

Los resultados que se pueden apreciar en el test son apropiados.

Referentes al test con Axios, para verificar la funcionalidad hacia la API Rest de productos, verificando la correcta lectura de productos disponibles, incorporación de nuevos productos, modificación y borrado. Los resultados son los siguientes:

```
[
  {
    timestamp: 1682090593159,
    name: 'testName',
    description: 'testDescription',
    code: 'testCode',
    photo: 'testPhoto',
    price: 100,
    stock: 100,
    id: '0'
  },
  [
    {
      timestamp: 1682090593162,
      name: 'updateName',
      description: 'updateDescription',
      code: 'updateCode',
      photo: 'updatePhoto',
      price: 101,
      stock: 101,
      id: '0'
    }
  ]
]
```

En cuanto a las mismas pruebas, a través de un código de test, que utilice mocha, chai y Supertest, para probar cada uno de los métodos HTTP de la API Rest de productos. Podemos apreciar los siguientes resultados:

```
Products controller
  getProducts
    ✓ Debería retorna un array de productos luego de agregar productos
    ✓ Debería retorna una array vacío si no tiene productos
  getProductById
    ✓ Debería retorna null sino existe ningún producto
    ✓ Debería retorna el producto si ya existe
  updateProduct
    ✓ Debería actualizar el producto en la base de datos
  deleteProduct
    ✓ Debería eliminar el producto de la base de datos

6 passing (9ms)
```

#### Products api test

GET api/productos

✓ Debería retornar status 200 y un array vacío si no tiene productos (66ms)

POST /api/productos

✓ Debería retornar status 200 y agregar un producto valido (53ms)

PUT /api/productos

✓ Debería retornar status 200 si se actualiza correctamente el producto (106ms)

✓ Debería retornar status 404 si no existe el producto a actualizar (54ms)

DELETE /api/productos

✓ Debería retornar status 404 sino existe el producto a eliminar (52ms)

✓ Debería retornar status 200 si se elimino correctamente el producto (103ms)

6 passing (445ms)