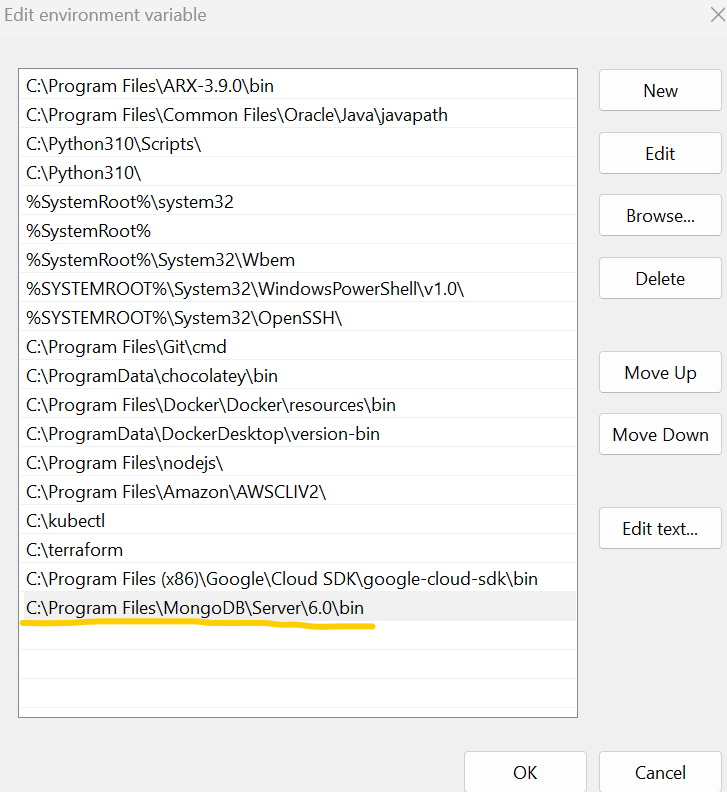
9.1P: Adding a database to your application

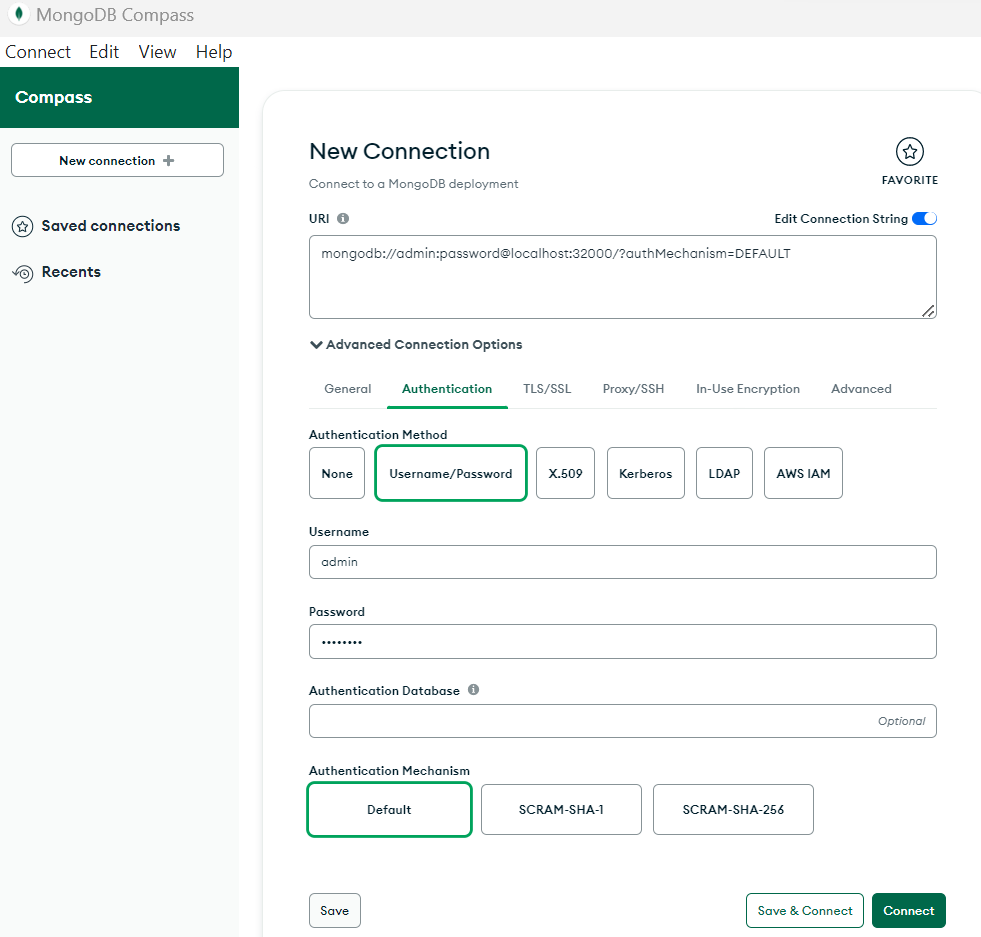
1. Installed MongoDB

* Downloaded version 6.0 of Mongo db and installed the complete version.
* Set the environment variable path to add mongo db’s bin folder location



1. Create a MongoDB user with appropriate permissions for your application.

* Created new connection in MongoDB Compass by using New Connection > Authentication > username as admin and password as password



1. Configurations and commands –

* Configured **persistent** storage for the MongoDB database by creating a Persistent Volume and Persistent Volume Claim.
* Ran the command **kubectl apply -f .** to configure all the yamls at once
* Configured Kubernetes Secret file for the MongoDB user credentials and added them to the deployment manifest. Secret is an object that contains sensitive information such as password etc.

There is a small yaml to configure this –

apiVersion: v1

kind: Secret

metadata:

  name: mongodb-secret

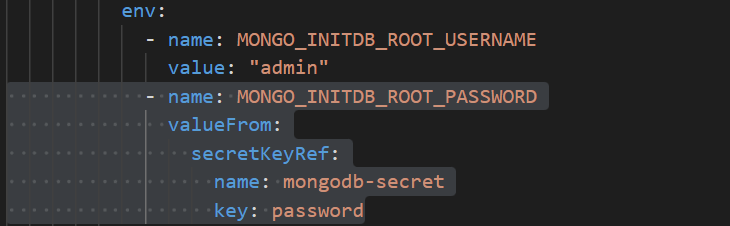
immutable: false

type: Opaque

data:

  password: cGFzc3dvcmQ

- Added MongoDB **secret** to deployment manifest.

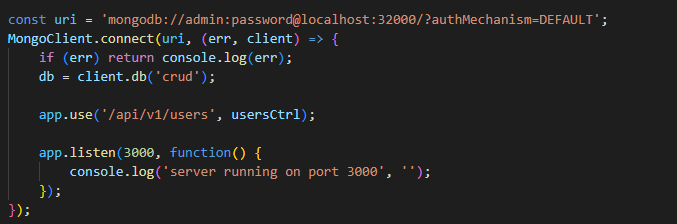
-

* **MongoDb** is configured in server.js by -

Imported mongodb library as –->

const MongoClient = require('mongodb').MongoClient;

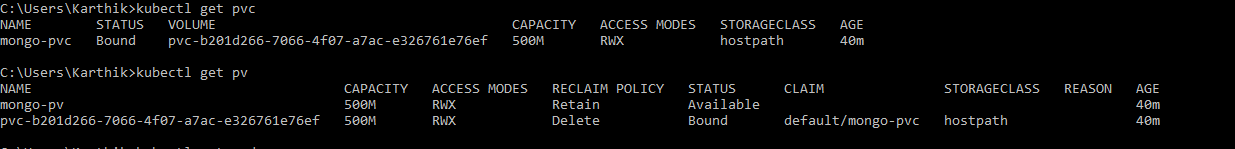
and then used the connection url shown as ‘uri’ below. Database name is ‘crud’ -



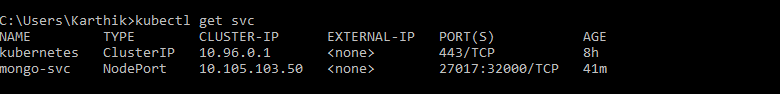
1. Below are the screenshots for all the created configurations explained above and execurted with command **kubectl apply -f .**

These screenshot demonstrate successful creation of pv, pvc, etc that happened by running the kubectl get commands–

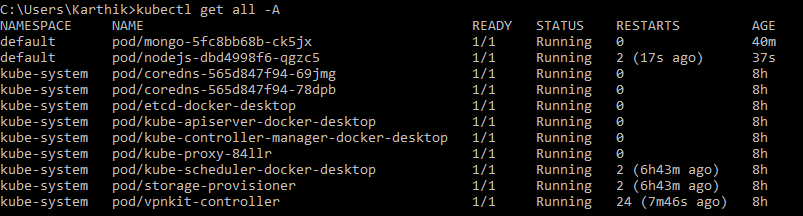
Pv and pvc –



Svc-



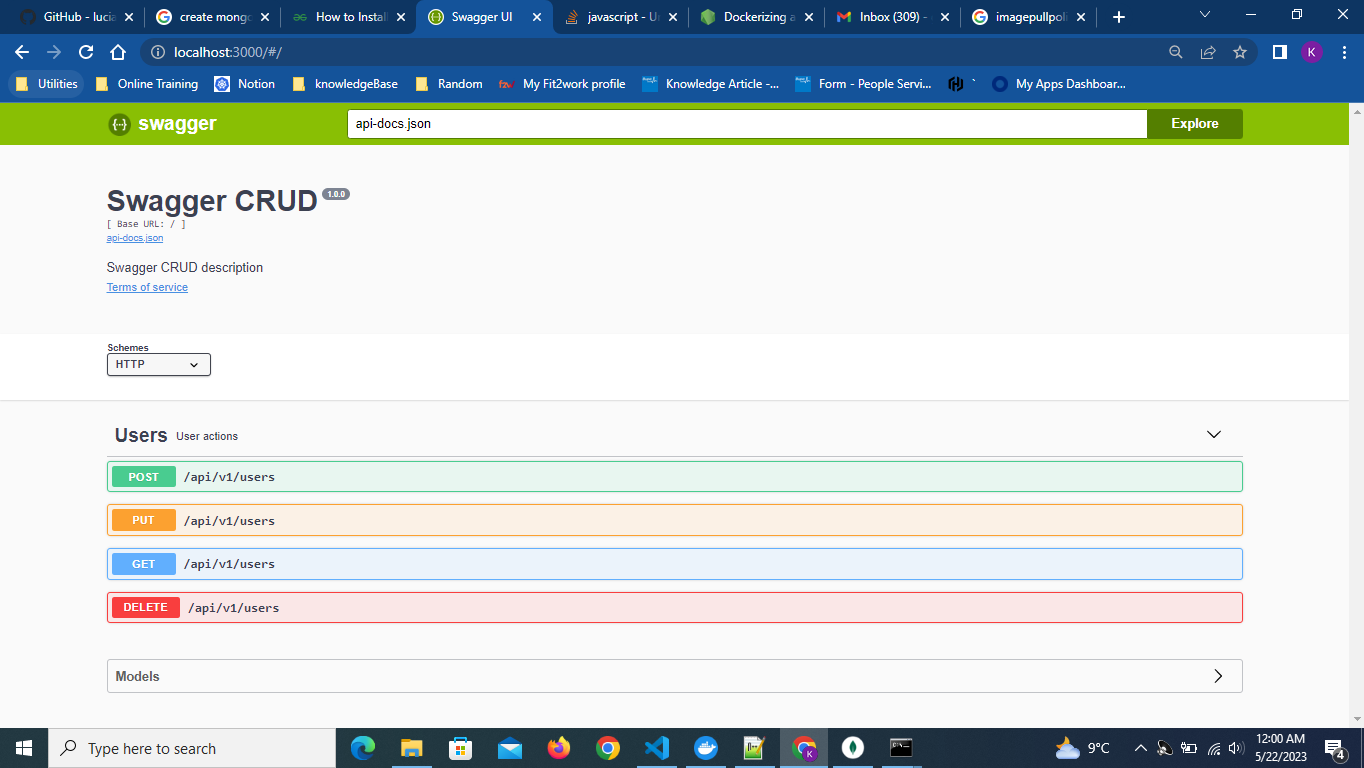
Pods-



**Both mongodb pod and nodejs pods are running**

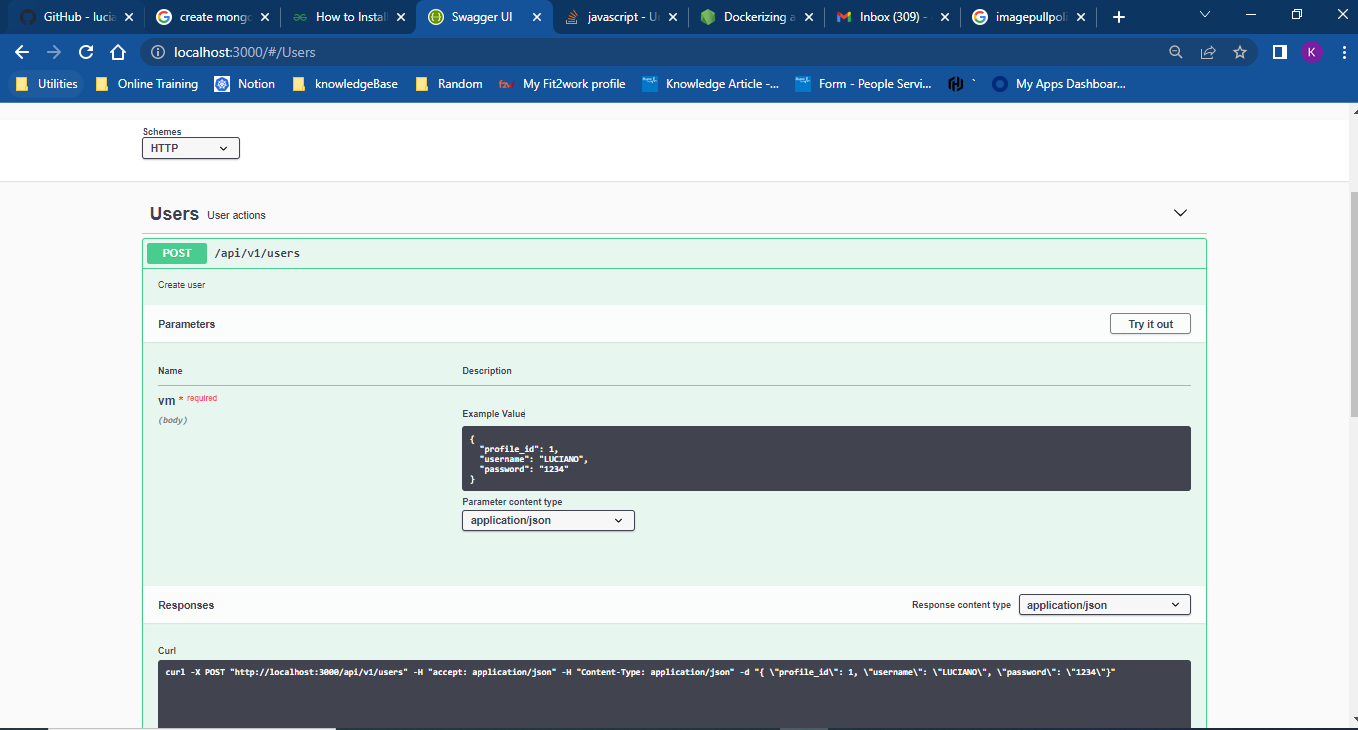
1. Tested the deployment for CRUD (Create, Read, Update, Delete) operations

**Launched localhost:3000 as –**

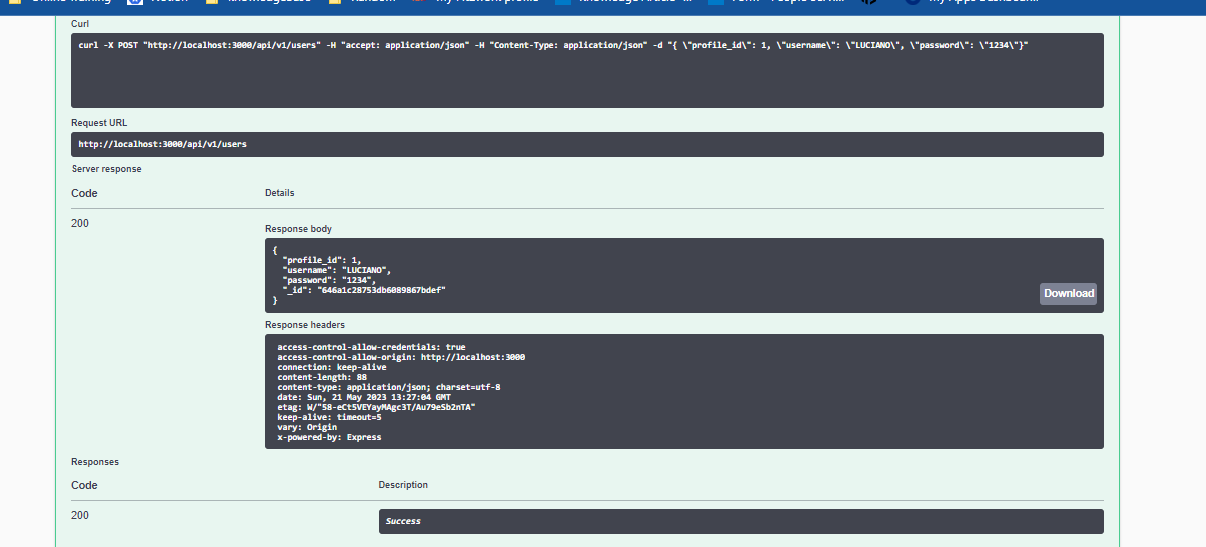


**Step 1.** Testing POST - /Create endpoint. The input and output screenshots are -

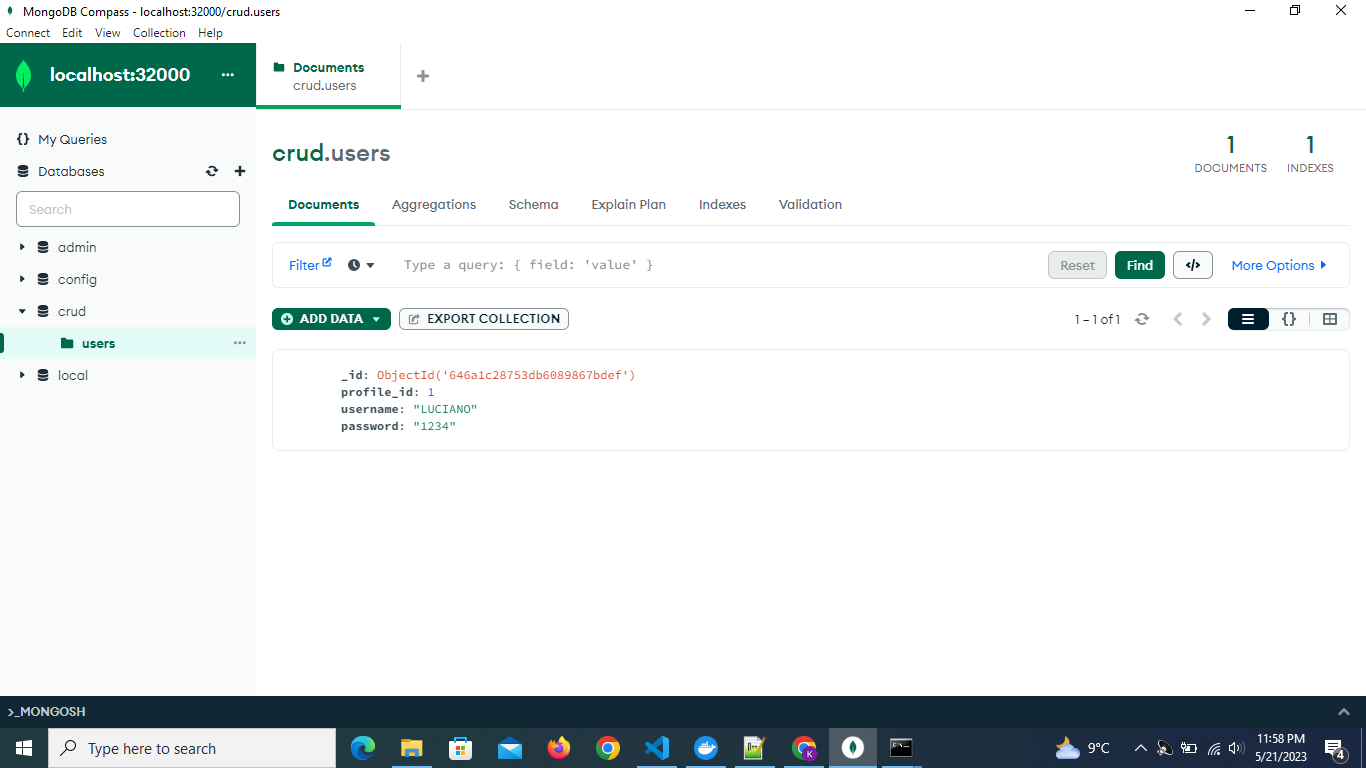
Inputted the body in POST as follows-



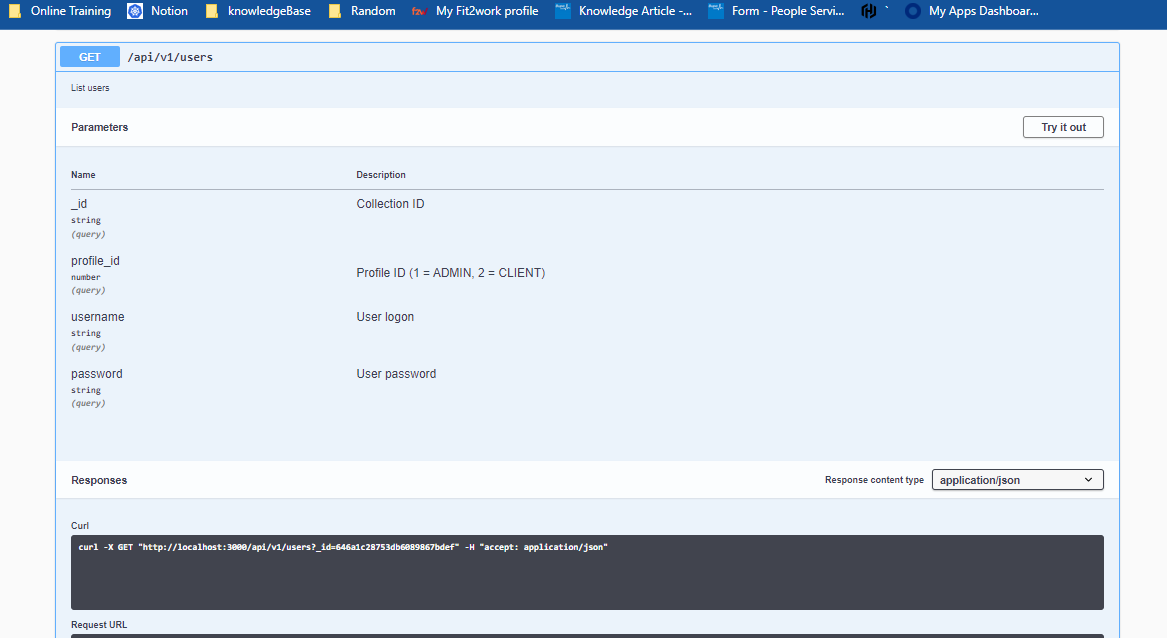
The output 200 was received as –

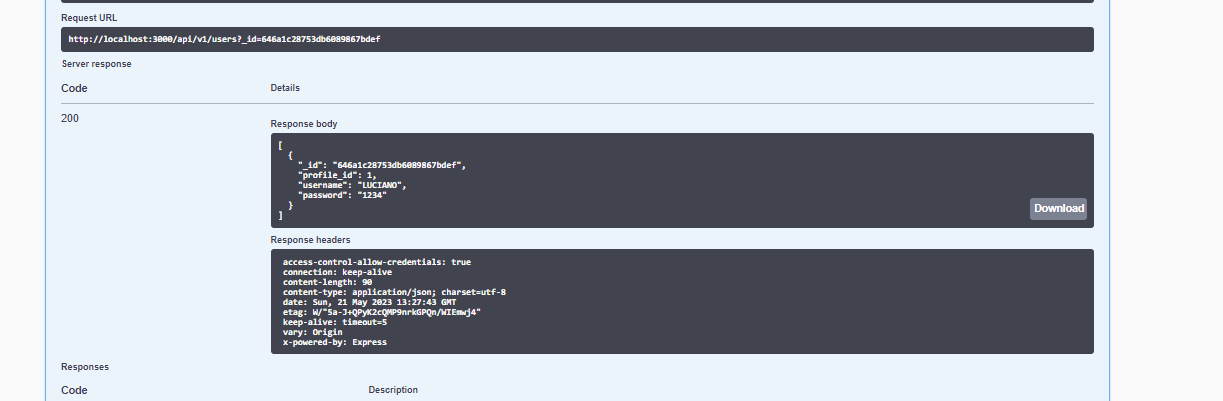


**The Mongo db compass resultantly showed –**



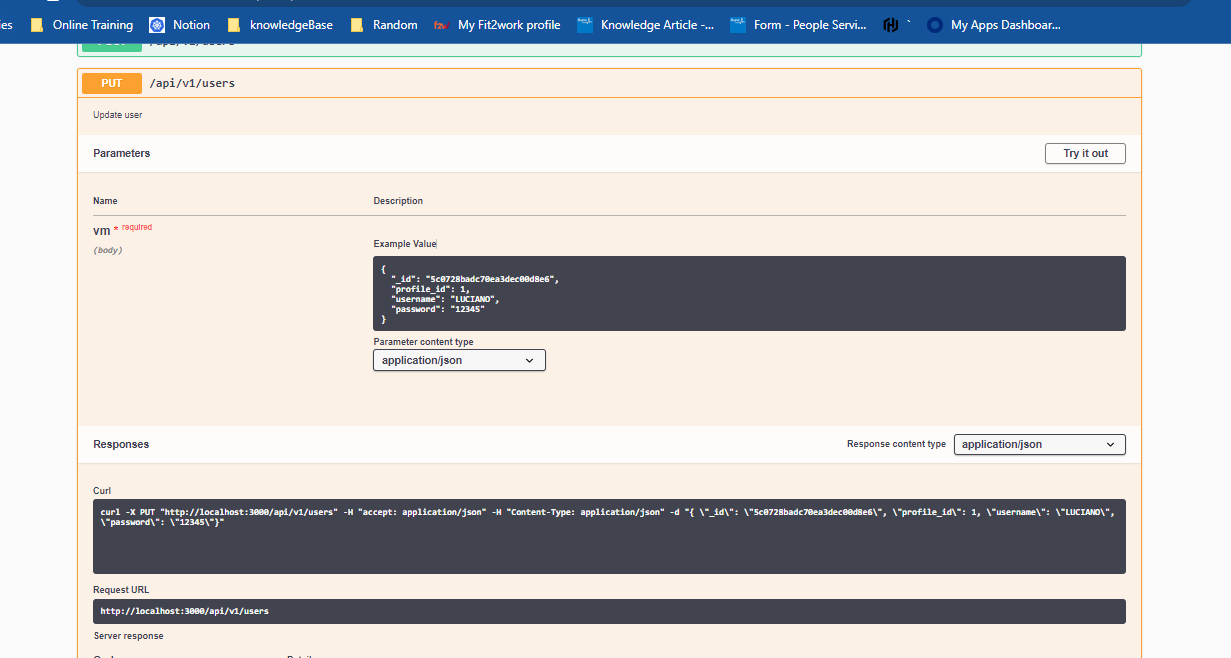
**Step 2.** GET operation –





**Step 3.** PUT operation – Changed id from 1234 to 12345 for the object id created in the output of POST operation.

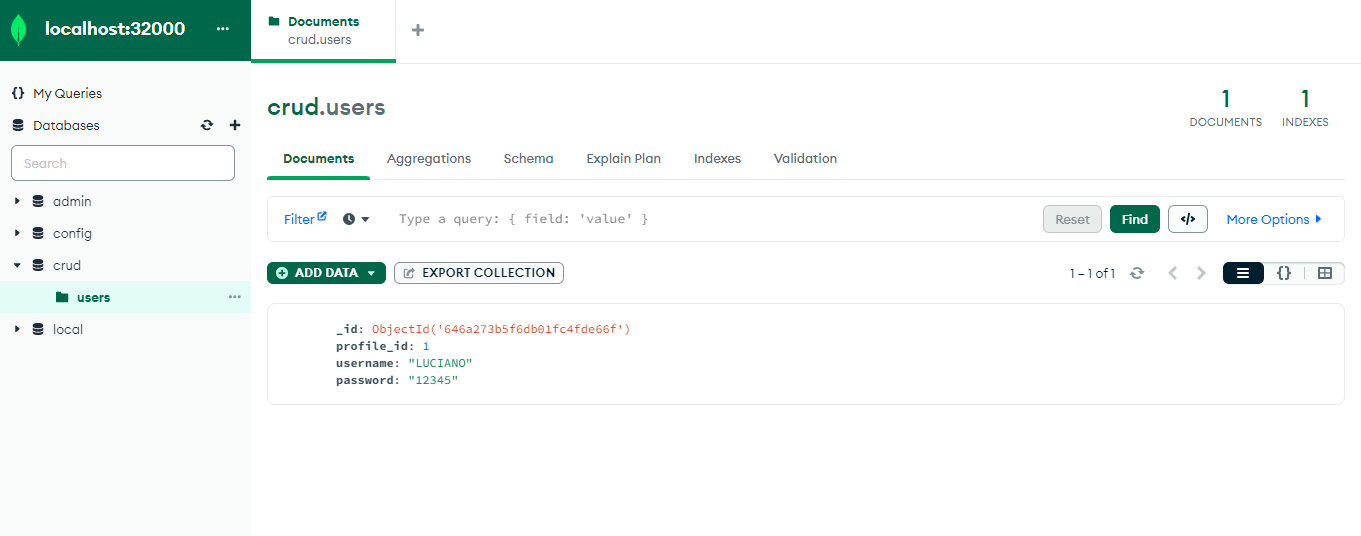
Input –



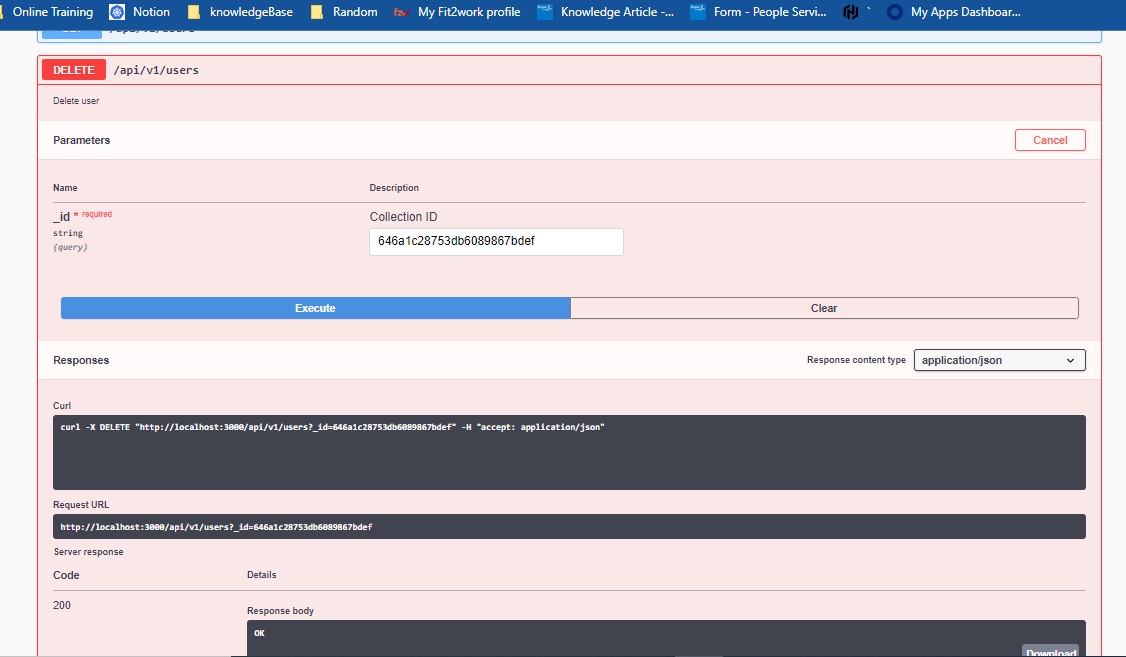
Output –



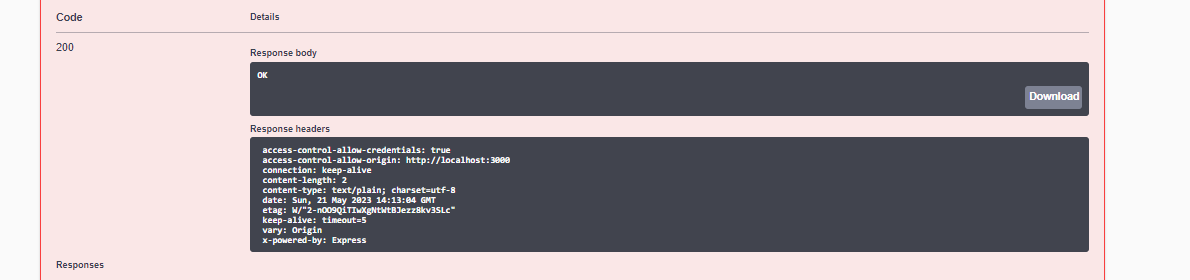
Corresponding change in db –



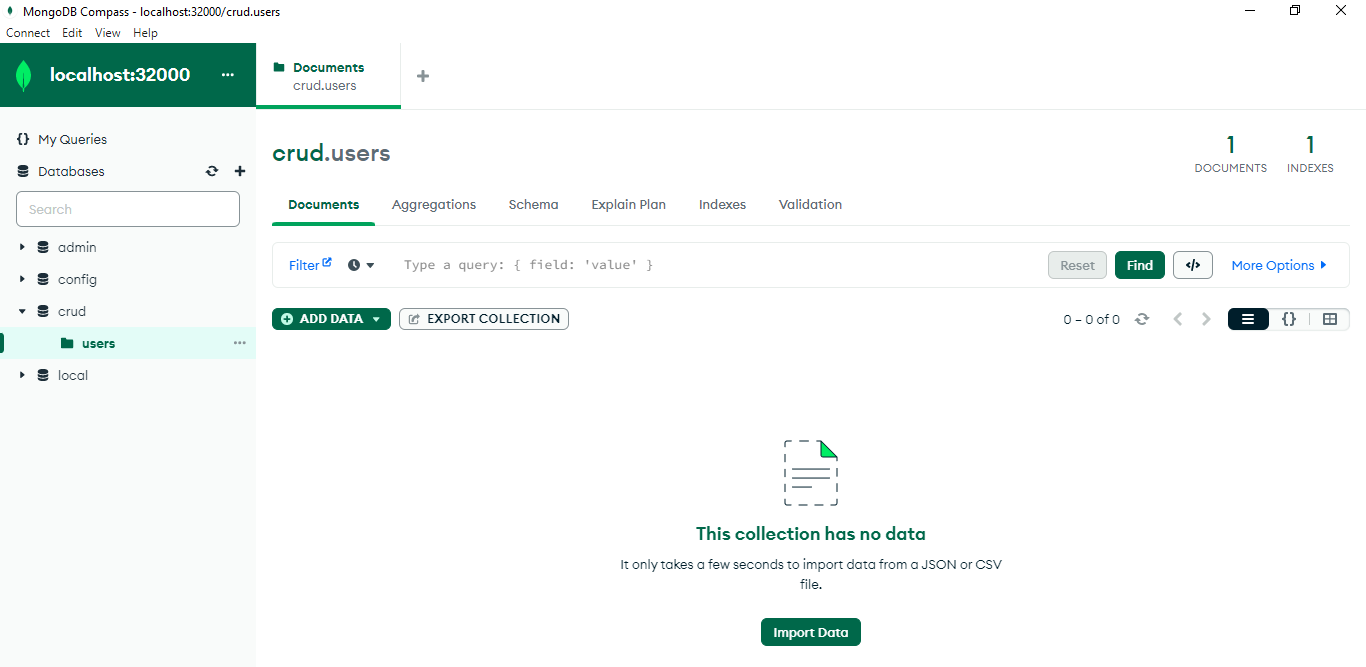
**Step** **4.** DELETE operation using the id



Deletion success code 200 -



Effect on mongodb compass –



**Note**:

In the code submitted, all the Kubernetes code and configurations are done as per my understanding from the workshop content.

CRUD operations are learnt from research, mainly from the source – [https://github.com/lucianopereira86/NodeJS-MongoDB-Kubernetes/tree/master]