

Evidencias de Pruebas Endpoint (register, login, CRUD marcas, CRUD productos)

Login y Registro de Usuarios.

LOGIN DE USUARIO.

The screenshot shows the Postman interface for a login request. The URL is `http://127.0.0.1:8000/api/login` and the method is **POST**. The request body is in **form-data** format with the following parameters:

key	value
name	Eduardo
email	eduardo@example.com
password	12345678
password_confirmation	12345678

The response status is **200 OK** with a response time of 2.95 s and a body size of 483 B. The response body is in JSON format:

```
{  "message": "Login exitoso",  "token": "2|cnusCZCik91wUwYcfc1J9Na1J2P7wSdnQWrbNgd87a03188",  "user": {    "id": 1,    "name": "Eduardo",    "email": "eduardo@example.com",    "email_verified_at": null,    "created_at": "2025-08-04T01:45:09.000000Z",    "updated_at": "2025-08-04T01:45:09.000000Z"  }}
```

REGISTRO DE USUARIOS.

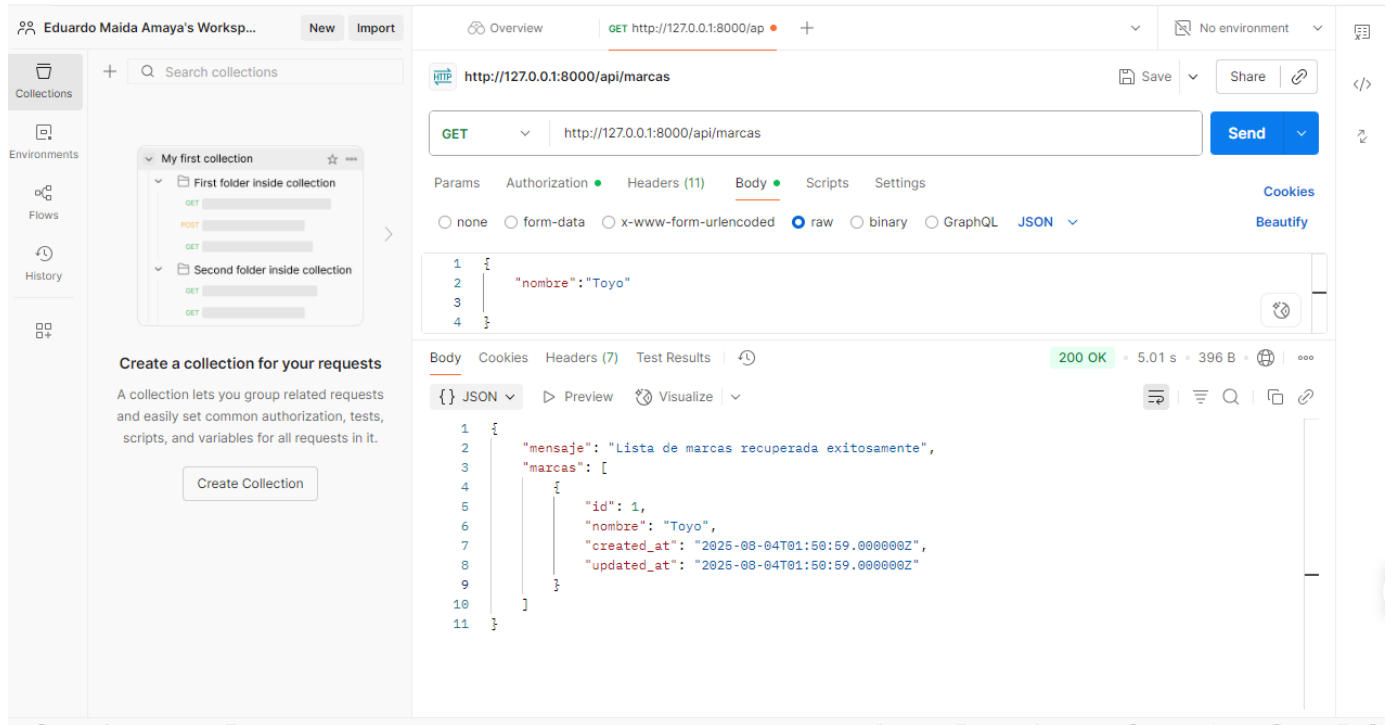
The screenshot shows the Postman interface for a user registration request. The URL is `http://127.0.0.1:8000/api/register` and the method is **POST**. The request body is in **form-data** format with the following parameters:

key	value
name	Eduardo
email	eduardo@example.com
password	12345678
password_confirmation	12345678

The response status is **201 Created** with a response time of 3.58 s and a body size of 482 B. The response body is in JSON format:

```
{  "message": "Usuario registrado correctamente",  "token": "1|AubTcc6PlD8PmHvIlgYAs0PVbF888okg1N1P92Awebb16936",  "user": {    "name": "Eduardo",    "email": "eduardo@example.com",    "updated_at": "2025-08-04T01:45:09.000000Z",    "created_at": "2025-08-04T01:45:09.000000Z",    "id": 1  }}
```

CRUD MARCA INDEX



GET `http://127.0.0.1:8000/api/marcas` **Send**

Params Authorization Headers (11) Body Scripts Settings

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

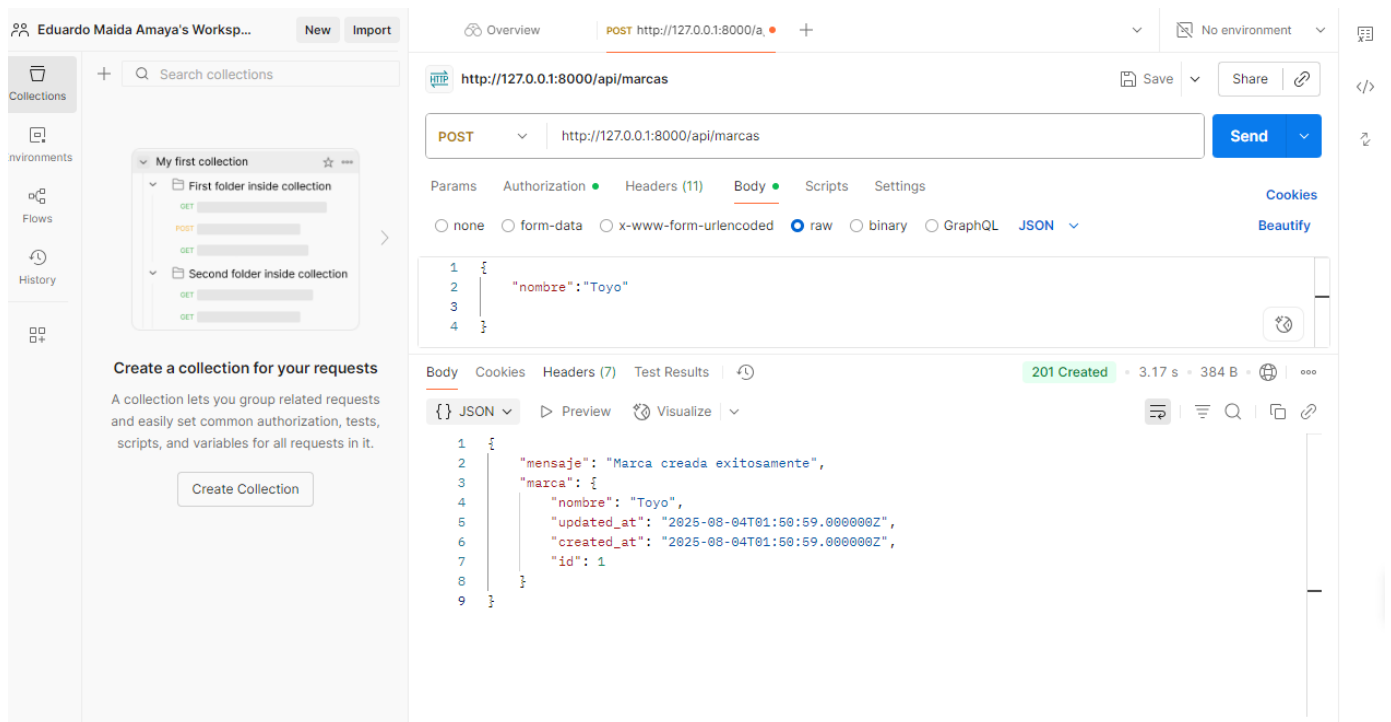
```
1 {
2   "nombre": "Toyo"
3 }
4 }
```

Body Cookies Headers (7) Test Results **200 OK** 5.01 s 396 B

JSON Preview Visualize

```
1 {
2   "mensaje": "Lista de marcas recuperada exitosamente",
3   "marcas": [
4     {
5       "id": 1,
6       "nombre": "Toyo",
7       "created_at": "2025-08-04T01:50:59.000000Z",
8       "updated_at": "2025-08-04T01:50:59.000000Z"
9     }
10  ]
11 }
```

CRUD MARCA CREATE



POST `http://127.0.0.1:8000/api/marcas` **Send**

Params Authorization Headers (11) Body Scripts Settings

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

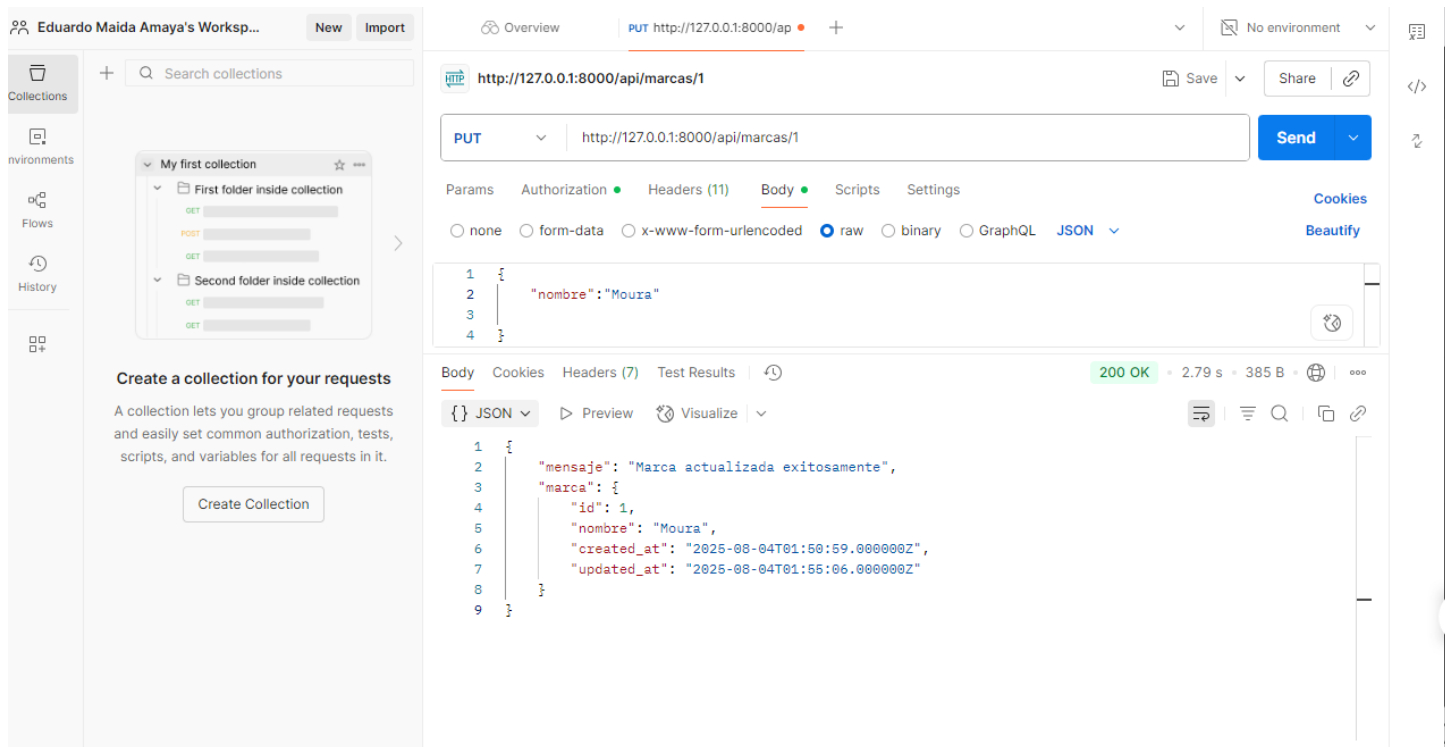
```
1 {
2   "nombre": "Toyo"
3 }
4 }
```

Body Cookies Headers (7) Test Results **201 Created** 3.17 s 384 B

JSON Preview Visualize

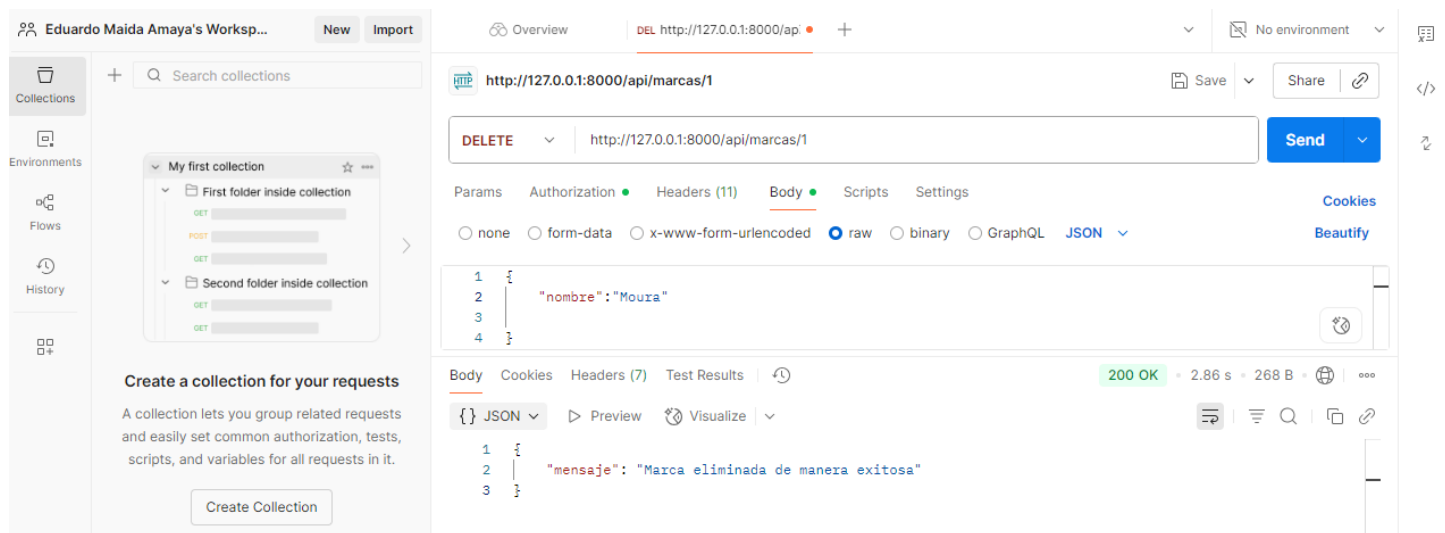
```
1 {
2   "mensaje": "Marca creada exitosamente",
3   "marca": {
4     "nombre": "Toyo",
5     "updated_at": "2025-08-04T01:50:59.000000Z",
6     "created_at": "2025-08-04T01:50:59.000000Z",
7     "id": 1
8   }
9 }
```

CRUD MARCA EDITAR



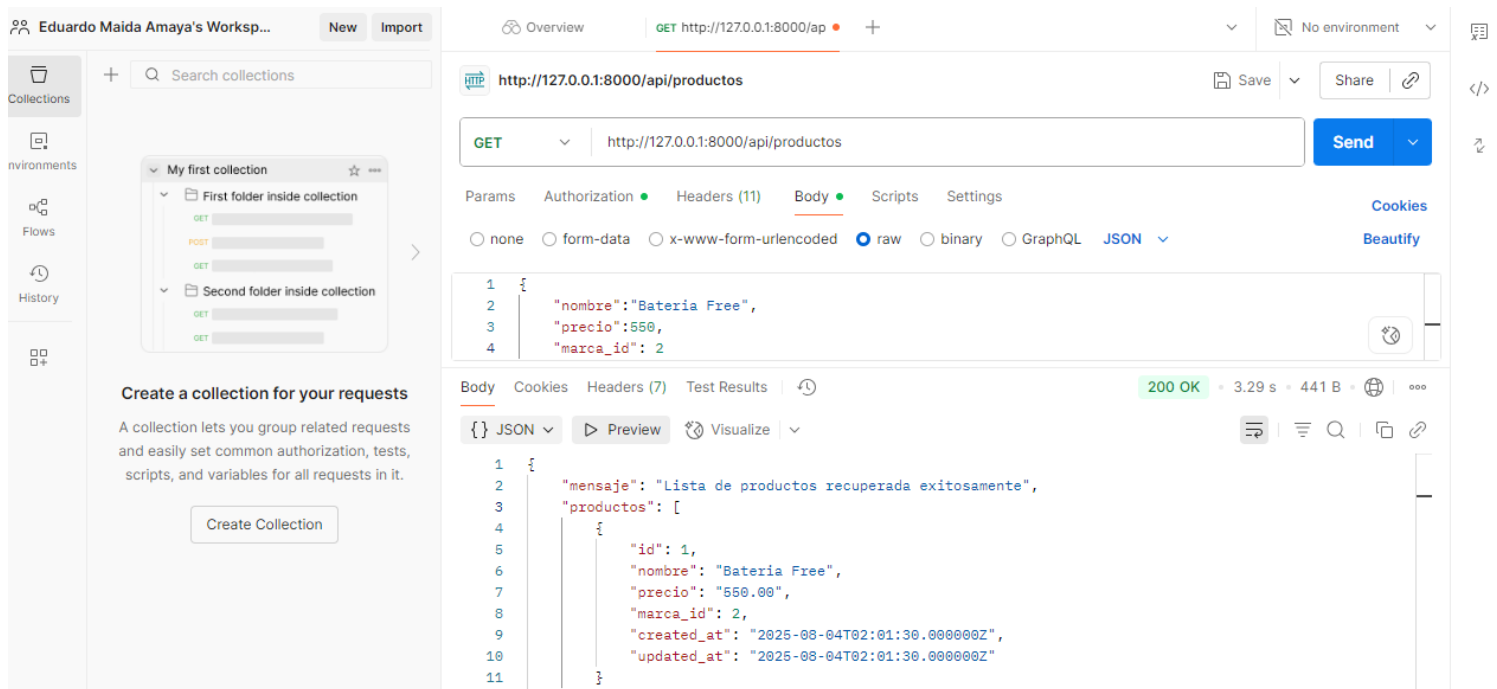
The screenshot shows the Postman interface for a PUT request to `http://127.0.0.1:8000/api/marcas/1`. The request body is a JSON object: `{ "nombre": "Mouza" }`. The response is a 200 OK status with a response time of 2.79 s and a body size of 385 B. The response body is a JSON object: `{ "mensaje": "Marca actualizada exitosamente", "marca": { "id": 1, "nombre": "Mouza", "created_at": "2025-08-04T01:58:59.000000Z", "updated_at": "2025-08-04T01:55:06.000000Z" } }`.

CRUD MARCA ELIMINAR (DELETE)



The screenshot shows the Postman interface for a DELETE request to `http://127.0.0.1:8000/api/marcas/1`. The request body is a JSON object: `{ "nombre": "Mouza" }`. The response is a 200 OK status with a response time of 2.86 s and a body size of 268 B. The response body is a JSON object: `{ "mensaje": "Marca eliminada de manera exitosa" }`.

CRUD PRODUCTOS INDEX



Eduardo Maida Amaya's Worksp... New Import

Overview GET http://127.0.0.1:8000/ap

http://127.0.0.1:8000/api/productos

GET http://127.0.0.1:8000/api/productos

Params Authorization Headers (11) Body Scripts Settings

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

200 OK • 3.29 s • 441 B

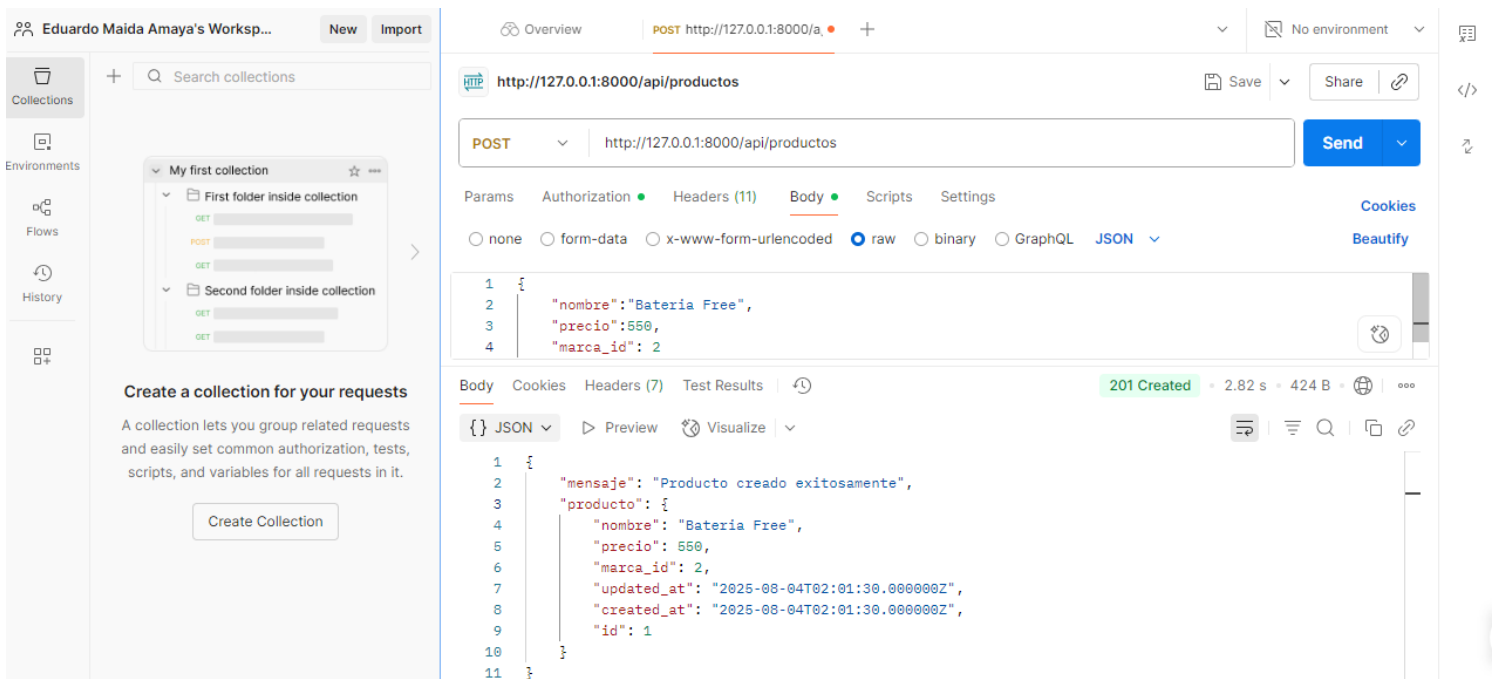
```
1 {
2   "nombre": "Bateria Free",
3   "precio": 550,
4   "marca_id": 2
}
```

Body Cookies Headers (7) Test Results

JSON Preview Visualize

```
1 {
2   "mensaje": "Lista de productos recuperada exitosamente",
3   "productos": [
4     {
5       "id": 1,
6       "nombre": "Bateria Free",
7       "precio": "550.00",
8       "marca_id": 2,
9       "created_at": "2025-08-04T02:01:30.000000Z",
10      "updated_at": "2025-08-04T02:01:30.000000Z"
11    }
12  ]
13 }
```

CRUD PRODUCTO CREATE



Eduardo Maida Amaya's Worksp... New Import

Overview POST http://127.0.0.1:8000/a

http://127.0.0.1:8000/api/productos

POST http://127.0.0.1:8000/api/productos

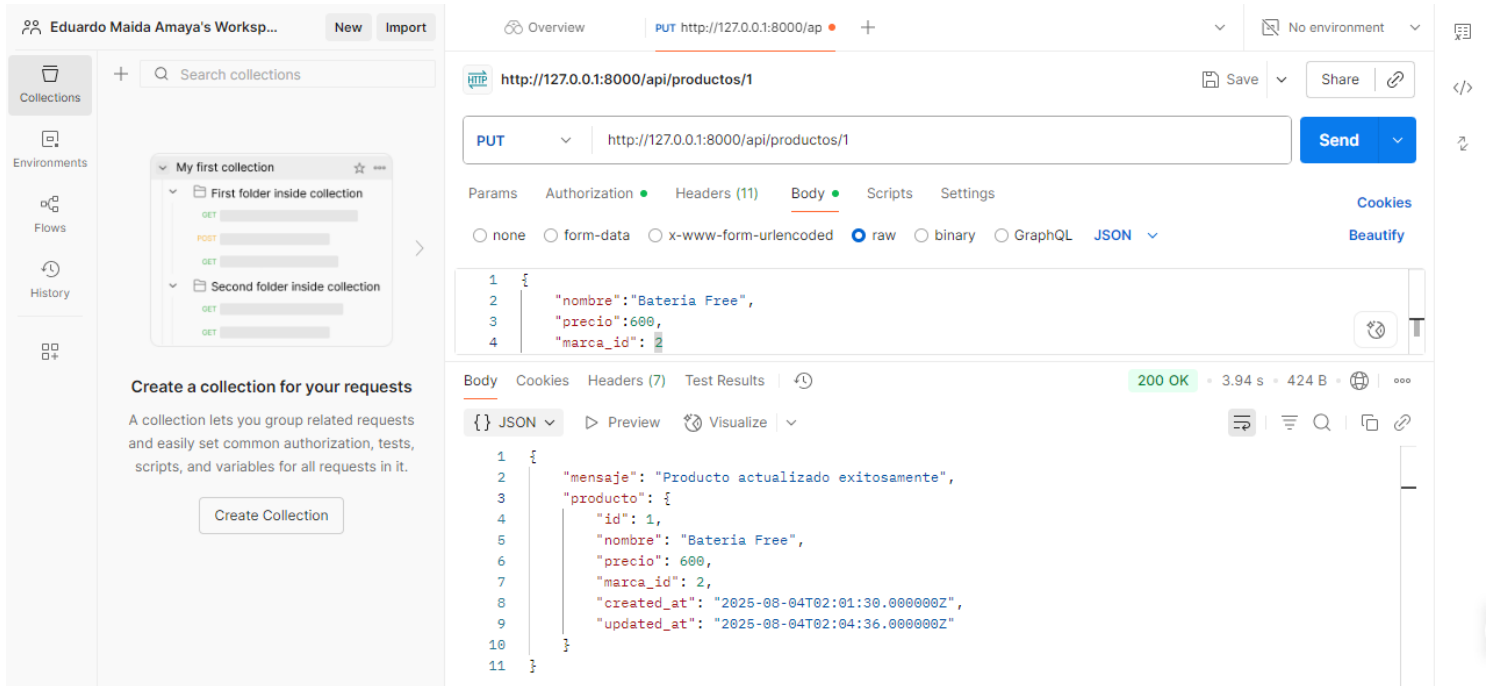
Params Authorization Headers (11) Body Scripts Settings

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

201 Created • 2.82 s • 424 B

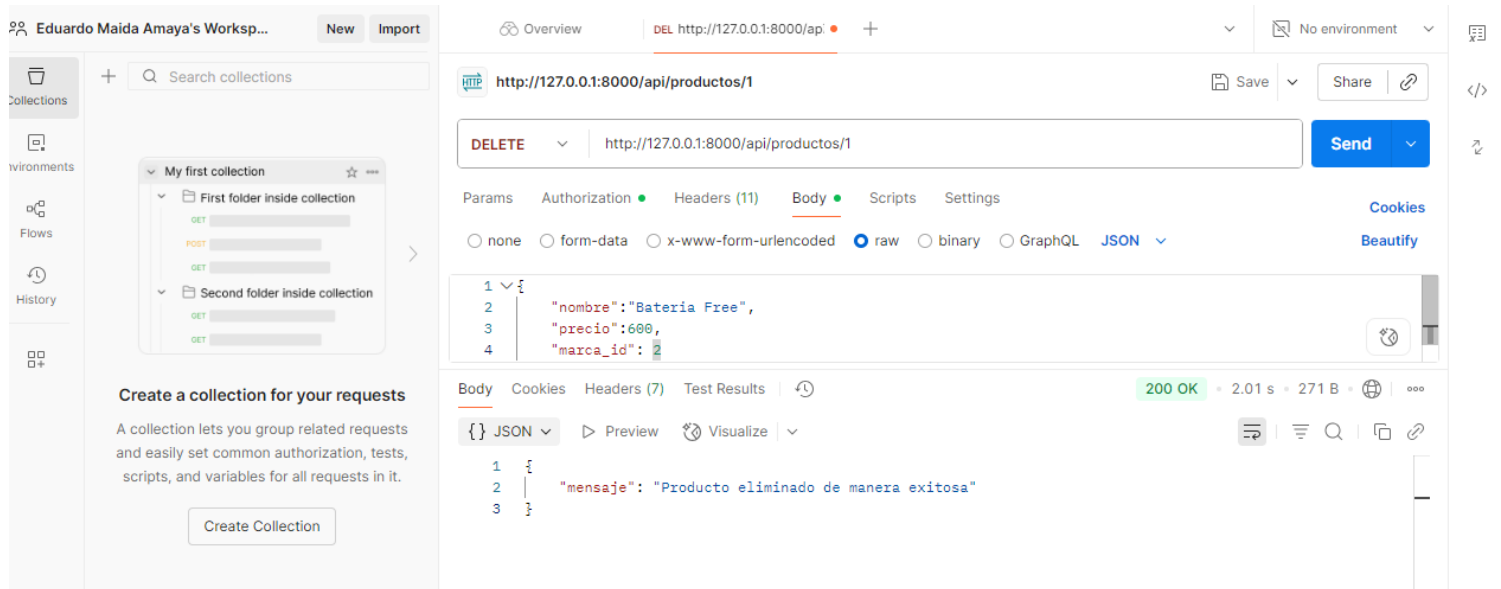
```
1 {
2   "mensaje": "Producto creado exitosamente",
3   "producto": {
4     "nombre": "Bateria Free",
5     "precio": 550,
6     "marca_id": 2,
7     "updated_at": "2025-08-04T02:01:30.000000Z",
8     "created_at": "2025-08-04T02:01:30.000000Z",
9     "id": 1
10  }
11 }
```

CRUD PRODUCTO EDITAR



The screenshot shows the Postman interface for a PUT request to `http://127.0.0.1:8000/api/productos/1`. The request body is a JSON object: `{ "nombre": "Bateria Free", "precio": 600, "marca_id": 2 }`. The response is a 200 OK status with a JSON body: `{ "mensaje": "Producto actualizado exitosamente", "producto": { "id": 1, "nombre": "Bateria Free", "precio": 600, "marca_id": 2, "created_at": "2025-08-04T02:01:30.000000Z", "updated_at": "2025-08-04T02:04:36.000000Z" } }`.

CRUD PRODUCTO ELIMINAR



The screenshot shows the Postman interface for a DELETE request to `http://127.0.0.1:8000/api/productos/1`. The request body is a JSON object: `{ "nombre": "Bateria Free", "precio": 600, "marca_id": 2 }`. The response is a 200 OK status with a JSON body: `{ "mensaje": "Producto eliminado de manera exitosa" }`.

