**Proyecto Final**

**Desarrollo Backend NodeJS**

**ECOMMERCE**

**Nancy Barea**

**09/08/2022**

**Contenido**

[Notas antes de empezar 3](#_Toc110980387)

[Github 3](#_Toc110980388)

[Heroku 3](#_Toc110980389)

[**Menú** 4](#_Toc110980390)

[**Chat** 5](#_Toc110980391)

[**Subir Archivo (registro de usuario)** 5](#_Toc110980392)

[**Información Servidor** 8](#_Toc110980393)

[Postman – carga de archivo 8](#_Toc110980394)

[**Usuario – registro nuevo usuario** 8](#_Toc110980395)

[**Producto – alta de nuevo producto** 10](#_Toc110980396)

[Swagger 10](#_Toc110980397)

[Graphl 11](#_Toc110980398)

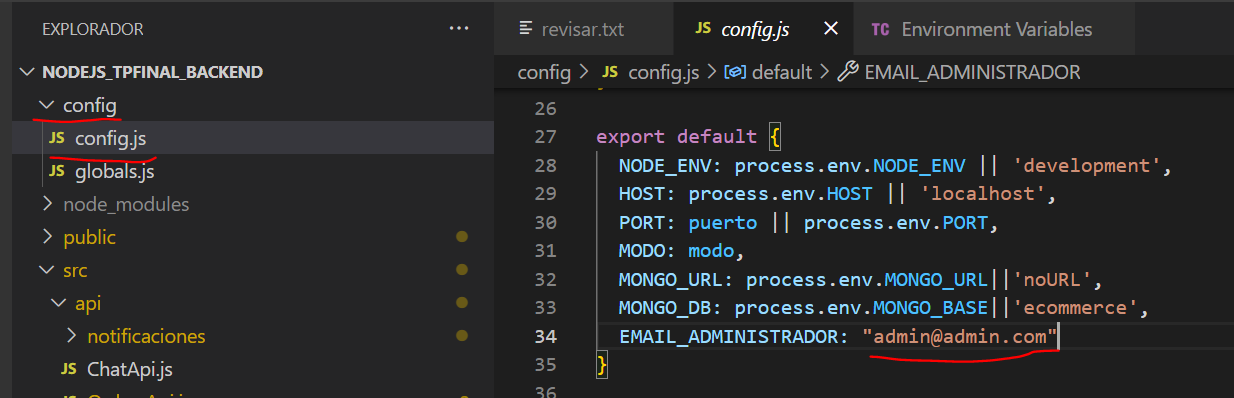
[Testing 11](#_Toc110980399)

# Notas antes de empezar

**Ejecutar el programa**

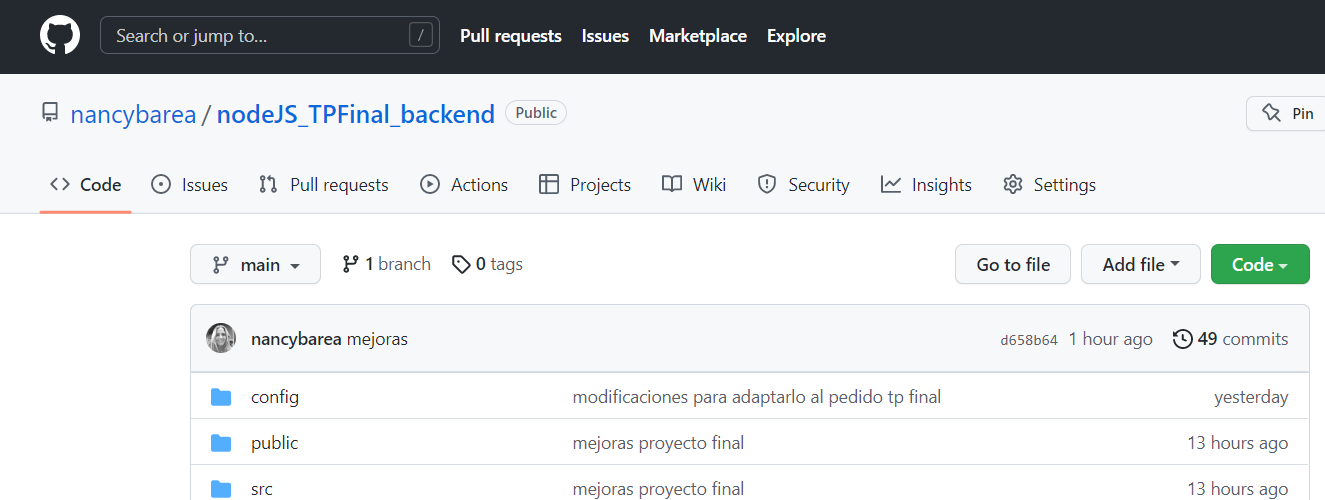


**Configuración usuario administrador**



# Github

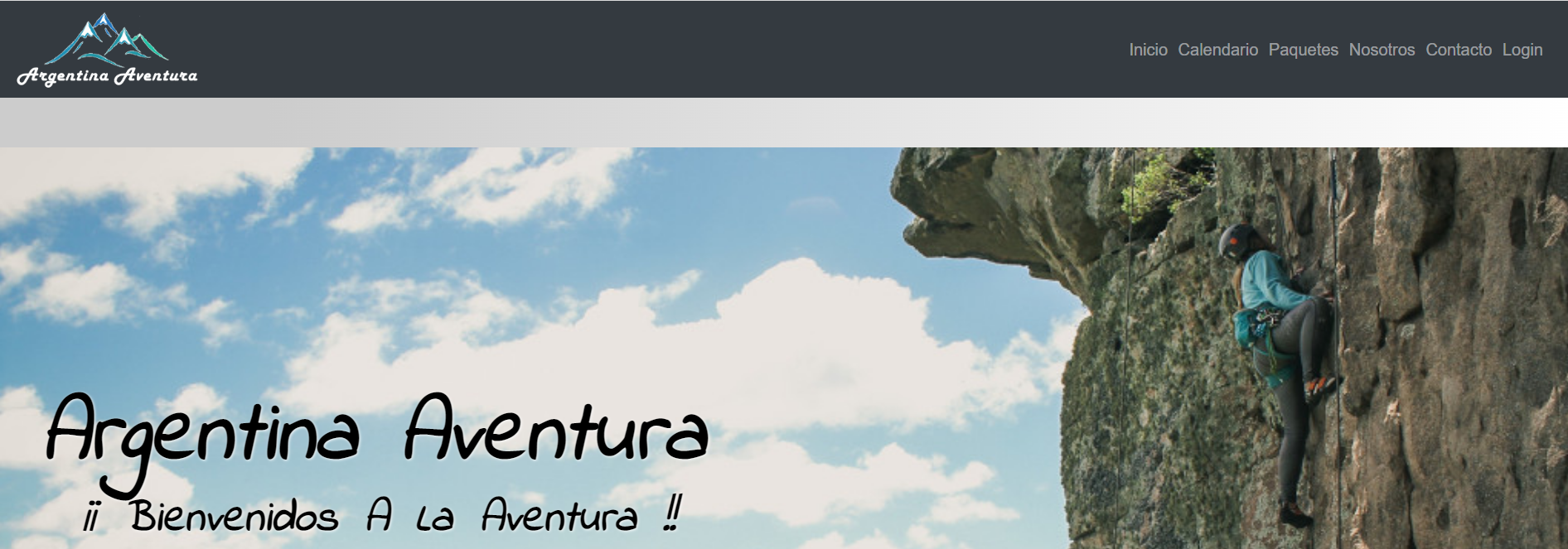
<https://github.com/nancybarea/nodeJS_TPFinal_backend>



# Heroku

<https://ecommerce-bac.herokuapp.com/web/>

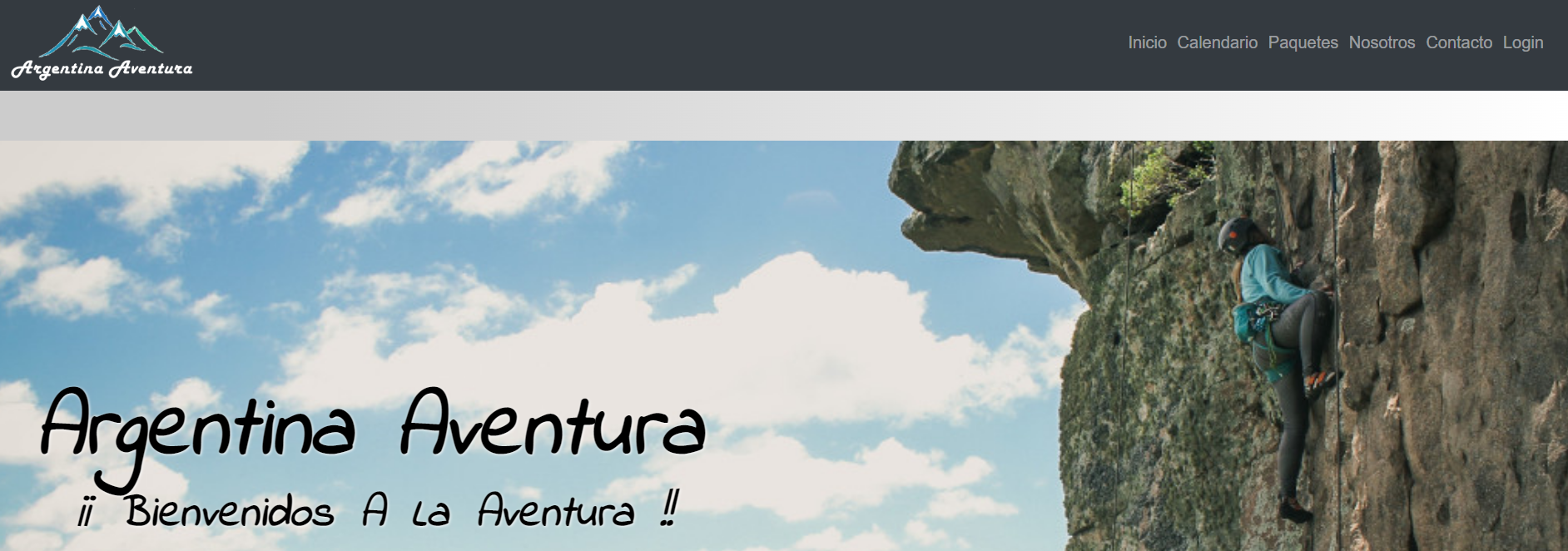
(está conectado con el repositorio de github)



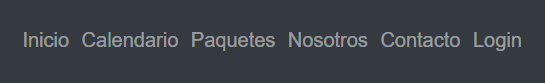
A continuación se detalla solo las funcionalidades más relevantes para este proyecto

* Menú: como se modifica el menú según el rol del usuario logueado.
* Chat: se solicito realizar un chat con socket donde pueden hablar todos los usuarios.
* Subir Archivo : subir imagen con multer y lo guarde en carpeta public
* Información servidor: solicitada en el proyecto con el uso de plantillas.

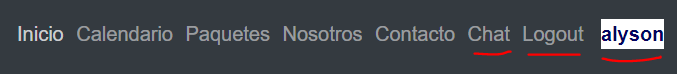
## **Menú**



* **Menu Sin usuario logueado**

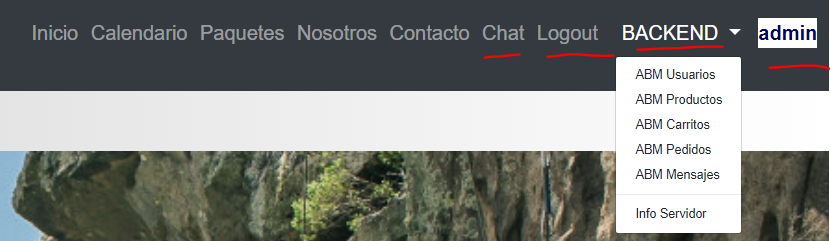


* **Menu con usuario logueado y rol “usuario”**



Siendo “alyson” el nombre del usuario logueado

* **Menu con usuario logueado y rol “admin”**

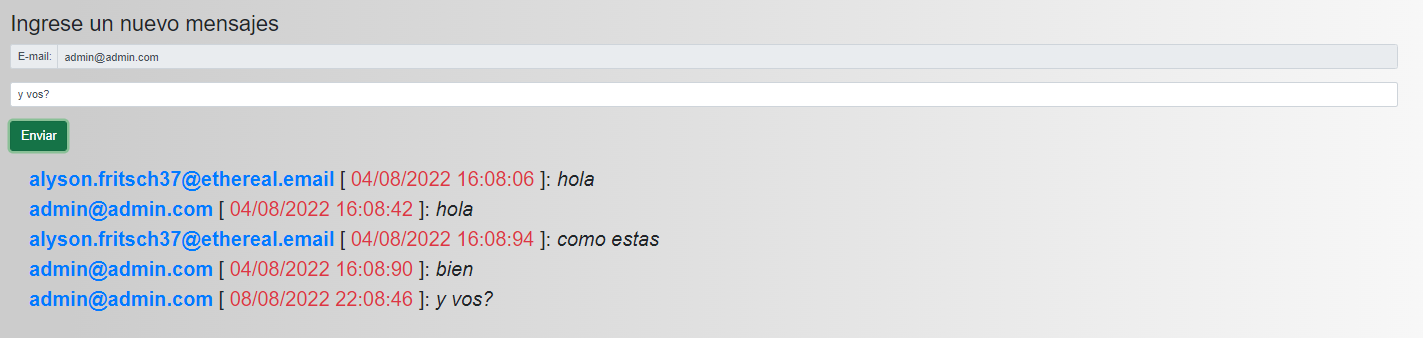


Siendo “admin” el nombre del usuario logueado

## **Chat**

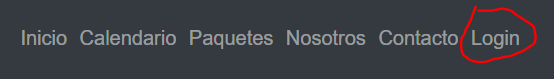
En el cuadro de email aparecerá sin poder editar, el email del usuario logueado.

El cuadro de abajo es para escribir los mensajes



## **Subir Archivo (registro de usuario)**

El ejemplo para subir archivo se va a encontrar en el registro del nuevo usuario





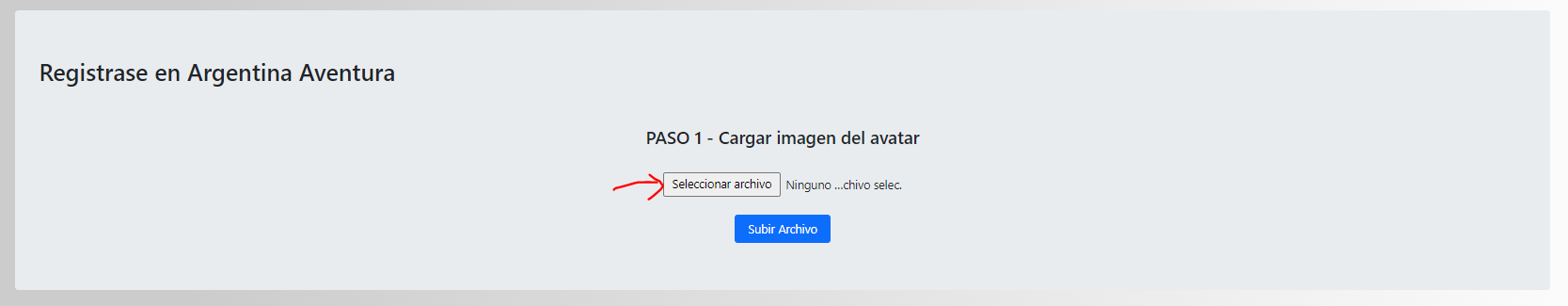
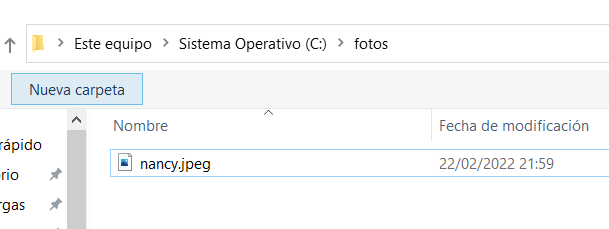
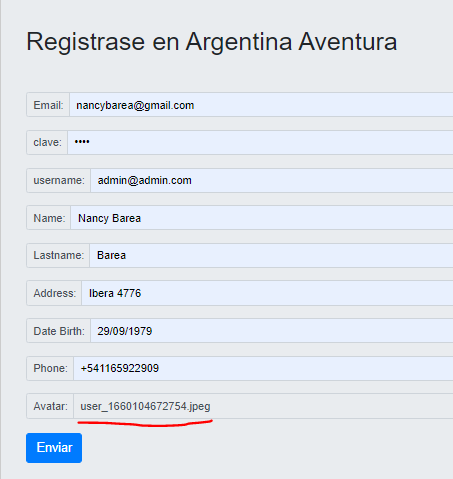
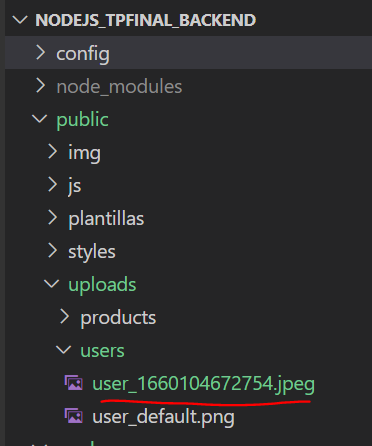


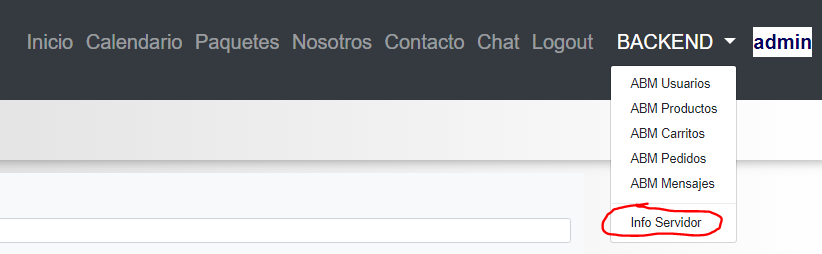
Imagen por defecto 🡪 public/uploads/users/UserDefault.png





## **Información Servidor**





# Postman – carga de archivo

## **Usuario – registro nuevo usuario**

Imagen por defecto (cuando el usuario no lo sube)

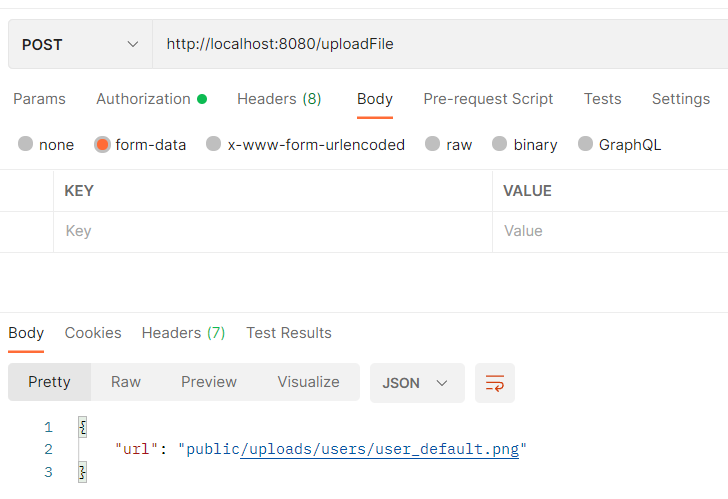
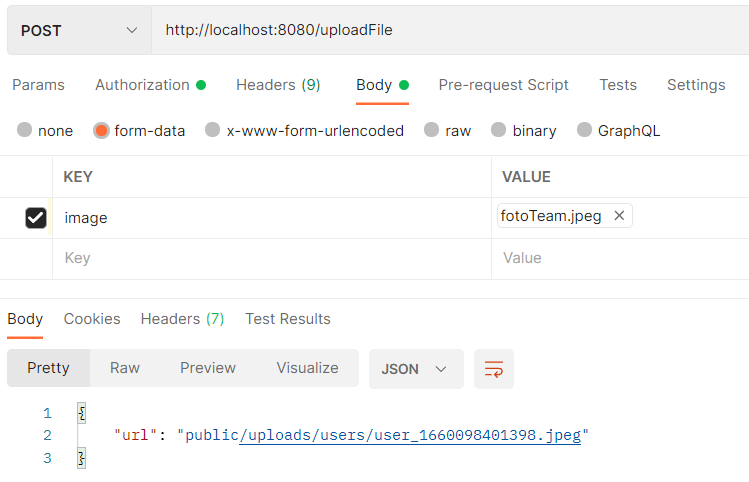
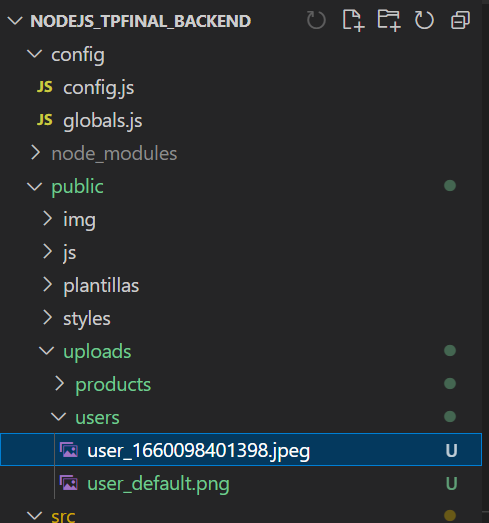


Imagen subida por el usuario





## **Producto – alta de nuevo producto**

Imagen por defecto (cuando el usuario no lo sube)

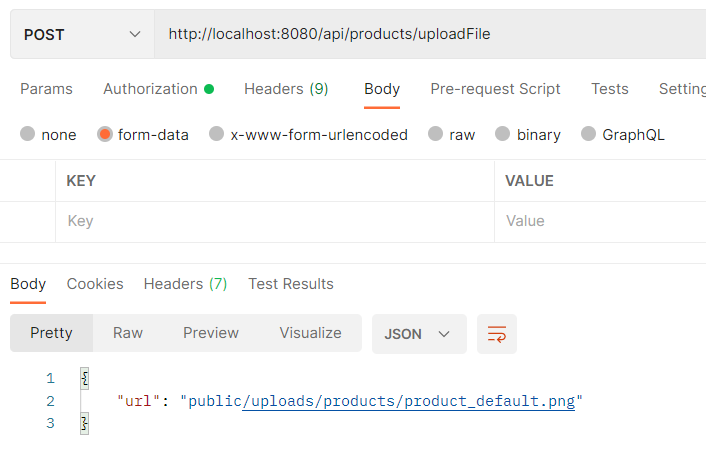
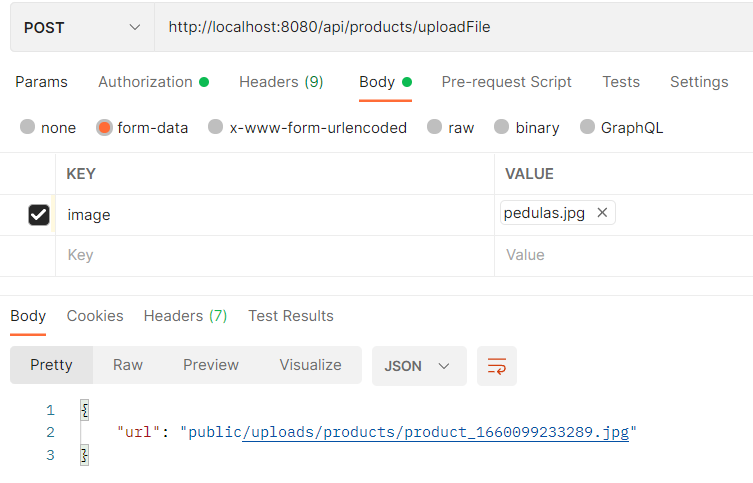
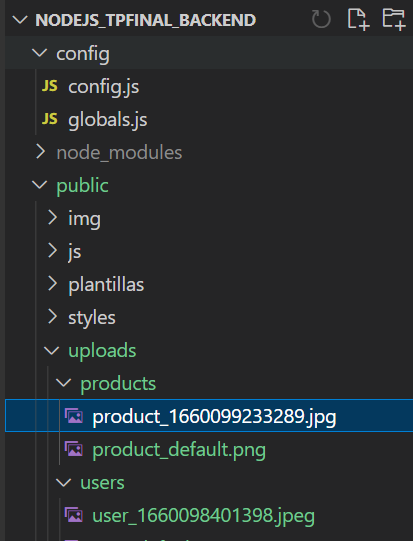


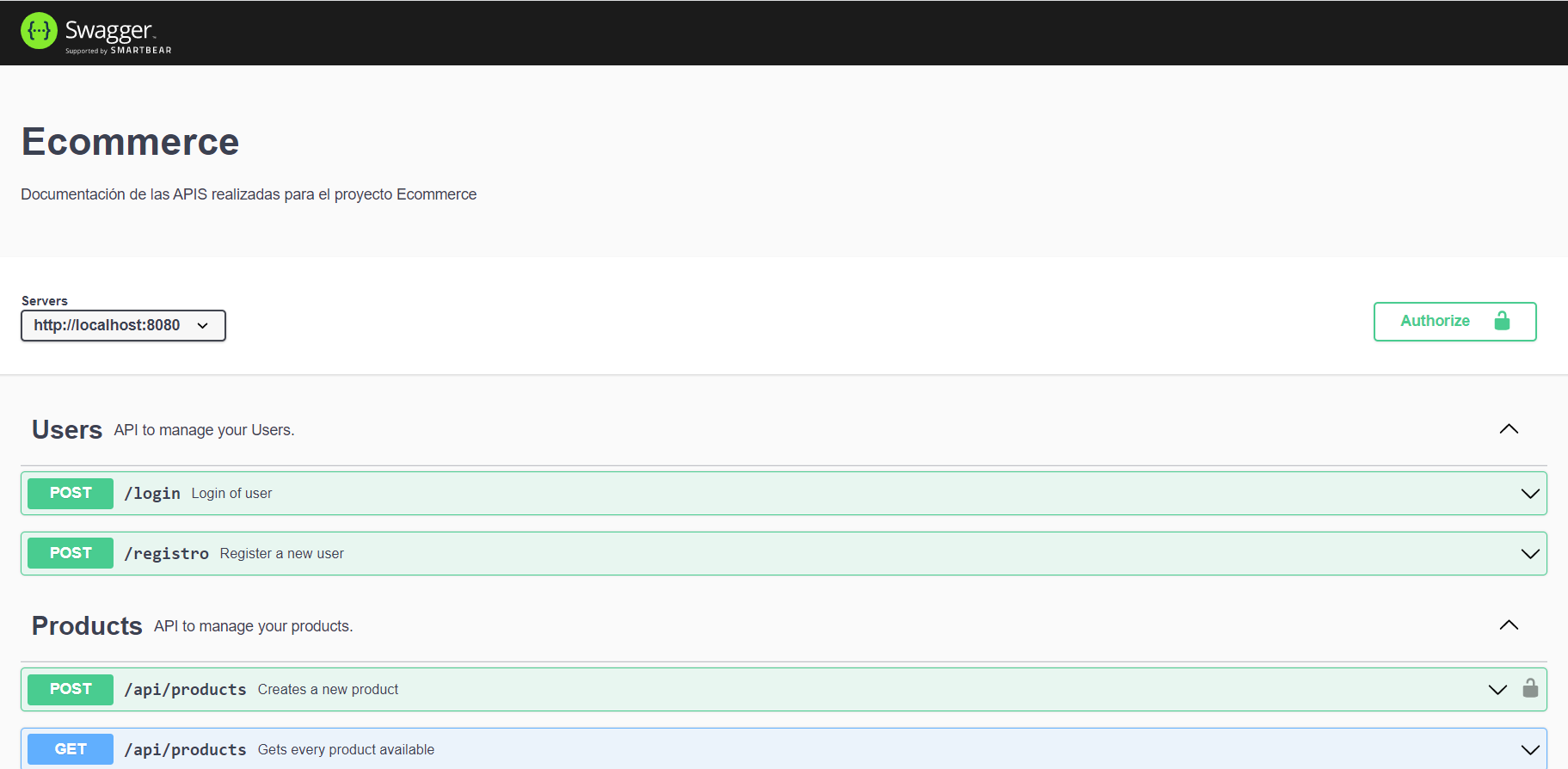
Imagen subida por el usuario

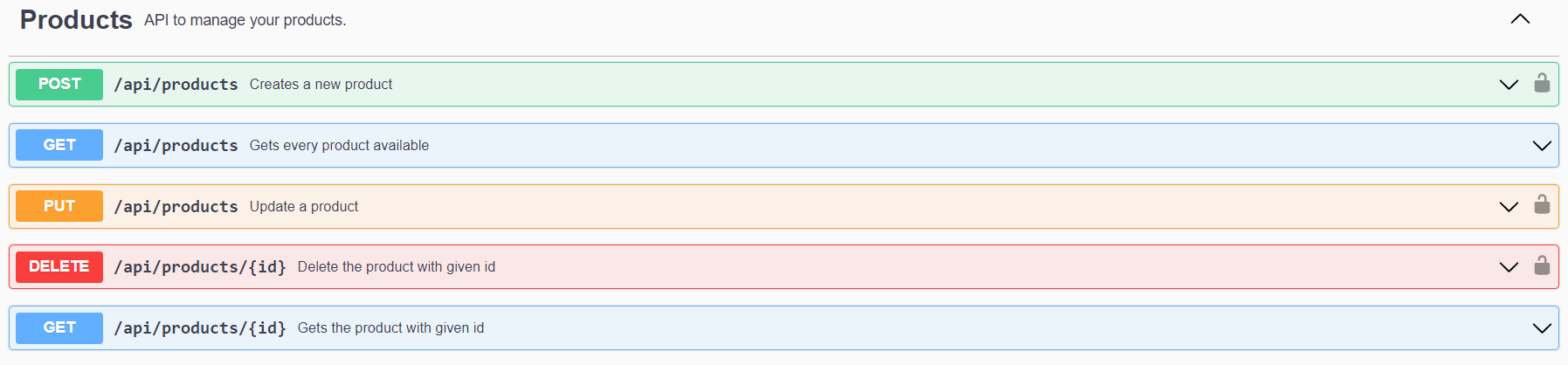


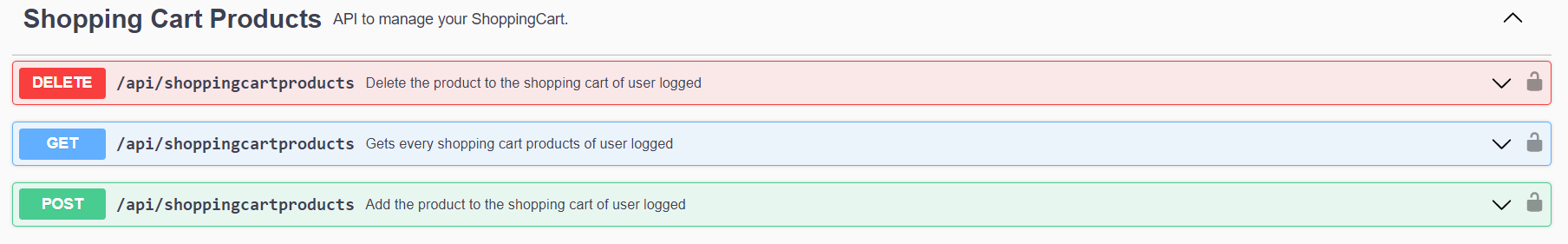


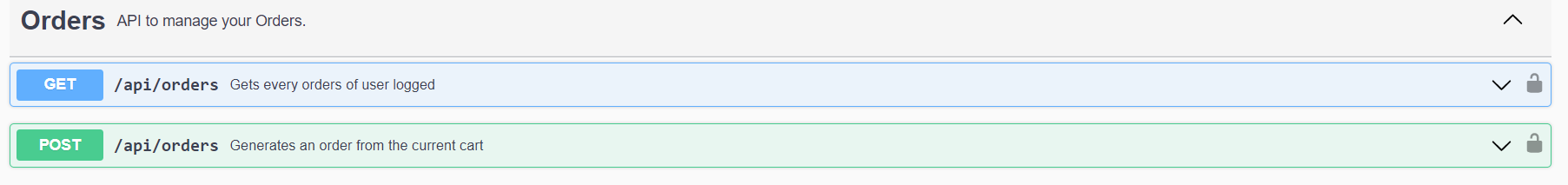
# Swagger

<http://localhost:8080/api/docs/>







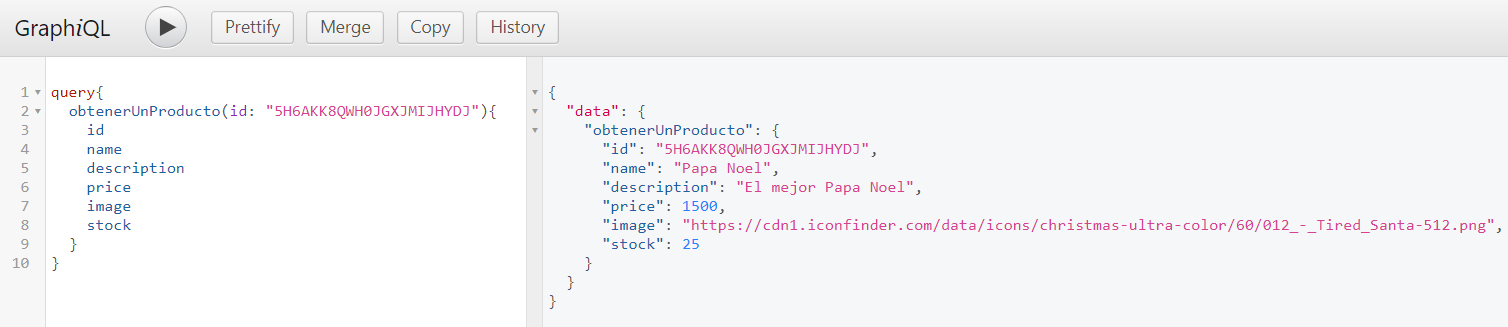




# Graphl

<http://localhost:8080/graphql>

Ejemplo:



# Testing



