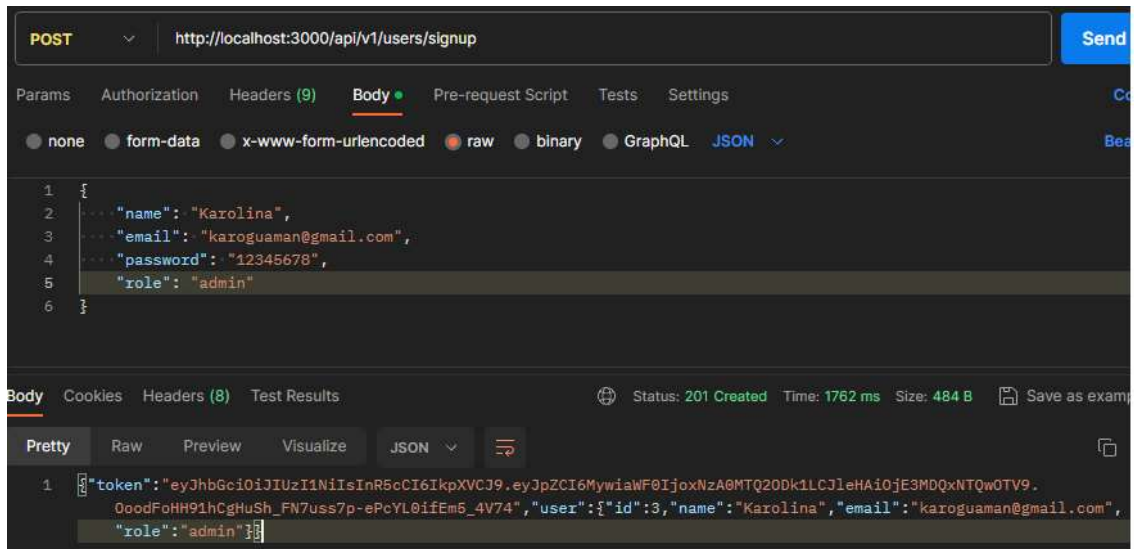


ENTREGABLE FINAL

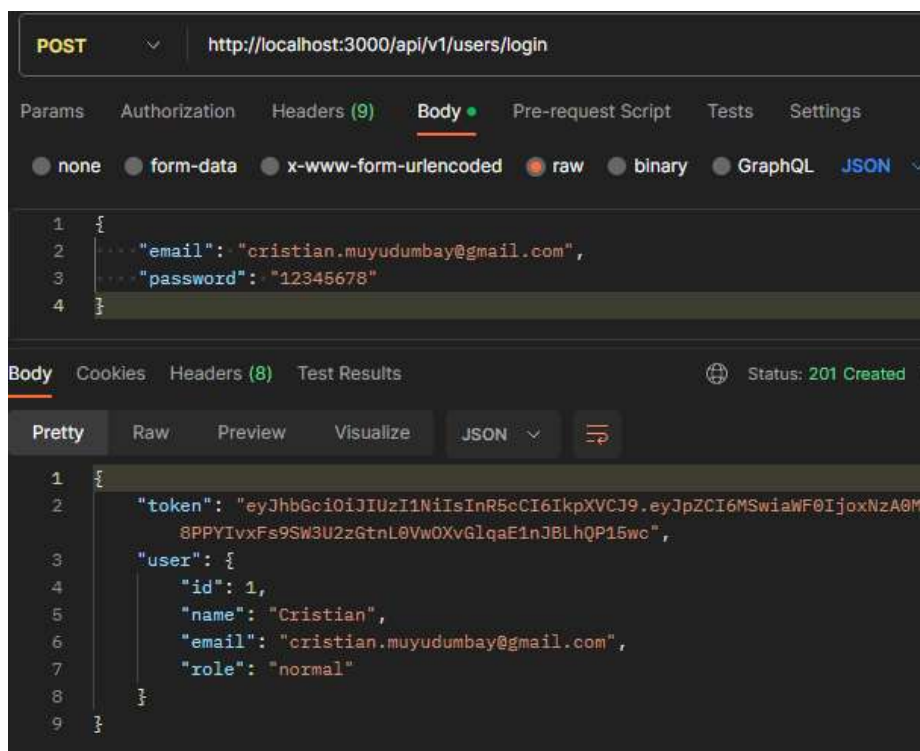
USERS

POST /signup - Crear usuario (enviar name, email, y password por req.body) (opcional el role)

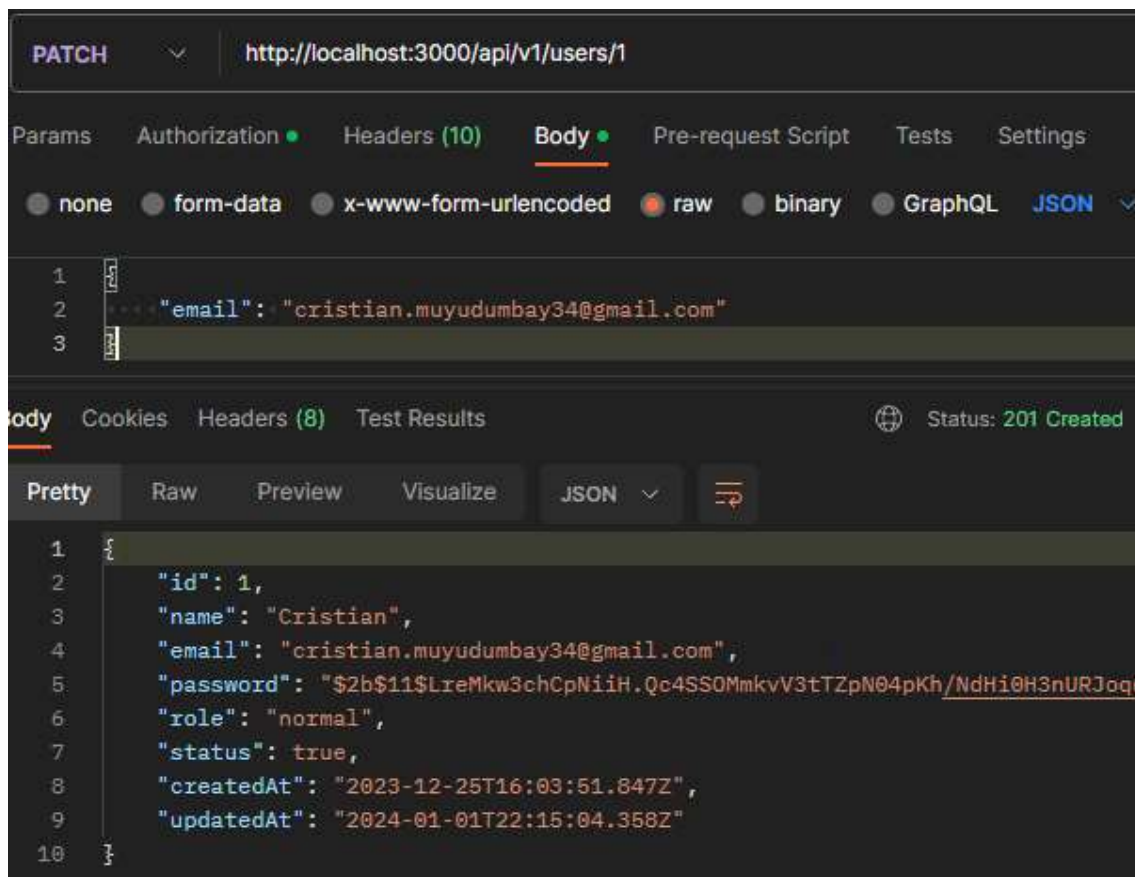


id	name	email	password	role	status
1	Cristian	cristian.muyudumbay@gmail.com	\$2b\$11\$LreMkw3chCpNii...	admin	TRUE
2	Paul	paulProano@gmail.com	\$2b\$11\$QHaS8j9Tymi8I8E...	normal	TRUE
3	Karolina	karoguaman@gmail.com	\$2b\$11\$MgLSQ8ka.G9k1...	admin	TRUE

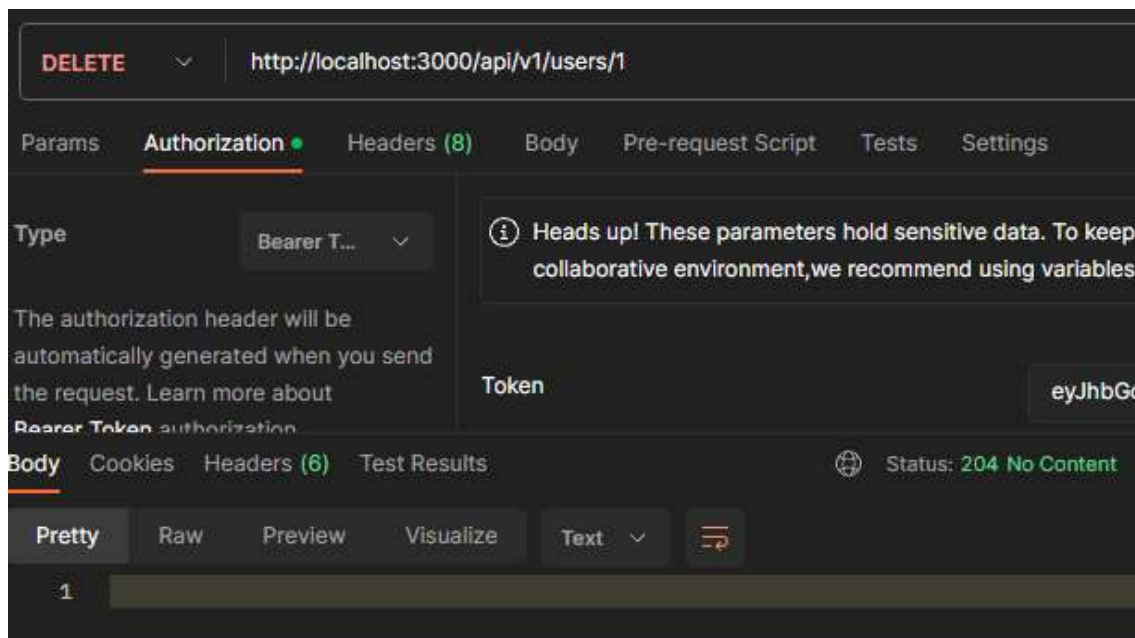
POST /login - Iniciar sesión (enviar email y password por req.body)



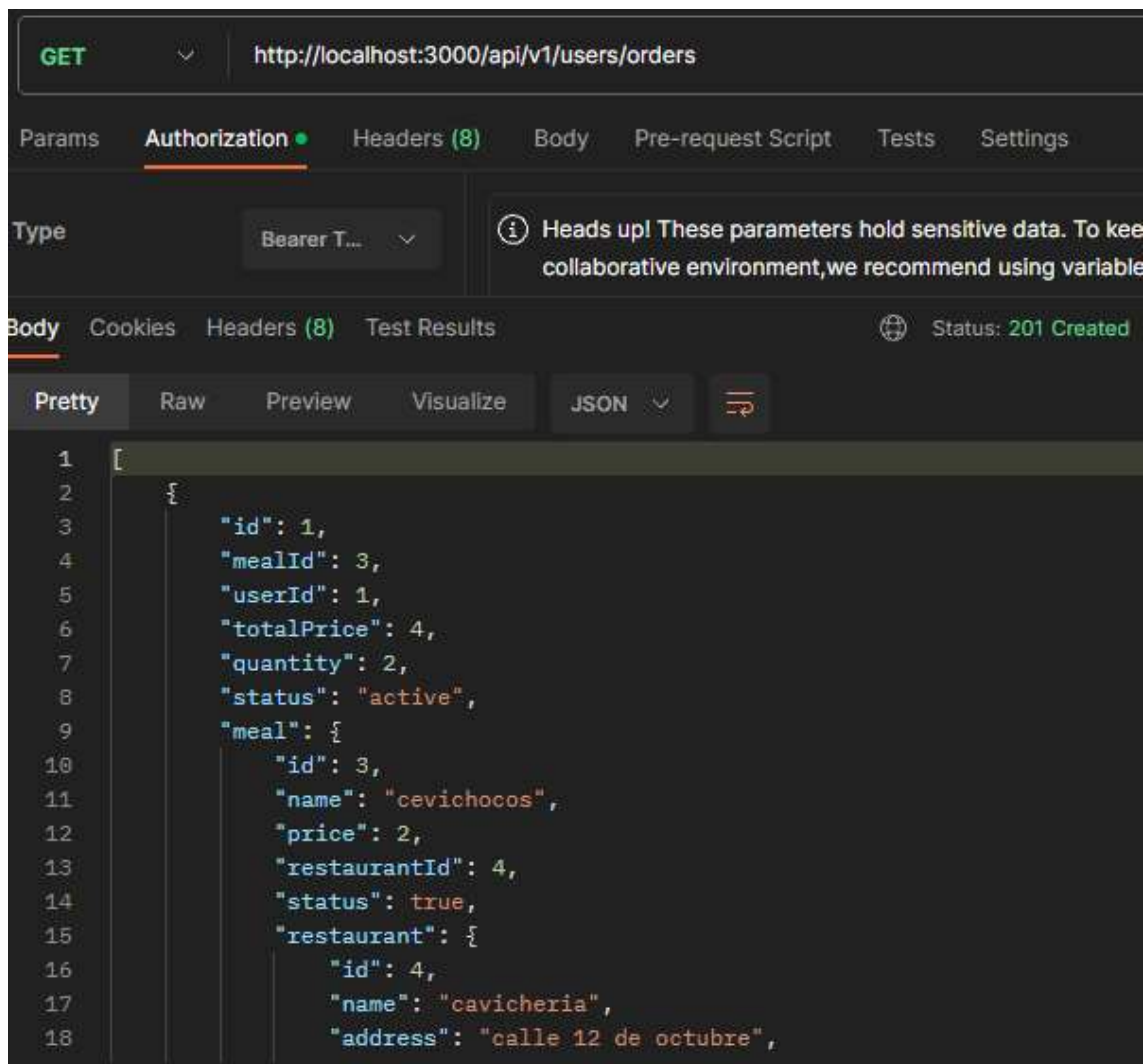
PATCH /:id - Actualizar perfil de usuario (solo name y email)



DELETE /:id - Deshabilitar cuenta de usuario



GET /orders - Obtener todas las ordenes hechas por el usuario



GET <http://localhost:3000/api/v1/users/orders>

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

Type Bearer T... *Heads up! These parameters hold sensitive data. To keep your collaborative environment, we recommend using variables.*

Body Cookies Headers (8) Test Results Status: 201 Created

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "id": 1,
4     "mealId": 3,
5     "userId": 1,
6     "totalPrice": 4,
7     "quantity": 2,
8     "status": "active",
9     "meal": {
10      "id": 3,
11      "name": "cevichocos",
12      "price": 2,
13      "restaurantId": 4,
14      "status": true,
15      "restaurant": {
16        "id": 4,
17        "name": "cavicheria",
18        "address": "calle 12 de octubre",
```

```
      "rating": 4,
      "status": true
    }
  },
  {
    "id": 2,
    "mealId": 2,
    "userId": 1,
    "totalPrice": 6.75,
    "quantity": 3,
    "status": "active",
    "meal": {
      "id": 2,
      "name": "papas fritas",
      "price": 2.25,
      "restaurantId": 3,
      "status": true,
```

```

    "restaurant": {
      "id": 3,
      "name": "kfc",
      "address": "calle 24 de mayo",
      "rating": 4,
      "status": true
    }
  }
}
]

```

GET /orders/:id - Obtener detalles de una sola orden dado un ID

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

- Method:** GET
- URL:** http://localhost:3000/api/v1/users/orders/2
- Authorization:** Bearer T...
- Status:** 201 Created
- Response Body (JSON):**

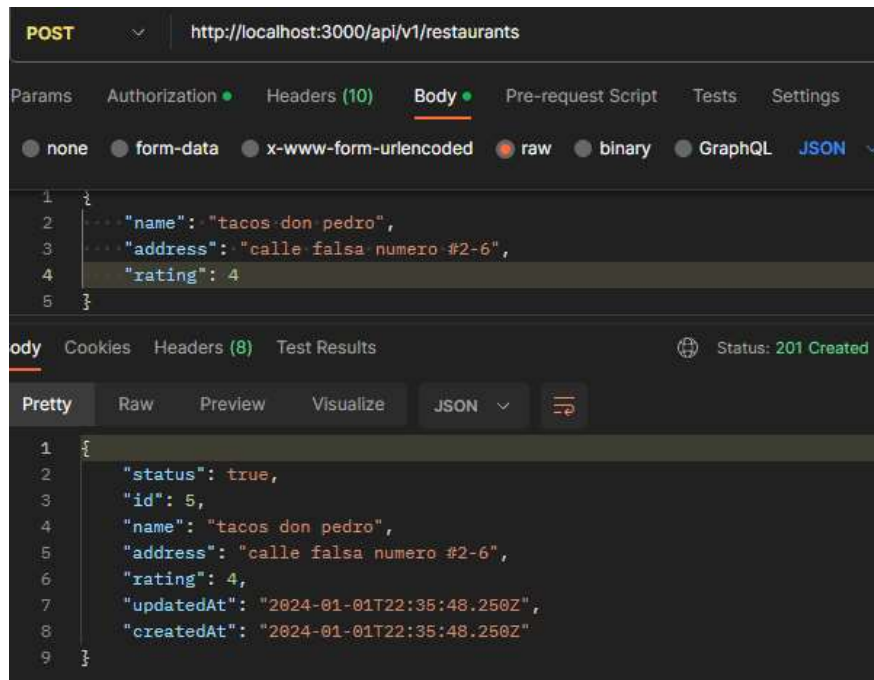
```

{
  "id": 2,
  "mealId": 2,
  "userId": 1,
  "totalPrice": 6.75,
  "quantity": 3,
  "status": "active",
  "meal": {
    "id": 2,
    "name": "papas fritas",
    "price": 2.25,
    "restaurantId": 3,
    "status": true,
    "restaurant": {
      "id": 3,
      "name": "kfc",
      "address": "calle 24 de mayo",
      "rating": 4,

```

RESTAURANTS

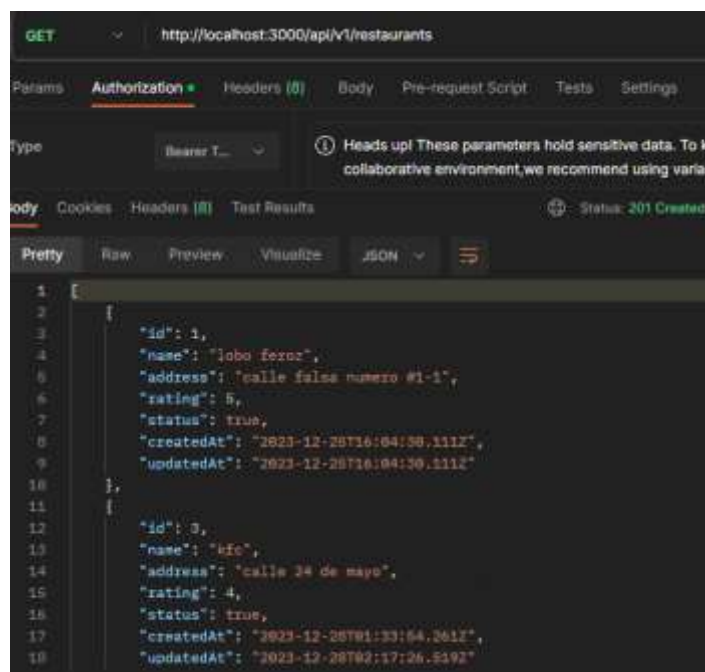
POST / - Crear un nuevo restaurant (enviar name, address, rating (INT)) rating debe ser un valor del 1 al 5



The screenshot shows a REST client interface with a POST request to `http://localhost:3000/api/v1/restaurants`. The request body is a JSON object: `{ "name": "tacos don pedro", "address": "calle falsa numero #2-6", "rating": 4 }`. The response status is 201 Created. The response body is a JSON object: `{ "status": true, "id": 5, "name": "tacos don pedro", "address": "calle falsa numero #2-6", "rating": 4, "updatedAt": "2024-01-01T22:35:48.250Z", "createdAt": "2024-01-01T22:35:48.250Z" }`.

id	name	address	rating	status	createdAt	updatedAt
1	lobo feroz	calle falsa numero #1-1	5	TRUE	2023-12-2...	2023-12-25 16:04:30.111+00
2	macdonalds	calle falsa numero #1-1	5	TRUE	2023-12-2...	2023-12-25 17:27:55.344+00
3	kfc	calle 24 de mayo	4	TRUE	2023-12-2...	2023-12-28 02:17:26.519+00
4	cavicheria	calle 12 de octubre	4	TRUE	2023-12-2...	2023-12-28 21:58:59.508+00
5	tacos don pedro	calle falsa numero #2-6	4	TRUE	2024-01-0...	2024-01-01 22:35:48.25+00

GET / - Obtener todos los restaurants con status active



The screenshot shows a REST client interface with a GET request to `http://localhost:3000/api/v1/restaurants`. The response status is 201 Created. The response body is a JSON array of restaurant objects: `[{ "id": 1, "name": "lobo feroz", "address": "calle falsa numero #1-1", "rating": 5, "status": true, "createdAt": "2023-12-28T16:04:30.111Z", "updatedAt": "2023-12-28T16:04:30.111Z" }, { "id": 3, "name": "kfc", "address": "calle 24 de mayo", "rating": 4, "status": true, "createdAt": "2023-12-28T01:33:04.261Z", "updatedAt": "2023-12-28T02:17:26.519Z" }]`.

```
{
  "id": 2,
  "name": "macdonalds",
  "address": "calle falsa numero #1-1",
  "rating": 5,
  "status": true,
  "createdAt": "2023-12-25T17:27:55.344Z",
  "updatedAt": "2023-12-25T17:27:55.344Z"
},
{
  "id": 4,
  "name": "cavicheria",
  "address": "calle 12 de octubre",
  "rating": 4,
  "status": true,
  "createdAt": "2023-12-28T21:52:20.886Z",
  "updatedAt": "2023-12-28T21:58:59.508Z"
},
}
```

```
{
  "id": 5,
  "name": "tacos don pedro",
  "address": "calle falsa numero #2-6",
  "rating": 4,
  "status": true,
  "createdAt": "2024-01-01T22:35:48.250Z",
  "updatedAt": "2024-01-01T22:35:48.250Z"
}
```

GET /:id - Obtener restaurant por id

GET <http://localhost:3000/api/v1/restaurants/5>

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

Query Params

Key	Value
Key	Value

body Cookies Headers (8) Test Results ⊕ Status: **201 Created**

Pretty Raw Preview Visualize JSON ⌵ ⌵

```

1  {
2    "id": 5,
3    "name": "tacos don pedro",
4    "address": "calle falsa numero #2-6",
5    "rating": 4,
6    "status": true,
7    "createdAt": "2024-01-01T22:35:48.250Z",
8    "updatedAt": "2024-01-01T22:35:48.250Z"
9  }
```


PATCH /:id - Actualizar restaurant (name, address)

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

- Method:** PATCH
- URL:** http://localhost:3000/api/v1/restaurants/5
- Body:** A JSON object with the following structure:

```
{  "address": "calle 16 de noviembre"}
```
- Status:** 201 Created
- Response Body:** A JSON object with the following structure:

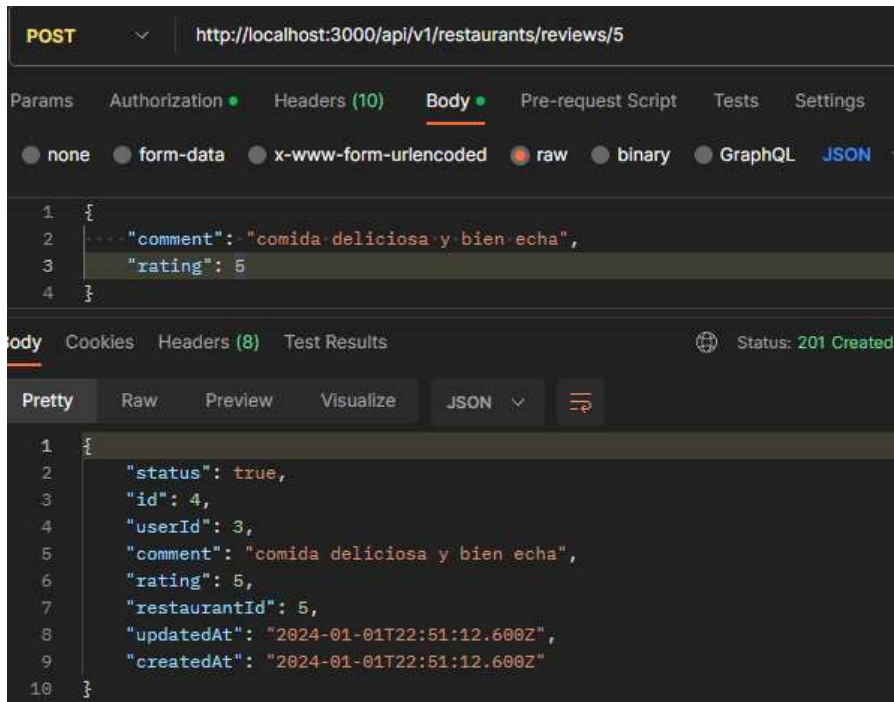
```
{  "id": 5,  "name": "tacos don pedro",  "address": "calle 16 de noviembre",  "rating": 4,  "status": true,  "createdAt": "2024-01-01T22:35:48.250Z",  "updatedAt": "2024-01-01T22:43:51.600Z"}
```

DELETE /:id - Deshabilitar restaurant.

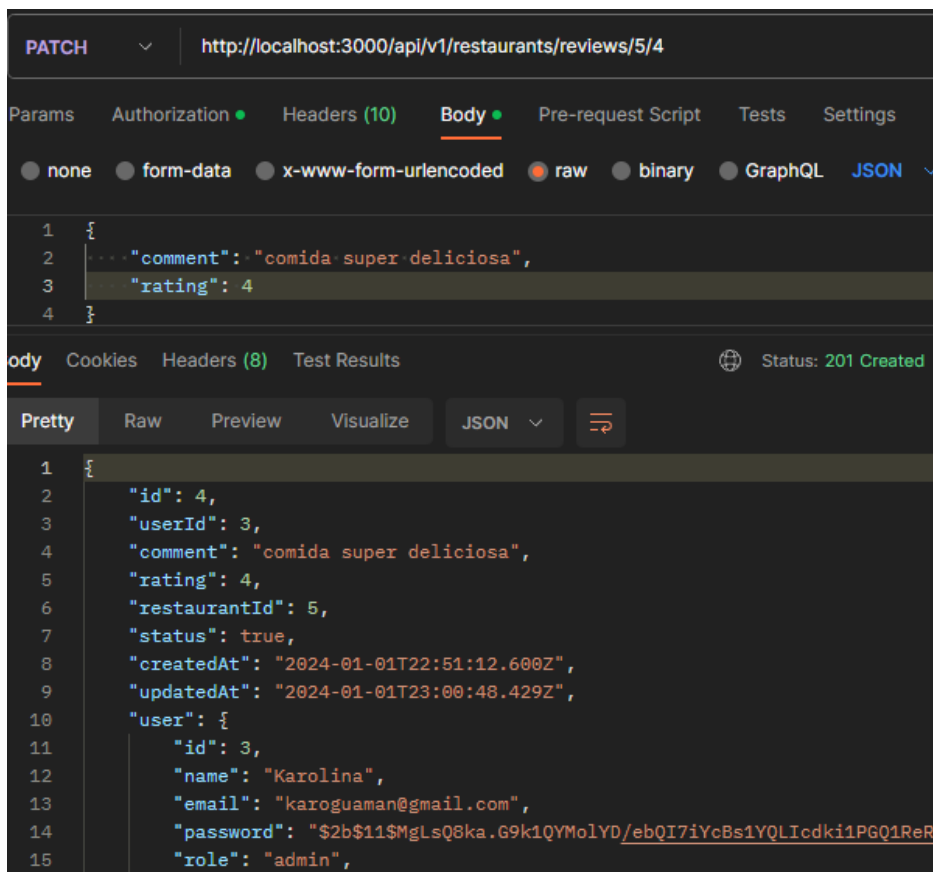
The screenshot shows a REST client interface with the following details:

- Method:** DELETE
- URL:** http://localhost:3000/api/v1/restaurants/5
- Authorization:** Bearer Token (Type: Bearer T...)
- Status:** 204 No Content
- Response Body:** Empty (Text view)

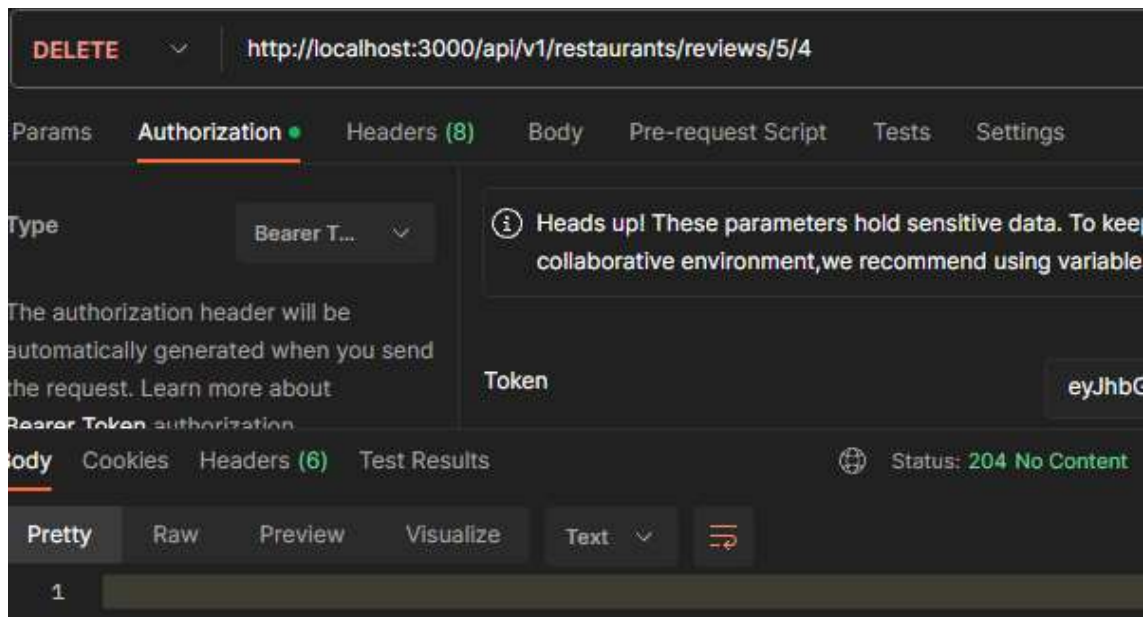
POST /reviews/:id - Crear una nueva reseña en el restaurant, siendo :id el id del restaurant (enviar comment, rating (INT) en req.body)



PATCH /reviews/:restaurantId/:id - Actualizar una reseña hecha en un restaurant, siendo :id el id del restaurant (comment, rating) SOLO EL AUTOR DE LA RESEÑA PUEDE ACTUALIZAR SU PROPIA RESEÑA

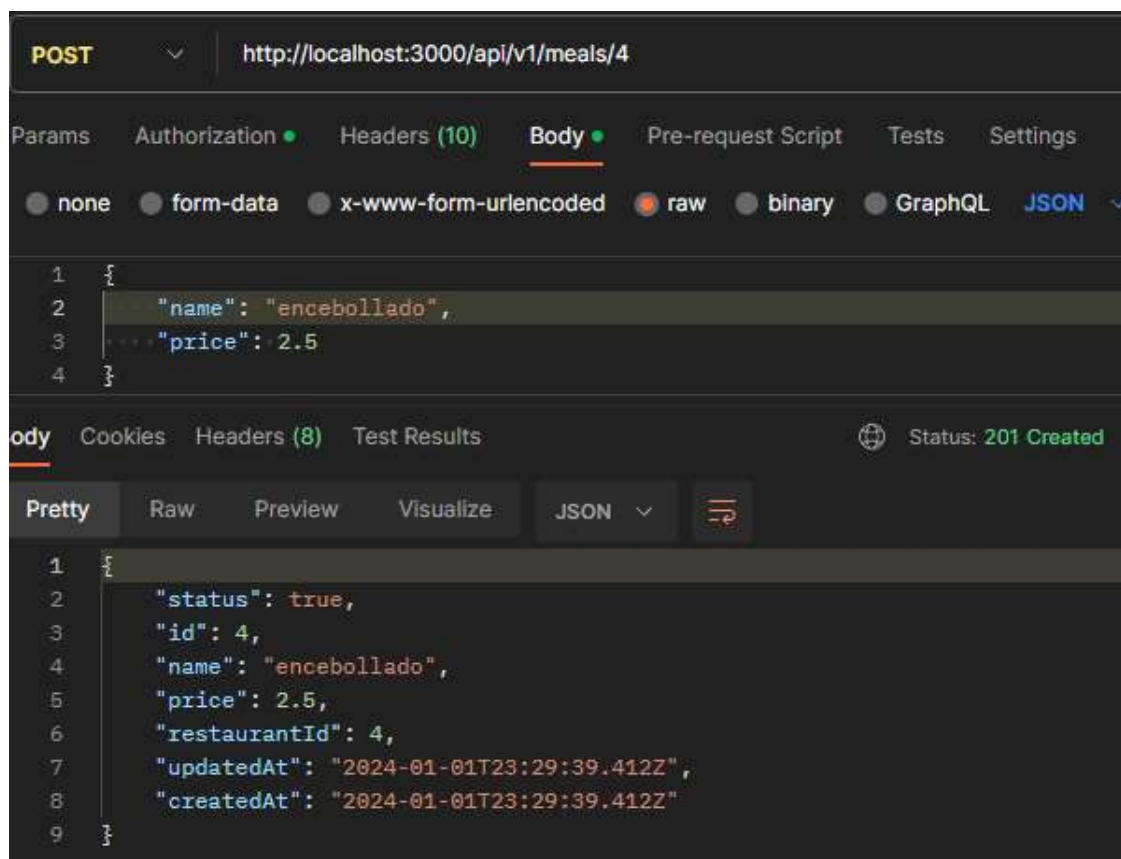


DELETE /reviews/:restaurantId/:id - Actualizar una reseña hecha en un restaurant a status deleted, siendo :id el id del restaurant. SOLO EL AUTOR DE LA RESEÑA PUEDE ACTUALIZAR SU PROPIA RESEÑA



MEALS

POST /:id - Crear una nueva comida en el restaurant, siendo :id el id del restaurant (enviar name, price (INT) en req.body)



GET / - Obtener todas las comidas con status active

GET <http://localhost:3000/api/v1/meals>

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

Headers 8 hidden

Key	Value
-----	-------

body Cookies Headers (8) Test Results Status: 201 Created

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "id": 1,
4     "name": "arroz con pollo",
5     "price": 2.5,
6     "restaurantId": 3,
7     "status": true,
8     "createdAt": "2023-12-28T23:18:46.471Z",
9     "updatedAt": "2023-12-28T23:18:46.471Z",
10    "restaurant": {
11      "id": 3,
12      "name": "kfc",
13      "address": "calle 24 de mayo",
14      "rating": 4
15    }
16  }
```

```
    {
      "id": 3,
      "name": "cevichocos",
      "price": 2,
      "restaurantId": 4,
      "status": true,
      "createdAt": "2023-12-28T23:56:45.159Z",
      "updatedAt": "2023-12-28T23:56:45.159Z",
      "restaurant": {
        "id": 4,
        "name": "cavicheria",
        "address": "calle 12 de octubre",
        "rating": 4
      }
    },
    {
      "id": 2,
      "name": "papas fritas",
```

```
        "price": 2.25,
        "restaurantId": 3,
        "status": true,
        "createdAt": "2023-12-28T23:21:43.806Z",
        "updatedAt": "2023-12-28T23:43:35.840Z",
        "restaurant": {
          "id": 3,
          "name": "kfc",
          "address": "calle 24 de mayo",
          "rating": 4
        }
      },
      {
        "id": 4,
        "name": "encebollado",
        "price": 2.5,
        "restaurantId": 4,
        "status": true,
```

```
    "status": true,  
    "createdAt": "2024-01-01T23:29:39.412Z",  
    "updatedAt": "2024-01-01T23:29:39.412Z",  
    "restaurant": {  
      "id": 4,  
      "name": "cavicheria",  
      "address": "calle 12 de octubre",  
      "rating": 4  
    }  
  }  
}
```

GET /:id - Obtener por id una comida con status active

GET <http://localhost:3000/api/v1/meals/4>

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

Headers 8 hidden

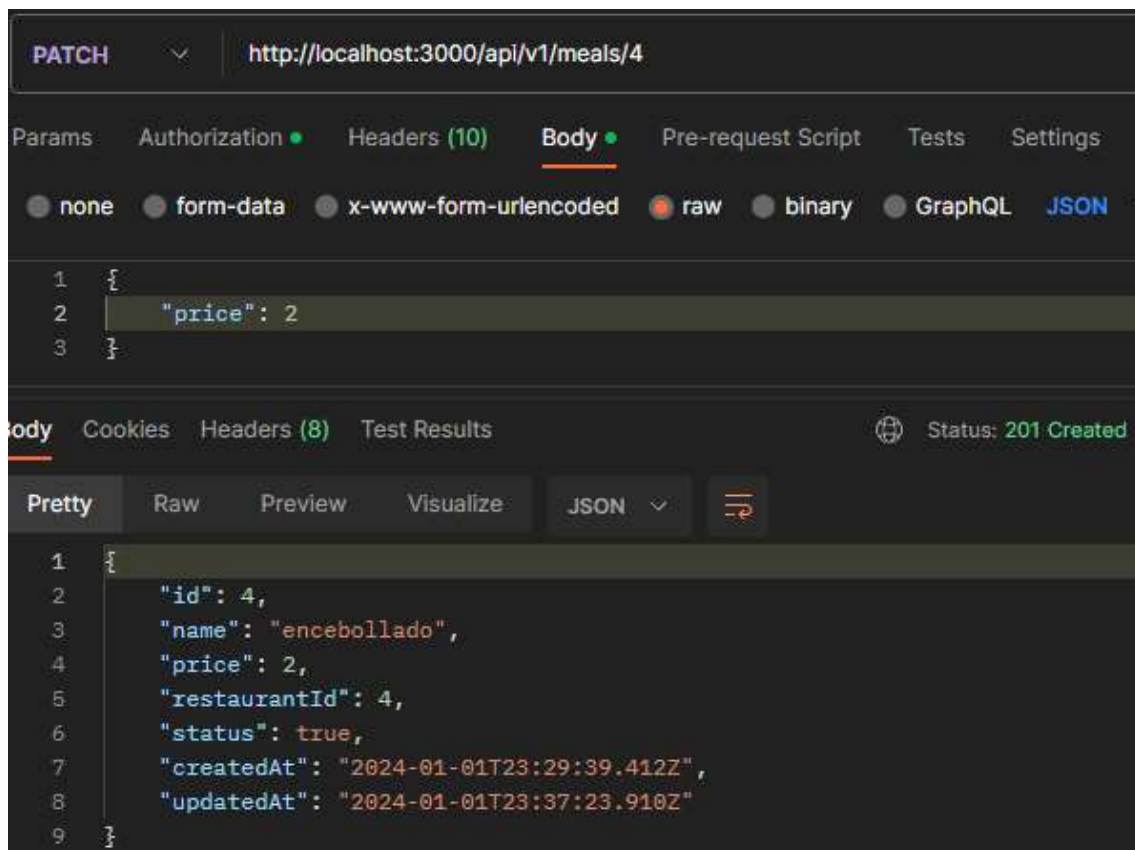
Key	Value
Key	Value

Body Cookies Headers (8) Test Results Status: 201 Created

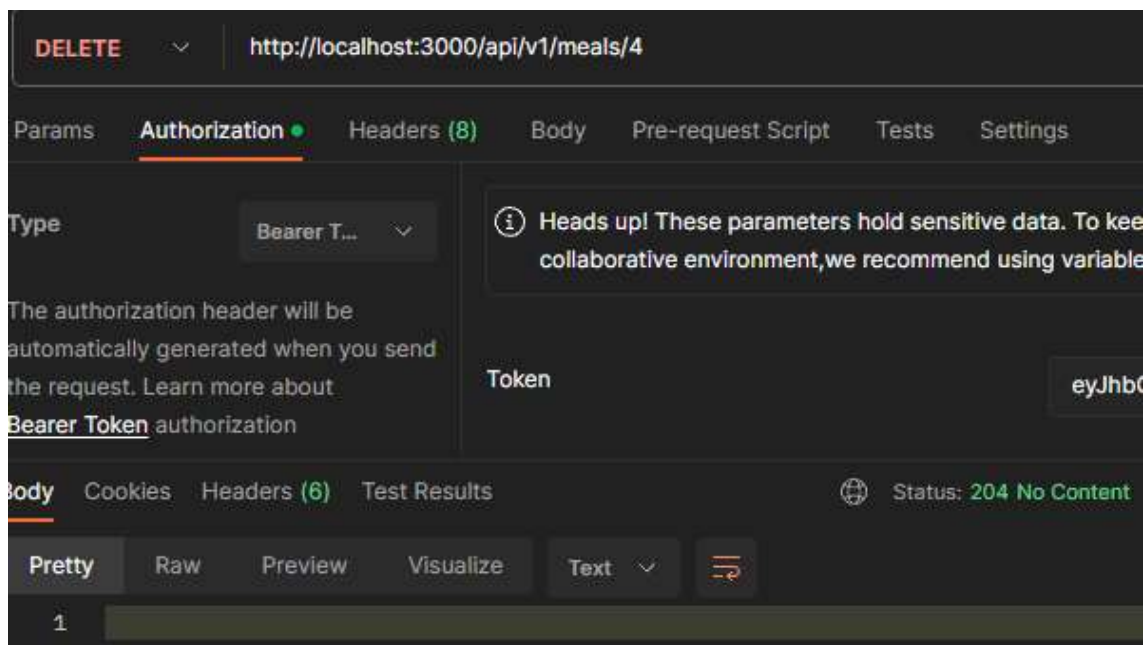
Pretty Raw Preview Visualize JSON

```
1 {  
2   "id": 4,  
3   "name": "encebollado",  
4   "price": 2.5,  
5   "restaurantId": 4,  
6   "status": true,  
7   "createdAt": "2024-01-01T23:29:39.412Z",  
8   "updatedAt": "2024-01-01T23:29:39.412Z"  
9 }
```

PATCH /:id - Actualizar comida (name, price)

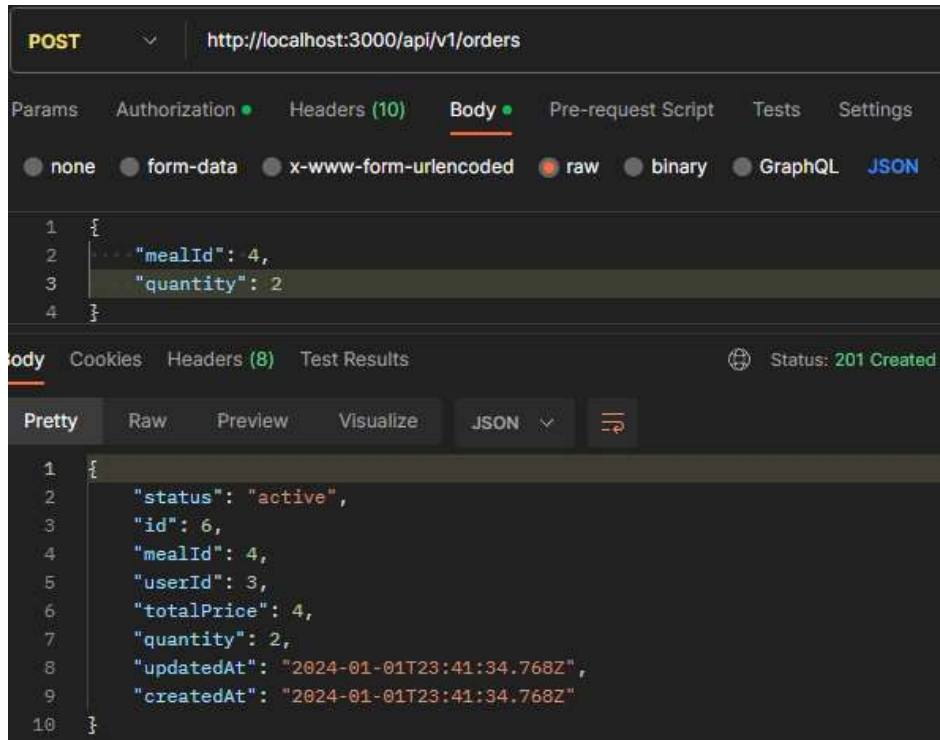


DELETE /:id - Deshabilitar comida

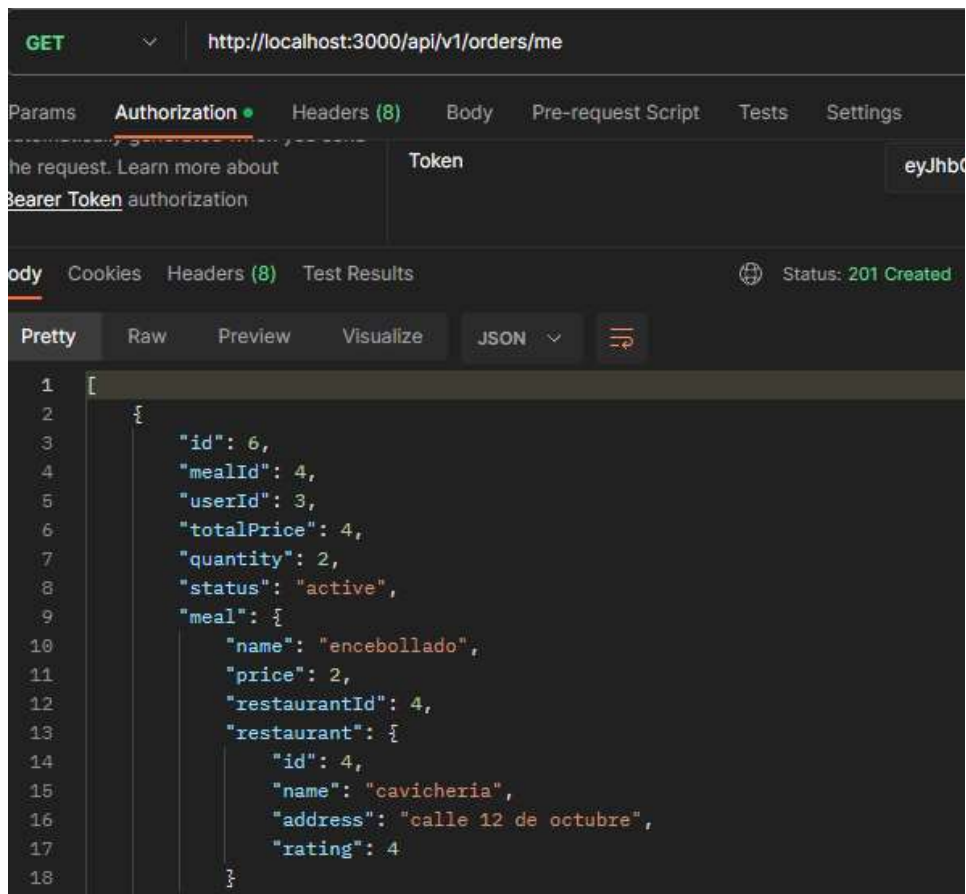


ORDERS

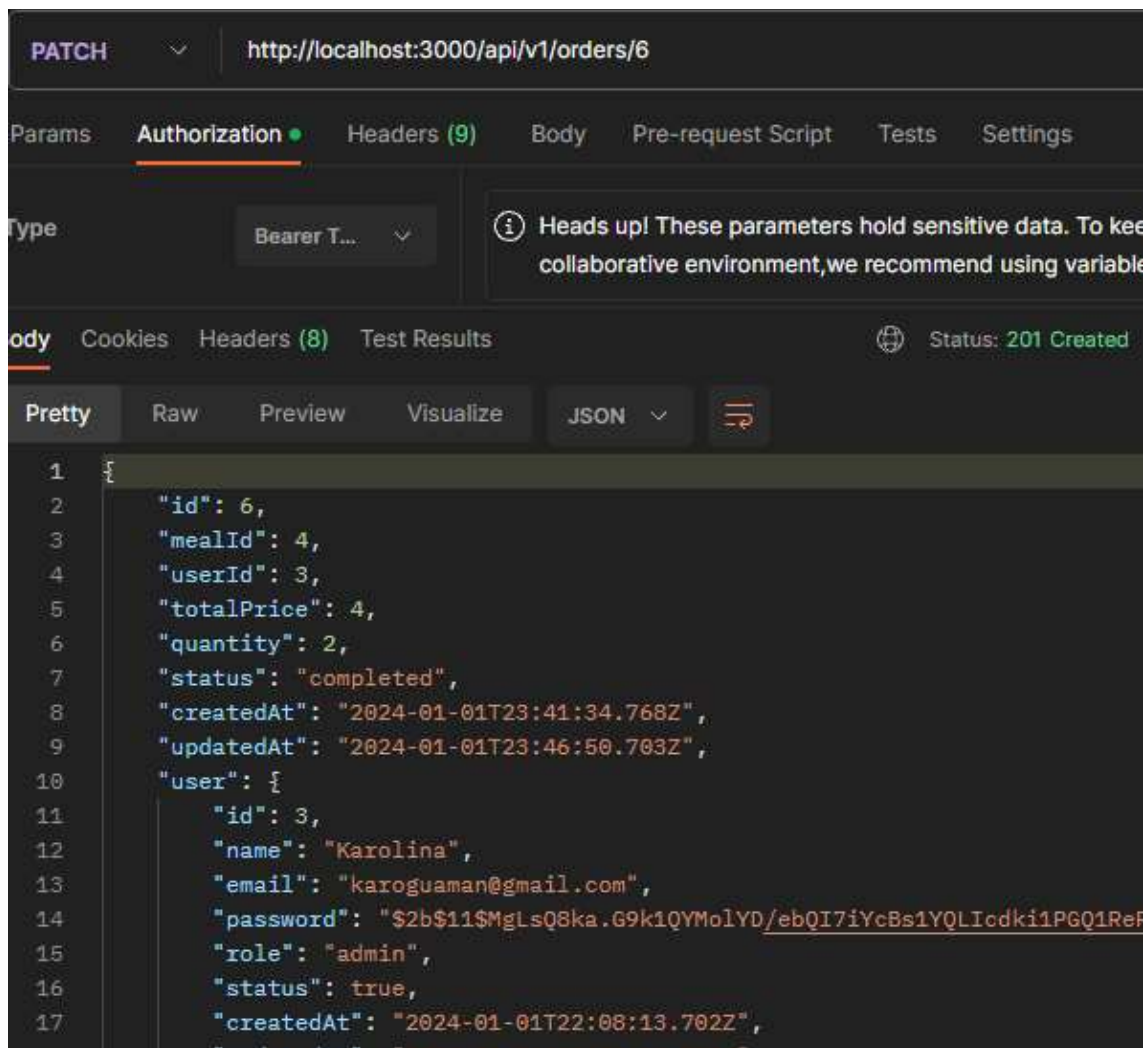
POST / - Crear una nueva order (enviar quantity y mealId por req.body)



GET /me - Obtener todas las órdenes del usuario



PATCH /:id - Marcar una orden por id con status completed



DELETE /:id - Marcar una orden por id con status cancelled

