

Generación de token:

The screenshot shows a Postman interface with a POST request to `localhost:3000/signin`. The request body is raw JSON:

```
1 {
2   "email": "jerry@example.com",
3   "password": "password"
4 }
```

The response is a 200 OK status with a token in the body:

```
1 {
2   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
3   eyJpZCI6MSwiZW1haWmiOiJqZXJyeUBleGFTcGx1LmNvbSIiMlhdCI6MTc2MjAxOTU0NiwiZXhwIjoxNzYyMDIzMTQ2fQ.
4   lkadsMQnRt88tElCajDxzbu7XrnXYuPPRvOmwfFH1oo"
```

Capturas de pantalla de endpoints:

Sin token:

The screenshot shows a GET request to `localhost:3000/users` without a token. The request body is raw JSON:

```
1 {
2   "email": "jerry@example.com",
3   "password": "password"
4 }
```

The response is a 401 Unauthorized status with a message in the body:

```
1 {
2   "message": "Unauthorized"
3 }
```

Con token:

The screenshot shows the Postman interface with a successful API call. The URL is `localhost:3000/signin`. The method is `GET` and the endpoint is `localhost:3000/users`. The response status is `200 OK` with a response time of `634 ms` and a size of `7.45 KB`. The response body is a JSON array containing two user objects:

```
1 [  
2 {  
3   "id": 17,  
4   "name": "antonio",  
5   "email": "antonio@gmail.com",  
6   "password": "$2b$10$obF9fhon7VI8JrnSpC8Md0c2ye0mkEUrNfXlwQgZ.pBZGIKUBQ6ly"  
7 },  
8 {  
9   "id": 18,  
10  "name": "Gerardo Cornejo",  
11  "email": "00043524@uca.edu.sv",  
12  "password": "$2b$10$obF9fhon7VI8JrnSpC8Md0c2ye0mkEUrNfXlwQgZ.pBZGIKUBQ6ly"  
]
```

The screenshot shows the Postman interface with a successful API call. The URL is `localhost:3000/signin`. The method is `GET` and the endpoint is `localhost:3000/users/120`. The response status is `200 OK` with a response time of `83 ms` and a size of `394 B`. The response body is a JSON object representing a single user:

```
1 [  
2 {  
3   "id": 120,  
4   "name": "Diego",  
5   "email": "dars@gmail.com",  
6   "password": "$2b$10$0PwTNyg1hWYJLseBh4U06uW3Bauk70IE0EndD6Cip5KbdTwGxxS.W"  
7 }  
8 ]
```

POST localhost:3000/signin

localhost:3000/signin

POST localhost:3000/users

Params Authorization Headers (9) Body Scripts Tests Settings Cookies Beautify

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "name": "Diego Antonio",
3   "email": "diego123@gmail.com",
4   "password": "contra4321"
5 }
```

Body Cookies Headers (8) Test Results

HTML Preview Visualize

201 Created 1.10 s 288 B

1 User added with ID: 121

This screenshot shows a Postman request for a POST operation at `localhost:3000/signin`. The body contains a JSON object with fields `name`, `email`, and `password`. The response status is `201 Created` with a response time of 1.10 seconds and a response size of 288 bytes. The response body indicates that a user was added with ID `121`.

GET localhost:3000/signin

localhost:3000/signin

GET localhost:3000/users

Params Authorization Headers (9) Body Scripts Tests Settings Cookies Beautify

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "name": "Diego Antonio",
3   "email": "diego123@gmail.com",
4   "password": "contra4321"
5 }
```

Body Cookies Headers (8) Test Results

{ } JSON Preview Visualize

200 OK 180 ms 7.59 KB

```
369   "id": 120,
370   "name": "Diego",
371   "email": "dars@gmail.com",
372   "password": "$2b$10$QPtNTyglhWYJLseBh4U06uW3Bauk70IE0EndD6Cip5KbdTwGxxS.W"
373 },
374 {
375   "id": 121,
376   "name": "Diego Antonio",
377   "email": "diego123@gmail.com",
378   "password": "$2b$10$G1.qtNt0U6xAoxMsPv8N6.c1aSm1Nfp3V1Dy8M.fwGh53iWkqPCW2"
379 }
380 ]
```

Runner Start Proxy Cookies Vault Trash

This screenshot shows a Postman request for a GET operation at `localhost:3000/signin`. The body contains a JSON object with fields `name`, `email`, and `password`. The response status is `200 OK` with a response time of 180 ms and a response size of 7.59 KB. The response body is a JSON array containing two user objects, each with `id`, `name`, `email`, and `password` fields.

PUT localhost:3000/signin

localhost:3000/signin

Send

PUT localhost:3000/users/121

Params Authorization Headers (9) Body Scripts Tests Settings Cookies Beautify

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "name": "Diego Retana",
3   "email": "dars123@gmail.com",
4   "password": "contra1234"
5 }
```

Body Cookies Headers (8) Test Results

HTML Preview Visualize

200 OK 748 ms 286 B

1 User modified with ID: 121

This screenshot shows a POST request to 'localhost:3000/signin' with a JSON body containing a user object. The response status is 200 OK, and the body contains the message 'User modified with ID: 121'.

GET localhost:3000/signin

localhost:3000/signin

Send

GET localhost:3000/users

Params Authorization Headers (9) Body Scripts Tests Settings Cookies Beautify

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "name": "Diego Retana",
3   "email": "dars123@gmail.com",
4   "password": "contra1234"
5 }
```

Body Cookies Headers (8) Test Results

JSON Preview Visualize

200 OK 664 ms 7.84 KB

```
368 [
369   {
370     "id": 120,
371     "name": "Diego",
372     "email": "dars@gmail.com",
373     "password": "$2b$10$QPWTNyg1hWYJLseBh4U06uW3Bauk70IE0EndD6Cip5KbdTwGxxS.W"
374   },
375   {
376     "id": 121,
377     "name": "Diego Retana",
378     "email": "dars123@gmail.com",
379     "password": "$2b$10$w7pWrBBTWOIkXWINoRVcI07sLDz/0b389EM/Zr61p49z.vUSh0sMq"
380   }
]
```

Runner Start Proxy Cookies Vault Trash

This screenshot shows a GET request to 'localhost:3000/users'. The response status is 200 OK, and the body is a JSON array containing two user objects with IDs 120 and 121.

DEL localhost:3000/signin

localhost:3000/signin

DELETE localhost:3000/users/120

Send

Params Authorization Headers (9) Body Scripts Tests Settings Cookies

Auth Type Bearer Token

Token

The authorization header will be automatically generated when you send the request. Learn more about [Bearer Token](#) authorization.

Body Cookies Headers (8) Test Results

HTML Preview Visualize

200 OK 743 ms 285 B

1 User deleted with ID: 120

This screenshot shows a Postman request for a DELETE operation on the endpoint localhost:3000/users/120. The request is made with a Bearer Token authentication type. The response is a 200 OK status with a duration of 743 ms and a body size of 285 B. The response body contains the message "User deleted with ID: 120".

GET localhost:3000/signin

localhost:3000/signin

GET localhost:3000/users

Send

Params Authorization Headers (9) Body Scripts Tests Settings Cookies

Auth Type Bearer Token

Token

The authorization header will be automatically generated when you send the request. Learn more about [Bearer Token](#) authorization.

Body Cookies Headers (8) Test Results

{ } JSON Preview Visualize

200 OK 648 ms 7.72 KB

```
362 {  
363   "id": 119,  
364   "name": "Pedro",  
365   "email": "pedro20@gmail.com",  
366   "password": null  
367 },  
368 {  
369   "id": 121,  
370   "name": "Diego Retana",  
371   "email": "darsi23@gmail.com",  
372   "password": "$2b$10$w7pWrBBTwOIkXWINoRVcI07sL0z/8b389EM/ZrC1p49Z.vUSh0sMq"  
373 },
```

Runner Start Proxy Cookies Vault Trash

This screenshot shows a Postman request for a GET operation on the endpoint localhost:3000/users. The request is made with a Bearer Token authentication type. The response is a 200 OK status with a duration of 648 ms and a body size of 7.72 KB. The response body contains two user objects, each with an id, name, email, and password field.