

220531770_61_bootcamp

Pasos para desplegar el proyecto

Obtener el proyecto de github

```
git clone https://github.com/esancaro/220531770_61_bootcamp
cd 220531770_61_bootcamp
```

Correr el siguiente comando docker. Se utilizó yaml para automatizar la creación de ambos servicios y que la red de mongodb esté disponible para nodejs.

```
docker compose up -d --build
```

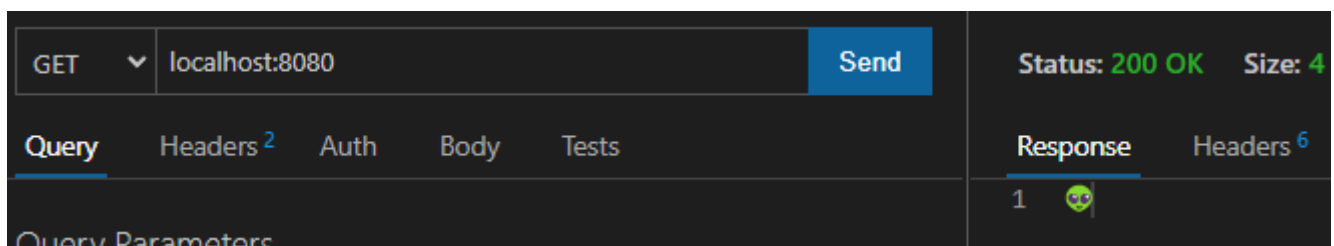
Dar de baja el servicio

Para dar de baja el servicio y borrar los contenedores.

```
docker-compose down
```

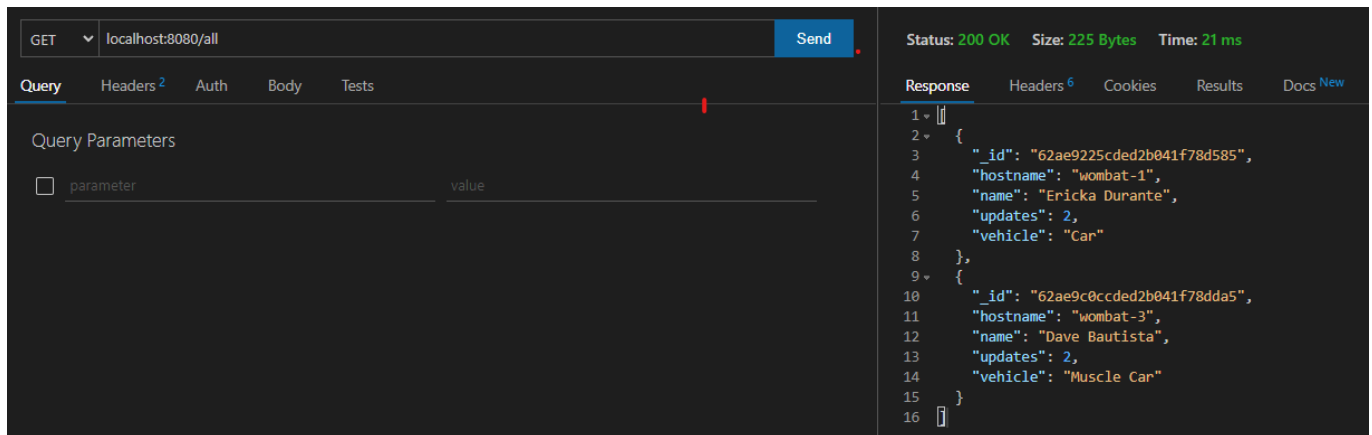
Opciones del sistema

Raíz del servicio, desde una máquina local: **localhost:8080**. Despliega un comodín: 🕒



GET /all :

- Parámetros: ninguno
- Devuelve: un JSON con la lista de todos los objetos de la base de datos. Cuando la base de datos es nueva, devuelve un arreglo vacío: []



GET localhost:8080/all Send

Status: 200 OK Size: 225 Bytes Time: 21 ms

Query Parameters

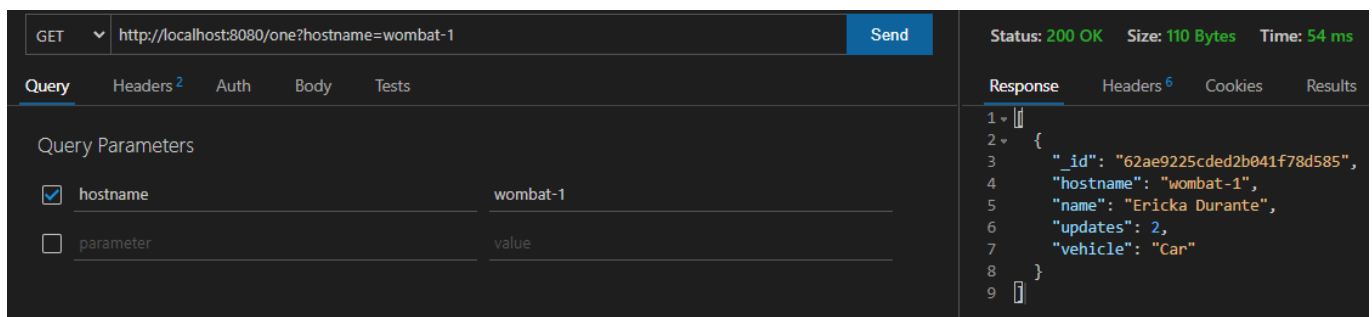
parameter	value
-----------	-------

Response

```
1 [
2   {
3     "_id": "62ae9225cded2b041f78d585",
4     "hostname": "wombat-1",
5     "name": "Ericka Durante",
6     "updates": 2,
7     "vehicle": "Car"
8   },
9   {
10    "_id": "62ae9c0ccded2b041f78dda5",
11    "hostname": "wombat-3",
12    "name": "Dave Bautista",
13    "updates": 2,
14    "vehicle": "Muscle Car"
15  }
16 ]
```

GET /one :

- Parámetros: requiere el hostname en la cadena get. Ejemplo: ?hostname=wombat-1
- Devuelve: JSON con el detalle del registro. Cuando la base de datos es nueva o no encuentra la opción solicitada, devuelve un arreglo vacío: []



GET http://localhost:8080/one?hostname=wombat-1 Send

Status: 200 OK Size: 110 Bytes Time: 54 ms

Query Parameters

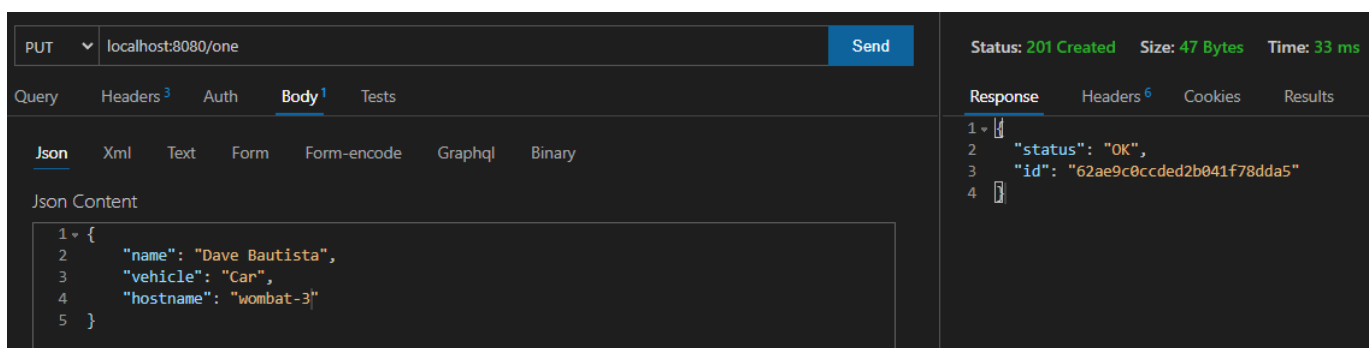
parameter	value
<input checked="" type="checkbox"/> hostname	wombat-1
<input type="checkbox"/> parameter	value

Response

```
1 [
2   {
3     "_id": "62ae9225cded2b041f78d585",
4     "hostname": "wombat-1",
5     "name": "Ericka Durante",
6     "updates": 2,
7     "vehicle": "Car"
8   }
9 ]
```

PUT /one :

- Parámetros: JSON con el detalle del registro a insertar. Los valores posibles son: name, vehicle, hostname. Si se omite hostname, este será insertado automáticamente con el nombre del host del servidor (no tiene sentido práctico).
- Devuelve: estructura con status 201 'OK', id del registro en caso de inserción o status 200 'OK' con id null en caso de actualización.



PUT localhost:8080/one Send

Status: 201 Created Size: 47 Bytes Time: 33 ms

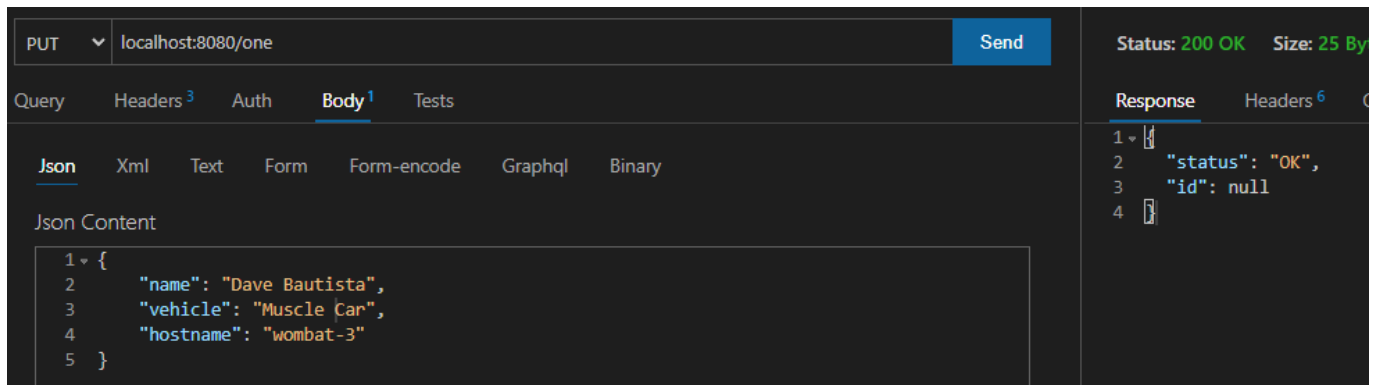
Query Headers Auth Body Tests

Json Content

```
1 {
2   "name": "Dave Bautista",
3   "vehicle": "Car",
4   "hostname": "wombat-3"
5 }
```

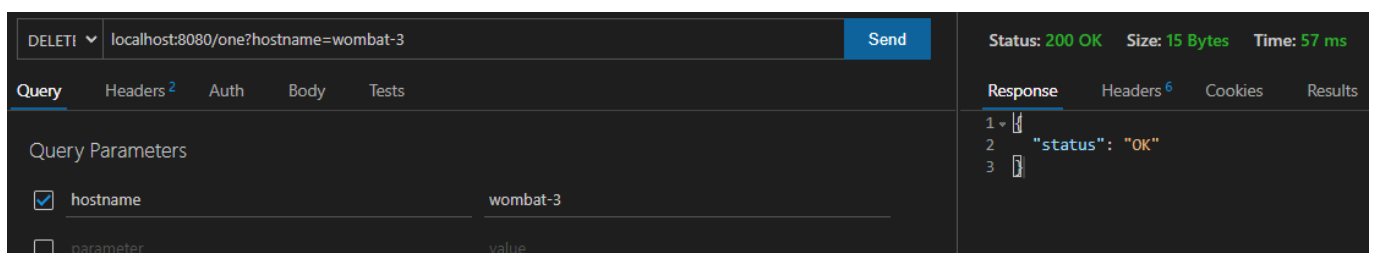
Response

```
1 {
2   "status": "OK",
3   "id": "62ae9c0ccded2b041f78dda5"
4 }
```



DELETE /one :

- Parámetros: Cadena GET con el hostname. Ejemplo: ?hostname=wombat-3
- Devuelve: status 200 'OK' y estructura JSON con el valor {"status": "OK"}



Manejo de errores

El sistema devuelve una estructura JSON con la estructura:

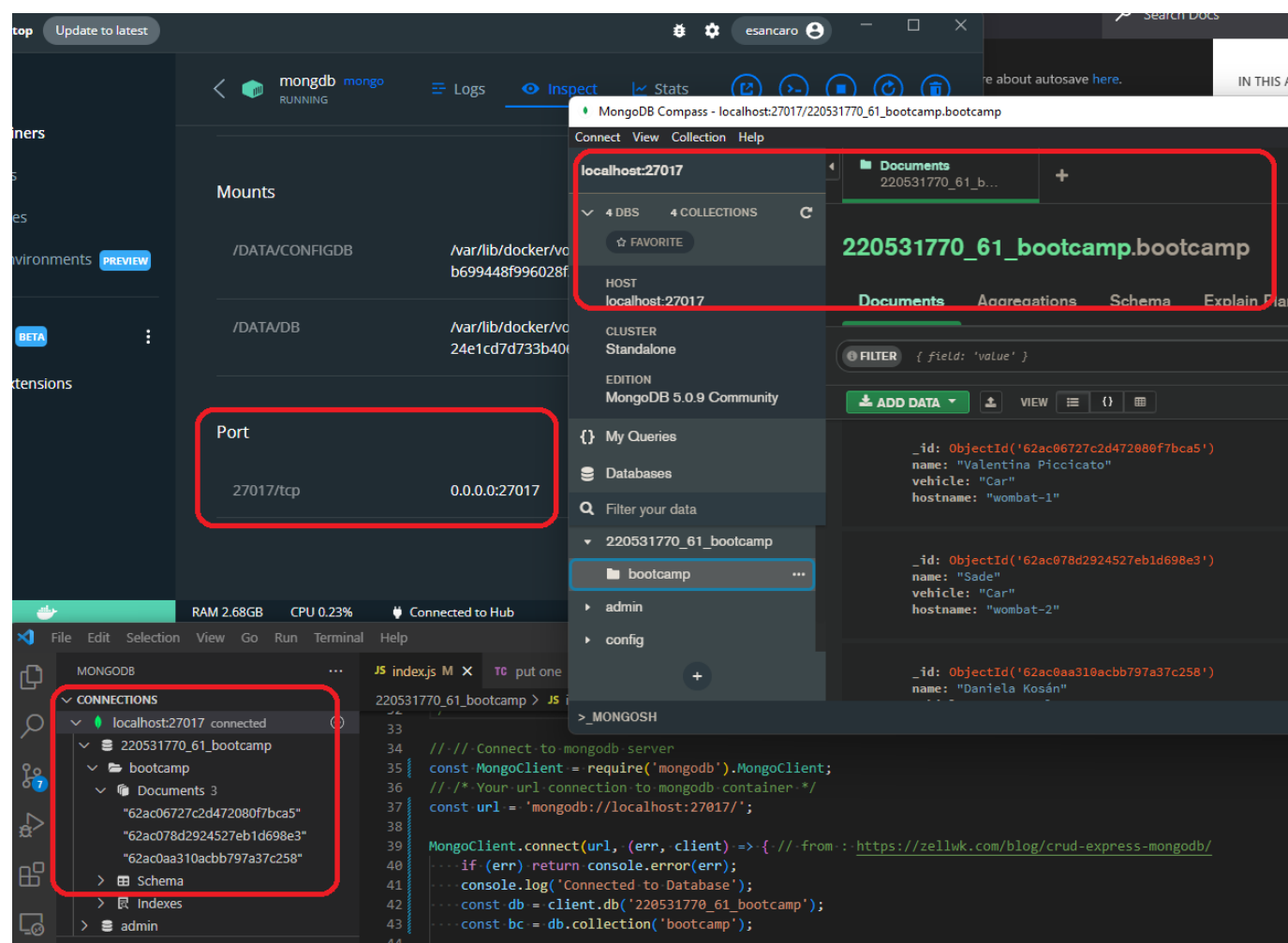
```
{"status": "NOK", "message": "<detalle del error>"}
```

Detalles a considerar

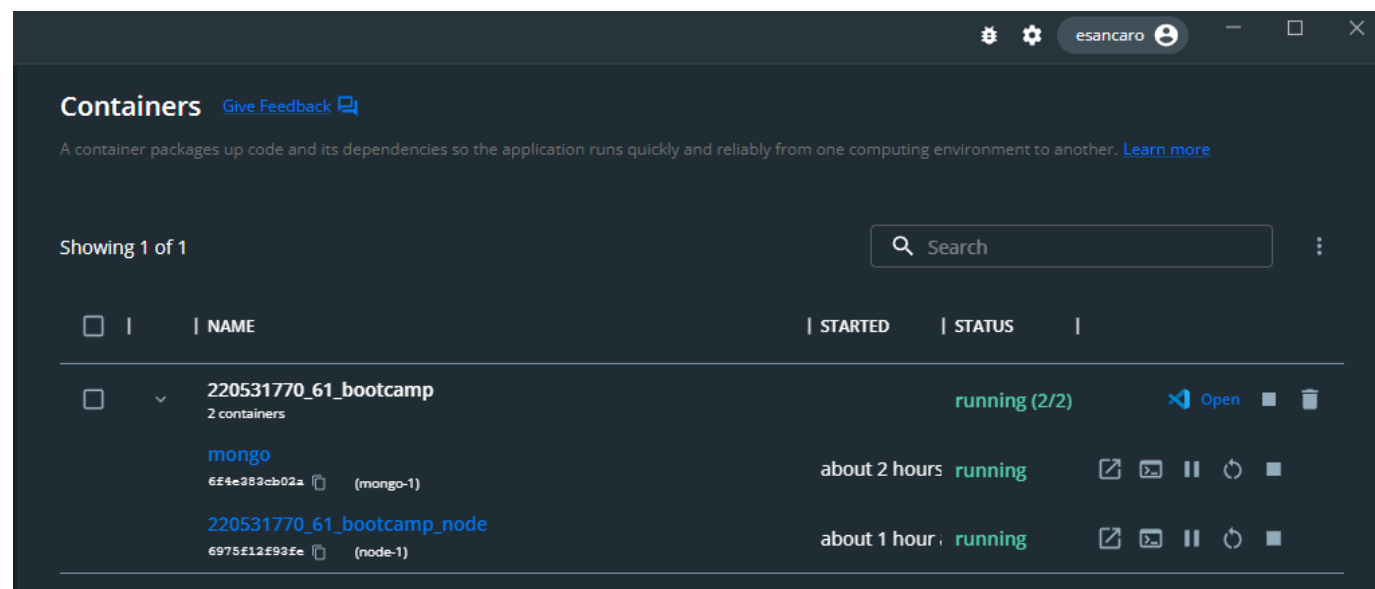
- Si desea correr el programa con el depurador local, por favor cambiar, en server.js, la constante mongoHost = "mongo" a mongoHost = "localhost". Esta debe apuntar a mongo para satisfacer el requerimiento de la configuración de docker compose como se estableció en el archivo docker-compose.yml.

Pantallas de programas instalados y configuraciones

Mongo



Detalle del grupo de contenedores de compose. Mongo y Node.



kubectl

```
# cd C:\Users\esanc\opt

# kubectl.exe
kubectl controls the Kubernetes cluster manager.

Find more information at: https://kubernetes.io/docs/reference/kubectl/overview/

Basic Commands (Beginner):
  create      Create a resource from a file or from stdin
  expose      Take a replication controller, service, deployment or pod and expose
  run         Run a particular image on the cluster
  set         Set specific features on objects
```

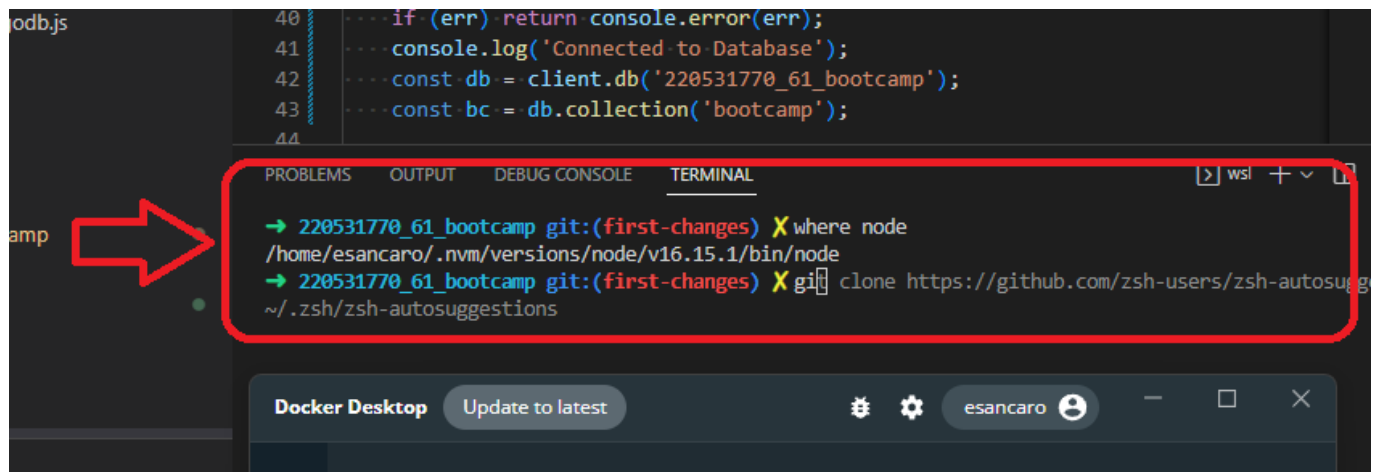
minikube

```
Command Prompt
Microsoft Windows [Version 10.0.22000.739]
(c) Microsoft Corporation. All rights reserved.

C:\Users\esanc>minikube
minikube provisions and manages local Kubernetes clusters optimized for development work

Basic Commands:
  start      Starts a local Kubernetes cluster
  status     Gets the status of a local Kubernetes cluster
  stop       Stops a running local Kubernetes cluster
  delete     Deletes a local Kubernetes cluster
  dashboard  Access the Kubernetes dashboard running within the minikube cluster
  pause      pause Kubernetes
  unpause    unpause Kubernetes
```

Línea de comandos con zsh



Referencias

- Conocimiento general
 - <https://zellwk.com/blog/crud-express-mongodb/>
 - <https://www.mongodb.com/docs/manual/reference/method/db.collection.updateOne/>
 - <https://www.mongodb.com/docs/manual/tutorial/remove-documents/>
 - <https://nodejs.org/api/process.html>
- Para docker compose. Conectividad es posible configurando los contenedores con esta opción.
 - <https://stackoverflow.com/questions/66803166/i-have-mongo-db-in-a-docker-container-and-i-want-to-connect-my-app-which-is-run>

- <https://www.bogotobogo.com/DevOps/Docker/Docker-Compose-Node-MongoDB.php>
- <https://docs.docker.com/compose/>
- <https://stackoverflow.com/questions/51011552/mongodb-on-with-docker-failed-to-connect-to-server-localhost27017-on-first-c>