

La página inicial cuenta con dos login, uno para Google y otro por formulario, ambos enviarán a la página de gestión de libros si la autenticación es correcta y generarán un localStorage con el token asignado.



Login

Email

Password

Login

Login con Google

 Inicia sesión como Gabo
gabinavalon@gmail.com 

Logout de Google

Con el botón de logout se elimina el token y se cierra la sesión de Google. Cuando la autenticación es errónea, nos aparece un alert con el error, si es correcta, vamos a la gestión, sin importar con que login sea.

Tanto en la página de libros como en la de subir libros o archivos, comprobamos que el token que se guarda en el localStorage existe para así permitir el acceso a la página, si no, redireccionamos al login.





```
<script>
  if (localStorage.getItem('token') == null) {
    localStorage.setItem('mensaje', "Inicia sesión para entrar");
    window.location.href = "index.html";
  }
</script>
```

Gestión de Libros

[Subir archivos al servidor](#)

Cargar Libros

Nuevo Libro

6229d68e393cf057ea88f5d2	La Sombra del Viento	15	Carlos Ruíz Zafón		<button>Borrar</button>	<button>Actualizar</button>
6229d6a4393cf057ea88f5d4	Ritos Iguales	20	Terry Pratchett		<button>Borrar</button>	<button>Actualizar</button>
6229d726393cf057ea88f5e3	Circe	20.5	Madeline Miller		<button>Borrar</button>	<button>Actualizar</button>
6229d746393cf057ea88f5e5	Madrid será la tumba	15.99	Elizabeth Duval		<button>Borrar</button>	<button>Actualizar</button>

En esta página podemos editar los libros que hay en la base de datos o borrarlos. Todas las operaciones (hasta mostrar) requieren del token. Este se manda por cabeceras en los fetch.

```
const token = localStorage.getItem('token');
```

```
function cargaLibroPorId(idLibro) {  
  $('#myModal').modal('show');  
  
  btnInsertaLibro.style.display = "none";  
  btnActualizaLibro.style.display = "inline";  
  fetch(getAbsolutePath() + 'webresources/generic/libro/' + idLibro, {  
    headers: {  
      'x-token': token  
    }  
  })  
  .then(response => {
```

```
//delete libros  
this.app.delete(  
  "/webresources/generic/libros/:id",  
  validarJWT,  
  async function (req, res) {  
    const id = req.params.id;  
    await Libro.findByIdAndDelete(id);  
    res.status(200).json({  
      ok: true,  
      msg: "delete API",  
    });  
  })  
);
```

Como la página de subir archivos al servidor y subir libros son básicamente la misma, solo que la segunda cuenta con un formulario para añadir libros a la base de datos, vamos a ver la de subir libros.

Nuevo libro

Título

Autor/a

Precio

Imagen

franken.jpg

Insertamos el libro al igual que en el código de ejemplo, y nos guardará un libro en la base de datos además de guardar la foto en la carpeta /img para poder así mostrarla o consultarla más adelante.

6229e86845d0a5374c576f0b

Frankenstein

8.5

Mary W. Shelley

