

Readora

Nombre: Javier Moreno Salas
Curso: 2º DAW

Curso Académico: 2024-2025
Ciclo Formativo: Desarrollo de
Aplicaciones Web



Índice

Índice.....	1
Introducción.....	2
Identificación de las necesidades del proyecto.....	4
Breve análisis/comparativa con las alternativas del mercado.....	5
Justificación del proyecto.....	6
Uso de stack tecnológico.....	6
Esquema de la BD.....	10
Prototipo en Figma.....	13
Diseño Final.....	18
Definición API REST.....	25
Manual de despliegue.....	101
Contexto del despliegue.....	101
Pasos de la creación del despliegue.....	103
Pasos para desplegar.....	129
Muestra de Diagrama de Despliegue.....	129
Conclusiones y Postmortem.....	130
Anexo.....	131



Introducción

La idea principal del proyecto sería hacer una web la cual unifique tu universo de lectura, es decir, un espacio donde puedas almacenar todos los libros que has leído y llevar un control de tu lectura de una forma fácil y accesible.

The screenshot shows the homepage of the Readora website. At the top, there is a navigation bar with icons for 'Inicio', 'Buscador', 'Mi Biblioteca', 'user', and a profile icon. Below the navigation bar, there is a large blue rectangular area with rounded corners. Inside this area, the text 'Tu biblioteca personal en Readora' is displayed in bold black font. Below this, a smaller text reads: 'Gestiona tus lecturas, descubre nuevos títulos y lleva un registro de tu viaje literario de manera simple y elegante.' At the bottom of this blue area, there are two buttons: 'Crear cuenta' and 'Iniciar sesión'.

Tu biblioteca personal en Readora

Gestiona tus lecturas, descubre nuevos títulos y lleva un registro de tu viaje literario de manera simple y elegante.

[Crear cuenta](#) [Iniciar sesión](#)

Lo que te ofrece Readora

¿De dónde sale esta idea?

La motivación surge porque, después de haber navegado mucho en la búsqueda de libros por internet, la experiencia no ha sido nada cómoda. ¿Por qué no existe una web en la que un lector pueda gestionar todo su historial de lectura de forma sencilla? Esta pregunta es la que impulsa la creación de esta plataforma, cuyo principal objetivo es facilitar la gestión y el control de los libros leídos por cada usuario.

¿Por qué es necesario este proyecto?

Actualmente, organizar y llevar un registro de los libros leídos y por leer resulta complicado. ¿Cuántos lectores saben exactamente qué han leído en los últimos años o qué títulos les quedan por leer? Muchos, seguramente, no encuentran una manera sencilla y práctica de llevar ese control.



¿Qué carencias tienen las plataformas actuales?

Otra de las cuestiones que surgen es: ¿Por qué las plataformas actuales no son cómodas ni están realmente centradas en el usuario? En la mayoría de los casos, estas herramientas son poco flexibles, difíciles de usar o carecen de funciones personalizables, obligando al usuario a adaptarse a sistemas que no se ajustan a sus necesidades.

¿Existe un lugar donde consultar toda la información relevante sobre la lectura personal?

En resumen: no. No hay un espacio centralizado y práctico donde los usuarios puedan consultar fácilmente sus libros, progresos y notas personales sobre lecturas de una manera sencilla y accesible.



Identificación de las necesidades del proyecto

- **Rol sin Registro**

RF01 Un usuario sin registrar podrá registrarse gracias a un formulario (Registrado y Admin).

RF02 Un usuario sin registrar podrá iniciar sesión a través de un formulario

- **Rol Registrado**

RF04 Un usuario registrado podrá guardar libros como leídos y opinar sobre ellos en una sección de la web.

RF05 Un usuario registrado podrá llevar un control de los libros que está leyendo actualmente.

RF03 El usuario registrado podrá buscar libros, a través de un buscador

- **Rol Admin**

RF07 Un usuario administrador podrá crear, modificar y eliminar libros y autores desde un panel de administración.

RF08 Un usuario administrador podrá modificar la información de los usuarios desde un panel de administración

- **Funcionalidades Objetivo de la web**

RF Un administrador podrá crear y modificar libros y autores dentro del “Panel de Administración” (ADMIN)

RF Un usuario registrado podrá buscar libros utilizando el buscador (REGISTRADO)

RF Un usuario registrado podrá almacenar libros leídos, opinar y llevar un control de su lectura (REGISTRADO)



Breve análisis/comparativa con las alternativas del mercado

A la hora de plantear este proyecto, es importante preguntarse:

¿Existen ya webs o aplicaciones que solucionen este mismo problema?

En la actualidad, sí hay algunas plataformas dedicadas a la gestión de libros y el control de lecturas, pero después de usarlas y compararlas, se pueden observar varias limitaciones relevantes.

Por ejemplo, existen sitios conocidos como [Goodreads](#) o incluso apps para móviles que permiten llevar un registro de libros. Sin embargo, ¿son realmente cómodas y adaptadas a lo que busca un usuario?

Muchas veces, estas plataformas resultan poco intuitivas o están sobrecargadas de funciones secundarias que dificultan la experiencia. Al estar pensadas para un público muy general, también suelen carecer de opciones sencillas para personalizar la organización de los libros o llevar notas personales de forma práctica.

Además, algunas de estas alternativas requieren métodos de registro poco accesibles, muestran demasiada publicidad o incluso limitan el acceso a ciertas funciones si no eres usuario premium. Todo esto termina alejando a quienes, como yo, buscan simplemente una herramienta clara y centrada en el usuario.

Por eso, la propuesta de este proyecto se diferencia en la simplicidad, la facilidad de uso y la centralización total de la experiencia de lectura. El objetivo es mejorar lo que no ofrecen las alternativas actuales: un espacio unificado y realmente cómodo, donde organizar, revisar y disfrutar del seguimiento de la lectura sea sencillo para cualquier persona.

Alternativas:



[Goodreads](#)



Justificación del proyecto

¿Por qué considero que este proyecto es necesario?

La principal razón es que, después de buscar soluciones en internet durante mucho tiempo, he visto que ninguna plataforma se adapta realmente a lo que necesito como lector.

Esta búsqueda constante me ha llevado a probar decenas de herramientas: aplicaciones móviles, webs complejas, hojas de cálculo, incluso cuadernos físicos... **Ninguna funciona como debería**. Siempre termino igual: frustrado porque lo que encuentro **no encaja con mi rutina**, no entiende mis hábitos de lectura ni respeta mi necesidad de simplicidad.

Las plataformas que existen **fallan en lo esencial**:

- **No se adaptan** a cómo yo organizo mis libros (leídos, en pausa, pendientes).
- **No respetan** mi forma de tomar notas rápidas o registrar progresos sin burocracia.
- **No ofrecen** esa claridad inmediata que busco al abrir la aplicación.

Es como si todas estuvieran diseñadas para otro tipo de lector – alguien que quiere estadísticas complejas, redes sociales literarias o desafíos de lectura. Pero **¿Y lo que YO necesito?** Un lugar limpio, sin ruido, donde *mis libros* sean los protagonistas. Donde no tenga que pelearme con menús escondidos o funciones que nunca usaré.

Por eso insisto: **ninguna se adapta REALMENTE**. No es que les falten funciones, es que les sobra todo lo que estorba. Y esa es la necesidad concreta que este proyecto viene a cubrir: **un espacio digital hecho a la medida del lector real, no del algoritmo**.

Uso de stack tecnológico

Front-end

El front-end está construido con Angular (versión 19). Junto a Angular, utilizo Tailwind



CSS y Bootstrap, ya que ambos me permiten crear interfaces atractivas, rápidas y responsive. Todo esto se apoya en los lenguajes esenciales: HTML, CSS y TypeScript.

En Resumen:



Angular v19



Tailwind



Bootstrap



HTML



CSS



TypeScript

Back-end

Para el back-end he usado Spring Boot con Java 21, montando una API REST. La



razón principal de elegir Spring Boot es la experiencia adquirida a lo largo del grado. La base de datos seleccionada es MariaDB, ya que igual que con Spring la he usado anteriormente.

En Resumen:



Spring Boot



Java 21



MariaDB

Despliegue y otras herramientas

En cuanto al despliegue, quería asegurarme de que la aplicación fuera fácilmente trasladable, por eso he apostado por Docker.

Además, he usado Jenkins para la integración y entrega continua (CI/CD), automatizando despliegues, lo cual mejora el flujo de trabajo y la fiabilidad del código.

Por último, El despliegue está implementado en los servicios de AWS, con su EC2 (Instancia del servidor), S3 (Almacenamiento escalable) y RDS (Base de datos) correctamente configurados y operativos.

En Resumen:



Docker



Jenkins

Jenkins



AWS

API Externas

Para mejorar y brindar ciertos datos he usado varias API externas

- Google Books API (Usado para los libros)
- Wikidata API (Usado para los autores)
- MediaWiki API (Usado para las biografías de los autores)



Esquema de la BD

Voy a describir las tablas principales de la Base de datos de forma general:

Usuario

En esta tabla se almacena toda la información básica de cada persona registrada en la plataforma. Los campos principales son:

- **id_usuario**: Identificador único de cada usuario.
- **usuario**: Nombre de usuario único.
- **nombre**: Nombre real del usuario.
- **apellido**: Apellido del usuario.
- **gmail**: Correo electrónico único.
- **contrasenna**: Contraseña cifrada.
- **enabled**: Indica si la cuenta está activa.
- **fecha_creacion**: Fecha y hora de creación del usuario.

Role

Esta tabla sirve para diferenciar los distintos roles o permisos que puede tener cada usuario dentro de la web. Los campos son:

- **id_role**: Identificador único del rol.
- **nombre**: Nombre del rol, único.

Libro

Aquí se guarda la información esencial de cada libro disponible. Los campos incluyen:

- **id_libro**: Identificador único del libro
- **titulo**: Título del libro
- **isbn**: Código ISBN
- **editorial**: Nombre de la editorial.
- **anio_publicacion**: Año de publicación
- **genero**: Género literario.
- **sinopsis**: Resumen o descripción breve del libro.
- **portada_url**: Enlace a la imagen de la portada



- **numero_paginas**: Número total de páginas

Autor

En esta tabla se recogen los datos principales de cada autor, como:

- **id_autor**: Identificador único del autor
- **nombre**: Nombre del autor.
- **apellido**: Apellido del autor.
- **fecha_nacimiento**: Fecha de nacimiento.
- **fecha_fallecimiento**: Fecha de fallecimiento
- **nacionalidad**: Nacionalidad del autor.
- **biografia**: Breve biografía o descripción.
- **foto_url**: Enlace a la foto del autor.

Relaciones libro/autor (libro_autor)

Esta es una tabla intermedia que permite establecer una relación de muchos a muchos entre libros y autores. Los campos son:

- **id_libro**: Referencia al libro
- **id_autor**: Referencia al autor

usuario_role

Esta tabla se utiliza para vincular a cada usuario con uno o varios roles. Los campos son:

- **id_usuario**: Referencia al usuario
- **id_role**: Referencia al rol

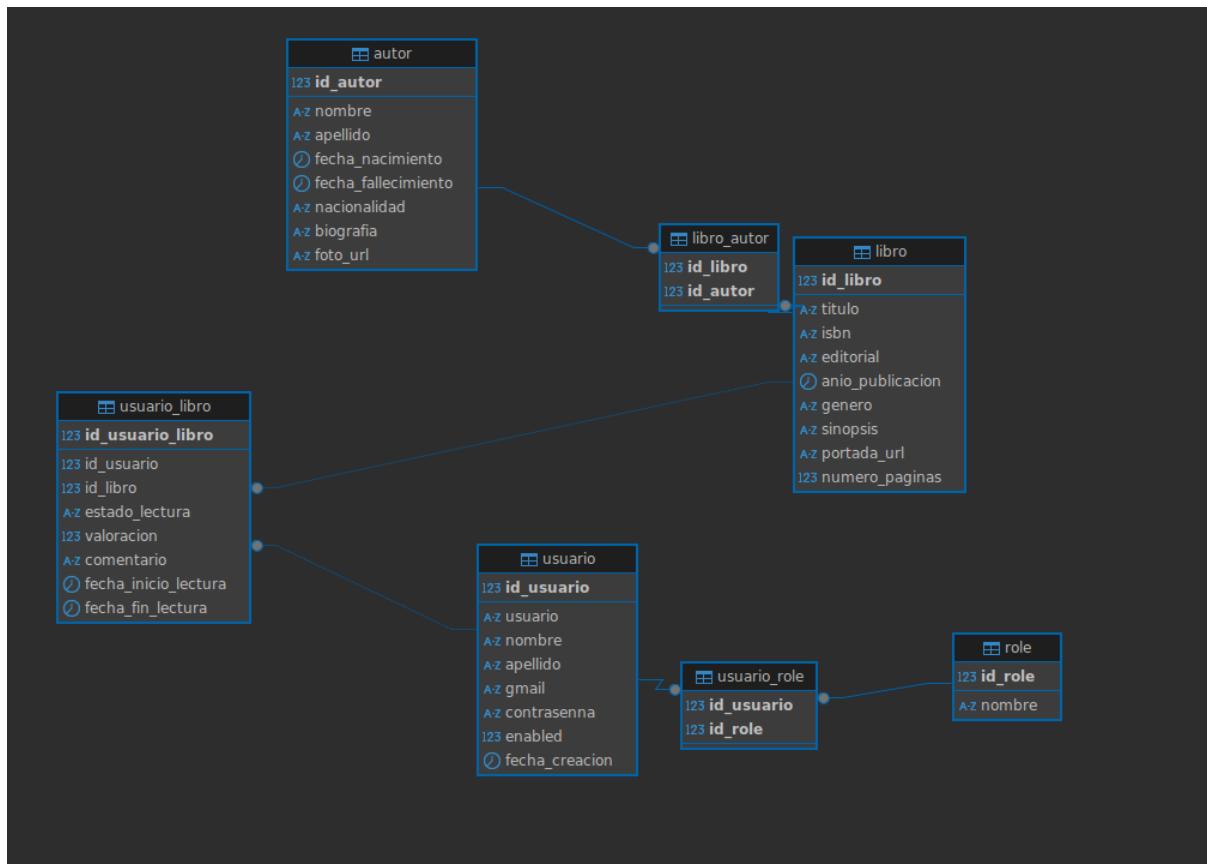
usuario_libro

Aquí se almacena el historial de lectura de cada usuario. Los campos principales son:

- **id_usuario_libro**: Identificador único de la relación
- **id_usuario**: Referencia al usuario



- **id_libro:** Referencia al libro
- **estado_lectura:** Estado actual de la lectura (solo puede ser: LEYENDO, TERMINADO, PENDIENTE, ABANDONADO).
- **valoracion:** Valoración del libro, entre 1 y 5.
- **comentario:** Comentarios personales del usuario sobre el libro.
- **fecha_inicio_lectura:** Fecha en la que el usuario empezó a leer el libro.
- **fecha_fin_lectura:** Fecha en la que terminó de leerlo
- **paginas_leidas:** Número de páginas leídas





Prototipo en Figma

Enlace Figma: [Readora](#)

Este es el prototipo en Figma de mi proyecto. Con la presentación, podemos ver cómo ha cambiado la idea o hacia dónde va.





Home - Sin registrar

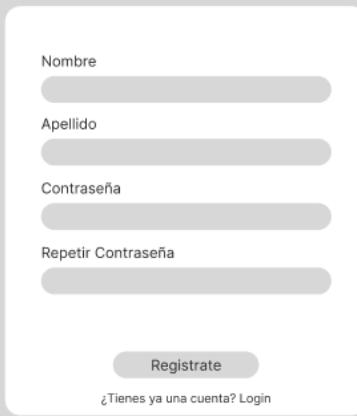
The screenshot shows a dark-themed web application. At the top, there is a navigation bar with a logo icon, a search bar, and buttons for "Libros", "Register/Login", and a user icon. The main content area features a large "Bienvenido" (Welcome) message. Below this, there is a row of three placeholder text blocks labeled "Lorem ipsum".

Login - Sin registrar

The screenshot shows a dark-themed web application. At the top, there is a navigation bar with a logo icon, a search bar, and buttons for "Libros", "Register/Login", and a user icon. The main content area features a login form with fields for "Usuario" and "Contraseña", a "Login" button, and a link for "¿No tienes cuenta? Regístrate". Below the form, there is a row of three placeholder text blocks labeled "Lorem ipsum".



Registro - Sin registrar



The registration form consists of four input fields: Nombre, Apellido, Contraseña, and Repetir Contraseña. Below the fields is a 'Regístrate' button and a link to '¿Tienes ya una cuenta? Login'.

Nombre
Apellido
Contraseña
Repetir Contraseña

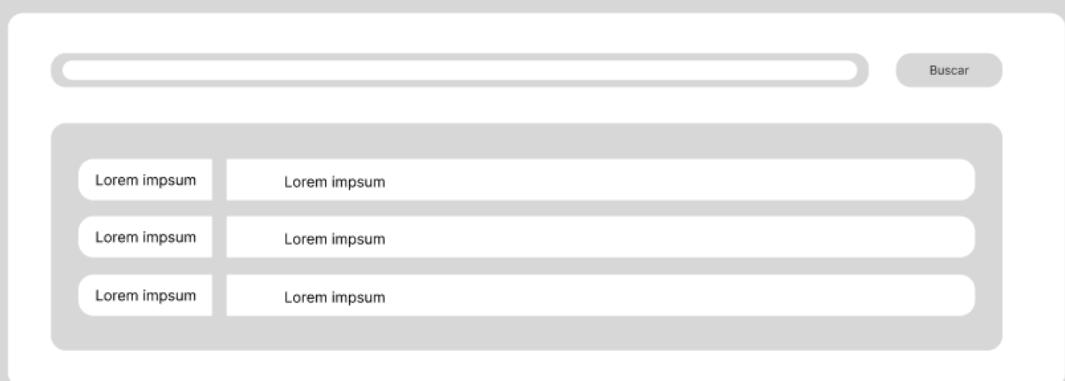
Regístrate

¿Tienes ya una cuenta? Login

Libros Register/Login

Placeholder text: Lorem ipsum

Buscar Libros - User



The search results page displays a grid of book entries, each with two columns: 'Lorem ipsum' and 'Lorem ipsum'.

Placeholder text: Lorem ipsum

Libros Register/Login

Placeholder text: Lorem ipsum



Buscar Autores - User

The interface features a dark header bar with a logo icon, a search bar, and navigation links for 'Libros', 'Register/Login', and a user icon. The main content area contains a search form with a placeholder 'Buscar' and three results cards, each showing a thumbnail, title ('Lorem ipsum'), and subtitle ('Lorem ipsum'). Below the search form is a grid of four items, each with a thumbnail and the text 'Lorem ipsum' repeated three times.

Libros de Usuario - User

The interface features a dark header bar with a logo icon, a search bar, and navigation links for 'Libros', 'Register/Login', and a user icon. The main content area contains a search form with a placeholder 'Buscar' and four results cards, each showing a thumbnail, title ('Lorem ipsum'), and subtitle ('Lorem ipsum'). Each card also includes a small icon with a magnifying glass. Below the search form is a grid of four items, each with a thumbnail and the text 'Lorem ipsum' repeated three times.



Detalles Libro - User

The wireframe shows a top navigation bar with a user icon, 'Libros', 'Register/Login', and a search icon. Below is a large white card area containing a book cover placeholder with a camera icon, followed by three input fields labeled 'Titulo', 'Autor', and 'Detalles'. At the bottom, there are three columns of placeholder text: 'Lorem ipsum' repeated three times.

Vista Acciones Libro - User

The wireframe shows a top navigation bar with a user icon, 'Libros', 'Register/Login', and a search icon. Below is a large white card area containing a book cover placeholder with a camera icon, followed by two input fields labeled 'Anotaciones' and 'Paginas Leidas'. To the right is a large empty box with a 'Opinar' button at the top. At the bottom, there are three columns of placeholder text: 'Lorem ipsum' repeated three times.



Detalles Usuario - User

The screenshot shows a mobile application interface. At the top, there is a header bar with a blue square icon on the left, followed by three buttons: "Libros", "Register/Login", and a camera icon. Below the header is a large white rectangular area containing a placeholder image for a user profile (a camera icon inside a square). To the right of the image, the word "Usuario" is written. Further to the right, the word "Detalles" is repeated above and below a large, empty rectangular box. At the bottom of the screen, there is a dark footer bar with the same blue square icon on the left, followed by three lines of text: "Lorem ipsum", "Lorem ipsum", and "Lorem ipsum" repeated three times.

Panel de Administrador - Admin

The screenshot shows a mobile application interface. At the top, there is a header bar with a blue square icon on the left, followed by three buttons: "Libros", "Register/Login", and a camera icon. Below the header is a large white rectangular area containing five tabs at the top: "Libro", "Autor", "Usuario", "Role", and "Usuario Libro". Below the tabs is a grid of three rows, each consisting of two columns. The first column of each row contains the text "Lorem ipsum". The second column of each row contains the text "Lorem ipsum". At the bottom of the screen, there is a dark footer bar with the same blue square icon on the left, followed by three lines of text: "Lorem ipsum", "Lorem ipsum", and "Lorem ipsum" repeated three times.

Diseño Final

Además del prototipado en Figma voy a añadir como se muestra actualmente las vistas, ya en Producción la app.



[!\[\]\(a14fa82ea835376918c1448ab620eed8_img.jpg\) Readora](#)

[Inicio](#) [Registro](#) [Login](#) [...](#)

Tu biblioteca personal en Readora

Gestiona tus lecturas, descubre nuevos títulos y lleva un registro de tu viaje literario de manera simple y elegante.

[Crear cuenta](#) [Iniciar sesión](#)



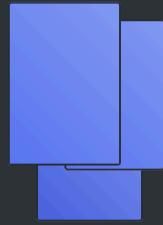
[!\[\]\(b01edc297f25865a2c3fdd8cef87e3c9_img.jpg\) Readora](#)

[Inicio](#) [Registro](#) [Login](#) [...](#)

Tu biblioteca personal en Readora

Gestiona tus lecturas, descubre nuevos títulos y lleva un registro de tu viaje literario de manera simple y elegante.

[Crear cuenta](#) [Iniciar sesión](#)



Lo que te ofrece Readora



Readora - Javier Moreno Salas

Readora

Inicio Registro Login

¡Bienvenido de nuevo!

Inicia sesión para acceder a tu biblioteca personal

Usuario *

Contraseña *

→ Iniciar Sesión

O

Iniciar sesión con Google

[¿No tienes cuenta? Regístrate aquí](#)

[← Volver al inicio](#)

Readora

Inicio Registro Login

Únete a Readora

Crea tu cuenta y comienza a gestionar tu biblioteca personal

Usuario *

Correo electrónico *

Nombre *

Apellido *

Contraseña *

● La contraseña debe tener al menos 6 caracteres. Te recomendamos usar letras, números y símbolos.

→ Crear cuenta

O

Registrarse con Google



Readora - Javier Moreno Salas

Readora

Inicio Buscador Mi Biblioteca Panel Admin admin

Mi Biblioteca Personal

Organiza y gestiona todas tus lecturas en un solo lugar

Leyendo 0 Por leer 0 Leídos 0 Abandonados 0

Tu biblioteca está vacía

No tienes libros en tu biblioteca personal. Utiliza el buscador para encontrar libros y añadirlos a tu colección.

Buscar libros

Inicio Buscador Biblioteca

Readora

Panel de administración

Gestión de contenidos del sistema

Usuarios Libros Autores Roles Lecturas

Gestión de Roles

ID	NOMBRE	ACCIONES
1	ROLE_ADMIN	editar eliminar
2	ROLE_USER	editar eliminar

Mostrando 1 a 2 de 2 registros Mostrar: 10 1

Readora

Tu biblioteca personal, siempre contigo

Inicio Buscador Biblioteca

© 2025 Readora. Todos los derechos reservados.



Readora - Javier Moreno Salas

Readora

Inicio Buscador Mi Biblioteca Panel Admin admin

Mi perfil

Gestiona tu información personal y configuración de seguridad

Información personal

Nombre de usuario * admin

Nombre * javier

Apellido * moreno

Correo electrónico * example1@gmail.com

Seguridad

Actualiza tu contraseña

*** Mantén tu cuenta segura cambiando tu contraseña regularmente

Cambiar contraseña

Información de cuenta

Estado de cuenta Activa

Fecha de registro 04/06/2025



Readora - Javier Moreno Salas

Readora

Inicio Buscador Mi Biblioteca Panel Admin admin

Buscador

Buscar por título, palabra clave... Buscar Libros Autores

Readora Tu biblioteca personal, siempre contigo

Inicio Buscador Biblioteca © 2025 Readora. Todos los derechos reservados. Créditos y licencias de datos

Readora

Inicio Buscador Mi Biblioteca Panel Admin admin

Buscador

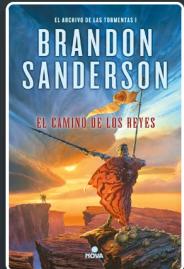
Buscar por título, palabra clave... Buscar Libros Autores

Resultados (30)

<p>El desarrollo web desde el entorno cliente Javier Gómez Delgado 2023-09-21 · Diseño</p> <p>Uno de los perfiles digitales más demandados en el mundo de los negocios es el de desarrollador web, un profesional altamente valorado en las...</p> <p>Añadir a mi biblioteca</p>	<p>El Hobbit, o, Historia de una Ida y de una Vuelta J. R. R. Tolkien 2003-09-02 · Fantasía</p> <p>Bilbo Bolsón es como cualquier hobbit: no mide más de metro y medio, vive pacíficamente en la Comarca, y su máxima aspiración es disfrutar d...</p> <p>Añadir a mi biblioteca</p>	<p>El poder de los 5 segundos Mel Robbins 2018-05-08 · Autoayuda</p> <p>En El Poder de los 5 segundos descubrirás que son sólo cinco los segundos que necesitas para confiar, romper el hábito de la procrastinación y ...</p> <p>Añadir a mi biblioteca</p>
<p>El Principito (edición original) Antoine de Saint-Exupéry 2015-08-10 · Ficción</p>	<p>España, un reino caído Autor desconocido 2021-05-31 · Historia</p>	<p>STEPHEN KING FIN DE GUARDIA Stephen King 2021-12-07 · Ficción</p>



Readora Inicio Buscador Mi Biblioteca Panel Admin admin



El camino de los reyes (El Archivo de las Tormentas 1)

por Brandon Sanderson

Ficción 2015 NOVA

8490691770 1200 2015 NOVA

ISBN PÁGINAS PUBLICACIÓN EDITORIAL

[Añadir a mi biblioteca](#)

Sinopsis

El camino de los reyes es el primer volumen de «El Archivo de las Tormentas», el resultado de más de una década de construcción y escritura de universos, convertido en una obra maestra de la fantasía contemporánea en diez volúmenes. Con ella, Brandon Sanderson se postula como el autor del género que más lectores está ganando en todo el mundo.

Anhelo los días previos a la Última Desolación.

Los días en que los Heraldos nos abandonaron y los Caballeros Radiantes se giraron en nuestra contra. Un tiempo en que aún había magia en el mundo y honor en el corazón de los hombres.

El mundo fue nuestro, pero lo perdimos. Probablemente no hay nada más estimulante para las almas de los hombres que la victoria.

Readora Inicio Buscador Mi Biblioteca Panel Admin admin



Brandon Sanderson

Autor 1975 - Presente

Nacionalidad Estados Unidos Nacimiento 19/12/1975

Biografía

Brandon Sanderson (Lincoln, Nebraska, 19 de diciembre de 1975) es un escritor estadounidense de literatura fantástica y ciencia ficción. Es conocido sobre todo por el universo ficticio de Cosmère, en el que se ambientan la mayoría de sus novelas de fantasía, entre las que destacan las series Nacidos de la bruma (Mistborn) y El archivo de las tormentas. Fuera del Cosmère, ha escrito varias series juveniles y para jóvenes adultos, como The Reckoners, la serie Skyward y la serie Alcatraz. También es conocido por haber terminado la serie de alta fantasía de Robert Jordan La Rueda del Tiempo. Sanderson ha creado varias novelas gráficas de fantasía, como White Sand o Dark One.

Creó las Leyes de la Magia de Sanderson y popularizó la idea de sistemas de «magia dura» y «magia blanda». [1][2][3] En 2008, Sanderson inició un podcast con el escritor Dan Wells y el dibujante Howard Tayler llamado Writing Excuses, en el que se tratan temas sobre crear escritura de géneros y webcómics. En 2016, la compañía de medios estadounidense DMG Entertainment obtuvo los derechos de todo el universo de Cosmère creado por Sanderson. [4] pero desde entonces los derechos han vuelto a manos de Sanderson. La campaña de Kickstarter de Sanderson de marzo de 2022 se convirtió en la más exitosa de la historia, con 185.341 donantes que prometieron 41.754.153 dólares. [5]

Nacido en Nebraska, [6] es miembro de La Iglesia de Jesucristo de los Santos de los Últimos Días [7] y actualmente reside en Provo (Utah), con su esposa Emily con la que contrajo matrimonio el 7 de julio de 2006. [8] Obtuvo un máster en Literatura Creativa en 2005 en la Brigham Young University, [9] donde fue compañero de habitación del campeón de Jeopardy! Ken Jennings. [10] Ha sido nominado dos veces para el Premio John



Definición API REST

Sobre la API REST tengo una muy buena documentación en el swagger, para poder probar el swagger tenemos un api-docs.json para ver la API en el postman. Además de un PDF con la API de swagger completa.



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

/api-docs

Explore

API de Readora

[/api-docs](#)

API REST para la gestión de lecturas de usuarios

Servers

[https://localhost:8443 - Generated server url](https://localhost:8443)

Authorize

Autores

 Operaciones relacionadas con la gestión de autores

^

GET[/api/autores/{id}](#) Obtener un autor por ID

^

Devuelve los datos de un autor dado su ID

Parameters[Try it out](#)**Name** **Description****id** * required

integer(\$int64)

(path)

id

Responses**Code** **Description****Links**

200 Autor encontrado

No links

Media type



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<pre>/*</pre> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "id": "9007199254740991", "nombre": "string", "apellido": "string", "fechaNacimiento": "2025-06-10T13:34:17.785Z", "fechaFallecimiento": "2025-06-10T13:34:17.785Z", "nacionalidad": "string", "biografia": "string", "fotoUrl": "string" }</pre>	
404	Autor no encontrado	No links

PUT /api/autores/{id} Actualizar un autor

Modifica los datos de un autor existente

Parameters

Name	Description
id * required integer(\$int64) (path)	<input type="text" value="id"/>

Request body required



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>**Example Value** Schema

```
{  
    "nombre": "string",  
    "apellido": "string",  
    "fechaNacimiento": "2025-06-10T13:34:17.799Z",  
    "fechaFallecimiento": "2025-06-10T13:34:17.799Z",  
    "nacionalidad": "string",  
    "biografia": "string",  
    "fotoUrl": "string"  
}
```

Responses

Code	Description	Links
200	Autor actualizado correctamente Media type <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">/*</div> Controls Accept header.	<i>No links</i>
404	Autor no encontrado Media type <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">/*</div>	<i>No links</i>



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

DELETE /api/autores/{id} Eliminar un autor

Elimina un autor del sistema por su ID

Parameters

Name **Description**

id * required integer(\$int64) (path)	<input type="text" value="id"/>
--	---------------------------------

Responses

Code	Description	Links
204	Autor eliminado correctamente Media type <input type="text" value="/*"/> Controls Accept header. Example Value Schema string	No links
404	Autor no encontrado Media type <input type="text" value="/*"/> Example Value Schema string	No links

GET /api/autores Obtener todos los autores



Swagger UI

[https://localhost:8443/swagger-ui/index.html#/Autores...](https://localhost:8443/swagger-ui/index.html#/Autores/)

Devuelve una lista de todos los autores

Parameters

No parameters

Responses

Code	Description	Links
200	Lista de autores obtenida correctamente	No links

Media type

/

Controls Accept header.

Example Value Schema

```
[  
  {  
    "id": 9007199254740991,  
    "nombre": "string",  
    "apellido": "string",  
    "fechaNacimiento": "2025-06-10T13:34:17.813Z",  
    "fechaFallecimiento": "2025-06-10T13:34:17.813Z",  
    "nacionalidad": "string",  
    "biografia": "string",  
    "fotoUrl": "string"  
  }  
]
```

POST /api/autores Crear un nuevo autor 🔒 ⌂

Registra un nuevo autor en el sistema

Parameters

No parameters

Request body required

application/json

Example Value Schema



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

```
{
  "nombre": "string",
  "apellido": "string",
  "fechaNacimiento": "2025-06-10T13:34:17.817Z",
  "fechaFallecimiento": "2025-06-10T13:34:17.817Z",
  "nacionalidad": "string",
  "biografia": "string",
  "fotoUrl": "string"
}
```

Responses

Code	Description	Links
201	Autor creado correctamente Media type <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">/*</div> Controls Accept header.	No links
500	Error al crear autor Media type <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">/*</div> Controls Accept header.	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Devuelve los libros escritos por un autor específico

Parameters

Try it out

Name	Description
id * required integer(\$int64) (path)	<input type="text" value="id"/>

Responses

Code	Description	Links
200	Libros obtenidos correctamente	No links
	Media type <input type="text" value="/*"/> Controls Accept header.	
	Example Value Schema	
	<pre>[{ "id": 9007199254740991, "titulo": "string", "isbn": "string", "editorial": "string", "anioPublicacion": "2025-06-10", "genero": "string", "sinopsis": "string", "portadaUrl": "string", "numeroPaginas": 1073741824 }]</pre>	
500	Error al obtener los libros	No links
	Media type <input type="text" value="/*"/>	
	Example Value Schema	
	<pre>[{ "id": 9007199254740991, "titulo": "string", "isbn": "string", "editorial": "string", "anioPublicacion": "2025-06-10", "genero": "string", "sinopsis": "string", "portadaUrl": "string", "numeroPaginas": 1073741824 }]</pre>	



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
<pre>"genero": "string", "sinopsis": "string", "portadaUrl": "string", "numeroPaginas": 1073741824 }</pre>		

GET /api/autores/search Buscar autores		
Busca autores por nombre o biografía, con paginación		
Parameters		
Name	Description	
query string (query)	<input type="text" value="query"/>	
page integer(\$int32) (query)	<input type="text" value="0"/> Default value : 0	
size integer(\$int32) (query)	<input type="text" value="10"/> Default value : 10	
sort string (query)	<input type="text" value="nombre"/> Default value : nombre	
direction string (query)	<input type="text" value="asc"/> Default value : asc	
Responses		
Code	Description	Links
200	Autores encontrados Media type	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
<pre>/*</pre> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "totalElements": 9007199254740991, "totalPages": 1073741824, "pageable": { "paged": true, "pageNumber": 1073741824, "pageSize": 1073741824, "offset": 9007199254740991, "sort": { "sorted": true, "empty": true, "unsorted": true }, "unpaged": true }, "first": true, "last": true, "size": 1073741824, "content": [{ "id": 9007199254740991, "nombre": "string", "apellido": "string", "fechaNacimiento": "2025-06-10T13:34:17.841Z", "fechaFallecimiento": "2025-06-10T13:34:17.841Z", "nacionalidad": "string", "biografia": "string", }] }</pre>		

GET /api/autores/paginados Obtener autores paginados

Devuelve autores paginados y ordenados

Parameters

Name	Description
page	Default value : 0 integer(\$int32) (query) <input type="text" value="0"/>
size	Default value : 10 integer(\$int32) (query) <input type="text" value="10"/>



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Name	Description
sort string (query)	<i>Default value : nombre</i> <input type="text" value="nombre"/>
direction string (query)	<i>Default value : asc</i> <input type="text" value="asc"/>

Responses

Code	Description	Links
200	Autores paginados obtenidos correctamente Media type <input type="text" value="/*"/> Controls Accept header.	No links

Example Value Schema

```
{  
    "totalElements": 9007199254740991,  
    "totalPages": 1073741824,  
    "pageable": {  
        "paged": true,  
        "pageNumber": 1073741824,  
        "pageSize": 1073741824,  
        "offset": 9007199254740991,  
        "sort": {  
            "sorted": true,  
            "empty": true,  
            "unsorted": true  
        },  
        "unpaged": true  
    },  
    "first": true,  
    "last": true,  
    "size": 1073741824,  
    "content": [  
        {  
            "id": 9007199254740991,  
            "nombre": "string",  
            "apellido": "string",  
            "fechaNacimiento": "2025-06-10T13:34:17.849Z",  
            "fechaFallecimiento": "2025-06-10T13:34:17.849Z",  
            "nacionalidad": "string",  
            "biografia": "string",  
            "creado": "2025-06-10T13:34:17.849Z",  
            "actualizado": "2025-06-10T13:34:17.849Z"  
        }  
    ]  
}
```



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Relaciones Usuario-Libro

Operaciones para gestionar las relaciones entre usuarios y libros.

**GET**`/api/usuario-libros/{id}` Obtener relación usuario-libro por ID

Devuelve la relación entre un usuario y un libro especificado por su ID.

Parameters**Try it out**

Name	Description
id * required integer(\$int64) (path)	<input type="text" value="id"/>

Responses

Code	Description	Links
200	Relación obtenida exitosamente	<i>No links</i>
	Media type <input type="text" value=" */*"/> Controls Accept header.	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "usuarioId": 9007199254740991, "libroId": 9007199254740991, "estadoLectura": "string", "valoracion": 1073741824, "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }</pre>	
404	Relación no encontrada	<i>No links</i>
	Media type <input type="text" value=" */*"/>	
	Example Value Schema	



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<pre>{ "id": 9007199254740991, "usuarioId": 9007199254740991, "libroId": 9007199254740991, "estadoLectura": "string", "valoracion": 1073741824, "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }</pre>	
500	Error interno del servidor	No links

PUT /api/usuario-libros/{id} Actualizar relación usuario-libro

Actualiza los datos de una relación entre usuario y libro existente.

Parameters

Name	Description
id * required integer(\$int64) (path)	<input type="text" value="id"/>

Request body required

Example Value Schema

```
{  
    "usuarioId": 9007199254740991,
```



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

```
"libroId": 9007199254740991,  
"estadoLectura": "string",  
"valoracion": 1073741824,  
"comentario": "string",  
"fechaInicioLectura": "2025-06-10",  
"fechaFinLectura": "2025-06-10",  
"paginasLeidas": 1073741824  
}
```

Responses

Code	Description	Links
200	Relación actualizada exitosamente	<i>No links</i>
	Media type	
	<pre> */*</pre>	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "usuarioId": 9007199254740991, "libroId": 9007199254740991, "estadoLectura": "string", "valoracion": 1073741824, "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }</pre>	
404	Relación no encontrada	<i>No links</i>
	Media type	
	<pre> */*</pre>	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "usuarioId": 9007199254740991, "libroId": 9007199254740991, "estadoLectura": "string", "valoracion": 1073741824, "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }</pre>	
500	Error interno del servidor	<i>No links</i>
	Media type	



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links										
<pre>/*</pre>	<p>Example Value Schema</p> <pre>{ "id": 9007199254740991, "usuarioId": 9007199254740991, "libroId": 9007199254740991, "estadoLectura": "string", "valoracion": 1073741824, "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }</pre>											
<p>DELETE /api/usuario-libros/{id} Eliminar relación usuario-libro 🔒 ⌂</p> <p>Elimina una relación entre un usuario y un libro del sistema.</p> <p>Parameters Try it out</p> <table border="1"><thead><tr><th>Name</th><th>Description</th></tr></thead><tbody><tr><td>id * required integer(\$int64)</td><td>id (path)</td></tr></tbody></table> <p>Responses</p> <table border="1"><thead><tr><th>Code</th><th>Description</th><th>Links</th></tr></thead><tbody><tr><td>204</td><td>Relación eliminada exitosamente</td><td>No links</td></tr></tbody></table> <p>Media type <pre>/*</pre></p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>string</pre>			Name	Description	id * required integer(\$int64)	id (path)	Code	Description	Links	204	Relación eliminada exitosamente	No links
Name	Description											
id * required integer(\$int64)	id (path)											
Code	Description	Links										
204	Relación eliminada exitosamente	No links										



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
404	Relación no encontrada Media type <div style="border: 1px solid black; padding: 2px;">*/*</div> Example Value Schema <code>string</code>	<i>No links</i>
500	Error interno del servidor Media type <div style="border: 1px solid black; padding: 2px;">*/*</div> Example Value Schema <code>string</code>	<i>No links</i>

GET /api/usuario-libros Obtener todas las relaciones usuario-libro

Devuelve todas las relaciones entre usuarios y libros registradas.

Parameters **Try it out**

No parameters

Responses

Code	Description	Links
200	Relaciones obtenidas exitosamente Media type <div style="border: 1px solid green; padding: 2px;">*/*</div> Controls Accept header.	<i>No links</i>



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
Example Value Schema		
500	<pre>{ "id": 9007199254740991, "usuarioId": 9007199254740991, "libroId": 9007199254740991, "estadoLectura": "string", "valoracion": 1073741824, "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }</pre> Error interno del servidor	<i>No links</i>
Media type <code>*/*</code>		

POST /api/usuario-libros Crear nueva relación usuario-libro

Crea una nueva relación entre un usuario y un libro.

Parameters

No parameters

Request body required

Example Value Schema

```
{  
    "usuarioId": 9007199254740991,  
    "libroId": 9007199254740991,  
    "estadoLectura": "string",
```



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

```
"valoracion": 1073741824,  
"comentario": "string",  
"fechaInicioLectura": "2025-06-10",  
"fechaFinLectura": "2025-06-10",  
"paginasLeidas": 1073741824  
}
```

Responses

Code	Description	Links
201	Relación creada exitosamente	<i>No links</i>
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">*/*</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "id": 9007199254740991, "usuarioId": 9007199254740991, "libroId": 9007199254740991, "estadoLectura": "string", "valoracion": 1073741824, "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }</pre>	

PATCH /api/usuario-libros/{id}/paginas-leidas Actualizar páginas leídas



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Actualiza el número de páginas leídas de un libro por un usuario.

Parameters

Try it out

Name	Description
id * required integer(\$int64) (path)	id
paginasLeidas * required integer(\$int32) (query)	paginasLeidas

Responses

Code	Description	Links
200	Páginas leídas actualizadas exitosamente	No links
	Media type <code>*/*</code>	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "usuarioId": 9007199254740991, "libroId": 9007199254740991, "estadoLectura": "string", "valoracion": 1073741824, "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }</pre>	
400	Número de páginas inválido	No links
	Media type <code>*/*</code>	
	Example Value Schema	
	<code>string</code>	



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
404	Relación no encontrada Media type <div style="border: 1px solid black; padding: 2px; display: inline-block;">*/*</div> Example Value Schema <code>string</code>	No links
500	Error interno del servidor Media type <div style="border: 1px solid black; padding: 2px; display: inline-block;">*/*</div> Example Value Schema <code>string</code>	No links

GET /api/usuario-libros/usuario/{usuarioId} Obtener libros por usuario

Devuelve todas las relaciones entre un usuario y sus libros.

Parameters **Try it out**

Name	Description
usuarioid * required integer(\$int64) (path)	usuarioid

Responses

Code	Description	Links
------	-------------	-------



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
200	Relaciones obtenidas exitosamente Media type <div style="border: 1px solid green; padding: 2px; display: inline-block;">/*</div> Controls Accept header. Example Value Schema <pre>{ "id": 9007199254740991, "usuarioId": 9007199254740991, "libroId": 9007199254740991, "estadoLectura": "string", "valoracion": 1073741824, "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }</pre>	<i>No links</i>
500	Error interno del servidor Media type <div style="border: 1px solid green; padding: 2px; display: inline-block;">/*</div> Example Value Schema <pre>[{ "id": 9007199254740991, "usuarioId": 9007199254740991, "libroId": 9007199254740991, "estadoLectura": "string", "valoracion": 1073741824, "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }]</pre>	<i>No links</i>

GET /api/usuario-libros/usuario/{usuarioId}/buscar 🔒 ⌂

Buscar libros por estado y título.

Busca libros en la biblioteca personal de un usuario filtrados por estado de lectura y título.

Parameters **Try it out**

Name	Description
------	-------------



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Name	Description
usuarioid * required integer(\$int64) (path)	usuarioid
estado * required string (query)	estado
query * required string (query)	query

Responses		
Code	Description	Links
200	Búsqueda realizada exitosamente Media type <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">/*</div> Controls Accept header.	No links
400	Parámetros de búsqueda inválidos Media type <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">/*</div> Controls Accept header.	No links

Code	Description	Links
200	Búsqueda realizada exitosamente Media type <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">/*</div> Controls Accept header.	No links
400	Parámetros de búsqueda inválidos Media type <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">/*</div> Controls Accept header.	No links

Example Value Schema

```
{  
    "id": 9007199254740991,  
    "usuarioId": 9007199254740991,  
    "libroId": 9007199254740991,  
    "estadoLectura": "string",  
    "valoracion": 1073741824,  
    "comentario": "string",  
    "fechaInicioLectura": "2025-06-10",  
    "fechaFinLectura": "2025-06-10",  
    "paginasLeidas": 1073741824  
}
```

Example Value Schema

```
[  
    {  
        "id": 9007199254740991,  
        "usuarioId": 9007199254740991,  
        "libroId": 9007199254740991,  
        "estadoLectura": "string",  
        "valoracion": 1073741824,  
        "comentario": "string",  
        "fechaInicioLectura": "2025-06-10",  
        "fechaFinLectura": "2025-06-10",  
        "paginasLeidas": 1073741824  
    }]
```



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<pre> "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }]</pre>	
500	Error interno del servidor	No links

GET /api/usuario-libros/paginados

Obtener relaciones usuario-libro paginadas

🔒 ⌂

Devuelve una página de relaciones entre usuarios y libros, con soporte de paginación y ordenamiento.

Parameters

Try it out

Name	Description
page integer(\$int32) (query)	Default value : 0 0
size integer(\$int32) (query)	Default value : 10 10
sort string (query)	Default value : id id



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Name	Description	
direction string (query)	<i>Default value : asc</i> <input type="text" value="asc"/>	
Responses		
Code	Description	Links
200	Relaciones obtenidas exitosamente Media type <input type="text" value="/*"/> Controls Accept header. Example Value Schema <pre>{ "id": 9007199254740991, "usuarioId": 9007199254740991, "libroId": 9007199254740991, "estadoLectura": "string", "valoracion": 1073741824, "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }</pre>	<i>No links</i>
500	Error interno del servidor Media type <input type="text" value="/*"/> Example Value Schema <pre>{ "totalElements": 9007199254740991, "totalPages": 1073741824, "pageable": { "paged": true, "pageNumber": 1073741824, "pageSize": 1073741824, "offset": 9007199254740991, "sort": { "sorted": true, "empty": true, "unsorted": true }, "unpaged": true }, "first": true, "last": true, "size": 1073741824, "number": 1 }</pre>	<i>No links</i>



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
<pre>"content": [{ "id": 9007199254740991, "usuarioId": 9007199254740991, "libroId": 9007199254740991, "estadoLectura": "string", "valoracion": 1073741824, "comentario": "string", "fechaInicioLectura": "2025-06-10", "fechaFinLectura": "2025-06-10", "paginasLeidas": 1073741824 }]</pre>		

GET /api/usuario-libros/libro/{libroId} Obtener usuarios por libro 🔒 ⌂

Devuelve todas las relaciones entre usuarios y un libro específico.

Parameters Try it out

Name	Description
libroId * required integer(\$int64) (path)	libroId

Responses

Code	Description	Links
200	Relaciones obtenidas exitosamente	No links

Media type /*

Controls Accept header.

Example Value Schema

```
{  
  "id": 9007199254740991,  
  "usuarioId": 9007199254740991,  
  "libroId": 9007199254740991,  
  "estadoLectura": "string",  
  "valoracion": 1073741824,  
  "comentario": "string",  
  "fechaInicioLectura": "2025-06-10",  
  "fechaFinLectura": "2025-06-10",  
  "paginasLeidas": 1073741824  
}
```



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
500	Error interno del servidor	No links

Media type

/

Example Value Schema

```
[  
  {  
    "id": 9007199254740991,  
    "usuarioId": 9007199254740991,  
    "libroId": 9007199254740991,  
    "estadoLectura": "string",  
    "valoracion": 1073741824,  
    "comentario": "string",  
    "fechaInicioLectura": "2025-06-10",  
    "fechaFinLectura": "2025-06-10",  
    "paginasLeidas": 1073741824  
  }  
]
```

Autenticación

Operaciones relacionadas con login, logout y validación de sesión

POST /api/v1/logout Cerrar sesión

Elimina la cookie JWT y termina la sesión del usuario.

Parameters **Try it out**

No parameters

Responses

Code	Description	Links
200	Sesión cerrada exitosamente	No links

Media type

/

Controls Accept header.

Example Value Schema



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	string	
500	Error interno Media type <div style="border: 1px solid black; padding: 2px; display: inline-block;">/*</div>	No links

POST /api/v1/authenticate Autenticar usuario ^

Autentica con usuario y contraseña. Devuelve un JWT en una cookie segura.

Parameters **Try it out**

No parameters

Request body required application/json

Example Value Schema

```
{  
  "usuario": "string",  
  "contraseña": "string"  
}
```

Responses

Code	Description	Links
200	Autenticación exitosa Media type <div style="border: 1px solid green; padding: 2px; display: inline-block;">/*</div>	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "token": "string", "mensage": "string" }</pre>	
400	Datos inválidos Media type <div style="border: 1px solid black; padding: 2px; display: inline-block;">/*</div> <p>Example Value Schema</p> <pre>{ "token": "string", "mensage": "string" }</pre>	No links
401	Credenciales inválidas Media type <div style="border: 1px solid black; padding: 2px; display: inline-block;">/*</div> <p>Example Value Schema</p> <pre>{ "token": "string", "mensage": "string" }</pre>	No links
500	Error interno Media type <div style="border: 1px solid black; padding: 2px; display: inline-block;">/*</div> <p>Example Value Schema</p> <pre>{ "token": "string", "mensage": "string" }</pre>	No links

GET	/api/v1/user-info Obtener información del usuario	
Obtiene los datos del usuario autenticado.		



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Parameters

No parameters

Responses

Code	Description	Links
200	Datos del usuario	No links
	Media type	
	<code>/*</code>	
	Controls Accept header.	
	Example Value Schema	
	<code>string</code>	
401	Usuario no autenticado	No links
	Media type	
	<code>/*</code>	
	Example Value Schema	
	<code>string</code>	
500	Error interno	No links
	Media type	
	<code>/*</code>	
	Example Value Schema	
	<code>string</code>	

GET /api/v1/check-auth Verificar autenticación



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Verifica si el usuario está actualmente autenticado.

Parameters

No parameters

Responses

Code	Description	Links
200	Estado de autenticación Media type <code>*/*</code> Controls Accept header. Example Value Schema <code>string</code>	No links
500	Error interno Media type <code>*/*</code> Example Value Schema <code>string</code>	No links

Roles Operaciones relacionadas con la gestión de roles del sistema

^

GET	/api/roles/{id} Obtener rol por ID	🔒	^
Recupera un rol específico según su identificador único			



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Parameters

Name **Description**

id * required integer(\$int64) (path)	id
--	----

Responses

Code	Description	Links
200	Rol encontrado Media type */ Controls Accept header. Example Value Schema <pre>{ "id": 9007199254740991, "nombre": "string" }</pre>	No links
404	Rol no encontrado Media type */ Example Value Schema <pre>{ "id": 9007199254740991, "nombre": "string" }</pre>	No links
500	Error interno del servidor Media type */ Example Value Schema <pre>{ "id": 9007199254740991, "nombre": "string" }</pre>	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

PUT /api/roles/{id} Actualizar rol existente

Actualiza los datos de un rol ya existente

Parameters **Try it out**

Name	Description
id * required integer(\$int64) (path)	id

Request body required application/json

Example Value Schema

```
{ "nombre": "string" }
```

Responses

Code	Description	Links
200	Rol actualizado correctamente Media type */ Controls Accept header.	No links
404	Rol no encontrado Media type */ Example Value Schema	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
500	<p>Error al actualizar el rol</p> <p>Media type</p> <pre>/*</pre> <p>Example Value Schema</p> <pre>{ "id": 9007199254740991, "nombre": "string" }</pre>	<i>No links</i>

DELETE /api/roles/{id} Eliminar rol 🔒 ⌂

Elimina un rol del sistema según su ID

Parameters **Try it out**

Name	Description
id * required integer(\$int64) (path)	<input type="text" value="id"/>

Responses

Code	Description	Links
204	Rol eliminado correctamente	<i>No links</i>

Media type

```
/*
```

Controls Accept header.

Example Value Schema

```
string
```



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
404	Rol no encontrado Media type <div style="border: 1px solid black; padding: 2px;">*/*</div> Example Value Schema string	No links
500	Error al eliminar el rol Media type <div style="border: 1px solid black; padding: 2px;">*/*</div> Example Value Schema string	No links

GET /api/roles Obtener todos los roles ^

Recupera la lista completa de roles disponibles

Parameters **Try it out**

No parameters

Responses

Code	Description	Links
200	Lista de roles obtenida correctamente Media type <div style="border: 1px solid green; padding: 2px;">*/*</div> Controls Accept header.	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<p>Example Value Schema</p> <pre>[{ "id": 9007199254740991, "nombre": "string" }]</pre>	
500	Error interno del servidor	<i>No links</i>

POST /api/roles Crear nuevo rol

Crea un nuevo rol en el sistema a partir de los datos proporcionados

Parameters

No parameters

Request body required

Example Value Schema

```
{  
  "nombre": "string"  
}
```

Responses

Code	Description	Links
------	-------------	-------



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
201	Rol creado correctamente Media type <div style="border: 1px solid green; padding: 2px; display: inline-block;">*/*</div> Controls Accept header. Example Value Schema <pre>{ "id": 9007199254740991, "nombre": "string" }</pre>	<i>No links</i>
500	Error al crear el rol Media type <div style="border: 1px solid green; padding: 2px; display: inline-block;">*/*</div> Example Value Schema <pre>{ "id": 9007199254740991, "nombre": "string" }</pre>	<i>No links</i>

GET /api/roles/paginados Obtener roles paginados

Recupera una página de roles ordenada por un campo específico

Parameters **Try it out**

Name	Description
page <small>integer(\$int32) (query)</small>	Default value : 0 <input type="text" value="0"/>
size <small>integer(\$int32) (query)</small>	Default value : 10 <input type="text" value="10"/>
sort <small>string (query)</small>	Default value : id <input type="text" value="id"/>



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Name	Description	
direction string (query)	<i>Default value : asc</i> <input type="text" value="asc"/>	
Responses		
Code	Description	Links
200	Página de roles obtenida correctamente	<i>No links</i>
<p>Media type</p> <div style="border: 2px solid green; padding: 2px; display: inline-block;">/*</div> <p>Controls Accept header.</p>		
<p>Example Value Schema</p> <pre>{ "totalElements": 9007199254740991, "totalPages": 1073741824, "pageable": { "paged": true, "pageNumber": 1073741824, "pageSize": 1073741824, "offset": 9007199254740991, "sort": { "sorted": true, "empty": true, "unsorted": true }, "unpaged": true }, "first": true, "last": true, "size": 1073741824, "content": [{ "id": 9007199254740991, "nombre": "string" }], "number": 1073741824, "sort": { "sorted": true, } }</pre>		
500	Error interno del servidor	<i>No links</i>
<p>Media type</p> <div style="border: 2px solid black; padding: 2px; display: inline-block;">/*</div> <p>Controls Accept header.</p>		
<p>Example Value Schema</p> <pre>{ "totalElements": 9007199254740991, }</pre>		



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<pre> "totalPages": 1073741824, "pageable": { "paged": true, "pageNumber": 1073741824, "pageSize": 1073741824, "offset": 9007199254740991, "sort": { "sorted": true, "empty": true, "unsorted": true }, "unpaged": true }, "first": true, "last": true, "size": 1073741824, "content": [{ "id": 9007199254740991, "nombre": "string" }], "number": 1073741824, "sort": { "...": ... } }</pre>	

Libros

 Operaciones relacionadas con los libros ^

GET	/api/libros/{id} Obtener libro por ID	
Busca un libro por su ID (sin incluir autores asociados).		
Parameters		Try it out
<hr/>		
Name	Description	
id * required	integer(\$int64)	<input type="text" value="id"/>
Responses		
Code	Description	Links
200	Libro encontrado	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">*/*</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "id": 9007199254740991, "titulo": "string", "isbn": "string", "editorial": "string", "anioPublicacion": "2025-06-10", "genero": "string", "sinopsis": "string", "portadaUrl": "string", "numeroPaginas": 1073741824 }</pre>	
404	<p>Libro no encontrado</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">*/*</div> <p>Example Value Schema</p> <pre>{ "id": 9007199254740991, "titulo": "string", "isbn": "string", "editorial": "string", "anioPublicacion": "2025-06-10", "genero": "string", "sinopsis": "string", "portadaUrl": "string", "numeroPaginas": 1073741824 }</pre>	No links

PUT /api/libros/{id} Actualizar libro existente 🔒 ⌂

Actualiza los datos de un libro usando su ID.

Parameters Try it out

Name	Description
id * required integer(\$int64) (path)	<input type="text" value="id"/>



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Request body required

application/json

Example Value Schema

```
{
  "titulo": "string",
  "isbn": "string",
  "editorial": "string",
  "anioPublicacion": "2025-06-10",
  "genero": "string",
  "sinopsis": "string",
  "portadaUrl": "string",
  "numeroPaginas": 1073741824,
  "autoresIds": [
    9007199254740991
  ]
}
```

Responses

Code	Description	Links
200	Libro actualizado exitosamente	<i>No links</i>
	Media type	
	/	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "titulo": "string", "isbn": "string", "editorial": "string", "anioPublicacion": "2025-06-10", "genero": "string", "sinopsis": "string", "portadaUrl": "string", "numeroPaginas": 1073741824 }</pre>	
404	Libro no encontrado	<i>No links</i>
	Media type	
	/	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "titulo": "string", "isbn": "string", "editorial": "string", "anioPublicacion": "2025-06-10", "genero": "string", "sinopsis": "string", "portadaUrl": "string", "numeroPaginas": 1073741824 }</pre>	



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
<pre>"anioPublicacion": "2025-06-10", "genero": "string", "sinopsis": "string", "portadaUrl": "string", "numeroPaginas": 1073741824 }</pre>		

DELETE /api/libros/{id} Eliminar un libro 🔒 ⌂

Elimina un libro por su ID.

Parameters Try it out

Name	Description
id * required integer(\$int64)	id (path)

Responses

Code	Description	Links
204	Libro eliminado exitosamente Media type */ Controls Accept header. Example Value Schema string	No links
404	Libro no encontrado Media type */ Example Value Schema string	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links						
<p>GET /api/libros Obtener todos los libros </p> <p>Devuelve una lista completa de todos los libros registrados.</p> <p>Parameters </p> <p>No parameters</p> <p>Responses</p> <table border="1"><thead><tr><th>Code</th><th>Description</th><th>Links</th></tr></thead><tbody><tr><td>200</td><td>Libros obtenidos exitosamente</td><td><i>No links</i></td></tr></tbody></table> <p>Media type */ Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "id": 9007199254740991, "titulo": "string", "isbn": "string", "editorial": "string", "anioPublicacion": "2025-06-10", "genero": "string", "sinopsis": "string", "portadaUrl": "string", "numeroPaginas": 1073741824 }]</pre> <p>POST /api/libros Crear un nuevo libro </p> <p>Registra un nuevo libro en el sistema.</p>			Code	Description	Links	200	Libros obtenidos exitosamente	<i>No links</i>
Code	Description	Links						
200	Libros obtenidos exitosamente	<i>No links</i>						



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Parameters

No parameters

Request body required

application/json

Example Value Schema

```
{  
    "titulo": "string",  
    "isbn": "string",  
    "editorial": "string",  
    "anioPublicacion": "2025-06-10",  
    "genero": "string",  
    "sinopsis": "string",  
    "portadaUrl": "string",  
    "numeroPaginas": 1073741824,  
    "autoresIds": [  
        9007199254740991  
    ]  
}
```

Responses

Code	Description	Links
201	Libro creado exitosamente	<i>No links</i>
	Media type <code>*/*</code> Controls Accept header.	
	Example Value Schema <code>string</code>	
400	Error de validación (ISBN duplicado u otro error)	<i>No links</i>
	Media type <code>*/*</code>	
	Example Value Schema <code>string</code>	



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
500	Error al crear el libro Media type <div style="border: 1px solid black; padding: 5px; display: inline-block;">/*</div> Example Value Schema <code>string</code>	No links

GET /api/libros/{id}/autores Obtener autores por ID de libro

Devuelve la lista de autores asociados a un libro específico.

Parameters **Try it out**

Name	Description
id * required integer(\$int64) (path)	<input type="text" value="id"/>

Responses

Code	Description	Links
200	Autores obtenidos exitosamente Media type <div style="border: 1px solid green; padding: 5px; display: inline-block;">/*</div> Controls Accept header. Example Value Schema <pre>[{"id": 9007199254740991, "nombre": "string", "apellido": "string", "fechaNacimiento": "2025-06-10T13:34:17.987Z", "correoElectronico": "string", "telefono": "string", "genero": "string", "pais": "string", "ciudad": "string", "edad": 100, "libro": {"id": 1, "titulo": "string", "autor": "string", "year": 2025, "genero": "string", "pais": "string", "ciudad": "string", "editorial": "string", "formato": "string", "precio": 100.0, "stock": 100}, "libro2": {"id": 2, "titulo": "string", "autor": "string", "year": 2025, "genero": "string", "pais": "string", "ciudad": "string", "editorial": "string", "formato": "string", "precio": 100.0, "stock": 100}}, {"id": 9007199254740992, "nombre": "string", "apellido": "string", "fechaNacimiento": "2025-06-10T13:34:17.987Z", "correoElectronico": "string", "telefono": "string", "genero": "string", "pais": "string", "ciudad": "string", "edad": 100, "libro": {"id": 1, "titulo": "string", "autor": "string", "year": 2025, "genero": "string", "pais": "string", "ciudad": "string", "editorial": "string", "formato": "string", "precio": 100.0, "stock": 100}, "libro2": {"id": 2, "titulo": "string", "autor": "string", "year": 2025, "genero": "string", "pais": "string", "ciudad": "string", "editorial": "string", "formato": "string", "precio": 100.0, "stock": 100}}]</pre>	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
<pre>"fechaFallecimiento": "2025-06-10T13:34:17.987Z", "nacionalidad": "string", "biografia": "string", "fotoUrl": "string" }</pre>		

POST [`/api/libros/{id}/autores`](#) Asignar autores a un libro 🔒 ⌂

Asigna uno o más autores a un libro específico.

Parameters [Try it out](#)

Name	Description
id * required integer(\$int64) (path)	<input type="text" value="id"/>

Request body required application/json

Example Value Schema

```
{ "autoresIds": [
  9007199254740991
]}
```

Responses

Code	Description	Links
200	Autores asignados exitosamente	No links

Media type */*
Controls Accept header.

Example Value Schema

```
string
```



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
404	<p>Libro no encontrado</p> <p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">*/*</div> <p>Example Value Schema</p> <p>string</p>	No links
500	<p>Error al asignar autores</p> <p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">*/*</div> <p>Example Value Schema</p> <p>string</p>	No links

DELETE /api/libros/{id}/autores Eliminar todos los autores de un libro

Elimina todas las relaciones entre un libro y sus autores.

Parameters **Try it out**

Name	Description
id * required integer(\$int64) (path)	<input type="text" value="id"/>

Responses



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
204	Autores eliminados exitosamente	No links
	<p>Media type</p> <div style="border: 1px solid green; padding: 2px; display: inline-block;">*/*</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <p>string</p>	
404	Libro no encontrado	No links
	<p>Media type</p> <div style="border: 1px solid green; padding: 2px; display: inline-block;">*/*</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <p>string</p>	
500	Error al eliminar autores	No links
	<p>Media type</p> <div style="border: 1px solid green; padding: 2px; display: inline-block;">*/*</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <p>string</p>	

GET /api/libros/search Buscar libros por filtros 🔒 ⌂

Realiza una búsqueda avanzada filtrando por texto, género, y rango de años.

Parameters **Try it out**

Name	Description



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Name	Description	
query string (query)	query	
genero string (query)	genero	
yearFrom integer(\$int32) (query)	yearFrom	
yearTo integer(\$int32) (query)	yearTo	
page integer(\$int32) (query)	Default value : 0 0	
size integer(\$int32) (query)	Default value : 10 10	
sort string (query)	Default value : titulo titulo	
direction string (query)	Default value : asc asc	
Responses		
Code	Description	Links
200	Resultados de búsqueda obtenidos exitosamente Media type */* Controls Accept header. Example Value Schema	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
<pre>"totalElements": 9007199254740991, "totalPages": 1073741824, "pageable": { "paged": true, "pageNumber": 1073741824, "pageSize": 1073741824, "offset": 9007199254740991, "sort": { "sorted": true, "empty": true, "unsorted": true }, "unpaged": true }, "first": true, "last": true, "size": 1073741824, "content": [{ "id": 9007199254740991, "titulo": "string", "isbn": "string", "editorial": "string", "anioPublicacion": "2025-06-10", "genero": "string", ... }</pre>		

GET /api/libros/paginados Obtener libros paginados 🔒 ⌂

Devuelve los libros paginados con ordenamiento opcional.

Parameters Try it out

Name	Description
page <small>integer(\$int32) (query)</small>	<small>Default value : 0</small> 0
size <small>integer(\$int32) (query)</small>	<small>Default value : 10</small> 10
sort <small>string (query)</small>	<small>Default value : id</small> id
direction <small>string (query)</small>	<small>Default value : asc</small> asc



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Responses

Code	Description	Links
200	Página de libros obtenida exitosamente Media type <div style="border: 1px solid #ccc; padding: 5px; display: inline-block;">/*</div> Controls Accept header.	No links

Example Value Schema

```
{
    "totalElements": 9007199254740991,
    "totalPages": 1073741824,
    "pageable": {
        "paged": true,
        "pageNumber": 1073741824,
        "pageSize": 1073741824,
        "offset": 9007199254740991,
        "sort": {
            "sorted": true,
            "empty": true,
            "unsorted": true
        },
        "unpaged": true
    },
    "first": true,
    "last": true,
    "size": 1073741824,
    "content": [
        {
            "id": 9007199254740991,
            "titulo": "string",
            "isbn": "string",
            "editorial": "string",
            "anioPublicacion": "2025-06-10",
            "genero": "string",
            "sinopsis": "string",
            "autor": {
                "id": 9007199254740991,
                "nombre": "string",
                "apellidos": "string"
            }
        }
    ]
}
```

Usuarios Operaciones relacionadas con la gestión de usuarios

^

GET /api/usuarios/{id} Obtener usuario por ID

Obtiene un usuario por su ID. Restringido a admins o al propio usuario.

Parameters

Try it out



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Name	Description	
id * required	integer(\$int64) (path)	<input type="text" value="id"/>
Responses		
Code	Description	Links
200	Usuario encontrado	<i>No links</i>
	Media type	
	<input type="text" value="/*"/>	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.019Z" }</pre>	
403	Acceso denegado	<i>No links</i>
	Media type	
	<input type="text" value="/*"/>	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.020Z" }</pre>	
404	Usuario no encontrado	<i>No links</i>
	Media type	
	<input type="text" value="/*"/>	
	Example Value Schema	



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.022Z" }</pre>	
500	Error interno del servidor	No links

PUT /api/usuarios/{id} Actualizar usuario 🔒 ⌂

Actualiza un usuario. Solo ADMIN o el usuario correspondiente pueden hacerlo.

Parameters Try it out

Name	Description
id * required integer(\$int64) (path)	<input type="text" value="id"/>

Request body required application/json

Example Value Schema

```
{  
    "nombre": "string",  
    "usuario": "string",  
    "apellido": "string",  
    "gmail": "string",  
    "contrasenna": "string",  
    "enabled": true,  
    "fechaCreacion": "2025-06-10T13:34:18.023Z"  
}
```



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

```
"enabled": true,  
"rolesIds": [  
    9007199254740991  
]  
}
```

Responses

Code	Description	Links
200	Usuario actualizado Media type <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">/*</div> Controls Accept header. Example Value Schema <pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.029Z" }</pre>	<i>No links</i>
403	Acceso denegado Media type <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">/*</div> Example Value Schema <pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.030Z" }</pre>	<i>No links</i>
404	Usuario no encontrado Media type <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">/*</div> Example Value Schema <pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.030Z" }</pre>	<i>No links</i>



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<pre>"nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.032Z" }</pre>	
500	Error interno al actualizar	No links

DELETE /api/usuarios/{id} Eliminar usuario

Elimina un usuario por ID

Parameters

Name	Description
id * required integer(\$int64)	<input type="text" value="id"/> (path)

Responses

Code	Description	Links
204	Usuario eliminado exitosamente	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<p>Controls Accept header.</p> <p>Example Value Schema</p> <p>string</p>	
404	<p>Usuario no encontrado</p> <p>Media type</p> <div style="border: 1px solid black; padding: 2px; width: fit-content;"> */*</div> <p>Example Value Schema</p> <p>string</p>	<i>No links</i>
409	<p>Usuario tiene asociaciones y no puede ser eliminado</p> <p>Media type</p> <div style="border: 1px solid black; padding: 2px; width: fit-content;"> */*</div> <p>Example Value Schema</p> <p>string</p>	<i>No links</i>
500	<p>Error interno al eliminar</p> <p>Media type</p> <div style="border: 1px solid black; padding: 2px; width: fit-content;"> */*</div> <p>Example Value Schema</p> <p>string</p>	<i>No links</i>

PUT /api/usuarios/admin-update/{id} Actualizar usuario como admin

Permite a un ADMIN editar un usuario sin modificar la sesión actual ni devolver JWT.



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Parameters

Name **Description**

id * required integer(\$int64) (path)	<input type="text" value="id"/>
--	---------------------------------

Request body required

application/json

Example Value Schema

```
{
  "nombre": "string",
  "usuario": "string",
  "apellido": "string",
  "gmail": "string",
  "contrasenna": "string",
  "enabled": true,
  "rolesIds": [
    9007199254740991
  ]
}
```

Responses

Code	Description	Links
200	Usuario actualizado exitosamente por admin	No links
	Media type	
	<code>*/*</code>	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.046Z" }</pre>	
403	Acceso denegado	No links
	Media type	



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<pre>/*</pre> <p>Example Value Schema</p> <pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.047Z" }</pre>	
404	Usuario no encontrado	<i>No links</i>
	<p>Media type</p> <pre>/*</pre> <p>Example Value Schema</p> <pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.048Z" }</pre>	

GET	/api/usuarios Obtener todos los usuarios	
Recupera la lista completa de usuarios registrados		



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Parameters

No parameters

Responses

Code	Description	Links
200	Lista obtenida exitosamente	<i>No links</i>
	Media type	
	<code>/*</code>	
	Controls Accept header.	
	Example Value Schema	
	<pre>[{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.053Z" }]</pre>	
500	Error interno del servidor	<i>No links</i>
	Media type	
	<code>/*</code>	
	Example Value Schema	
	<pre>[{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.054Z" }]</pre>	

POST /api/usuarios Crear nuevo usuario



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Crea un nuevo usuario en el sistema

Parameters

No parameters

Request body required

application/json

Example Value Schema

```
{  
    "nombre": "string",  
    "usuario": "string",  
    "apellido": "string",  
    "gmail": "string",  
    "contrasenna": "string",  
    "enabled": true,  
    "rolesIds": [  
        9007199254740991  
    ]  
}
```

Responses

Code	Description	Links
201	Usuario creado exitosamente	<i>No links</i>
	Media type	
	/*	
	Controls Accept header.	
	Example Value Schema	
	{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.058Z" }	
500	Error interno al crear usuario	<i>No links</i>
	Media type	
	/*	



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links						
<p>Example Value Schema</p> <pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.061Z" }</pre>								
<p>POST /api/usuarios/verificar-contraseña Verificar contraseña 🔒 ⌂</p> <p>Verifica si las credenciales proporcionadas son válidas</p> <p>Parameters Try it out</p> <p>No parameters</p> <p>Request body required application/json</p> <p>Example Value Schema</p> <pre>{ "additionalProp1": "string", "additionalProp2": "string", "additionalProp3": "string" }</pre> <p>Responses</p> <table border="1"><thead><tr><th>Code</th><th>Description</th><th>Links</th></tr></thead><tbody><tr><td>200</td><td>Resultado de la verificación entregado</td><td><i>No links</i></td></tr></tbody></table> <p>Media type /* <small>Controls Accept header.</small></p> <p>Example Value Schema</p> <pre>{</pre>			Code	Description	Links	200	Resultado de la verificación entregado	<i>No links</i>
Code	Description	Links						
200	Resultado de la verificación entregado	<i>No links</i>						



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
500	<p>Error durante la verificación</p> <p>Media type</p> <pre>/*</pre> <p>Example Value Schema</p> <pre>{ "additionalProp1": true, "additionalProp2": true, "additionalProp3": true }</pre>	No links

POST /api/usuarios/admin-create Crear usuario como admin

Permite a un ADMIN crear un usuario sin modificar la sesión actual ni devolver JWT.

Parameters

No parameters

Request body required

Example Value Schema

```
{
  "nombre": "string",
  "usuario": "string",
  "apellido": "string",
  "gmail": "string",
  "contrasenna": "string",
  "enabled": true,
  "rolesIds": [
    9007199254740991
  ]
}
```

Responses



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
201	Usuario creado exitosamente por admin	<i>No links</i>
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">*/*</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.071Z" }</pre>	
403	Acceso denegado	<i>No links</i>
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">*/*</div> <p>Example Value Schema</p> <pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.073Z" }</pre>	
500	Error interno al crear usuario	<i>No links</i>
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">*/*</div> <p>Example Value Schema</p> <pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.074Z" }</pre>	

GET	/api/usuarios/paginados Obtener usuarios paginados	



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Devuelve usuarios en formato paginado y ordenado

Parameters

Try it out

Name	Description
page <small>integer(\$int32) (query)</small>	<i>Default value : 0</i> 0
size <small>integer(\$int32) (query)</small>	<i>Default value : 10</i> 10
sort <small>string (query)</small>	<i>Default value : id</i> id
direction <small>string (query)</small>	<i>Default value : asc</i> asc

Responses

Code	Description	Links
200	Paginación exitosa	No links

Media type
/

Controls Accept header.

Example Value Schema

```
{  
    "totalElements": 9007199254740991,  
    "totalPages": 1073741824,  
    "pageable": {  
        "paged": true,  
        "pageNumber": 1073741824,  
        "pageSize": 1073741824,  
        "offset": 9007199254740991,  
        "sort": {  
            "sorted": true,  
            "empty": true,  
            "unsorted": true  
        }  
    }  
}
```



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
500	<p>Error interno del servidor</p> <p>Media type</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">*/*</div> <p>Example Value Schema</p> <pre>{ "totalElements": 9007199254740991, "totalPages": 1073741824, "pageable": { "paged": true, "pageNumber": 1073741824, "pageSize": 1073741824, "offset": 9007199254740991, "sort": { "sorted": true, "empty": true, "unsorted": true }, "unpaged": true }, "first": true, "last": true, "size": 1073741824, "content": [{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.084Z" }] }</pre>	No links

GET	/api/usuarios/buscar-por-usuario/{username}	
Buscar usuario por nombre de usuario		
Devuelve un usuario por su nombre de usuario si existe		
Parameters		Try it out



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Name	Description
username * required string (path)	<input type="text" value="username"/>

Responses

Code	Description	Links
200	Usuario encontrado	No links
	Media type <input type="text" value=" */*"/> Controls Accept header.	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.087Z" }</pre>	
404	Usuario no encontrado	No links
	Media type <input type="text" value=" */*"/>	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.089Z" }</pre>	

GET	/api/usuarios/buscar-por-email/{email}	
	Buscar usuario por correo electrónico	
	Devuelve un usuario por su correo electrónico si existe	



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Parameters

Try it out

Name	Description
email * required string (path)	email

Responses

Code	Description	Links
200	Usuario encontrado	No links
	Media type <code>*/*</code> Controls Accept header.	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.091Z" }</pre>	
404	Usuario no encontrado	No links
	Media type <code>*/*</code>	
	Example Value Schema	
	<pre>{ "id": 9007199254740991, "usuario": "string", "nombre": "string", "apellido": "string", "gmail": "string", "enabled": true, "fechaCreacion": "2025-06-10T13:34:18.093Z" }</pre>	

DELETE /api/usuarios/admin-delete/{id} Eliminar usuario como admin



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Permite a un ADMIN eliminar un usuario sin modificar la sesión actual ni devolver JWT.

Parameters

Try it out

Name	Description
id * required integer(\$int64) (path)	<input type="text" value="id"/>

Responses

Code	Description	Links
204	Usuario eliminado exitosamente por admin	No links
403	Acceso denegado	No links
404	Usuario no encontrado	No links
409	Usuario con libros asociados	No links
500	Error interno al eliminar usuario	No links

Archivos

Operaciones relacionadas con la gestión de archivos (subida, descarga y eliminación)

^

POST /api/files/upload/{tipo} Subir archivo

Sube una imagen al servidor, asociándola a un tipo (autor/libro)

Parameters

Try it out

Name	Description
------	-------------



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Name	Description	
tipo * required string (path)	tipo	
Request body		application/json
Example Value Schema		
{ "file": "string" }		
Responses		
Code	Description	Links
200	Archivo subido exitosamente Media type */ Controls Accept header.	No links
400	Formato de archivo inválido Media type */ Controls Accept header.	No links
500	Error al subir el archivo Media type	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
<pre>/*</pre>	<p>Example Value Schema</p> <pre>{ "additionalProp1": "string", "additionalProp2": "string", "additionalProp3": "string" }</pre>	

GET /api/files/{tipo}/{filename} Descargar archivo		🔒 ^
Descarga una imagen previamente almacenada		
Parameters		Try it out
Name	Description	
tipo * required string (path)	<input type="text" value="tipo"/>	
filename * required string (path)	<input type="text" value="filename"/>	
Responses		
Code	Description	Links
200	Archivo descargado exitosamente Media type <pre>/*</pre> Controls Accept header. Example Value Schema <code>string</code>	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
404	Archivo no encontrado Media type <div style="border: 1px solid black; padding: 2px; display: inline-block;">*/*</div> Example Value Schema <code>string</code>	No links

DELETE /api/files/{tipo}/{filename} Eliminar archivo

Elimina una imagen del sistema según su tipo y nombre

Parameters **Try it out**

Name	Description
tipo * required string (path)	<input type="text" value="tipo"/>
filename * required string (path)	<input type="text" value="filename"/>

Responses

Code	Description	Links
200	Archivo eliminado exitosamente Media type <div style="border: 2px solid green; padding: 2px; display: inline-block;">*/*</div> Controls Accept header. Example Value Schema	No links



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<pre>{ "additionalProp1": "string", "additionalProp2": "string", "additionalProp3": "string" }</pre>	
404	Archivo no encontrado	<i>No links</i>

Google Books

Operaciones relacionadas con la búsqueda e importación de libros desde Google Books

^

POST	/api/books/google/import/{googleBookId} Importar libro desde Google Books	
Importa un libro desde Google Books a la base de datos local		
Parameters		
		Try it out
Name	Description	
googleBookId * required string (path)	googleBookId	
Responses		
Code	Description	Links
200	Libro importado exitosamente	<i>No links</i>



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
	<p><code>/*</code></p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <p><code>string</code></p>	
400	Error en la solicitud	<i>No links</i>
	<p>Media type</p> <p><code>/*</code></p> <p>Example Value Schema</p> <p><code>string</code></p>	
500	Error al importar el libro	<i>No links</i>
	<p>Media type</p> <p><code>/*</code></p> <p>Example Value Schema</p> <p><code>string</code></p>	

GET /api/books/google/search Buscar libros en Google Books

Busca libros en Google Books API por título o palabras clave, con soporte para paginación

Parameters

Name	Description
query * required string (query)	query



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Name	Description	
maxResults <code>integer(\$int32) (query)</code>	<i>Default value : 10</i> 10	
startIndex <code>integer(\$int32) (query)</code>	<i>Default value : 0</i> 0	
Responses		
Code	Description	Links
200	Búsqueda realizada exitosamente Media type <code>/*</code> Controls Accept header.	<i>No links</i>
400	Parámetros de búsqueda inválidos Media type <code>/*</code>	<i>No links</i>



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

Code	Description	Links
Example Value Schema		
500	<pre>{ "totalItems": 1073741824, "items": [{ "id": "string", "volumeInfo": { "title": "string", "authors": ["string"], "publisher": "string", "publishedDate": "string", "description": "string", "industryIdentifiers": [{ "type": "string", "identifier": "string" }], "pageCount": 1073741824, "categories": ["string"], "imageLinks": { "smallThumbnail": "string", "thumbnail": "string", "small": "string" } } }] }</pre> Error al realizar la búsqueda	<i>No links</i>
Media type		
<pre>/*</pre>		
Example Value Schema		
	<pre>{ "totalItems": 1073741824, "items": [{ "id": "string", "volumeInfo": { "title": "string", "authors": ["string"], "publisher": "string", "publishedDate": "string", "description": "string", "industryIdentifiers": [{ "type": "string", "identifier": "string" }], "pageCount": 1073741824, "categories": ["string"], "imageLinks": { "smallThumbnail": "string", "thumbnail": "string", "small": "string" } } }] }</pre>	



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

OAuth2 Operaciones relacionadas con la autenticación OAuth2

[^](#)**GET****/api/v1/oauth2/config** Obtener configuración OAuth2[^](#)

Devuelve la configuración necesaria para la integración de OAuth2 en el cliente

Parameters[Try it out](#)

No parameters

Responses**Code****Description****Links**

200

Configuración obtenida con éxito

No links

Media type

/

Controls Accept header.

Example Value Schema

```
{  
    "additionalProp1": "string",  
    "additionalProp2": "string",  
    "additionalProp3": "string"  
}
```

Registro de usuarios

Endpoint para registrar nuevos usuarios y generar JWT

[^](#)**POST****/api/v1/register** Registrar nuevo usuario[^](#)

Registra un nuevo usuario y devuelve un JWT en una cookie HttpOnly.

Parameters[Try it out](#)



Swagger UI

<https://localhost:8443/swagger-ui/index.html#/Autore...>

No parameters

Request body required **application/json**

Example Value Schema

```
{  
  "usuario": "string",  
  "nombre": "string",  
  "apellido": "string",  
  "gmail": "string",  
  "contrasenna": "string"  
}
```

Responses

Code	Description	Links
201	Usuario registrado exitosamente Media type /* Controls Accept header. Example Value Schema { "token": "string", "mensaje": "string" }	No links
500	Error al registrar usuario Media type /* Example Value Schema { "token": "string", "mensaje": "string" }	No links

Schemas



Manual de despliegue

Contexto del despliegue

Esquema general

- El proyecto está formado por:
 - Un backend (Java/Spring)
 - Un frontend (Angular)
 - Servicios auxiliares
- El despliegue se realiza mediante contenedores Docker usando **docker-compose**, lo que permite levantar el entorno de forma rápida y consistente.
- Jenkins automatiza el pipeline de CI/CD, permitiendo que el despliegue en AWS sea automático y controlado.
- **Infraestructura en AWS:**
 - **EC2** para ejecutar la aplicación.
 - **RDS** para la base de datos.
 - **S3** para almacenar las imágenes subidas por los usuarios.

Docker

Actualmente, se utilizan tres archivos **docker-compose** diferentes:

- **docker-compose.des.yml**: Usado para desarrollo local. Incluye un archivo `.env` con variables de desarrollo y un servicio de base de datos MariaDB local, evitando dependencias externas.
- **docker-compose.prod.yml**: Usado en producción (AWS). Define solo el backend y el frontend; utiliza un `.env` específico para AWS.
- **docker-compose.yml**: Se emplea para desplegar el contenedor de Jenkins, encargándose del despliegue en AWS.

Cada componente del proyecto dispone de su propio `Dockerfile` para la construcción de su imagen correspondiente.



Nginx

- Nginx se utiliza en el frontend para servir los archivos estáticos de Angular y redirigir las peticiones API al backend, permitiendo así una gestión centralizada del tráfico.

Jenkins

- Jenkins se implementa con su propio docker-compose. Se encarga de:
 - Detectar cada push.
 - Conectarse a AWS.
 - Ejecutar/desplegar los contenedores de producción en la instancia EC2 de forma automática.

AWS

- **RDS:** Permite disponer de una base de datos MariaDB gestionada, encargándose de backups y administración, únicamente es necesario configurar la conexión en el backend.
- **S3:** Utilizado para almacenar todas las imágenes subidas por los usuarios, optimizando el almacenamiento y el acceso seguro a ese contenido.
- **EC2:** Aquí se ejecutan los contenedores del backend y frontend. Jenkins realiza el despliegue automático en esta instancia.
- **IP Elástica:** Se utiliza una IP elástica en la instancia EC2, asegurando que la IP pública no cambie, facilitando el apuntado del dominio.

Dominio

- El dominio se apunta, mediante DNS, directamente a la IP pública de la instancia EC2.
- Los certificados SSL se gestionan con Let's Encrypt a través del script `setup-letsencrypt.sh` situado en la carpeta `certs/`, permitiendo disponer de HTTPS gratuito y renovable automáticamente.



Pasos de la creación del despliegue

En esta explicación, voy a comentar como ha sido el montaje del despliegue para producción en AWS, por lo que no comentaré nada sobre el docker-compose.des.yml de desarrollo local.

Docker

Lo primero que configuraremos será Docker, en este caso.

Lo que deberemos crear para que funcione nuestro Docker compose, será crear primero los Dockerfile en nuestras rutas raíz de nuestros repositorios, uno para el back-end y otro para el front-end.

Quedando algo así:

Front-end:

```
└── TFG_Readora_Frontend
    ├── docs
    └── src
        └── readora
            ├── .angular
            ├── .vscode
            ├── docs
            ├── node_modules
            ├── public
            ├── src
            │   ├── .editorconfig
            │   ├── .gitignore
            │   └── angular.json
            └── Dockerfile
```



Back-end:

```
TFG_Readora_Backend
├── docs
└── src/readora
    ├── .mvn
    ├── src
    ├── target
    ├── .gitattributes
    ├── .gitignore
    └── Dockerfile
    └── mvnw
    └── mvnw.cmd
    └── pom.xml
    └── README.md
```

Una vez creado los archivos deberemos configurarlos correctamente:
Construiremos Dockerfile simples, el único detalle que añadiremos es, la descarga de un script llamado wait-for-it, el cual nos facilita la espera entre servicios en el docker-compose.

Empecemos por el front-end:

```
# Etapa de compilación
FROM node:20 AS build

# Establece el directorio de trabajo dentro del contenedor
WORKDIR /app

# Copia los archivos de dependencias primero para aprovechar la caché de
Docker
COPY package*.json ./

# Instala Angular CLI globalmente y las dependencias del proyecto
RUN npm install -g @angular/cli && npm install

# Copia el resto del código fuente
COPY . .
```



```
# Compila la aplicación para producción
RUN ng build --configuration production

# Etapa de producción con Nginx
FROM nginx:stable-alpine

# Instala wget para descargar wait-for-it
RUN apk add --no-cache wget bash

# Descarga wait-for-it.sh y hazlo ejecutable
RUN wget -O /wait-for-it.sh
https://raw.githubusercontent.com/vishnubob/wait-for-it/master/wait-for-it.sh \
    && chmod +x /wait-for-it.sh

# Copia la configuración personalizada de Nginx
COPY nginx.conf /etc/nginx/nginx.conf

# Copia los archivos compilados desde la etapa de compilación
COPY --from=build /app/dist/readora/browser /usr/share/nginx/html

# Crea el directorio para uploads si no existe
RUN mkdir -p /uploads

EXPOSE 80

# Inicia Nginx en primer plano
CMD ["nginx", "-g", "daemon off;"]
```

En el back-end:

```
# Imagen base con JDK 21 de Eclipse Temurin
FROM eclipse-temurin:21-jdk

# Instala wget para descargar wait-for-it
RUN apt-get update && apt-get install -y wget

# Descarga wait-for-it.sh y hazlo ejecutable
RUN wget -O /wait-for-it.sh
https://raw.githubusercontent.com/vishnubob/wait-for-it/master/wait-for-it.sh \
    && chmod +x /wait-for-it.sh
```



```
# Establece el directorio de trabajo dentro del contenedor
WORKDIR /app

# Copia los archivos necesarios para Maven
COPY .mvn/ .mvn
COPY mvnw .
COPY pom.xml .

# Da permisos de ejecución al script de Maven
RUN chmod +x mvnw

# Resuelve las dependencias del proyecto y verifica que Maven está
# funcionando
RUN ./mvnw --version && ./mvnw dependency:resolve

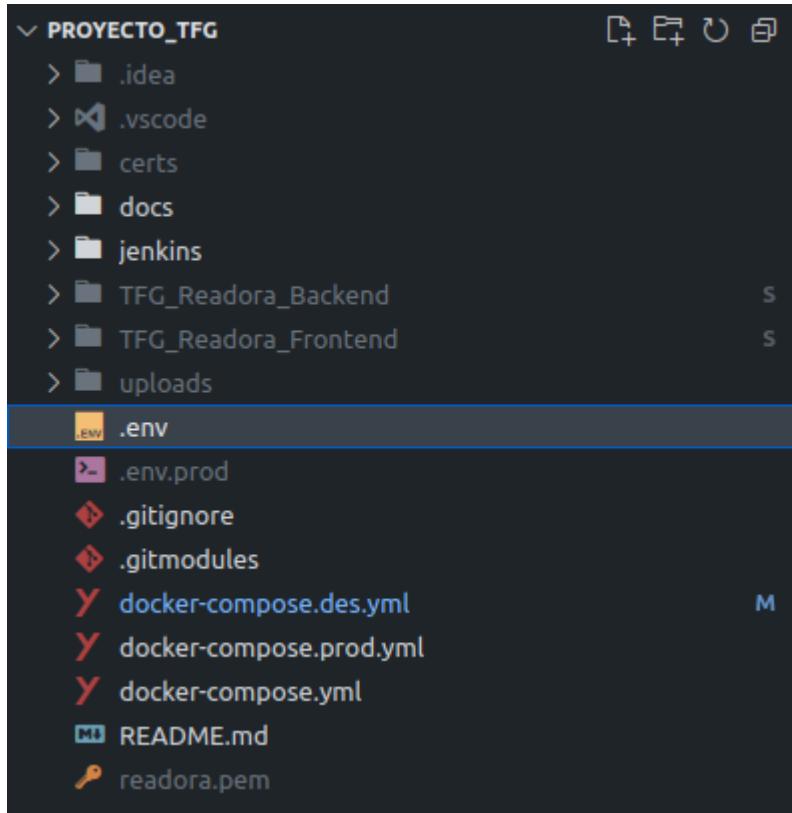
# Copia el código fuente
COPY src ./src

# Compila el proyecto y crea el JAR ejecutable con todas las dependencias
RUN ./mvnw clean package -DskipTests

# Expone el puerto 8080 para la API REST
EXPOSE 8080

# Comando para ejecutar la aplicación
CMD ["java", "-jar", "target/readora-1.0.0.jar"]
```

Una vez creado los dos Dockerfile nos iremos a crear el archivo .env, este deberá estar en la ruta padre del proyecto:



El archivo .env tendrá una configuración privada, pero puedo mostrar un ejemplo del contenido:

```
# Configuración de Base de Datos AWS
DB_URL=====
DB_ROOT_PASSWORD=====
DB_DATABASE=====
DB_USER=====
DB_PASSWORD=====
DB_DRIVER=====

# JWT y Seguridad (Valores sensibles)
JWT_SECRET=====
JWT_KEYSTORE_PATH=====
JWT_KEYSTORE_PASSWORD=====

# URLs y Rutas
CORS_ALLOWED_ORIGINS=====

# Configuración S3 para almacenamiento en AWS
```



```
STORAGE_TYPE==*****
AWS_REGION==*****
AWS_BUCKET_NAME==*****
AWS_BUCKET_URL==*****  
  
# Credenciales AWS  
# Hay que modificarlo cada sesion de AWS Academy  
AWS_ACCESS_KEY_ID==*****  
AWS_SECRET_ACCESS_KEY==*****  
AWS_SESSION_TOKEN==*****  
  
# SSL Configuration  
SSL_ENABLED==*****  
SSL_KEYSTORE_PATH==*****  
SSL_KEYSTORE_PASSWORD==*****  
SSL_KEY_ALIAS==*****  
SSL_KEYSTORE_TYPE==*****  
  
# Google Auth  
GOOGLE_API_KEY==*****  
GOOGLE_CLIENT_ID==*****  
GOOGLE_CLIENT_SECRET==*****  
  
# Frontend  
FRONTEND_URL==*****  
FRONTEND_REDIRECT_URL==*****
```

Este .env es un ejemplo y no el real, ya que hay configuraciones diferentes en local que en el entorno AWS, pese a que en este ejemplo las he incluido todas

Tras hacer esto podremos pasar a configurar el docker-compose el cual estará en la misma ubicación del .env anteriormente creado, por lo que no mostraré de nuevo la ruta.

Para configurar este docker-compose hemos creado dos servicios que están correctamente configurados en la misma red de Docker
Como consideraciones especiales, aun que ya comentado anteriormente hemos configurado un script para esperar que el servicio este listo para arrancar, y así evitar posibles errores

```
services:
```



```
backend:
  build:
    context: ./TFG_Readora_Backend/src/readora
  container_name: readora-backend
  command: >
    sh -c "java -jar target/readora-1.0.0.jar"
  ports:
    - "8443:8443"
  env_file: .env.prod
  volumes:
    - ./certs:/app/certs:ro
    - ./env.prod:/app/.env
  networks:
    - readora-network
  restart: always

frontend:
  build:
    context: ./TFG_Readora_Frontend/src/readora
  container_name: readora-frontend
  command: >
    sh -c "/wait-for-it.sh backend:8443 --timeout=60 -- nginx -g 'daemon off;'"
  depends_on:
    - backend
  ports:
    - "80:80"
    - "443:443"
  volumes:
    - ./certs:/etc/nginx/certs:ro
  env_file: .env.prod
  networks:
    - readora-network
  restart: always

networks:
  readora-network:
    driver: bridge
```

Con todo esto habríamos terminado con Docker, ya que el despliegue real en AWS lo hacemos con Jenkins.

Nginx



Ya que nginx tiene un uso muy simple y no se complica, no tengo pensado explicarla más de lo necesario

En el front-end, hay un archivo de configuración de Nginx llamado nginx.conf en la ruta base del front-end. Este archivo define cómo Nginx debe servir la aplicación Angular.

Cuando se construye la imagen Docker del front-end, el Dockerfile copia tanto la aplicación Angular ya compilada como el archivo nginx.conf dentro de la imagen. Así, cuando se ejecuta el contenedor, Nginx usa esa configuración para servir la aplicación.

AWS

Vamos a crear Amazon web service antes que Jenkins, ya que así podremos configurar Jenkins de una y seguiremos un despliegue en orden sin muchos quebraderos de cabeza.

Empecemos con el RDS.

Lo primero es definir su uso: “**RDS (Relational Database Service)** es el servicio de base de datos que ofrece AWS”. En lugar de instalar una base de datos dentro de la instancia EC2, usamos RDS para alojar una base de datos, en nuestro caso MariaDB.

Para crearla nos dirigiremos a el apartado en AWS de RDS:



Q rds X

Servicios Ver los 12 resultados

- Aurora and RDS** Servicio de bases de datos relacionales administrado
- AWS Glue DataBrew** Herramienta de preparación de datos visuales para limpiar y normalizar datos para análisis
- Database Migration Service** Servicio de migración de bases de datos administrado

Características Ver los 16 resultados

- Información sobre bases de datos**
 - Función CloudWatch
- Instancias reservadas**
 - Función Aurora and RDS
- Proxies**
 - Función Aurora and RDS

Recursos / para una búsqueda enfocada

¿Fueron útiles estos resultados? X

Presentación de la búsqueda de recursos
Para buscar recursos, el Explorador de recursos debe estar activo en al menos una región de AWS y debe tener permiso para usar la vista predeterminada en la cuenta.
[Más información](#)

Seleccionamos Aurora and RDS, esto nos lleva a un panel de configuración, donde deberemos pulsar “Crear una base de datos”.



Readora - Javier Moreno Salas

The screenshot shows the AWS RDS console for Aurora and RDS. The main dashboard provides an overview of resources in the US East (N. Virginia) region. Key sections include:

- Recursos:** Shows 1 instance of Aurora and 0 clusters.
- Servicios recomendados:** A section for recommended services, currently empty.
- Información adicional:** Links to documentation and support resources.
- Database Preview Environment:** Information about the preview environment.

Tras haber pulsado el botón, esto nos llevará a un apartado para configurar el RDS, en mi caso explicaré lo que debemos configurar para nuestro ejemplo.

The screenshot shows the 'Create database' wizard in the AWS RDS console. The process is titled 'Crear base de datos' and is currently at the step 'Elegir un método de creación de base de datos'. The 'MariaDB' option is selected. Other options shown include Aurora (MySQL Compatible), Aurora (PostgreSQL Compatible), MySQL, PostgreSQL, Oracle, Microsoft SQL Server, and IBM Db2. To the right, detailed information about MariaDB is provided, including its compatibility with MySQL and its performance characteristics.

Configuraremos una base de datos mariaDB, ademas de configurar un nombre para la bd, y un nombre y contraseña de usuario maestro. Pulsamos Crear.

Y con esto estaría por el momento ya que una vez que creamos la instancia EC2 volveremos aqui.

Seguimos con el S3



Nos dirigimos a S3 de AWS:

The screenshot shows the AWS search interface with the query 's3' entered. The results are categorized into 'Servicios' (Services), 'Características' (Features), and 'Recursos' (Resources).

Servicios (9)

- S3** Almacenamiento escalable en la nube
- S3 Glacier** Almacenamiento de archivos en la nube
- AWS Snow Family** Transporte de datos a gran escala

Características (40)

- Importaciones de S3**
 - Función DynamoDB
- Características destacadas**
 - Función S3
- Analizador de acceso de IAM para S3**
 - Función S3

Recursos / para una búsqueda enfocada

¿Fueron útiles estos resultados?

Publicaciones de blog Ver los 35 resultados

Seleccionamos S3, esto nos abrirá un panel de configuración donde deberemos pulsar en crear bucket.



Readora - Javier Moreno Salas

The screenshot shows the AWS Management Console interface for Amazon S3. The top navigation bar includes links for Aurora and RDS, S3, EC2, and Amazon S3. A search bar and a 'Buscar' button are also present. The main content area displays an 'Instantánea de la cuenta: actualizada cada 24 horas' (CloudWatch Metrics) for all regions. Below this, the 'Buckets de uso general' section lists one bucket: 'readora-uploads'. The bucket details show it was created on June 2, 2025, at 5:03:13 PM CEST, located in 'EE.UU. Este (Norte de Virginia) us-east-1'. There are buttons for 'Copiar ARN', 'Vaciar', 'Eliminar', and 'Crear bucket'.

El bucket por el momento no tiene mucha configuración, modificaremos el nombre al gusto y le daremos a crear.

The screenshot shows the 'Crear bucket' (Create Bucket) wizard. Step 1: Configuration general. It asks for the Region of AWS (selected: EE.UU. Este (Norte de Virginia) us-east-1). It then asks for the Type of bucket, with 'Uso general' (General Purpose) selected. Other options include 'Directorio' (Directory). The Name of the bucket field contains 'myownbucket'. Below this, there's a 'Copy the configuration of the existing bucket' (Copy configuration of existing bucket) section with an optional checkbox. The next section is 'Propiedad de objetos' (Object Properties), which includes 'ACL deshabilitadas (recomendado)' (ACL disabled (recommended)) and 'ACL habilitadas' (ACL enabled). The final section is 'Propiedad del objeto' (Object Properties), stating 'Aplicada al propietario del bucket' (Applied to owner of bucket).

Seguiremos con el EC2

Lo primero como en los otros pasos es dirigirnos al EC2



Q ec2 X

Servicios

Ver los 13 resultados

- Servicios (13)**
 - Características (60)
 - Recursos New
 - Publicaciones de blog (46)
 - Documentación (4526)
 - Artículos de conocimiento (481)
 - Tutoriales (19)
 - Marketplace (2997)
- EC2** Servidores virtuales en la nube
- EC2 Image Builder** Un servicio administrado para automatizar la creación, personalización e implementaci...
- EC2 Global View** EC2 Global View provides a global dashboard and search functionality that lets you fin...

Características

Ver los 60 resultados

- EC2 Instances**
 - Función CloudWatch
- Panel**
 - Función EC2
- AMI**
 - Función EC2

Recursos / para una búsqueda enfocada

¿Fueron útiles estos resultados?

Sí No

① Presentación de la búsqueda de recursos

Para buscar recursos, el Explorador de recursos debe estar activo en al menos una región de AWS y debe tener permiso para usar la vista predeterminada en la cuenta.

Más información [?]

X

Publicaciones de blog

Ver los 46 resultados

Ya en el panel del EC2 pulsaremos en lanzar instancia, esto nos abrirá la configuración de la instancia.



Readora - Javier Moreno Salas

The screenshot shows the AWS Management Console with the EC2 service selected. The main pane displays a table of instances. One instance is listed: 'readora-server' (ID: i-0f572ebdb5d81d8a5), which is currently running in the 'En ejecución' state. The instance type is 't2.micro'. Other columns include 'Comprobación de' (2/2 comprobador), 'Estado de la al.' (t2.micro), 'Zona de dispon...', 'DNS de IPv4 pública' (ec2-3-218-179-158.co...), 'Dirección IP...' (3.218.179.158), and 'IP elástica' (3.218.179.158). The left sidebar has sections for 'Panel', 'Vista global de EC2', 'Eventos', 'Instancias' (selected), 'Tipos de Instancia', 'Plantillas de lanzamiento', 'Solicitudes de spot', 'Savings Plans', 'Instancias reservadas', 'Alojamientos dedicados', 'Reservas de capacidad', 'Imagenes' (selected), 'AMIs', 'Catálogo de AMI', 'Elastic Block Store', 'Volúmenes', 'Instantáneas', 'Administrador del ciclo de vida', 'Red y seguridad', 'Security Groups', 'Direcciones IP elásticas', 'Grupos de ubicación', and 'Pares de claves'. At the bottom, there are links for 'CloudShell', 'Comentarios', and copyright information.

Lo primero que configuraremos en el EC2 es el nombre que se podrá elegir al gusto, y el SO que usaremos será Ubuntu, ya que tenemos más conocimientos del mismo.

The screenshot shows the 'Launch a new instance' wizard. The first step, 'Nombre y etiquetas', is active. It asks for a name ('prueba') and provides an option to 'Agregar etiquetas adicionales'. The second step, 'Imágenes de aplicaciones y sistemas operativos (Imagen de máquina de Amazon)', is shown below. It lists various AMI options, with 'Ubuntu Server 24.04 LTS (HVM), SSD Volume Type' selected. This option is described as being 'Apto para la capa gratuita'. To the right, a 'Resumen' (Summary) box contains details about the selected AMI: 'Canonical, Ubuntu, 24.04, amd64...más información' (ami-0731becdfb322781e), 'Tipo de servidor virtual (tipo de instancia)' (t2.micro), 'Firewall (grupo de seguridad)' (Nuevo grupo de seguridad), and 'Almacenamiento (volúmenes)' (Volúmenes: 1 (8 GiB)). A callout box provides more information about the free tier: 'Nivel gratuito: Durante el primer año que abre una cuenta de AWS, obtiene 750 horas al mes de uso de instancias t2.micro (o t3.micro cuando t2.micro no esté disponible) si se utiliza con AMI de nivel gratuito. 750 horas al mes de uso de direcciones IPv4 públicas, 30 GiB de almacenamiento de EBS, 2 millones de E/S, 1 GB de instantáneas y 100 GB de ancho de banda de Internet.' At the bottom, there are 'Cancelar', 'Lanzar instancia', and 'Código de versión preliminar' buttons.

Deberemos crear un par de claves rsa en formato .pem, que deberemos guardar, ya que la necesitamos a futuro.



The screenshot shows the 'Launch instance' wizard. In the 'Type of instance' step, an 't2.micro' instance is selected. In the 'Key pair' step, a new RSA key pair is being created with the name 'ED25519'. The 'Create key pair' button is highlighted. The 'Network & security' step is partially visible in the background.

En el tipo de instancia usaremos t2.micro, ya que es el único que poseemos gratis.

En la configuración de red deberemos pulsar editar y modificar algunos campos, teniéndonos que quedar algo como esto:

The screenshot shows the 'Network & security' step of the launch wizard. It displays four security groups with their respective rules. A tooltip at the bottom left of the interface says: 'Las reglas con origen 0.0.0.0/0 permiten que todas las direcciones IP tengan acceso a la instancia. Le recomendamos que configure las reglas del grupo de seguridad para permitir el acceso únicamente desde direcciones IP conocidas.' The 'Launch instance' button is highlighted.

Con todo esto podremos lanzar la instancia.

Ahora configuraremos la IP elástica para tener una IP fija en el EC2 pese a que lo apaguemos, reiniciemos o lo que necesitemos.



Esto se hará desde el mismo panel de control de EC2

The screenshot shows the AWS EC2 Home page. On the left, a sidebar lists various services: Instancias reservadas, Alojamientos dedicados, Reservas de capacidad, Imágenes (AMI Catalogo de AMI), Elastic Block Store (Volumenes Instantáneas Administrador del ciclo de vida), Red y seguridad (Security Groups Direcciones IP elásticas Grupos de ubicación Pares de claves Interfaces de red), Equilibrio de carga (Balanceadores de carga Grupos de destino Trust Stores), Auto Scaling (Grupos de Auto Scaling Configuration), and CloudWatch Metrics.

The main content area displays the following sections:

- Recursos:** Shows 0 instances (en ejecución), 0 load balancers, 0 capacity reservations, 5 elastic IPs, 0 auto scaling groups, 0 host dedicated instances, 0 instant instances, 0 key pairs, 0 security groups, 0 volumes, and 0 instance types.
- Atributos de la cuenta:** Displays VPC predeterminada (vpc-00c102a6d9f94584) and Configuration (Protection and security of data, AMI permissions, Zones, Console of series of EC2, Specification of credit default, Preferences of the console of EC2).
- Lanzar la instancia:** Buttons for "Lanzar la instancia" and "Migrar un servidor". A note says "Nota: Sus instancias se lanzarán en la región Estados Unidos (Norte de Virginia)".
- Estado del servicio:** Shows the service is running ("Este servicio funciona con normalidad").
- Zonas:** A table listing availability zones (us-east-1a, us-east-1b, us-east-1c, us-east-1d, us-east-1e) and their respective AZ codes (use1-az4, use1-az6, use1-az1, use1-az2, use1-az5).
- Información adicional:** Links to Instructions para comenzar, Guía de Introducción, Documentación, Todos los recursos de EC2, Foros, Precios, and Póngase en contacto con nosotros.

At the bottom, the URL is us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Addresses: and the page footer includes © 2025, Amazon Web Services, Inc. o sus filiales. Privacidad Términos Preferencias de cookies.

Esto no tiene mucha configuración, por lo que los pasos son sencillos, deberemos dirigirnos al crear IP elástica y la asignaremos al EC2 anteriormente creado.

The screenshot shows the AWS EC2 Elastic IP Addresses page. The sidebar is identical to the previous screenshot. The main content area shows a table of assigned elastic IPs:

Name	Dirección IPv4 asignada	Tipo	ID de asignación	Registro DNS inverso	ID de instancia asociada	Dirección IP privada
-	18.205.208.190	IP pública	eipalloc-0d970172ff11ae6c	-	-	-
-	3.218.179.158	IP pública	eipalloc-095a2194258915ab2	-	I-0f572ebd85d81d8a5	172.31.82.19
-	3.237.54.95	IP pública	eipalloc-0b64f952bdac03ac6	-	-	-
-	54.159.72.104	IP pública	eipalloc-0511314dca7546d7c	-	-	-
-	54.225.127.69	IP pública	eipalloc-01e83d0003db42e94	-	-	-

Below the table, there is a section titled "Seleccionar una dirección IP elástica" with a note: "Vea el uso de las direcciones IP y las recomendaciones para liberar las IP no utilizadas con Public IP insights".

At the bottom, the URL is us-east-1.console.aws.amazon.com/ec2/elasticips?region=us-east-1#Addresses: and the page footer includes © 2025, Amazon Web Services, Inc. o sus filiales. Privacidad Términos Preferencias de cookies.



Readora - Javier Moreno Salas

The screenshot shows the 'Assign elastic IP address' step in the AWS EC2 console. It's a multi-step process where you select the instance and then assign an IP. This is the second step.

Todavía no hemos terminado con el EC2 al completo pero ya con esto creado vamos a dejar terminado el RDS.

Para esto editaremos el RDS, y indicaremos la instancia anteriormente creada.

The screenshot shows the 'Modify database instance' step in the AWS RDS console. It's a multi-step process where you select the instance and then modify its settings. This is the second step.

Tanto en el RDS como en el EC2 deberán estar configurados el mismo grupo de seguridad.



Ahora vamos a terminar de configurar el EC2 para esto usaremos el .pem creado en la instancia y nos conectaremos por ssh, para tener el comando listo en un momento nos iremos al panel del EC2 y pulsaremos en conectar. El requisito es que la instancia debe estar encendida.

The screenshot shows the AWS CloudShell interface. At the top, there's a navigation bar with 'aws' and various service icons like Aurora and RDS, S3, and EC2. Below it, the path 'EC2 > Instancias > i-0f572ebd05d81d8a5 (readora-server) > Conectarse a la instancia' is visible. The main area is titled 'Conectar Información' with the sub-section 'Conexión de la instancia EC2'. It shows the instance ID 'i-0f572ebd05d81d8a5 (readora-server)'. Below this, a numbered list of steps for connecting via SSH is provided:

1. Abra un cliente SSH.
2. Localice el archivo de clave privada. La clave utilizada para lanzar esta instancia es readora.pem.
3. Ejecute este comando, si es necesario, para garantizar que la clave no se pueda ver públicamente.
chmod 400 'readora.pem'
4. Conéctese a la instancia mediante su DNS público:
ssh -i "readora.pem" ubuntu@ec2-3-218-179-158.compute-1.amazonaws.com

Below the steps, there's an example command: 'ssh -i "readora.pem" ubuntu@ec2-3-218-179-158.compute-1.amazonaws.com'. A note at the bottom states: 'Nota: En la mayoría de los casos, el nombre de usuario adivinado es correcto. Sin embargo, lea las instrucciones de uso de la AMI para comprobar si el propietario de la AMI ha cambiado el nombre de usuario predeterminado de la AMI.'

At the bottom right of the interface, there's a 'Cancelar' button.

Copiaremos el comando ssh de ejemplo ya que nos servirá más adelante. Una vez con este comando y con el .pem en la ruta donde estés lanzaremos ese comando.

The screenshot shows a terminal window with a dark theme. On the left, there's a file explorer sidebar showing a directory structure with files like '.vscode', 'certs', 'docs', 'jenkins', 'TFG_Readera_Backend', 'TFG_Readera_Frontend', 'uploads', 'env', 'env.prod', '.gitignore', '.gitmodules', 'docker-compose.desyml', 'docker-compose.prod.yml', 'dockercompose.yml', 'README.md', and 'readora.pem'. The main terminal window has tabs for 'PROBLEMAS', 'SALIDA', 'CONSOLA DE DÉBAGACIÓN', 'TERMINAL', and 'PUERTOS'. The 'TERMINAL' tab is active, displaying the following text:

```
o ./Documentos/Escuela/TFG/main > ssh -i "readora.pem" ubuntu@ec2-3-218-179-158.compute-1.amazonaws.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1029-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System Information as of Fri Jun 6 16:47:26 UTC 2025
System load: 0.63 Processes: 125
Usage of /: 21% of 23.17GB Users logged in: 0
Memory usage: 67%
Swap usage: 0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Fri Jun 6 15:18:59 2025 from 84.78.245.185
ubuntu@ip-172-31-82-19:~$
```

At the bottom of the terminal window, there's a status bar with icons for Maven, Project_TFG, main, Git Graph, Go Live, and other system status indicators.



Una vez aquí deberemos instalar varias cosas y configurar otras, daré un comando para automatizar todo esto pero principalmente necesitamos configurar un swap permanente ya que poseemos muy poca memoria, debemos instalar Docker y docker-compose y algunas utilidades.

```
# Actualizar sistema
sudo apt update && sudo apt upgrade -y

# Instalar paquetes útiles y dependencias
sudo apt install -y \
    git \
    curl \
    wget \
    unzip \
    ca-certificates \
    gnupg \
    lsb-release \
    software-properties-common

# Instalar Docker
curl -fsSL https://get.docker.com -o get-docker.sh && sh get-docker.sh

# Instalar Docker Compose
sudo apt install -y docker-compose

# Habilitar Docker para que arranque automáticamente
sudo systemctl enable docker
sudo systemctl start docker

# Agregar tu usuario actual al grupo docker para no usar sudo
sudo usermod -aG docker $USER

# Instalar OpenJDK 21
sudo apt install -y openjdk-21-jdk

# Crear archivo swap de 2GB
sudo fallocate -l 2G /swapfile
sudo chmod 600 /swapfile
sudo mkswap /swapfile
sudo swapon /swapfile
```



```
# Hacer que la swap persista tras reinicio
echo '/swapfile none swap sw 0 0' | sudo tee -a /etc/fstab
```

Con esto tendremos el EC2 correctamente configurado.

Ahora pasaremos a Jenkins, ya que se encarga del despliegue

Jenkins

Para configurar Jenkins e instalar Jenkins principalmente hemos creado un Dockerfile y un docker compose donde se despliega

Dockerfile:

```
FROM jenkins/jenkins:lts-jdk21

USER root

# Instalar herramientas necesarias
RUN apt-get update && \
    apt-get -y install \
    apt-transport-https \
    ca-certificates \
    curl \
    gnupg2 \
    git \
    wget \
    unzip \
    zip \
    maven \
    software-properties-common

# Instalar Docker CLI
RUN curl -fsSL https://download.docker.com/linux/debian/gpg | apt-key add - \
&& \
    echo "deb [arch=amd64] https://download.docker.com/linux/debian $(grep \
VERSION_CODENAME /etc/os-release | cut -d= -f2) stable" > \
/etc/apt/sources.list.d/docker.list && \
    apt-get update && \
    apt-get install -y docker-ce-cli

# Instalar Node.js
```



```
RUN curl -fsSL https://deb.nodesource.com/setup_20.x | bash - && \
apt-get install -y nodejs

# Copiar el archivo de plugins
COPY plugins.txt /usr/share/jenkins/ref/plugins.txt

# Instalar los plugins listados
RUN jenkins-plugin-cli --plugin-file /usr/share/jenkins/ref/plugins.txt

# Crear directorio para volumen
RUN mkdir -p /var/jenkins_home/workspace

# Instala rsync para permitir despliegue desde Jenkins
RUN apt-get update && apt-get install -y rsync && rm -rf
/var/lib/apt/lists/*

# Cambiar a usuario jenkins
USER jenkins
```

Docker-compose:

```
services:
  jenkins:
    build:
      context: ./jenkins
      dockerfile: Dockerfile-jenkins
    container_name: jenkins
    restart: always
    ports:
      - "8080:8080"
      - "50000:50000"
    volumes:
      - jenkins_home:/var/jenkins_home
      - /var/run/docker.sock:/var/run/docker.sock
      - ./jenkins/Jenkinsfile:/var/jenkins_home/Jenkinsfile:ro
    env_file: .env
    environment:
      - JENKINS_ADMIN_ID=${JENKINS_ADMIN_ID}
      - JENKINS_ADMIN_PASSWORD=${JENKINS_ADMIN_PASSWORD}
    networks:
      - readora-network

networks:
```



```
readora-network:  
  driver: bridge
```

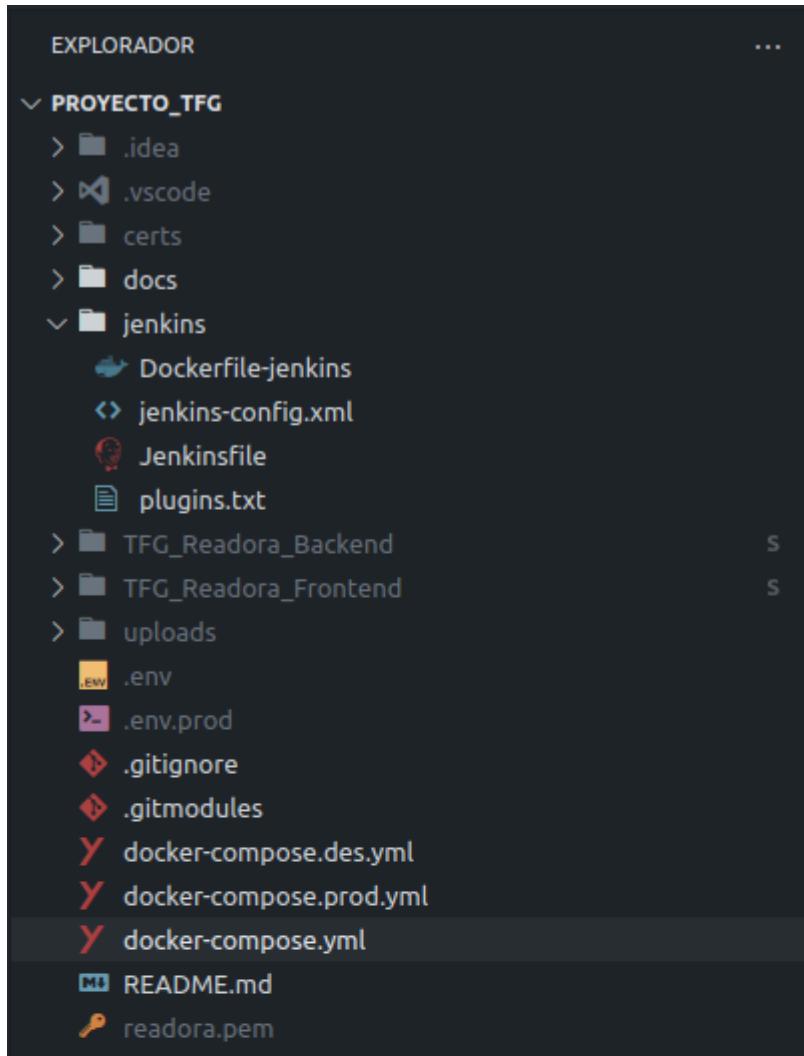
```
volumes:  
  jenkins_home:
```

Tendremos que configurar el jenkinsfile pero antes probaremos y configuraremos las credenciales que hemos configurado:

Nos dirigimos al puerto donde se está corriendo Jenkins, en mi caso es <http://localhost:8080/>, y en la interfaz nos dirigimos a administrar Jenkins > credenciales aquí en global crearemos una con los datos del .pem y otro en mí con las credenciales de GitHub ya que el repositorio a usar es privado

Type	Store	Domain	ID	Name
SSH	System	(global)	33b77252-d263-434a-9ed4-32299d38f19e	JavierMoren/*********
EC2	System	(global)	readora-ec2-key	ubuntu-ec2

El jenkinsfile igual que los archivos de configuración de Jenkins lo crearemos en una carpeta llamada Jenkins que va en la ruta padre del repositorio.



El jenkinsfile configurara el despliegue en el ec2 para eso configuraremos el jenkinsfile de esta manera:

```
pipeline {
    agent any

    parameters {
        choice(name: 'MAIN_BRANCH', choices: ['main'], description: 'Rama a
        usar para el repositorio principal')
    }

    environment {
        EC2_USER = 'ubuntu'
```



```
EC2_HOST = 'ec2-3-218-179-158.compute-1.amazonaws.com'
REPO_URL = 'git@github.com:JavierMoren/Proyecto_TFG.git'
PROJECT_DIR = 'Proyecto_TFG'
AWS_BUCKET_NAME = 'readora-uploads'
AWS_REGION = 'us-east-1'
}

stages {
    stage('Checkout') {
        steps {
            checkout([
                $class: 'GitSCM',
                branches: [[name: "${params.MAIN_BRANCH}"]],
                extensions: [[$class: 'SubmoduleOption',
                    disableSubmodules: false,
                    parentCredentials: true,
                    recursiveSubmodules: true
                ]],
                userRemoteConfigs: [
                    url:
'https://github.com/JavierMoren/Proyecto_TFG.git',
                    credentialsId: '33b77252-d263-434a-9ed4-32299d38f19e'
                ]
            ])
            sh '''
                git submodule update --init --recursive
                echo "Verificando estructura de directorios después del
checkout:"
                find . -type d -name "readora" | sort
            '''
        }
    }

    stage('Despliegue') {
        steps {
            script {
                // --- Despliegue en AWS EC2 ---
                sshagent(['readora-ec2-key']) {
                    sh '''
                        # Sincronizar archivos al EC2 con más información de
depuración
                        echo "Iniciando sincronización con rsync..."
                        rsync -avz --delete \
                            --exclude 'jenkins/' \
                    '''
                }
            }
        }
    }
}
```



```
--exclude '.git/' \
--exclude '.gitignore' \
--exclude 'README.md' \
--exclude 'uploads/' \
--exclude 'certs/' \
--exclude '.env.prod' \
-e "ssh -o StrictHostKeyChecking=no" \
./ $EC2_USER@$EC2_HOST:~/Proyecto_TFG/

# Verificar y preparar entorno en EC2
echo "Verificando y preparando entorno en EC2..."
ssh -o StrictHostKeyChecking=no $EC2_USER@$EC2_HOST \
    set -e
    cd ~/Proyecto_TFG
    if [ ! -d "TFG_Readora_Backend/src/readora" ];
then
        echo "ERROR: No se encuentra el directorio
TFG_Readora_Backend/src/readora"
        find TFG_Readora_Backend -type d | sort
        exit 1
    fi
    if [ ! -d "TFG_Readora_Frontend/src/readora" ];
then
        echo "ERROR: No se encuentra el directorio
TFG_Readora_Frontend/src/readora"
        find TFG_Readora_Frontend -type d | sort
        exit 1
    fi
    if [ ! -f ".env.prod" ]; then
        echo "ADVERTENCIA: No se encontró archivo
.env.prod"
        exit 1
    fi
    echo "Iniciando despliegue con Docker Compose
(PROD)..."
    docker-compose -f docker-compose.prod.yml down ||
true
    docker-compose -f docker-compose.prod.yml build
    docker-compose -f docker-compose.prod.yml up -d
    ...
}
}
```



```
        }

    }

post {
    always {
        cleanWs()
    }
    success {
        echo '¡Despliegue completado con éxito!'
    }
    failure {
        echo 'El despliegue ha fallado!'
    }
}
}
```

Este jenkinsfile es de ejemplo y habrá que modificarlo según nuestro repositorio de GitHub y los nombres e IP de nuestra AWS

Una vez configurado todo esto podremos lanzar nuestro primero build en Jenkins

Página de Control > ReadoraPipeline >

Pipeline ReadoraPipeline

Esta ejecución requiere parámetros adicionales:

MAIN_BRANCH

Rama a usar para el repositorio principal

main

Ejecución Cancel

Builds

Filter

Today

- #173 15:47
- #172 15:42
- #171 15:38
- #170 15:35
- #169 15:31
- #168 15:12

<http://localhost:8080/job/ReadoraPipeline/build?delay=0sec>

Este va a tardar, incluso puede dar error, ya que nos faltan unos últimos pasos por hacer en el EC2, para poder completar el despliegue correctamente debemos subir ciertos archivos al EC2 y ponerlos en la ruta correcta, estos son el .env que deberemos ponerlo correctamente en nuestra ruta del proyecto, ademas en la



carpeta padre deberemos configurar tambien una carpeta certs, con los certificados que tenemos, en mi caso tengo uno para el JWT y con un script genero los SSL.

Una vez todo esto este hecho la build funcionara perfectamente.

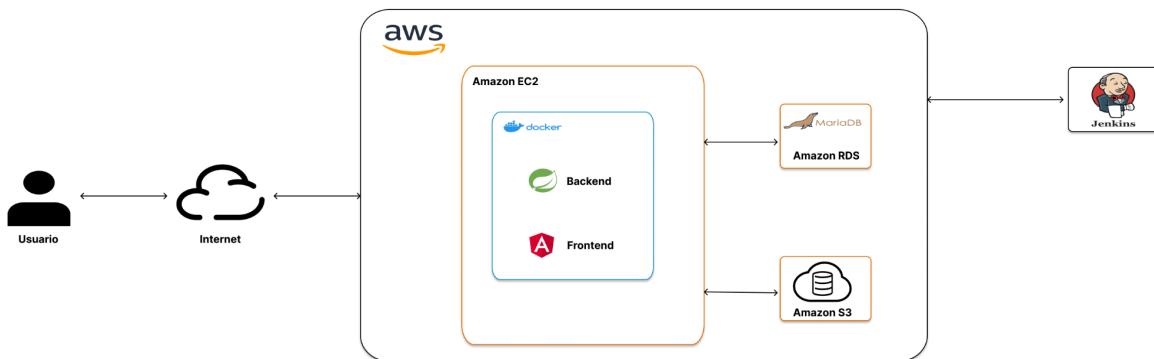
Lo ultimo hecho es comprar y configurar un dominio simple que con un DNS, usa la IP elástica del ec2, como no tiene mucha complicación no entrare en detalle.

Ya con todo esto hecho y arrancado el EC2 podremos buscar nuestro dominio y el proyecto funcionara a la perfección

Pasos para desplegar

Para desplegar el proyecto ya con todo configurado, solo hará falta tener encendido el ec2 en caso de hacer algún cambio, solo tendremos que hacer un push en GitHub y un build con Jenkins y automáticamente estará en el ec2

Muestra de Diagrama de Despliegue





Conclusiones y Postmortem

Tras finalizar el desarrollo y despliegue de la plataforma, puedo afirmar que el proyecto ha cumplido el objetivo principal: crear un espacio cómodo y accesible para la gestión de la lectura.

Sin embargo, no todo ha sido sencillo. La parte de despliegue automatizado con Jenkins y Docker en AWS ha generado complicaciones, sobre todo en la configuración inicial de los entornos y los permisos de los diferentes servicios. Ha sido una etapa que me ha obligado a aprender sobre despliegue, integración continua y administración de recursos cloud, todos ellos conocimientos muy útiles para el futuro.

Si tuviera que repetir el proceso, enfocaría antes ciertos aspectos, como la automatización de tests y la documentación técnica. También creo que podría mejorar la modularización del código y pulir aún más la experiencia de usuario.

En cuanto a resultados, la plataforma funciona correctamente, es estable y cumple todo lo planteado en los requisitos iniciales. Me quedo con la satisfacción de haber creado un servicio que podría ayudar de verdad a los lectores a gestionar su universo de lectura.

Como aprendizaje general, destaco la importancia de una buena planificación y de no subestimar la fase de despliegue y monitorización en entornos reales.

Recomiendo a quien afronte un proyecto así, apoyarse desde el inicio en buenas prácticas de CI/CD, dedicar tiempo a probar los procesos de despliegue y no dejar la documentación para el final.

En resumen, ha sido un proyecto desafiante pero muy enriquecedor, que me ha permitido evolucionar tanto técnica como profesionalmente.



Anexo

URL GitHub:

- Frontend: https://github.com/JavierMoren/TFG_Readora_Frontend
- Backend: https://github.com/JavierMoren/TFG_Readora_Backend

Figma:

<https://www.figma.com/design/Vhp20o9o2nMTOMnQg6kEFh/Readora?node-id=2-3&t=PUSpiTndKpU6tAup-1>

URL Despliegue: <https://readora.site/>