

API RESTFUL con validaciones

- General:
 - Campos vacíos.

■ Libro

Se intenta meter un libro sin autor

POST ▼ http://localhost:8090/libros

Params Authorization Headers (9) **Body** ● Scripts ● Settings

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

```

1 {
2   "isbn": 9785647238892,
3   "titulo": "La última pregunta"
4 }

```

Body Cookies Headers (4) Test Results (0/1) 🔄 **400 Bad Request**

{} JSON ▼ ▶ Preview 🔍 Visualize ▼

```

1 {
2   "timestamp": "2025-01-30T19:40:00.059+00:00",
3   "status": 400,
4   "error": "Bad Request",
5   "message": "Validation failed for object='libro'. Error count: 1",
6   "path": "/libros"
7 }

```

default message [Autor vacío]]

■ Ejemplar

Se intenta meter un ejemplar sin estado

POST ▼ http://localhost:8090/ejemplar Send

Params Authorization Headers (9) **Body** ● Scripts ● Settings Coo

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

```

1 {
2   "isbn": {
3     "isbn": "9789876543210",
4     "titulo": "Cien Años de Soledad",
5     "autor": "Gabriel Garcia Márquez"
6   }
7 }

```

Body Cookies Headers (4) Test Results (0/1) 🔄 **500 Internal Server Error** - 11 ms - 643 B - 🔍 Save Response

{} JSON ▼ ▶ Preview 🔍 Visualize ▼ 🔍

```

1 {
2   "timestamp": "2025-01-30T22:12:23.883+00:00",
3   "status": 500,
4   "error": "Internal Server Error",
5   "message": "Validation failed for classes [org.example.miprimeraapi.restful.DAOS.Ejemplar] during persist time for groups [jakarta.validation.groups.Default, ]\nList of constraint violations: [\n\tConstraintViolationImpl{interpolatedMessage='Estado vacío', propertyPath=Estado, rootBeanClass=class org.example.miprimeraapi.restful.DAOS.Ejemplar, messageTemplate='Estado vacío'}\n]",
6   "path": "/ejemplar"
7 }

```

■ Préstamo

Se intenta meter un préstamo sin ejemplar

POST http://localhost:8090/prestamos

Params Authorization Headers (9) Body ● Scripts Settings

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

```
1 {
2   "usuario": {
3     "id": 4,
4     "dni": "54732311Q",
5     "nombre": "Unai",
6     "email": "unainieto2004@gmail.com",
7     "password": "unai123",
8     "tipo": "administrador",
9     "penalizacionHasta": null
10  }
11 }
```

Body Cookies Headers (4) Test Results 500 Internal Server Error 316 ms 662 B

{ } JSON Preview Visualize

```
for groups [jakarta.validation.groups.Default, ]\nList of constraint violations:
[\n\tConstraintViolationImpl{interpolatedMessage='ID de ejemplar vacio', propertyPath=ejemplar,
rootBeanClass=class org.example.miprimeraapirestful.DAOS.Prestamo, messageTemplate='ID de ejemplar vacio'}
\n],
{
  "path": "/prestamos"
}
```

■ Usuario

Se intenta meter un usuario sin nombre

POST http://localhost:8090/usuario

Params Authorization Headers (9) Body ● Scripts Settings

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

```
1 {
2   "dni": "02345689T",
3   "email": "unai@gmail.com",
4   "password": "unai123",
5   "tipo": "normal"
6 }
```

Body Cookies Headers (4) Test Results

{ } JSON Preview Visualize

```
1 {
2   "timestamp": "2025-01-30T22:29:28.634+00:00",
3   "status": 400,
4   "error": "Bad Request",
5   "message": "Validation failed for object='usuario'. Error count: 2",
6   "path": "/usuario"
7 }
```

```
default message [Nombre de usuario vacio] ]
```

- Campos null

- Libro

Se intenta añadir un libro con autor null

POST http://localhost:8090/libros

Params Authorization Headers (9) Body Scripts Settings

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

```
1 {
2   "isbn": 9785647238892,
3   "titulo": "La última pregunta",
4   "autor": null
5 }
```

Body Cookies Headers (4) Test Results (0/1) 400 Bad Request

{ } JSON Preview Visualize

```
1 {
2   "timestamp": "2025-01-30T22:03:57.859+00:00",
3   "status": 400,
4   "error": "Bad Request",
5   "message": "Validation failed for object='libro'. Error count: 1",
6   "path": "/libros"
7 }
```

default message [Autor vacío]]

- Ejemplar

Se intenta meter un ejemplar con estado null

POST http://localhost:8090/ejemplar Send

Params Authorization Headers (9) Body Scripts Settings Cool

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

```
1 {
2   "isbn": {
3     "isbn": "9789876543210",
4     "titulo": "Cien Años de Soledad",
5     "autor": "Gabriel García Márquez"
6   },
7   "estado": null
8 }
```

Body Cookies Headers (4) Test Results (0/1) 500 Internal Server Error 53 ms 643 B Save Response

{ } JSON Preview Visualize

```
1 {
2   "timestamp": "2025-01-30T22:07:23.129+00:00",
3   "status": 500,
4   "error": "Internal Server Error",
5   "message": "Validation failed for classes [org.example.miprimeraapirestful.DAOS.Ejemplar] during persist time for groups [jakarta.validation.groups.Default, ]\nList of constraint violations: [\n\tConstraintViolationImpl{interpolatedMessage='Estado vacío', propertyPath=Estado, rootBeanClass=class org.example.miprimeraapirestful.DAOS.Ejemplar, messageTemplate='Estado vacío'}\n]",
6   "path": "/ejemplar"
7 }
```

■ Préstamo

Se intenta meter un préstamo con un ejemplar null

POST http://localhost:8090/prestamos

Params Authorization Headers (9) Body Scripts Settings

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

```
1 {
2   "usuario": {
3     "id": 4,
4     "dni": "54732311Q",
5     "nombre": "Unai",
6     "email": "unaieto2004@gmail.com",
7     "password": "unai123",
8     "tipo": "administrador",
9     "penalizacionHasta": null
10  },
11  "ejemplar": null
12 }
```

Body Cookies Headers (4) Test Results 500 Internal Server Error 12 ms 662 B

JSON Preview Visualize

```
5 {
6   "message": "Validation failed for classes [org.example.miprimeraapi.restful.DAOS.Prestamo] during persist time
  for groups [jakarta.validation.groups.Default, ]\nList of constraint violations:
  [\n\tConstraintViolationImpl{interpolatedMessage='ID de ejemplar vacio', propertyPath=ejemplar,
  rootBeanClass=class org.example.miprimeraapi.restful.DAOS.Prestamo, messageTemplate='ID de ejemplar vacio'}
  \n]",
  "path": "/prestamos"
}
```

■ Usuario

Se intenta añadir un usuario con contraseña null

POST http://localhost:8090/usuario

Params Authorization Headers (9) Body Scripts Settings

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

```
1 {
2   "dni": "02345689T",
3   "nombre": "Pepe",
4   "email": "pepito247@gmail.com",
5   "password": null,
6   "tipo": "normal"
7 }
```

Body Cookies Headers (4) Test Results

JSON Preview Visualize

```
1 {
2   "timestamp": "2025-01-30T22:37:00.636+00:00",
3   "status": 400,
4   "error": "Bad Request",
5   "message": "Validation failed for object='usuario'. Error count: 2",
6   "path": "/usuario"
7 }
```

- Libros:
 - ISBN: formato correcto, ejemplo ISBN-13: 978-0-596-52068-7.

Se intenta añadir un libro con un isbn no valido

POST http://localhost:8090/libros

Params Authorization Headers (9) Body Scripts Settings

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

```

1 {
2   "isbn": 348832,
3   "titulo": "La última pregunta",
4   "autor": "Isaac Asimov"
5 }

```

Body Cookies Headers (4) Test Results (0/1) 400 Bad Request

{ } JSON Preview Visualize

```

1 {
2   "timestamp": "2025-01-30T19:36:19.573+00:00",
3   "status": 400,
4   "error": "Bad Request",
5   "message": "Validation failed for object='libro'. Error count: 2",
6   "path": "/libros"
7 }

```

default message [El ISBN debe comenzar con 978 y seguir con 10 dígitos]]]

- Título y autor: solo caracteres alfanuméricos de longitud máxima del campo de la base de datos.

Se intenta meter un libro el cual contiene _ y ´ en su nombre

POST http://localhost:8090/libros

Params Authorization Headers (9) Body Scripts Settings

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

```

1 {
2   "isbn": 9785647238892,
3   "titulo": "La última pregunta_´´´´",
4   "autor": "Isaac Asimov"
5 }

```

Body Cookies Headers (4) Test Results (0/1) 400 Bad Request

{ } JSON Preview Visualize

```

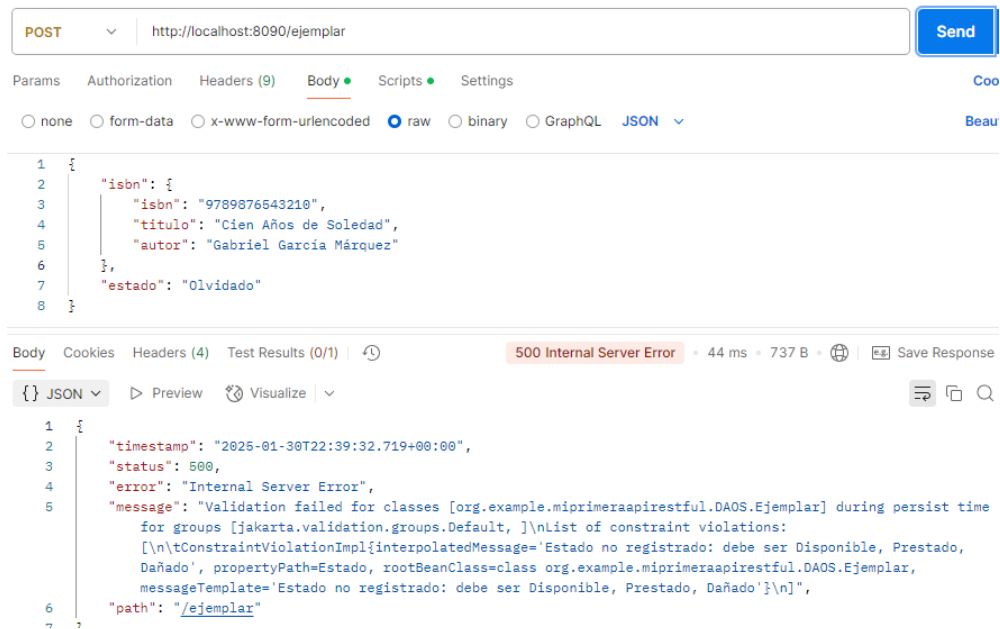
1 {
2   "timestamp": "2025-01-30T22:32:23.875+00:00",
3   "status": 400,
4   "error": "Bad Request",
5   "message": "Validation failed for object='libro'. Error count: 1",
6   "path": "/libros"
7 }

```

- Ejemplares:

- Estado: solo puede contener uno de los siguientes valores: disponible, prestado, dañado.

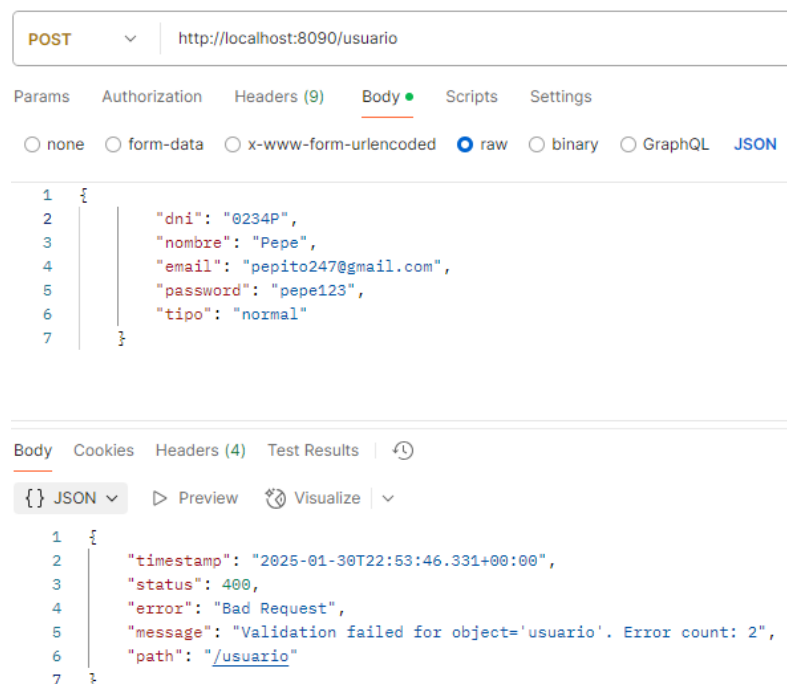
Se intenta añadir un ejemplar en un estado no registrado, te lo pone en el error junto a los estados disponibles



- Usuarios:

- DNI: validar mediante método específico.

Se intenta añadir un usuario con un DNI inválido.



default message [El DNI debe contener 8 números seguidos de una letra mayúscula.]]]

- Nombre: solo caracteres alfanuméricos de longitud máxima del campo de la base de datos

Se intenta añadir un usuario con una `_` en su nombre

```
POST http://localhost:8090/usuario

{
  "dni": "02345609T",
  "nombre": "Pedro_Sánchez",
  "email": "elpresi2025@gmail.com",
  "password": "pedrito123",
  "tipo": "normal"
}
```

```
{
  "timestamp": "2025-01-30T22:56:22.611+00:00",
  "status": 400,
  "error": "Bad Request",
  "message": "Validation failed for object='usuario'. Error count: 2",
  "path": "/usuario"
}
```

```
default message [El nombre solo puede contener caracteres alfanuméricos]] ]
```

- Tipo: solo puede contener uno de los siguientes valores: normal, administrador.

Se intenta añadir un usuario de un tipo no válido

```
POST http://localhost:8090/usuario

{
  "dni": "02345609T",
  "nombre": "Pedro_Sánchez",
  "email": "elpresi2025@gmail.com",
  "password": "pedrito123",
  "tipo": "presidente"
}
```

```
{
  "timestamp": "2025-01-30T22:57:55.762+00:00",
  "status": 400,
  "error": "Bad Request",
  "message": "Validation failed for object='usuario'. Error count: 2",
  "path": "/usuario"
}
```

```
default message [Tipo de usuario no reconocido]] ]
```

- Email: Solo emails que acaben en un determinado proveedor de correo, por ejemplo gmail. Ejemplo de Pattern - ([A-Za-z0-9]{1,50})@gmail.com)

Se intenta añadir un usuario con un correo que no es @gmail.com

```

1 {
2   "dni": "02345609T",
3   "nombre": "Pedro_Sánchez",
4   "email": "elpresi2025@yahoo.com",
5   "password": "pedrito123",
6   "tipo": "normal"
7 }

```

```

1 {
2   "timestamp": "2025-01-30T22:58:57.590+00:00",
3   "status": 400,
4   "error": "Bad Request",
5   "message": "Validation failed for object='usuario'. Error count: 3",
6   "path": "/usuario"
7 }

```

default message [El correo debe ser de tipo @gmail.com.]

- Password: Cadena alfanumérica de entre 4 y 12 caracteres.

```

1 {
2   "dni": "02345609T",
3   "nombre": "Pedro_Sánchez",
4   "email": "elpresi2025@gmail.com",
5   "password": "pedritoelpresi123456789",
6   "tipo": "normal"
7 }

```

```

1 {
2   "timestamp": "2025-01-30T23:01:19.281+00:00",
3   "status": 400,
4   "error": "Bad Request",
5   "message": "Validation failed for object='usuario'. Error count: 1",
6   "path": "/usuario"
7 }

```

default message [La contraseña debe tener entre 4 y 12 caracteres]

- Penalizacion Hasta: por ejemplo aquí sí que admitiría nulos.