

**Universidad ORT Uruguay**  
**Facultad de Ingeniería**  
**Bernard Wand Polak**

**Diseño de Aplicaciones II**

**Evidencia de la ejecución de las pruebas de la API con  
Postman**

**<https://github.com/ORT-DA2/164660-185720-234059>**

**Santiago Alvarez – N° Estudiante: 185720**

**Juan Castro – N° Estudiante: 164660**

**Marcelo Tilve – N° Estudiante: 234059**

## Contenido

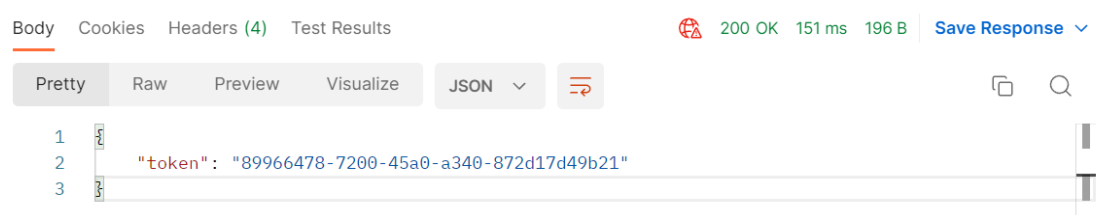
<b>Casos de prueba para funcionalidades</b>	<b>3</b>
<b>Pruebas Anónimas</b>	<b>3</b>
Login	3
Crear Usuario	3
Realizar compra	5
<b>Pruebas Administrador</b>	<b>7</b>
Crear invitación	7
Pharmacy	8
Crear Farmacia	9
Drug	11
<b>Pruebas Dueño</b>	<b>14</b>
Listar compras por mes	14
Ver solicitudes de reposición de stock	15
<b>Pruebas Empleado</b>	<b>16</b>
Listar compras	16
Crear solicitud de stock	16
<b>Video con las pruebas</b>	<b>17</b>

## Casos de prueba para funcionalidades

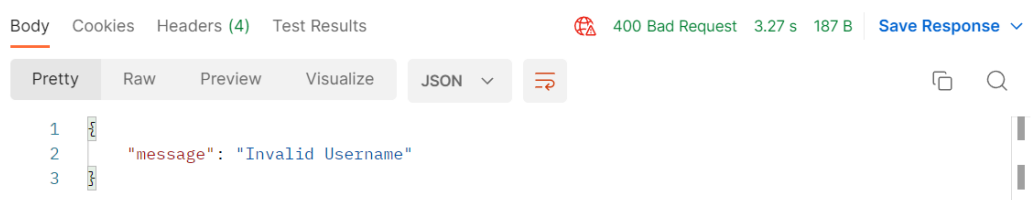
### Pruebas Anónimas

#### Login

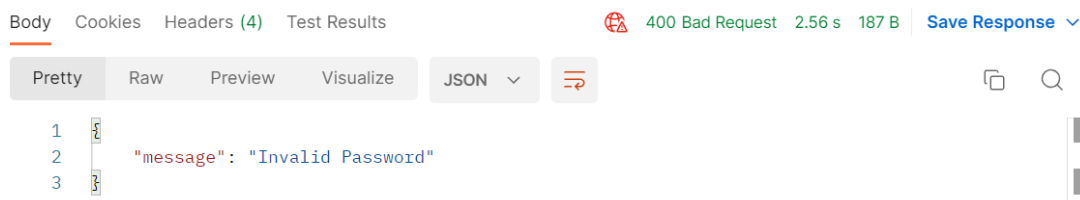
- Login de un usuario - Retorna el token de autorización correspondiente a la sesión, el mismo es un GUID - Ok 200



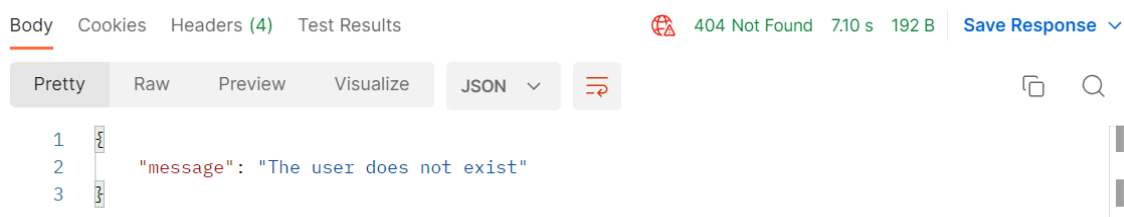
- Error 400 - Nombre de Usuario no válido.



- Error 400 - Password no válido.



- Error 404 - Usuario no encontrado



#### Crear Usuario

- Crear un usuario a partir del UserName y el UserCode
  - Ok - 200
  - Retorna un json con los datos del usuario recién creado.

Body Cookies Headers (4) Test Results 200 OK 1419 ms 301 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "userName": "pablo1986",
3   "email": "roberto@gmail.com",
4   "address": "Br. Artigas 1486 entre Palmar y Av. Rivera",
5   "registrationDate": "2022-09-23T20:22:14.101Z"
6 }
    
```

- Error 400 - Nombre de usuario no válido.

Body Cookies Headers (4) Test Results 400 Bad Request 4.59 s 187 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "message": "Invalid Username"
3 }
    
```

- Error 400 - Código de Usuario vacío o no es numérico de 6 dígitos.

Body Cookies Headers (4) Test Results 400 Bad Request 4.76 s 187 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "message": "Invalid UserCode"
3 }
    
```

- Error 400 - El Email está vacío o no cumple con el formato de correo electrónico válido.

Body Cookies Headers (4) Test Results 400 Bad Request 3.40 s 184 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "message": "Invalid Email"
3 }
    
```

- Error 400 - El Password es vacío o no cumple con que tenga mínimo 8 caracteres, por lo menos una mayúscula, un número y un carácter especial.

Body Cookies Headers (4) Test Results 400 Bad Request 5.29 s 187 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "message": "Invalid Password"
3 }
    
```

- Error 400 - La dirección está vacía.

Body Cookies Headers (4) Test Results 400 Bad Request 4.74 s 186 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Invalid Address"
3 }
```

- Error 400 - El nombre de usuario ya existe en el sistema.

Body Cookies Headers (4) Test Results 400 Bad Request 12.68 s 212 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Invalid Username, Username already exists"
3 }
```

- Error 400 - El Email ya existe en el sistema.

Body Cookies Headers (4) Test Results 400 Bad Request 5.23 s 206 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Invalid Email, Email already exists"
3 }
```

- Error 404 - La invitación ya fue utili

Body Cookies Headers (4) Test Results 404 Not Found 20.54 s 216 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Invitation not found or is not currently active"
3 }
```

## Realizar compra

- Realizar la compra de uno o varios medicamentos
  - Ok - 200
  - Retorna un json con los datos de la compra a nivel cabecera (email comprador, fecha de la compra, farmacia donde se realizó la compra, monto total de la compra y el detalle de cada ítem comprado)

Body Cookies Headers (4) Test Results 200 OK 14.45 s 433 B Save Response

Pretty Raw Preview Visualize JSON

```

1  {
2    "buyerEmail": "norberto.perez@gmail.com",
3    "purchaseDate": "2022-10-01T15:21:31.816Z",
4    "pharmacyId": 1,
5    "pharmacyName": "Farmacia 1",
6    "totalAmount": 1250.00,
7    "details": [
8      {
9        "code": "XF324",
10       "name": "Aspirina",
11       "quantity": 5,
12       "price": 100.00
13     },
14     {
15       "code": "RS546",
16       "name": "Abrilar",
17       "quantity": 3,
18       "price": 250.00
19     }
20   ]
21 }

```

- Error 400 - Email invalido

Body Cookies Headers (4) Test Results 400 Bad Request 3.75 s 184 B Save Response

Pretty Raw Preview Visualize JSON

```

1  {
2    "message": "Invalid Email"
3  }

```

- Error 400 - Lista de items vacía

Body Cookies Headers (4) Test Results 400 Bad Request 11.88 s 208 B Save Response

Pretty Raw Preview Visualize JSON

```

1  {
2    "message": "The list of items can't be empty"
3  }

```

- Error 400 - Cantidad de algún medicamento es mayor al del stock disponible

Body Cookies Headers (4) Test Results 400 Bad Request 5.61 s 201 B Save Response

Pretty Raw Preview Visualize JSON

```

1  {
2    "message": "The Drug XF324 is out of stock"
3  }

```

- Error 404 - No se encuentra la farmacia asociada a la compra

Body Cookies Headers (4) Test Results 404 Not Found 14.24 s 192 B Save Response

Pretty Raw Preview Visualize JSON

```

1  {
2    "message": "Pharmacy 1000 not found"
3  }

```

- Error 400 - La cantidad de un medicamento es  $\leq 0$

Body Cookies Headers (4) Test Results 400 Bad Request 6.59 s 204 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "The Quantity is a mandatory field"
3 }
```

- Error 404 - No se encuentra alguno de los medicamentos a comprar

Body Cookies Headers (4) Test Results 404 Not Found 8.67 s 190 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Drug XF3200 not found"
3 }
```

## Pruebas Administrador

### Crear invitación

- Respuesta: 200 OK
- Retorna mensaje JSON con la información del “userName” y “userCode” creado para que el usuario que recibe la invitación pueda ingresar al sistema.

Body Cookies Headers (4) Test Results Status: 200 OK Time: 167 ms Size: 188 B

Pretty Raw Preview Visualize JSON

```
1 {
2   "userName": "test1",
3   "userCode": "123973"
4 }
```

- Crear invitación usuario administrador con farmacia
- Respuesta: 400 Bad Request
- Retorna mensaje JSON indicando que el usuario administrador no debe tener una farmacia.

Body Cookies Headers (4) Test Results Status: 400 Bad Request Time: 1753 ms Size: 198 B

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "A phazarmacy is not required."
3 }
```

- Crear invitación usuario dueño y empleado sin farmacia
- Respuesta: 400 Bad Request
- Retorna mensaje JSON indicando que el usuario dueño o empleado debe tener una farmacia asociada.

```

1 {
2   "message": "A pharmacy is required."
3 }

```

### Crear invitación con usuario repetido

- Respuesta: 400 Bad Request
- Retorna mensaje JSON indicando que el usuario utilizado para para crear la invitación no se puede usar.

```

1 {
2   "message": "Invitation already exist."
3 }

```

- Crear invitación sin usuario
- Respuesta: 400 Bad Request
- Retorna mensaje JSON indicando que el usuario no es válido.

```

1 {
2   "message": "Invalid UserName."
3 }

```

- Crear invitación sin rol
- Respuesta: 400 Bad Request
- Retorna mensaje JSON indicando que el rol no es válido.

```

1 {
2   "message": "Invalid Role."
3 }

```

- Crear invitación con rol inválido
- Respuesta: 400 Bad Request
- Retorna mensaje JSON indicando que el rol no es válido.

```

1 {
2   "message": "Invalid Role."
3 }

```

### Pharmacy

Para este controlador, todos los endpoints deben ser accedidos por un usuario Administrador logueado. De lo contrario la response es la siguiente:



Body Cookies Headers (4) Test Results

Status: 401 Unauthorized Time: 3.16 s Size: 199 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "message": "Invalid authorization token"
3 }

```

- Error 401 - Unauthorized

## Crear Farmacia

- Se crea una Farmacia con su nombre y dirección.

Body Cookies Headers (4) Test Results

Status: 200 OK Time: 383 ms Size: 216 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "id": 1,
3   "name": "pharmacy",
4   "address": "address",
5   "users": [],
6   "drugs": []
7 }

```

- Status: 200 Ok
- Body: Retorna la farmacia creada en la base de datos.

## Crear Farmacia ya existente

Body Cookies Headers (4) Test Results

Status: 400 Bad Request Time: 5.79 s Size: 198 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "message": "The pharmacy already exist."
3 }

```

- Status: 400 Bad Request
- Body: mensaje explicando que la farmacia ya existe.

## Crear Farmacia con nombre incorrecto

Body Cookies Headers (4) Test Results

Status: 400 Bad Request Time: 2.59 s Size: 209 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "message": "The Pharmacy is not correctly created."
3 }

```

- Status: 400 Bad Request
- Body: Mensaje explicando que la farmacia no fue correctamente creada.

## Obtener Farmacias sin parámetros

Body Cookies Headers (4) Test Results

Status: 200 OK Time: 63 ms Size: 204 B Save Response

Pretty Raw Preview Visualize JSON

```

1 [
2   {
3     "id": 1,
4     "name": "pharmacy"
5   },
6   {
7     "id": 2,
8     "name": "pharmacy2"
9   }
10 ]

```

- Status: 200 Ok
- Body: Listado de todas las Farmacias de la base de datos con su Id y Name.

## Obtener Farmacias con parámetros nombre y dirección



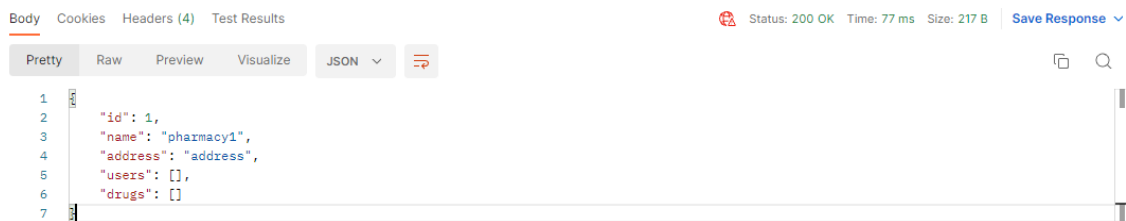
- Status: 200 Ok
- Body: Listado de Farmacias con el nombre (pharmacy) y dirección (address) del query string.

## Obtener Farmacias con parámetros inexistentes



- Status: 200 Ok
- Body: Listado vacío de Farmacias con nombre y dirección inexistentes del query string.

## Modificar Farmacia



- Status: 200 Ok
- Body: Farmacia de la base de datos con el nombre actualizado

## Modificar Farmacia inexistente



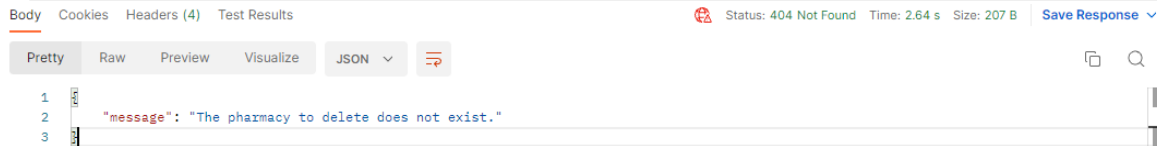
- Status: 404 Not Found
- Body: Mensaje explicando que la farmacia a modificar no existe.

## Borrar Farmacia



- Status: 200 Ok

## Borrar Farmacia inexistente



```

1  {
2    "message": "The pharmacy to delete does not exist."
3  }

```

- Status: 404 Not Found
- Body: Mensaje explicando que la farmacia a borrar no existe.

## Droga

Para este controlador, los endpoints Create y Delete deben ser accedidos por un usuario Employee logueado. Para los endpoints restantes, el usuario logueado debe ser Administrator. De lo contrario la response es la siguiente:



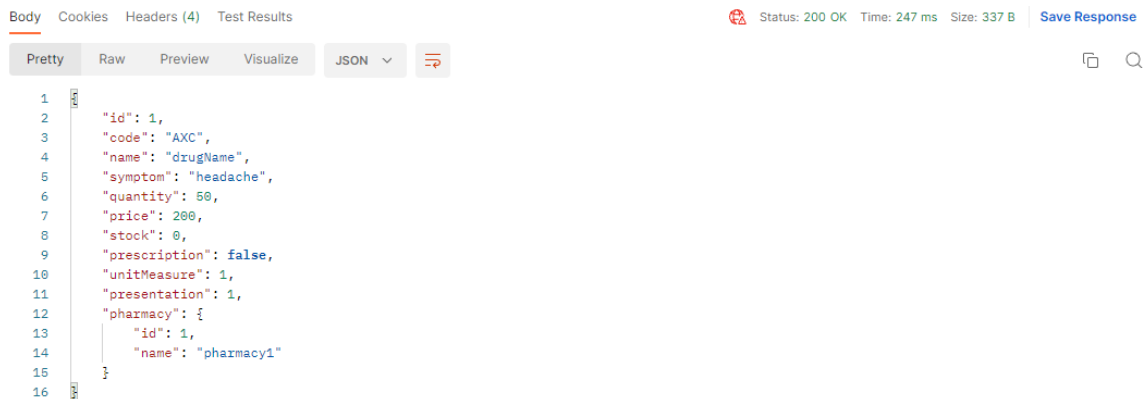
```

1  {
2    "message": "Invalid authorization token"
3  }

```

- Error 401 - Unauthorized

## Crear Drug



```

1  {
2    "id": 1,
3    "code": "AXC",
4    "name": "drugName",
5    "symptom": "headache",
6    "quantity": 60,
7    "price": 200,
8    "stock": 0,
9    "prescription": false,
10   "unitMeasure": 1,
11   "presentation": 1,
12   "pharmacy": {
13     "id": 1,
14     "name": "pharmacy1"
15   }
16 }

```

- Status: 200 Ok
- Body: Retorna la Drug creada de la base de datos.

## Crear Drug con código existente en la farmacia



```

1  {
2    "message": "The drug already exists in that pharmacy."
3  }

```

- Status: 400 Bad Request
- Body: Mensaje explicando que la drug ya existe para esa farmacia

## Crear Drug con farmacia inexistente



Body Cookies Headers (4) Test Results Status: 404 Not Found Time: 9.40 s Size: 209 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "The pharmacy of the drug does not exist."
3 }
```

- Status: 404 Not Found
- Body: Mensaje explicando que la farmacia de la drug no existe.

### Obtener Drugs sin parámetros




Body Cookies Headers (4) Test Results Status: 200 OK Time: 30 ms Size: 189 B Save Response

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "id": 1,
4     "code": "AXC",
5     "name": "drugName"
6   }
7 ]
```

- Status: 200 Ok
- Body: Listado de drugs de la base de datos

### Obtener Drugs con parámetros



Body Cookies Headers (4) Test Results Status: 200 OK Time: 50 ms Size: 189 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 1,
3   "code": "AXC",
4   "name": "drugName"
5 }
```

- Status: 200 Ok
- Body: Listado de drugs de la base de datos con el nombre y farmacia de la query

### Obtener Drugs con parámetro de farmacia inexistente



Body Cookies Headers (4) Test Results Status: 404 Not Found Time: 6.73 s Size: 213 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "The pharmacy to get drugs of does not exist."
3 }
```

- Status: 404 Not Found
- Body: Mensaje que explica que la farmacia del query param no existe en la base de datos.

### Borrar Drug



Body Cookies Headers (3) Test Results Status: 200 OK Time: 70 ms Size: 92 B Save Response

Pretty Raw Preview Visualize Text

```
1
```

- Status: 200 Ok

### Borrar Drug inexistente



Body Cookies Headers (4) Test Results

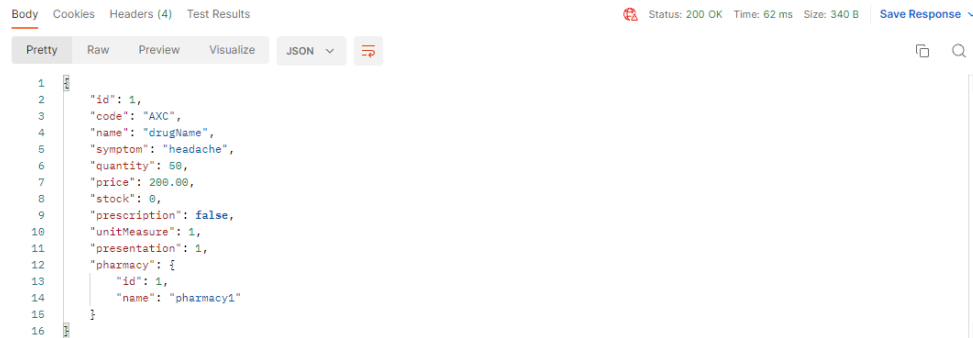
Status: 404 Not Found Time: 2.74 s Size: 203 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "The drug to delete does not exist."
3 }
```

- Status: 404 Not Found
- Body: Mensaje explicando que la drug a borrar no existe.

### Obtener Drug por Id



Body Cookies Headers (4) Test Results

Status: 200 OK Time: 62 ms Size: 340 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 1,
3   "code": "AXC",
4   "name": "drugName",
5   "symptom": "headache",
6   "quantity": 50,
7   "price": 200.00,
8   "stock": 0,
9   "prescription": false,
10  "unitMeasure": 1,
11  "presentation": 1,
12  "pharmacy": {
13    "id": 1,
14    "name": "pharmacy1"
15  }
16 }
```

- Status: 200 Ok
- Body: Retorna la Drug de la base de datos con el id correspondiente

### Obtener Drug por Id inexistente



Body Cookies Headers (4) Test Results

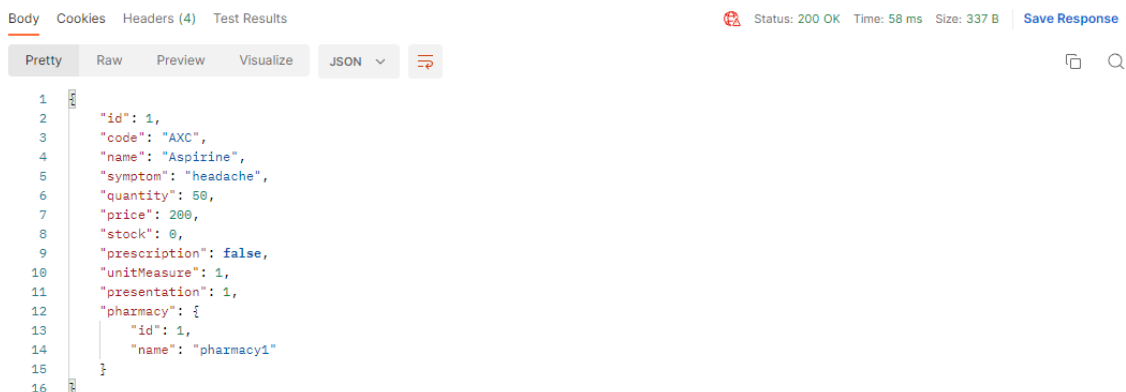
Status: 404 Not Found Time: 1734 ms Size: 193 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "The drug does not exist."
3 }
```

- Status: 404 Not Found
- Body: Mensaje explicando que la droga que se quiere obtener no existe en la base de datos.

### Modificar Drug



Body Cookies Headers (4) Test Results

Status: 200 OK Time: 58 ms Size: 337 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 1,
3   "code": "AXC",
4   "name": "Aspirine",
5   "symptom": "headache",
6   "quantity": 50,
7   "price": 200,
8   "stock": 0,
9   "prescription": false,
10  "unitMeasure": 1,
11  "presentation": 1,
12  "pharmacy": {
13    "id": 1,
14    "name": "pharmacy1"
15  }
16 }
```

- Status: 200 Ok
- Body: Retorna la Drug actualizada de la base de datos.

### Modificar Drug con Id inexistente



Body Cookies Headers (4) Test Results Status: 404 Not Found Time: 1869 ms Size: 203 B Save Response

Pretty Raw Preview Visualize JSON

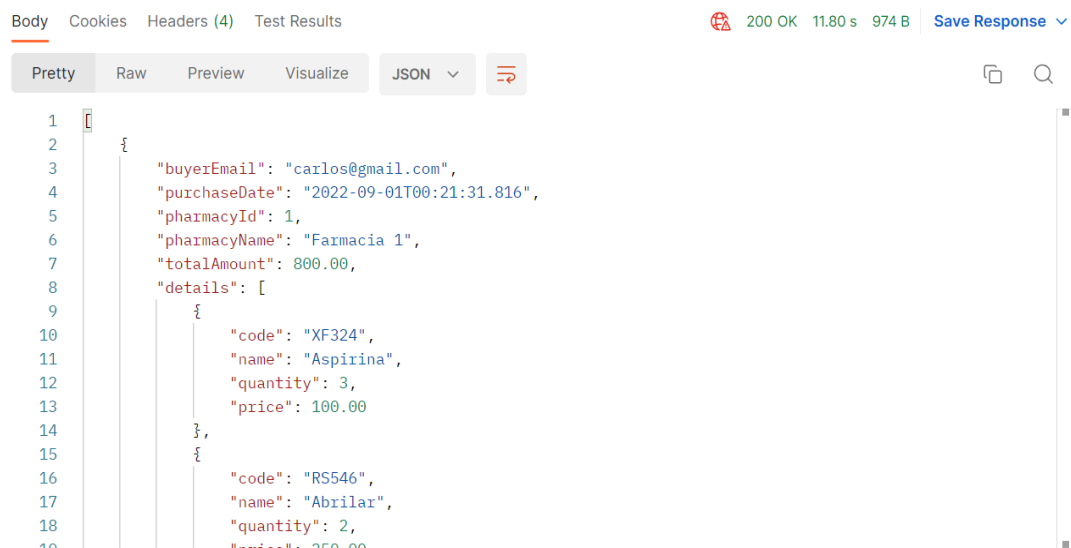
```
1 {
2   "message": "The drug to update does not exist."
3 }
```

- Status: 404 Not Found
- Body: Mensaje explicando que la droga que se quiere modificar no existe en la base de datos.

## Pruebas Dueño

### Listar compras por mes

- Ver todas las compras realizadas por usuarios anónimos en un mes dado - Ok 200
  - Retorna un json con la información de cada compra detallada realizada por usuarios anónimos.

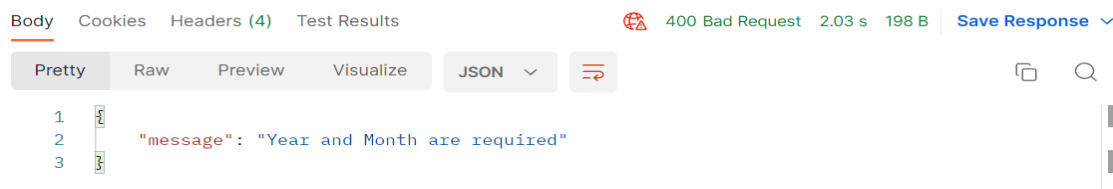


Body Cookies Headers (4) Test Results Status: 200 OK Time: 11.80 s Size: 974 B Save Response

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "buyerEmail": "carlos@gmail.com",
4     "purchaseDate": "2022-09-01T00:21:31.816",
5     "pharmacyId": 1,
6     "pharmacyName": "Farmacia 1",
7     "totalAmount": 800.00,
8     "details": [
9       {
10        "code": "XF324",
11        "name": "Aspirina",
12        "quantity": 3,
13        "price": 100.00
14      },
15      {
16        "code": "RS546",
17        "name": "Abrilar",
18        "quantity": 2,
19        "price": 250.00
20      }
21    ]
22  }
23 ]
```

- Error 400 - Mes o Año vacío



Body Cookies Headers (4) Test Results Status: 400 Bad Request Time: 2.03 s Size: 198 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Year and Month are required"
3 }
```

- Error 400 - Mes no válido.



Body Cookies Headers (4) Test Results Status: 400 Bad Request Time: 2.32 s Size: 228 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Invalid month 13: should be 'MM' (01-12) format"
3 }
```

- Error 400 - Año no válido.

```

1
2  "message": "Invalid year 20224: should be 'yyyy' (1900-2099) format"
3

```

### Ver solicitudes de reposición de stock

- Respuesta: 200 Ok
- Retorna mensaje JSON con la información de las solicitudes de stock de los empleados de la farmacia a la que pertenece el dueño.

```

1
2 {
3   "id": 2,
4   "status": "Approved",
5   "requestDate": "2022-10-04T02:13:31.241902",
6   "details": [
7     {
8       "drug": {
9         "code": "RS546",
10        "name": "Abrilar"
11      },
12      "quantity": 2
13    }
14  ],
15 },
16 {
17   "id": 3,
18   "status": "Pending",
19   "requestDate": "2022-10-04T02:16:14.934838",
20   "details": [
21     {
22       "drug": {
23         "code": "RS546",
24         "name": "Abrilar"
25       },
26       "quantity": 2
27     }
28  ],
29 }
30 ]

```

- Ver solicitudes de reposición de stock
- Respuesta: 403 Forbidden
- Retorna mensaje JSON indicando que el usuario no tiene permisos para ejecutar el request.

```

1
2  "message": "Forbidden role"
3

```

- Aprobar y rechazar solicitud de stock
- Respuesta: 200 Ok
- Retorna mensaje JSON indicando que la solicitud fue aprobada

```

1 true

```

### Aprobar y rechazar solicitud de stock rechazada

- Respuesta: 400 Bad Request
- Retorna mensaje JSON indicando que la solicitud ya fue rechazada.

Body Cookies Headers (4) Test Results Status: 400 Bad Request Time: 2.51 s Size: 202 B

Pretty Raw Preview Visualize JSON

```

1 {
2   "message": "Stock request already rejected."
3 }
    
```

Aprobar y rechazar solicitud de stock aprobada

- Respuesta: 400 Bad Request
- Retorna mensaje JSON indicando que la solicitud ya fue aprobada.

Body Cookies Headers (4) Test Results Status: 400 Bad Request Time: 5.80 s Size: 202 B

Pretty Raw Preview Visualize JSON

```

1 {
2   "message": "Stock request already approved."
3 }
    
```

## Pruebas Empleado

### Listar compras

- Ver todas las compras realizadas por usuarios anónimos - Ok 200
  - Retorna un json con la información de cada compra (sin los detalles) realizada por usuarios anónimos, retornando el email del comprador, la fecha de la compra, la farmacia donde se realizó la compra y el monto total de la misma.

Body Cookies Headers (4) Test Results Status: 200 OK 22.68 s 913 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   {
3     "buyerEmail": "marcelo@gmail.com",
4     "purchaseDate": "2022-08-22T22:21:31.816",
5     "pharmacyId": 1,
6     "pharmacyName": "Farmacia 1",
7     "totalAmount": 1350.00,
8     "details": []
9   },
10  {
11    "buyerEmail": "carlos@gmail.com",
12    "purchaseDate": "2022-09-01T00:21:31.816",
13    "pharmacyId": 1,
14    "pharmacyName": "Farmacia 1",
15    "totalAmount": 800.00,
16    "details": []
17  },
18  {
19    "buyerEmail": "roberto@gmail.com",
20    "purchaseDate": "2022-09-05T12:00:00",
21    "pharmacyId": 1,
    
```

### Crear solicitud de stock

- Respuesta: 200 Ok
- Recibe un mensaje JSON indicando que la solicitud fue creada.



Body Cookies Headers (4) Test Results Status: 200 OK Time: 281 ms Size: 164 B

Pretty Raw Preview Visualize JSON

```

1  {
2    "created": true
3  }
```

Crear solicitud de stock con un medicamento inválido

- Respuesta: 400 Bad Request
- Recibe un mensaje JSON indicando que la solicitud tiene un medicamento que no es válido.

Body Cookies Headers (4) Test Results Status: 400 Bad Request Time: 3.09 s Size: 202 B

Pretty Raw Preview Visualize JSON

```

1  {
2    "message": "Stock request has invalid drug."
3  }
```

Crear solicitud de stock sin medicamentos

- Respuesta: 400 Bad Request
- Retorna mensaje JSON indicando que la solicitud no tiene detalles de medicamentos.

Body Cookies Headers (4) Test Results Status: 400 Bad Request Time: 1951 ms Size: 193 B

Pretty Raw Preview Visualize JSON

```

1  {
2    "message": "Invalid stock details."
3  }
```

Ver solicitud de stock

- Respuesta: 200 Ok
- Retorna mensaje JSON con las solicitudes de stock realizadas por el usuario

Body Cookies Headers (4) Test Results Status: 200 OK Time: 3.35 s Size: 1.45 KB

Pretty Raw Preview Visualize JSON

```

1  [
2    {
3      "id": 2,
4      "status": "Approved",
5      "requestDate": "2022-10-04T02:13:31.241902",
6      "details": [
7        {
8          "drug": {
9            "code": "RS546",
10           "name": "Abrilar"
11         }
12       ]
13     }
14  ]
```

## Video con las pruebas

- Pruebas Postman | DA2/OBL1

<https://youtu.be/W606QQzu87I>