

ENTREGABLE 4

Buscar los servicios pendientes y completados adjuntada la información del usuario que pidió el servicio

GET http://localhost:3000/api/v1/repairs

Params Authorization Headers (8) Body Pre-request Script

Type Bearer T... Heads up! These parameters ho collaborative environment, we re

Body Cookies Headers (8) Test Results

Pretty Raw Preview Visualize JSON

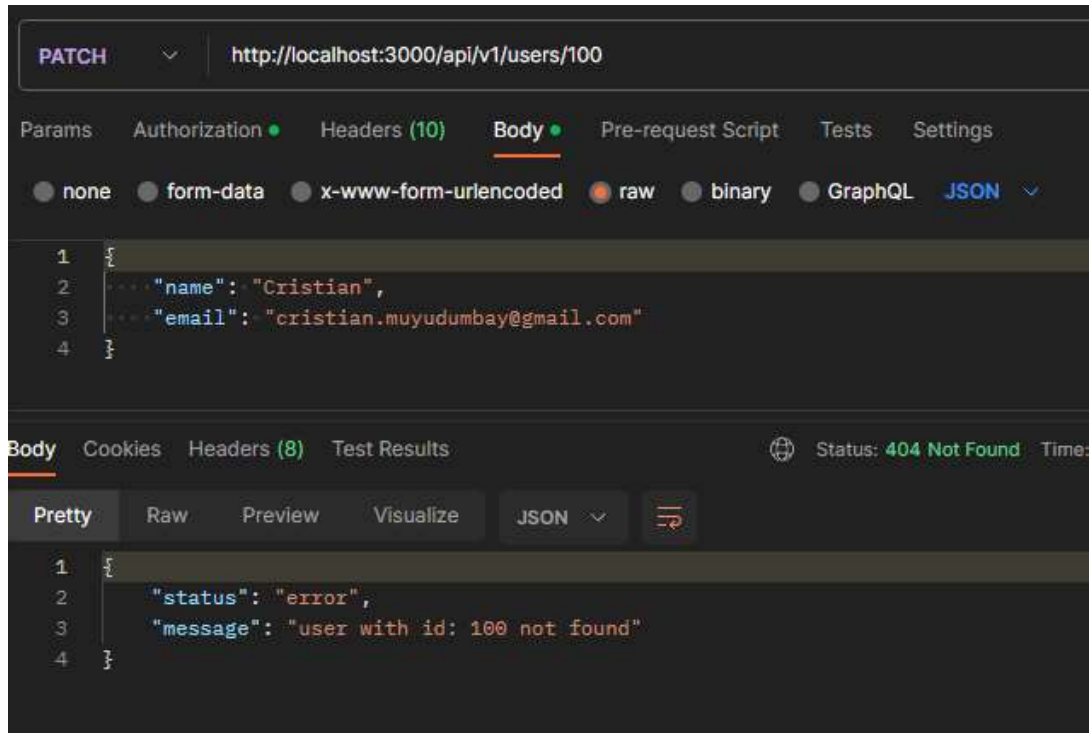
```
1  [
2    {
3      "id": 10,
4      "date": "2023-11-19T00:00:00.000Z",
5      "motorsNumber": 346579,
6      "description": "auto terminado con aire",
7      "status": "completed",
8      "userId": 7,
9      "user": {
10       "name": "Paul",
11       "email": "paul@gmail.com",
12       "role": "cliente"
13     }
14   },
15 ]
```

```
},
{
  "id": 6,
  "date": "2023-11-19T00:00:00.000Z",
  "motorsNumber": 34652,
  "description": "no tiene aire",
  "status": "pending",
  "userId": 1,
  "user": {
    "name": "Karolina",
    "email": "karolinaguaman@gmail.com",
    "role": "empleado"
  }
},
{
```

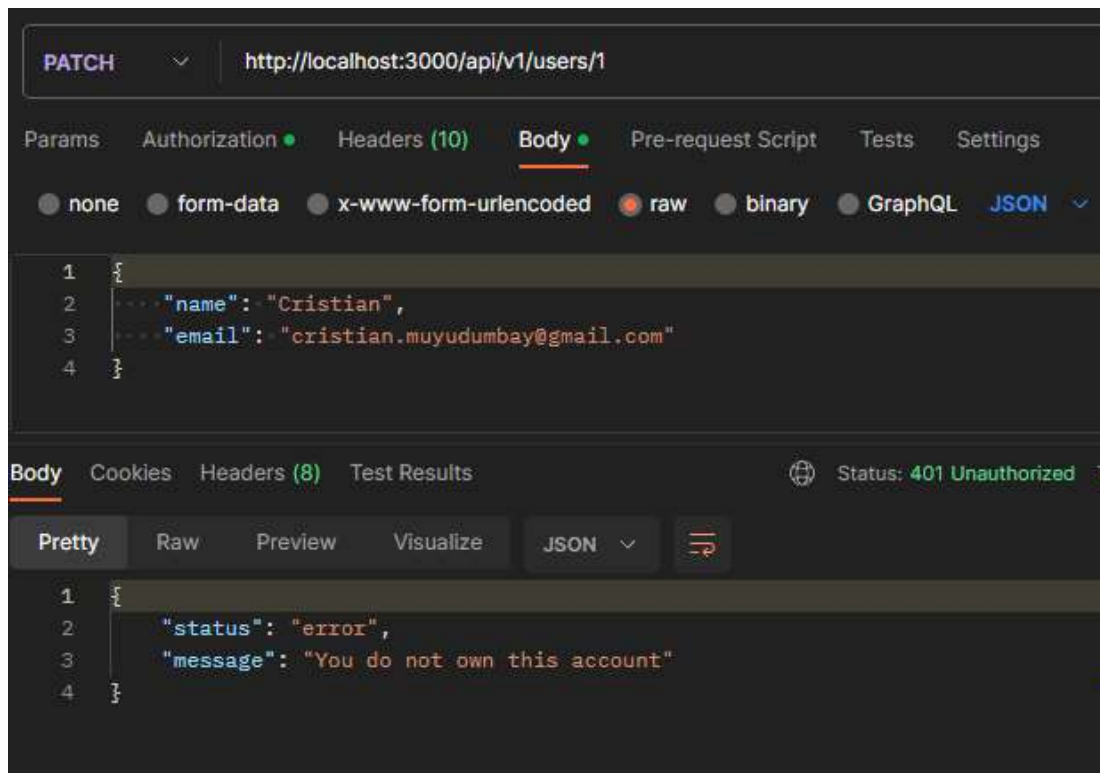
Endpoints protegidos /api/v1/users solo para que el usuario en sesión pueda hacer el patch y delete de su respectiva cuenta

PATCH

Usuario que no existe



Usuario que existe, pero no es el usuario que está en sesión



Hacer el patch del usuario en sesión

The screenshot shows a REST client interface with a PATCH request to `http://localhost:3000/api/v1/users/5`. The request body is a JSON object with the following fields:

```
1 {
2   "name": "Cristian",
3   "email": "cristian.muyudumbay@gmail.com"
4 }
```

The response is a 200 OK status. The response body is a JSON object with the following fields:

```
1 {
2   "id": 5,
3   "name": "Cristian",
4   "email": "cristian.muyudumbay@gmail.com",
5   "password": "$2b$11$Grmm.ZNxeq7wT0gAb74/heE/SSzuUU1/stFYI28jUIf8I3DHarzQi",
6   "role": "empleado",
7   "status": "available",
8   "createdAt": "2023-12-17T02:23:10.230Z",
9   "updatedAt": "2023-12-18T02:27:54.683Z"
10 }
```

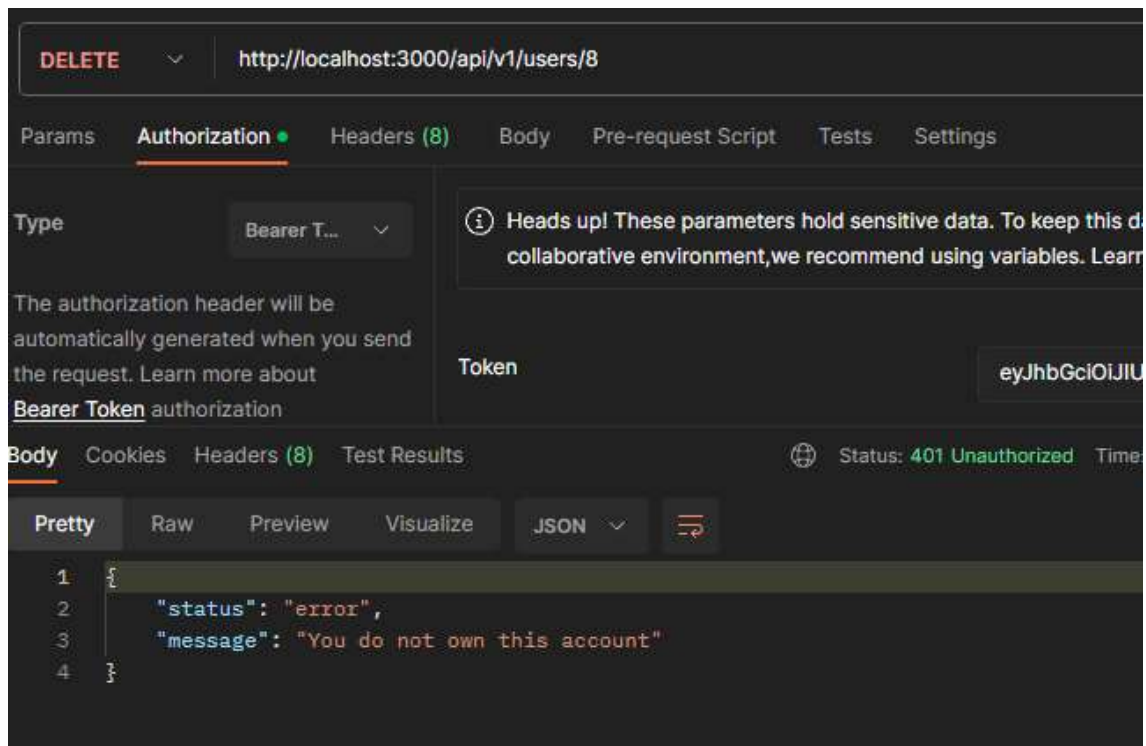
DELETE

Usuario que no existe

The screenshot shows a REST client interface with a DELETE request to `http://localhost:3000/api/v1/users/10`. The request is authorized with a Bearer Token. The response is a 404 Not Found status. The response body is a JSON object with the following fields:

```
1 {
2   "status": "error",
3   "message": "user with id: 10 not found"
4 }
```

Usuario que existe, pero no es el que esta logeado



Hacer el delete de su propia cuenta

