

Documentación Web API BIBLIOTECA

1. LIBROS

Para obtener listado de libros ordenados de forma ascendente por autor.

REQUEST

GET /api/libros?autor=Stephen+King&orden=ascendente
Host: api.biblioteca.com

RESPONSE

HTTP/1.1 200 OK
Content-Type: application/json

```
{
  "libros": [
    {
      "id": 2,
      "titulo": "Carrie",
      "autor": "Stephen King",
      "anio_publicacion": 1974
    },
    {
      "id": 1,
      "titulo": "El resplandor",
      "autor": "Stephen King",
      "anio_publicacion": 1977
    },
    {
      "id": 3,
      "titulo": "It",
      "autor": "Stephen King",
      "anio_publicacion": 1986
    }
  ]
}
```

Los libros del autor "Stephen King" se presentan en orden ascendente por título.

RESPONSE (ERROR LIBRO NO ENCONTRADO)

HTTP/1.1 404 Not Found
Content-Type: application/json

```
{  
  "error": "No se encontraron libros del autor 'Stephen King'.",  
  "codigo_error": "LIBROS_NO_ENCONTRADOS"  
}
```

En caso de que no se encuentren libros del autor especificado, la API responde con un código de estado HTTP 404 Not Found y un mensaje de error informativo.

2. USUARIOS

Para agregar un nuevo usuario a la biblioteca.

REQUEST:

POST /api/usuarios
Host: api.biblioteca.com
Content-Type: application/json

```
{  
  "nombre": "Maria Rodriguez",  
  "email": "maria@example.com",  
  "telefono": "555-555-5555",  
  "direccion": "Calle Principal, 123",  
  "ciudad": "Ciudad Ejemplo",  
  "codigo_postal": "12345"  
}
```

RESPONSE (ÉXITO)

HTTP/1.1 201 Created
Content-Type: application/json

```
{  
  "id": 101,  
  "mensaje": "El usuario Maria Rodriguez ha sido agregado exitosamente."  
}
```

El servidor responde con un código de estado HTTP 201 Created para indicar que el recurso (el nuevo usuario) ha sido creado con éxito.

RESPONSE (ERROR DE VALIDACIÓN)

HTTP/1.1 400 Bad Request
Content-Type: application/json

```
{
  "error": "Error de validación",
  "detalles": [
    "El campo 'email' es requerido.",
    "El campo 'telefono' debe ser un número válido."
  ]
}
```

La API responde con un código de estado HTTP 400 Bad Request y un mensaje de error que proporciona detalles específicos sobre los problemas de validación. En este ejemplo, se indica que el campo "email" es requerido y que el campo "telefono" debe ser un número válido.

3. PRÉSTAMOS

Para obtener un listado de préstamos de libros activos de un usuario específico con ID 101.

REQUEST

GET /api/usuarios/101/prestamos/activos
Host: api.biblioteca.com

RESPONSE

HTTP/1.1 200 OK
Content-Type: application/json

```
{
  "prestamos_activos_usuario": [
    {
      "id": 1,
      "libro_id": 5,
      "fecha_prestamo": "2023-09-10",
      "fecha_devolucion_prevista": "2023-09-24"
    },
    {
```

```
    "id": 4,  
    "libro_id": 8,  
    "fecha_prestamo": "2023-09-15",  
    "fecha_devolucion_prevista": "2023-09-29"  
  },  
],  
"informacion_usuario": {  
  "id": 101,  
  "nombre": "Juan Pérez",  
  "email": "juan@example.com",  
  "telefono": "555-123-4567"  
}
```

El servidor responde con un código de estado HTTP 200 OK para indicar que la solicitud se ha procesado correctamente. El contenido de la respuesta está en formato JSON y contiene dos partes principales:

1. "usuario": Esta sección proporciona información sobre el usuario con el ID 101, como su nombre, email y número de teléfono.
2. "prestamos_activos": Esta sección contiene una lista de los préstamos de libros activos para el usuario. Cada préstamo incluye información sobre el libro prestado, como el título y el autor, así como las fechas de préstamo y devolución prevista.

RESPONSE (ERROR ID NO ENCONTRADO)

HTTP/1.1 404 Not Found
Content-Type: application/json

```
{  
  "error": "El usuario con ID 101 no fue encontrado.",  
  "codigo_error": "USUARIO_NO_ENCONTRADO"  
}
```

El servidor responde con un código de estado HTTP 404 Not Found para indicar que el recurso solicitado (el usuario con ID 101) no se encontró.