Hola Lucho!  
  
Te comparto el link para el último desafío.  
  
link repo de GtHub:  
<https://github.com/Federico-Gomez/Proyecto-Backend>  
  
Comentarios:  
1. Tengo todo el proyecto con la sintaxis require() de Common JS. Chai me da error y me pide que migre todo para usar dynamic imports. Intenté usar los imports solo en el testing a ver qué pasaba, pero me pedía que agregue type: ‘module’ al package.json y esto hace que no funcione todo lo que usa la sintaxis require(). Traté de encontrar si había forma de usar chai sin migrar todo el proyecto a la sintaxis de imports pero no pude. Si hay alguna forma, te agradecería el dato! En fin, debido a esto, no me quedó opción que usar Mocha y Supertest para esta entrega porque si intentaba cambiar todo a sintaxis de imports no iba a llegar a tiempo.

2. Hay tests para Carts, Products y Sessions con Mocha y Supertest. Hay tests para Users con Mocha.

3. Tengo una función main() en app.js que inicializa Mongo en la base de datos con dbName: ‘ecommerce’. Esto genera conflictos con el dbName: ‘testing’ que tengo en los archivos .test.js. Los test funcionan (salvo los de Sessions, ver más abajo) pero los productos, carts y users van a la base de datos ‘ecommerce’.

4. Los tests de Sessions funcionan solamente si se usa la base de datos indicada en main(), por eso dice dbName: ‘ecommerce’ en Sessions.test.js

5. Para evitar borrar la base de datos del proyecto sin querer, los hooks after y beforeEach están comentados. Hay que tener cuidado con con los dbName que se usan en main() y en los tests para no borrar datos sin querer (me pasó jaja).

6. Tengo varios de los enpoints que se testean protegidos por middlewares de autenticación, para que funcionen los tests vas a tener que reemplazar los campos de email y password que están en el test de Products y Carts por los de un usuario de la DB. Vas a necesitar un user con role: user y uno con role: premium (agarrá el \_id de un role: user y con el endpoint GET /api/users/premium/:uid hay una vista que te permite cambiarlo a premium). Hay comments en el código.

7. GET de Carts: para que funcione, el test requiere que vayas al carts.router.js y cambies la response del res.render en el que está ahora al res.json() de abajo. Hay comments en el código.

Archivo .env.dev:

MONGO\_URI=mongodb+srv://fedehgz:Amard0nandShem1n@ecommerce.nojdknt.mongodb.net/?retryWrites=true&w=majority

ENV=dev

SECRET=MySecretKey

PORT=8080

PRIVATE\_KEY=claveprivadajwtasasecret

GH\_CLIENT\_ID='Iv1.bb7b667e5b6b81ad'

GH\_CLIENT\_SECRET='8a03ab4aa7f946ada7ed743f6d738e8f5f90690c'

GH\_CALLBACK\_URL='http://localhost:8080/api/sessions/githubcallback'

GMAIL\_ACCOUNT=fedehgz@gmail.com

GMAIL\_PASSWORD=gvmk oapm ztte rgnm

TWILIO\_ACCOUNT\_SID=ACa35ec9e6ce182847fb99c2cdf35dcf26

TWILIO\_AUTH\_TOKEN=a84d4a8ef6b4f6bb0bd4f2378c8fa793

TWILIO\_PHONE\_NUMBER=+12072093266

ADMIN\_EMAIL=adminCoder@coder.com

ADMIN\_PASSWORD=adminCod3r123

Archivo .env.prod:

MONGO\_URI=mongodb+srv://fedehgz:Amard0nandShem1n@ecommerce.nojdknt.mongodb.net/?retryWrites=true&w=majority

ENV=prod

SECRET=MySecretKey

PORT=3000

PRIVATE\_KEY=claveprivadajwtasasecret

GH\_CLIENT\_ID='Iv1.bb7b667e5b6b81ad'

GH\_CLIENT\_SECRET='8a03ab4aa7f946ada7ed743f6d738e8f5f90690c'

GH\_CALLBACK\_URL='http://localhost:8080/api/sessions/githubcallback'

GMAIL\_ACCOUNT=fedehgz@gmail.com

GMAIL\_PASSWORD=gvmk oapm ztte rgnm

TWILIO\_ACCOUNT\_SID=ACa35ec9e6ce182847fb99c2cdf35dcf26

TWILIO\_AUTH\_TOKEN=a84d4a8ef6b4f6bb0bd4f2378c8fa793

TWILIO\_PHONE\_NUMBER=+12072093266

ADMIN\_EMAIL=adminCoder@coder.com

ADMIN\_PASSWORD=adminCod3r123