

Juan Esteban Santander Acero

TDS VI

Pruebas de fuego DevLocker V1

1) Registrar user A y user B

Inicializamos proyecto con npm run dev

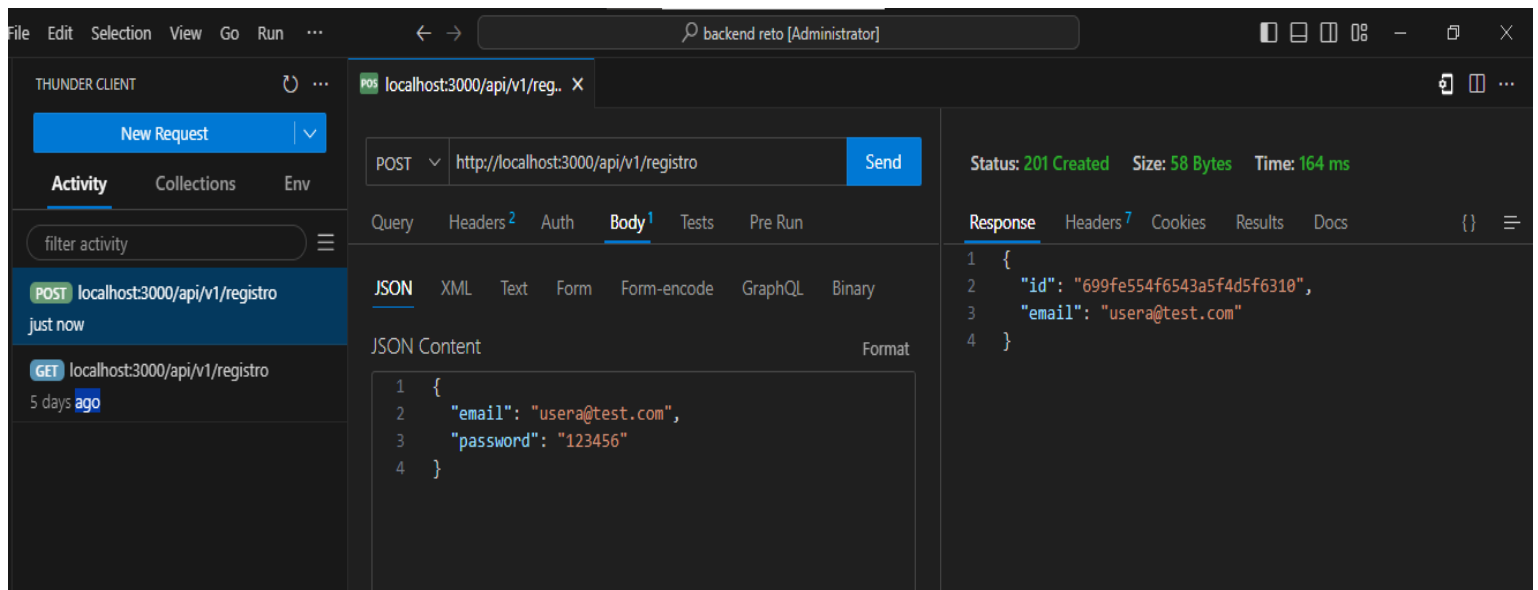
```
PS D:\DOCUMENTOS USUARIO\Documents\backend reto> npm run dev

> devlocker-v1@1.0.0 dev
> nodemon server.js

[nodemon] 3.1.14
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node server.js`
Server running on port 3000
```

Abrimos Thunder Client y creamos el primer usuario (User A) para obtener credenciales válidas y luego iniciar sesión.

En el body ponemos el user y la contraseña del usuario en formato JSON



Viewing Collection: users

New DocumentNew Index

Simple

Advanced

Key

Value

String

Find

Delete all 1 documents retrieved

_id	email	password	createdAt	updatedAt	__v
<div><div></div><div></div><div>699fe554f6543a5f4d5f6310</div></div>	usera@test.com	\$2a\$10\$8ePVHI0TZbKT7EoE0rDoJ.MSIq2rR759yPB7D7Hean...	Thu Feb 26 2026 06:16:52 GMT+0000 (Coordinated Universal Time)	Thu Feb 26 2026 06:16:52 GMT+0000 (Coordinated Universal Time)	0

Obtuvimos como resultado un status 201 por lo tanto se recibió id y email del usuario A y quedó creado.

Mismo proceso con user B

THUNDER CLIENT

New Request

ActivityCollectionsEnv

filter activity

POSTlocalhost:3000/api/v1/registrojust now

GETlocalhost:3000/api/v1/registro5 days ago

POST

http://localhost:3000/api/v1/registro

Send

QueryHeadersAuthBodyTestsPre Run

JSONXMLTextFormForm-encodeGraphQLBinary

JSON ContentFormat

```
1 {
2   "email": "userb@test.com",
3   "password": "12345678"
4 }
```

Status: 201 CreatedSize: 58 BytesTime: 93 ms

ResponseHeaders7CookiesResultsDocs

```
1 {
2   "id": "699fe785f6543a5f4d5f6313",
3   "email": "userb@test.com"
4 }
```

Viewing Collection: users

New DocumentNew Index

Simple

Advanced

Key

Value

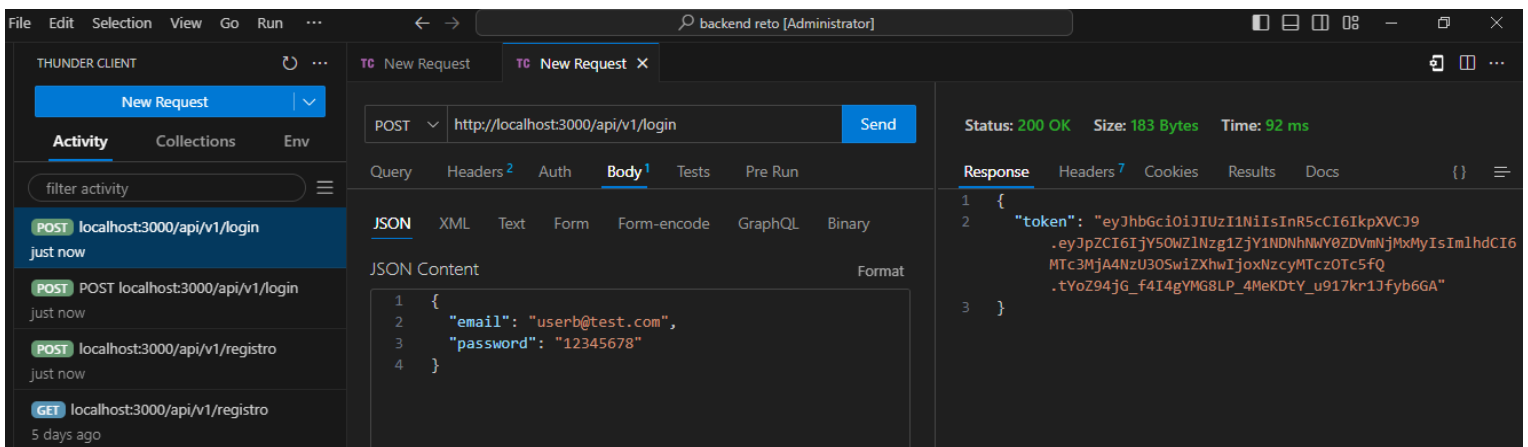
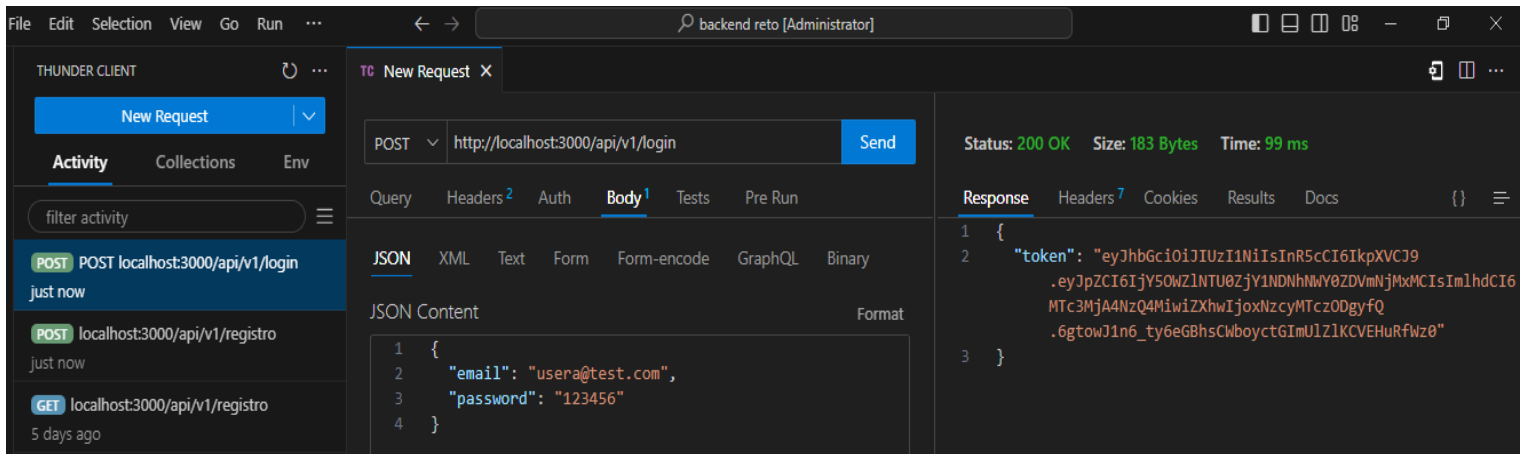
String

Find

Delete all 2 documents retrieved

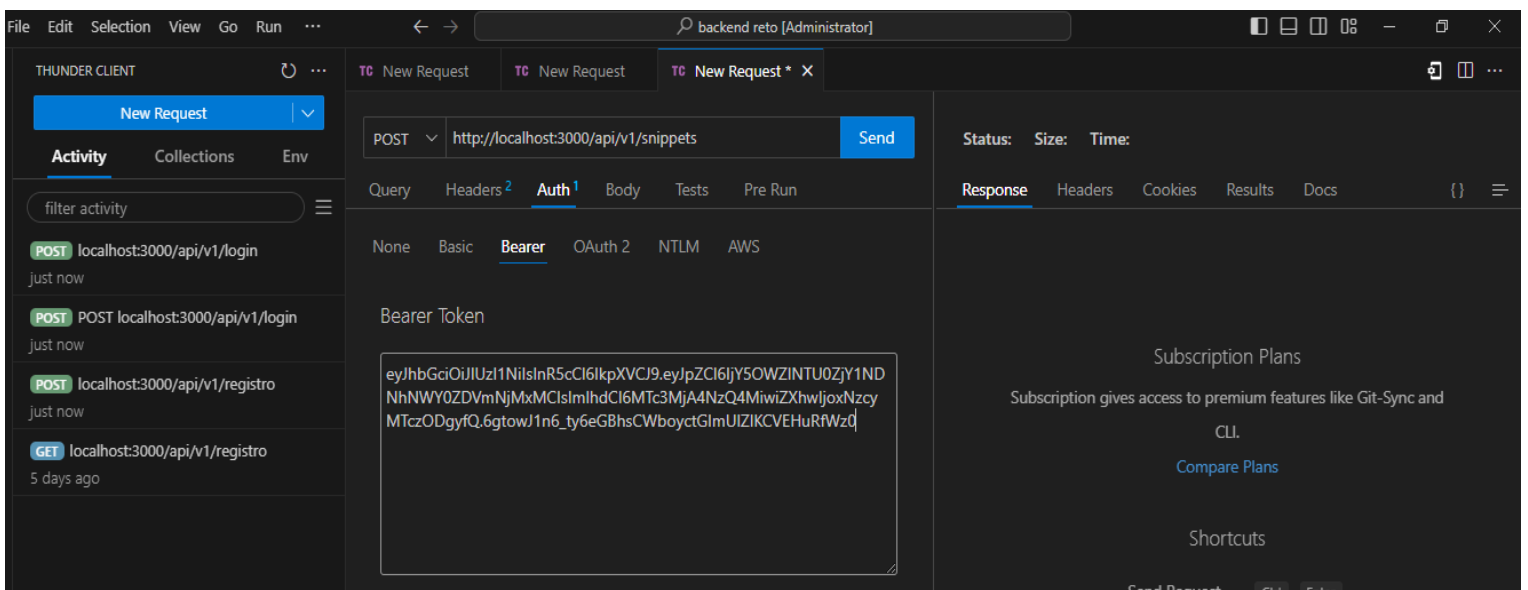
_id	email	password	createdAt	updatedAt	__v
<div><div></div><div></div><div>699fe554f6543a5f4d5f6310</div></div>	usera@test.com	\$2a\$10\$8ePVHI0TZbKT7EoE0rDoJ.MSIq2rR759yPB7D7Hean...	Thu Feb 26 2026 06:16:52 GMT+0000 (Coordinated Universal Time)	Thu Feb 26 2026 06:16:52 GMT+0000 (Coordinated Universal Time)	0
<div><div></div><div></div><div>699fe785f6543a5f4d5f6313</div></div>	userb@test.com	\$2a\$10\$CNJmgw0OsgBqUr9VW.WYnOQk0Ggu/2pwgdegVtiKtF...	Thu Feb 26 2026 06:26:13 GMT+0000 (Coordinated Universal Time)	Thu Feb 26 2026 06:26:13 GMT+0000 (Coordinated Universal Time)	0

2) Login con los users para obtener token jwt para que podamos acceder a los endpoint protegidos:

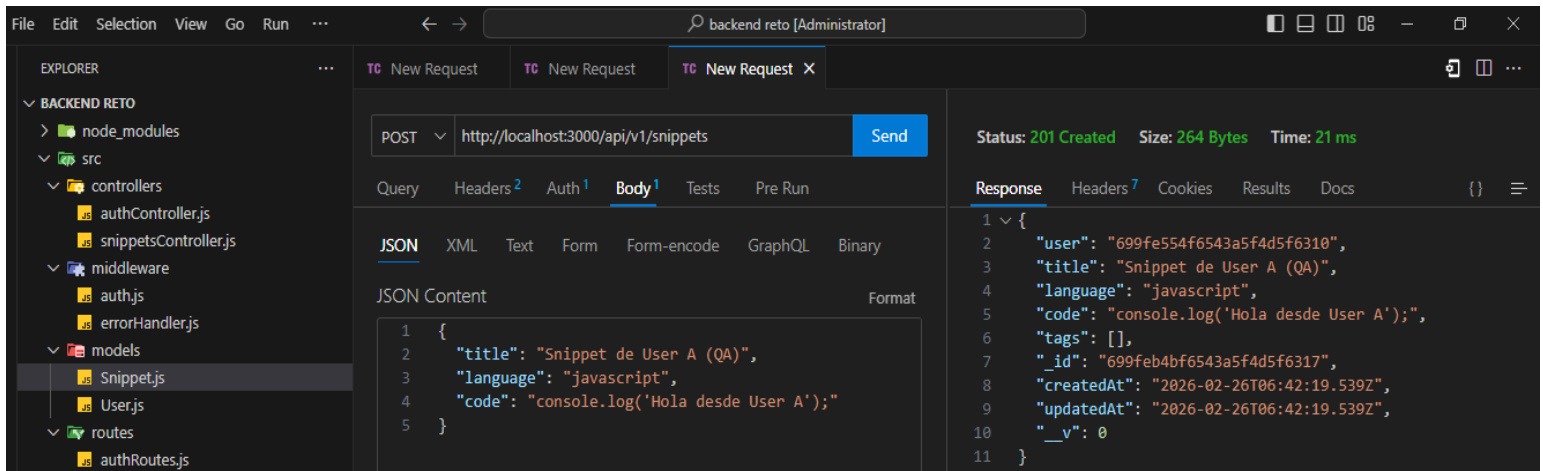


la autenticación fue exitosa y obtuvimos los tokens de user A y B

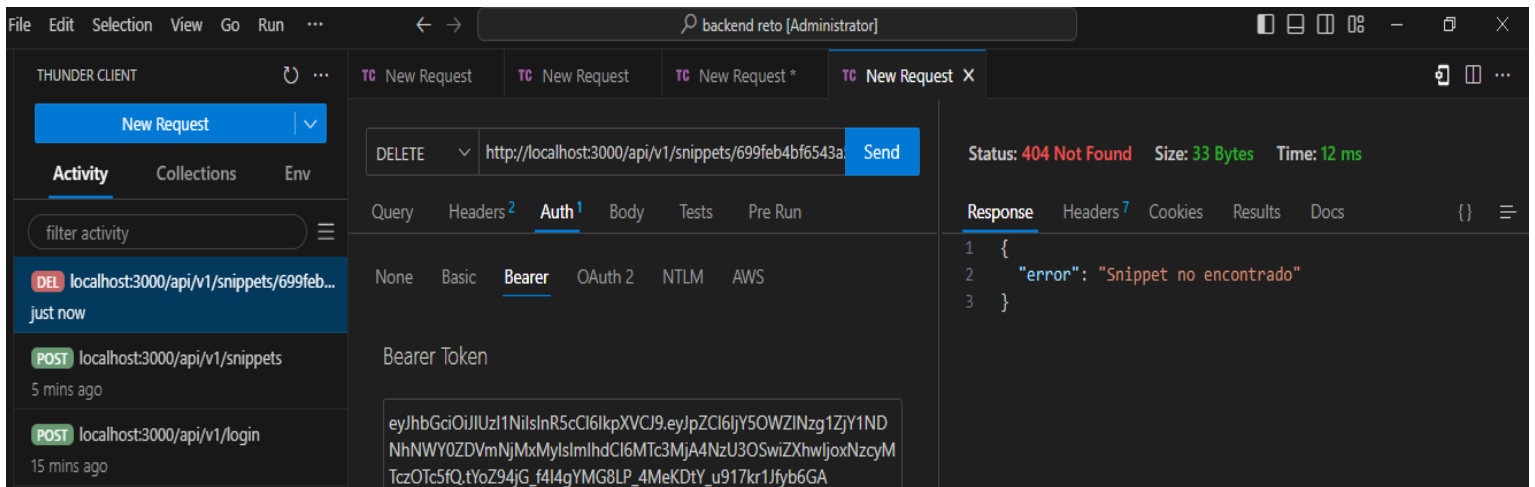
3) Creación de Snippet con user A



Con el método POST en Auth->Bearer ponemos el token del user A y luego en Body procedemos a poner el snippet



4) Se intenta Borrar el snippet creado por el usuario A por el usuario B usando el token del usuario B:



Obtenemos como resultado snippet no encontrado, eso significa que funciona correctamente y si hay privacidad, el snippet sigue estando.



Mongo Express Database: devlocker Collection: snippets

New Document New Index

Simple Advanced

Key Value String Find

Delete all 1 documents retrieved

_id	user	title	language	code	tags	createdAt	updatedAt	_v
  699feb4bf6543a5f4d5f6317	699fe554f6543a5f4d5f6310	Snippet de User A (QA)	javascript	console.log("Hola desde User A");		Thu Feb 26 2026 06:42:19 GMT+0000 (Coordinated Universal Time)	Thu Feb 26 2026 06:42:19 GMT+0000 (Coordinated Universal Time)	0

Si usamos el token A, que si esta autorizado:

File Edit Selection View Go Run ... backend reto [Administrator]

THUNDER CLIENT New Request

Activity Collections Env

filter activity

DEL localhost:3000/api/v1/snippets/699feb... just now

POST localhost:3000/api/v1/snippets 5 mins ago

POST localhost:3000/api/v1/login 15 mins ago

DELETE http://localhost:3000/api/v1/snippets/699feb4bf6543a5f4d5f6317 Send

Query Headers 2 Auth 1 Body Tests Pre Run

None Basic Bearer OAuth 2 NTLM AWS

Bearer Token

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjY5OWZINTU0ZjY1NDNhNWY0ZDVmNjMxMCI6ImlhdCI6MTc3MjA4NzQ0MiwiZXhwIjoxNzcyMTczODgyfQ.6gtowJ1n6_ty6eGBhsCWboyctGlmUIZIKCvEHuRfWz0
```

Status: 200 OK Size: 31 Bytes Time: 10 ms

Response Headers 7 Cookies Results Docs

```
1 {
2   "mensaje": "Snippet eliminado"
3 }
```

Mongo Express Database: devlocker Collection: snippets

New Document New Index

Simple Advanced

Key Value String Find

No documents found.

Se elimina correctamente