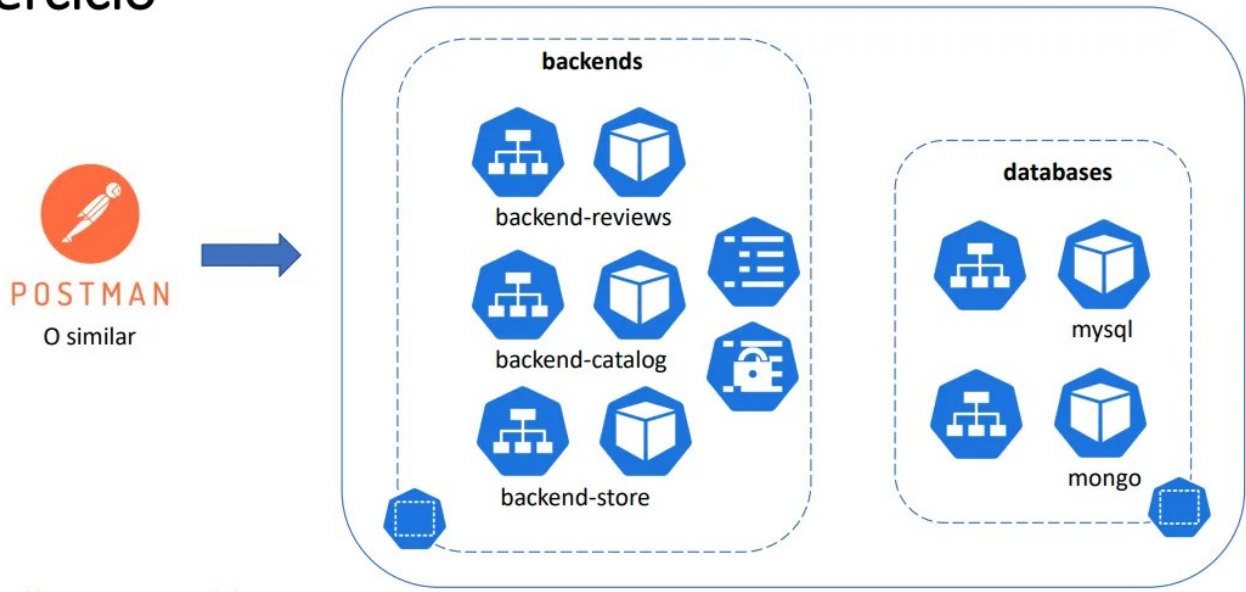


Desplegando bookstore con kubernetes

Integrantes:

- Juan Camilo Lozada Garavito – 2205560
- Daniel Camilo Barrera Pérez – 2205562
- Jerson Julian Cañon Castillo – 2205633
- Santiago González Flores - 2200165

Ejercicio



1. Creamos los namespaces backends y databases

```
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl create namespace backends
namespace/backends created
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl create namespace databases
namespace/databases created
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl get namespaces
NAME                STATUS    AGE
backends             Active    35s
databases            Active    28s
default              Active    44d
kube-node-lease      Active    44d
kube-public          Active    44d
kube-system          Active    44d
```

2. Configuramos los archivos .yaml para las bases de datos y creamos los recursos asociados a estos

Para mysql:

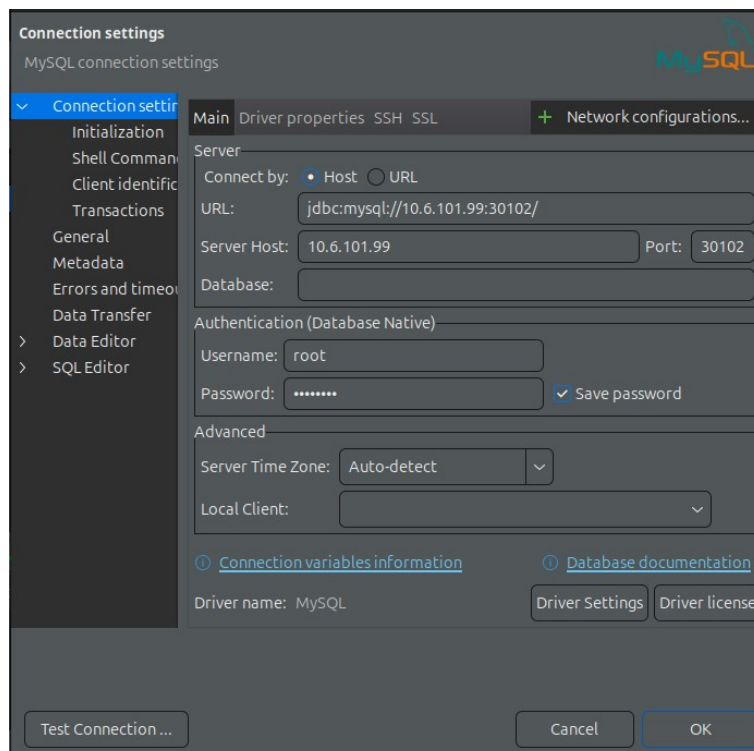
```
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl apply -f mysql.yaml --namespace=databases
deployment.apps/mysql-deployment created
service/mysql-service created
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl get pod -n databases
NAME                                READY   STATUS    RESTARTS   AGE
mysql-deployment-846b49bf59-mrmm6   1/1     Running   0           2m15s
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl get services -n databases
NAME          TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
mysql-service NodePort    10.152.183.161 <none>        3306:30102/TCP   2m22s
```

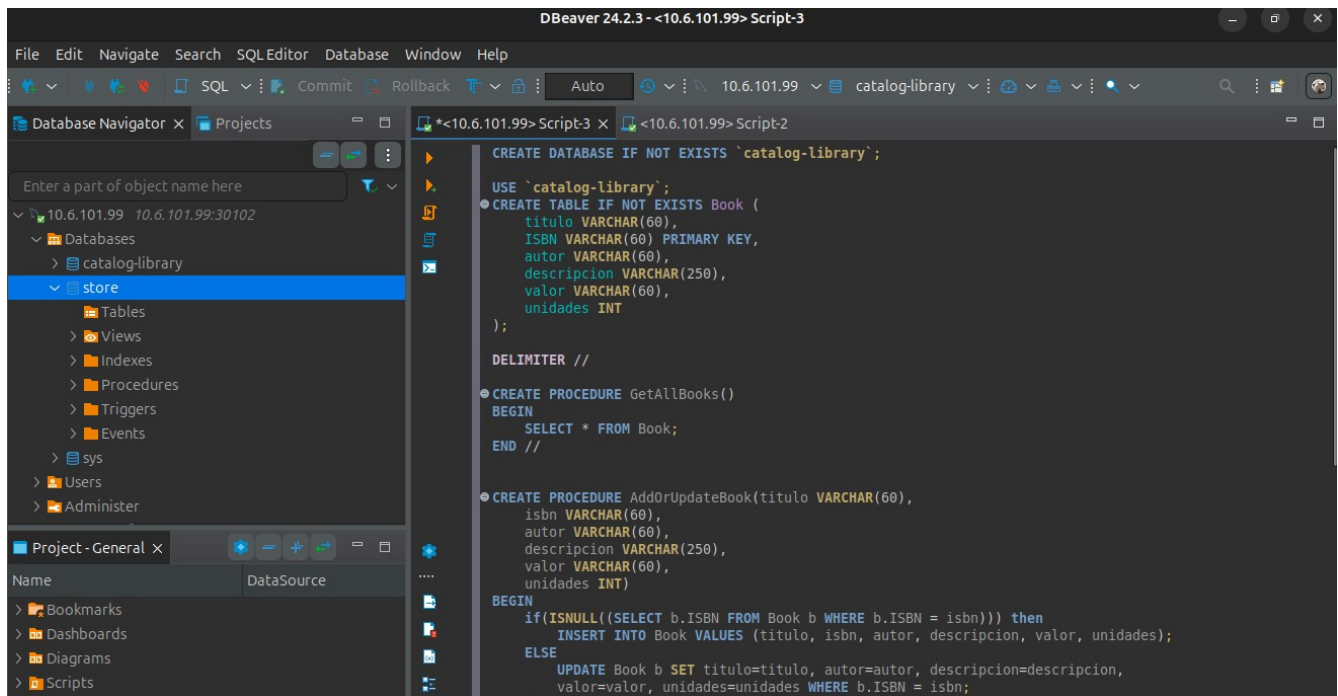
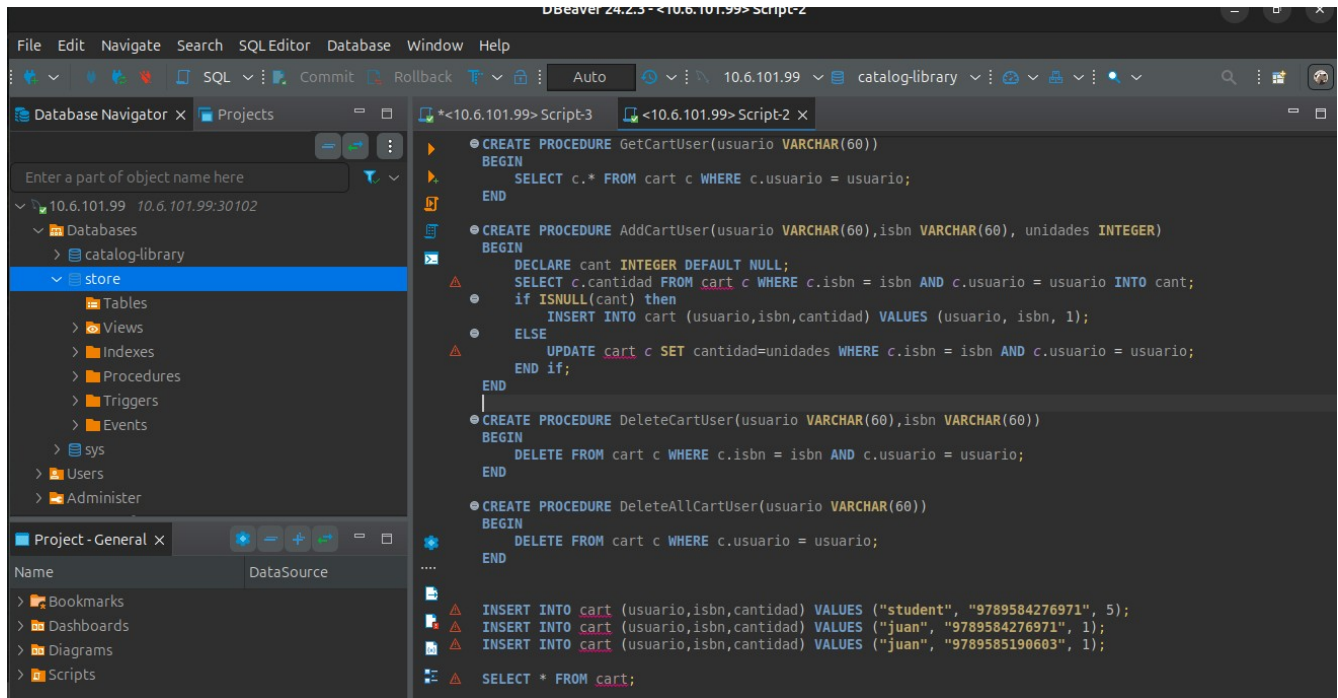
Para mongoDB:

```
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl apply -f mongo.yaml --namespace=databases
deployment.apps/mongo-deployment created
service/mongo-service created
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl get pods -n databases
NAME                                READY   STATUS    RESTARTS   AGE
mongo-deployment-56446cc9c9-pmpdf   1/1     Running   0           3m19s
mysql-deployment-846b49bf59-mrmm6   1/1     Running   0           12m
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl get services -n databases
NAME          TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
mongo-service NodePort    10.152.183.138 <none>        27017:30101/TCP  3m27s
mysql-service NodePort    10.152.183.161 <none>        3306:30102/TCP   12m
```

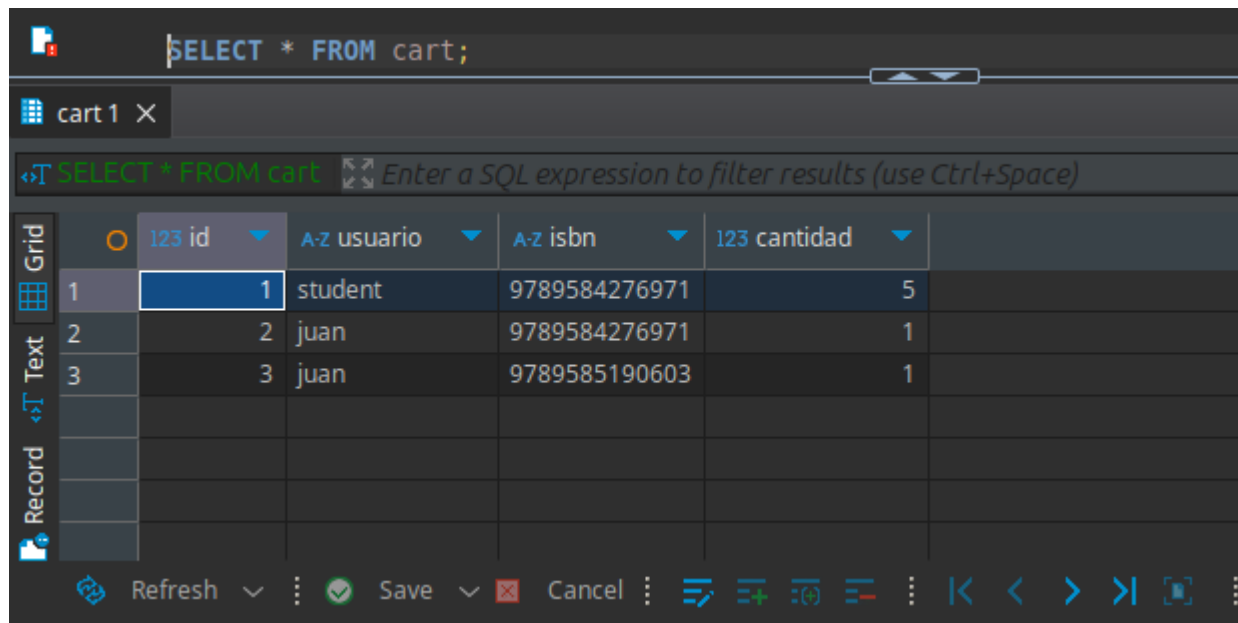
3. Cargamos los datos

Para las bases de datos relacionales, lo hacemos conectándonos desde Dbeaver y ejecutando los scripts



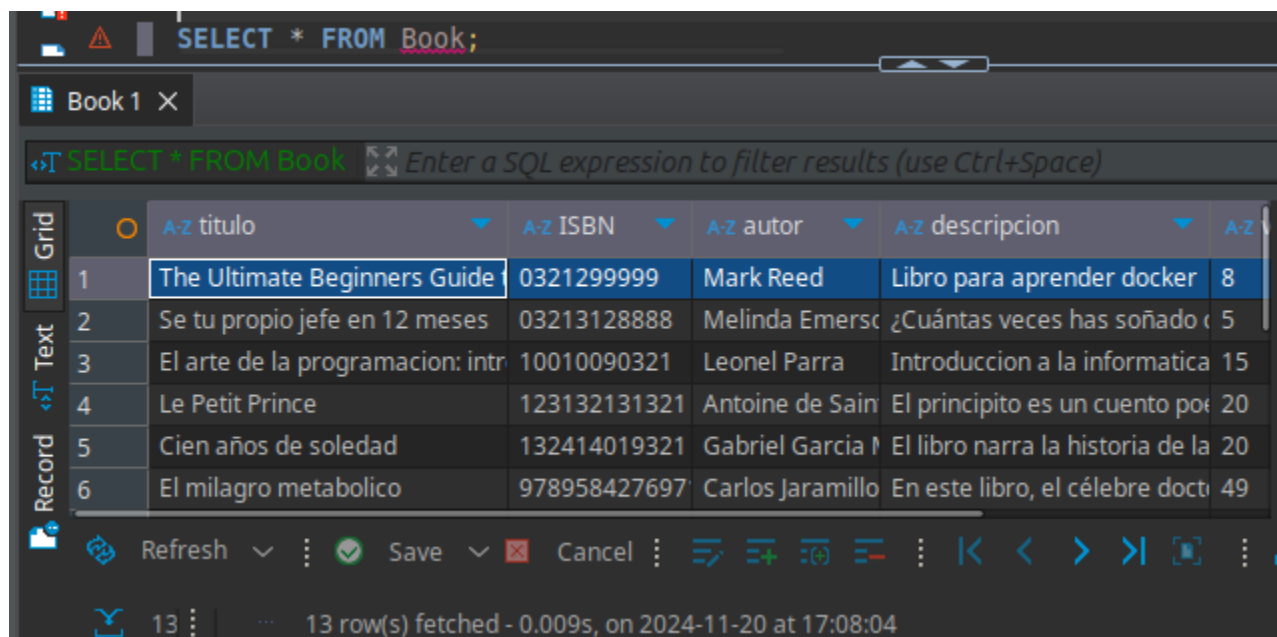


Podemos hacer una consulta para comprobar que los datos se cargaron



The screenshot shows a database client interface with the SQL query `SELECT * FROM cart;` executed. The results are displayed in a table with 5 columns: `id`, `usuario`, `isbn`, and `cantidad`. The first row is highlighted.

	id	usuario	isbn	cantidad
1	1	student	9789584276971	5
2	2	juan	9789584276971	1
3	3	juan	9789585190603	1



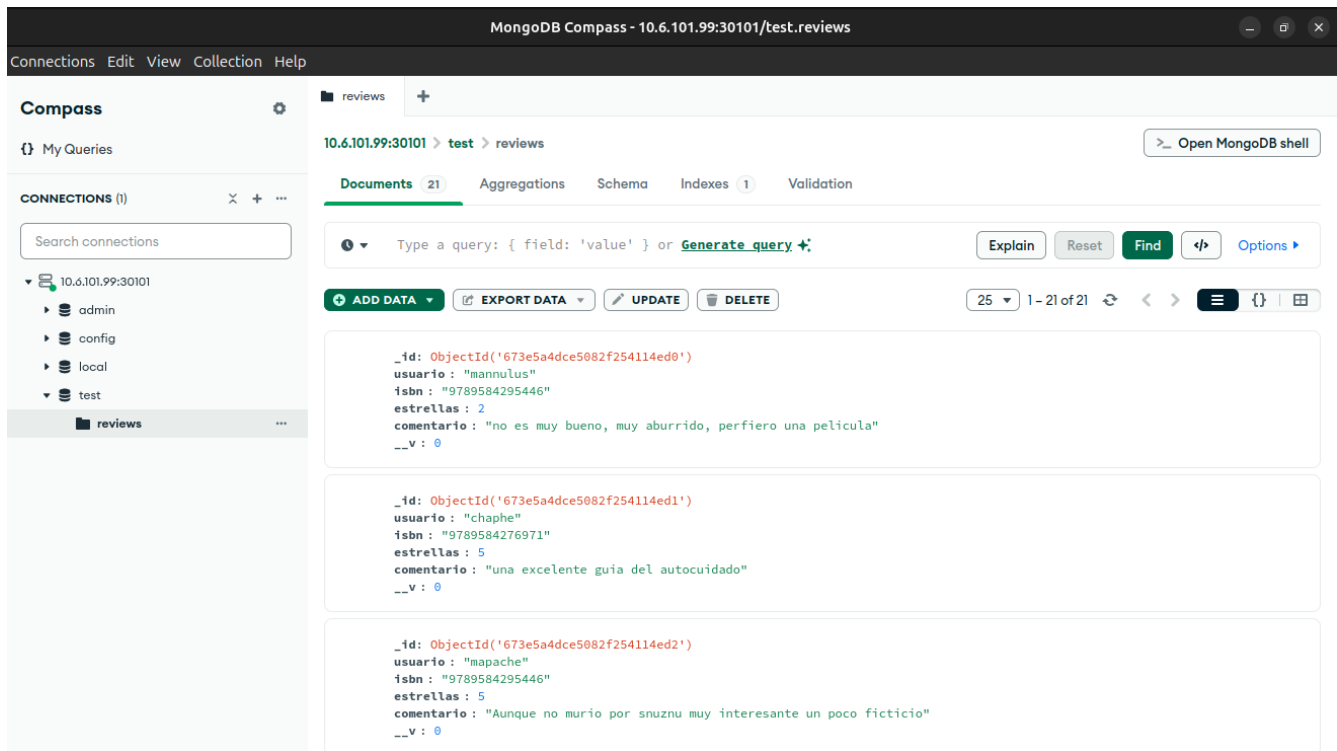
The screenshot shows a database client interface with the SQL query `SELECT * FROM Book;` executed. The results are displayed in a table with 6 columns: `titulo`, `ISBN`, `autor`, `descripcion`, and `cantidad`. The first row is highlighted.

	titulo	ISBN	autor	descripcion	cantidad
1	The Ultimate Beginners Guide	0321299999	Mark Reed	Libro para aprender docker	8
2	Se tu propio jefe en 12 meses	03213128888	Melinda Emerson	¿Cuántas veces has soñado con	5
3	El arte de la programacion: introduccion a la informatica	10010090321	Leonel Parra	Introduccion a la informatica	15
4	Le Petit Prince	123132131321	Antoine de Saint-Exupéry	El principito es un cuento poe	20
5	Cien años de soledad	132414019321	Gabriel Garcia Marquez	El libro narra la historia de la	20
6	El milagro metabolico	978958427697	Carlos Jaramillo	En este libro, el célebre docto	49

Para mongo es necesario instalar nodeJs y npm y correr este script reviews-script.js. Esto lo hacemos ejecutando estos comando

```
npm init
npm i mongoose
node reviews-script.js
```

Podemos comprobar que los datos se carguen desde mongoDB Compass



4. Configuramos los archivos .yaml para los backends y creamos los recursos asociados a estos

```
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl apply -f backend-catalog.yaml --namespace=backends
deployment.apps/catalog-backend-deployment created
service/catalog-backend-service created
```

```
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl apply -f backend-reviews.yaml --namespace=backends
deployment.apps/reviews-backend-deployment created
service/reviews-backend-service created
```

```
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl apply -f backend-store.yaml --namespace=backends
deployment.apps/backend-store-deployment created
service/backend-store-service unchanged
```

5. Configuramos los archivos .yaml para los secrets y los config creando los recursos asociados a estos

```
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl apply -f config-mongo.yaml --namespace=backends
configmap/mongo-config created
```

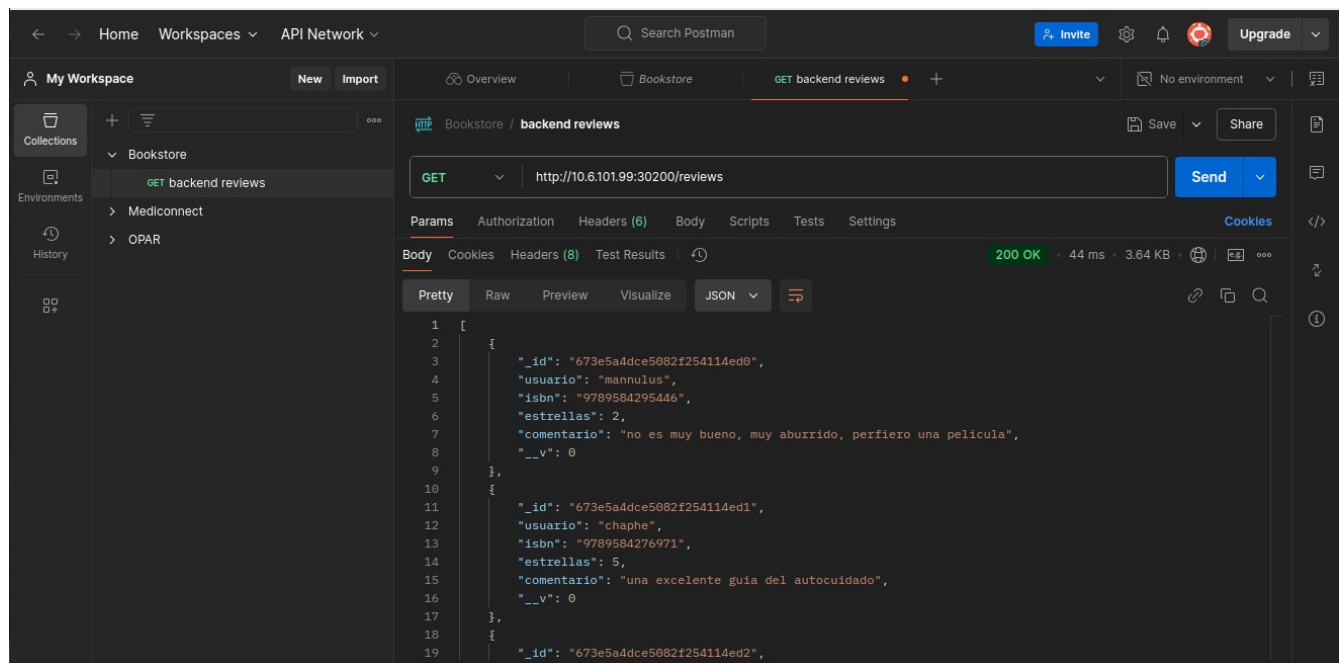
```
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl apply -f secret-mongo.yaml --namespace=backends
secret/mongo-secret created
```

```
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl apply -f config-mysql.yaml --namespace=backends
configmap/mysql-config created
```

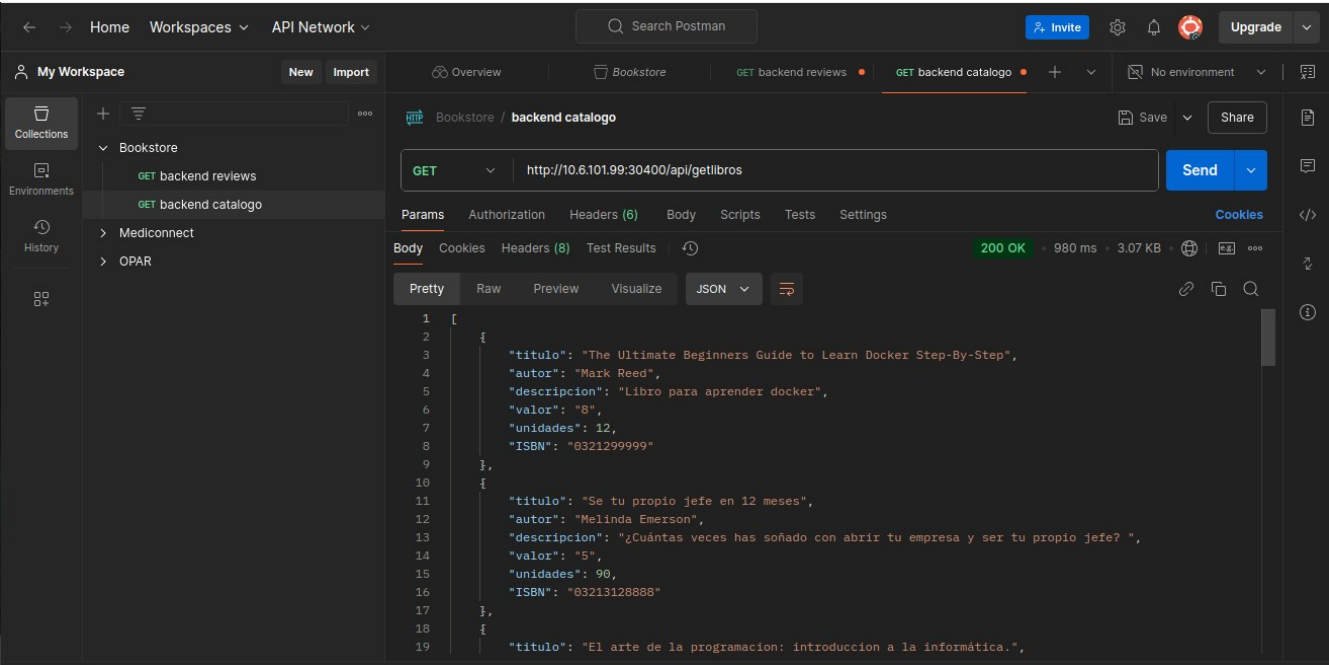
```
student@student99:~/kubernetes-bookstore$ sudo microk8s kubectl apply -f secret-mysql.yaml --namespace=backends
secret/mysql-secret created
```

6. Probamos los backends con postman

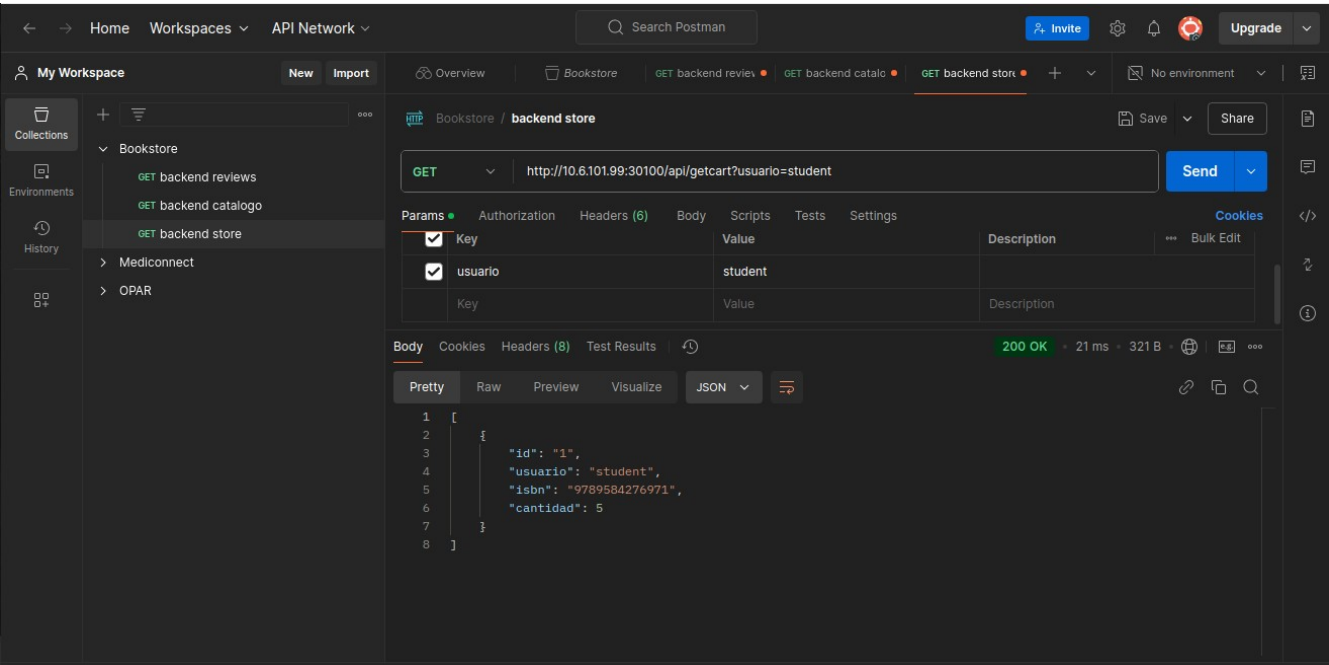
El de las reviews



El del catalogo



El de store



Repositorio de la solución: <https://github.com/JulianCastillo14/kubernetes-bookstore>

También se encuentra en la máquina virtual asignada en clase en la ruta:

/home/student/kubernetes-bookstore accediendo con la dirección ip 10.6.101.99

Video: <https://youtu.be/5xfshLTXWkE>