

USERS/REGISTER

POST localhost:3001/api/v1/users/register

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

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

```
1 {
2   "name": "luis",
3   "email": "luis@gmail.com",
4   "password": "pas1118794"
5 }
```

Body Cookies Headers (7) Test Results 🌐 Status: 200 OK

Pretty Raw Preview Visualize **JSON** ▾

```
1 {
2   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MTcsIm1hdCI6MTcwMjI3NjgwOSwiZXhwIjoxNzAyMjg0MDA5fQ.8zZWwJ4mts",
3   "status": "success",
4   "user": {
5     "role": "client",
6     "status": "available",
7     "id": 17,
8     "name": "luis",
9     "email": "luis@gmail.com",
10    "password": "$2b$12$XI/xAHgfJ4co8Q16Db5NV.o6t9bdYk7w0964lYSuu2dNFgDAM5lem",
11    "updatedAt": "2023-12-11T06:40:07.660Z",
12    "createdAt": "2023-12-11T06:40:07.660Z"
13  }
14 }
```

USERS/LOGIN

POST localhost:3001/api/v1/users/login

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings

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

```
1  {
2    "email": "mario@gmail.com",
3    "password": "pas99998574"
4  }
```

Body Cookies Headers (7) Test Results 🌐 Status: 200 OK

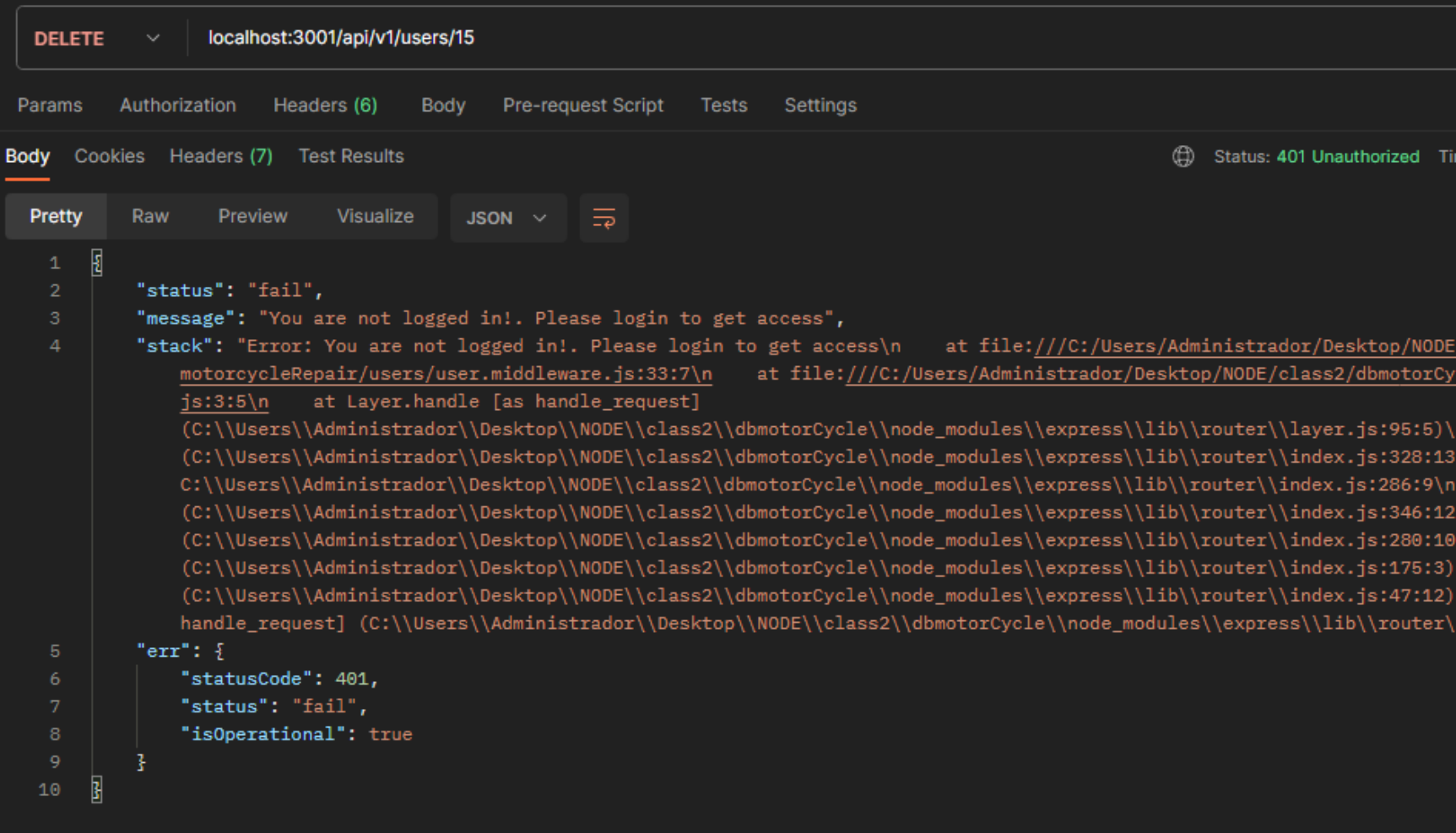
Pretty Raw Preview Visualize **JSON** ▼ 🔗

```
1  {
2    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MTgsIm1hdCI6MTcwMjI3OTU0OiwiaWF0IjoxNzAyMjg2MzQ4fQ.zMicPZ0Q0nR6",
3    "user": {
4      "id": 18,
5      "name": "mario",
6      "email": "mario@gmail.com",
7      "role": "employee"
8    }
9  }
```

USERS/(DATABASE)

id	name	email	password	role	status
2	Raul	Raul@gmail.com	7894	client	available
3	Maria Clara Restrepo	mariaclara@gmail.com	9632	client	available
4	johana	johana@gmail.com	7531	client	disabled
1	tomas	juandavid@gmail.com	1234	employee	available
5	julio	julio@gmail.com	3030	client	available
9	erika	erika@gmail.com	3654	client	available
10	Alex hernandez	alexgmail.com	2035	client	disabled
11	fabio	fabio@gmail.com	pass14789	employee	available
15	1	fabio1@gmail.com	pass14789	employee	available
17	luis	luis@gmail.com	\$2b\$12\$Xl/xAHgfJ4co8Q16Db5NV.o6t9bdYk7wO964lYSuu2dNFgDA M5lem	client	available
18	mario	mario@gmail.com	\$2b\$12\$GeEYFa3gxs0mUe86YWEkeBZvBnW87rBcRlvGr4TzgGbsYAe X01Xu	employee	available

USUARIO NO LOGEADO



The screenshot shows a REST client interface with a DELETE request to `localhost:3001/api/v1/users/15`. The response is a 401 Unauthorized status. The response body is displayed in JSON format, showing an error message and a stack trace. The status bar indicates "Status: 401 Unauthorized".

```
DELETE localhost:3001/api/v1/users/15

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Body Cookies Headers (7) Test Results Status: 401 Unauthorized

Pretty Raw Preview Visualize JSON ↗

1 {
2   "status": "fail",
3   "message": "You are not logged in!. Please login to get access",
4   "stack": "Error: You are not logged in!. Please login to get access\n    at file:///C:/Users/Administrador/Desktop/NODE\n      motorcycleRepair/users/user.middleware.js:33:7\n      at file:///C:/Users/Administrador/Desktop/NODE/class2/dbmotorCy\n        js:3:5\n      at Layer.handle [as handle_request]\n        (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\layer.js:95:5)\n        (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:328:13)\n        C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:286:9\n        (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:346:12)\n        (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:280:10)\n        (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:175:3)\n        (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:47:12)\n        handle_request] (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:47:12)"
5   "err": {
6     "statusCode": 401,
7     "status": "fail",
8     "isOperational": true
9   }
10 }
```

NO DEJA ELMINAR SI EL USUARIO NO ES EL DUEÑO DE LA CUENTA

DELETE localhost:3001/api/v1/users/15

Params **Authorization** Headers (7) Body Pre-request Script Tests Settings

Type Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about [authorization](#)

Heads up! These parameters hold sensitive data. To keep this data secure while working in a co... variables. Learn more about [variables](#)

Token eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...

Body Cookies Headers (7) Test Results Status: 401 Unauthorized

Pretty Raw Preview Visualize JSON ≡

```
1 {
2   "status": "fail",
3   "message": "You do not own this account",
4   "stack": "Error: You do not own this account\n    at protectAccountOwner (file:///C:/Users/Administrador/Desktop/NODE/class2/dbmotorCycle/src/motorcycleRepair/users/user.middleware.js:74:17)\n    at Layer.handle [as handle_request] (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\layer.js:95:5)\n    at C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\route.js:14:12\n    at Administrador/Desktop/NODE/class2/dbmotorCycle/src/motorcycleRepair/users/user.middleware.js:16:3\n    at process (node:internal/process/task_queues:95:5)",
5   "err": {
6     "statusCode": 401,
7     "status": "fail",
8     "isOperational": true
9   }
10 }
```

NO DEJA LOGEAR, SI EL PASSWORD NO ESTA ENCRYPTADO

The screenshot shows a REST client interface with a POST request to `localhost:3001/api/v1/users/login`. The request body is a JSON object with a plain text password. The response is a 401 Unauthorized status with a detailed error message.

Request:

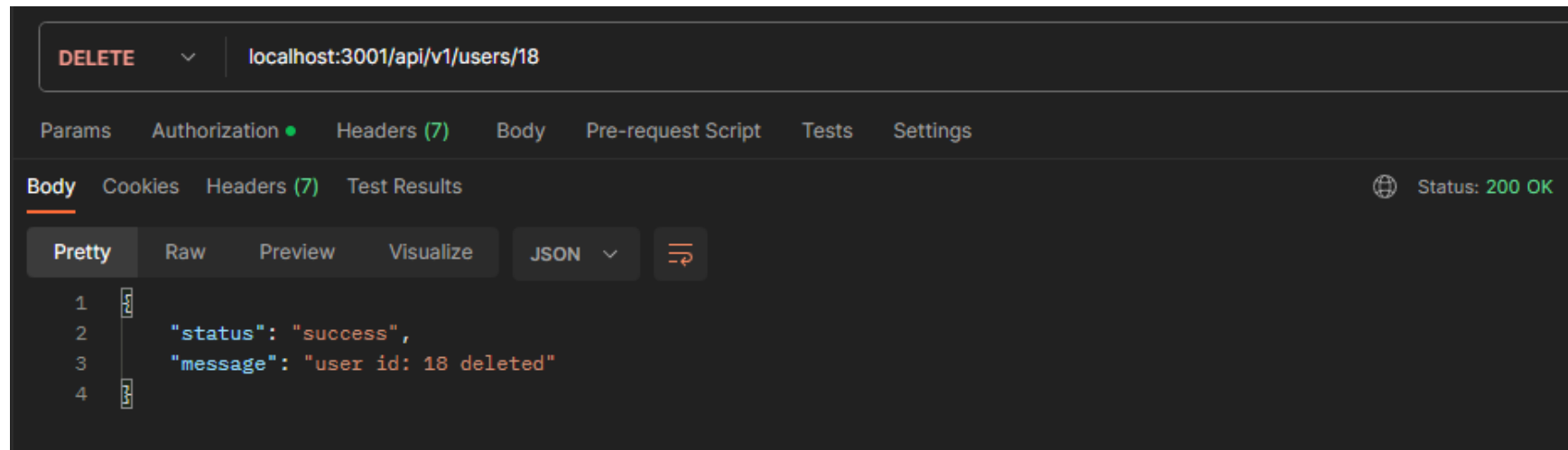
```
POST localhost:3001/api/v1/users/login

{
  "password": "pass14789",
  "email": "fabio1@gmail.com"
}
```

Response: Status: 401 Unauthorized

```
{
  "status": "fail",
  "message": "Incorrect email or password",
  "stack": "Error: Incorrect email or password\n    at file:///C:/Users/Administrador/Desktop/NODE/class2/dbmotorCycle/controller.js:79:17",
  "err": {
    "statusCode": 401,
    "status": "fail",
    "isOperational": true
  }
}
```

SOLO USUARIO LOGEADO SE PUEDO ELIMINAR A SI MISMO



USUARIO (ROLE CLIENTE) LOGEADO

The screenshot displays a REST client interface for a POST request to `localhost:3001/api/v1/users/login`. The request body is a JSON object with email and password. The response body is a JSON object containing a token and user details.

Request:

```
POST localhost:3001/api/v1/users/login

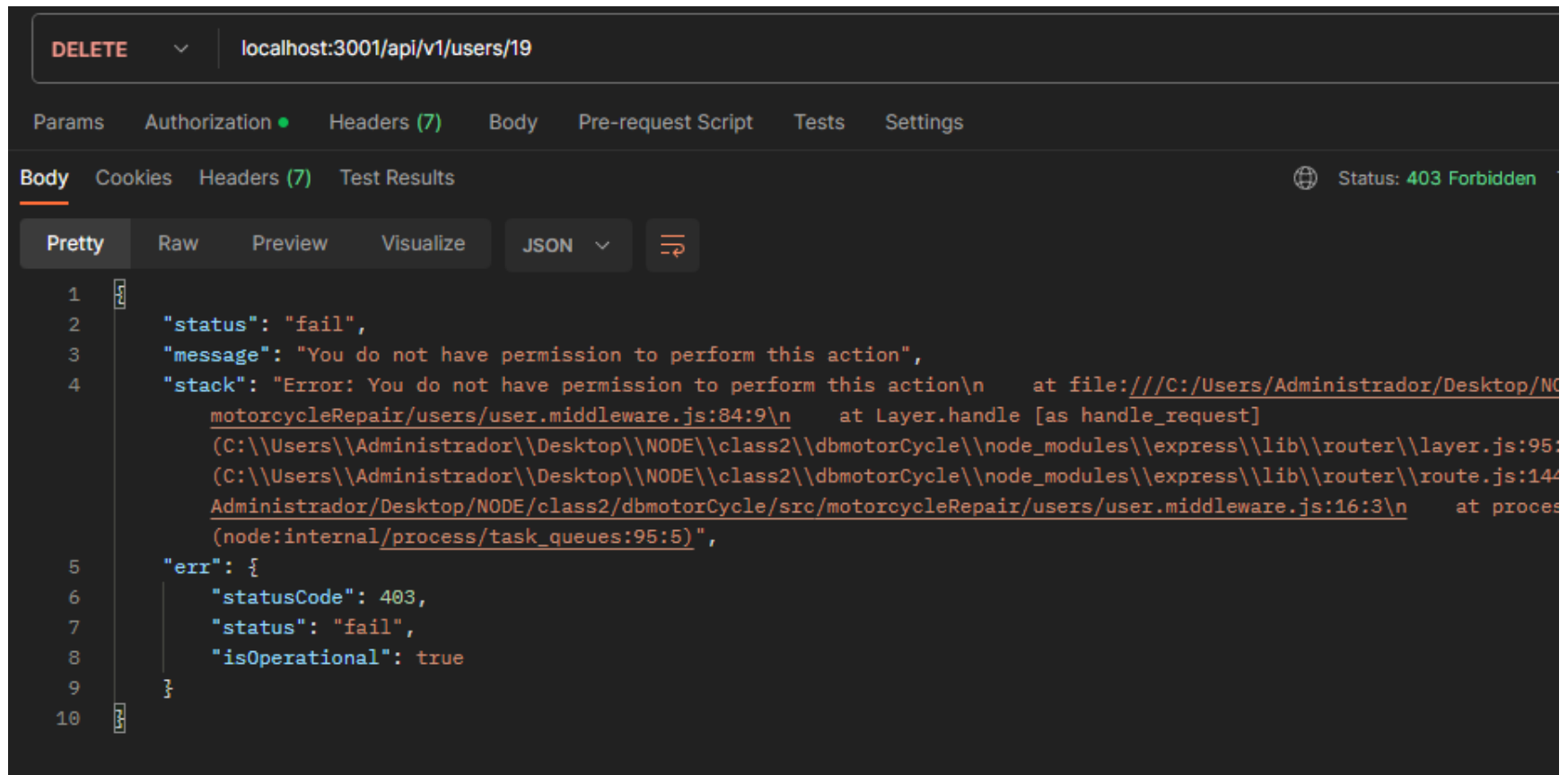
{
  "email": "luz@gmail.com",
  "password": "pass1111"
}
```

Response:

```
{
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MTksIm1hdCI6MTcwMjI4MTI4NiwiZXhwIjoxNzAyMjg4NDg2fQ.cl6ArtsCR9",
  "user": {
    "id": 19,
    "name": "luz maria",
    "email": "luz@gmail.com",
    "role": "client"
  }
}
```

Status: 200 OK

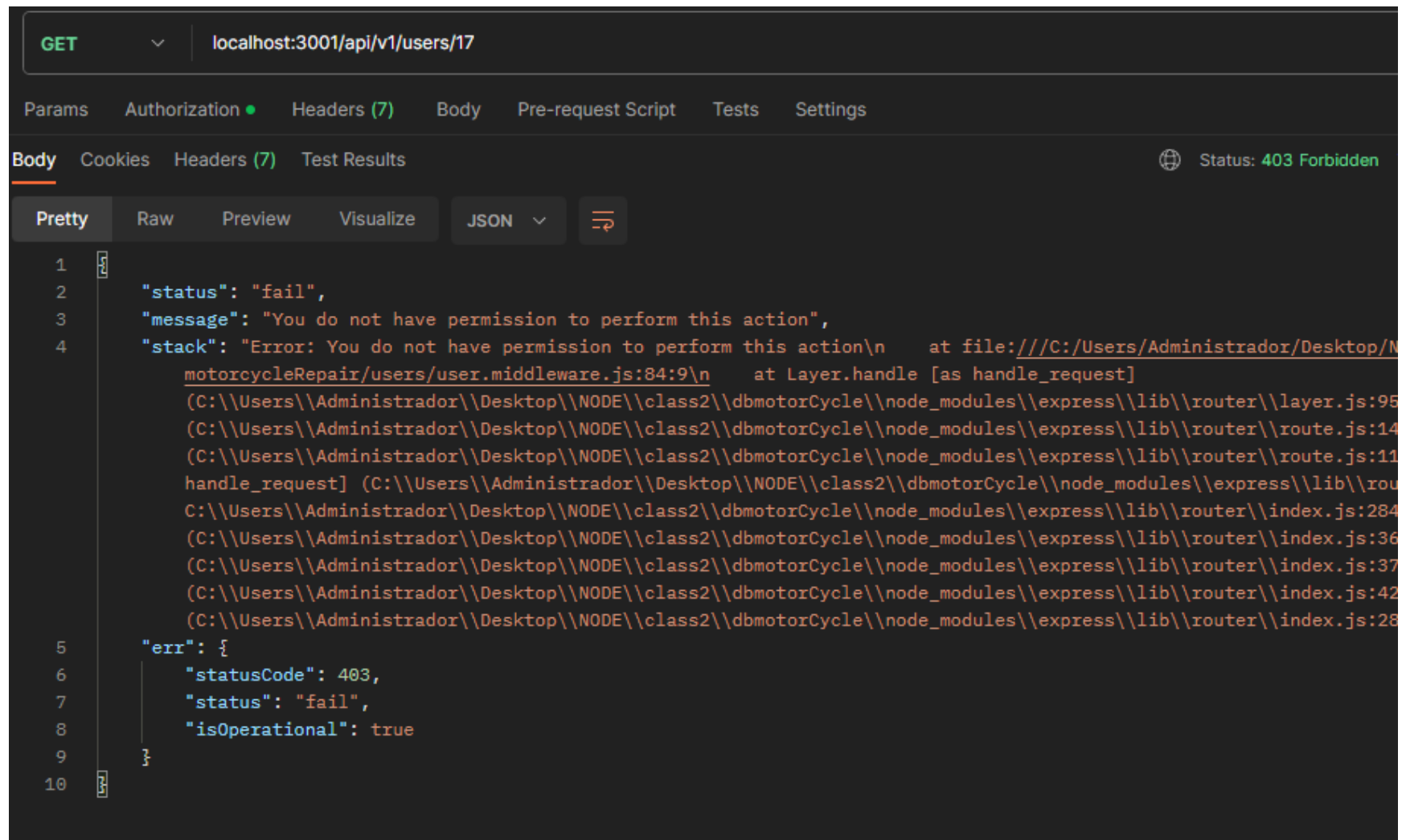
SOLO (ROLE EMPLEADO PUEDE ELIMINAR UNA CUENTA



The screenshot shows a REST client interface with a DELETE request to `localhost:3001/api/v1/users/19`. The response is a 403 Forbidden status. The JSON body of the response is as follows:

```
1  {
2    "status": "fail",
3    "message": "You do not have permission to perform this action",
4    "stack": "Error: You do not have permission to perform this action\n    at file:///C:/Users/Administrador/Desktop/NODE/class2/dbmotorCycle/src/motorcycleRepair/users/user.middleware.js:84:9\n    at Layer.handle [as handle_request] (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\layer.js:95:17)\n    at C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\route.js:144:13\n    at Administrador/Desktop/NODE/class2/dbmotorCycle/src/motorcycleRepair/users/user.middleware.js:16:3\n    at process (node:internal/process/task_queues:95:5)",
5    "err": {
6      "statusCode": 403,
7      "status": "fail",
8      "isOperational": true
9    }
10 }
```

SOLO (ROLE EMPLEADO PUEDE HACER UNA BUSQUEDA



GET localhost:3001/api/v1/users/17

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Body Cookies Headers (7) Test Results Status: 403 Forbidden

Pretty Raw Preview Visualize JSON

```
1  {
2    "status": "fail",
3    "message": "You do not have permission to perform this action",
4    "stack": "Error: You do not have permission to perform this action\n    at file:///C:/Users/Administrador/Desktop/motorcycleRepair/users/user.middleware.js:84:9\n    at Layer.handle [as handle_request] (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\layer.js:95:17)\n    at C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\route.js:14:7\n    at C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\route.js:11:12\n    at handle_request (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:284:15)\n    at C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:36:18\n    at C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:37:12\n    at C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:42:12\n    at C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:28:12"
5    "err": {
6      "statusCode": 403,
7      "status": "fail",
8      "isOperational": true
9    }
10 }
```

SOLO (STATUS "AVAILABLE" Y EMPLEADO PUEDE LOGEARSE

The screenshot shows a REST client interface with a POST request to `localhost:3001/api/v1/users/login`. The request body is a JSON object with email and password. The response is a 404 Not Found status with a detailed error message in the body.

Request:

```
POST localhost:3001/api/v1/users/login
```

Body:

```
{  "email": "mario@gmail.com",  "password": "pas99998574"}
```

Response:

Status: 404 Not Found

Body:

```
{  "status": "fail",  "message": "This account does not exist",  "stack": "Error: This account does not exist\n    at file:///C:/Users/Administrador/Desktop/NODE/class2/dbmotorCyclecontroller.js:69:17\n    at process.processTicksAndRejections (node:internal/process/task_queues:95:5)",  "err": {    "statusCode": 404,    "status": "fail",    "isOperational": true  }}
```

EMPLEADO LOGEADO PUDO HACER UNA BUSQUEDA

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** localhost:3001/api/v1/users/17
- Authorization:** Bearer Token. The token value is: eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...
- Status:** 200 OK
- Response Body (JSON):**

```
1 {
2   "status": "success",
3   "user": {
4     "id": 17,
5     "name": "luis",
6     "email": "luis@gmail.com",
7     "password": "$2b$12$XI/xAHgfJ4co8Q16Db5NV.o6t9bdYk7w0964lYSuu2dNFgDAM5lem",
8     "role": "client",
9     "status": "available",
10    "createdAt": "2023-12-11T06:40:07.660Z",
11    "updatedAt": "2023-12-11T06:40:07.660Z"
12  }
13 }
```

SOLO (DUEÑO DE LA CUENTA PUEDE ELIMINAR SU CUENTA

DELETE localhost:3001/api/v1/users/17

Params Authorization Headers (7) **Body** Pre-request Script Tests Settings

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

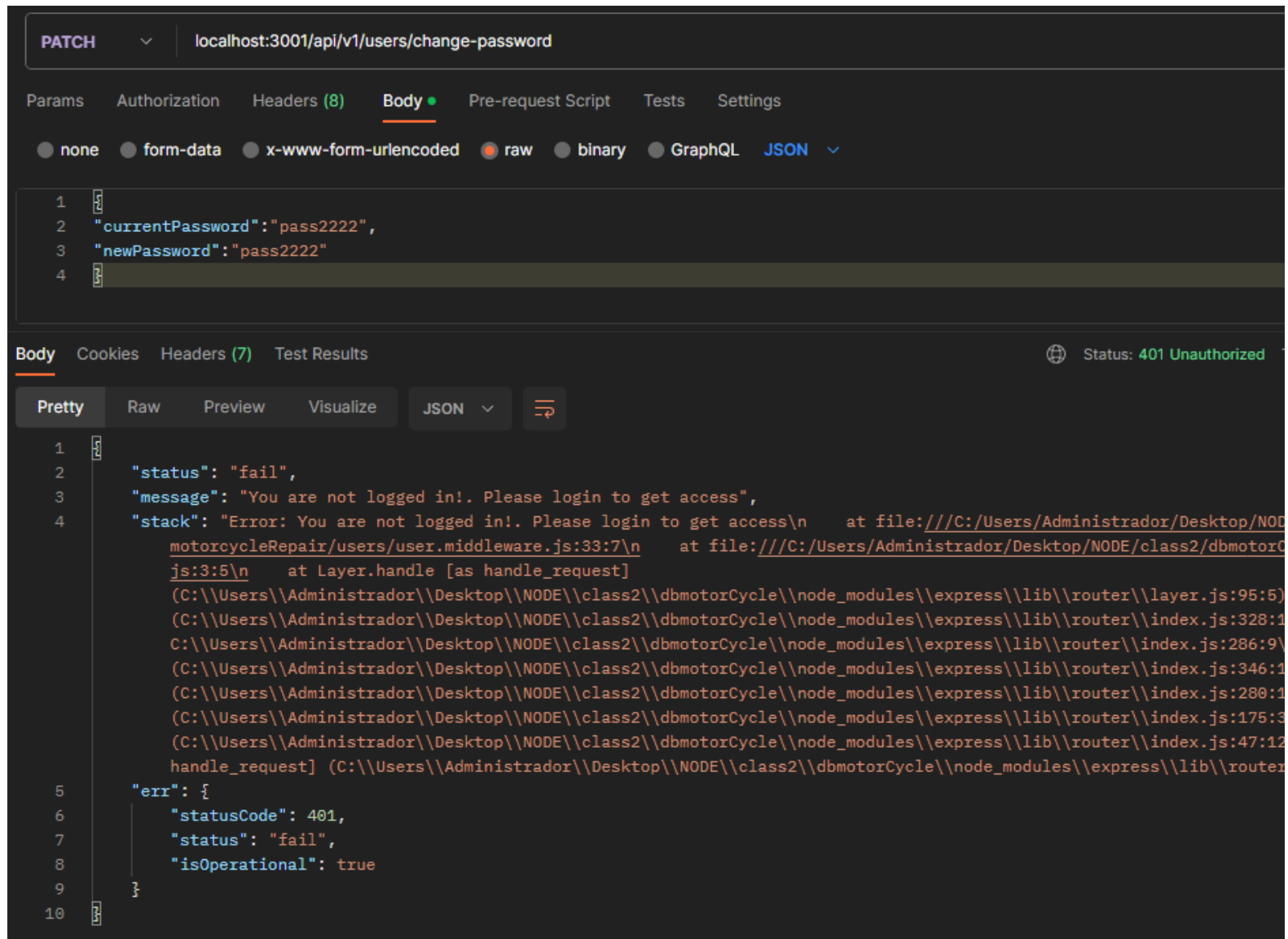
1

Body Cookies Headers (7) Test Results Status: 401 Unauthorized

Pretty Raw Preview Visualize JSON

```
1 {
2   "status": "fail",
3   "message": "You do not own this account",
4   "stack": "Error: You do not own this account\n    at protectAccountOwner (file:///C:/Users/Administrador/Desktop/NODE/motorcycleRepair/users/user.middleware.js:74:17)\n    at Layer.handle [as handle_request] (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\layer.js:95:17)\n    at C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\route.js:144:13\n    at Administrador/Desktop/NODE/class2/dbmotorCycle/src/motorcycleRepair/users/user.middleware.js:87:5\n    at Layer.handle [as handle_request] (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\layer.js:95:17)\n    at C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\route.js:144:13\n    at Administrador/Desktop/NODE/class2/dbmotorCycle/src/motorcycleRepair/users/user.middleware.js:16:3\n    at process (node:internal/process/task_queues:95:5)",
5   "err": {
6     "statusCode": 401,
7     "status": "fail",
8     "isOperational": true
9   }
10 }
```

NO SE PUEDE ACTUALIZAR SIN ESTAR LOGGEADO



NO SE PUEDE ACTUALIZAREL MISMO PASSWORD

PATCH localhost:3001/api/v1/users/change-password

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings

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

```
1 {}
2 "currentPassword": "pass2222",
3 "newPassword": "pass2222"
4 {}
```

Body Cookies Headers (7) Test Results 🌐 Status: 400 Bad Request

Pretty Raw Preview Visualize **JSON** ▼ ≡

```
1 {}
2 "status": "fail",
3 "message": "The password cannot be equal",
4 "stack": "Error: The password cannot be equal\n    at file:///C:/Users/Administrador/Desktop/NODE/class2/dbmotorCycle/controller.js:187:17\n    at file:///C:/Users/Administrador/Desktop/NODE/class2/dbmotorCycle/src/common/errors/[as handle_request] (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\(C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\route.js:14(C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\route.js:11handle_request] (C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\rouC:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:284(C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:34(C:\\Users\\Administrador\\Desktop\\NODE\\class2\\dbmotorCycle\\node_modules\\express\\lib\\router\\index.js:28Administrador/Desktop/NODE/class2/dbmotorCycle/src/motorcycleRepair/users/user.middleware.js:67:3",
5 "err": {
6   "statusCode": 400,
7   "status": "fail",
8   "isOperational": true
9 }
10 {}
```

PASSWORD ACTUALIZADO

The screenshot displays a REST client interface with a PATCH request to `localhost:3001/api/v1/users/change-password`. The request body is a JSON object with `currentPassword` and `newPassword` fields. The response status is 200 OK, and the response body contains a success message.

Request:

- Method: PATCH
- URL: `localhost:3001/api/v1/users/change-password`
- Body (JSON):

```
{  "currentPassword": "pass2222",  "newPassword": "pass3333"}
```

Response:

- Status: 200 OK
- Body (JSON):

```
{  "message": "The user password was updated successfully!"}
```


VALIDACIÓN DE USUARIO, NO EXISTE EN BASE DE DATOS

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** localhost:3001/api/v1/users/23
- Authorization:** Bearer Token
- Status:** 404 Not Found
- Response Body (JSON):**

```
1 {
2   "status": "fail",
3   "message": "user with id: 23 not found",
4   "stack": "Error: user with id: 23 not found\n    at file:///C:/Users/Administrador/Desktop/NODE/class2/dbmotorCycle/
5     middleware.js:13:17\n    at process.processTicksAndRejections (node:internal/process/task_queues:95:5)",
6   "err": {
7     "statusCode": 404,
8     "status": "fail",
9     "isOperational": true
10  }
```

VALIDACIÓN DE USUARIO, EXISTE EN BASE DE DATOS

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** localhost:3001/api/v1/users/17
- Authorization:** Bearer Token
- Status:** 200 OK
- Response Body (JSON):**

```
{  "status": "success",  "user": {    "id": 17,    "name": "luis",    "email": "luis@gmail.com",    "password": "$2b$12$XI/xAHgfJ4co8Q16Db5NV.o6t9bdYk7w0964lYSuu2dNFgDAM51em",    "role": "client",    "status": "available",    "createdAt": "2023-12-11T06:40:07.660Z",    "updatedAt": "2023-12-11T06:40:07.660Z"  }}
```

MENSAJE DE ERROR AL LOGEARSE POR CAMBIO DE PASSWORD

The screenshot displays a REST client interface for a POST request to `localhost:3001/api/v1/users/login`. The request body is a JSON object with email and password fields. The response status is 401 Unauthorized, and the body contains a detailed error message indicating incorrect credentials.

Request:

```
POST localhost:3001/api/v1/users/login
```

Body:

```
{
  "email": "humberto@gmail.com",
  "password": "pass2222"
}
```

Response:

Status: 401 Unauthorized

Body:

```
{
  "status": "fail",
  "message": "Incorrect email or password",
  "stack": "Error: Incorrect email or password\n    at file:///C:/Users/Administrador/Desktop/NODE/class2/dbmotorCycle/controller.js:79:17",
  "err": {
    "statusCode": 401,
    "status": "fail",
    "isOperational": true
  }
}
```

LOGEADO CON EL NUEVO PASSWORD

The screenshot shows a REST client interface with a POST request to `localhost:3001/api/v1/users/login`. The request body is a JSON object with email and password. The response is a 200 OK status with a JSON body containing a token and user details.

Request:

```
POST localhost:3001/api/v1/users/login
```

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings

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

```
1 {
2   "email": "humberto@gmail.com",
3   "password": "pass3333"
4 }
```

Response:

Body Cookies Headers (7) Test Results 🌐 Status: 200 OK

Pretty Raw Preview Visualize **JSON** ▾

```
1 {
2   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MjAsIm5hbGlhdCI6MTcwMjMyMTAwMCwiZXhwIjozNzAyMzI4MjAwfQ.GnXZn6Iz0v
3   "user": {
4     "id": 20,
5     "name": "humberto",
6     "email": "humberto@gmail.com",
7     "role": "employee"
8   }
9 }
```

DEVELOPMENT | PostgreSQL 15.4 : TLSv1.3...

ItemQueryHistory

Search for item...

Functions

Tables

errors

pg_stat_state

pg_stat_state

repairs

users

id	name	email	password	role	status
17	luis	luis@gmail.com	\$2b\$12\$XI/xAHgfJ4co8...	client	available
18	mario	mario@gmail.com	\$2b\$12\$GeEYFa3gxsn0...	employee	disabled
19	luz m...	luz@gmail.com	\$2b\$12\$t7FduFIMRK6u...	client	available
20	humb...	humberto@gmail.com	\$2b\$12\$0L580rPLdJXCS...	employee	available

id	name	email	password	role	status
17	luis	luis@gmail.com	\$2b\$12\$XI/xAHgfJ4co8Q16Db5NV.o6t9bdYk7wO964lYSuu2dNFgDAM5lem	client	available
18	mario	mario@gmail.com	\$2b\$12\$GeEYFa3gxsn0mUe86YWEkeBZvBnW87rBcRlvGr4TzgGbsYAeX01Xu	employee	disabled
19	luz maria	luz@gmail.com	\$2b\$12\$t7FduFIMRK6uDpwW0Bmvn.Fcs8auulKdN/oBOoRy3Qi8qPxRQ2BdC	client	available
20	humberto	humberto@gmail.com	\$2b\$12\$0L580rPLdJXCSOUUuFsqDen0Qw0Pf/RMguydWiHt5br3b0U6OGQP.	employee	available

REPAIRS DB

ItemsQueryHistory		users				X	
Search for item...		id	date	description	userId	status	
Functions		1	2023-11-28 00:00:00+00	afinacion motor AKT 2023	1212	canceled	
Tables		2	2023-11-28 00:00:00+00	cambio tanque yamaha 2023	8884	completed	
errors		3	2023-11-28 00:00:00+00	pintura chasis honda 2023	1258	completed	
pg_stat_state		4	2023-11-28 00:00:00+00	cambio caja velocidades auteco plus...	3030	completed	
pg_stat_state		5	2023-11-28 00:00:00+00	platoneria y intura chasis zuzuky 2023	1114	canceled	
repairs		6	2023-11-28 00:00:00+00	sistema electrico zuzuky 2023	1010	completed	
users		7	2023-11-30 00:00:00+00	sistema electrico AKT 2023	4546	canceled	
		8	2023-12-01 05:00:00+00	cambio motor bajac 2023	2587	canceled	

CAMBIO DB REPAIRS POR FORCE:TRUE

id	date	motorNumber	description	user_Id	status
1	2023-12-10 00:00:00 +0000	65656565dffd	motor cambio	1	pending
2	2023-12-10 00:00:00 +0000	4449654ertw	pintura cambio	2	pending

POST REPARACIONES

POST localhost:3001/api/v1/repairs

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings

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

```
1 {
2   "date": "2023-12-10",
3   "motorNumber": "4449654ertw",
4   "description": "pintura cambio",
5   "userId": 2
6 }
```

Body Cookies Headers (7) Test Results 🌐 Status: 200 OK

Pretty Raw Preview Visualize **JSON** ▼ ↔

```
1 {
2   "status": "success",
3   "repair": {
4     "status": "pending",
5     "id": 2,
6     "date": "2023-12-10T00:00:00.000Z",
7     "motorNumber": "4449654ertw",
8     "description": "pintura cambio",
9     "userId": 2,
10    "updatedAt": "2023-12-12T04:27:05.684Z",
11    "createdAt": "2023-12-12T04:27:05.684Z"
12  }
13 }
```

PATCH REPARACIONES

The screenshot displays a REST client interface with the following details:

- Method:** PATCH
- URL:** localhost:3001/api/v1/repairs/1
- Body Type:** JSON
- Request Body:**

```
1 {  
2  
3   "status": "completed"  
4  
5 }
```
- Response Status:** 200 OK
- Response Body (Pretty):**

```
1 {  
2   "status": "success",  
3   "message": "updated id: 1 ",  
4   "repairUpdated": {  
5     "id": 1,  
6     "date": "2023-12-10T00:00:00.000Z",  
7     "motorNumber": "65656565dffd",  
8     "description": "motor cambio",  
9     "userId": 1,  
10    "status": "completed",  
11    "createdAt": "2023-12-12T04:09:16.205Z",  
12    "updatedAt": "2023-12-12T04:43:54.553Z"  
13  }  
14 }
```


GET REPARACIONES PENDING

GET localhost:3001/api/v1/repairs

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Query Params

Key	Value	Description
-----	-------	-------------

Body Cookies Headers (7) Test Results Status: 200 OK

Pretty Raw Preview Visualize JSON

```
1  {
2    "status": "success",
3    "repairs": [
4      {
5        "id": 2,
6        "date": "2023-12-10T00:00:00.000Z",
7        "motorNumber": "4449654extw",
8        "description": "pintura cambio",
9        "userId": 2,
10       "status": "pending",
11       "createdAt": "2023-12-12T04:27:05.684Z",
12       "updatedAt": "2023-12-12T04:27:05.684Z"
13     },
14     {
15       "id": 3,
16       "date": "2023-12-11T05:00:00.000Z",
17       "motorNumber": "7996100AKT",
18       "description": "cambio luces akt5002023",
19       "userId": 3,
20       "status": "pending",
21       "createdAt": "2023-12-12T04:53:01.214Z",
22       "updatedAt": "2023-12-12T04:53:01.214Z"
23     }
24   ]
25 }
```

GET DE UNA REPARACIÓN POR ID (SOLO ESTADO PENDING)

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** localhost:3001/api/v1/repairs/2
- Params:** Authorization, Headers (6), Body, Pre-request Script, Tests, Settings
- Query Params:** Table with columns Key, Value, and Description.
- Body:** Cookies, Headers (7), Test Results
- Status:** 200 OK
- Response Format:** Pretty, Raw, Preview, Visualize, JSON
- Response Body (JSON):**

```
1  {
2    "status": "success",
3    "repairs": {
4      "id": 2,
5      "date": "2023-12-10T00:00:00.000Z",
6      "motorNumber": "4449654ertw",
7      "description": "pintura cambio",
8      "userId": 2,
9      "status": "pending",
10     "createdAt": "2023-12-12T04:27:05.684Z",
11     "updatedAt": "2023-12-12T04:27:05.684Z"
12   }
13 }
```

DATABAS

id	date	motorNumber	description	user_Id	status
2	2023-12-10 00:00:00 +0000	4449654ertw	pintura cambio	2	pending
1	2023-12-10 00:00:00 +0000	65656565dff	motor cambio	1	completed
3	2023-12-11 05:00:00 +0000	7996100AKT	cambio luces akt5002023	3	pending
4	2023-12-11 05:00:00 +0000	7996100ymh	reparar chasis yamaha	4	canceled

DELETE REPAIR

DELETE localhost:3001/api/v1/repairs/4

Params Authorization Headers (6) **Body** Pre-request Script Tests Settings

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

This request does not have a body

Body Cookies Headers (7) Test Results Status: 200 OK

Pretty Raw Preview Visualize JSON ≡

```
1  {
2    "status": "success",
3    "message": "repair id: 4 canceled",
4    "deleteService": {
5      "id": 4,
6      "date": "2023-12-11T05:00:00.000Z",
7      "motorNumber": "7996100ymh",
8      "description": "reparar chasis yamaha",
9      "userId": 4,
10     "status": "canceled",
11     "createdAt": "2023-12-12T05:03:42.331Z",
12     "updatedAt": "2023-12-12T05:59:23.590Z"
13   }
14 }
```