPER"
        }
    ]
}
```

-

- End Point : /api/users/{id}
- URL :- http://localhost:8080/api/users/{id}
- To update user
- Method : PUT
- Headers : Basic Auth
- Authorization : Username → admin and Password → password
- Request Body as in screenshot



http://localhost:8080/api/users/4

PUT http://localhost:8080/api/users/4

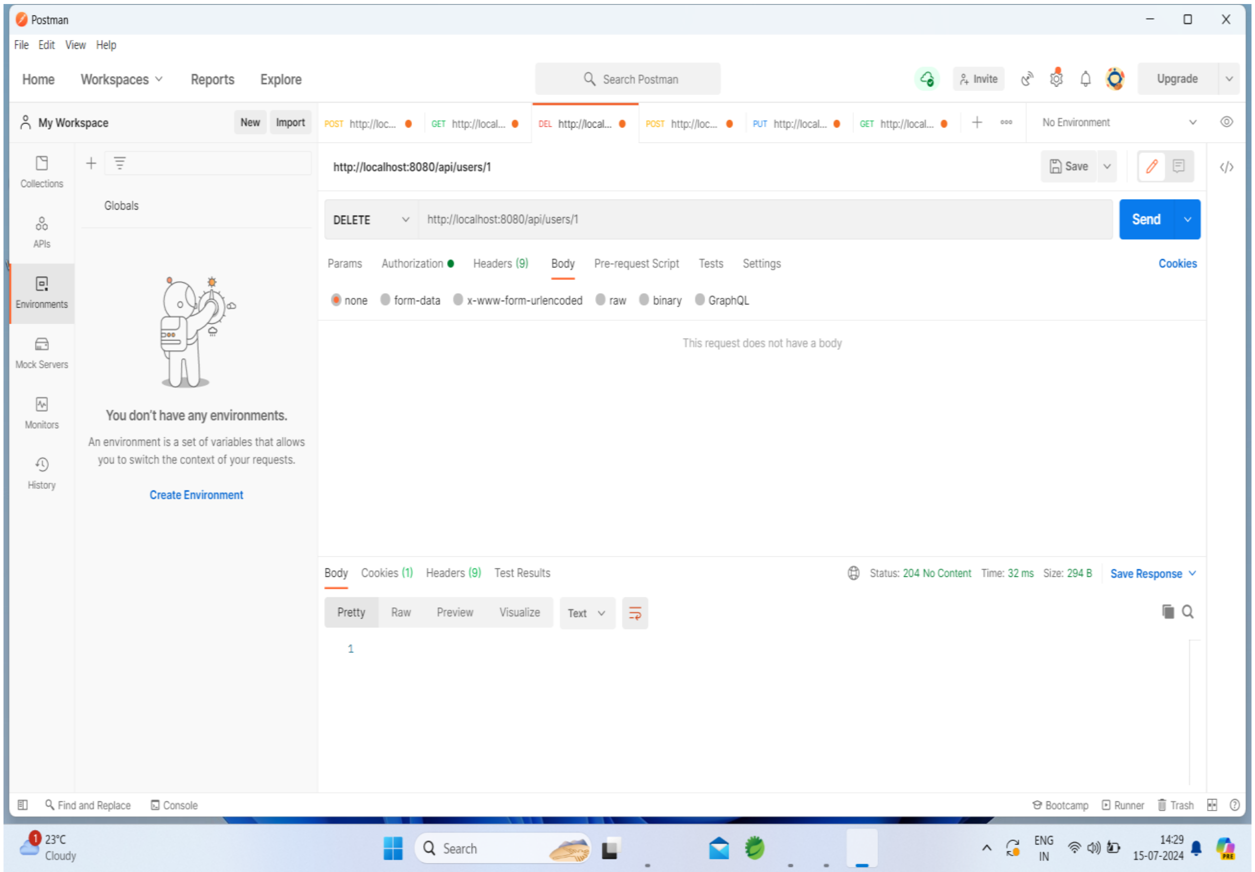
```
{
    "user":{"username": "Sanjana",
    "password": "password010",
    "email": "um79682@gmail.com"},
    "roles": ["ROLE_DEVELOPER", "ROLE_ADMIN"]
    
    
}
```

Status: 200 OK Time: 109 ms Size: 549 B

```
{
    "id": 4,
    "username": "Sanjana",
    "password": "$2a$10$hB.ZxygxA/6wDPCxRLZ1S.9BwOvLXZOoqMXEWv8gCg4XDsdTN.iqa",
    "email": "um79682@gmail.com",
    "roles": [
        {
            "id": 3,
            "name": "ROLE_DEVELOPER"
        },
        {
            "id": 2,
```

-

- End Points : /api/users/{id}
- To delete user
- Method: DELETE
- URL :- http://localhost:8080/api/users/{id}
- Headers : Basic Auth
- Authorization : Username → admin and Password → password
- Request as in screen shot



http://localhost:8080/api/users/1

DELETE http://localhost:8080/api/users/1

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

● none ○ form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

This request does not have a body

You don't have any environments.

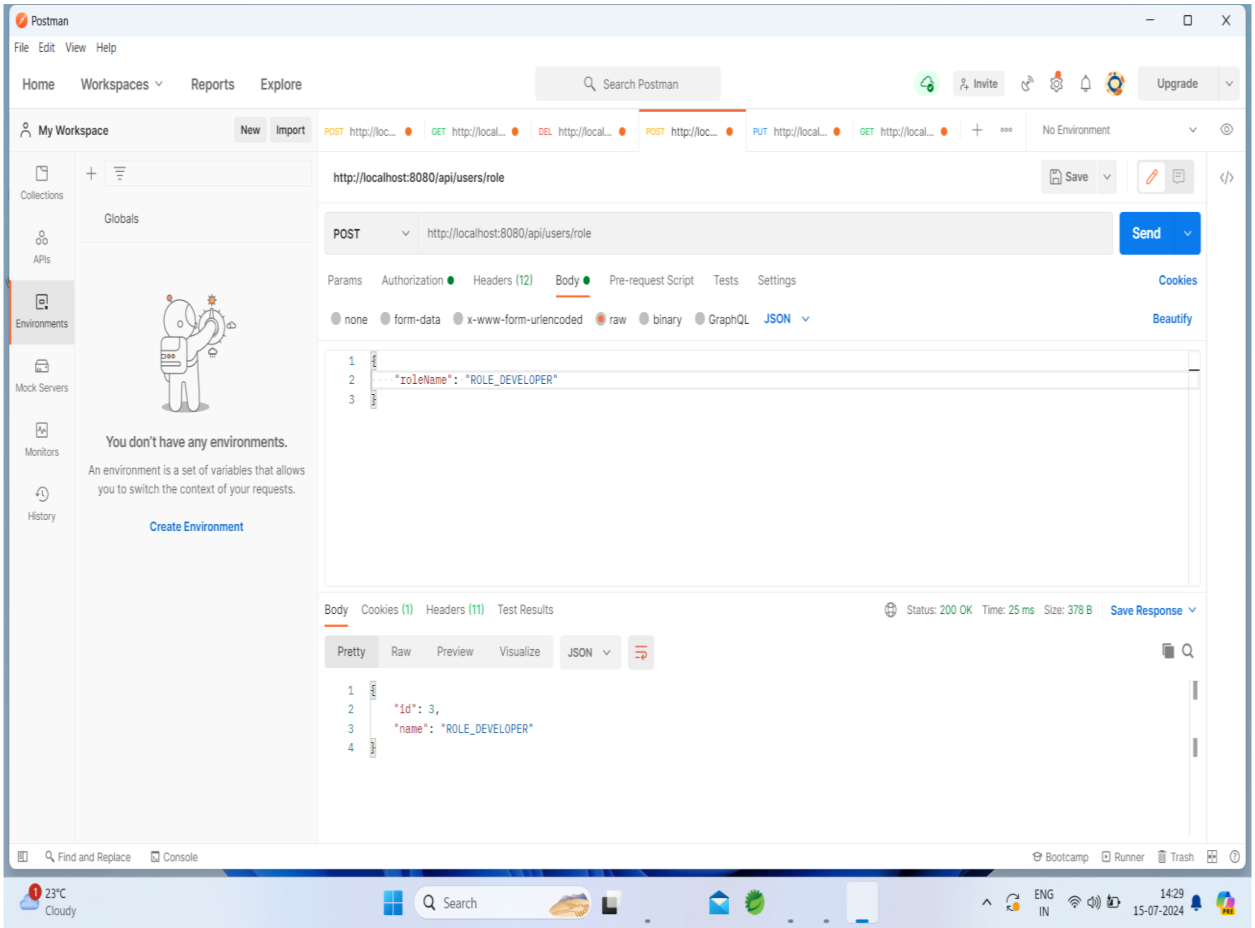
An environment is a set of variables that allows you to switch the context of your requests.

Create Environment

Body | Cookies (1) | Headers (9) | Test Results

Status: 204 No Content Time: 32 ms Size: 294 B

```
1
```

- End Points : /api/users/role
- To create Role
- Method: POST
- URL :- http://localhost:8080/api/users/role
- Headers : Basic Auth
- Authorization : Username → admin and Password → password
- Response as in screenshot
- 

Postman
File Edit View Help
Home Workspaces Reports Explore
Search Postman
Invite
Upgrade
My Workspace
New Import
No Environment
Collections
APIs
Environments
Mock Servers
Monitors
History
Globals
You don't have any environments.
An environment is a set of variables that allows you to switch the context of your requests.
Create Environment
Save
Send
Params Authorization Headers (12) Body Pre-request Script Tests Settings
Cookies
Beautify
Body Cookies (1) Headers (11) Test Results
Save Response
Pretty Raw Preview Visualize JSON
Find and Replace Console
Bootcamp Runner Trash
23°C Cloudy
Search
ENG IN
14:29 15-07-2024

http://localhost:8080/api/users/role

POST http://localhost:8080/api/users/role

Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

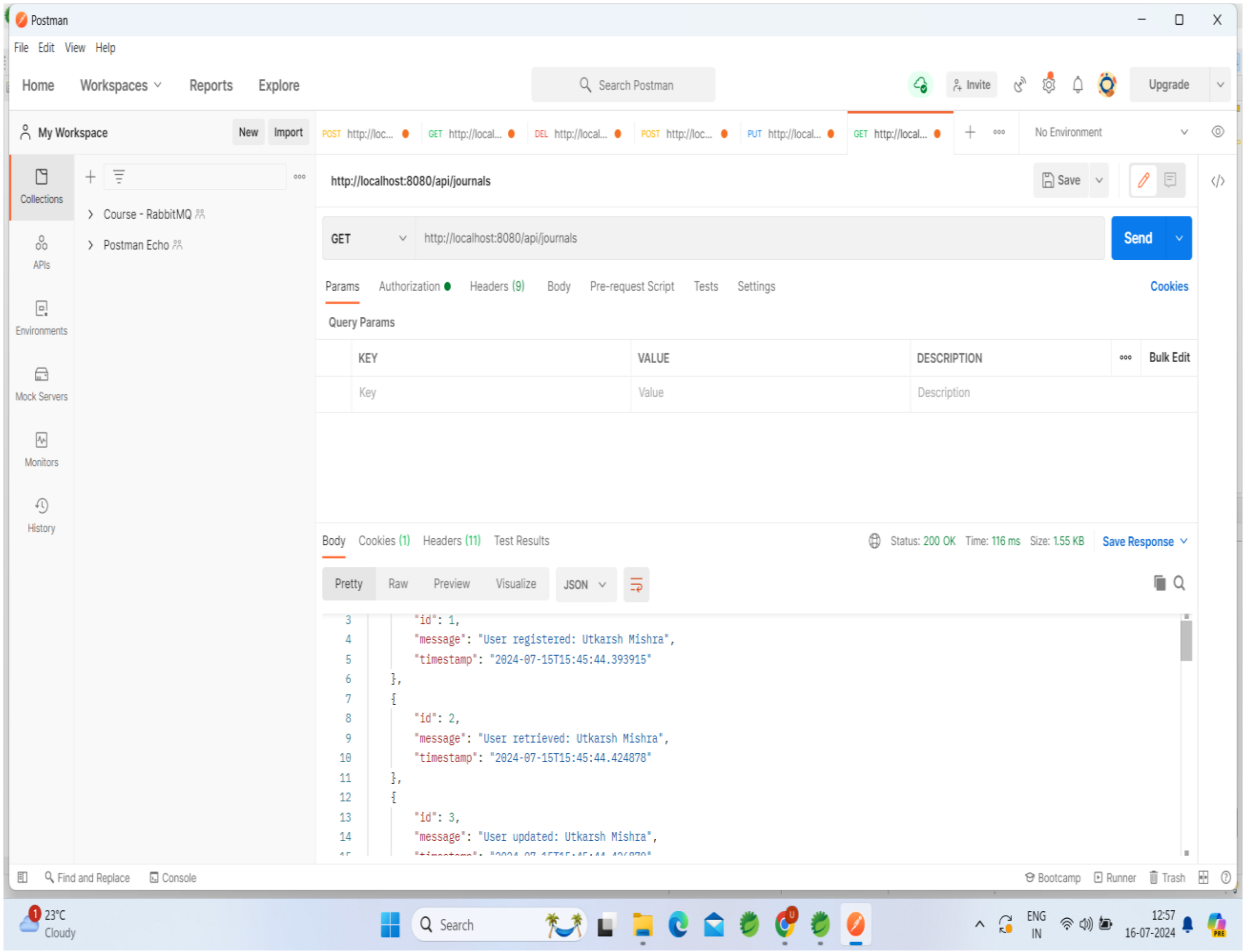
```
{
    "roleName": "ROLE_DEVELOPER"
}
```

Status: 200 OK Time: 25 ms Size: 378 B

```
{
    "id": 3,
    "name": "ROLE_DEVELOPER"
}
```

3. Implement another Spring Boot Micro service responsible for journaling user events.
4. Journal Controller

- End Points : /api/journals
- To get all journals
- Method: GET
- URL :- http://localhost:8080/api/journals
- Headers : Basic Auth
- Authorization : Username → admin and Password → password
- Response as in screenshot



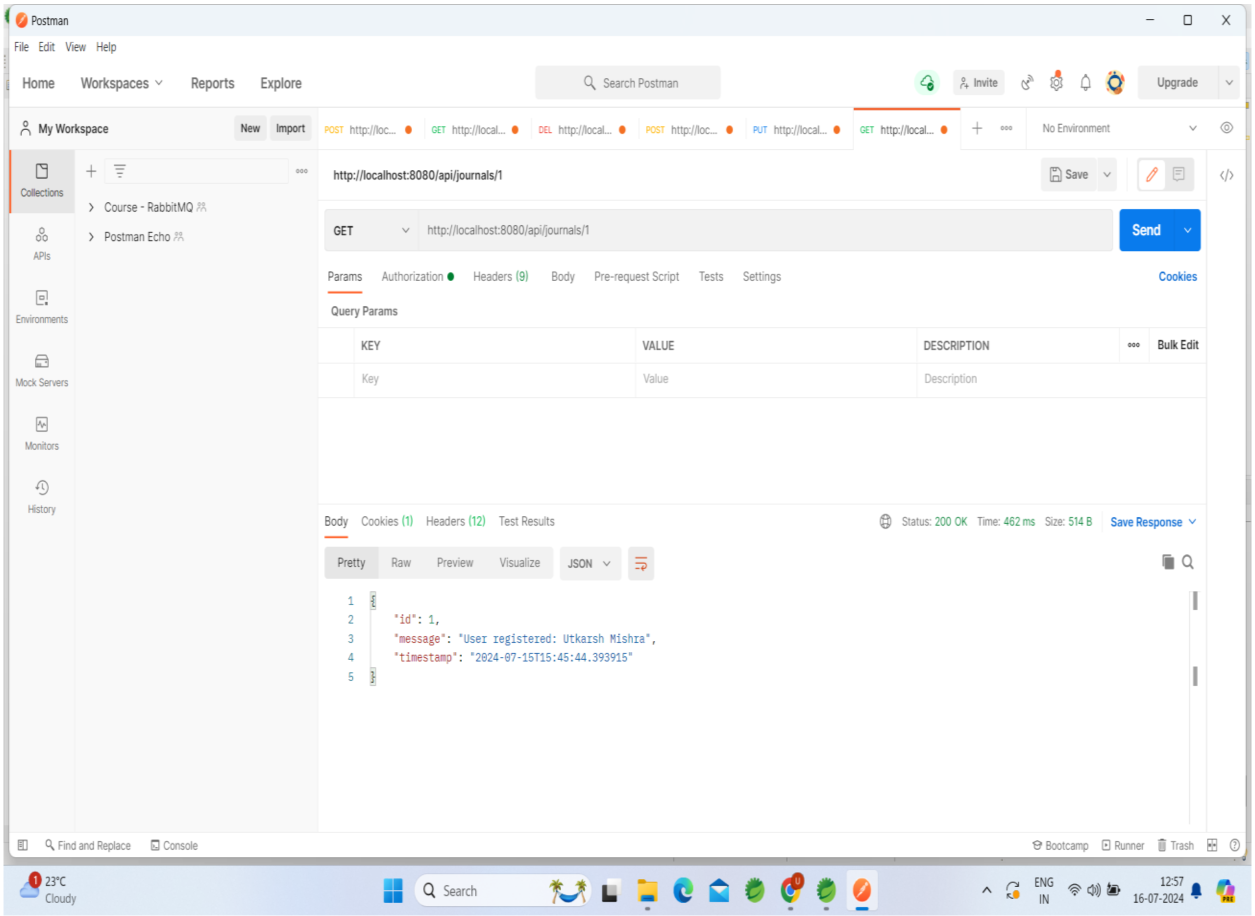
http://localhost:8080/api/journals

GET http://localhost:8080/api/journals

```
        "id": 1,
        "message": "User registered: Utkarsh Mishra",
        "timestamp": "2024-07-15T15:45:44.393915"
    },
    {
        "id": 2,
        "message": "User retrieved: Utkarsh Mishra",
        "timestamp": "2024-07-15T15:45:44.424878"
    },
    {
        "id": 3,
        "message": "User updated: Utkarsh Mishra",
```

-

- End Points : /api/journals/{id}
- To get single journal
- Method: GET
- URL :- http://localhost:8080/api/journal/{id}
- Headers : Basic Auth
- Authorization : Username → admin and Password → password
- Response as in screenshot



http://localhost:8080/api/journals/1

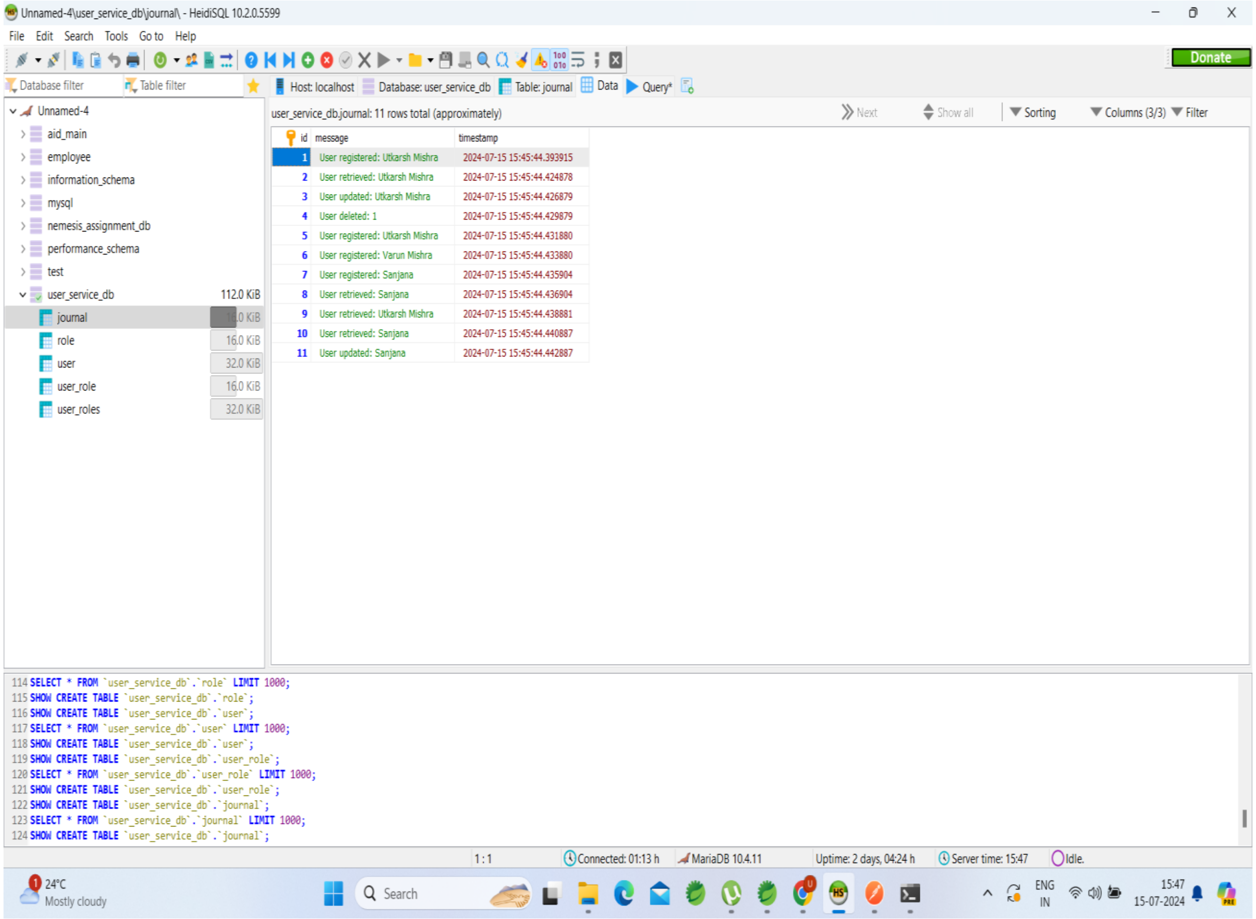
GET http://localhost:8080/api/journals/1

Status: 200 OK Time: 462 ms Size: 514 B

```
{
    "id": 1,
    "message": "User registered: Utkarsh Mishra",
    "timestamp": "2024-07-15T15:45:44.393915"
}
```

-

- Screenshot of Journal entry in database



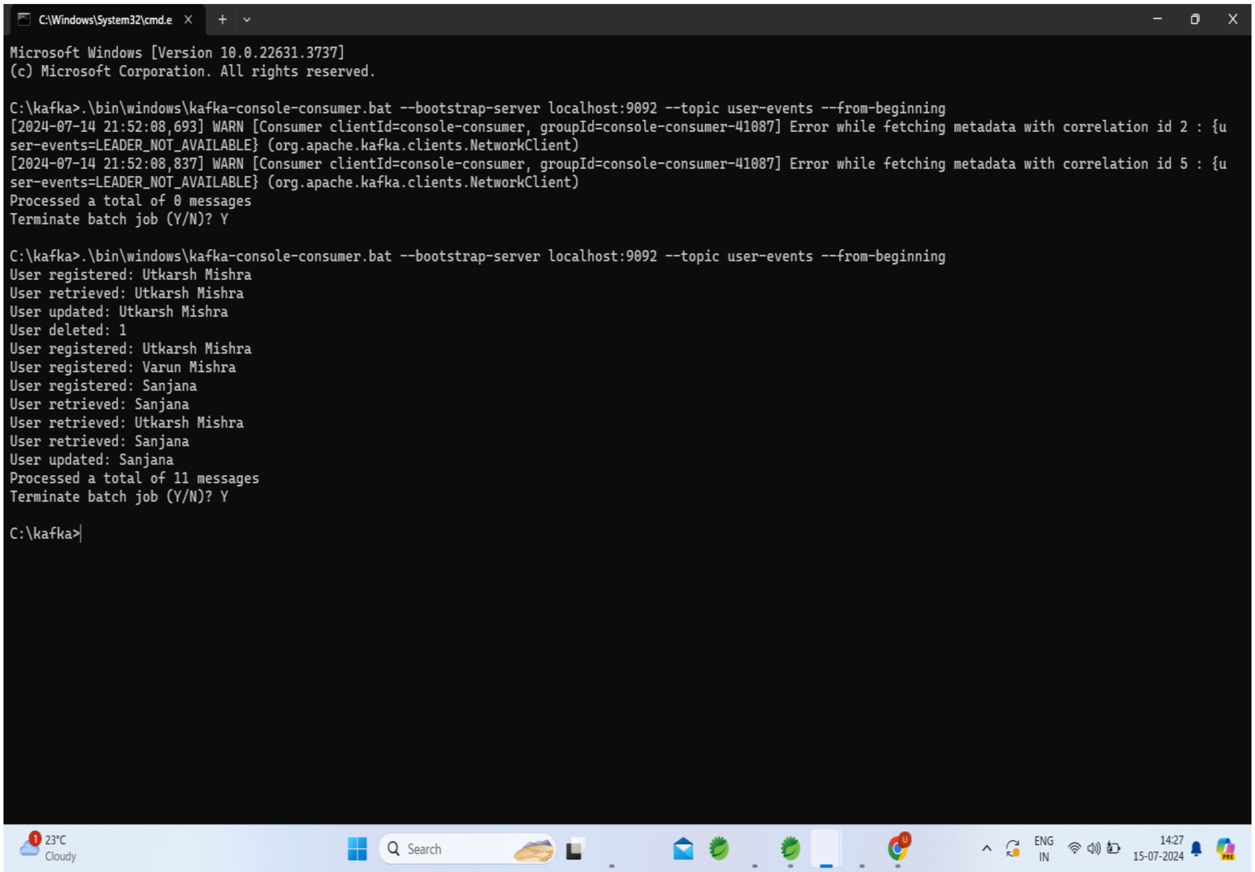
user_service_db.journal: 11 rows total (approximately)

id	message	timestamp
1	User registered: Utkarsh Mishra	2024-07-15 15:45:44.393915
2	User retrieved: Utkarsh Mishra	2024-07-15 15:45:44.424878
3	User updated: Utkarsh Mishra	2024-07-15 15:45:44.426879
4	User deleted: 1	2024-07-15 15:45:44.429879
5	User registered: Utkarsh Mishra	2024-07-15 15:45:44.431880
6	User registered: Varun Mishra	2024-07-15 15:45:44.433880
7	User registered: Sanjana	2024-07-15 15:45:44.435904
8	User retrieved: Sanjana	2024-07-15 15:45:44.436904
9	User retrieved: Utkarsh Mishra	2024-07-15 15:45:44.438881
10	User retrieved: Sanjana	2024-07-15 15:45:44.440887
11	User updated: Sanjana	2024-07-15 15:45:44.442887

```sql
SELECT * FROM `user_service_db`.`role` LIMIT 1000;
SHOW CREATE TABLE `user_service_db`.`role`;
SHOW CREATE TABLE `user_service_db`.`user`;
SELECT * FROM `user_service_db`.`user` LIMIT 1000;
SHOW CREATE TABLE `user_service_db`.`user`;
SHOW CREATE TABLE `user_service_db`.`user_role`;
SELECT * FROM `user_service_db`.`user_role` LIMIT 1000;
SHOW CREATE TABLE `user_service_db`.`user_role`;
SHOW CREATE TABLE `user_service_db`.`journal`;
SELECT * FROM `user_service_db`.`journal` LIMIT 1000;
SHOW CREATE TABLE `user_service_db`.`journal`;
```

-

- Screen Shot Of Kafka Console



```
Microsoft Windows [Version 10.0.22631.3737]
(c) Microsoft Corporation. All rights reserved.

C:\kafka>.\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic user-events --from-beginning
[2024-07-14 21:52:08,693] WARN [Consumer clientId=console-consumer, groupId=console-consumer-41087] Error while fetching metadata with correlation id 2 : {u
ser-events=LEADER_NOT_AVAILABLE} (org.apache.kafka.clients.NetworkClient)
[2024-07-14 21:52:08,837] WARN [Consumer clientId=console-consumer, groupId=console-consumer-41087] Error while fetching metadata with correlation id 5 : {u
ser-events=LEADER_NOT_AVAILABLE} (org.apache.kafka.clients.NetworkClient)
Processed a total of 0 messages
Terminate batch job (Y/N)? Y

C:\kafka>.\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic user-events --from-beginning
User registered: Utkarsh Mishra
User retrieved: Utkarsh Mishra
User updated: Utkarsh Mishra
User deleted: 1
User registered: Utkarsh Mishra
User registered: Varun Mishra
User registered: Sanjana
User retrieved: Sanjana
User retrieved: Utkarsh Mishra
User retrieved: Sanjana
User updated: Sanjana
Processed a total of 11 messages
Terminate batch job (Y/N)? Y

C:\kafka>
```

- To run application

- Create Docker file

- Command :- docker build –t user-service:latest

- Command :- docker run –d –p 8080:8080 –name user-service-container user-service:latest

- Or the project can be run directly from ide

- Run Application → Run As Spring Boot APP