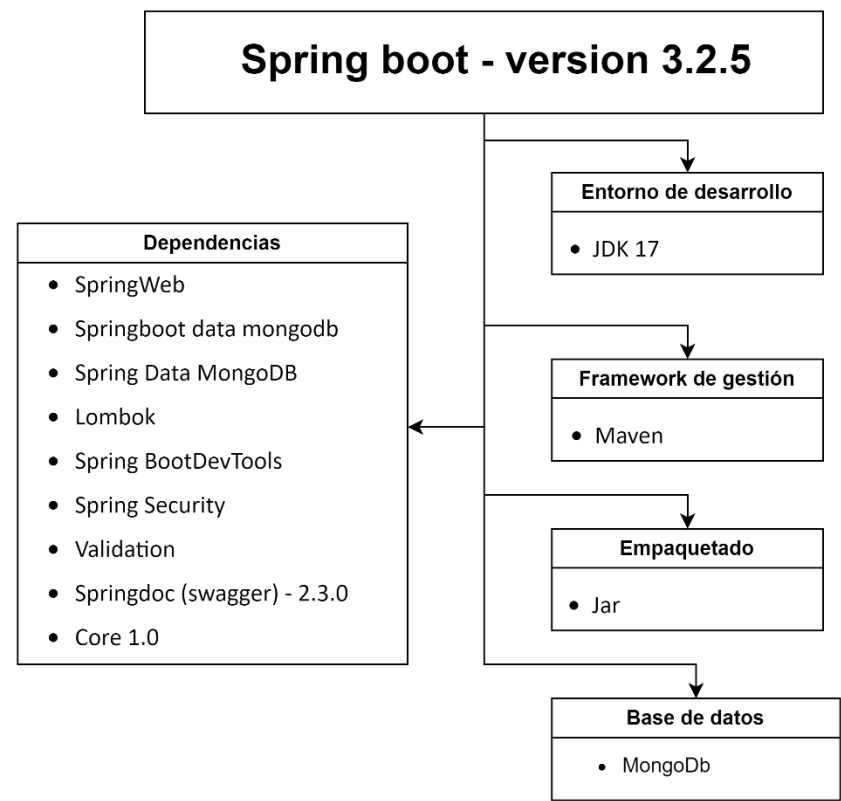


USER-APP

Tecnología utilizada



Estructura

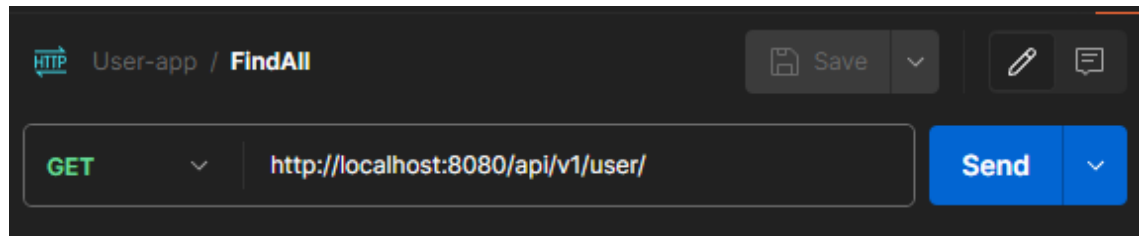
Atributos	Metodos
<ul style="list-style-type: none">• id:string• name:string• email:string• city:string• password:string• rolUser:string• clientId:string• clientSecret:string• createdDate:Data• lastModifiedDate:Data	<ul style="list-style-type: none">• createdUser()• findAllUser()• getUserById• updateUserById()• deleteUserById()• findAllRoles()

CONSUMO DEL MICROSERVICIO

OBTENER TODOS LOS USUARIOS

Petición:

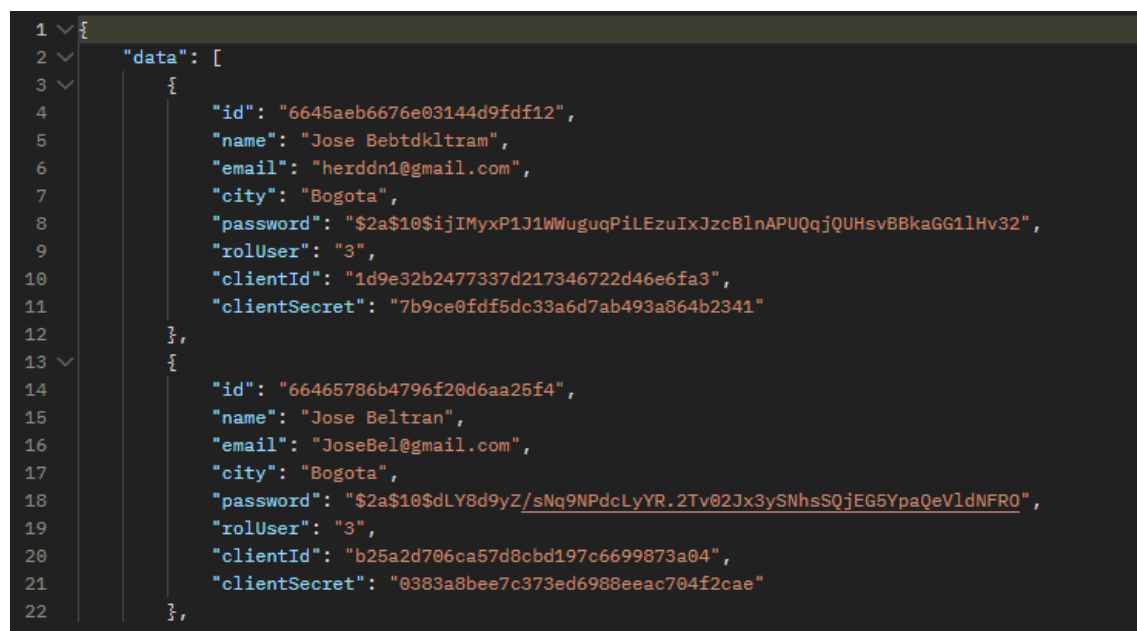
Utilizamos un método **GET** y la ruta: [*Tipo de enlace*] [*Dominio*] [*api/v1/user/*]



Respuesta exitosa:

Estado 200

En caso de ser exitoso obtenemos una respuesta con todos los usuarios existentes.



Errores:

Estado 404

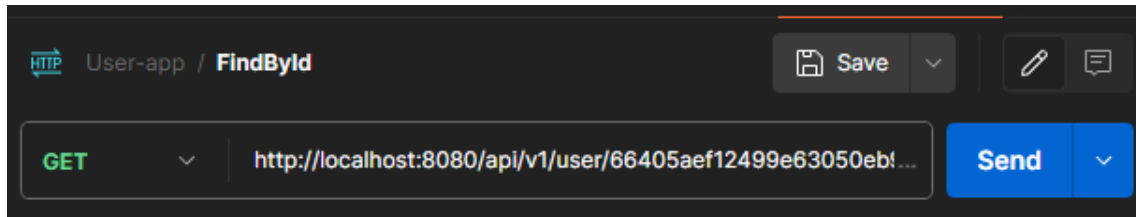
En caso de que la colección no tenga registros:



BUSCAR POR ID DEL USUARIO

Petición:

Utilizamos un método **GET** y la ruta: [*Tipo de enlace*] [*Dominio*] [*api/v1/user/*] [*userId*]



Respuesta exitosa:

Estado 200

En caso de ser exitoso obtenemos una respuesta con los datos el usuario existente (Excluyendo la contraseña).

```
1  {
2    "data": {
3      "id": "66465d5cb4796f20d6aa25f8",
4      "name": "Carlos Beltran",
5      "email": "CarlosBelt@gmail.com",
6      "city": "Bogota",
7      "password": "$2a$10$TI8aTtRXr8izh3zRLLYA8uzPtYje3Boqvsj6P0S0JsQNq2TQnTFpK",
8      "rolUser": "2",
9      "clientId": "3f5b955826433192a69c5265012fe3f8",
10     "clientSecret": "e4e806a2ef934db1611f62b0abdddb6b"
11   },
12   "responseDetails": {
13     "code": "200",
14     "message": "La solicitud ha tenido éxito",
15     "timestamp": "2024-05-16T14:24:37.7920308"
16   }
17 }
```

Errores:

Estado 404

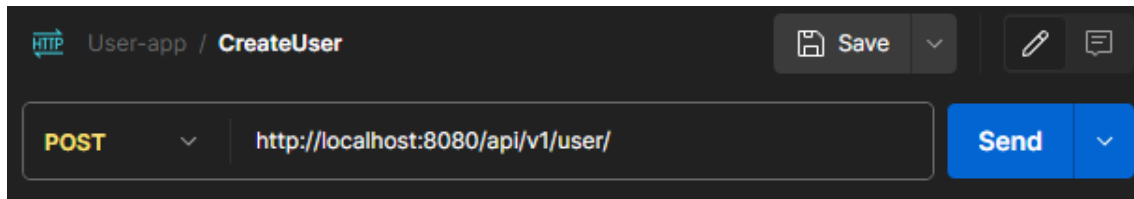
En caso de que el **usuario** no exista:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "404",
5      "message": "Este usuario no existe.",
6      "timestamp": "2024-05-12T10:08:21.4085542"
7    }
8  }
```

CREAR NUEVO USUARIO

Petición:

Utilizamos un método **POST** y la ruta: [*Tipo de enlace*] [*Dominio*] [*api/v1/user/*]



Cuerpo de la petición:

```
1  {
2    "name": "Jose Beltran",
3    "email": "JoseBel@gmail.com",
4    "city": "Bogota",
5    "password": "12345e",
6    "rolUser": "3"
7  }
```

Respuesta exitosa:

Estado 200

En caso de ser exitoso obtenemos una respuesta con los datos el usuario recién creado.

```
1  {
2    "data": {
3      "id": "66465786b4796f20d6aa25f4",
4      "name": "Jose Beltran",
5      "email": "JoseBel@gmail.com",
6      "city": "Bogota",
7      "password": "$2a$10$dLY8d9yZ/sNq9NPdcLyYR.
8                  2Tv02Jx3ySNhsSQjEG5YpaQeVldNFR0",
9      "rolUser": "3",
10     "clientId": "b25a2d706ca57d8cbd197c6699873a04",
11     "clientSecret": "0383a8bee7c373ed6988eeac704f2cae"
12   },
13   "responseDetails": {
14     "code": "200",
15     "message": "El usuario se creo exitosamente.",
16     "timestamp": "2024-05-16T13:59:18.1126729"
17   }
18 }
```

Errores:

Estado 400

El valor **id** de usuario no debe ser parte del cuerpo de la petición:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "400",
5      "message": "Error de esquema en el mensaje XML",
6      "timestamp": "2024-05-12T11:48:00.7504096"
7    }
8  }
```

El **correo electrónico** no debe estar vinculado a otro usuario existente:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "400",
5      "message": "El correo electronico: 'JoseBel@gmail.com' ya esta
6                  registrado.",
7      "timestamp": "2024-05-16T14:02:14.8272256"
8    }
9  }
```

El **correo electrónico** debe tener un formato correcto:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "400",
5      "message": "debe ser una dirección de correo electrónico con
6                  formato correcto",
7      "timestamp": "2024-05-12T11:51:25.7982724"
8    }
9  }
```

El **correo electrónico** no puede ser nulo o vacío:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "400",
5      "message": "El correo electronico es requerido.",
6      "timestamp": "2024-05-18T23:25:16.2912351"
7    }
8  }
```

El **rol id** debe ser valido, 1 para administradores, 2 para empleados y 3 para usuarios:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "400",
5      "message": "Este rol de usuario no es valido",
6      "timestamp": "2024-05-16T14:06:19.8875917"
7    }
8  }
```

En **rol Id** debe de enviarse siempre:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "400",
5      "message": "Especifique el rol del usuario.",
6      "timestamp": "2024-05-18T23:31:52.6782668"
7    }
8  }
```

La **contraseña** no debe de ser vacía o nula:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "400",
5      "message": "Debe especificar un valor para la contraseña.",
6      "timestamp": "2024-05-18T23:45:05.4554279"
7    }
8  }
```

Malas peticiones:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "474",
5      "message": "Error en el servidor, contacte con un administrador.",
6      "timestamp": "2024-05-18T23:52:03.1006171"
7    }
8  }
```

ACTUALIZAR USUARIO

Petición:

Utilizamos un método **PUT** y la ruta: **[Tipo de enlace] [Dominio] [api/v1/user/]**

PUT	▼	http://localhost:8080/api/v1/user/	Send	▼
------------	---	---	-------------	---

Además de enviar el cuerpo de la petición:

```
1  {
2      "id": "66465d5cb4796f20d6aa25f8",
3      "name": "Carlos Beltran",
4      "email": "CarlosBern@gmail.com",
5      "city": "Bogota",
6      "password": "45dfsdf6",
7      "rolUser": "2",
8      "clientId": "3f5b955826433192a69c5265012fe3f8",
9      "clientSecret": "e4e806a2ef934db1611f62b0abdddb6b"
10 }
```

Respuesta exitosa:

Estado 200

En caso de ser exitoso obtenemos una respuesta con los datos el usuario recién actualizado (Excluyendo la contraseña).

```
1  {
2      "data": {
3          "id": "66465d5cb4796f20d6aa25f8",
4          "name": "Carlos Beltran",
5          "email": "CarlosBern@gmail.com",
6          "city": "Bogota",
7          "password": "$2a$10$TI8aTtRXr8izh3zRLLYA8uzPtYje3Boqvsj6P0S0JsQNq2TQnTFpK",
8          "rolUser": "2",
9          "clientId": "3f5b955826433192a69c5265012fe3f8",
10         "clientSecret": "e4e806a2ef934db1611f62b0abdddb6b"
11     },
12     "responseDetails": {
13         "code": "200",
14         "message": "El usuario se actualizo exitosamente.",
15         "timestamp": "2024-05-16T14:29:05.0414258"
16     }
17 }
```

Errores:

Estado 404

El id del usuario debe corresponder a un usuario existente:

```
1  {
2      "data": null,
3      "responseDetails": {
4          "code": "404",
5          "message": "Este usuario no existe.",
6          "timestamp": "2024-05-12T16:44:56.9624299"
7      }
8  }
```

Estado 400

El **correo electrónico** no puede estar registrado con otro usuario:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "400",
5      "message": "El correo electronico: 'Josepo@gmail.com' ya esta
6                registrado.",
7      "timestamp": "2024-05-12T11:45:52.6267581"
8    }
9  }
```

El **correo electrónico** debe tener un formato correcto:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "400",
5      "message": "debe ser una dirección de correo electrónico con
6                formato correcto",
7      "timestamp": "2024-05-12T11:51:25.7982724"
8    }
9  }
```

El **correo electrónico** no debe de ser nulo o vacío:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "400",
5      "message": "El valor del correo electronico no puede ser vacio o nulo.",
6      "timestamp": "2024-05-16T14:36:14.2648281"
7    }
8  }
```

El **clientId** y **ClientSecret** no pueden ser los mismos de otro usuario:

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "400",
5      "message": "Este usuario ya existe.",
6      "timestamp": "2024-05-12T11:46:45.9868264"
7    }
8  }
```


El **id** del usuario es obligatorio en el cuerpo de la petición:

```
1 {
2   "data": null,
3   "responseDetails": {
4     "code": "400",
5     "message": "Error de esquema en el mensaje XML",
6     "timestamp": "2024-05-12T11:48:00.7504096"
7   }
8 }
```

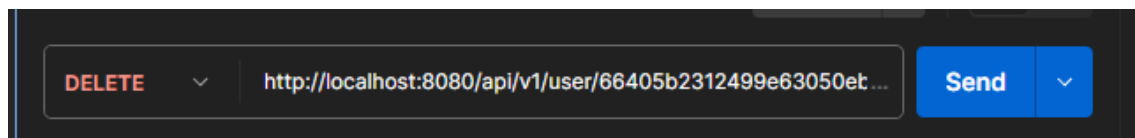
El **clientId** y **clientSecret** no deben ser nulos o vacíos.

```
1 {
2   "data": null,
3   "responseDetails": {
4     "code": "400",
5     "message": "Error de esquema en el mensaje XML",
6     "timestamp": "2024-05-12T15:37:02.5225488"
7   }
8 }
```

ELIMINAR USUARIO

Petición:

Utilizamos un método **DELETE** y la ruta: **[Tipo de enlace] [Dominio] [api/v1/user/][userId]**



Respuesta exitosa:

Estado 200

En caso de ser exitoso obtenemos una respuesta con los datos el usuario recién eliminado (Excluyendo la contraseña).

```
1 {
2   "data": {
3     "id": "66465d5cb4796f20d6aa25f8",
4     "name": "Carlos Beltran",
5     "email": "CarlosBern@gmail.com",
6     "city": "Bogota",
7     "password": "$2a$10$TI8aTtRXr8izh3zRLLYA8uzPtYje3Boqvsj6P0S0JsQNq2TQnTFpK",
8     "rolUser": "2",
9     "clientId": "3f5b955826433192a69c5265012fe3f8",
10    "clientSecret": "e4e806a2ef934db1611f62b0abdddb6b"
11  },
12  "responseDetails": {
13    "code": "200",
14    "message": "El usuario se elimino exitosamente.",
15    "timestamp": "2024-05-16T14:44:10.7172436"
16  }
17 }
```

Errores:

Estado 400

En caso que el usuario no existe.

```
1  {
2    "data": null,
3    "responseDetails": {
4      "code": "404",
5      "message": "Este usuario no existe.",
6      "timestamp": "2024-05-12T17:03:28.1923571"
7    }
8  }
```