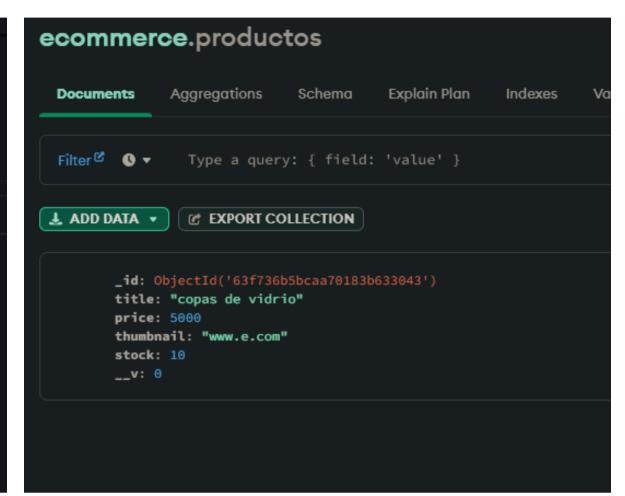
SERVIDOR

En una consola levante el servidor con npm run dev, y en la otra ejecuté npm run dev:axios para el test y no tiró ningun error

```
SALIDA CONSOLA DE DEPURACIÓN TERMINAL
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `babel-node ./src/index.js
Servidor express escuchando en el puerto 8080
Conectado a MongoDB!
 title: 'copas de vidrio',
 price: 5000,
 thumbnail: 'www.e.com',
 stock: 10,
 _id: new ObjectId("63f737e7405f02520e6faf03")
 title: 'copas de vidrio',
 price: 5000,
 thumbnail: 'www.e.com',
 stock: 10,
 _id: new ObjectId("63f737e7405f02520e6faf03"),
                                                                    MongoDB Compass - ecommerce/
```

PRUEBA AXIOS POST

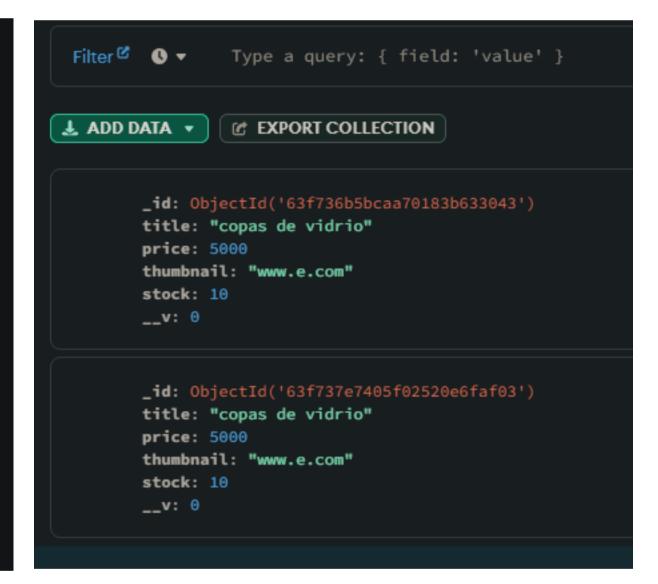
```
CLASE42
                                         src > test > JS axios.js > ...
   routes
                                                    try {
      JS carrito.js
                                                         const resp = await axios.delete(urlDelete);
      JS formulario.js
                                                         console.log(resp.data);
                                                    } catch (err) {
                                                         console.log(err);
      JS productos.js
      JS user.is
   services
      JS auth.js
      JS carrito.services.js
                                          42
                                                axiosPost();
                                                //axiosDelete();
      JS productos.services.js
      JS sms.service.js
                                         PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL SQL CONSOLE GITLENS
  test
                                          [nodemon] watching extensions: js,mjs,json
                                          [nodemon] starting `node ./src/test/axios.js`
     JS productos.test.js
   > views
                                           title: 'copas de vidrio',
   JS index.js
                                           price: 5000.
                                           thumbnail: 'www.e.com',
B.babelrc
                                           stock: 10,
.env
                                           id: '63f736b5bcaa70183b633043',
  .gitignore
                                           __v: 0
                                         [nodemon] clean exit - waiting for changes before restart
{} normalizado.json
                                         ¿Desea terminar el trabajo por lotes (S/N)? s
{} package-lock.json
                                         PS C:\Users\Aramonda\Downloads\BackEnd\clase42> \[
{} package.json
```



PRUEBA GET AXIOS

Para la siguiente prueba, terminé la ejecución y la inicié de nuevo con la funcion get que esta comentada.

```
[nodemon] clean exit - waiting for changes before restart
¿Desea terminar el trabajo por lotes (S/N)? s
PS C:\Users\Aramonda\Downloads\BackEnd\clase42> npm run dev:axios
> tercerentrega@1.0.0 dev:axios
> nodemon ./src/test/axios
[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node ./src/test/axios.js`
    id: '63f736b5bcaa70183b633043',
   title: 'copas de vidrio',
   price: 5000
 },
   id: '63f737e7405f02520e6faf03',
   title: 'copas de vidrio',
   price: 5000
```



PRUEBA DELETE AXIOS

Para la siguiente prueba, terminé la ejecución y la inicié de nuevo con la funcion delete que esta comentada, con el id del producto en la función.

```
//axiosGet();
      //axiosPost();
42
      axiosDelete();
44
PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL SQL CONSOLE
                                                                   GITLENS
   title: 'copas de vidrio',
   price: 5000
   id: '63f737e7405f02520e6faf03',
   title: 'copas de vidrio',
   price: 5000
[nodemon] clean exit - waiting for changes before restart
¿Desea terminar el trabajo por lotes (S/N)? s
PS C:\Users\Aramonda\Downloads\BackEnd\clase42> npm run dev:axios
> tercerentrega@1.0.0 dev:axios
> nodemon ./src/test/axios
[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node ./src/test/axios.js`
{ data: { acknowledged: true, deletedCount: 1 } }
[nodemon] clean exit - waiting for changes before restart
```

```
__id: ObjectId('63f737e7405f02520e6faf03')
title: "copas de vidrio"
price: 5000
thumbnail: "www.e.com"
stock: 10
__v: 0
```

PRUEBA SUPERTEST Y JEST

En la consola ejecuté npm run test:win, post get y delete por separado y luego todo junto.

```
> set NODE OPTIONS=--experimental-vm-modules && npx jest
(node:10840) ExperimentalWarning: VM Modules is an experimental feature. This featur
(Use `node --trace-warnings ...` to show where the warning was created)
 console.log
   Servidor express escuchando en el puerto 8080
     at Server.log (src/index.js:15:10)
  console.log
   Conectado a MongoDB!
     at log (src/persistencia/daos/daoMongoDB/mongodb.productos.js:13:13)
  console.log
     title: 'detergente',
     price: 1600,
     thumbnail: 'www.google.com',
     stock: 20000,
     _id: new ObjectId("63f74160f4dbcbf726663ef7"),
     __v: 0
     at ProductosRepository.log [as createProduct] (src/persistencia/repository/pro
 PASS src/test/productos.test.js
  Tests productos
   √ guardar producto (1252 ms)
```

```
_id: ObjectId('63f74160f4dbcbf726663ef7')
title: "detergente"
price: 1600
thumbnail: "www.google.com"
stock: 200000
__v: 0
```

```
PS C:\Users\Aramonda\Downloads\BackEnd\clase42> npm run test:win
  tercerentrega@1.0.0 test:win
 > set NODE_OPTIONS=--experimental-vm-modules && npx jest
(node:4592) ExperimentalWarning: VM Modules is an experimental feature. This feature could change
(Use `node --trace-warnings ...` to show where the warning was created)
    Servidor express escuchando en el puerto 8080
      at Server.log (src/index.js:15:10)
  console.log
   Conectado a MongoDB!
      at log (src/persistencia/daos/daoMongoDB/mongodb.productos.js:13:13)
  ASS src/test/productos.test.js
  Tests productos

√ traer productos (1359 ms)

Test Suites: 1 passed, 1 total
         1 passed, 1 total
Tests:
Snapshots: 0 total
```

```
PS C:\Users\Aramonda\Downloads\BackEnd\clase42> npm run test:win

> tercerentrega@1.0.0 test:win
> set NODE_OPTIONS=--experimental-vm-modules && npx jest

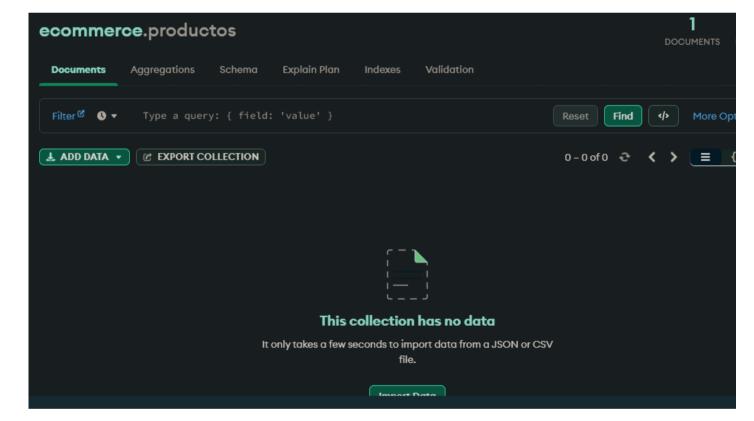
(node:5048) ExperimentalWarning: VM Modules is an experimental feature. This f
(Use `node --trace-warnings ...` to show where the warning was created)
    console.log
        Servidor express escuchando en el puerto 8080

        at Server.log (src/index.js:15:10)

console.log
        Conectado a MongoDB!

        at log (src/persistencia/daos/daoMongoDB/mongodb.productos.js:13:13)

PASS        src/test/productos.test.js
Tests productos
        v delete producto (1304 ms)
```



```
(node:212) Experimentalwarning: VM Modules is an experimental feature. This fo
(Use `node --trace-warnings ...` to show where the warning was created)
  console.log
    Servidor express escuchando en el puerto 8080
      at Server.log (src/index.js:15:10)
  console.log
    Conectado a MongoDB!
      at log (src/persistencia/daos/daoMongoDB/mongodb.productos.js:13:13)
 PASS src/test/productos.test.js
  Tests productos

√ guardar producto (1165 ms)

√ traer productos (172 ms)

√ delete producto (165 ms)

Test Suites: 1 passed, 1 total
             3 passed, 3 total
Tests:
Snapshots: 0 total
Time:
             4.971 s, estimated 5 s
Ran all test suites.
Jest did not exit one second after the test run has completed.
```