

EVIDENCIAS CAPTURAS (POSTMAN)

REGISTER

The screenshot shows the Postman interface with a POST request to `http://127.0.0.1:8000/api/auth/register`. The request body is form-data with the following fields:

Key	Type	Value
name	Text	Israel
password	Text	Israel2025*
password_confirmation	Text	Israel2025*
email	Text	Israel@gmail.com

The response is a 201 Created status with a JSON body:

```
{  "message": "Usuario registrado correctamente",  "token": "1jgdQSLjR802o24okRQKVZ8sKj38XKApIHn2Um003d4e2ad",  "user": {    "name": "Israel",    "email": "Israel@gmail.com",    "updated_at": "2025-08-03T06:45:30.000000Z",    "created_at": "2025-08-03T06:45:30.000000Z",    "id": 1  }}
```

LOGIN

The screenshot shows the Postman interface with a POST request to `http://127.0.0.1:8000/api/auth/login`. The request body is form-data with the following fields:

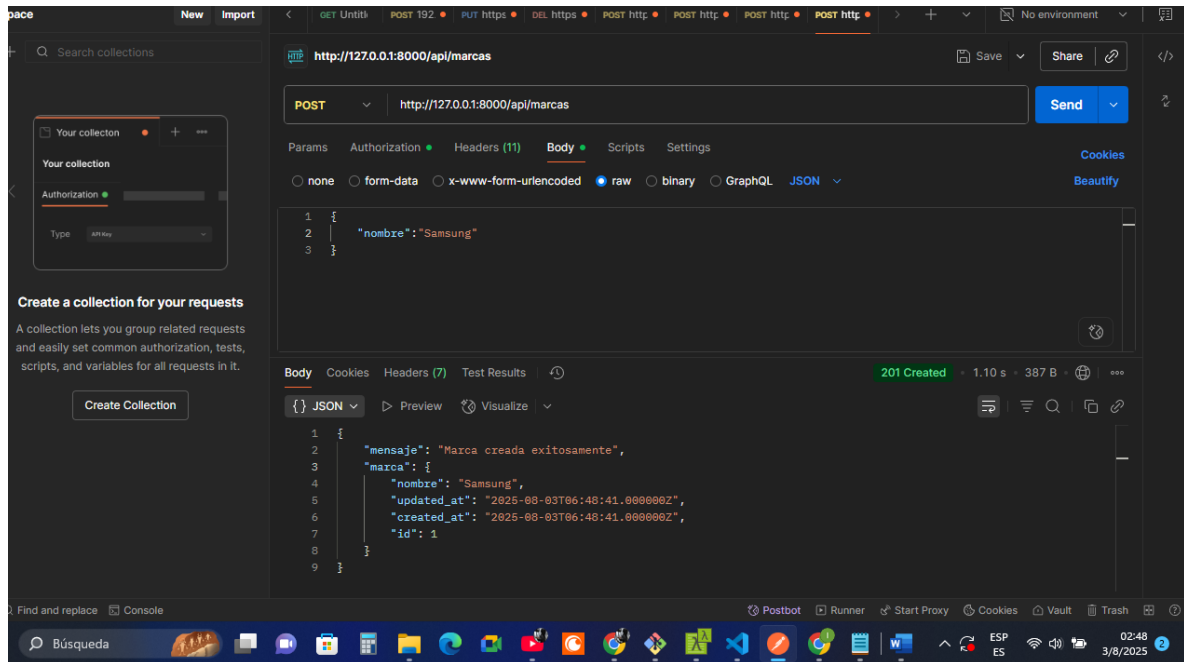
Key	Type	Value
email	Text	Israel@gmail.com
password	Text	Israel2025*

The response is a 200 OK status with a JSON body:

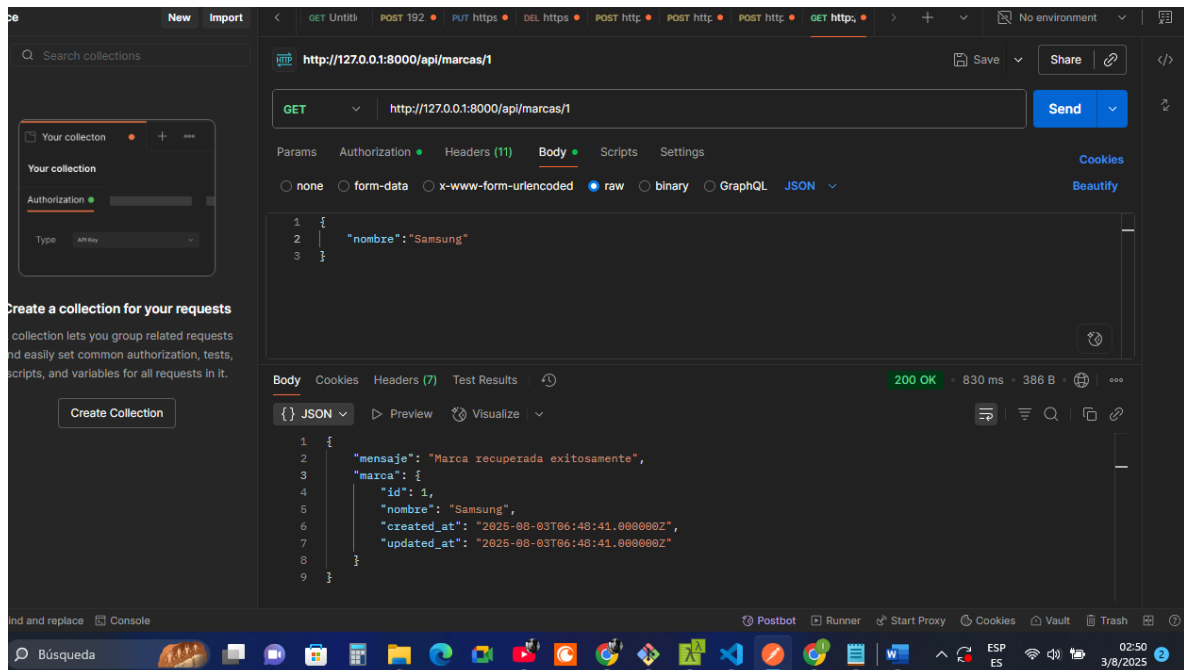
```
{  "message": "Login exitoso",  "token": "2jDCK156MLJ4CdwjJUS1Tgb67lIfGi7Fwt8BII83t38ff6fa2e",  "user": {    "id": 1,    "name": "Israel",    "email": "Israel@gmail.com",    "email_verified_at": null,    "created_at": "2025-08-03T06:45:30.000000Z",    "updated_at": "2025-08-03T06:45:30.000000Z"  }}
```

CRUD Marcas

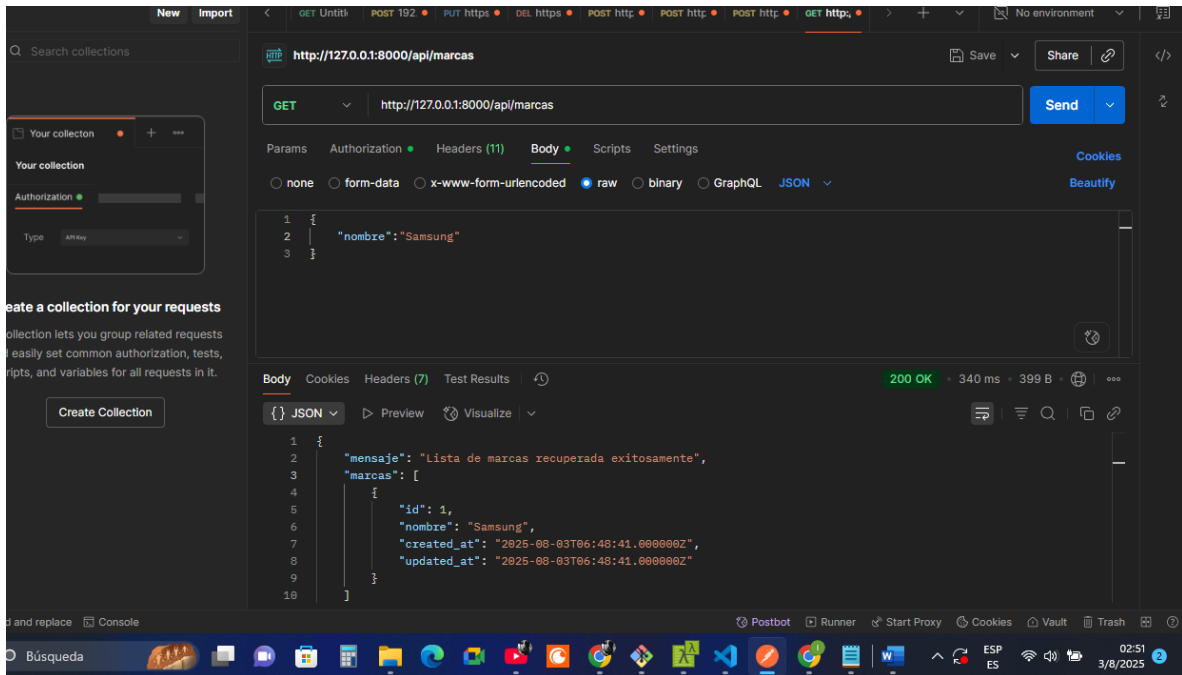
Create (marcas)



Show (marcas)



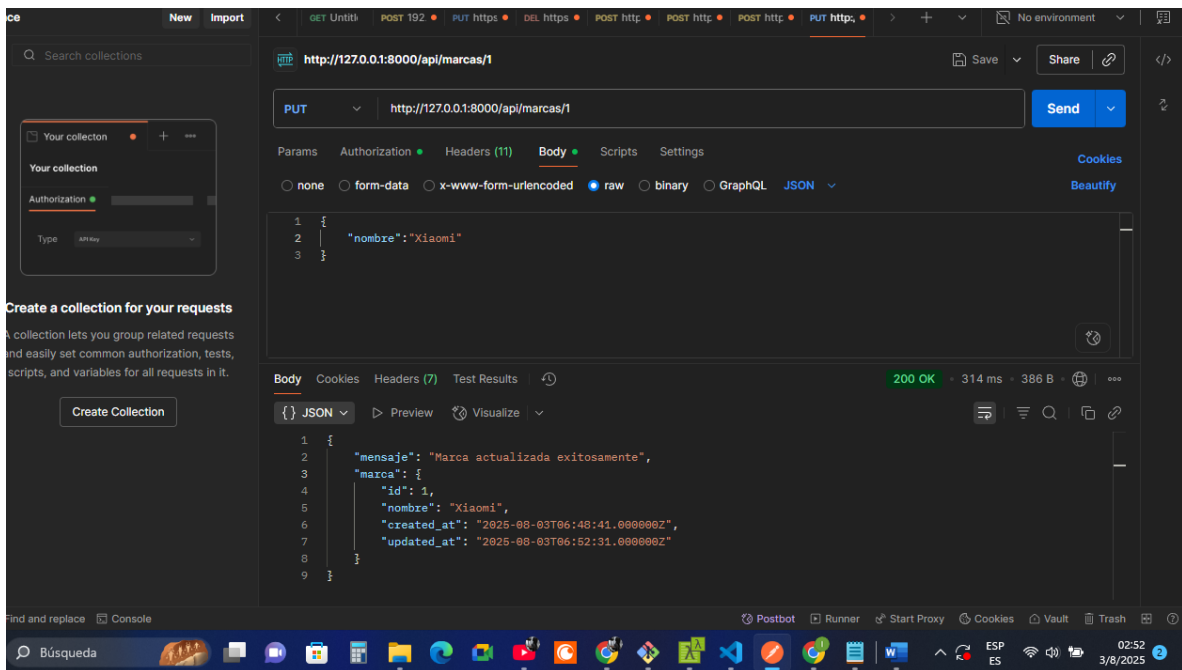
Index (marcas)



The screenshot shows the Postman interface with a GET request to `http://127.0.0.1:8000/api/marcas`. The request is successful, returning a 200 OK status. The response body is a JSON object with the following structure:

```
1 {
2   "mensaje": "Lista de marcas recuperada exitosamente",
3   "marcas": [
4     {
5       "id": 1,
6       "nombre": "Samsung",
7       "created_at": "2025-08-03T06:48:41.000000Z",
8       "updated_at": "2025-08-03T06:48:41.000000Z"
9     }
10  ]
}
```

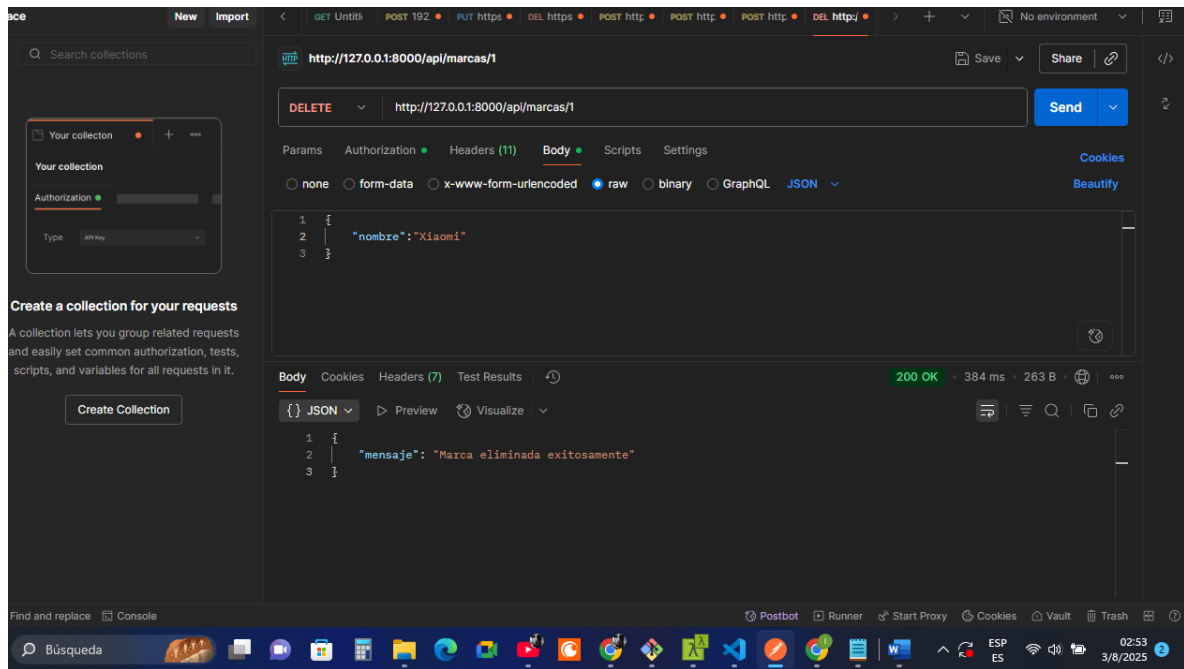
Update (marcas)



The screenshot shows the Postman interface with a PUT request to `http://127.0.0.1:8000/api/marcas/1`. The request is successful, returning a 200 OK status. The response body is a JSON object with the following structure:

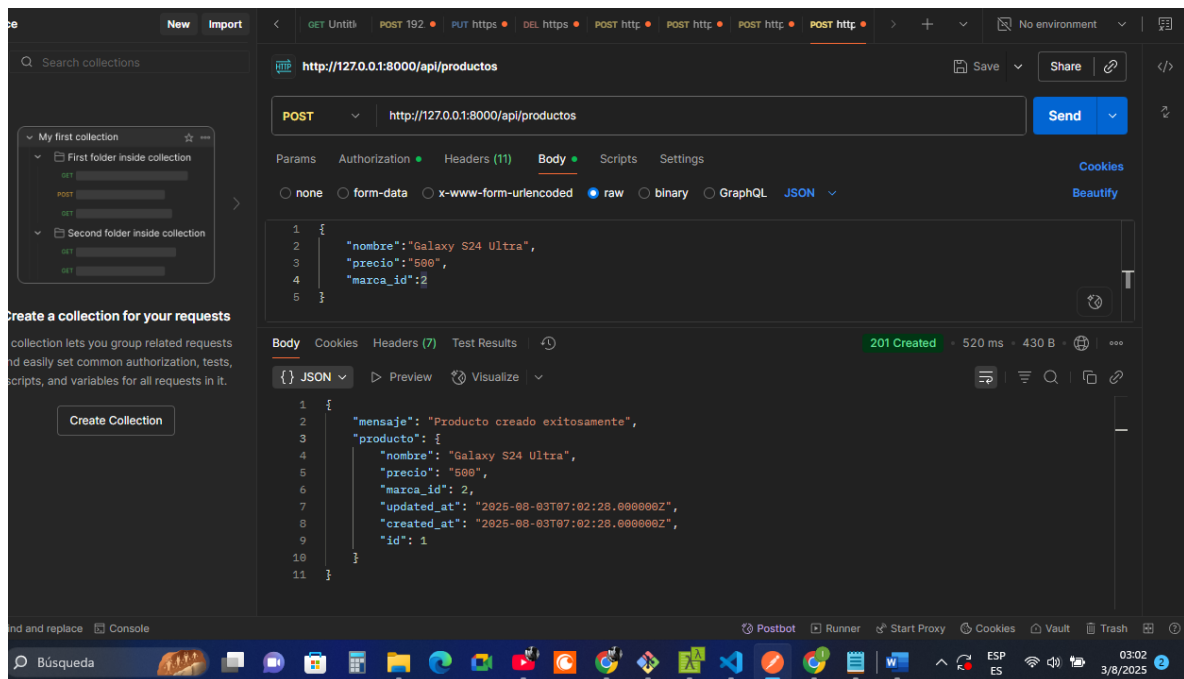
```
1 {
2   "mensaje": "Marca actualizada exitosamente",
3   "marca": {
4     "id": 1,
5     "nombre": "Xiaomi",
6     "created_at": "2025-08-03T06:48:41.000000Z",
7     "updated_at": "2025-08-03T06:52:31.000000Z"
8   }
9 }
```

Destroy/delete (marcas)



CRUD PRODUCTOS

Create (productos)



Index (productos)

The screenshot shows the Postman interface with a GET request to `http://127.0.0.1:8000/api/productos`. The request is sent, and the response is a 200 OK status with a 273 ms response time and 445 B of data. The response body is a JSON object with the following structure:

```
1 {
2   "mensaje": "Lista de productos recuperada exitosamente",
3   "productos": [
4     {
5       "id": 1,
6       "nombre": "Galaxy S24 Ultra",
7       "precio": "500.00",
8       "marca_id": 2,
9       "created_at": "2025-08-03T07:02:28.000000Z",
10      "updated_at": "2025-08-03T07:02:28.000000Z"
11     }
12   ]
13 }
```

Show (productos)

The screenshot shows the Postman interface with a GET request to `http://127.0.0.1:8000/api/productos/1`. The request is sent, and the response is a 200 OK status with a 276 ms response time and 432 B of data. The response body is a JSON object with the following structure:

```
1 {
2   "mensaje": "Producto recuperado exitosamente",
3   "producto": {
4     "id": 1,
5     "nombre": "Galaxy S24 Ultra",
6     "precio": "500.00",
7     "marca_id": 2,
8     "created_at": "2025-08-03T07:02:28.000000Z",
9     "updated_at": "2025-08-03T07:02:28.000000Z"
10   }
11 }
```

Update (productos)

The screenshot shows the Postman interface with a PUT request to `http://127.0.0.1:8000/api/productos/1`. The request body is a JSON object: `{ "nombre": "Galaxy S25 Pro", "precio": "700", "marca_id": 2 }`. The response is a 200 OK status with a JSON body: `{ "mensaje": "Producto actualizado exitosamente", "producto": { "id": 1, "nombre": "Galaxy S25 Pro", "precio": "700", "marca_id": 2, "created_at": "2025-08-03T07:02:28.000000Z", "updated_at": "2025-08-03T07:05:05.000000Z" } }`. The left sidebar shows a collection named "My first collection" with two folders: "First folder inside collection" and "Second folder inside collection". The bottom status bar shows the time as 03:05 on 3/8/2025.

Destroy/delete (productos)

The screenshot shows the Postman interface with a DELETE request to `http://127.0.0.1:8000/api/productos/1`. The request body is a JSON object: `{ "nombre": "Galaxy S25 Pro", "precio": "700", "marca_id": 2 }`. The response is a 200 OK status with a JSON body: `{ "mensaje": "Producto eliminado exitosamente" }`. The left sidebar shows the same collection structure as the previous screenshot. The bottom status bar shows the time as 03:05 on 3/8/2025.