

1. Un post de un artículo

The screenshot shows a REST client interface with a POST request to `http://localhost:3000/webresources/generic/plantas`. The request body is a JSON object with the following fields: `nombre`, `imagen`, `categoria`, and `precio`. The response body is a JSON object with `ok` set to `true` and `msg` set to `"post API plantas"`.

```
POST http://localhost:3000/webresources/generic/plantas

{
  "nombre": "planta prueba evelyn",
  "imagen": "evelyn.jpg",
  "categoria": "interior",
  "precio": "1"
}
```

```
{
  "ok": true,
  "msg": "post API plantas"
}
```

```
_id: ObjectId("620a3c3e024da275177a21c0")
nombre: "planta prueba evelyn"
imagen: "evelyn.jpg"
categoria: "interior"
precio: 1
__v: 0
```

2. Un put (actualización)

The screenshot shows a REST client interface with a PUT request to `http://localhost:3000/webresources/generic/plantas/620a3c3e024da275177a21c0`. The request body is a JSON object with the following fields:

```
1 {
2   "nombre": "planta actualizada prueba evelyn",
3   "imagen": "evelynactualizada.jpg",
4   "categoria": "actualiza",
5   "precio": "1"
6 }
```

The response body is also in JSON format, showing a successful update:

```
1 {
2   "ok": true,
3   "msg": "post API planta",
4   "body": {
5     "nombre": "planta actualizada prueba evelyn",
6     "imagen": "evelynactualizada.jpg",
7     "categoria": "actualiza",
8     "precio": "1"
9   }
10 }
```

3. Un get de un artículo

`http://localhost:3000/webresources/generic/plantas/620a3c3e024da275177a21c0`

The screenshot shows a REST client interface with a GET request to `http://localhost:3000/webresources/generic/plantas/620a3c3e024da275177a21c0`. The response body is a JSON object representing the plant article:

```
1 {
2   "nombre": "planta actualizada prueba evelyn",
3   "imagen": "evelynactualizada.jpg",
4   "categoria": "actualiza",
5   "precio": 1,
6   "__v": 0,
7   "id": "620a3c3e024da275177a21c0"
8 }
```

4. Un get de todos los artículos

GET ▼ http://localhost:3000/webresources/generic/plantas

Params Authorization Headers (10) Body ● Pre-request Script Tests Settings

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON ▼

```
1  [
2    {
3      "nombre": "orquidea",
4      "imagen": "or.jpg",
5      "categoria": "interior",
6      "precio": 3,
7      "__v": 0,
8      "id": "620a29daa1325cea35d96f69"
9    },
10   {
11     "nombre": " margarita",
12     "imagen": "margarita.jpg",
13     "categoria": "exterior",
14     "precio": 5,
15     "__v": 0,
16     "id": "620a38076111b05e18865ec8"
17   }
18 ]
```

GET ▼ http://localhost:3000/webresources/generic/plantas

Params Authorization Headers (10) Body ● Pre-request Script Tests Settings

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON ▼

```
18  {
19    "nombre": "rosa",
20    "imagen": "rosa.jpg",
21    "categoria": "exterior",
22    "precio": 2,
23    "__v": 0,
24    "id": "620a3bed024da275177a21bc"
25  },
26  {
27    "nombre": "planta actualizada prueba evelyn",
28    "imagen": "evelynactualizada.jpg",
29    "categoria": "actualiza",
30    "precio": 1,
31    "__v": 0,
32    "id": "620a3c3e024da275177a21c0"
33  }
34 ]
```

5. Un delete

http://localhost:3000/webresources/generic/plantas/620a3c3e024da275177a21c0


DELETE ⌵ http://localhost:3000/webresources/generic/plantas/620a3c3e024da275177a21c0

Params Authorization Headers (10) **Body** ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL **JSON** ⌵

1 4

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize **JSON** ⌵ 

```
1 {
2   "ok": true,
3   "msg": "delete API"
4 }
```

QUERY RESULTS 1-2 OF 2

```
_id: ObjectId("620a29daa1325cea35d96f69")
nombre: "orquidea"
imagen: "or.jpg"
categoria: "interior"
precio: 3
__v: 0
```

```
_id: ObjectId("620a38076111b05e18865ec8")
nombre: " margarita"
imagen: "margarita.jpg"
categoria: "exterior"
precio: 5
__v: 0
```

