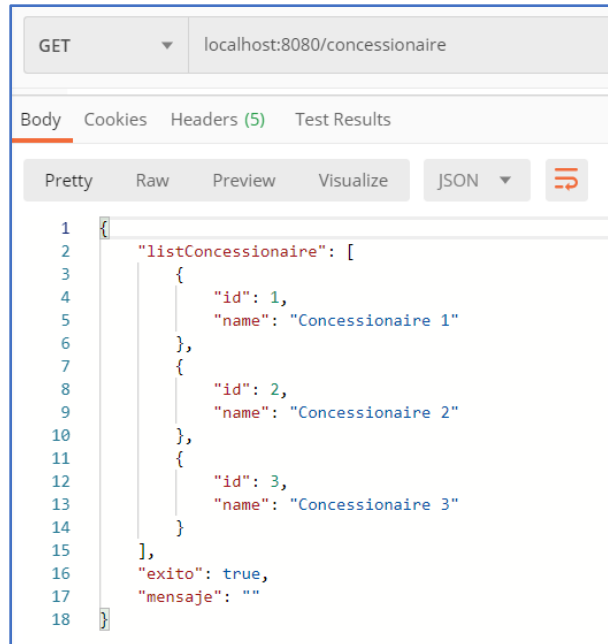


5. Capturas Postman

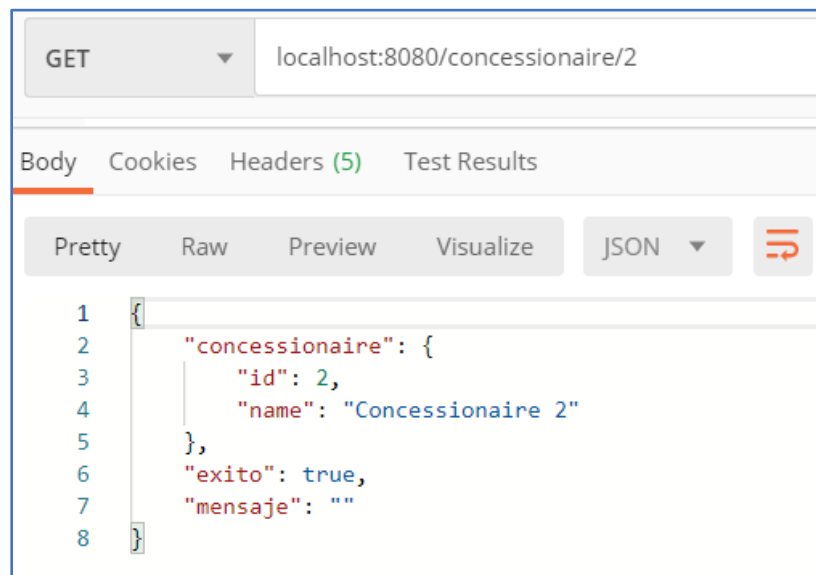
Dado que habilite tantos end points, solo agregare las capturas de pantalla de los relacionados a **concessionaire** y a **device**, ya que con estos dos se puede evidenciar tanto el CRUD de una tabla con OneToMany (Concessionaire – Bus) y ManyToOne(Device – Bus, Device – DeviceType).

Concessionaire:

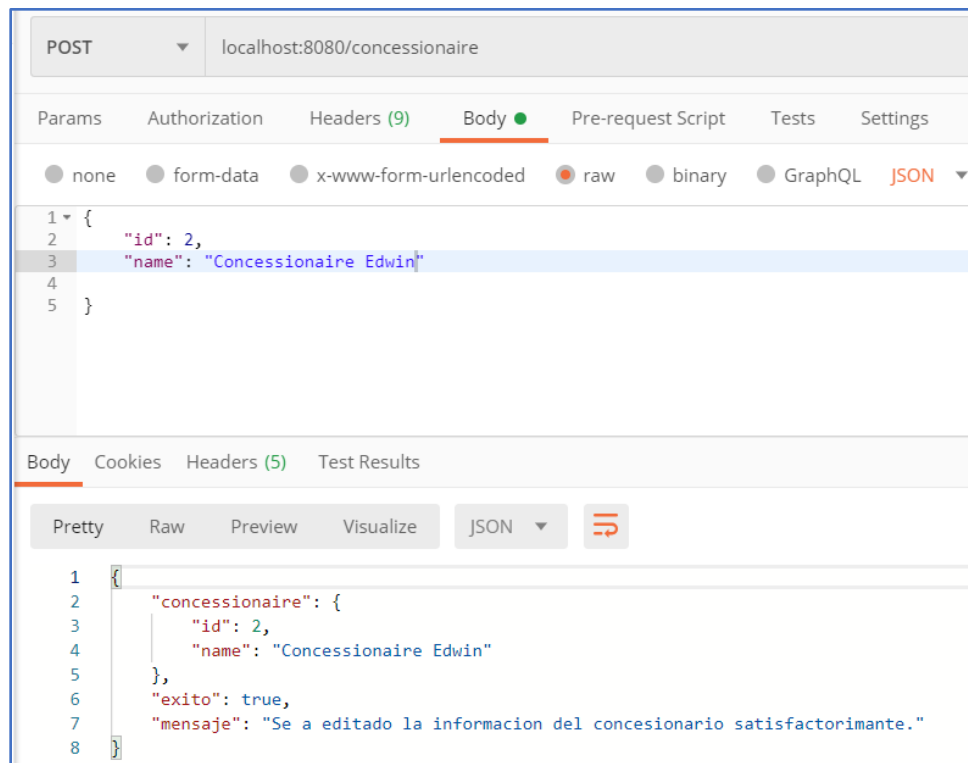
Primer Get, lista los concesionarios.



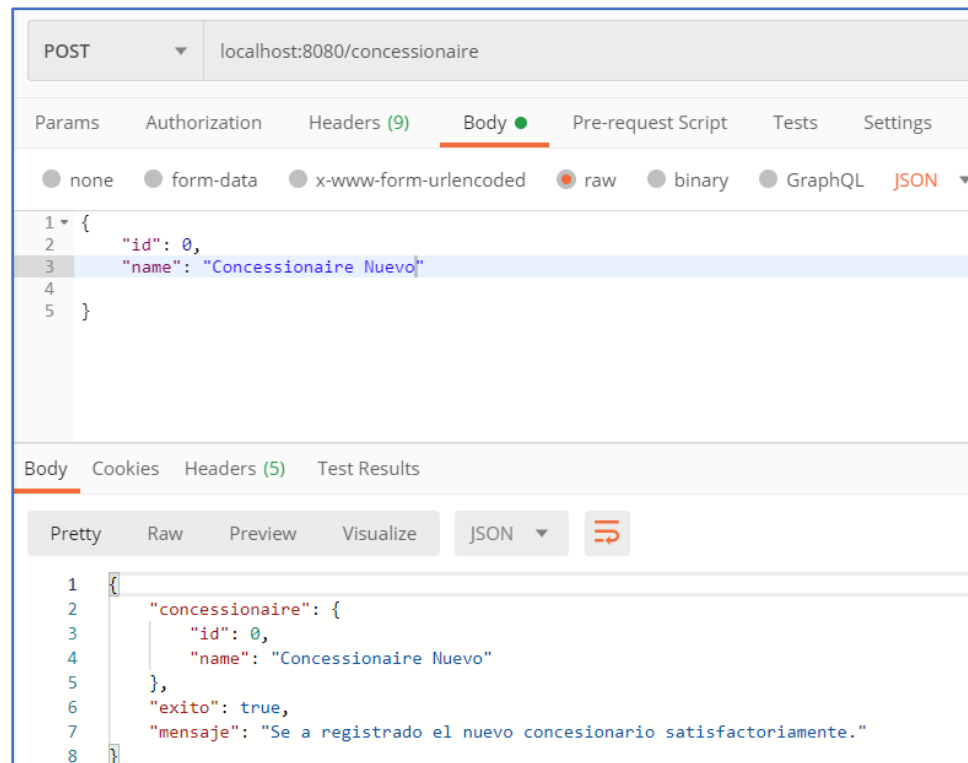
Segundo Get, Trae un concesionario en específico, en este caso el número 2.



Primer Post, para editarlo, y ponerle el nombre de “Concessionaire Edwin”.



Segundo Post, para agregar uno nuevo.



Estos lo podemos validar haciendo nuevamente un get list

```
GET localhost:8080/concessionaire

Pretty Raw Preview Visualize JSON ↺

1 {
2   "listConcessionaire": [
3     {
4       "id": 1,
5       "name": "Concessionaire 1"
6     },
7     {
8       "id": 2,
9       "name": "Concessionaire Edwin"
10    },
11    {
12      "id": 3,
13      "name": "Concessionaire 3"
14    },
15    {
16      "id": 4,
17      "name": "Concessionaire Nuevo"
18    }
19  ],
20  "exito": true,
21  "mensaje": ""
22 }
```

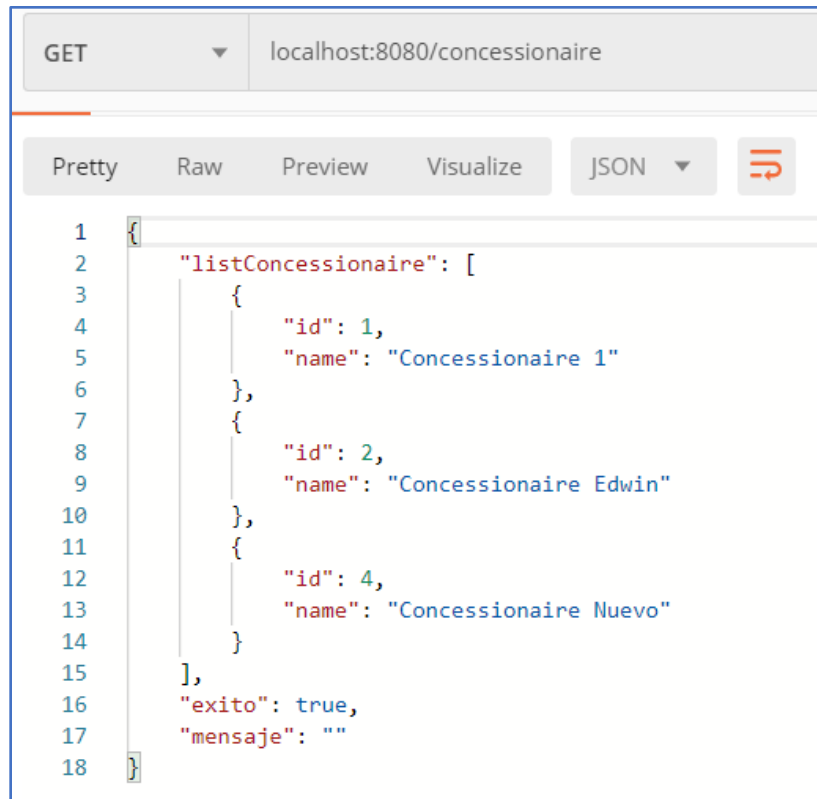
Ahora haremos un Delete, del concesionario 3.

```
DELETE localhost:8080/concessionaire/3

Pretty Raw Preview Visualize JSON ↺

1 {
2   "exito": true,
3   "mensaje": "Se a eliminado la informacion del concesionario."
4 }
```

Y lo comprobamos con el get de la lista nuevamente



The screenshot shows a REST client interface with a GET request to `localhost:8080/concessionaire`. The response is a JSON object with a list of concessionaires and status fields. The JSON is displayed in a 'Pretty' format. The response body is as follows:

```
1 {
2   "listConcessionaire": [
3     {
4       "id": 1,
5       "name": "Concessionaire 1"
6     },
7     {
8       "id": 2,
9       "name": "Concessionaire Edwin"
10    },
11    {
12      "id": 4,
13      "name": "Concessionaire Nuevo"
14    }
15  ],
16  "exito": true,
17  "mensaje": ""
18 }
```

Device:

Para esta haremos el mismo proceso, mas el consumo del end point para trar la lista de "Device" para un bus especifico.



The screenshot shows a REST client interface with a GET request to `localhost:8080/device`. The response is a JSON array of two device objects, formatted in 'Pretty' mode. The first device has ID 1, IP 192.168.0.10, and is associated with a bus of type 'Bi-articulado'. The second device has ID 2, IP 192.168.0.11, and is associated with a bus of type 'Autobus'.

```
1 {
2   "listDevice": [
3     {
4       "id": 1,
5       "ip": "192.168.0.10",
6       "deviceType": {
7         "id": 2,
8         "name": "NVR"
9       },
10      "bus": {
11        "id": 1,
12        "type": "Bi-articulado",
13        "motor": "Electric",
14        "brakes": "Freno Regenerativo",
15        "consessionaire": {
16          "id": 1,
17          "name": "Concessionaire 1"
18        }
19      },
20      "status": "Active"
21    },
22    {
23      "id": 2,
24      "ip": "192.168.0.11",
25      "deviceType": {
26        "id": 2,
27        "name": "NVR"
28      },
29      "bus": {
30        "id": 2,
31        "type": "Autobus",
32        "motor": "Hybrid",
33        "brakes": "Frenos mecánicos",
```

Ahora preguntamos por uno en específico, en este caso el “3”

```
GET localhost:8080/device/3

Pretty Raw Preview Visualize JSON

1 {
2   "exito": true,
3   "mensaje": "",
4   "device": {
5     "id": 3,
6     "ip": "192.168.0.12",
7     "deviceType": {
8       "id": 1,
9       "name": "CANBUS"
10    },
11    "bus": {
12      "id": 3,
13      "type": "Bi-articulado",
14      "motor": "Electric",
15      "brakes": "Freno Regenerativo",
16      "concessionaire": {
17        "id": 2,
18        "name": "Concessionaire Edwin"
19      }
20    },
21    "status": "Active"
22  }
23 }
```

Ahora le modificaremos la ip.

```
POST localhost:8080/device

1 {
2   "id": 3,
3   "ip": "192.168.0.128",
4   "deviceType": {
5     "id": 1,
6     "name": "CANBUS"
7   },
8   "bus": {
9     "id": 3,
10    "type": "Bi-articulado",
11    "motor": "Electric",
12    "brakes": "Freno Regenerativo",
13    "concessionaire": {
14      "id": 2,
15      "name": "Concessionaire Edwin"
16    }
17  },
18  "status": "Active"
19 }

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

1 {
2   "exito": true,
3   "mensaje": "Se a editado la informacion del dispositivo satisfactorimante.",
4   "device": {
5     "id": 3,
6     "ip": "192.168.0.128",
7     "deviceType": {
8       "id": 1,
9       "name": "CANBUS"
10    },
11    "bus": {
12      "id": 3,
13      "type": "Bi-articulado",
14      "motor": "Electric",
15      "brakes": "Freno Regenerativo",
16      "concessionaire": {
17        "id": 2,
18        "name": "Concessionaire Edwin"
19      }
20    },
21    "status": "Active"
22  }
23 }
```

Ahora agregaremos un nuevo Device.

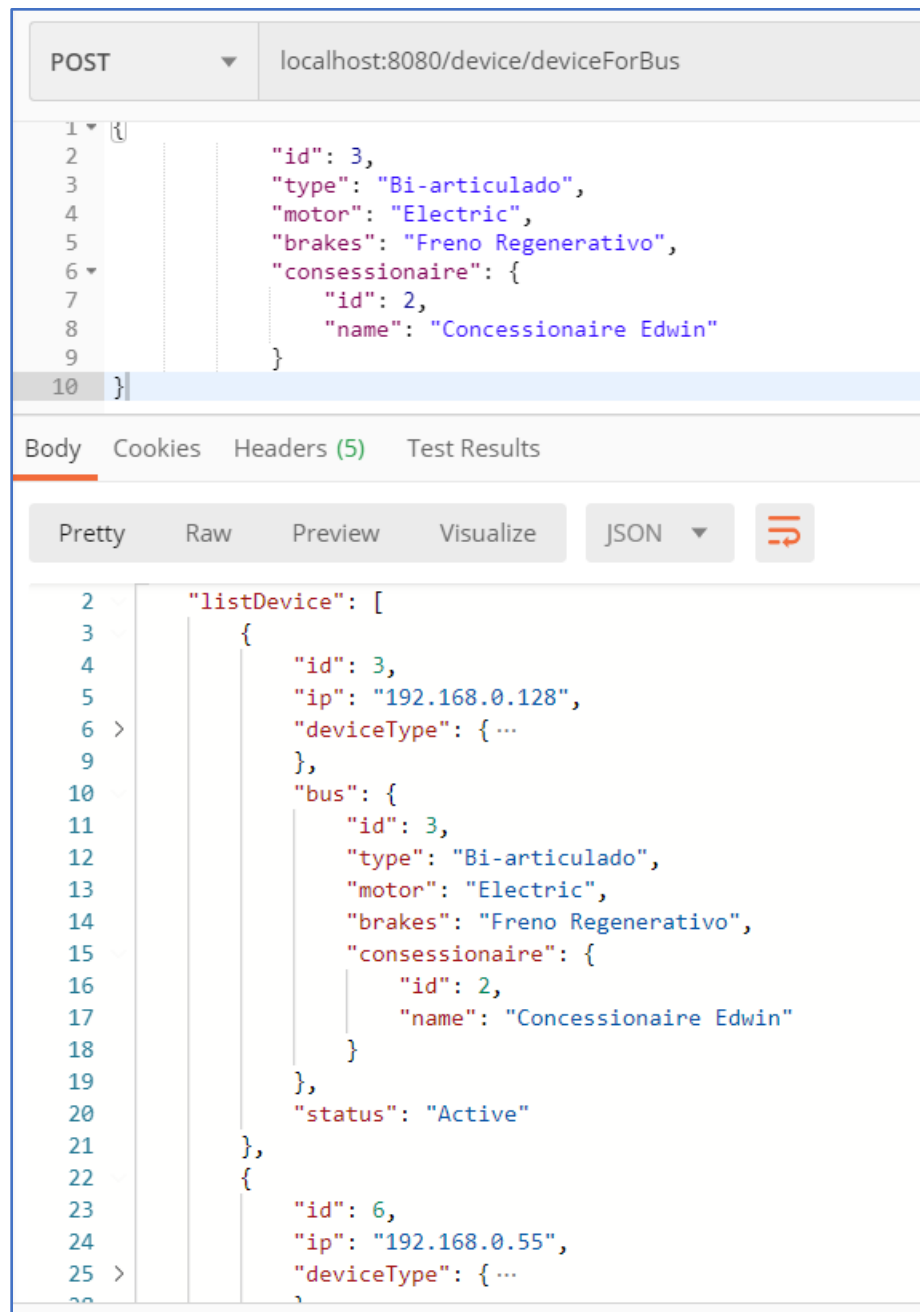
The screenshot shows a REST client interface with a POST request to `localhost:8080/device`. The request body is a JSON object with the following structure:

```
{
  "id": 0,
  "ip": "192.168.0.55",
  "deviceType": {
    "id": 1,
    "name": "CANBUS"
  },
  "bus": {
    "id": 3,
    "type": "Bi-articulado",
    "motor": "Electric"
  }
}
```

The response body is a JSON object with the following structure:

```
{
  "exito": true,
  "mensaje": "Se a registrado el nuevo dispositivo satisfactoriamente.",
  "device": {
    "id": 0,
    "ip": "192.168.0.55",
    "deviceType": {
      "id": 1,
      "name": "CANBUS"
    },
    "bus": {
      "id": 3,
      "type": "Bi-articulado",
      "motor": "Electric",
      "brakes": "Freno Regenerativo",
      "consessionaire": {
        "id": 2,
        "name": "Concessionaire Edwin"
      }
    }
  }
}
```

Ahora probaremos el end point para consultar los device de un bus específico.



The screenshot displays a REST client interface. At the top, a POST request is configured to `localhost:8080/device/deviceForBus`. The request body is a JSON object representing a device. Below the request, the 'Body' tab is selected, showing the response in a 'Pretty' JSON format. The response is an array of objects under the key `'listDevice'`. The first object in the array contains details for a device with `'id': 3`, including its `'ip'`, `'deviceType'`, `'bus'` information, and `'status': 'Active'`. The `'bus'` object is identical to the one in the request body. The second object in the array starts with `'id': 6`.

```
POST localhost:8080/device/deviceForBus

{
  "id": 3,
  "type": "Bi-articulado",
  "motor": "Electric",
  "brakes": "Freno Regenerativo",
  "consessionaire": {
    "id": 2,
    "name": "Concessionaire Edwin"
  }
}

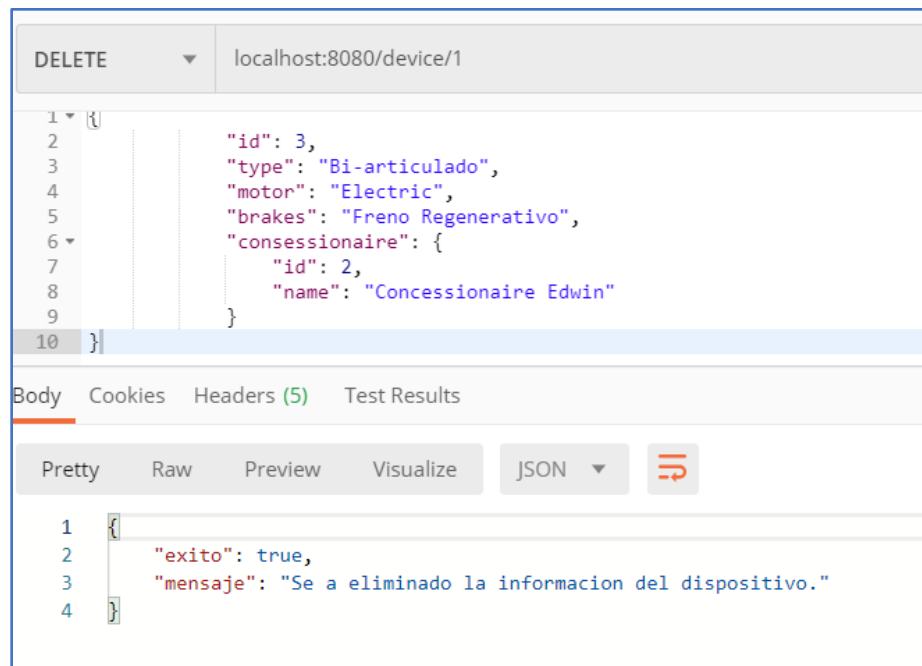
Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

{
  "listDevice": [
    {
      "id": 3,
      "ip": "192.168.0.128",
      "deviceType": { ...
    },
      "bus": {
        "id": 3,
        "type": "Bi-articulado",
        "motor": "Electric",
        "brakes": "Freno Regenerativo",
        "consessionaire": {
          "id": 2,
          "name": "Concessionaire Edwin"
        }
      },
      "status": "Active"
    },
    {
      "id": 6,
      "ip": "192.168.0.55",
      "deviceType": { ...
    },

```


Por ultimo haremos Delete del device 1



The screenshot shows a REST client interface. The top bar indicates a **DELETE** request to `localhost:8080/device/1`. The request body is a JSON object:

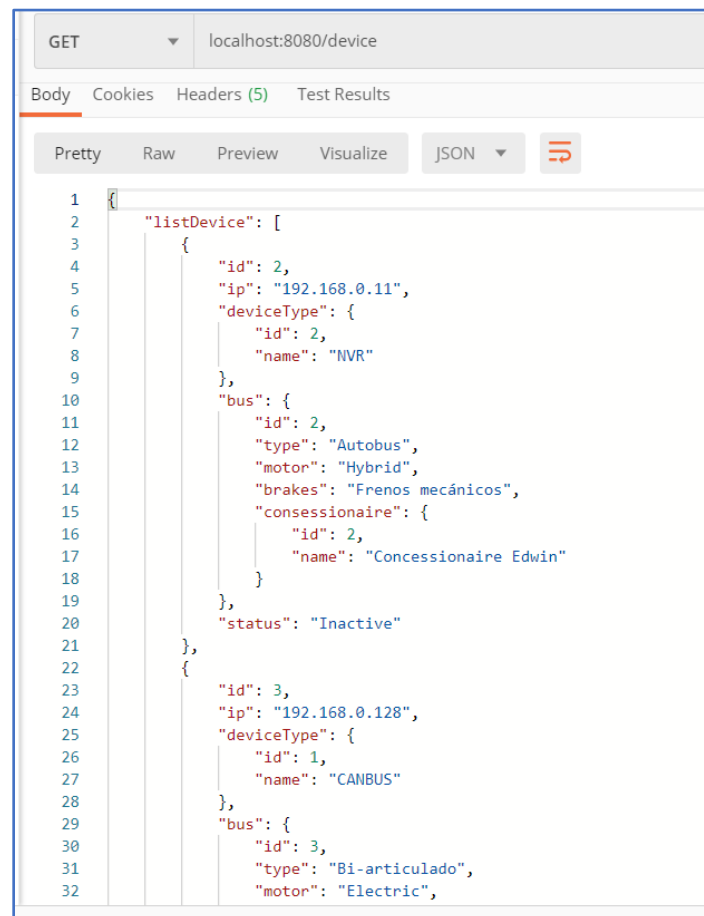
```
{
  "id": 3,
  "type": "Bi-articulado",
  "motor": "Electric",
  "brakes": "Freno Regenerativo",
  "concessionaire": {
    "id": 2,
    "name": "Concessionaire Edwin"
  }
}
```

. Below the request, the **Body** tab is selected, showing the response in JSON format:

```
{
  "exito": true,
  "mensaje": "Se a eliminado la informacion del dispositivo."
}
```

. The response is formatted as JSON, and the status is 200.

Y listamos para comprobar



The screenshot shows a REST client interface. The top bar indicates a **GET** request to `localhost:8080/device`. The **Body** tab is selected, showing the response in JSON format:

```
{
  "listDevice": [
    {
      "id": 2,
      "ip": "192.168.0.11",
      "deviceType": {
        "id": 2,
        "name": "NVR"
      },
      "bus": {
        "id": 2,
        "type": "Autobus",
        "motor": "Hybrid",
        "brakes": "Frenos mecánicos",
        "concessionaire": {
          "id": 2,
          "name": "Concessionaire Edwin"
        }
      },
      "status": "Inactive"
    },
    {
      "id": 3,
      "ip": "192.168.0.128",
      "deviceType": {
        "id": 1,
        "name": "CANBUS"
      },
      "bus": {
        "id": 3,
        "type": "Bi-articulado",
        "motor": "Electric",
        "brakes": "Freno Regenerativo"
      }
    }
  ]
}
```

. The response is formatted as JSON, and the status is 200.