

**Desarrollo de una herramienta de creación
de contenidos de ficción interactiva**
*Development of a tool for the creation of
interactive fiction*



Andrea Martínez Fernández
Javier Sanz Castro
Andrea del Vado Puell

Dirigido por: Antonio Sarasa Cabezuelo

Curso 2019-2020

Trabajo de fin de grado del Grado en Ingeniería Informática,
Facultad de Informática, Universidad Complutense de Madrid

A nuestra familia. No estaríamos aquí sin ellos.

AGRADECIMIENTOS

Queríamos agradecer en primer lugar a nuestro tutor Antonio Sarasa Cabezuelo por guiarnos durante todo el año en la realización de este proyecto, a pesar de la situación tan complicada que nos ha tocado vivir durante la segunda mitad del curso por el virus.

También les estamos muy agradecidos a nuestra familia, en especial a nuestros padres y hermanos por estar ahí cuando más lo necesitábamos. Y a nuestros amigos y compañeros por su interés en el proyecto y apoyo durante los momentos más complicados.

En último lugar, queremos agradecer a nuestras parejas (los que las tengan) por estar todo el tiempo a nuestro lado.

ÍNDICE

AGRADECIMIENTOS	5
ÍNDICE DE FIGURAS	11
RESUMEN	17
SUMMARY	18
PALABRAS CLAVE	19
CAPÍTULO I: INTRODUCCIÓN	20
MOTIVACIÓN	20
OBJETIVOS	20
ESTRUCTURA DE LA MEMORIA	21
CHAPTER I: INTRODUCTION	22
MOTIVATION	22
OBJECTIVES	22
DOCUMENTATION STRUCTURE	23
CAPÍTULO II: ESTADO DEL ARTE	24
APLICACIONES WEB DE LECTURA Y ESCRITURA DE LIBROS	24
APLICACIONES DE ESCRITURA DE LIBROS INTERACTIVOS	25
CAPÍTULO III: TECNOLOGÍA EMPLEADA	27
VUE	27
VUE.JS	27
VUE CLI	27
BOOTSTRAP VUE	27
BABEL-ESLINT	27
NPM	28
FIREBASE	28
OTROS	28
GOOGLE DRIVE	28
TRELLO	28
GITHUB	29
EDITORES DE CÓDIGO	29
VISUAL STUDIO CODE	29
SUBLIME	29
CAPÍTULO IV: CASOS DE USO	30
DESCRIPCIÓN DE LOS ACTORES	30

MÓDULO GESTIÓN DE USUARIO	30
MÓDULO GESTIÓN DE BIBLIOTECAS DE LIBROS	34
MÓDULO GESTIÓN DE LIBROS	38
MÓDULO GESTIÓN DE SECCIONES	43
MÓDULO GESTIÓN DE GADGETS	46
MÓDULO DE GADGETS	49
CAPÍTULO V: MODELO DE DATOS	60
MODELO ENTIDAD-RELACIÓN	60
BASE DE DATOS NOSQL	62
USERS	63
LIBRARIES	64
BOOKS	65
SECTIONS	67
CAPÍTULO VI: ARQUITECTURA DE LA APLICACIÓN	82
ARQUITECTURA CLIENTE-SERVIDOR	82
EL MODELO DE VUE	82
CAPÍTULO VII: DISEÑO DE LA APLICACIÓN	85
DISEÑO GRÁFICO	85
FUNCIONALIDAD DE LA APLICACIÓN	86
USUARIOS	86
INICIAR SESIÓN	87
CERRAR SESIÓN	88
CREAR USUARIO	88
MOSTRAR USUARIO	89
MODIFICAR USUARIO	89
ELIMINAR USUARIO	90
LIBROS	90
CREAR O MODIFICAR LIBRO	91
EDITAR LIBRO	92
LEER Y PREVISUALIZAR LIBRO	93
MOSTRAR LIBRO	94
ELIMINAR LIBRO	95
BIBLIOTECAS	95
CREAR O MODIFICAR BIBLIOTECA	96
MOSTRAR BIBLIOTECA	96
BORRAR BIBLIOTECAS	97
AÑADIR LIBRO A BIBLIOTECA	97
RETIRAR LIBRO DE BIBLIOTECAS	98
MOVER LIBRO EN LA BIBLIOTECA	100

MOSTRAR LISTA DE BIBLIOTECAS	100
VACIAR HISTORIAL	101
GADGETS	101
INSERTAR GADGET	101
EDITAR GADGET	102
BORRAR GADGET	104
DUPLICAR GADGET	105
PREVISUALIZAR GADGET	106
MOVER GADGET	106
SECCIONES	106
CREAR SECCIÓN	107
MODIFICAR SECCIÓN	108
BORRAR SECCIÓN	109
DUPLICAR SECCIÓN	110
MOVER SECCIÓN	110
CAPÍTULO VIII: EVALUACIÓN DE LA USABILIDAD	112
METODOLOGÍA	112
RESPUESTAS DEL CUESTIONARIO	112
EVALUACIÓN “CREAR USUARIO”	112
EVALUACIÓN “INICIAR SESIÓN”	113
EVALUACIÓN “ACCEDER A DATOS DE USUARIO”	113
EVALUACIÓN “CREAR BIBLIOTECA ”	114
EVALUACIÓN “AGREGAR LIBRO A BIBLIOTECA”	114
EVALUACIÓN “VER LIBRO AÑADIDO A LA BIBLIOTECA”	115
EVALUACIÓN “ABRIR LIBRO”	115
EVALUACIÓN “INTERACCIONES DENTRO DEL LIBRO”	116
EVALUACIÓN “LECTURA DEL LIBRO PIRATILLAS”	116
EVALUACIÓN “CREAR LIBRO”	117
EVALUACIÓN “AÑADIR LIBRO CREADO A BIBLIOTECA”	117
EVALUACIÓN “AÑADIR PORTADA AL LIBRO”	118
EVALUACIÓN “EDITAR LIBRO CREADO”	118
EVALUACIÓN “GADGET TÍTULO”	119
EVALUACIÓN “GADGET TEXTO NORMAL”	119
EVALUACIÓN “CAMBIAR VALORES GADGETS”	120
EVALUACIÓN “AÑADIR SECCIÓN”	120
EVALUACIÓN “GUARDAR CAMBIOS DE SECCIONES”	121
EVALUACIÓN “DESPLAZAMIENTO ENTRE SECCIONES”	121
EVALUACIÓN “AÑADIR GADGETS Y CONFIGURARLOS”	122
EVALUACIÓN “PREVISUALIZAR GADGET”	122
EVALUACIÓN “GADGET SIGUIENTE SECCIÓN”	123

EVALUACIÓN “PREVISUALIZAR LIBRO”	123
EVALUACIÓN “VOLVER A EDICIÓN DE LIBRO”	124
EVALUACIÓN “MOVER GADGETS”	124
EVALUACIÓN “COPIAR GADGETS”	125
EVALUACIÓN “ELIMINAR GADGETS”	125
EVALUACIÓN “COPIAR SECCIÓN”	126
EVALUACIÓN “INTERACTUAR CON GADGETS”	126
EVALUACIÓN “GUARDAR LIBRO”	127
EVALUACIÓN “PUBLICAR LIBRO”	127
EVALUACIÓN “LEER LIBRO PUBLICADO”	128
EVALUACIÓN “ELIMINAR LIBRO”	128
CAPÍTULO IX: CONCLUSIONES Y TRABAJO FUTURO	130
CONCLUSIONES	130
TRABAJO FUTURO	130
CHAPTER IX: CONCLUSIONS AND FUTURE WORK	132
CONCLUSIONES	132
TRABAJO FUTURO	132
CAPÍTULO X: APORTACIONES INDIVIDUALES	134
ANDREA MARTÍNEZ FERNÁNDEZ	134
Contribuciones en la escritura de la memoria	134
Contribuciones en el desarrollo de la aplicación	134
ANDREA DEL VADO PUELL	136
Contribuciones en la escritura de la memoria	136
Contribuciones en el desarrollo de la aplicación	136
JAVIER SANZ CASTRO	138
Contribuciones en la escritura de la memoria	138
Contribuciones en el desarrollo de la aplicación	138
BIBLIOGRAFÍA	141
ANEXOS	144
ANEXO I: GUÍA DE USUARIO	145
BARRA DE NAVEGACIÓN	145
INICIO O VISTA DE LIBROS PUBLICADOS	145
GESTIÓN DE USUARIO	147
CREAR USUARIO	147
INICIAR SESIÓN	148
VISTA DE INFORMACIÓN DE USUARIO	148
MODIFICAR USUARIO	149
CERRAR SESIÓN	149

BIBLIOTECAS	150
VISTA PRINCIPAL BIBLIOTECAS	150
CREAR BIBLIOTECA	151
MODIFICAR BIBLIOTECA	151
AÑADIR LIBRO A BIBLIOTECA	152
VISTA CONTENIDO BIBLIOTECA	152
LIBROS	153
VISTA MIS LIBROS	153
CREAR LIBRO	153
OPCIONES DEL LIBRO Y ELIMINAR	154
MODIFICAR LIBRO	155
LECTURA DE UN LIBRO	156
EDICIÓN DE UN LIBRO	157
GESTIÓN DE SECCIONES	158
AÑADIR GADGETS	159
GESTIONAR GADGETS	160
LIBROS	161
TÍTULO	161
TEXTO NORMAL	162
IMAGEN	162
VÍDEO	162
ANOTACIONES	163
TEXTO EXPANDIBLE	164
HIPERVÍNCULO	165
SPOILER	165
SIGUIENTE SECCIÓN	166
REPETIR SECCIÓN	166
DECISIONES	166
ADIVINANZA	167
SECUENCIA	168
TARJETAS DE MEMORIA	169
NÚMERO ALEATORIO	171
PISTAS	172
CASILLA PERSONALIZADA	173
ANEXO II: GUÍA DE INSTALACIÓN	176
ANEXO III: EVALUACIÓN DE LA USABILIDAD	177

ÍNDICE DE FIGURAS

Figura 1: Diagrama de casos de uso del módulo usuario	30
Figura 2: Diagrama de casos de uso del módulo bibliotecas de libros	34
Figura 3: Diagrama de casos de uso del módulo libro	38
Figura 4: Diagrama de casos de uso del módulo secciones	43
Figura 5: Diagrama de casos de uso del módulo gestión de gadgets	46
Figura 6: Diagrama de casos de uso del módulo gadgets	49
Figura 7: Modelo entidad-relación	60
Figura 8: Esquema de la base de datos	62
Figura 9: Documento de colección de usuario	63
Figura 10: Documento de colección de bibliotecas	64
Figura 11: Documento de colección de libros	65
Figura 12: Documento de colección de secciones	67
Figura 13: Gadget de texto normal	68
Figura 14: Gadget de título	68
Figura 15: Gadget de imagen	69
Figura 16: Gadget de vídeo	69
Figura 17: Gadget de texto expandible	70
Figura 18: Gadget de texto emergente	71
Figura 19: Gadget de hipervínculo	71
Figura 20: Gadget de spoiler	72
Figura 21: Gadget de cambio de sección	72
Figura 22: Gadget de repetición de sección	73
Figura 23: Gadget de toma de decisiones	73
Figura 24: Gadget de adivinanza	74
Figura 25: Gadget de completar la secuencia	75
Figura 26: Gadget de tarjetas de memoria	76
Figura 27: Gadget de número aleatorio	77
Figura 28: Gadget de completar las pistas	79
Figura 29: Gadget de casilla personalizada	80
Figura 30: Esquema de modelo cliente-servidor usado por Vue	82
Figura 31: Esquema del modelo de vista usado por Vue	83
Figura 32: Esquema de ejemplo de la organización por componentes de Vue	84
Figura 33: Ícono de la aplicación	85
Figura 34: Logo de la aplicación	85
Figura 35: Diagrama de funcionalidades	86
Figura 36: Inicio de sesión	87
Figura 37: Código de comprobación de inicio de sesión	87
Figura 38: Almacenamiento de los datos de la sesión actual	87
Figura 39: Confirmación de cierre de sesión	88

Figura 40: Limpieza de los datos de sesión	88
Figura 41: Crear usuario	88
Figura 42: Código de creación de usuario	89
Figura 43: Mostrar usuario	89
Figura 44: Modificar usuario	90
Figura 45: Eliminar usuario	90
Figura 46: Código de eliminar usuario	90
Figura 47: Crear libro	91
Figura 48: Modificar libro	91
Figura 49: Código de crear libro	92
Figura 50: Lectura y previsualización del libro	92
Figura 51: Carga del contenido de la sección	93
Figura 52: Código de mostrar el contenido de la sección	93
Figura 53: Lectura y previsualización del libro	94
Figura 54: Mostrar libro - Mis libros	94
Figura 55: Mostrar libro - Inicio	95
Figura 56: Eliminar libro	95
Figura 57: Confirmar eliminar libro	95
Figura 58: Crear biblioteca	96
Figura 59: Modificar biblioteca	96
Figura 60: Mostrar biblioteca	97
Figura 61: Mostrar biblioteca - Historial	97
Figura 62: Eliminar biblioteca	97
Figura 63: Añadir libro a biblioteca	98
Figura 64: Código de añadir libro a biblioteca	98
Figura 65: Código de quitar libro de biblioteca	99
Figura 66: Código de mover un libro en la biblioteca	99
Figura 67: Lista de bibliotecas	100
Figura 68: Vaciar historial	100
Figura 69: Añadir gadget	101
Figura 70: Código de añadir gadget	101
Figura 71: Editar gadget	102
Figura 72: Código que añade el componente del gadget en edición	102
Figura 73: Formulario del gadget de añadir imagen	103
Figura 74: Guardado del gadget de añadir imagen	103
Figura 75: Guardado del gadget en 'editBook'	103
Figura 76: Eliminar gadget	104
Figura 77: Código de eliminar gadget	104
Figura 78: Código de 'checkRead'	104
Figura 79: Duplicar gadget	105
Figura 80: Código de duplicar gadget	105
Figura 81: Previsualizar gadget	106

Figura 82: Previsualización de gadget	106
Figura 83: Mover gadget	106
Figura 84: Crear sección	107
Figura 85: Código de crear sección	108
Figura 86: Código de modificar sección	108
Figura 87: Código de guardar sección	109
Figura 88: Código de eliminar sección	109
Figura 89: Código de clonar sección	110
Figura 90: Código de mover sección	111
Figura 91: Resultados de crear usuario	113
Figura 92: Resultados de iniciar sesión	113
Figura 93: Resultados de acceso a los datos del usuario	114
Figura 94: Resultados de crear biblioteca	114
Figura 95: Resultados de agregar libro a biblioteca	115
Figura 96: Resultados de buscar el libro añadido en la biblioteca	115
Figura 97: Resultados de abrir el libro ‘Piratillas’	116
Figura 98: Valoración de la sencillez de las interacciones en el libro	116
Figura 99: Valoración de la lectura del libro	117
Figura 100: Resultados de la creación del libro	117
Figura 101: Resultados de añadir el libro creado a la biblioteca	118
Figura 102: Resultados de añadir portada al libro	118
Figura 103: Resultados de iniciar la edición de un libro	119
Figura 104: Resultados de escribir título	119
Figura 105: Resultados de escribir texto normal	120
Figura 106: Resultados de modificar gadgets	120
Figura 107: Resultados de añadir nueva sección	121
Figura 108: Resultados de guardar los cambios en las secciones	121
Figura 109: Resultados de desplazarse entre secciones	122
Figura 110: Resultados de añadir un gadget y configurarlo	122
Figura 111: Resultados de previsualizar gadget	123
Figura 112: Resultados de añadir un gadget de siguiente sección y configurarlo correctamente	123
Figura 113: Resultados de previsualizar el libro completo	124
Figura 114: Resultados de volver a la edición desde la previsualización	124
Figura 115: Resultados de mover los gadgets	125
Figura 116: Resultados de copiar los gadgets	125
Figura 117: Resultados de eliminar los gadgets	126
Figura 118: Resultados de copiar una sección	126
Figura 119: Valoración de la interacción con los distintos gadgets	127
Figura 120: Valoración de guardar el libro	127
Figura 121: Valoración de publicar libro	128
Figura 122: Valoración de leer libro publicado	128
Figura 123: Valoración de eliminar libro	129

Figura 124: Barra de navegación sin iniciar sesión	145
Figura 125: Barra de navegación al iniciar sesión	145
Figura 126: Vista de libros publicados	146
Figura 127: Datos de cada libro	146
Figura 128: Vista de registro	147
Figura 129: Vista de iniciar sesión	148
Figura 130: Vista de información de usuario	148
Figura 131: Vista de modificar usuario	149
Figura 132: Vista de cierre de sesión	150
Figura 133: Vista de bibliotecas	150
Figura 134: Vista de crear biblioteca	151
Figura 135: Vista de modificar biblioteca	151
Figura 136: Vista de añadir libro a biblioteca	152
Figura 137: Vista del contenido de una biblioteca	152
Figura 138: Vista de mis libros	153
Figura 139: Vista de crear libro	154
Figura 140: Vista de eliminar libro	155
Figura 141: Vista de confirmación de eliminación	155
Figura 142: Vista de modificar libro	156
Figura 143: Vista de lectura de libro	157
Figura 144: Vista de editar un libro	157
Figura 145: Vista de gestión de secciones	158
Figura 146: Vista de añadir gadgets	159
Figura 147: Vista de gestión de gadgets	161
Figura 148: Vista de títulos	162
Figura 149: Vista de texto normal	162
Figura 150: Vista de imagen	162
Figura 151: Vista de vídeo	163
Figura 152: Vista de anotaciones	163
Figura 153: Vista de anotaciones con imagen	164
Figura 154: Vista de texto expandible	165
Figura 155: Vista de hipervínculos	165
Figura 156: Vista de spoiler	166
Figura 157: Vista de siguiente sección	166
Figura 158: Vista de repetir sección	166
Figura 159: Vista de toma de decisiones	167
Figura 160: Vista de adivinanza	167
Figura 161: Vista de completar la secuencia	168
Figura 162: Vista de tarjetas de memoria	169
Figura 163: Vista de tarjetas de memoria por colores	170
Figura 164: Vista de tarjetas de memoria por palabras	171
Figura 165: Vista de número aleatorio	172

Figura 166: Vista de completar pistas	173
Figura 167: Vista de casilla personalizada en modo escritura	174
Figura 168: Vista de casilla personalizada en modo escritura y en lectura del libro	174
Figura 169: Vista de casilla personalizada en modo lectura	175
Figura 170: Vista de casilla personalizada en modo de modificar escritura	175
Figura 171: Introducción de la evaluación.	177
Figura 172: Pregunta “Crear un usuario”	177
Figura 173: Pregunta “Iniciar sesión”	177
Figura 174: Pregunta “Acceder a datos de usuario”	178
Figura 175: Pregunta “Crear biblioteca”	178
Figura 176: Pregunta “Agregar libro a una biblioteca”	178
Figura 177: Pregunta “Ver libro añadido en biblioteca”	178
Figura 178: Pregunta “Abrir libro Piratillas”	179
Figura 179: Pregunta “Interacciones dentro del libro”	179
Figura 180: Pregunta “Lectura de Piratillas”	179
Figura 181: Pregunta “Crear un libro”	179
Figura 182: Pregunta “Añadir libro creado a biblioteca”	180
Figura 183: Pregunta “Añadir portada al libro”	180
Figura 184: Pregunta “Editar libro”	180
Figura 185: Pregunta “Gadget título”	180
Figura 186: Pregunta “Gadget texto normal”	181
Figura 187: Pregunta “Cambiar valores de los gadgets”	181
Figura 188: Pregunta “Añadir una nueva sección”	181
Figura 189: Pregunta “Guardar cambios secciones”	181
Figura 190: Pregunta “Desplazarse entre secciones”	182
Figura 191: Pregunta “Añadir gadget”	182
Figura 192: Pregunta “Previsualizar gadget”	182
Figura 193: Pregunta “Gadget siguiente sección”	182
Figura 194: Pregunta “Previsualizar libro”	183
Figura 195: Pregunta “Volver a edición”	183
Figura 196: Pregunta “Mover gadgets”	183
Figura 197: Pregunta “Copiar gadgets”	183
Figura 198: Pregunta “Eliminar gadgets”	184
Figura 199: Pregunta “Copiar sección”	184
Figura 200: Pregunta “Interacción con gadgets”	184
Figura 201: Pregunta “Guardar libro”	184
Figura 202: Pregunta “Publicar libro”	185
Figura 203: Pregunta “Leer libro publicado”	185
Figura 204: Pregunta “Eliminar libro”	185

RESUMEN

El objetivo de este trabajo de fin de grado es realizar una aplicación web para el desarrollo de libros con contenidos de ficción interactiva. En la aplicación se implementa un editor de texto básico, centrado en facilitar la inclusión de elementos interactivos en el libro. También se incluye un lector que permite la lectura interactiva.

Para realizarlo, se han implementado funcionalidades como las bibliotecas y los gadgets. Las bibliotecas son listas de libros en la que los usuarios puedan ordenar y acceder fácilmente tanto a sus lecturas como a sus escritos. Los gadgets son plantillas para los distintos elementos interactivos con el objetivo de facilitar la escritura de los libros.

Para hacer uso de todas estas herramientas, los usuarios tienen que estar registrados e iniciar sesión en la plataforma.

SUMMARY

The objective of this end of degree project is to develop a web application used for writing interactive fiction. In the application, there will be a text editor focused on allowing inserting interactive elements into the book that is being edited. There will also be included a way to read these books.

To accomplish the aforementioned, things like libraries and gadgets will be implemented. The libraries are lists of books made to help users organize and gain easy access the books they are reading and writing. Gadgets are templates of the different interactive elements.

Users have to be registered and logged in to use all these tools described above.

PALABRAS CLAVE

- Autor
- Lector
- Escritor
- Libro
- Interactivo
- Gadget
- Biblioteca
- Usuario

1. CAPÍTULO I: INTRODUCCIÓN

En este capítulo se va a explicar la motivación de la aplicación desarrollada, los objetivos planteados y la estructura de la memoria.

1.1. MOTIVACIÓN

La ficción interactiva es un tipo de narrativa en la que los lectores participan de una manera más activa en la lectura, ya sea accediendo a hipervínculos, tomando decisiones que cambian el curso de la historia redirigiendo al lector a otra página o pulsando en algún personaje.

Hasta ahora la forma más famosa de ficción interactiva son las novelas, y las más populares se encuentran en dispositivos móviles, permitiendo hacer lo descrito en el párrafo anterior, es decir, pulsar en distintos elementos interactivos como pueden ser hipervínculos o personajes. Otras maneras que provocan que el lector intervenga en la toma de decisiones es directamente preguntándole qué acción desea realizar. También existen las novelas interactivas físicas, que son libros que indican a qué página se debe ir con cada decisión, moviéndose así el lector entre los distintos capítulos según sus elecciones.

No obstante, desde los libros en físico de la segunda mitad del siglo XX hasta su llegada a los dispositivos móviles, este tipo de ficción se ha mantenido prácticamente sin variaciones con el paso del tiempo, debido, entre otras razones, a la falta de herramientas para que los escritores creen contenido de una manera rápida y eficaz, pues hasta ahora realizarlos requería conocimientos de diseño y programación o una gran habilidad para organizar el texto y que todo tuviese sentido. Esto se observa en que la gran mayoría de aplicaciones online que ofrecen herramientas para la escritura de ficción interactiva cuentan con poco más aparte de poder configurar el texto, poner títulos y añadir elementos multimedia, pues dichas aplicaciones limitan mucho lo que los escritores pueden hacer con dichas herramientas o tienen una complejidad de uso muy elevada. Esta falta de herramientas para dar efectos al texto, sumado a la complejidad actual de escribir libros interactivos de calidad, es lo que limita las novedades y variaciones de este tipo de obras.

En este trabajo se ha desarrollado una aplicación que pretende facilitar y agilizar la escritura de este tipo de ficción mediante una interfaz rápida e intuitiva y herramientas e interacciones novedosas, además de acercar su lectura a un gran y creciente público de lectores gracias al componente social de la aplicación, en la que sus usuarios pueden compartir sus obras para que el resto las puedan leer.

1.2. OBJETIVOS

El objetivo principal de este proyecto era desarrollar un sistema que permitiera la escritura y lectura de novelas con elementos interactivos. Este objetivo se desarrolla en los siguientes objetivos más específicos:

- Desarrollar una aplicación web que permita a los usuarios la escritura de novelas con elementos interactivos proporcionados por la propia aplicación, así como la lectura interactiva de dichas novelas.
- Permitir la organización de novelas en bibliotecas personalizadas por los usuarios.
- Estructurar la base de datos para almacenar los datos de los usuario, bibliotecas y libros.
- Permitir que los usuarios puedan leer las novelas de otros usuarios.
- Facilitar la configuración de los elementos interactivos para que los usuarios no necesiten apenas conocimientos para usarlos tanto en la escritura como en la lectura.
- Ofrecer una interfaz sencilla que encapsule toda la funcionalidad de la aplicación y que esté adaptada a los estándares actuales.

1.3. ESTRUCTURA DE LA MEMORIA

La memoria de este proyecto se estructura de la siguiente manera:

- En el capítulo 1 se presenta la motivación y los objetivos del proyecto, así como la estructura interna de la memoria.
- En el capítulo 2 se presentan algunas herramientas similares a la desarrollada.
- En el capítulo 3 se describe la tecnología empleada para el desarrollo del proyecto.
- En el capítulo 4 se presentan los casos de uso que describen las funcionalidades de la aplicación.
- En el capítulo 5 se describe el modelo de datos utilizado para persistir la información así como su implementación mediante una base de datos no relacional.
- En el capítulo 6 se explica la arquitectura de la aplicación desarrollada.
- En el capítulo 7 se describe el diseño de la aplicación.
- En el capítulo 8 se presentan los resultados de una evaluación de la usabilidad que se ha realizado sobre distintos usuarios.
- En el capítulo 9 se enuncian un conjunto de conclusiones y se plantean las principales líneas de trabajo futuro.
- En el capítulo 10 se describe las tareas desarrolladas por cada integrante del equipo.

1. CHAPTER I: INTRODUCTION

In this first chapter we are going to explain what the motivation of the developed application is, the goal objectives that we had and what the structure of the documentation is going to be.

1.1. MOTIVATION

Interactive fiction is a type of narrative in which the readers are able to enjoy the book in a more active way, it can be in many ways, such as clicking in an external link, making decisions that change the flow of the story taking the reader to another page or clicking in some character.

The most famous way to represent interactive fiction are novels, and the most popular are the ones accessible with mobile phones. These interactive novels allow the interaction in the way described above, clicking in different interactive elements such as links or in characters, or allow making decisions asking directly what course of action does the reader want to take. There is also another way of interactive novels which is in physical books. The way these books allow the interaction is with an indicator of the page the reader has to go to depending the decision he made.

However, since physical books on the second half of the 20th century until their arrival at mobile phones, this type of fiction has not suffered many transformations through time. This is due to, apart from other reasons, to the lack of suitable tools at the disposition of writers to write in an efficient and fast way, as until now to do it, writers required design and programming knowledge or a great capacity to organize the written text so that everything made sense. This can be observed in that most of the online application which offer tools for the writing of interactive fiction lack options apart from allowing to customize text, write titles and add multimedia elements, this is due to the limitation these applications enforce for users and that those which do not enforce it have tools with high complexity of use. This lack of tools and the complexity to write quality interactive novels is what limits innovation and variation in these kind of novels.

In these project we have developed a web application that helps make the writing of this type of fiction easier and faster thanks to a fast and intuitive interface and new interactive tools. It also allows a bigger group of people to have access to this type of novels thanks to the social component of the application, which allows writers to publish and share their work with just a click.

1.2. OBJECTIVES

The main objective of this project was to develop a system that allows the writing and reading of novels with different interactive elements. The specific objectives defined were as follows:

- Develop a web application that allows users to write novels with the interactive elements given by it, as well as the reading of these novels.
- Allow the organization of novels in libraries customized by the users.
- Make a structured database to store the data related to users, libraries and books, being specially interesting the writing and reading of them with their interactive elements.
- Allow users to read novels written by other people.
- Make the interactive elements easy to customize so that the users do not need extensive knowledge to use them, both in the reading and writing of a book.
- Offer a simple interface that encapsulates all the functionality of the application and that is up to date with modern standards.

1.3. DOCUMENTATION STRUCTURE

The documentation of this project is structured in the following way:

- The first chapter presents the motivation and objectives of the project, as well as its internal structure.
- The second chapter presents similar applications as the one developed.
- The third chapter describes the technology used to develop the project.
- The fourth chapter presents the use cases with the functionality of the application.
- The fifth chapter describes the data model used to persist all the information in the application as well as its implementation through a non relational database.
- The sixth chapter explains the architecture of the developed application.
- The seventh chapter describes the design of the application.
- The eighth chapter presents a summary of the result of the evaluation answered by other users.
- The ninth chapter includes a mention of the possible improvements in the future and the conclusions obtained during the development of the project.
- The tenth chapter contains all the work done by the different members of the project.

2. CAPÍTULO II: ESTADO DEL ARTE

En este capítulo se van a mostrar algunas herramientas similares a la desarrollada en este proyecto. En primer lugar, se van a presentar herramientas que permiten leer y escribir libros online, pero que carecen del elemento interactivo. El punto fuerte de estas aplicaciones es el aspecto social que permite más interacción entre los usuarios con comentarios en los libros y la opción de seguirse unos a otros, entre otras cosas. En el segundo grupo de herramientas se encuentran las que permiten al usuario crear su propia ficción interactiva.

2.1. APLICACIONES WEB DE LECTURA Y ESCRITURA DE LIBROS

Estas webs proporcionan al usuario las herramientas necesarias para leer, escribir y compartir historias. Cuentan con los módulos de usuarios, libros y bibliotecas. Sin embargo, no disponen del módulo de gadgets que proporciona la característica interactiva a los libros. Se tienen los siguientes proyectos:

- Wattpad [1]: Esta aplicación web se diferencia por su enfoque social, que permite a los usuarios seguirse entre ellos y comunicarse a través de mensajes. Entre sus funcionalidades destaca:
 - Gestión de bibliotecas.
 - Creación y publicación de libros por partes (equivalente a las secciones).
 - La comunidad, que presenta foros y premios a los libros.
 - El componente social, seguir y enviar mensajes.
 - La valoración de libros por parte de los usuarios.
 - Libros de pago.
 - Autoguardado.
 - Comentarios de usuario por párrafo de libro.
- Megustaescribir [2]: Se distingue por permitir a los usuarios la descarga y la subida de los libros. Entre sus funcionalidades destaca:
 - La opción de descargar libro.
 - La opción de subir libro.
 - Procesador de texto con más opciones.
 - Creación y publicación de libros por capítulos (equivalente a las secciones).
 - El componente social, seguir a usuarios.
 - La valoración de libros por parte de los usuarios.
 - Foros de debate.
 - Opciones de pago (revisión del texto, mejores portadas...).
- SoopBook [40]: Destacaba por permitir a sus usuarios la creación de libros colaborativos que se podían escribir por varios usuarios. Lamentablemente, la página web ya no se encuentra disponible. Los aspectos que más la caracterizan son:
 - Opción de venta online de los libros.
 - Herramientas de gestión de equipos.

- El propietario del libro tiene el control sobre que usuarios pueden ver y editar su libro.
- Escritura y publicación de los libros por capítulos.
- Gestión de edición editorial.
- Generación de libros en formato EPUB.
- Estadísticas sobre el número de lecturas y ventas de los libros del usuario.
- ZooBurst [41]: Esta aplicación móvil permite crear libros con imágenes 3D que se pueden disfrutar visualizándolos con la cámara del móvil. Actualmente solo está disponible en la AppStore y solo permite la visualización de los libros no la creación. Entre sus funcionalidades destaca:
 - Exportación de los libros en formato HTML.
 - Añadir elementos como texto, imágenes y grabaciones de sonido a los libros 3D.
 - Visualización 3D de los libros con la cámara del móvil en el espacio real.
 - Escáner de códigos QR para acceder una historia en concreto.

2.2. APLICACIONES DE ESCRITURA DE LIBROS INTERACTIVOS

Este tipo de herramientas permiten la creación de ficción interactiva con elementos como links a distintas partes de la narrativa, declaración de variables o uso de condicionales. Un ejemplo de estas herramientas es:

- Twine [3]: Es una herramienta disponible en aplicación de escritorio y aplicación web que permite crear historias con elementos interactivos a través del uso de determinados caracteres en el texto o código script. Sus características principales son:
 - Los libros se generan en código HTML.
 - Dispone de una gran variedad de elementos interactivos (links dentro del libro, condicionales lógicos, variables, imágenes, ...).
 - Permite el uso de código script para añadir funcionalidades al libro.
 - Soporta código CSS para modificar el estilo de la obra.
- Quest [30]: Al igual que Twine [3] esta herramienta también está disponible en versión de escritorio y en versión online. Con ella se pueden crear dos tipos de ficciones interactivas que ellos llaman aventuras de texto y librojuego [31]. El primero se asemeja más a la herramienta implementada con una estructura por páginas, tomas de decisiones que mueven al lector de una página a otra y casillas personalizadas entre otras cosas. El segundo se estructura por lugares que contienen objetos con los que el lector puede interactuar. Las transiciones de un lugar a otro se definen en la edición usando los puntos cardinales de manera que se va trazando un mapa de lugares por el que el lector se puede mover libremente. Las funcionalidades más destacadas de Quest son:

- Desde la versión online existe la opción de descargar e importar las ficciones interactivas.
- Acepta varios formatos de importación, el propio (.quest), html, zip, z5, z8 y más.
- Dentro de la aplicación web se pueden leer los libros publicados por otros usuarios.
- Los libros se agrupan por temáticas.
- Los usuarios pueden valorar, comentar y hacer reseñas de los libros.
- La edición de las ficciones interactivas es más guiada ya que cada página o lugar es un formulario a completar por el usuario con distintas opciones para personalizar los elementos. A diferencia de los anteriores en los que la experiencia de edición es más similar a la escritura de un libro con una hoja en blanco sobre la que escribir y añadir los elementos interactivos
- Dentro del modo de lectura hay una sección especial indicando los objetos que se encuentran en esa escena, otra para el contenido del inventario y una última con los posibles desplazamientos desde esa ubicación.
- Squiffy [42]: Está disponible desde la misma página web que Quest por lo que comparte el sistema de valoración de libros, la agrupación por temáticas y las opciones de importación. También se encuentra tanto en versión web como en versión de escritorio. Destaca por las siguientes funcionalidades:
 - El texto se separa por secciones que pueden ser enlazadas.
 - Dentro de cada sección se pueden añadir pasajes que por defecto están ocultos, pero al pulsar las palabras destacadas del texto se muestran.
 - La edición del libro es muy sencilla gracias al reducido número de elementos que lo componen. Consta de una sola hoja en la que las referencias de las secciones se identifican por encontrarse entre dobles corchetes y los pasajes entre corchetes simples. Para definir las secciones y los pasajes basta con añadir una línea con el componente seguido de dos puntos y en las líneas sucesivas el contenido del mismo.
 - Desde la edición se puede probar el libro en su forma de lectura.
 - Los libros se pueden descargar en distintos formatos: script Squiffy, HTML con JavaScript y solo JavaScript.

3. CAPÍTULO III: TECNOLOGÍA EMPLEADA

En este capítulo, se van a describir las tecnologías y las herramientas empleadas en el desarrollo de este proyecto.

3.1. VUE

3.1.1. VUE.JS

Vue.js [4] es un framework de código abierto para JavaScript que permite crear aplicaciones de una sola página (single-page applications). Se basa en el patrón “modelo-vista-modelo de vista” [5] para desacoplar la lógica de la interfaz. Además, posee un renderizado declarativo, es reactivo, permite el uso de “templates” y cuenta con componentes y directivas para facilitar la programación. También cuenta con bibliotecas propias y oficiales como Router [32], para manejar las rutas de forma sencilla o Vuex [6], que en resumen sirve para manejar estados de la aplicación, pudiendo transmitirlos entre componentes; y otras no oficiales como Vuefire [33], que permite realizar conexiones entre Firebase y Vue.

3.1.2. VUE CLI

Vue Cli [7] es la herramienta estándar para el desarrollo en Vue.js. Permite realizar distintas acciones tales como configurar el proyecto, compilar, cargar librerías, prototipar de manera instantánea o manejar librerías, complementos y dependencias, entre otras, a través de una interfaz gráfica. Además, desde Vue Cli se pueden gestionar los proyectos de Vue del dispositivo donde esté instalado.

3.1.3. BOOTSTRAP VUE

Bootstrap [8] es una biblioteca de código abierto especializada en el diseño de aplicaciones web. Se encuentra escrito en HTML, CSS, JavaScript, Less y SaaS. Contiene multitud de plantillas de diseño con sus componentes propios para usarlas y personalizarlas en pocas líneas de código. BootstrapVue [9] es la adaptación de Bootstrap para Vue.js. Posee las mismas funcionalidades pero adaptadas al framework, por lo que es requerido disponer de PortalVue [34], que no es más que un componente usado para renderizar otros componentes en cualquier parte del DOM.

3.1.4. BABEL-ESLINT

Lint [35] es una herramienta de programación usada para analizar el código buscando errores sintácticos que normalmente se escapan al compilador; ESLint [12] es la versión de Lint para JavaScript; y Babel es el transcompilador (una herramienta que transforma un código en otro entre mismos o distintos lenguajes) usado principalmente para poder usar las últimas funciones de JavaScript. Babel-ESLint [11] aúna las tres herramientas, y es usado para permitir ejecutar código fuente de ESLint que ha sido transformado por Babel.

3.2. NPM

NPM [14] es un gestor de paquetes de código abierto escrito en JavaScript y para JavaScript. Es el gestor por defecto del entorno de ejecución node.js. Consiste en una línea de comandos que interactúa con un registro remoto y online para consumir y distribuir módulos de JavaScript presentes en ese registro y administrar las dependencias locales de los proyectos.

3.3. FIREBASE

Firebase [13] es una herramienta para el desarrollo de aplicaciones web. Se encuentra ubicada en la nube y se usa, entre otros, para almacenar datos de una manera rápida y eficaz usando árboles JSON (Realtime Database) o agrupándolos por colecciones y documentos (Firestore), según como se desee. Además, cuenta con medidas de seguridad en caso de perder la conexión, con datos sobre su uso, con un almacenamiento propio para los archivos que la base de datos no soporta, y permite que los datos se puedan visualizar en directo, facilitando su uso en el mismo instante en que quedan guardados en sus servidores.

3.4. OTROS

3.4.1. GOOGLE DRIVE

Google Drive [19] es uno de los servicios de alojamiento de archivos más extendidos y más usados el mundo. Permite que sus usuarios almacenen todo tipo de archivos en sus servidores, y cuenta con 15GB gratuitos para cada uno con posibilidad de ampliarlos mediante planes de pago. Cuenta tanto con una versión web como con versiones adaptadas a otros dispositivos como la de escritorio o la de teléfonos móviles. Además, es posible conectar aplicaciones a las carpetas y archivos para que puedan abrirse en Drive. Entre esas aplicaciones de Google Drive se encuentra Google Docs [20], que es un procesador de textos en línea. Es gratuito, posee una amplia variedad de herramientas con las que dar formato a la escritura, permite la edición de los ficheros por parte de varias personas en directo, cuenta con un control de versiones para volver a estadios anteriores de los documentos en caso de ser necesario y permite realizar comentarios y sugerencias empleados para que las personas que los escriben puedan comunicarse y corregirse.

3.4.2. TRELLO

Trello [21] es un software basado en web usado como organizador y como sistema de control simple de las tareas asociadas a los proyectos. Posee una interfaz gráfica intuitiva, creación de grupos, de listas de tareas y de tareas en sí mismas, así como la posibilidad de asignar dichas tareas a los integrantes de los proyectos, añadirles etiquetas, escribir comentarios en ellas o crear listas de comprobaciones (checklists), entre otros.

3.4.3. GITHUB

GitHub [15] es uno de los sistemas de control de versiones más famosos que hay disponibles, permitiendo que todos los miembros de los distintos proyectos trabajen en paralelo de una forma eficaz. Para funcionar, GitHub hace uso del sistema de control de versiones de Git [16]. Cuenta con herramientas tales como ramificaciones del proyecto, interfaz gráfica, posibilidad de dar marcha atrás a los cambios realizados o estadísticas y datos relevantes para los colaboradores de los repositorios. Además, cuenta con el servicio de GitHub Pages [36], que es un software que permite albergar páginas web directamente desde sus repositorios. Para ello, es necesario que los repositorios contengan ficheros HTML, CSS y JavaScript y que, en algunos casos, se pase por una previa compilación. Además, da la opción de albergar las páginas tanto en los propios dominios de GitHub (github.io) como en dominios personalizados por el usuario.

3.5. EDITORES DE CÓDIGO

3.5.1. VISUAL STUDIO CODE

Visual Studio Code [17] es un editor de código fuente usado para desarrollar aplicaciones en multitud de lenguajes. Su código es abierto y se encuentra escrito en TypeScript, JavaScript y CSS. Posee herramientas para hacer debug, resaltado de texto, completado automático del código y soporte con Git [16], entre otros. Además, admite extensiones para añadir distintas funcionalidades.

3.5.2. SUBLIME

Sublime [18] es un editor de código fuente usado para desarrollar aplicaciones en multitud de lenguajes. Se encuentra desarrollado en Python y C++. Con Sublime se pueden hacer selecciones múltiples, desplazarse entre ficheros rápidamente, es altamente customizable y además soporta plugins externos para aumentar su funcionalidad.

4. CAPÍTULO IV: CASOS DE USO

En este capítulo se presentan los casos de uso de la aplicación desarrollada. La clasificación de los mismos se va a realizar en módulos dependiendo de la funcionalidad que representan dentro de la aplicación.

4.1. DESCRIPCIÓN DE LOS ACTORES

En la aplicación se han considerado 3 actores de acuerdo al tipo de actividades que realizan:

- Lector: Representa a los usuarios que utilizan la aplicación para leer, organizar bibliotecas o cualquier otra operación relacionada con el uso de libros digitales.
- Escritor: Representa a los usuarios que realizan cualquier acción relacionada con la creación o edición de libros digitales. Un escritor también puede ser llamado autor.
- Administrador: Representa al usuario encargado de confirmar la eliminación de una obra publicada o una obra no apta (ya sea por contenido ofensivo o similar) así como la realización de operaciones de mantenimiento de la aplicación.

4.2. MÓDULO GESTIÓN DE USUARIO

Este módulo representa toda la funcionalidad relativa a la gestión de la información de un usuario de la aplicación: creación de un usuario, borrado de un usuario, y modificación y visualización de la información del perfil de un usuario. Estos casos de uso son llevados a cabo por el lector. En la figura 1 se muestra el diagrama de caso de uso asociado a este módulo.

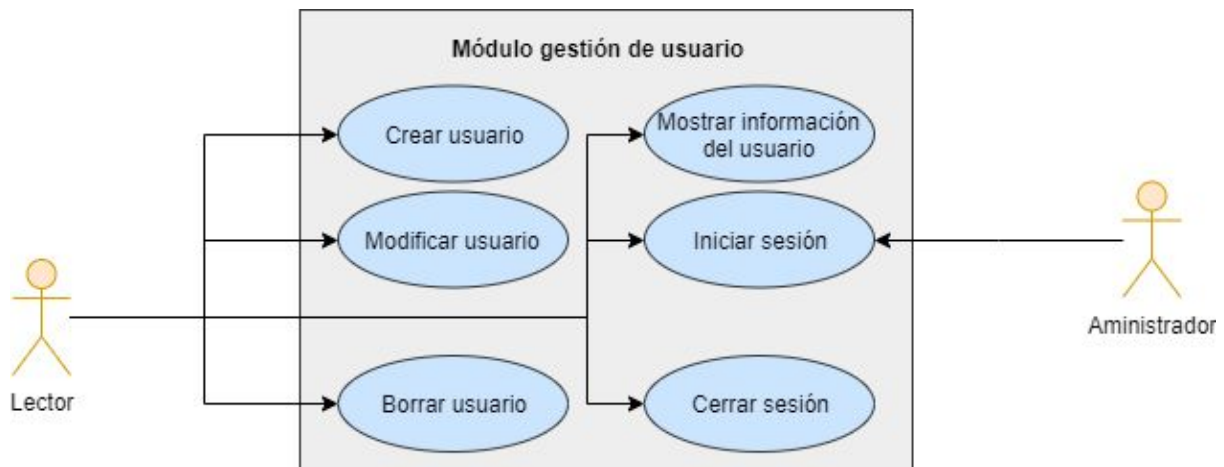


Figura 1. Diagrama de casos de uso del módulo usuario.

A continuación, se describe con detalle cada uno de los casos de uso asociados a este módulo:

Identificador	US01
---------------	------

Título del requisito	Crear usuario
Actores	Lector
Función	Añadir un nuevo usuario en el sistema
Prioridad	Alta
Entrada	Datos de usuario: Nombre (opcional), nombre de usuario único, correo electrónico, contraseña e imagen.
Salida	Mensaje confirmando el registro. Se inicia sesión automáticamente y te devuelve a la página de inicio.
Precondición	Que ni el email ni el nombre de usuario estén ya registrados.
Postcondición	Se añade el usuario a la BBDD de usuarios.
Descripción	Se añade un nuevo usuario al sistema. Para ello se presenta un formulario con los campos a rellenar. Una vez completados el usuario pulsa el botón de crear usuario y entonces se realiza la comprobación de que las contraseñas sean iguales, mínimo 6 caracteres, el nombre de usuario no esté ya registrado y el correo electrónico sea válido. Se le da al usuario la opción de añadir una fotografía, en caso de no añadir ninguna se le pone una imagen por defecto. Si todas las condiciones se cumplen se crea el usuario, en caso contrario, se vuelve al formulario para solucionar estos problemas.

Identificador	US02
Título del requisito	Modificar usuario
Actores	Lector
Función	Modificar la información del usuario.
Prioridad	Media
Entrada	Los nuevos datos: <ul style="list-style-type: none"> ● Editar, cambiar o eliminar foto de perfil ● Cambiar nombre ● Cambiar correo ● Cambiar contraseña
Salida	Cambios efectuados
Precondición	1. Usuario existente en la BBDD y con sesión iniciada.

	2. El usuario tiene que estar en el apartado de su perfil.
Postcondición	Los cambios se aplican al perfil del usuario.
Descripción	<p>El usuario dentro de su perfil selecciona la opción de modificar, introduce en la casilla correspondiente la nueva información, confirma los cambios y si ha rellenado todo correctamente se aplican los cambios y se guardan en la BBDD, si los datos introducidos no son correctos, entonces se le indica con un mensaje y vuelve a la pantalla de modificación.</p> <p>Los apartados que se pueden cambiar son: nombre, correo electrónico, contraseña y foto de perfil. También tiene la opción de dejar vacío cualquiera de los campos opcionales.</p>

Identificador	US03
Título del requisito	Borrar usuario
Actores	Lector
Función	Eliminar el usuario de la base de datos.
Prioridad	Media
Entrada	Mensaje pidiendo confirmación para eliminar el usuario.
Salida	Redireccionar la página al inicio.
Precondición	El usuario ha iniciado sesión y está en la base de datos
Postcondición	Se actualiza la base de datos para eliminar al usuario.
Descripción	En la pantalla de usuario se pulsa la opción borrar, el sistema te pide que confirmes que deseas eliminarlo. Al eliminar el usuario se elimina todo, incluidos los libros anteriormente publicados.

Identificador	US04
Título del requisito	Mostrar información de usuario
Actores	Lector
Función	Mostrar la información relevante del usuario
Prioridad	Alta
Entrada	Acceder al perfil

Salida	Datos del usuario
Precondición	El usuario existe en la base de datos
Postcondición	El sistema muestra por pantalla todos los datos del usuario
Descripción	Al acceder al perfil se muestran todos los datos: fotografía, nombre, nombre de usuario y correo electrónico.

Identificador	US05
Título del requisito	Iniciar sesión
Actores	Lector, escritor y administrador
Función	Iniciar sesión
Prioridad	Alta
Entrada	Nombre de usuario y contraseña
Salida	Redirige a la pestaña seleccionada con la sesión iniciada
Precondición	<ol style="list-style-type: none"> 1. Nombre de usuario existente en la BBDD 2. Contraseña correcta
Postcondición	Usuario inicia sesión
Descripción	El usuario accede a la pantalla de inicio de sesión donde tiene que rellenar los campos de nombre de usuario y contraseña. Si los introduce correctamente se le redirige a la pestaña que intentara acceder o al home con la sesión ya iniciada. Si los datos no están bien se le muestra el mensaje de error y se le da la posibilidad de modificar los campos.

Identificador	US06
Título del requisito	Cerrar sesión
Actores	Lector
Función	Cierra la sesión iniciada
Prioridad	Alta
Entrada	Confirmar pregunta de cerrar sesión.

Salida	Volver a la página de inicio o home.
Precondición	Usuario con sesión iniciada y se ha seleccionado la opción de cerrar sesión.
Postcondición	Sesión cerrada.
Descripción	Un usuario con su sesión iniciada en el sistema decide cerrarla, para ello selecciona la opción de cerrar sesión, aparece un popup con la pregunta y dos opciones: sí o cancelar. Si pulsa sí, se cierra la sesión, se le redirige a la pantalla principal y no puede seguir modificando libros ni bibliotecas. En caso contrario se queda como estaba.

4.3. MÓDULO GESTIÓN DE BIBLIOTECAS DE LIBROS

El módulo de las bibliotecas de libros comprende toda la funcionalidad relativa a la creación, borrado, gestión y visualización de bibliotecas de libros. Una biblioteca es una lista donde los usuarios insertan libros con el objetivo de clasificarlos o simplemente guardarlos para leerlos en otro momento. En la figura 2 se muestra el diagrama de casos de uso asociado a este módulo.

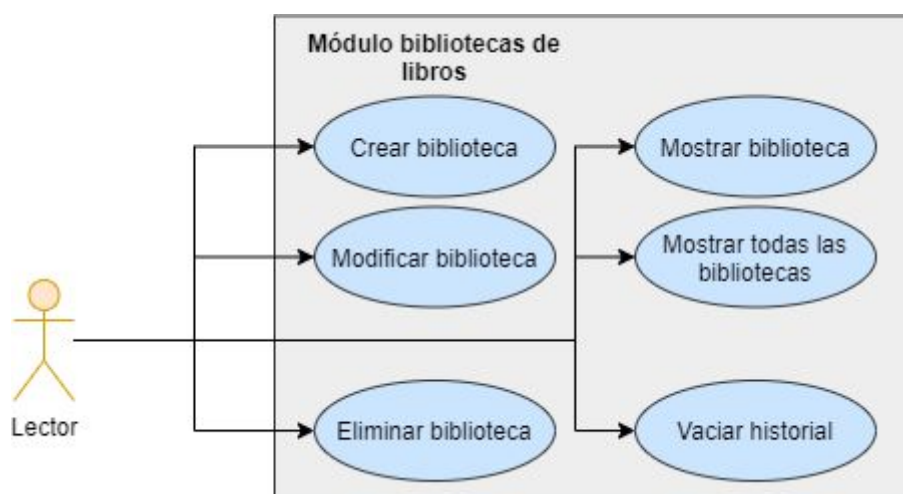


Figura 2. Diagrama de casos de uso del módulo bibliotecas de libros.

A continuación, se describe con detalle de cada uno de los casos de uso asociados a este módulo:

Identificador	BI01
Título del requisito	Crear biblioteca
Actores	Lector
Función	Crear una biblioteca

Prioridad	Baja
Entrada	Pulsar en el botón de crear biblioteca e introducir un nombre
Salida	Se crea una nueva biblioteca con el nombre asociado
Precondición	Ser un usuario con la sesión iniciada.
Condición	<ol style="list-style-type: none"> 1. Introducir un nombre distinto al de cualquier otra biblioteca existente del usuario en el momento de crearlo. 2. Introducir una descripción si se quiere. 3. Elegir la privacidad (público, privado). Por defecto, la privacidad es pública.
Postcondición	En el perfil del usuario se crea una biblioteca con el nombre asociado
Descripción	Cuando se quieren ordenar o almacenar distintos libros en un perfil, ya sean libros del perfil del usuario o de otros usuarios, se pueden crear bibliotecas donde ir agregandolos para poder verlos cuando se quiera de una forma sencilla y en forma de lista.

Identificador	BI02
Título del requisito	Modificar biblioteca
Actores	Lector
Función	Modificar biblioteca creada por el usuario siempre que no sea una creada automáticamente por el sistema
Prioridad	Baja
Entrada	Pulsar el botón de editar biblioteca
Salida	La biblioteca se modifica si los cambios son válidos
Precondición	<ol style="list-style-type: none"> 1. Ser un usuario con la sesión iniciada. 2. Que la biblioteca exista
Postcondición	En el perfil del usuario se actualiza la biblioteca con los cambios aplicados. Si no hay ninguno, se deja como estaba.
Descripción	<p>Si un usuario no se siente conforme con las características de sus bibliotecas, es libre de modificarlas cuando quiera para poder adaptarlas a sus necesidades. Estas modificaciones pueden ser:</p> <ol style="list-style-type: none"> 1. Cambiar el nombre a uno que no esté siendo usado por otra biblioteca del usuario en el momento de cambiarlo.

	<ol style="list-style-type: none"> 2. Eliminar un libro de la biblioteca, realizándose desde la vista al abrir la biblioteca. 3. Cambiar el orden de los libros de la biblioteca. 4. Cambiar la descripción, incluido borrarla. 5. Cambiar la privacidad a público o privado. <p>Una vez se han hecho las modificaciones deseadas se pulsa el botón de guardar. Si no hay cambios, se queda como estaba.</p>
--	--

Identificador	BI03
Título del requisito	Eliminar biblioteca
Actores	Lector
Función	Eliminar biblioteca creada por el usuario siempre que no sea una creada automáticamente por el sistema
Prioridad	Baja
Entrada	Pulsar el botón de eliminar biblioteca
Salida	Si la biblioteca no es una de las creadas automáticamente por el sistema, se elimina
Precondición	Ser un usuario con la sesión iniciada.
Postcondición	En la sección de bibliotecas del usuario, la biblioteca que se ha eliminado desaparece y deja de mostrarse.
Descripción	Si un usuario desea borrar una biblioteca, puede hacerlo simplemente pulsando el botón correspondiente. Al hacerlo, saltará un mensaje que pedirá confirmación de eliminación. Si se acepta, la biblioteca se borra por completo sin posibilidad de recuperarla.

Identificador	BI04
Título del requisito	Mostrar biblioteca
Actores	Lector
Función	Mostrar los libros de la biblioteca
Prioridad	Baja
Entrada	Pulsar sobre una biblioteca

Salida	Se muestran todos los libros que han sido añadidos en esa biblioteca en formato de lista, pudiendo ver de cada uno su nombre y su descripción.
Precondición	<ol style="list-style-type: none"> 1. Ser un usuario con la sesión iniciada. 2. Que exista una biblioteca
Postcondición	Se visualizan todos los libros de la biblioteca.
Descripción	Cuando se desee ver el contenido que se encuentra dentro de una biblioteca, se pulsa sobre ella para de esa forma, mostrar en formato de lista todos los libros que han sido añadidos.

Identificador	BI05
Título del requisito	Mostrar todas las bibliotecas
Actores	Lector
Función	Mostrar todas las bibliotecas del usuario
Prioridad	Baja
Entrada	Pulsar sobre el apartado correspondiente a la lista de bibliotecas
Salida	Todas las bibliotecas que el usuario tenga asociadas a su perfil se muestran en formato de lista, mostrando su nombre y su descripción.
Precondición	Ser un usuario con la sesión iniciada.
Postcondición	Se muestran todas las bibliotecas asociadas a ese usuario
Descripción	Cuando el usuario desee ver todas las bibliotecas que ha ido creando, puede hacerlo simplemente pulsando sobre el botón de la lista de bibliotecas. De esta forma, todas las bibliotecas asociadas a su perfil (las creadas automáticamente por el sistema y las creadas manualmente por él) se muestran en formato de lista.

Identificador	BI06
Título del requisito	Vaciar historial
Actores	Lector
Función	Permite vaciar la biblioteca en la que se guardan los últimos libros abiertos.

Prioridad	Baja
Entrada	Pulsar el botón de vaciar
Salida	El historial se queda vacío.
Precondición	Ser un usuario con la sesión iniciada.
Postcondición	Dentro del historial no hay ningún libro.
Descripción	La biblioteca historial es una biblioteca especial creada por el sistema que añade un libro automáticamente cada vez que se lee. Si un usuario desea vaciar su historial puede hacerlo pulsando el botón con esta opción.

4.4. MÓDULO GESTIÓN DE LIBROS

Este módulo representa la funcionalidad referida a la gestión de los libros del usuario (creación, modificación, edición, borrado, publicación o descarga). El escritor es el actor encargado de la gestión de todo lo concerniente a sus libros en la aplicación. En la figura 3 se muestra el diagrama de casos de uso asociado a este módulo:

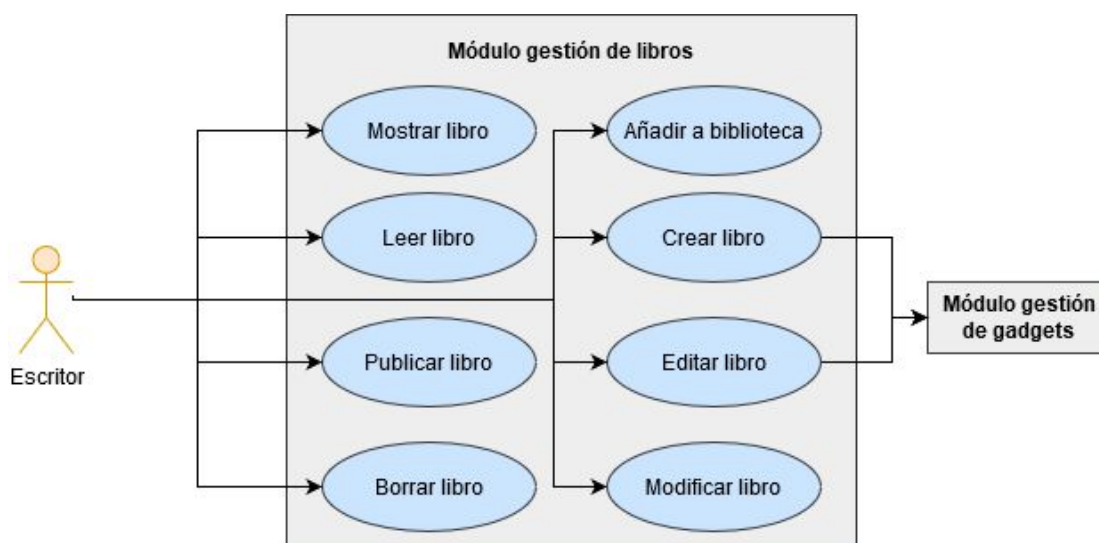


Figura 3. Diagrama de casos de uso del módulo libro.

A continuación, se describe con detalle cada uno de los casos de uso asociados a este módulo:

Identificador	LI01
Título del requisito	Crear libro
Actores	Escritor

Función	Crear un libro
Prioridad	Alta
Entrada	Obligatorio: nombre del título, incluir su nombre o nick name. Opcional: etiquetas, portada, descripción.
Salida	El libro se añade a la lista de libros del usuario.
Precondición	Usuario con sesión iniciada
Postcondición	Libro añadido a la base de datos y al apartado de ‘mis libros’ del usuario.
Descripción	El usuario selecciona la opción de crear un libro, introduce un título único y elige si se mostrar el nick o el nombre de usuario como nombre del autor para la obra y, si lo desea, una descripción y unas etiquetas. A continuación, selecciona crear y si todo está bien se añade a la BBDD, y si no, se le muestra un mensaje de error en la misma pantalla de creación para que vuelva a introducir la información correctamente.

Identificador	LI02
Título del requisito	Modificar libro
Actores	Escritor
Función	Modificar datos del libro
Prioridad	Alta
Entrada	Pulsar opción modificar
Salida	El libro se actualiza
Precondición	Tiene que existir el libro
Postcondición	Se actualiza la base de datos con la nueva información
Descripción	Cuando el usuario selecciona el libro que desea modificar, se da la opción de modificar el título, las etiquetas, la descripción, la portada o cambiar el estado de publicación..

Identificador	LI03
Título del requisito	Editar libro

Actores	Escritor
Función	Escribir libro
Prioridad	Alta
Entrada	Pulsar opción “editar”
Salida	El libro se actualiza
Precondición	Tiene que existir el libro
Postcondición	Se actualiza la base de datos con la nueva información
Descripción	Cuando el usuario elige la opción de editar del libro seleccionado, accede a una vista desde la que puede seguir escribiendo y añadiendo gadgets, El libro sólo se puede editar mientras no está publicado. Se puede dar a la opción de previsualizar para ver cómo está quedando, y una vez está listo, al botón de guardar, actualizando así la base de datos.

Identificador	LI04
Título del requisito	Borrar libro
Actores	Escritor
Función	Borrar un libro
Prioridad	Media
Entrada	Pulsar en el botón de borrar dos veces.
Salida	Se elimina el libro de la base de datos
Precondición	1. El usuario tenga la sesión iniciada 2. El libro exista y pertenezca a sus obras
Postcondición	El libro ya no aparece en la página web y tampoco está en la BBDD
Descripción	El usuario con sesión iniciada selecciona la opción de borrar en uno de sus libros creados. A continuación se resalta el botón con esta opción y en caso de estar seguro de eliminarlo, pulsa otra vez sobre la opción para volver a confirmarlo. Al pulsar por segunda vez el botón se borra de la BBDD de manera que deje de aparecer en la web. Si tras pulsar la opción pulsa en cualquier otro lado, se cancela la operación y no se realiza ningún cambio.

Identificador	LI05
Título del requisito	Mostrar libro
Actores	Escritor
Función	Mostrar tarjeta del libro
Prioridad	Alta
Entrada	Acceder a la vista de 'Inicio' o 'Mis libros'
Salida	Muestra la tarjeta del libro
Precondición	El libro existe
Postcondición	Se muestra una tarjeta con la información del libro
Descripción	Se accede a una de las vistas de 'Inicio' o 'Mis libros', desde ahí se visualizan las tarjetas de los libros que contienen su información: la imagen de portada, el título del libro, el nombre o nick del autor, la descripción del libro y las etiquetas. Desde la vista de 'Mis libros' también se puede ver un icono de un ojo tachado o sin tachar que indica si el libro no está o si está publicado.

Identificador	LI06
Título del requisito	Publicar libro
Actores	Escritor
Función	Publicar el libro para acceso a todos los usuarios
Prioridad	Media-alta
Entrada	Pulsar la opción de publicar
Salida	El libro es visible para todos.
Precondición	El libro existe.
Postcondición	Se actualiza la base de datos y es accesible por los demás usuarios.
Descripción	Una vez el libro está listo, se selecciona la opción de publicar, garantizando el acceso al libro para todos los lectores tal y como quedaba al previsualizar. Si el libro ya estaba publicado y se quiere modificar, es necesario despublicarlo para poder realizar los cambios deseados

Identificador	LI07
Título del requisito	Leer libro
Actores	Escritor
Función	Mostrar el libro como lectura
Prioridad	Alta
Entrada	Seleccionar opción lectura o previsualizar
Salida	Muestra el libro para poder interactuar como haría el lector
Precondición	El libro existe
Postcondición	Se muestra el libro en su formato final
Descripción	Se selecciona el libro que se desea leer o previsualizar, se muestra con formato de lectura para ver como quedaría al final. A la opción de previsualización se puede acceder desde la pantalla de edición de libro, y se puede interactuar como si fuera el libro publicado, para así poder comprobar si funciona todo correctamente. Para acceder a la pantalla de lectura basta con acceder al libro desde la pantalla de inicio o desde la pantalla de 'mis libros' en cuestión.

Identificador	LI08
Título del requisito	Añadir a biblioteca
Actores	Escritor
Función	Añadir el libro deseado a una biblioteca
Prioridad	Media
Entrada	Elegir a la biblioteca que se desea añadir el libro
Salida	El libro aparece en la biblioteca seleccionada
Precondición	El libro y la biblioteca existen.
Postcondición	El libro queda relacionado con la biblioteca del usuario.
Descripción	Para organizar los libros en categorías personalizadas del usuario, se le da la opción al usuario de añadir libros a sus bibliotecas. Esta acción se realiza desde el libro y no desde la biblioteca en cuestión.

4.5. MÓDULO GESTIÓN DE SECCIONES

El módulo gestión de secciones representa todas las funcionalidades que permiten al escritor insertar, modificar, borrar y gestionar las secciones de un libro. En la figura 4 se muestra el diagrama de casos de uso asociado a este módulo.

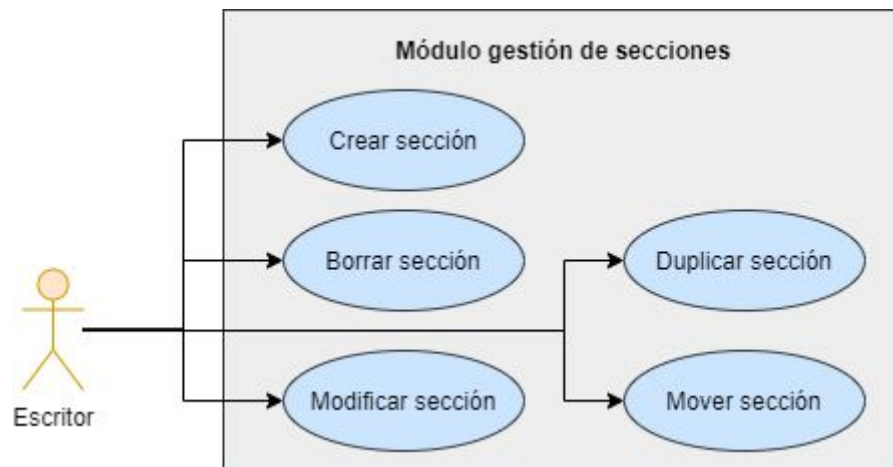


Figura 4. Diagrama de casos de uso del módulo secciones.

A continuación, se describe con detalle cada uno de los casos de uso asociados a este módulo:

Identificador	SC01
Título del requisito	Crear sección
Actores	Escritor
Función	Añadir una nueva sección al libro que se está editando
Prioridad	Alta
Entrada	Pulsar sobre la opción de nueva sección
Salida	Nueva sección vacía creada
Precondición	Estar editando un libro
Postcondición	Que se cree la sección en el libro
Descripción	En la pantalla de edición el escritor se encontrará con un panel desplegable donde se realiza la gestión de las secciones. Para añadir una nueva sección basta con pulsar en la opción de nueva sección, y esta se añadirá con un nombre predefinido y un id único.

Identificador	SC02
Título del requisito	Borrar sección
Actores	Escritor
Función	Borrar una sección del libro
Prioridad	Alta
Entrada	Elegir la opción de eliminar de la sección concreta
Salida	La sección se elimina del libro
Precondición	<ol style="list-style-type: none"> 1. Ser un usuario con la sesión iniciada 2. Estar editando un libro 3. Que exista la sección que se desea borrar 4. Que existan al menos dos secciones
Postcondición	En el libro, esa sección desaparece y no se puede volver a recuperar
Descripción	El autor puede querer borrar una sección entera, para ello en la ventana de gestión de secciones se le da la posibilidad de borrarla. Es necesario que, tras borrarla, un libro contenga al menos una sección, por lo que si sólo existe la sección que se desea borrar, esta posibilidad no aparece.

Identificador	SC03
Título del requisito	Modificar sección
Actores	Escritor
Función	Editar el nombre de una sección
Prioridad	media-alta
Entrada	Pulsar sobre el nombre de la sección que se desea modificar
Salida	Nombre de la sección actualizado
Precondición	Existe la sección
Postcondición	Se actualizan los detalles
Descripción	Una vez tenemos una sección creada, se puede editar el nombre con el que se representa a la sección, personalizando así más el libro.

Identificador	SC04
Título del requisito	Duplicar sección
Actores	Escritor
Función	Duplicar/clonar una sección
Prioridad	Media
Entrada	Selecciona la opción de clonar de una sección
Salida	Dos secciones con el mismo contenido
Precondición	Estar editando el libro y tener al menos una sección añadida
Postcondición	La sección queda duplicada en el libro
Descripción	El autor está en la ventana de gestión de secciones y tiene una sección ya creada, pulsa la opción de clonar y se duplica la sección con todo su contenido.

Identificador	SC05
Título del requisito	Mover sección
Actores	Escritor
Función	Modificar el orden de las secciones
Prioridad	media
Entrada	Selecciona una sección, pulsar opción de subir o bajar
Salida	Actualiza el libro con el nuevo orden
Precondición	Estar en gestión de secciones y que haya al menos dos secciones
Postcondición	Se mantiene el contenido pero cambia el orden
Descripción	El autor puede querer cambiar el orden de cómo aparecen sus secciones. Para ello, se le da la opción de subir o bajar, colocándolas antes de la anterior o después de la siguiente.

4.6. MÓDULO GESTIÓN DE GADGETS

El módulo gestión de gadgets representa todas las funcionalidades que permiten al escritor insertar, modificar, borrar y gestionar los gadgets empleados cuando se escriben libros. En la figura 5 se muestra el diagrama de casos de uso asociado a este módulo.

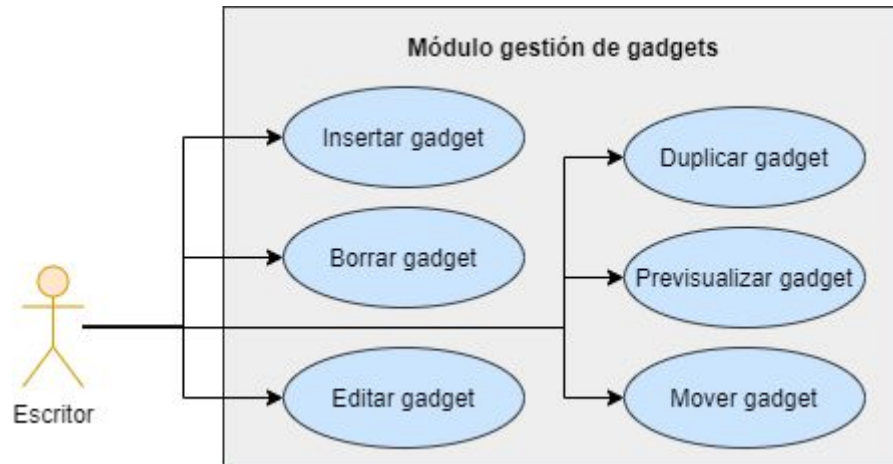


Figura 5. Diagrama de casos de uso del módulo gestión de gadgets.

A continuación, se describe con detalle cada uno de los casos de uso asociados a este módulo:

Identificador	GA01
Título del requisito	Insertar gadget
Actores	Escritor
Función	Insertar un gadget del panel de gadgets al libro que se está editando
Prioridad	Alta
Entrada	Pulsar sobre el gadget que se quiere insertar
Salida	Gadget insertado en el texto
Precondición	Estar editando un libro
Postcondición	Que se añada el gadget al libro
Descripción	En la pantalla de edición el escritor se encontrará con un panel donde estarán todos los gadgets. Para insertarlo a su libro solo tiene que pulsar sobre él y este se añadirá al inicio de la hoja o a continuación del gadget actualmente seleccionado. En el modo de edición del libro aparecerá el gadget con los parámetros que se pueden editar de cada uno, la vista en el modo lectura y escritura del gadget son diferentes.

Identificador	GA02
Título del requisito	Borrar gadget
Actores	Escritor
Función	Borrar una instancia de un gadget insertada por el usuario durante la edición de un libro
Prioridad	Alta
Entrada	Elegir la opción de eliminar del gadget concreto.
Salida	La instancia del gadget se borra del libro
Precondición	<ol style="list-style-type: none"> 1. Ser un usuario con la sesión iniciada 2. Estar editando un libro 3. Que esté insertado el gadget que se desea borrar
Postcondición	En el libro, esa instancia del gadget desaparece y no se puede volver a recuperar
Descripción	Tras insertar un gadget, es posible que se desee eliminarlo. Para ello, basta con pulsar el botón con icono de papelera sobre el gadget.

Identificador	GA03
Título del requisito	Editar gadget
Actores	Escritor
Función	Editar los parámetros de un gadget ya insertado
Prioridad	Media-alta
Entrada	Pulsar sobre el gadget ya insertado que se desea editar
Salida	Gadget con los parámetros actualizados
Precondición	El gadget está insertado
Postcondición	Se actualizan los detalles
Descripción	Una vez tenemos un gadget insertado, para modificarlo se pulsa sobre él y se modifican los parámetros que se deseen.

Identificador	GA04
---------------	------

Título del requisito	Duplicar gadget
Actores	Escritor
Función	Duplicar/clonar un gadget
Prioridad	Media
Entrada	Selecciona un gadget, copiarlo
Salida	El mismo gadget dos veces insertado en el libro
Precondición	Estar editando el libro y tener al menos un gadget añadido por el autor
Postcondición	El gadget queda duplicado en el libro
Descripción	El autor está editando su libro, tiene un gadget ya creado por él, lo selecciona, pulsa la opción de clonar y se inserta el mismo gadget duplicado en la sección.

Identificador	GA05
Título del requisito	Previsualizar gadget
Actores	Escritor
Función	Previsualizar un gadget
Prioridad	Media
Entrada	Selecciona un gadget, pulsar opción de previsualización.
Salida	Aparece cómo se vería el gadget en la lectura del libro.
Precondición	Estar editando el libro y tener al menos un gadget con esta opción añadido
Postcondición	Aparece una ventana nueva con el resultado final
Descripción	El autor quiere ver cómo es el resultado final del gadget que acaba de crear y está editando, para ello tiene que pulsar en la opción de previsualizar y aparecerá el gadget como se vería para el lector.

Identificador	GA06
Título del requisito	Mover gadget

Actores	Escritor
Función	Cambiar la posición del gadget en la sección
Prioridad	Media
Entrada	Selecciona un gadget, pulsar opción de subir o bajar
Salida	Actualiza el libro con el nuevo orden
Precondición	Estar editando el libro y que haya al menos dos gadgets
Postcondición	Se mantiene el contenido pero cambia el orden
Descripción	El autor puede querer cambiar el orden de sus gadgets. Para ello, se le da la opción de subir o bajar los gadgets, colocándolos antes del anterior o después del siguiente.

4.7. MÓDULO DE GADGETS

El módulo de gadgets incluye las especificaciones de cada gadget durante la edición del libro y su lectura. En la figura 6 se muestra el diagrama de casos de uso asociado a este módulo.

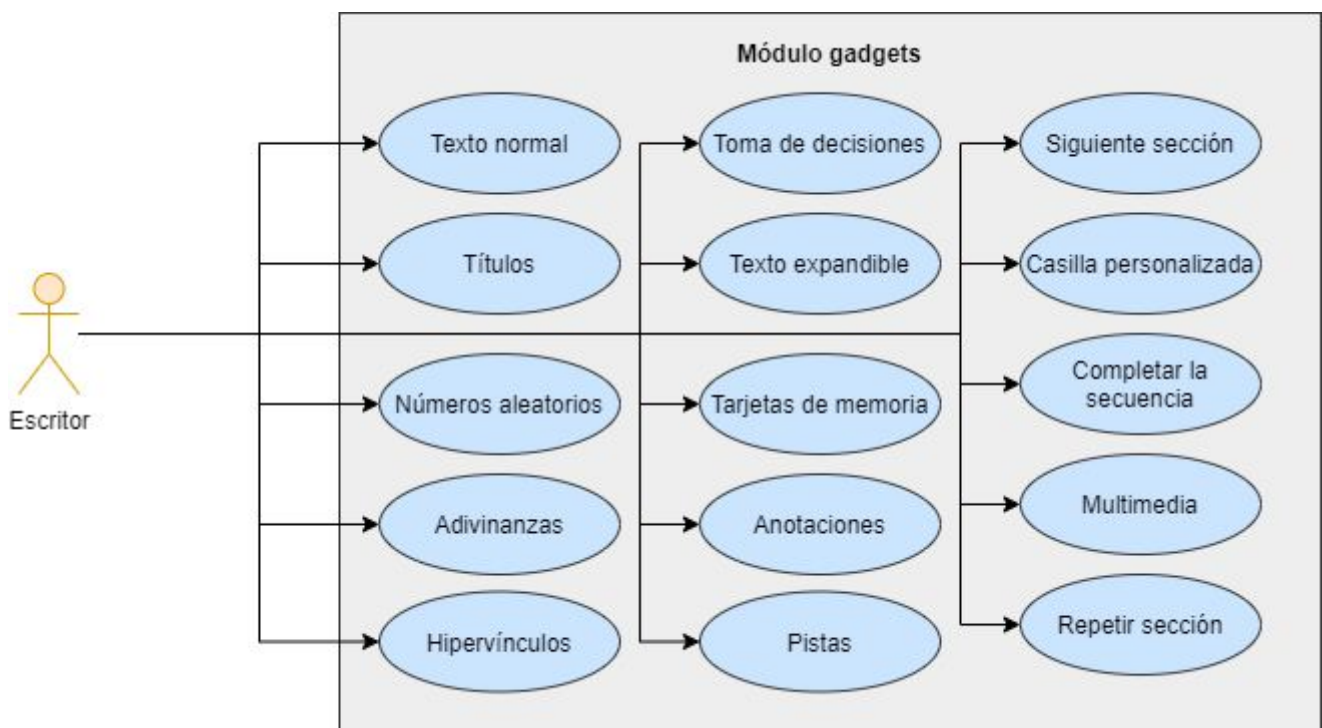


Figura 6. Diagrama de casos de uso del módulo gadgets.

A continuación, se describe con detalle cada uno de los casos de uso asociados a este módulo:

Identificador	TG01
Título del requisito	Texto normal
Actores	Escritor
Función	Escribir el contenido del libro. Se aplica, según se elija: escribir en negrita, cursiva, subrayado o tachado.
Prioridad	Media-alta
Entrada	Pulsar sobre el botón correspondiente a la funcionalidad.
Salida	El texto se comienza a escribir con los parámetros seleccionados.
Precondición	<ol style="list-style-type: none"> 1. Ser un usuario con la sesión iniciada. 2. Estar editando un libro.
Postcondición	En el libro se muestra el texto con los parámetros adecuados.
Descripción	Para escribir un libro es necesario poder incluir texto primero, y esa es la función de este gadget: habilitar la escritura de texto plano y añadir, además, la posibilidad de escribirlo en negrita, cursiva, subrayado, tachado o cualquier combinación de ellos.

Identificador	TG02
Título del requisito	Títulos
Actores	Escritor
Función	Añadir títulos
Prioridad	Alta
Entrada	El autor inserta un nuevo título
Salida	El elemento insertado al texto del libro
Precondición	Estar editando el libro
Postcondición	Insertar el título
Descripción	El autor que está editando el texto inserta un título al cual le puede cambiar el tamaño a su gusto, pudiendo elegir entre tres opciones.

Identificador	TG03
---------------	------

Título del requisito	Números aleatorios
Actores	Escritor
Función	Se muestran números aleatorios según el rango que el autor haya definido
Prioridad	Alta
Entrada	Botón izquierdo del ratón para obtener el valor aleatorio en caso de que el autor lo haya definido así
Salida	Un número aleatorios y cambio de sección según lo haya definido el autor.
Precondición	Estar leyendo o previsualizando un libro
Postcondición	Obtener el número aleatorio
Descripción	En ciertas ocasiones es necesario tener valores aleatorios para usarlos como medio interactivo. Con este sistema, si el escritor lo desea, podrá configurar un salto de sección para avanzar en la lectura dependiendo de los intervalos entre los que se encuentre el número obtenido.

Identificador	TG04
Título del requisito	Spoiler
Actores	Escritor
Función	Mostrar texto escrito sólo cuando se pasa el cursor por encima.
Prioridad	Alta
Entrada	Pasar el ratón por encima del texto de tipo spoiler
Salida	El texto se muestra
Precondición	Estar leyendo o previsualizando un libro
Postcondición	Se ve lo que está escrito
Descripción	El autor puede definir un texto de tipo spoiler, que aparece en negro sin nada escrito hasta que se pase el ratón por encima. El lector ve el fondo negro y cuando pasa el ratón por encima se muestran las letras en blanco, dando así la posibilidad de leer lo que pone.

Identificador	TG05
Título del requisito	Hipervínculos
Actores	Escritor
Función	Texto asociado a un link que al pulsarlo te abre una página externa.
Prioridad	Alta
Entrada	Pulsar sobre el texto con el link
Salida	Lugar a donde te envía el hipervínculo
Precondición	Estar leyendo o previsualizando un libro
Postcondición	Te envía el hipervínculo a donde sea necesario
Descripción	<p>Se añade un texto con un hipervínculo asociado. Para ello, se pulsa en el gadget, se agrega, se escribe el texto y el link asociado a donde se va a redirigir.</p> <p>A la hora de la lectura, para activarlo basta con pulsar sobre el texto que está resaltado en azul y se abrirá el link en una nueva pestaña en el navegador.</p>

Identificador	TG06
Título del requisito	Casilla personalizada
Actores	Escritor
Función	Personalizar la lectura con datos escritos por el lector
Prioridad	Alta
Entrada	Introducción de datos personalizados en función de lo que te pida la casilla en particular: número o texto.
Salida	Aparece el dato introducido como parte del libro.
Precondición	Estar leyendo o previsualizando un libro
Postcondición	Que se cree una casilla con el valor indicado.
Descripción	El escritor introduce una casilla para personalizar la lectura, con algún parámetro de texto o número. Al principio, hay que seleccionar

	<p>qué tipo de dato se desea introducir y a partir de ahí aparecen las opciones características de cada uno. Si se quiere que ese dato se repita a lo largo del libro, basta con copiar la casilla e introducirla en modo lectura donde se desee. La información de la casilla se mantiene, y si a lo largo de la lectura el autor desea dar la opción de modificar el dato ya introducido, puede hacerlo con la casilla configurada para desempeñar esa tarea.</p> <p>De cara al lector, este simplemente tiene que pulsar en la casilla e introducir el tipo de dato que se le pide y vea apropiado.</p>
--	--

Identificador	TG07
Título del requisito	Toma de decisiones
Actores	Escritor
Función	Al lector se le presentan distintas opciones de las que tiene que seleccionar una que determinara como continua la historia
Prioridad	Alta
Entrada	Pulsar sobre una de las opciones
Salida	Avanzar en el texto
Precondición	Estar previsualizando o leyendo un texto
Postcondición	Avanzar en el texto al punto del libro que corresponda a la decisión que a tomado el lector
Descripción	<p>El escritor elige varias opciones que el usuario va a poder seleccionar durante la historia, y configura cada una para que envíe a distintas partes del libro.</p> <p>Cuando estás leyendo un libro y llegas a una parte donde el autor ha introducido un gadget de este tipo, se te presentan varias opciones entre las que tienes que seleccionar una. Al seleccionarla, la historia avanzará a la sección que el autor haya asociado con esa opción.</p>

Identificador	TG08
Título del requisito	Texto expandible
Actores	Escritor

Función	Se expande un texto hacia abajo con la información adicional.
Prioridad	Alta
Entrada	Botón izquierdo del ratón sobre el texto o el icono de ampliar
Salida	Se expande el texto contraído
Precondición	Estar previsualizando o leyendo un libro
Postcondición	La utilidad aparece mostrada en el libro
Descripción	<p>Cuando un escritor desea incluir información extra sobre un determinado tema, puede recurrir a un texto expandible. Este gadget, asociado a determinadas partes del libro, permite mostrar texto que anteriormente estaba contraído para ampliar o complementar lo que el autor desee, desplazando todo el texto posterior.</p> <p>El lector simplemente tendrá que pulsar el texto o el icono de expandir correspondiente para mostrar el texto oculto.</p>

Identificador	TG09
Título del requisito	Tarjetas de memoria
Actores	Escritor
Función	Juego de descubrir las parejas
Prioridad	Media-baja
Entrada	Botón izquierdo del ratón para girar una cartas
Salida	Se muestran todas las cartas, las volteadas y las que no
Precondición	Estar previsualizando o leyendo un libro
Postcondición	<p>Dependiendo del resultado te envía a un sitio del libro u otro.</p> <p>La carta pulsada se voltea. Si son dos distintas las volteadas (y mostradas), se vuelven a girar automáticamente pasados unos segundos</p>
Descripción	<p>Pequeño juego de encontrar las parejas. Dado un tablero de $n \times n$, donde n es el tamaño deseado por el escritor, se rellena de cartas boca abajo con parejas en su otra cara. Cada vez que se pulse sobre una de ellas se voltea, y si son dos las mostradas cara arriba, se comprueba si son iguales. Si lo son, se pueden volver a dar la vuelta</p>

	<p>otras dos. Si no, se vuelven a poner boca abajo y hay que volver a seleccionar otras dos. El autor puede decidir si las parejas se van a tratar de colores, ya predefinidos o elegidos por él, o palabras personalizadas. También puede elegir cuántas parejas hay que encontrar y cuál es el número máximo de movimientos.</p> <p>El lector debe pulsar sobre las cartas para darles la vuelta.</p>
--	---

Identificador	TG10
Título del requisito	Anotaciones
Actores	Escritor
Función	Mostrar una pantalla emergente con la nota del autor.
Prioridad	Alta
Entrada	Pulsar sobre el icono que representa una anotación.
Salida	Aparece ventana emergente con la información.
Precondición	Estar leyendo o previsualizando un libro
Postcondición	Que aparezca la anotación
Descripción	<p>El autor asocia una parte del texto a una nota escrita por él, la cual puede contener una imagen, que aparecerá con la anotación en una ventana emergente.</p> <p>Al leer un libro en el cual el autor ha añadido una nota, el lector podrá pulsar sobre ella para abrir una ventana emergente con información adicional.</p>

Identificador	TG11
Título del requisito	Completar la secuencia
Actores	Escritor
Función	Reta al usuario con una secuencia incompleta
Prioridad	Alta
Entrada	Elementos ausentes en la secuencia
Salida	Avanzar en la lectura

Precondición	Estar leyendo o previsualizando un libro
Postcondición	Avanzar en la historia
Descripción	<p>El escritor puede introducir el gadget y generar una secuencia de números y/o letras para que el lector complete. Es el escritor el que decide qué parte de la secuencia se muestra y cuál tiene que completar el lector. También puede definir cuantos intentos tiene para completarlo y a dónde te lleva la lectura dependiendo del número de intentos necesarios para resolverlo o si se ha equivocado o no.</p> <p>Mientras se está leyendo el libro aparece una secuencia incompleta. Cuando el usuario introduce los elementos restantes, se le permite avanzar en la historia.</p>

Identificador	TG12
Título del requisito	Multimedia
Actores	Escritor
Función	Se agrega y se visualiza contenido multimedia: imágenes o vídeos.
Prioridad	Media - baja
Entrada	Arrastrar una imagen o vídeo con el botón izquierdo del ratón o buscarla manualmente en la opción adecuada pulsando con el botón izquierdo del ratón
Salida	Se muestra el contenido multimedia
Precondición	<ol style="list-style-type: none"> 1. Ser un usuario con la sesión iniciada 2. Estar editando un libro
Postcondición	El contenido se muestra en el lugar donde se ha añadido
Descripción	Para ilustrar lo que se está contando se puede recurrir a la inserción de contenido multimedia. Este contenido puede ser tanto en forma de imagen como de video, y se puede insertar en cualquier lugar del libro.

Identificador	TG13
Título del requisito	Adivinanzas

Actores	Escritor
Función	Reta al usuario con una adivinanza
Prioridad	Alta
Entrada	Solución a la adivinanza
Salida	Muestra la continuación de la historia
Precondición	Estar leyendo o previsualizando el libro
Postcondición	Progresar con la historia
Descripción	<p>Una adivinanza es un texto enigmático sin resolver y que tiene una solución asociada a ello. Cuando el escritor introduce este gadget, debe escribir en un campo el texto de la adivinanza y en otro la solución que el lector deberá encontrar. Se puede elegir el número de intentos que tendrá el lector para resolverla. Además, se puede escoger qué camino tomará el libro según si el lector encuentra o no la solución.</p> <p>Cuando el lector se encuentre con una adivinanza, verá el texto del enigma y un cuadro en blanco donde podrá escribir la solución.</p>

Identificador	TG14
Título del requisito	Siguiente sección
Actores	Escritor
Función	Cambia de sección
Prioridad	Alta
Entrada	Pulsar sobre el icono de cambio de sección
Salida	Muestra la continuación de la historia
Precondición	Estar leyendo o previsualizando el libro
Postcondición	Progresar con la historia
Descripción	<p>El autor puede decidir dar la posibilidad de continuar con la historia sin necesidad de ningún juego, para ello utilizará este gadget, que consiste en un salto a la sección elegida. A este cambio de sección lo puede acompañar de texto personalizado.</p> <p>El lector verá el texto acompañado de un icono y cuando quiera</p>

	cambiar de sección bastará con pulsar el icono.
--	---

Identificador	TG15
Título del requisito	Repetir sección
Actores	Escritor
Función	Vuelve a cargar la sección actual
Prioridad	Alta
Entrada	Pulsar sobre el icono que denota repetir sección.
Salida	Vuelve a cargar la sección restableciendo los gadgets antes de ser resueltos.
Precondición	Estar leyendo o previsualizando el libro
Postcondición	Volver a leer la sección
Descripción	<p>El autor puede dar la opción de volver a probar su suerte con la sección del libro al lector. Para ello dispone de este gadget, que le permite, acompañado de texto, facilitar recargar la sección para empezar de cero.</p> <p>El lector simplemente tiene que pulsar sobre la opción de recargar sección.</p>

Identificador	TG16
Título del requisito	Pistas
Actores	Escritor
Función	Reta al usuario con unas pistas
Prioridad	Alta
Entrada	Solución a las pistas
Salida	Muestra la continuación de la historia
Precondición	Estar leyendo o previsualizando el libro
Postcondición	Progresar con la historia

Descripción	<p>El autor define una serie de soluciones posibles y elige cuál sería la elección correcta por parte del lector. Puede personalizar el texto que acompaña al gadget y a qué sección se realizará el salto dependiendo de si contesta correctamente o no.</p> <p>Cuando el lector se encuentre con unas pistas, podrá elegir una de las opciones definidas por el autor y cuando esté listo, comprobará su respuesta, continuando con su lectura dependiendo de a dónde le lleve su respuesta.</p>
-------------	--

5. CAPÍTULO V: MODELO DE DATOS

En este capítulo se explica el modelo de datos que se ha definido para realizar persistencia de la información de la aplicación.

5.1. MODELO ENTIDAD-RELACIÓN

En la figura 7 se muestra el diagrama Entidad-Relación en el cual se representan las entidades del sistema, sus correspondientes atributos y las relaciones entre las mismas.

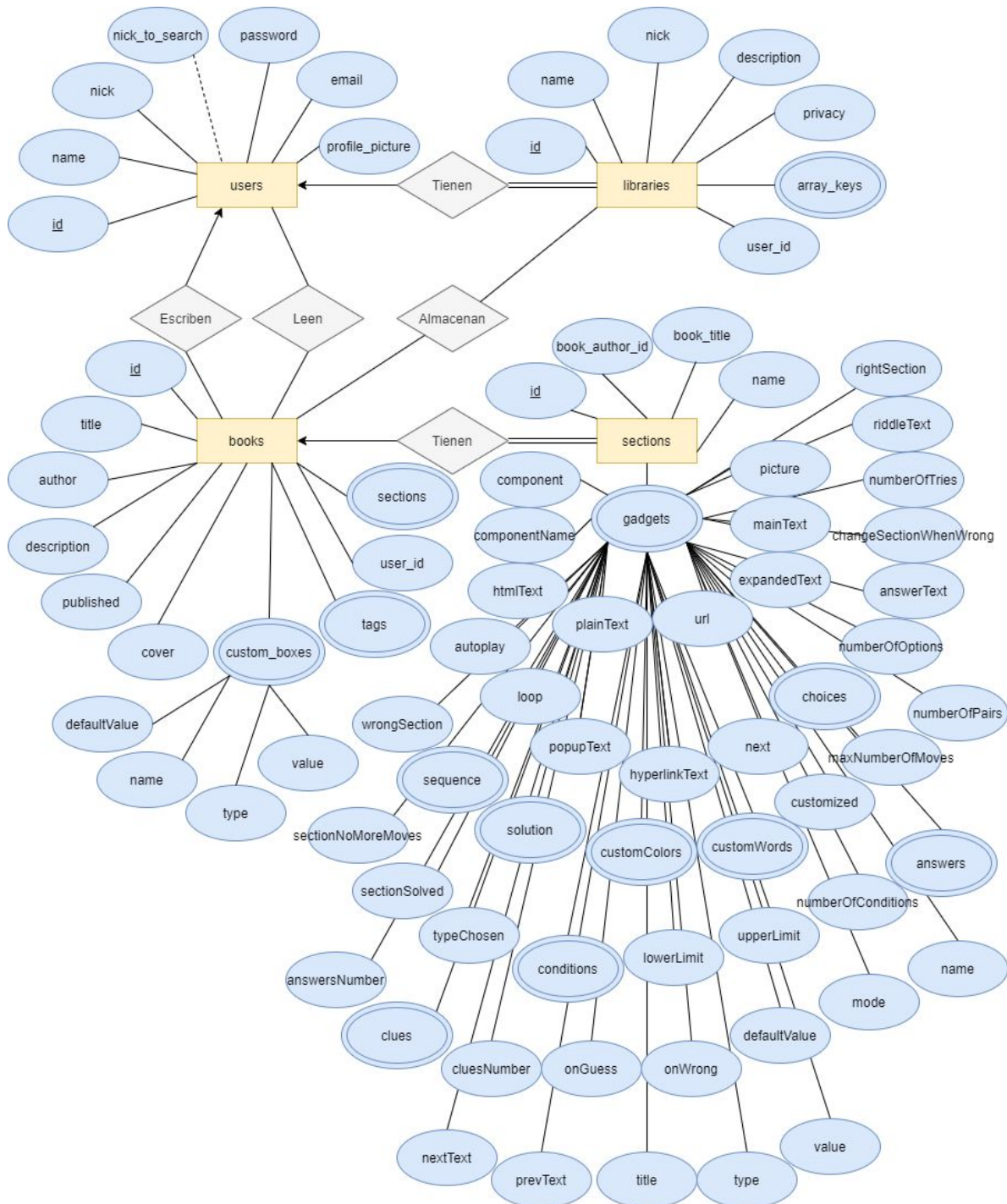


Figura 7. Modelo entidad-relación.

A continuación se explica qué representa cada entidad del diagrama anterior:

- *Users*: Representa a los usuarios, es decir, las personas que usan la aplicación. Por ello, esta entidad va enfocada a representar los datos personales y distintivos de cada uno tal como nombre, nick, correo y contraseña. Cuenta con restricciones de forma que los nicks son únicos para cada usuario. Los usuarios son la base del modelo de datos, y a partir de aquí se derivan el resto de entidades.
- *Libraries*: Las bibliotecas son listas pensadas para que los usuarios guarden conjuntos de libros, aunque también pueden estar vacías. Su nombre actúa como un identificador, de modo que no puede haber dos bibliotecas con el mismo nombre para el mismo usuario. Las bibliotecas, además, cuentan con un atributo que indica quien es el usuario propietario.
- *Books*: Los libros son la colección que representa los datos básicos del libro. Es decir, esta entidad no representa el cuerpo del libro, sino sus datos asociados tal como su nombre, descripción, privacidad... además de otros atributos especiales como un listado con las claves primarias de las secciones y otro con los datos básicos de las casillas personalizadas, que además es compuesto. No pueden existir dos libros con el mismo nombre dentro de la aplicación.
- *Sections*: Las secciones son la entidad más compleja de la aplicación. Contienen todo el cuerpo del libro con sus distintas características, además de un nombre y la información del autor y el libro, para crear una doble referencia. El cuerpo del libro se representa en el atributo compuesto gadgets, y como dicho cuerpo está constituido por cualquier tipo de gadget presente en la aplicación, se divide en todos y cada uno de los distintos atributos simples que componen cada gadget, siendo los comunes entre todos component y componentName, que representan el identificador del componente y su nombre, respectivamente.

Las relaciones entre las entidades, así como las acciones que se realizan en ellas, se han establecido de la siguiente manera:

- Un usuario debe tener como mínimo una biblioteca, mientras que una biblioteca sólo puede tener un usuario.
- Un usuario puede escribir tantos libros como desee pero cada libro sólo puede ser escrito por un usuario.
- Un usuario puede leer tantos libros como desee y los libros pueden ser leídos por todos los usuarios que quieran.
- Las bibliotecas contienen tantos libros como se quiera y los libros pueden ser contenidos por cualquier biblioteca.
- Los libros deben tener como mínimo una sección y las secciones sólo pueden pertenecer a un libro.

5.2. BASE DE DATOS NOSQL

Para implementar el modelo Entidad-relación descrito anteriormente se ha empleado Firebase Firestore [13] que permite agrupar cada entidad y atributo en colecciones y documentos, facilitando su posterior manipulación en la aplicación. Firebase Firestore, además, genera IDs automáticos para cada documento. Estos IDs son las claves primarias de la base de datos a través de las cuales se relacionan las colecciones. Teniendo todo esto anterior en cuenta, se ha decidido organizar los datos de la siguiente forma¹, tal y como está reflejado en la figura 8:

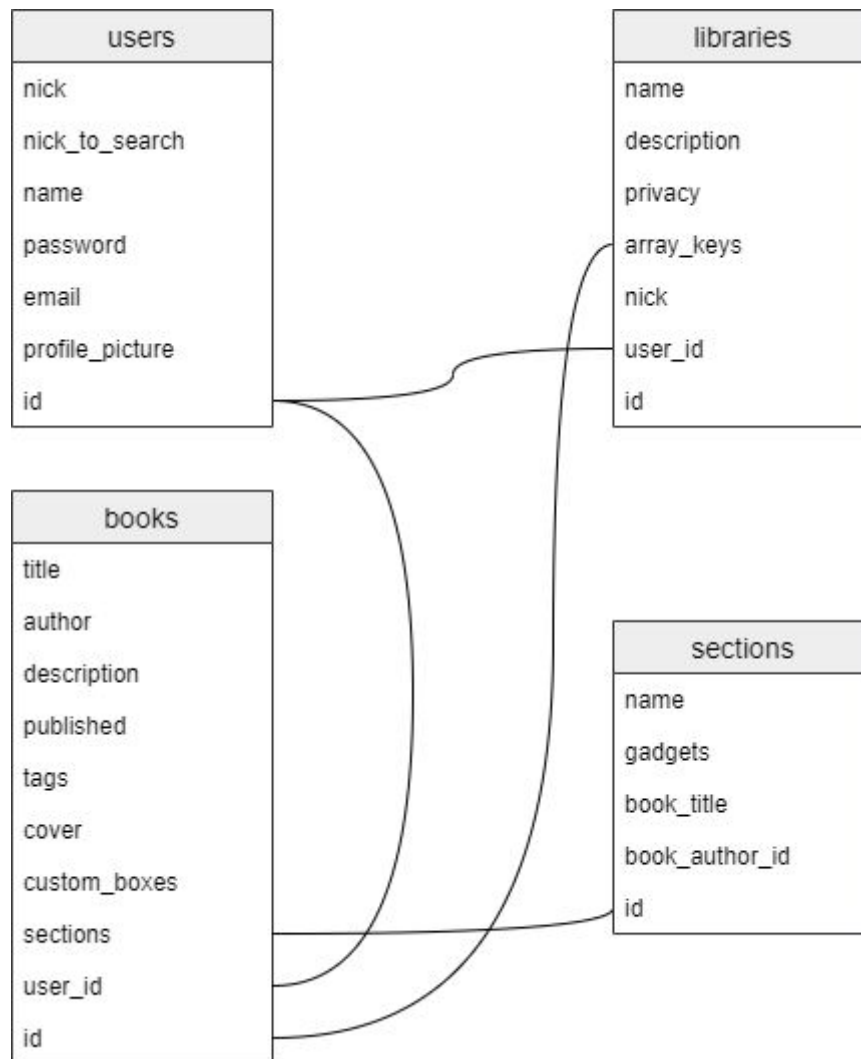


Figura 8. Esquema de la base de datos.

A continuación, se explica cada colección y los campos que contiene cada documento:

¹ Por cuestiones de claridad, en el siguiente diagrama no se reflejan los atributos simples derivados de los compuestos

5.2.1. USERS

La colección de usuarios (figura 9) contiene la información relevante de cada persona registrada en la aplicación, es decir, aquellas personas que usan la aplicación sin registrarse no tienen entrada en la base de datos.

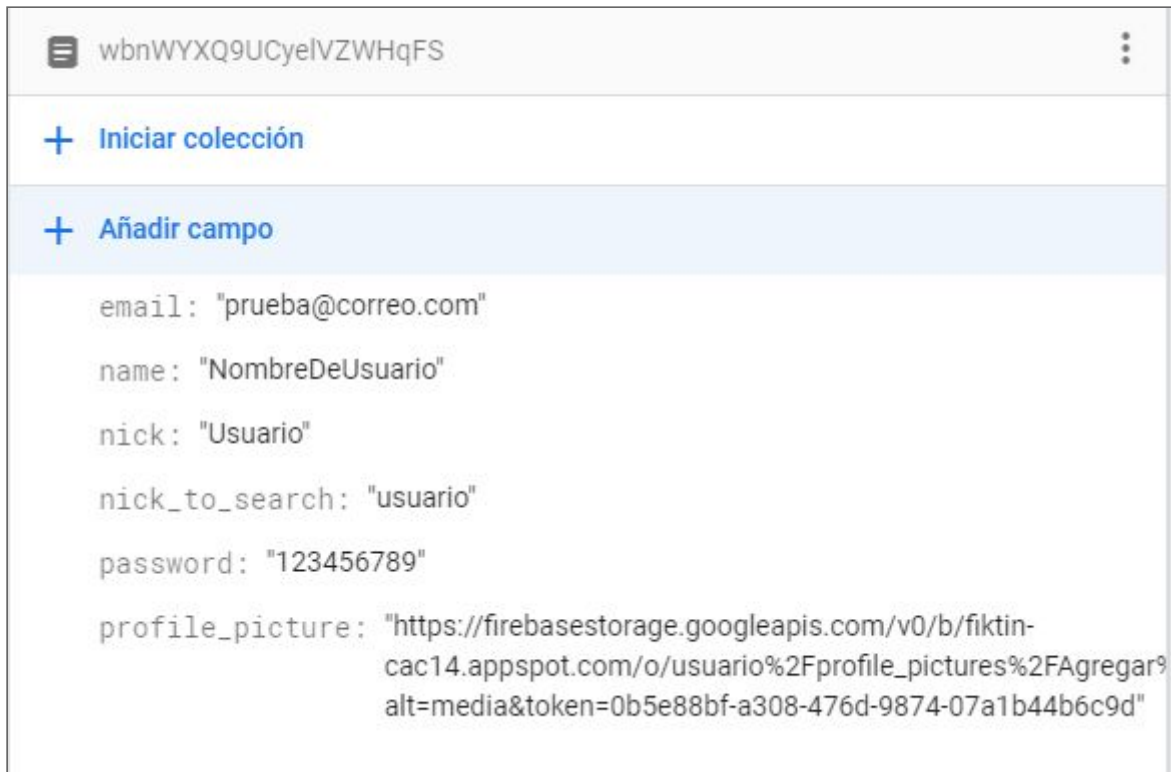


Figura 9. Documento de colección de usuarios.

La colección está formada por los siguiente campos:

- *nick*: El nick representa el alias del usuario. Es una clave única, por lo que aunque no actúe como clave primaria, no puede repetirse entre dos usuarios distintos. Es un campo sensible a mayúsculas y minúsculas.
- *nick_to_search*: Este campo es un atributo derivado de *nick*, que contiene su valor escrito íntegramente en minúsculas para prevenir la sensibilidad de dicho campo. Se usa para comprobar si el nick de un usuario está repetido o no.
- *name*: Representa el nombre del usuario. A diferencia del *nick*, este campo no es una clave única, por lo que varios usuarios pueden llamarse de la misma manera.
- *password*: Es la contraseña del usuario dada como medida de seguridad. Para poder establecerla, es necesario que siga las condiciones que se encuentran descritas en los casos de uso.
- *email*: Es el correo del usuario, aunque actualmente no se usa. Debe ser un correo válido según las normas descritas en los casos de uso.

- *profile_picture*: Representa la dirección a la imagen de perfil del usuario, ya que dicha imagen se encuentra en el almacenamiento de Firebase.
- *id*: Clave primaria generada de manera automática por Firebase.

5.2.2. LIBRARIES

La colección *libraries* (figura 10) contiene la información relevante de las bibliotecas asociadas a un usuario. Todos los usuarios tienen siempre una o más bibliotecas (siempre existe la biblioteca historial que sirve para almacenar los libros que el usuario haya ido leyendo, la cual no se puede borrar).

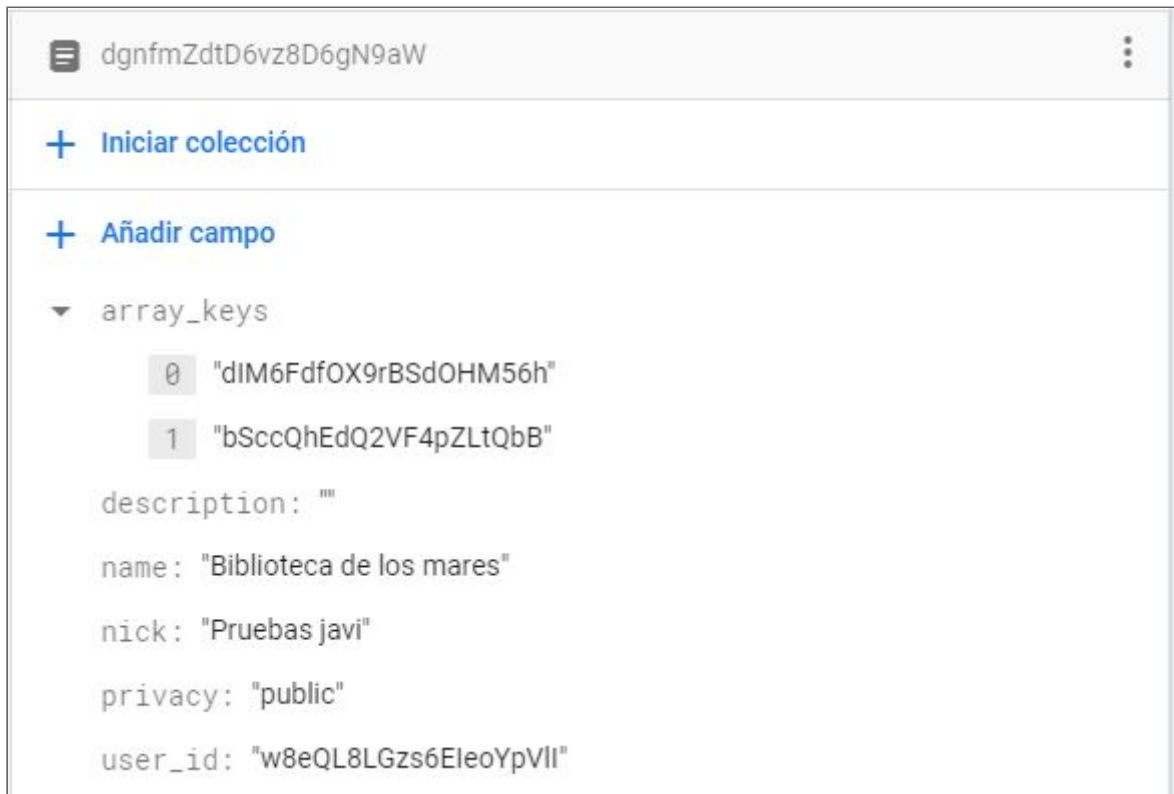


Figura 10. Documento de colección de bibliotecas.

En esta colección se tienen los campos:

- *name*: Corresponde al nombre de la biblioteca. Es una clave única por cada usuario. Hay que tener en cuenta que el nombre es sensible a las mayúsculas y minúsculas.
- *description*: Corresponde a la descripción de la biblioteca. No es obligatorio rellenar este campo en la aplicación tal y como se puede observar en la figura 10, donde la descripción se encuentra vacía.
- *privacy*: La privacidad de una biblioteca indica si otros usuarios pueden verla o no. Tiene dos valores únicos: *public* y *private*, que indican si la biblioteca es pública o privada, respectivamente.

- *array_keys*: Este campo es un array que contiene las IDs, en formato de String, de los libros que se encuentran en la biblioteca. Puede estar vacío y no tiene un límite de tamaño. No puede haber dos IDs iguales.
- *nick*: Nick del usuario.
- *user_id*: Corresponde al id único de cada usuario, mencionado en la colección *users*. Es necesario para identificar al autor de la biblioteca y para no poder poner dos nombres iguales al crearlas.
- *id*: Corresponde a la clave primaria de la biblioteca. Hay dos tipos de claves primarias: Las generadas automáticamente por Firebase y las claves primarias del historial, que son establecidas concatenando el nick del usuario y “_historial”.

5.2.3. BOOKS

La colección *books* (figura 11) contiene la información básica de los libros, no su contenido. Un usuario puede tener o no libros creados; no existen restricciones a este respecto.

```

ZUscXC7aOGUUi1GFFHaf
+ Iniciar colección
+ Añadir campo
author: "Nombre"
cover: "https://firebasestorage.googleapis.com/v0/b/fiktin-
cac14.appspot.com/o/pruebas%20javi%2Fbooks%2FZUscXC7aOGUUi1GFFH
alt=media&token=420afe3a-4b5b-44d3-b5ff-6090b32e1174"
customBoxes
  0
    defaultValue: "18"
    name: "Edad"
    type: "number"
    value: ""
  description: ""
  published: false
sections
  0 "hq4cDboZEiOEGh8XXJw"
tags
  title: "Un libro"
  user_id: "w8eQL8LGzs6EleoYpVII"

```

Figura 11. Documento de colección de libros.

Los campos que conforman los libros son:

- *title*: Corresponde al título del libro. Es una clave única para todos los usuarios. Hay que tener en cuenta que el nombre es sensible a las mayúsculas y minúsculas.
- *author*: Corresponde al autor del libro. Admite dos posibles valores: *Nombre* o *Nick*, representando al nombre o al nick del usuario, respectivamente.
- *description*: Corresponde a la descripción del libro. No es obligatorio rellenar este campo en la aplicación tal y como se puede observar en la figura 11, donde la descripción se encuentra vacía.
- *published*: Indica si el libro se encuentra o no publicado, para que otros usuarios puedan verlo o no. Es un booleano a *true* si el libro está publicado y a *false* si no.
- *tags*: Son las etiquetas del libro. Es un campo multivalor con forma de un array de Strings, siendo cada entrada una etiqueta distinta. No puede haber dos etiquetas iguales. Puede estar vacío y no tiene límite de tamaño.
- *cover*: Este campo representa la dirección a la imagen de la portada del libro, ya que dicha imagen se encuentra en el almacenamiento de Firebase.
- *custom_boxes*: En este campo se guardan los valores del gadget de casilla personalizada en modo escritura. Es un atributo multivalor en forma de array de objetos. Puede estar vacío y no tiene límite de tamaño. Cuenta con sus propios campos, que son:
 - *name*: Corresponde al nombre de la casilla personalizada. El nombre es único para cada libro. Se usa para identificar las casillas personalizadas.
 - *type*: El tipo de datos de la casilla personalizada. Puede ser tanto *text*, que representa un texto, como *number*, que representa un número. Se usa para garantizar que los datos introducidos por el lector sólo pueden ser del tipo definido.
 - *value*: El valor local que toma la casilla personalizada en la lectura.
 - *defaultValue*: El valor por defecto que debe tomar la casilla personalizada si el lector no introduce ningún dato. Este valor debe ser del tipo definido en el campo *type*.
- *sections*: Es un campo multivalor en forma de array de Strings donde cada entrada corresponde a la ID única de una sección. Siempre tiene como mínimo una entrada y carece de límite máximo. No puede haber dos IDs repetidas.
- *user_id*: Corresponde al ID único de cada usuario, mencionado en la colección *users*. Es necesario para identificar al autor del libro.
- *id*: Clave primaria generada de manera automática por Firebase.

5.2.4. SECTIONS

La colección *sections* (figura 12) es la encargada de guardar el contenido de los libros. Todo libro siempre debe tener como mínimo una sección, sin límite máximo. Los gadgets de cada sección contienen la información que lo hace funcionar, gestionando cada uno un tipo de información distinta.

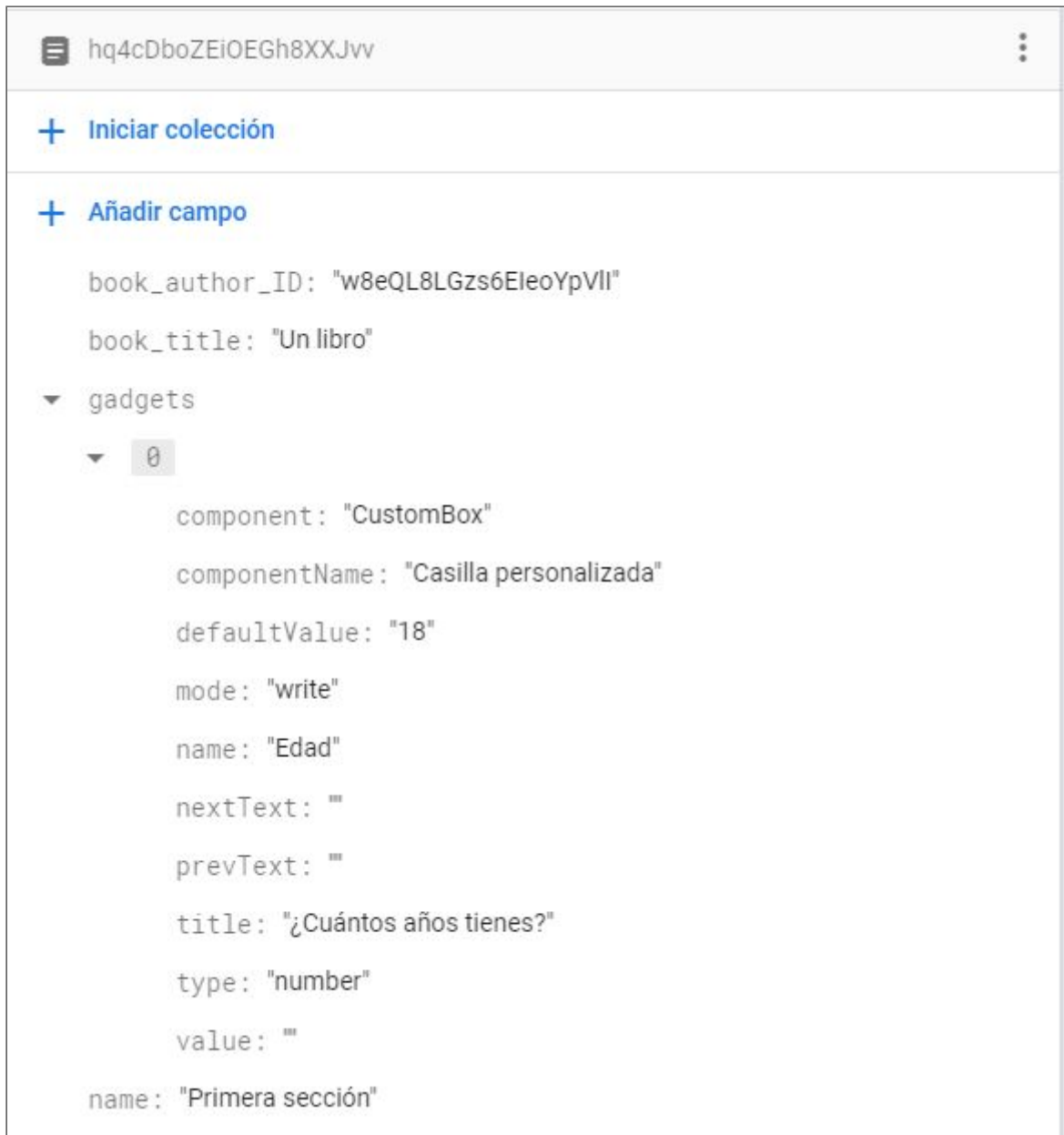


Figura 12. Documento de colección de secciones.

Se explica cada campo de la colección *sections* así como cada campo de cada gadget.

- *name*: Corresponde al nombre de la sección. No puede ser vacío o sólo contener espacios.

- *book_title*: Es el título del libro. Hace referencia al campo *title* de la colección *books*.
- *book_author_id*: Corresponde a la ID del autor del libro. Hace referencia a la clave primaria de la colección *users*.
- *id*: Clave primaria generada de manera automática por Firebase.
- *gadgets*: Corresponde a los datos propios de cada gadget. Se trata de un campo multivalor en forma de array de objetos. Puede estar vacío y no tiene límite máximo. Cada entrada de este campo corresponde a un gadget en la sección del libro. Como hay múltiples tipos de gadgets, existen distintos tipos de datos para cada uno. No obstante, existen dos campos compartidos entre todos los gadgets, que son:
 - *component*: Indica el tipo de componente. Es un identificador único para cada tipo de gadget.
 - *componentName*: Es el nombre en español del componente del gadget. Cada tipo de gadget tiene un único nombre.

Cada tipo de gadget solo tiene aquellos campos que va a usar. Se procede a explicar cuáles son los campos de cada uno, diferenciándolos por su componente:

```
component: "Normal"
componentName: "Texto normal"
htmlText: "Normal, <b>negrita</b>, <i>cursiva</i>, <u>subrayado,
</u>&nbsp;<strike>tachado</strike>&nbsp;<b><i><u>
<strike>todos juntos</strike></u></i></b>"
```

Figura 13. Gadget de texto normal.

- *Normal*: Corresponde al gadget de texto normal (figura 13). Cuenta con los campos:
 - *htmlText*: Se trata del texto escrito por el usuario en formato HTML. Puede ser un campo vacío si el usuario deja el gadget sin editar.

```
component: "Header1"
componentName: "Título"
htmlText: "<h1>Título 1</h1>"
plainText: "Título 1"
```

Figura 14. Gadget de título.

- *Header1*, *Header2*, *Header3*: Corresponde a los tres tipos de títulos presentes en la aplicación (figura 14). Cuentan con los campos:

- *htmlText*: Se trata del campo *plainText* escrito por el usuario adaptado a HTML.
- *plainText*: Es el texto plano, sin etiquetas HTML, del título. Se usa para cuando se quiere cambiar de tipo de título. Puede ser un campo vacío si el usuario deja el gadget sin editar.

```

component: "Picture"
componentName: "Imagen"
htmlText: "<img src='https://firebasestorage.googleapis.com/v0/b/fiktin-
cac14.appspot.com/o/pruebas%20javi%2Fbooks%2F0lWie7jO75Gbie71aEQs%
alt=media&token=486e8917-75d5-4b10-90ed-767342566743' style='margin-
auto;padding-top: 13px; padding-bottom: 13px;'"
url: "https://firebasestorage.googleapis.com/v0/b/fiktin-
cac14.appspot.com/o/pruebas%20javi%2Fbooks%2F0lWie7jO75Gbie71aEQs%
alt=media&token=486e8917-75d5-4b10-90ed-767342566743"

```

Figura 15. Gadget de imagen.

- *Picture*: Gadget usado para mostrar imágenes (figura 15). Cuenta con los campos:
 - *htmlText*: Se trata del texto generado automáticamente por la aplicación en formato HTML. Contiene el enlace a la imagen.
 - *url*: Contiene la dirección al almacenamiento de Firebase donde se encuentra almacenada la imagen.

Ambos pueden ser campos vacíos si el usuario opta por no añadir ninguna imagen.

```

autoplay: "true"
component: "Video"
componentName: "Video"
htmlText: "<video width='460' height='300' style='padding-top: 13px; padding-
src='https://firebasestorage.googleapis.com/v0/b/fiktin-
cac14.appspot.com/o/pruebas%20javi%2Fbooks%2F0lWie7jO75Gbie71aEQs%
alt=media&token=d9a5a12e-28a4-4422-b609-bf874779e5c8'>No se
loop: "true"
url: "https://firebasestorage.googleapis.com/v0/b/fiktin-
cac14.appspot.com/o/pruebas%20javi%2Fbooks%2F0lWie7jO75Gbie71aEQs%
alt=media&token=d9a5a12e-28a4-4422-b609-bf874779e5c8"

```

Figura 16. Gadget de vídeo.

- *Video*: Gadget usado para mostrar vídeos (figura 16). Cuenta con los campos:
 - *htmlText*: Se trata del texto generado automáticamente por la aplicación en formato HTML. Contiene el enlace al vídeo.
 - *url*: Contiene la dirección al almacenamiento de Firebase donde se encuentra almacenado el vídeo.
 - *autoplay*: Booleano que indica si el vídeo se reproduce automáticamente en la lectura o si hay que reproducirlo manualmente. *True* indica que sí se reproduce automáticamente; *false* que no.
 - *loop*: Booleano que indica si el vídeo se reproduce en modo bucle o si solo se reproduce una sola vez. *True* indica que sí se reproduce en bucle; *false* que no.

htmlText y *url* pueden estar vacíos si el usuario opta por no añadir ningún vídeo.

```

component: "ExpandableText"
componentName: "Texto expandible"
expandedText: "Sí, se expande"
mainText: "¿Si pulso se expande?"

```

Figura 17. Gadget de texto expandible.

- *ExpandableText*: El gadget de texto expandible se usa para mostrar texto oculto (figura 17). Cuenta con los campos:
 - *mainText*: El texto plano que se muestra en la lectura.
 - *expandedText*: El texto plano que se muestra cuando se pulsa sobre el botón de expandir en la aplicación.

Ambos campos pueden estar vacíos si el usuario opta por no escribir nada.

```

component: "PopupText"
componentName: "Texto emergente"
htmlText: "<img src='https://firebasestorage.googleapis.com/v0/b/fiktin-
cac14.appspot.com/o/pruebas%20javi%2Fbooks%2F0lWie7jO75Gbie71aE
alt=media&token=d9d6eb37-cf10-4d39-a694-11a3fcb71512' style='min-h
padding-bottom: 13px;'>"
mainText: "Si pulso se abre un modal"
picture: "https://firebasestorage.googleapis.com/v0/b/fiktin-
cac14.appspot.com/o/pruebas%20javi%2Fbooks%2F0lWie7jO75Gbie71aE
alt=media&token=d9d6eb37-cf10-4d39-a694-11a3fcb71512"
popupText: "Aquí está"

```

Figura 18. Gadget de texto emergente.

- *PopupText*: Corresponde al gadget de texto emergente (figura 18). Cuenta con los campos:

- *htmlText*: Se trata del texto generado automáticamente por la aplicación en formato HTML. Contiene el enlace a la imagen.
- *mainText*: El texto plano que se muestra en la lectura.
- *popupText*: El texto plano que se muestra dentro de un modal cuando el usuario pulsa sobre el botón de abrir en la aplicación.
- *picture*: Contiene la dirección al almacenamiento de Firebase donde se encuentra almacenada la imagen.

Pueden estar vacíos cualquiera de los campos anteriormente descritos si el usuario opta por no rellenarlos.

```

component: "Hyperlink"
componentName: "Hipervínculo"
htmlText: "<a href='https://github.com/avadop/Fiktin'
target='_blank'>Fiktin en la web</a>"
hyperlinkText: "https://github.com/avadop/Fiktin"
mainText: "Fiktin en la web"

```

Figura 19. Gadget de hipervínculo.

- *Hyperlink*: El gadget de hipervínculo (figura 19). Cuenta con los campos:

- *htmlText*: Se trata del texto generado automáticamente por la aplicación en formato HTML que contiene el enlace al hipervínculo.
- *hyperlinkText*: El enlace al que se quiere acceder.
- *mainText*: El texto plano que acompaña al enlace y que es visible al usuario.

Tanto *hyperlinkText* como *mainText* pueden estar vacíos si el usuario opta por no rellenarlos. *htmlText* nunca está vacío.

```
component: "Spoiler"
componentName: "Spoiler"
htmlText: "<span>Al final del libro, todos salen ganando</span>"
plainText: "Al final del libro, todos salen ganando"
```

Figura 20. Gadget de spoiler.

- *Spoiler*: Corresponde al gadget de spoiler (figura 20). Cuenta con los campos:
 - *htmlText*: Se trata del campo *plainText* escrito por el usuario adaptado a HTML.
 - *plainText*: Es el texto plano, sin etiquetas HTML, del spoiler. Se usa para generar a partir de él el campo *htmlText*. Puede estar vacío.

```
component: "ChangeSection"
componentName: "Cambio de sección"
htmlText: "<span>Vamos al siguiente capítulo:</span>"
next: "4WDt8jTk0XKxZNIktP0T"
plainText: "Vamos al siguiente capítulo:"
```

Figura 21. Gadget de cambio de sección.

- *ChangeSection*: El gadget de cambio de sección (figura 21). Cuenta con los campos:
 - *htmlText*: Se trata del campo *plainText* escrito por el usuario adaptado a HTML.
 - *plainText*: Es el texto plano, sin etiquetas HTML, que acompaña al cambio de sección. Se usa para generar a partir de él el campo *htmlText*. Puede estar vacío.

- *next*: Corresponde a la ID única de la sección a la que se quiere saltar.

```
component: "RepeatSection"  
componentName: "Repetición de sección"  
htmlText: "<span>Repitamos este día</span>"  
plainText: "Repitamos este día"
```

Figura 22. Gadget de repetición de sección.

- *RepeatSection*: Corresponde al gadget de repetición de sección (figura 22). Cuenta con los campos:
 - *htmlText*: Se trata del campo *plainText* escrito por el usuario adaptado a HTML.
 - *plainText*: Es el texto plano, sin etiquetas HTML, que acompaña a la repetición de sección. Se usa para generar a partir de él el campo *htmlText*. Puede estar vacío.

```
▼ choices  
  ▼ 0  
    action: "4WDt8jTk0XKxZNIktP0T"  
    choice: "Section"  
    htmlText: "<span>La izquierda es mejor</span>"  
    plainText: "La izquierda es mejor"  
  ▼ 1  
    action: "2vwN2BMqpuFL90KL6PWX"  
    choice: "Section"  
    htmlText: "<span>La derecha es mejor</span>"  
    plainText: "La derecha es mejor"  
component: "DecisionMaking"  
componentName: "Toma de decisiones"  
numberOfOptions: 2
```

Figura 23. Gadget de toma de decisiones.

- *DecisionMaking*: Corresponde al gadget de toma de decisiones (figura 23). Cuenta con los campos:
 - *numberOfOptions*: La cantidad de opciones que se le van a dar al usuario. Es un número entre dos y diez.
 - *choices*: Es un array con cada una de las elecciones que se le da al usuario. Hay tantas entradas como la cantidad del campo *numberOfOptions*. Cada entrada del array cuenta con cuatro campos:
 - *action*: La sección a la que se quiere saltar al pulsar sobre la opción.
 - *choice*: El tipo de cambio que se quiere realizar. *Section* corresponde a un cambio de sección. Actualmente no hay otros posibles valores.
 - *htmlText*: Se trata del campo *plainText* de cada entrada del array escrito por el usuario y adaptado a HTML.
 - Es el texto plano, sin etiquetas HTML, que acompaña a cada entrada del array. Se usa para generar a partir de él el campo *htmlText*. Puede estar vacío.

```

answerText: "mario"
changeSectionWhenWrong: true
component: "Riddle"
componentName: "Adivinanza"
numberOfTries: "1"
riddleText: "¿Quién es el ganador del concurso?"
rightSection: "4WDt8JTk0XKxZNIktPOT"
wrongSection: "2vwN2BMqpuFL90KL6PWX"

```

Figura 24. Gadget de adivinanza.

- *Riddle*: El gadget de adivinanza (figura 24). Cuenta con los campos:
 - *riddleText*: El texto plano con la pregunta de la adivinanza. Puede estar vacío si el usuario opta por no hacer una pregunta.
 - *answerText*: El texto plano con la respuesta de la adivinanza. Puede estar vacío si el usuario opta por no dar una respuesta.
 - *numberOfTries*: El número de intentos que tiene el usuario para resolver la adivinanza. Es un número entre uno y diez.
 - *changeSectionWhenWrong*: Booleano que indica si cuando se falla la adivinanza y el usuario se queda sin intentos, se pasa a

otra sección o se continúa leyendo. *True* corresponde a un cambio de sección, *false* a quedarse leyendo la actual.

- *rightSection*: Contiene la ID de la sección a la que se salta si se acierta la adivinanza.
- *wrongSection*: Contiene la ID de la sección a la que se salta si se falla la adivinanza. Si *changeSectionWhenWrong* tiene el valor falso, este campo puede estar vacío.

```
changeSectionWhenWrong: true
component: "Sequence"
componentName: "Secuencia"
numberOfTries: 1
rightSection: "4WDt8jTk0XKxZNIktP0T"
sequence
  0
    text: "a"
  1
    text: "b"
solution
  0
    text: "c"
wrongSection: "2vwN2BMqpuFL90KL6PWX"
```

Figura 25. Gadget de completar la secuencia.

- *Sequence*: Corresponde al gadget de completar la secuencia (figura 25). Contiene los campos:
 - *sequence*: Campo multivalor con los datos de la secuencia. Es un array donde cada una de sus entradas corresponde a un dato de los ofrecidos por el autor.
 - *solution*: Campo multivalor con los datos de la solución. Es un array donde cada una de sus entradas corresponde a una parte de la solución ofrecida por el autor.
 - *numberOfTries*: El número de intentos que se le da al lector para completar la secuencia. Es un número entre 1 y 10.
 - *changeSectionWhenWrong*: Booleano que indica si cuando se completa incorrectamente la secuencia y el usuario se queda sin

intentos, se pasa a otra sección o se continúa leyendo. *True* corresponde a un cambio de sección, *false* a quedarse leyendo la actual.

- *rightSection*: Contiene la ID de la sección a la que se salta si se completa correctamente la secuencia.
- *wrongSection*: Contiene la ID de la sección a la que se salta si se completa incorrectamente la secuencia. Si *changeSectionWhenWrong* tiene el valor falso, este campo puede estar vacío.

```
changeSectionWhenWrong: true
component: "MemoryCards"
componentName: "Tarjetas de memoria"
▼ customColors
▼ customWords
  0 "A"
  1 "E"
  2 "B"
  3 "C"
customized: true
maxNumberOfMoves: 8
numberOfPairs: 2
sectionNoMoreMoves: "2vwN2BMqpuFL90KL6PWX"
sectionSolved: "4WDt8jTk0XKxZNIktPOT"
typeChosen: "words"
```

Figura 26. Gadget de tarjetas de memoria.

- *MemoryCards*: Gadget de las tarjetas de memoria (figura 26). Contiene los campos:
 - *customColors*: Campo multivalor que contiene las parejas en caso de haberse seleccionado la opción de tarjetas con color. Puede ser vacío y su límite superior es de 24 entradas.
 - *customWords*: Campo multivalor que contiene las parejas en caso de haberse seleccionado la opción de tarjetas con texto. Puede ser vacío y su límite superior es de 24 entradas.

- *customized*: Booleano que indica si las tarjetas van a estar personalizadas o van a ser por defecto. *True* indica que sí van a ser personalizadas, por lo que se usará o el campo *customColors* o el campo *customWords*. *False* indica que no van a ser personalizadas, por lo que no se usará ninguno de esos dos campos.
- *typeChosen*: El tipo de tarjetas personalizadas que habrá. Puede contener dos valores, además de estar vacío. Los valores son: *words* y *color*, que indican si las tarjetas personalizadas están constituidas por palabras o colores, respectivamente.
- *numberOfPairs*: La cantidad de parejas a encontrar. El mínimo es dos, equivalente a cuatro tarjetas, y el máximo es doce, equivalente a veinticuatro tarjetas.
- *maxNumberOfMoves*: El número de movimientos máximos que se le dan al lector para encontrar todas las parejas.
- *changeSectionWhenWrong*: Booleano que indica si cuando no se encuentran las parejas y el usuario se queda sin intentos, se pasa a otra sección o se continúa leyendo. *True* corresponde a un cambio de sección, *false* a quedarse leyendo la actual.
- *sectionNoMoreMoves*: Contiene la ID de la sección a la que se salta si no se encuentran todas las parejas. Si *changeSectionWhenWrong* tiene el valor falso, este campo puede estar vacío.
- *sectionSolved*: Contiene la ID de la sección a la que se salta si se encuentran todas las parejas.

```

component: "RandomNumber"
componentName: "Número aleatorio"
▼ conditions
  ▼ 0
    condition: ">"
    number: 5
    section: "2vwN2BMqpuFL90KL6PWX"
lowerLimit: 0
numberOfConditions: 1
upperLimit: 10

```

Figura 27. Gadget de número aleatorio.

- *RandonNumber*: Corresponde al gadget de número aleatorio (figura 27).
Contiene los campos:
 - *lowerLimit*: Número que indica el límite inferior para encontrar el número aleatorio. Está comprendido entre menos mil novecientos noventa y nueve y mil novecientos noventa y nueve. Siempre es menor o igual que *upperLimit*.
 - *upperLimit*: Número que indica el límite superior para encontrar el número aleatorio. Está comprendido entre menos mil novecientos noventa y nueve y mil novecientos noventa y nueve. Siempre es mayor o igual que *lowerLimit*.
 - *conditions*: Campo multivalor en forma de array que indica las condiciones a cumplir para cambiar de sección según el número obtenido. Cada entrada corresponde a una condición. Puede estar vacío y contiene tantos elementos como *numberOfConditions*.
Contiene los campos:
 - *condition*: Condición, en forma de String, que se debe cumplir. Puede tomar cinco valores: >, >=, <, <=, ==.
 - *number*: El número sobre el que se aplica la condición.
 - *section*: La sección a la que se salta en caso de cumplirse la condición con el número.
 - *numberOfConditions*: La cantidad de condiciones que puede haber en el array de *conditions*. Es un número entre cero y cuatro.

```
▼ answers
  ▼ 0
    answer: "Izquierda"
  answersNumber: 1
  changeSectionWhenWrong: true
▼ clues
  ▼ 0
    clue: "Izquierda"
  ▼ 1
    clue: "Derecha"
  cluesNumber: 2
  component: "CompleteClues"
  componentName: "Pistas"
  onGuess: "4WDt8jTk0XKxZNIktP0T"
  onWrong: "2vwN2BMqpuFL90KL6PWX"
```

Figura 28. Gadget de completar las pistas.

- *CompleteClues*: El gadget de completar las pistas (figura 28). Contiene los campos:
 - *clues*: Campo multivalor que contiene las pistas definidas por el autor. Contiene tantas entradas como *cluesNumber*.
 - *answers*: Campo multivalor que contiene las respuestas definidas por el autor. Contiene tantas entradas como *answersNumber*.
 - *cluesNumber*: Entero que indica la cantidad de pistas. Es un número situado entre uno y cinco.
 - *answersNumber*: Entero que indica la cantidad de soluciones. Es un número situado entre uno y cinco.
 - *changeSectionWhenWrong*: Booleano que indica si cuando se falla al encontrar las pistas, se pasa a otra sección o se continúa leyendo. *True* corresponde a un cambio de sección, *false* a quedarse leyendo la actual.
 - *onGuess*: Contiene la ID de la sección a la que se salta si se resuelven las pistas.

- *onWrong*: Contiene la ID de la sección a la que se salta si no se resuelven todas las pistas. Si *changeSectionWhenWrong* tiene el valor falso, este campo puede estar vacío.

```
component: "CustomBox"
componentName: "Casilla personalizada"
defaultValue: "Marcos"
mode: "write"
name: "Nombre"
nextText: ""
prevText: ""
title: "¿Cuál es tu nombre?"
type: "string"
value: ""
```

Figura 29. Gadget de casilla personalizada.

- *CustomBox*: Corresponde al gadget de casilla personalizada (figura 29). Contiene los campos:
 - *mode*: El modo de la casilla personalizada. Puede tomar tres valores: *read*, *write* y *modifyWrite*, correspondiendo cada uno a un tipo de funcionalidad de las casillas personalizadas.
 - *name*: Nombre de la casilla personalizada. El nombre es único para todo el libro si la casilla se encuentra en modo escritura. Si la casilla se encuentra en modo lectura o de modificación de escritura, el nombre corresponde a uno de los nombres del array *custom_boxes* de la colección *books*.
 - *type*: El tipo de datos de la casilla personalizada. Puede ser tanto texto como número, representados en la base de datos por *string* o *number*, respectivamente.
 - *title*: El título de la casilla personalizada. Se usa cuando la casilla se encuentra en modo escritura. Puede estar vacío.
 - *value*: El valor de la casilla personalizada.
 - *defaultValue*: El valor por defecto de la casilla personalizada. Debe corresponder al tipo de datos del campo *type*. Puede estar vacío.
 - *prevText*: Texto anterior que acompaña a la casilla personalizada si esta se encuentra en modo lectura. Puede estar vacío.

- *nextText*: Texto posterior que acompaña a la casilla personalizada si esta se encuentra en modo lectura. Puede estar vacío.

6. CAPÍTULO VI: ARQUITECTURA DE LA APLICACIÓN

En este capítulo se describe la arquitectura de la aplicación.

6.1. ARQUITECTURA CLIENTE-SERVIDOR

La aplicación emplea un modelo cliente-servidor [22] en el que se distinguen dos entidades: clientes y servidores. El cliente lanza peticiones de servicio, y el servidor recoge esas peticiones, las procesa y devuelve al cliente una respuesta con los datos solicitados. Los clientes se pueden conectar a varios servidores y normalmente interactúan con el servidor a través de una interfaz de usuario mostrada a través de un navegador, mientras que los servidores pueden trabajar con un gran número de clientes de manera simultánea.

En la figura 30 se muestra la arquitectura cliente-servidor de la aplicación. El navegador ejecuta el código Vue de la aplicación -es decir, el código HTML, CSS y JavaScript generado tras la compilación de los ficheros Vue-. Esta parte constituye la parte del cliente. Para la parte del servidor, se ha empleado una conexión directa de Vue con Firebase. En la aplicación, esta conexión directa se realiza usando la biblioteca Vuefire [23], destinada a procesar y facilitar la conexión de Vue con Firebase [13]. Vuefire, por tanto, manda peticiones y espera respuestas que, posteriormente, son devueltas al lado del cliente para ser usadas.



Figura 30. Esquema del modelo cliente-servidor usado por Vue.

6.2. EL MODELO DE VUE

Vue implementa un patrón modelo-vista-modelo de vista (model-view-viewmodel) [24]. En ese patrón se desacopla lo máximo posible la interfaz de usuario de la lógica de la aplicación. Esto significa que cada instancia de Vue [25] es un objeto que conecta el modelo con la vista a través de enlaces de datos bidireccionales, lo que permite que la vista modifique el modelo y el modelo modifique la vista, tal y como se puede observar en la

figura 31. Esto último también se realiza mediante filtros, que son funciones usadas para procesar valores del modelo antes de actualizar la vista -como son las “expresiones mustache” y “expresiones v-bind”-, y directivas, que son atributos HTML que le dicen a Vue qué hacer con un elemento del DOM.

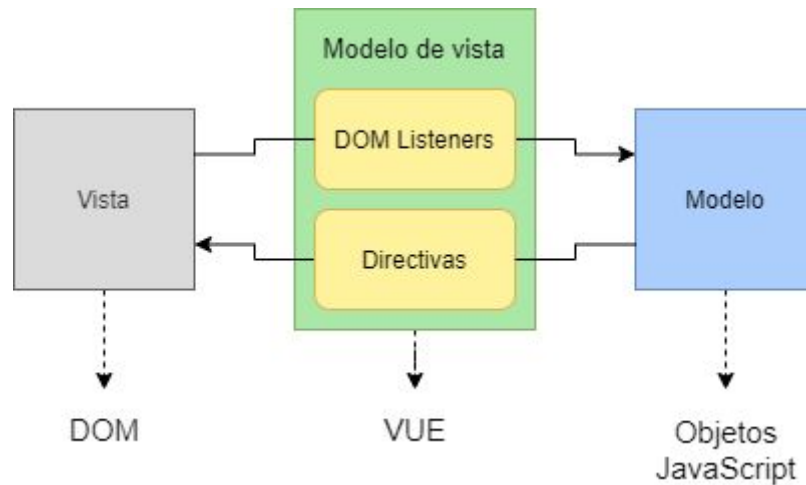


Figura 31. Esquema del modelo de vista usado por Vue [26].

Por otra parte, Vue se encuentra organizado internamente por componentes. Un componente es una instancia reutilizable de la raíz Vue con un nombre, es decir, cada componente sigue el modelo de vista de la figura 31, puede ser usado en distintas partes de la misma manera y puede usar todas las propiedades extendidas de la raíz Vue [25]. Cabe destacar que en Vue, la mayoría de ficheros son componentes, y todos se encuentran relacionados entre sí siguiendo una estructura en forma de árbol, como aparece reflejado en la figura 32, donde se puede observar la reusabilidad y la jerarquía de los componentes. Además se pueden comunicar entre sí por varios métodos tales como las propiedades -para pasar datos de padres a hijos-, la emisión de eventos -para pasar datos de hijos a padres- o mediante el “Store”, que hace de contenedor de variables globales.

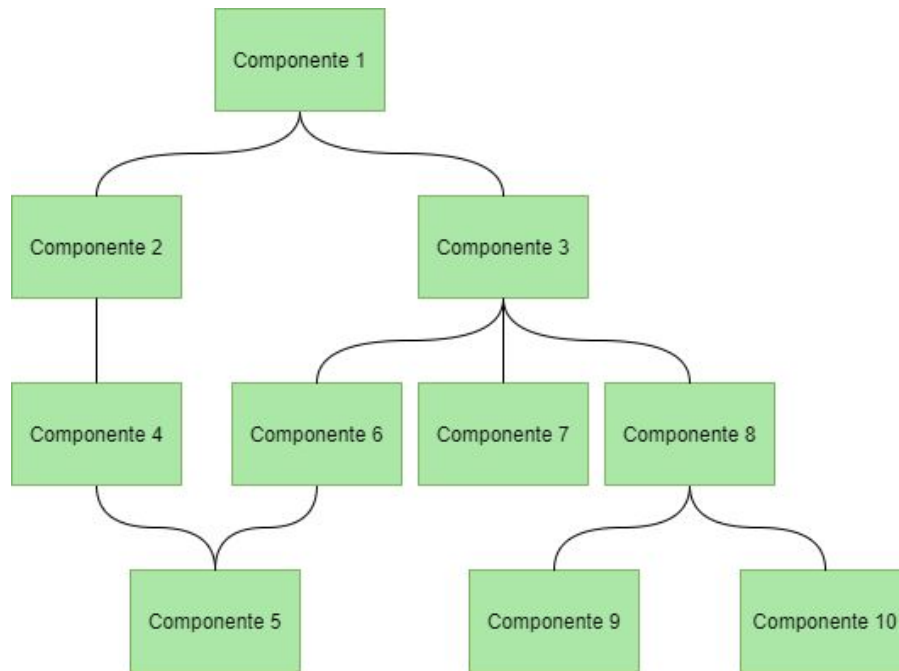


Figura 32. Esquema de ejemplo de la organización por componentes de Vue.

7. CAPÍTULO VII: DISEÑO DE LA APLICACIÓN

En este capítulo se describe el diseño y la implementación de la aplicación desarrollada.

7.1. DISEÑO GRÁFICO

Esta aplicación está orientada a usuarios con un nivel medio de conocimiento en el uso de aplicaciones web. Para conseguir este nivel de accesibilidad se ha simplificado la interfaz homogeneizando el diseño en las distintas vistas y mostrando las opciones de interacción de una forma intuitiva.

Se han seguido como ejemplo de diseño otras páginas web de venta de libros como La casa del libro [27] y Fnac [28]. Para mejorar el aspecto estético se ha usado la librería de BootstrapVue [9] que proporciona estilo a los componentes y también tiene un paquete de iconos.

Para construir la interfaz de la página se ha utilizado el framework de Vue. Las principales razones para utilizar Vue son:

- La simplicidad que aporta al desacoplar la lógica de la interfaz con su patrón “modelo-vista-modelo de vista”.
- La facilidad en la programación gracias al uso de componentes y directivas.

El diseño gráfico de la aplicación se caracteriza por:

- El predominio del color azul turquesa. Aparece en la barra de navegación y en los botones de creación de libro y biblioteca.
- El uso de la tipografía Arial. Una tipografía simple que facilita la lectura.
- El icono que aparece en la pestaña del navegador junto al nombre ‘fiktin’ (figura 33) y un logo (figura 34) que está a la derecha de la barra de navegación y hace las veces de botón para ir al inicio.



Figura 33. Icono de la aplicación.



Figura 34. Logo de la aplicación.

7.2. FUNCIONALIDAD DE LA APLICACIÓN

Dentro de la aplicación se encuentran las funcionalidades representadas en el diagrama de la figura 35 que se describen a continuación.

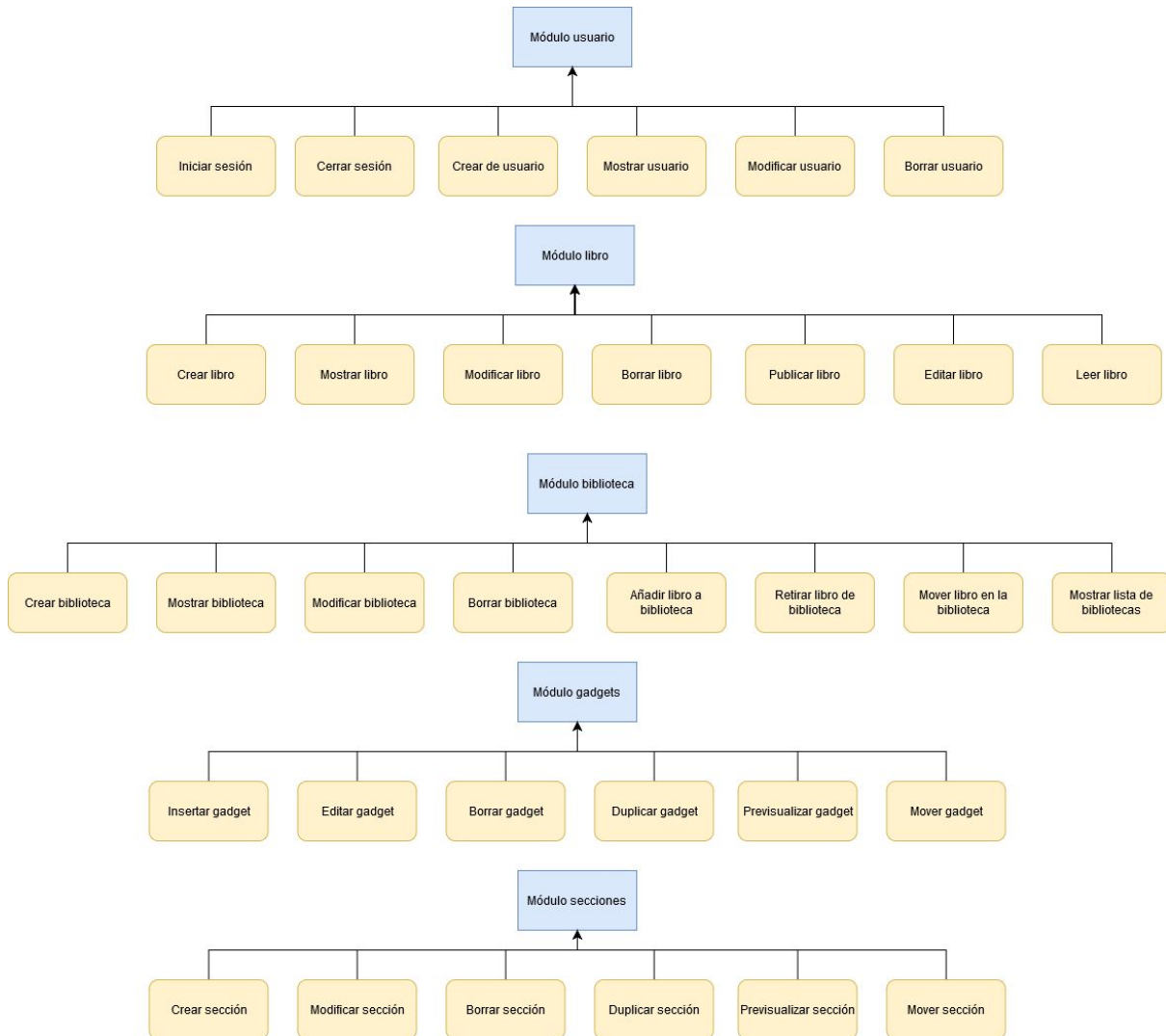


Figura 35. Diagrama de funcionalidades.

7.2.1. USUARIOS

En la aplicación hay algunas funcionalidades que están disponibles sin necesidad de tener un usuario, tal como ver la lista de libros publicados. Sin embargo, hay otras, como la creación y edición de libros, que requieren que se inicie sesión con un usuario para acceder a ellas.

Todos los usuarios tienen acceso a las mismas funcionalidades sin haber distintos tipos de usuario.

A continuación, se explican las funcionalidades relativas a los usuarios.

7.2.1.1. INICIAR SESIÓN

La opción de inicio de sesión es accesible desde la vista de 'Inicio' de la aplicación, pulsando sobre 'Iniciar sesión' o sobre los otros enlaces que redirigen a otras vistas de la aplicación solo accesibles para usuario con sesión iniciada.

Consiste en un simple formulario donde el usuario introduce su nombre de usuario y contraseña como se puede observar en la figura 36.

La autenticación se realiza con una consulta a la BBDD y comparando los datos allí almacenados con los introducidos por el usuario. En la figura 37 se puede observar el código correspondiente al proceso descrito.

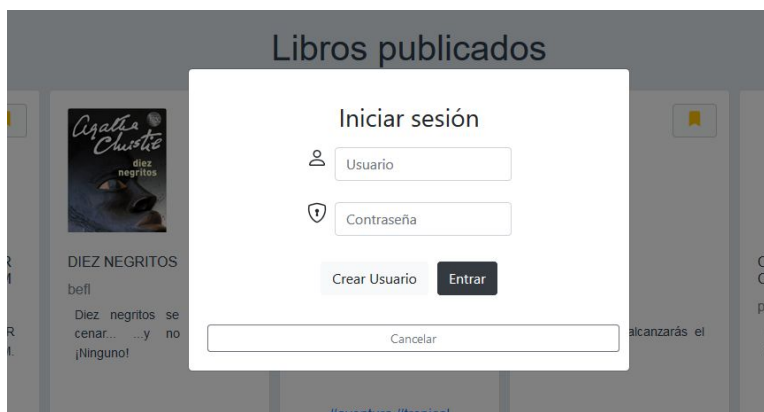


Figura 36. Inicio de sesión.

```
logIn: async function () {
  await this.$bind('user', userCollection.where('nick_to_search', '==', this.nick.toLowerCase()).limit(1)).then(docs => {})
  if (this.user.length === 0) {
    window.alert('Usuario no existe')
  } else {
    if (this.user[0].password === this.password) {
      store.commit('logIn', this.user[0].id)
      store.commit('nickChange', this.user[0].nick)
      this.nick = ''
      this.password = ''
      this.$emit('loggedIn')
      this.$router.push(this.url)
    } else {
      window.alert('Contraseña incorrecta')
    }
  }
},
```

Figura 37. Código de comprobación de inicio de sesión.

Si la autenticación ha sido exitosa se almacenan en el store de Vuex, un contenedor en el que se guardan los estados de la aplicación, los valores correspondientes a la sesión actual (figura 38).

```
logIn (state, id) {
  state.userID = id
  state.loggedIn = true
},
```

Figura 38. Almacenamiento de los datos de la sesión actual.

7.2.1.2. CERRAR SESIÓN

Para realizar el proceso de cierre de sesión es necesaria una confirmación por parte del usuario como la que se muestra en la figura 39.

Una vez confirmado se borran los datos de la sesión (véase figura 40)



Figura 39. Confirmación de cierre de sesión.

```
logout (state) {  
  state.userID = ''  
  state.loggedIn = false  
},
```

Figura 40. Limpieza de los datos de sesión.

7.2.1.3. CREAR USUARIO

La funcionalidad de crear usuario es accesible antes de iniciar sesión desde la vista de 'Inicio' y desde el modal de 'Iniciar sesión'. El formulario de crear usuario es el mostrado en la figura 41 y contiene los campos de: 'Usuario', al que también se refiere internamente como nick; nombre, email, contraseña e imagen de perfil.

El formulario tiene el título "Registro" y los siguientes campos:

- Un campo de texto con un ícono de persona a la izquierda y el placeholder "Usuario".
- Un campo de texto con un ícono de lista de personas a la izquierda y el placeholder "Nombre".
- Un campo de texto con un ícono de correo electrónico a la izquierda y el placeholder "Email".
- Un campo de texto con un ícono de escudo a la izquierda y el placeholder "Contraseña".
- Un campo de texto con un ícono de escudo a la izquierda y el placeholder "Confirmar contraseña".

Debajo de los campos, hay una sección "Imagen de perfil:" con un botón "Browse" y el texto "Selecciona una imagen o arrástrala".

Debajo de eso, hay un mensaje "No has subido ninguna imagen" y dos botones: "Iniciar sesión" (gris) y "Crear" (negro).

En la parte inferior del formulario, hay un botón "Cancelar" que abarca todo el ancho.

Figura 41. Crear usuario.

Al pulsar el botón de ‘Crear’ se realizan una serie de comprobaciones, en caso de satisfacerse se añade el nuevo usuario a la base de datos y, también se crea una nueva biblioteca ‘Historial’ asociada a este nuevo usuario (véase figura 42).

```
202   addUser: async function () {
203     if (this.correctNick) {
204       if (this.correctPassword) {
205         if (!this.exists) { // Si no existe user con el mismo nick, creamos usu
206           if (this.validEmail) {
207             if (this.samePasswords) {
208               var historial = this.newNick.toLowerCase() + '_historial'
209
210               if (this.picture === null) this.picture = defaultImageUser
211
212               let doc = await userCollection.add({
213                 nick: this.newNick,
214                 name: this.newName,
215                 email: this.newEmail,
216                 password: this.newPassword,
217                 profile_picture: this.picture,
218                 nick_to_search: this.newNick.toLowerCase()
219               })
220
221               librariesCollection.doc(historial).set({
222                 name: 'Historial',
223                 description: ('Aquí se guardaran los ultimos libros consultados de ' + this.newNick),
224                 privacy: 'private',
225                 nick: this.newNick,
226                 array_keys: [],
227                 user_id: doc.id
228               })
229
230               this.newNick = ''
231               this.newPassword = ''
232               this.newEmail = ''
233               this.newName = ''
234               this.newPassword2 = ''
235               this.picture = ''
236               this.$emit('switchCreate')
237             } else {
238               window.alert('Contraseñas diferentes')
```

Figura 42. Código de creación de usuario.

7.2.1.4. MOSTRAR USUARIO

Los usuarios pueden consultar la información de su perfil desde la vista de ‘user’ correspondiente a la figura 43. En esa ventana se ve el nombre del usuario, el nick, el correo y su imagen de perfil.

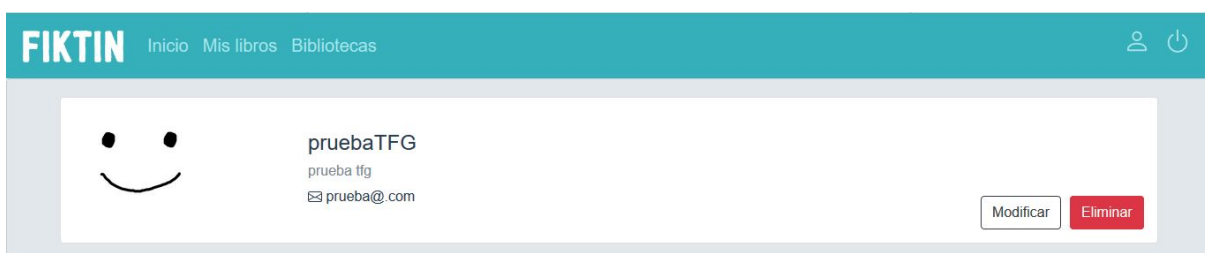



Figura 43. Mostrar usuario


7.2.1.5. MODIFICAR USUARIO

La funcionalidad de modificar usuario (figura 44) permite editar todos los datos del perfil menos el nick de usuario, ya que este se usa como clave única.

Cambia los datos que desees modificar

Imagen de perfil:  ✕

 prueba tfg

 prueba@.com



 

Figura 44. Modificar usuario.

7.2.1.6. ELIMINAR USUARIO

Es posible eliminar un usuario si este tiene la sesión iniciada. Para ello es necesario realizar una confirmación (figura 45) y a continuación se elimina sus datos de bibliotecas, libros y el propio usuario de la base de datos (figura 46).

Eliminar usuario

¿Está seguro que desea eliminar usuario?

Una vez realizado dicha operación no se podrá volver atrás

Cancelar

ELIMINAR

Figura 45. Eliminar usuario.

```

92 deleteUser: async function () {
93   await librariesCollection.where('user_id', '=', this.userKey).get().then(snapshot => {
94     snapshot.forEach(doc => {
95       librariesCollection.doc(doc.id).delete()
96     })
97   })
98   await booksCollection.where('user_id', '=', this.userKey)
99     .get().then(snapshot => {
100     snapshot.forEach(doc => {
101       booksCollection.doc(doc.id).delete()
102     })
103   })
104   userCollection.doc(this.userKey).delete()
105   this.$emit('deleteUser')
106   store.commit('logOut')
107   this.$router.push('/')
108 },

```

Figura 46. Código de eliminar usuario.

7.2.2. LIBROS

Los libros son los elementos que más valor aportan a la aplicación. Todos los usuarios pueden crear, modificar, eliminar, editar y leer libros.

7.2.2.1. CREAR O MODIFICAR LIBRO

Los usuarios con sesión iniciada tienen acceso a la funcionalidad de crear libro (figura 47) y, en caso de tener un libro ya creado, de modificar (figura 48) también. Ambas funcionalidades tienen un formulario similar, los campos son los mismos, salvo por el campo de la portada que solo está presente en el formulario de modificar.

Ambas opciones son accesibles desde la vista de ‘Mis libros’.

Crear libro

Título

Título del libro

Introduce al menos 3 caracteres

Descripción

Descripción

Autor

Nombre de usuario

Nickname

Etiquetas

Añade una etiqueta...

Publicar

Figura 47. Crear libro.

Modificar libro

Título

Las aventuras de Sauron

Autor

Nombre de usuario

Nickname

Portada

Selecciona una imagen o arrástrala aquí...

Eliminar

Descripción

Cuando parecía que toda la historia ya estaba contada resurge de sus cenizas el más temido villano.

Type a new tag and press enter

aventura x fantasía x Añade una etiqueta...

Publicar

Figura 48. Modificar libro.

Como se ve en el código de la figura 49, al crear un libro también se crea una sección que se añade al array de secciones del nuevo libro.

```

113     createButton: async function () {
114         var sections = []
115         sections.push(await sectionsCollection.add({
116             name: 'Primera sección',
117             book_author_ID: this.userID,
118             book_title: this.title,
119             gadgets: [{
120                 htmlText: '',
121                 component: 'Normal',
122                 componentName: 'Texto normal'
123             }]
124         }))
125         await booksCollection.add({
126             title: this.title,
127             author: this.author,
128             tags: this.tags,
129             description: this.description,
130             published: this.published,
131             user_id: this.userID,
132             sections: [sections[0].id],
133             customBoxes: [],
134             cover: defaultImageBook
135         })
136
137         this.title = ''
138         this.author = ''
139         this.tags = []
140         this.description = ''
141         this.published = 'not_published'
142         this.modalCreate = false
143         this.$emit('create')
144     },

```

Figura 49. Código de crear libro.

7.2.2.2. EDITAR LIBRO

En la vista de ‘editBook’ (figura 50) tenemos acceso a todas las funcionalidades necesarias para editar el libro, la gestión de los gadgets y las secciones, la previsualización y el guardado de los cambios.

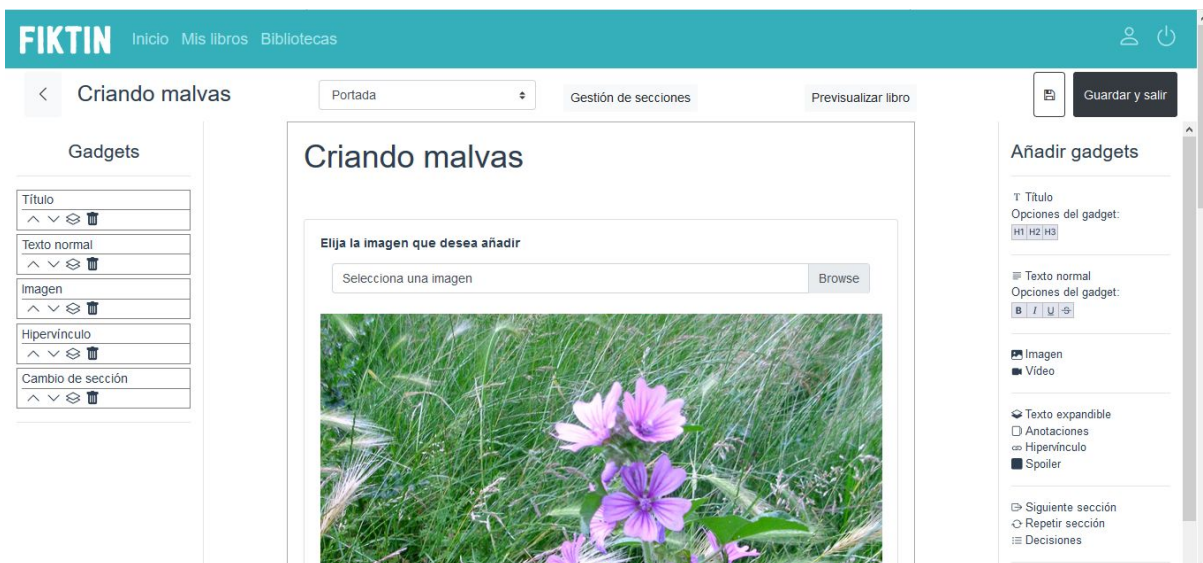


Figura 50. Lectura y previsualización del libro.

La vista de edición muestra la sección actual en una hoja que contiene los gadgets de la sección en su modo de edición. Al abrir la vista se carga el contenido de la sección y se guardan los gadgets en el array ‘data’ (figura 51).

```

refresh: async function (sectionID) {
  this.loading = true
  this.lastPress = -1
  this.sectionsData = []
  this.temporalCustomBoxes = []
  this.data = []
  store.commit('changeSection', sectionID)
  await booksCollection.doc(this.bookID).get().then(doc => {
    this.temporalCustomBoxes = doc.data().customBoxes
  })
  for (var i = 0; i < this.book.sections.length; ++i) {
    await sectionsCollection.doc(this.book.sections[i]).get().then(doc => {
      if (doc.exists) {
        this.sectionsData.push({ value: doc.id, text: doc.data().name })
      }
    })
  }
  await sectionsCollection.doc(sectionID).get().then(doc => {
    this.sectionName = doc.data().name
    this.sectionID = doc.id
    this.nextSectionID = doc.id
    this.data = this.data.concat(doc.data().gadgets)
  })
  this.loading = false
},

```

Figura 51. Carga del contenido de la sección.

El contenido de este array se plasma en la hoja de la vista añadiendo los componentes vue correspondientes, véase el código de la figura 52.

```

<div class="document" @keyup="checkStyles" @keydown.tab.prevent>
  <div class="editable" v-for="(text, index) in data" :key="index" @click="lastElementPressed(index), checkStyles()">
    <Normal v-if="text.component=='Normal'"
      :htmlTextAux="text.htmlText"
      :index="index"
      :boldUse="boldUse"
      :italicUse="italicUse"
      :underlineUse="underlineUse"
      :strikeThroughUse="strikeThroughUse"
      :lastPress="lastPress"
      @html="saveHTML"
      @delete="checkDelete"/>
    <Header1 v-if="text.component=='Header1'"
      :htmlTextAux="text.htmlText"
      :plainTextAux="text.plainText"
      :index="index"
      @html="savePlaneAndHTML"/>
  </div>
</div>

```

Figura 52. Código de mostrar el contenido de la sección.

Los botones de guardar llaman a una función que se encargan de actualizar la lista de gadgets de la sección y casillas personalizadas en la base de datos.

Las funcionalidades que corresponden a los módulos de gadgets y secciones se explican en los puntos 7.2.4 y 7.2.5 respectivamente.

7.2.2.3. LEER Y PREVISUALIZAR LIBRO

Las funcionalidades de leer y previsualizar libro corresponden, en realidad a la misma vista ‘readBook’ que se puede ver en la figura 53. Desde esta vista vemos el título del libro, un desplegable con las secciones y hoja con la sección actual.

< Piratillas

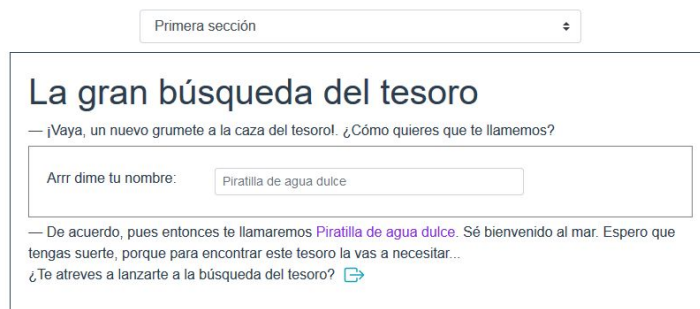


Figura 53. Lectura y previsualización del libro.

La forma de mostrar el contenido de la sección es la misma que la del apartado anterior. Al cargar la vista se descargan los datos de la sección de la base de datos, se guarda la lista de gadgets en un array local y usando la información de este array se cargan en la vista los componentes vue de los gadgets, pero en esta ocasión se cargan en su versión de lectura.

7.2.2.4. MOSTRAR LIBRO

Los libros son mostrados en la vista de ‘Mis libros’ (figura 54) y en el ‘Inicio’ (figura 55). Se muestran en tarjetas individuales con la imagen de la portada, el título, el nombre del autor, la descripción del libro y las etiquetas. Estas últimas se encuentran en la parte baja de la tarjeta.

Desde la vista de mis libros también es visible un icono de un ojo que indica si el libro se encuentra publicado o no.

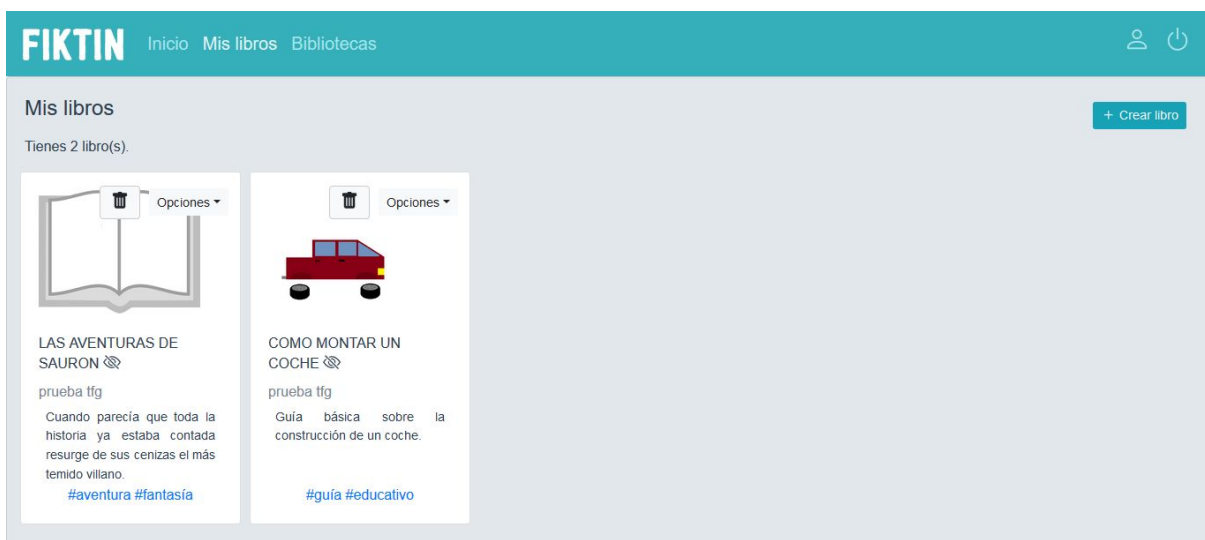


Figura 54. Mostrar libro - Mis libros.

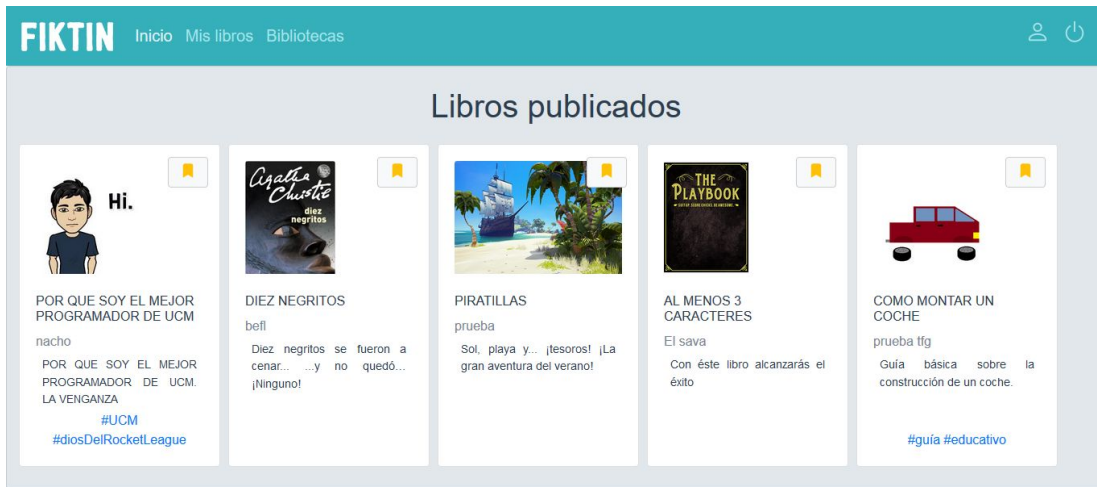


Figura 55. Mostrar libro - Inicio.

7.2.2.5. ELIMINAR LIBRO

Desde la vista de ‘Mis libros’ se tiene acceso a la opción de eliminar libro, que se encuentra en el icono de la papelera que aparece en la tarjeta de cada libro. El proceso de confirmación consiste en pulsar dos veces sobre el mismo botón (figuras 56 y 57) que cambia a color rojo para indicar al usuario la gravedad de su acción.

Tras la confirmación se elimina el libro de la base de datos, con lo que deja de estar disponible en la aplicación.



Figuras 56 y 57. Eliminar libro y confirmar eliminar.

7.2.3. BIBLIOTECAS

Las bibliotecas funcionan como colecciones de libros que el usuario puede crear, modificar y eliminar. Además de añadir, ordenar y retirar los libros como desee.

Sus funcionalidades se describen a continuación.

7.2.3.1. CREAR O MODIFICAR BIBLIOTECA

En la vista, la creación (figura 58) y modificación (59) de las bibliotecas tienen formularios con campos idénticos que permiten definir el nombre de la biblioteca, la privacidad y una breve descripción.

Figura 58. Crear biblioteca.

Figura 59. Modificar biblioteca.

7.2.3.2. MOSTRAR BIBLIOTECA

La visualización de una biblioteca muestra una lista de los libros asociada a esta. En esta lista cada tarjeta de libro contiene su portada, título y descripción, además de las opciones de mover y eliminar el libro. (Figura 60).

Para la biblioteca 'Historial' (figura 61) las opciones de mover o eliminar no están disponibles.

En caso de que un libro incluido en la biblioteca no se encuentre disponible, ya sea porque el autor lo elimino y este ya no se encuentre en la BBDD o porque se ha cambiado su privacidad a privado, se mostrará en su lugar una tarjeta roja con un mensaje explicativo de porque no está disponible.

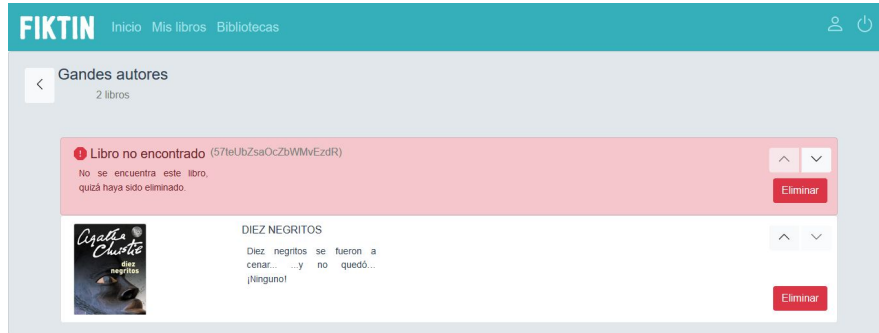


Figura 60. Mostrar biblioteca.



Figura 61. Mostrar biblioteca - Historial.

7.2.3.3. BORRAR BIBLIOTECAS

La opción de eliminar biblioteca está disponible desde la vista de 'Bibliotecas'. Requiere una confirmación por parte del usuario (figura 62) y resulta en la eliminación de la biblioteca de la BBDD.

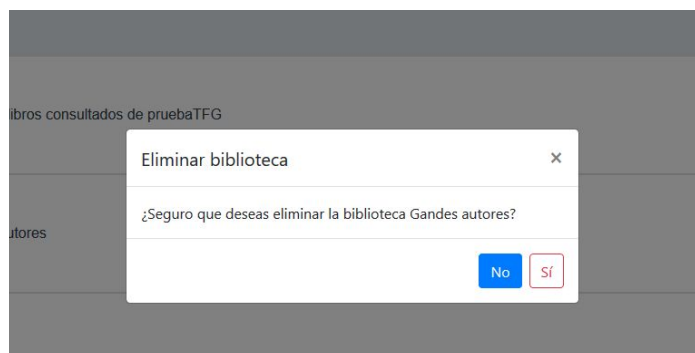


Figura 62. Eliminar biblioteca.

7.2.3.4. AÑADIR LIBRO A BIBLIOTECA

La funcionalidad de añadir libros a la biblioteca está disponible para los libros publicados y los propios. Se puede acceder a ella desde la vista de ‘Inicio’ a través del icono de un marcador amarillo o desde ‘Mis libros’ desde la pestaña de opciones.

Como se ve en la figura 63, la opción de añadir a biblioteca muestra una lista de las bibliotecas a las que se puede añadir el libro o en las que ya está añadido.

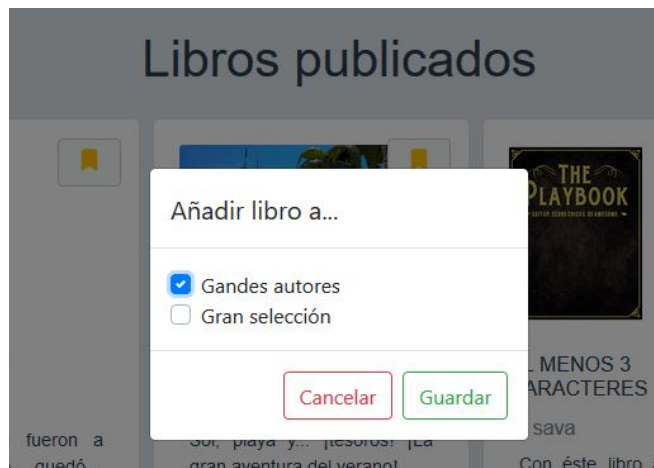


Figura 63. Añadir libro a biblioteca.

Su funcionamiento a nivel de código consiste en buscar las bibliotecas seleccionadas por el usuario en la BBDD, añadir el libro al array de array_keys de esa biblioteca y guardar los cambios. (Figura 64).

```
91 addToLibrary: async function (selectedLibraries) {
92   for (var i = 0; i < selectedLibraries.length; ++i) {
93     var a
94     await librariesCollection.doc(selectedLibraries[i]).get().then(doc => {
95       if (doc.exists) {
96         a = doc.data().array_keys
97       }
98     })
99     a.push(this.primaryKeys[this.showModal])
100    librariesCollection.doc(selectedLibraries[i]).update({
101      array_keys: a
102    })
103  }
104  this.addToLibraryButton(-1)
105 },
```

Figura 64. Código de añadir libro a biblioteca.

7.2.3.5. RETIRAR LIBRO DE BIBLIOTECAS

Para retirar un libro de una biblioteca hay que pulsar el botón eliminar de la tarjeta del libro dentro de la biblioteca en cuestión. Al hacerlo el libro se retira de la lista almacenada localmente de los libros de la biblioteca (referencesList) y se

actualiza en la BBDD la lista de libros (array_keys) de la librería con las claves actuales. (Figura 65).

```
211 /**
212  * Actualizamos el campo "array_keys" de la bbdd en base a la lista "referencesList".
213  */
214 updateReferencesList () {
215     librariesCollection.doc(this.libID).update({
216         array_keys: this.referencesList
217     })
218 },
219
220 /**
221  * @param {int} index: Índice en "booksList" del libro eliminado.
222  * Cada vez que se presiona el botón de borrar un libro se accede a este método.
223  * Su procedimiento es:
224  * Elimina el libro del array "referencesList".
225  * Llama a "updateReferencesList()" para actualizar la bbdd.
226  * Refresca la página.
227  */
228 deleteButton (index) {
229     this.referencesList.splice(index, 1)
230     this.updateReferencesList()
231     this.refresh()
232 },
233
```

Figura 65. Código de quitar libro de biblioteca.

7.2.3.6. MOVER LIBRO EN LA BIBLIOTECA

El cambio de orden de los libros dentro de una biblioteca se realiza usando las flechas hacia arriba y hacia abajo que se encuentran en la tarjeta del mismo. Al pulsar una de las flechas se intercambian los libros correspondientes a las posiciones afectadas en el array local de la biblioteca y se actualizan los valores en la BBDD (figura 66).

```
234 /**
235  * @param {int} index: Índice en "booksList" del libro subido.
236  * Cada vez que se presiona el botón de subir un libro se accede a este método.
237  * Su procedimiento es:
238  * Actualiza el array "referencesList" subiendo el libro de "index" y bajando el de "preIndex".
239  * Llama a "updateReferencesList()" para actualizar la bbdd.
240  * Refresca la página.
241  */
242 upButton (index) {
243     let preIndex = index - 1
244     let aux = this.referencesList[preIndex]
245     this.referencesList[preIndex] = this.referencesList[index]
246     this.referencesList[index] = aux
247     this.updateReferencesList()
248     this.refresh()
249 },
250
251 /**
252  * @param {int} index: Índice en "booksList" del libro bajado.
253  * Cada vez que se presiona el botón de bajar un libro se accede a este método.
254  * Su procedimiento es:
255  * Actualiza el array "referencesList" bajando el libro de "index" y subiendo el de "posIndex".
256  * Llama a "updateReferencesList()" para actualizar la bbdd.
257  * Refresca la página.
258  */
259 downButton (index) {
260     let posIndex = index + 1
261     let aux = this.referencesList[posIndex]
262     this.referencesList[posIndex] = this.referencesList[index]
263     this.referencesList[index] = aux
264     this.updateReferencesList()
265     this.refresh()
266 },
267
```

Figura 66. Código de mover un libro en la biblioteca.

7.2.3.7. MOSTRAR LISTA DE BIBLIOTECAS

En la vista de ‘Bibliotecas’ se muestra la lista de bibliotecas del usuario, como en la figura 67. Cada una en una tarjeta con su nombre, privacidad (ojo tachado/sin tachar), una breve descripción y el número de libros que contiene.

Desde esta lista de bibliotecas el usuario accede a la vista de las bibliotecas en particular, que le muestra los libros en esta contenida.

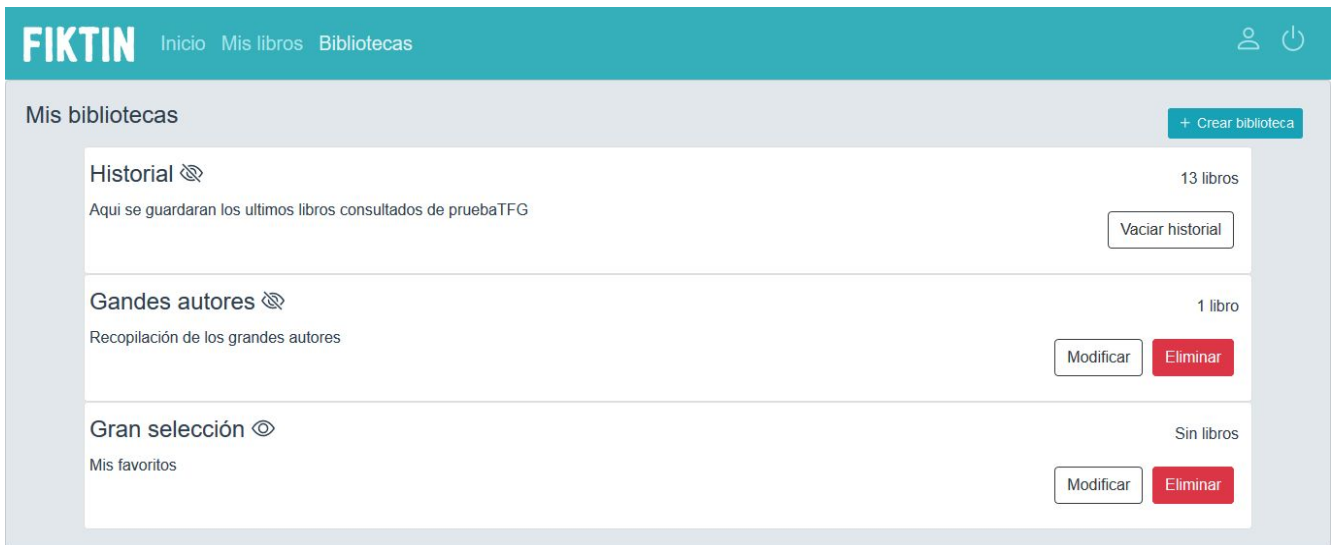


Figura 67. Lista de bibliotecas.

7.2.3.8. VACIAR HISTORIAL

La funcionalidad de vaciar historial solo disponible para la biblioteca ‘Historial’ actualiza la biblioteca en cuestión almacenando su array_keys como un array vacío (figura 68).

```
183  /**
184  * @param {String} id: ID de la biblioteca que se desea vaciar.
185  * Cada vez que se presiona el botón de vaciar, se llama a este método.
186  * Se encarga de vaciar de libros la biblioteca con el id pasado como parámetro. Después de hacerlo, recarga la página.
187  */
188  async btnEmptyHistory (id) {
189    await librariesCollection.doc(id).update({
190      array_keys: []
191    })
192    this.refresh()
193  },
```

Figura 68. Vaciar historial.

7.2.4. GADGETS

Los gadgets son los principales elementos de la edición del libro. Gracias a ellos se puede escribir texto, agregar puzzles o cambios de secciones.

7.2.4.1. INSERTAR GADGET

La inserción de gadgets se lleva a cabo usando el menú lateral izquierdo de la figura 69 que se encuentra en la vista ‘editBook’.

Al pulsar sobre uno de los gadgets se llama a la función ‘add’ del gadget (figura 70). Esta función añade al array ‘data’ la información del gadget en el índice que representa su posición dentro de la sección. No se almacena la misma información para todos los gadgets, cada uno tiene su propia función de añadir gadget para incluir en el array solo los campos que le correspondan.

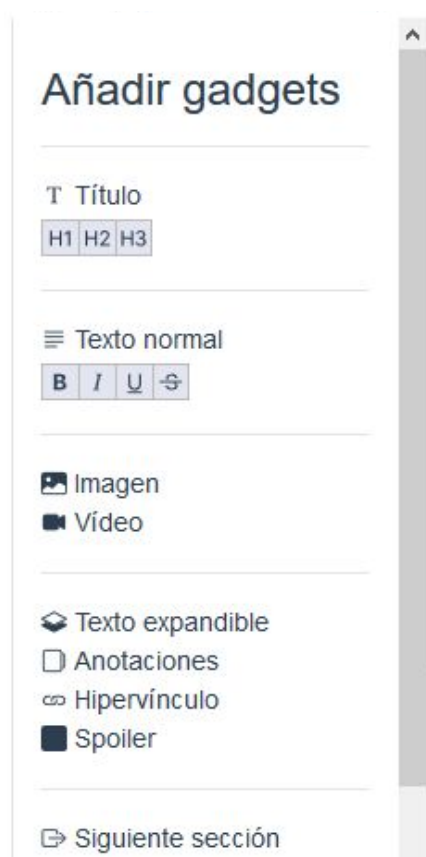


Figura 69. Añadir gadget.

```

addNormal () {
  this.data.splice(this.lastPress + 1, 0, { htmlText: '', component: 'Normal', componentName: 'Texto normal' })
},
addTitle () {
  this.data.splice(this.lastPress + 1, 0, { plainText: '', htmlText: '<h1></h1>', component: 'Header1', componentName: 'Título' })
},
addExpandableText () {
  this.data.splice(this.lastPress + 1, 0, { mainText: '', expandedText: '', component: 'ExpandableText', componentName: 'Texto expandible' })
},
addPopupText () {
  this.data.splice(this.lastPress + 1, 0, { mainText: '', popupText: '', htmlText: '', picture: '', component: 'PopupText', componentName: 'Texto emergente' })
},
addHyperlink () {
  this.data.splice(this.lastPress + 1, 0, { mainText: '', htmlText: '', hyperlinkText: '', component: 'Hyperlink', componentName: 'Hipervínculo' })
},
addSpoiler () {
  this.data.splice(this.lastPress + 1, 0, { plainText: '', htmlText: '<span></span>', component: 'Spoiler', componentName: 'Spoiler' })
},
addPicture () {
  this.data.splice(this.lastPress + 1, 0, { htmlText: '', url: '', component: 'Picture', componentName: 'Imagen' })
},
addVideo () {
  this.data.splice(this.lastPress + 1, 0, { htmlText: '', url: '', component: 'Video', componentName: 'Vídeo' })
},
addSectionChange () {
  if (this.sectionsData.length > 1) {
    this.data.splice(this.lastPress + 1, 0, { plainText: '', htmlText: '<span></span>', next: this.sectionsData[0].value, component: 'ChangeSection', componentName: 'Cambio de sección' })
  } else window.alert('Para añadir un cambio de sección debes tener más de una sección creada')
},

```

Figura 70. Código de añadir gadget.

7.2.4.2. EDITAR GADGET

La edición de los gadgets se hace en la misma vista de 'editBook'. Al añadir un gadget este se inserta en la hoja de la sección en su forma de edición, como se muestra en la figura 71.

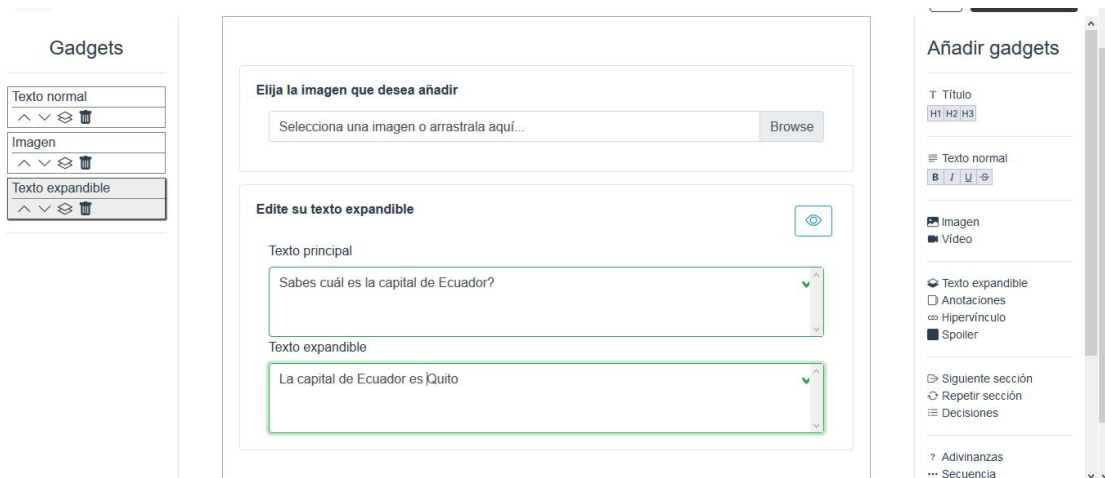


Figura 71. Editar gadget.

Cada gadget es un componente dentro de la aplicación que se añade a la vista de edición cuando este está presente en el array 'data', el código correspondiente se encuentra en la figura 74.

```
<div class="document" @keyup="checkStyles" @keydown.tab.prevent>
<div class="editable" v-for="(text, index) in data" :key="index" @click="lastElementPressed(index), checkStyles()">
  <Normal v-if="text.component=='Normal'"
    :htmlTextAux="text.htmlText"
    :index="index"
    :boldUse="boldUse"
    :italicUse="italicUse"
    :underlineUse="underlineUse"
    :strikeThroughUse="strikeThroughUse"
    :lastPress="lastPress"
    @html="saveHTML"
    @delete="checkDelete"/>
  <Header1 v-if="text.component=='Header1'"
    :htmlTextAux="text.htmlText"
    :plainTextAux="text.plainText"
    :index="index"
    @html="savePlaneAndHTML"/>
  <Header2 v-if="text.component=='Header2'"
    :htmlTextAux="text.htmlText"
    :plainTextAux="text.plainText"
    :index="index"
    @html="savePlaneAndHTML"/>
  <Header3 v-if="text.component=='Header3'"
    :htmlTextAux="text.htmlText"
    :plainTextAux="text.plainText"
    :index="index"
    @html="savePlaneAndHTML"/>
  <ExpandableText v-if="text.component=='ExpandableText'"
    :index="index"
    :mainTextAux="text.mainText"
    :expandedTextAux="text.expandedText"
    :lastPressed="lastPress"
    @html="saveExpandableText"/>
  <PopupText v-if="text.component=='PopupText'"
    :index="index"
    :mainTextAux="text.mainText"
    :popupTextAux="text.popupText"
    :htmlTextAux="text.htmlText"
    @html="savePopupText"/>
</div>
</div>
```

Figura 72. Código que añade el componente del gadget en edición.

Es dentro de cada uno de esos componentes donde se encuentran los formularios (figura 73) para rellenar los datos del gadget y se guarda la información introducida por el usuario en los campos correspondientes del gadget.

```
<template>
<div>
  <b-card>
    <div class="d-flex justify-content-start">
      <h6 class="title">Elija la imagen que desea añadir</h6>
    </div>

    <b-container fluid class="col">
      <b-form-file
        @change="onFileSelected"
        class="my-2"
        placeholder="Selecciona una imagen o arrastrala aquí..."
        drop-placeholder="Arrastra aquí la imagen..."
        accept="image/*"
      ></b-form-file>
      <b-row class="my-1">
        <p v-if="this.selectedFile !== '' && this.picture === ''">Espere a que cargue la imagen</p>
        <b-img v-if="this.picture !== '' :src="this.picture" style="min-width: 710px; max-width: 710px; height: auto;padding-top: 13px; padding-bottom: 13px;"></b-img>
      </b-row>
    </b-container>
  </b-card>
</div>
</template>
```

Figura 73. Formulario del gadget de añadir imagen.

Los cambios en el gadget se guardan automáticamente con la función ‘save’ de la figura 74 que emite un evento a la vista ‘editBook’ pasando los valores actualizados de los datos del gadget. El evento es recogido y llama a otra función save que guarda los cambios del gadget al array ‘data’ (figura 75).

```
save: async function () {
  var htmlText = ('')
  this.$emit('html', htmlText, this.picture, this.index)
}
```

Figura 74. Guardado del gadget de añadir imagen.

```
saveHTMLMultimedia (htmlText, url, index) {
  this.data[index].htmlText = htmlText
  this.data[index].url = url
},
saveHTML (htmlText, index) {
  this.data[index].htmlText = htmlText
},
savePlaneAndHTML (plainText, htmlText, index) {
  this.data[index].plainText = plainText
  this.data[index].htmlText = htmlText
},
saveHTMLAndSection (htmlText, plainText, section, index) {
  this.data[index].htmlText = htmlText
  this.data[index].plainText = plainText
  this.data[index].next = section
},
saveChoices (decisions, numberOfOptions, index) {
  this.data[index].choices = decisions
  this.data[index].numberOfOptions = numberOfOptions
},
```

Figura 75. Guardado de gadget en ‘editBook’.

7.2.4.3. BORRAR GADGET

En la figura 76 se muestra que la opción de eliminar esta accesible desde el panel de gadgets en el icono de la papelera.

Al pulsar el icono se llama a la función ‘itemDelete’, figura 77, pasándole la posición en el array del gadget.

Esta función hace una comprobación, si el gadget es de tipo ‘CustomBox’ y está en modo escritura llama a la función ‘checkread’ (figura 78) que consulta el resto de ‘CustomBoxes’. En caso de encontrar alguna que tenga el mismo nombre (es decir, que use la misma variable) la elimina del array ‘temporalCustomBoxes’.

Después se retira el gadget del array ‘data’, en caso de que la sección no contenga más gadgets la variable ‘lastPress’, que indica el último gadget pulsado, toma el valor ‘-1’.

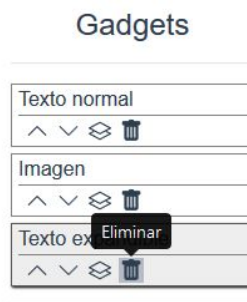


Figura 76. Eliminar gadget.

```
itemDelete (index) {  
  if (this.data[index].component === 'CustomBox' && this.data[index].mode === 'write') this.checkRead(this.data[index].name)  
  this.data.splice(index, 1)  
  if (this.data.length === 0) this.lastPress = -1  
},
```

Figura 77. Código de eliminar gadget.

```
checkRead (name) {  
  // Si se cambia el modo de la casilla personalizada a lectura, se llama a este método.  
  // Como todos los nombres son únicos, nunca se borrará un valor que se use en otros sitios  
  // Esto además se garantiza ya que si el nombre no es válido, la casilla personalizada manda '' como nombre  
  // Sin embargo, si el nombre es válido, manda dicho nombre  
  if (name !== '') {  
    var end = false  
    for (var i = 0; i < this.temporalCustomBoxes.length && !end; ++i) {  
      if (this.temporalCustomBoxes[i].name === name) {  
        this.temporalCustomBoxes.splice(i, 1)  
        end = true  
      }  
    }  
  }  
},
```

Figura 78. Código de ‘checkRead’.

7.2.4.4. DUPLICAR GADGET

Como se muestra en la figura 79 la opción de duplicar es accesible desde el panel de gadgets en el icono de dos capas.

Cuando se pulsa el icono se llama a la función 'itemClone' que añade al array 'data' el nuevo gadget con los datos del gadget clonado. (Figura 80).



Figura 79. Duplicar gadget.

```
itemClone (index) {
  // Debido a problemas de clonaciones, es necesario poner los textos manualmente
  if (this.data[index].component === 'Normal') this.data.splice(index + 1, 0, { htmlText: this.data[index].htmlText, component: 'Normal',
    componentName: 'Texto normal' })
  else if (this.data[index].component === 'Header1') this.data.splice(index + 1, 0, { plainText: this.data[index].plainText, htmlText: this.data[
    index].htmlText, component: 'Header1', componentName: 'Titulo' })
  else if (this.data[index].component === 'Header2') this.data.splice(index + 1, 0, { plainText: this.data[index].plainText, htmlText: this.data[
    index].htmlText, component: 'Header2', componentName: 'Titulo' })
  else if (this.data[index].component === 'Header3') this.data.splice(index + 1, 0, { plainText: this.data[index].plainText, htmlText: this.data[
    index].htmlText, component: 'Header3', componentName: 'Titulo' })

  else if (this.data[index].component === 'ExpandableText') this.data.splice(index + 1, 0, { mainText: this.data[index].mainText, expandedText: this.
    data[index].expandedText, component: 'ExpandableText', componentName: 'Texto expandible' })
  else if (this.data[index].component === 'PopupText') this.data.splice(index + 1, 0, { mainText: this.data[index].mainText, popupText: this.data[
    index].popupText, htmlText: this.data[index].htmlText, picture: this.data[index].picture, component: 'PopupText', componentName: 'Texto emergente
' })
  else if (this.data[index].component === 'Hyperlink') this.data.splice(index + 1, 0, { htmlText: this.data[index].htmlText, mainText: this.data[
    index].mainText, hyperlinkText: this.data[index].hyperlinkText, component: 'Hyperlink', componentName: 'Hipervinculo' })
  else if (this.data[index].component === 'Spoiler') this.data.splice(index + 1, 0, { plainText: this.data[index].plainText, htmlText: this.data[
    index].htmlText, component: 'Spoiler', componentName: 'Spoiler' })
}
```

Figura 80. Código de duplicar gadget.

7.2.4.5. PREVISUALIZAR GADGET

Algunos gadgets tienen la opción de ser previsualizados en su modo de edición. Esta funcionalidad es accesible desde el propio gadget en modo de edición pulsando sobre el botón con el icono de un ojo (figura 81). Al pulsarlo se añade a la vista del gadget un apartado donde se muestra el gadget como lo haría en la lectura (figura 82).

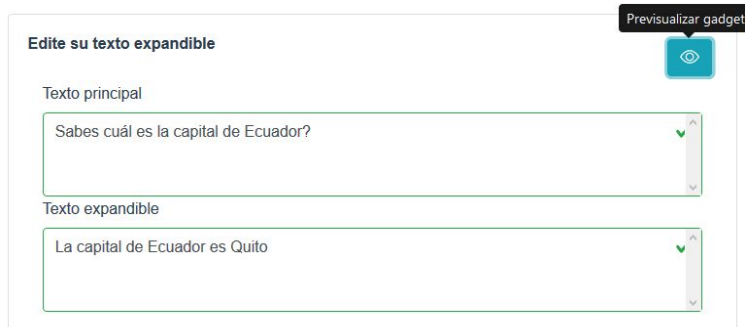


Figura 81. Previsualizar gadget.

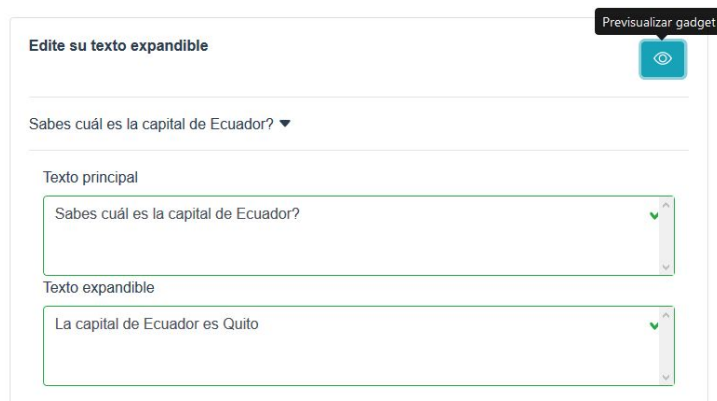


Figura 82. Previsualización de gadget.

7.2.4.6. MOVER GADGET

La funcionalidad de mover gadget intercambia las posiciones de los gadgets afectados dentro del array 'data' (figura 83).



Figura 83. Mover gadget.

7.2.5. SECCIONES

El contenido de los libros se estructura en la aplicación y en la base de datos por secciones. Cada libro contiene el array de sus secciones en el orden de lectura y cada sección guarda un array con sus gadgets.

La gestión de las secciones (crear, mover, eliminar, ...) se realiza a través de una ventana emergente a la que se accede desde la vista de 'editBook' pulsando sobre 'Gestión de secciones'.

7.2.5.1. CREAR SECCIÓN

Para crear una sección hay que pulsar sobre 'Nueva sección' dentro de la ventana emergente de gestión de secciones (figura 84).

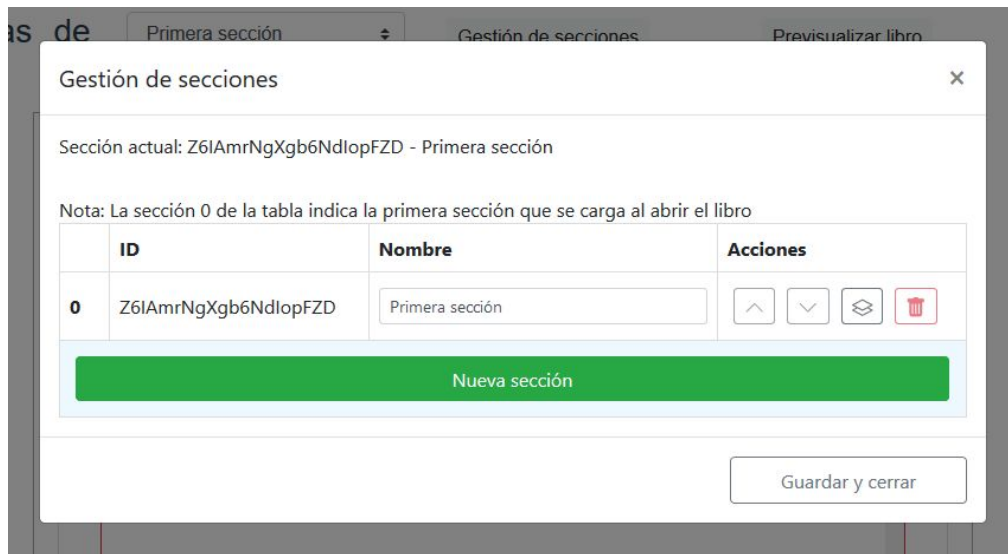


Figura 84. Crear sección.

Cuando se crea una nueva sección esta se añade a la base de datos a la colección de secciones con los siguientes datos: nombre de la sección, id del usuario autor del libro, título del libro y el array de gadgets con un gadget de tipo 'Texto normal'.

También se añade el id y el nombre de la sección (que por defecto se pone a 'Nueva sección') al array local 'sectionsData' que es el array utilizado para mostrar las secciones en la ventana emergente. A continuación, se extraen todos los IDs de las secciones actuales y se actualiza la información mostrada (figura 85).

```

async newSection () {
  this.busy = true
  this.changes = true
  var a = await sectionsCollection.add({
    name: 'Nueva sección',
    book_author_ID: this.book_author_ID,
    book_title: this.book_title,
    gadgets: [{
      htmlText: '',
      component: 'Normal',
      componentName: 'Texto normal'
    }]
  })
  this.sectionsData.push({ id: a.id, name: 'Nueva sección' })
  var c = this.extractIDs()
  await this.$emit('update', c)
  this.busy = false
},

```

Figura 85. Código de crear sección.

7.2.5.2. MODIFICAR SECCIÓN

La funcionalidad de modificar sección consiste en cambiar el nombre de la sección. Este valor se modifica en el array local y se guarda en la BBDD al cerrar la ventana emergente pulsando en el botón de ‘Guardar y cerrar’ o la cruz de arriba a la derecha (figura 86).

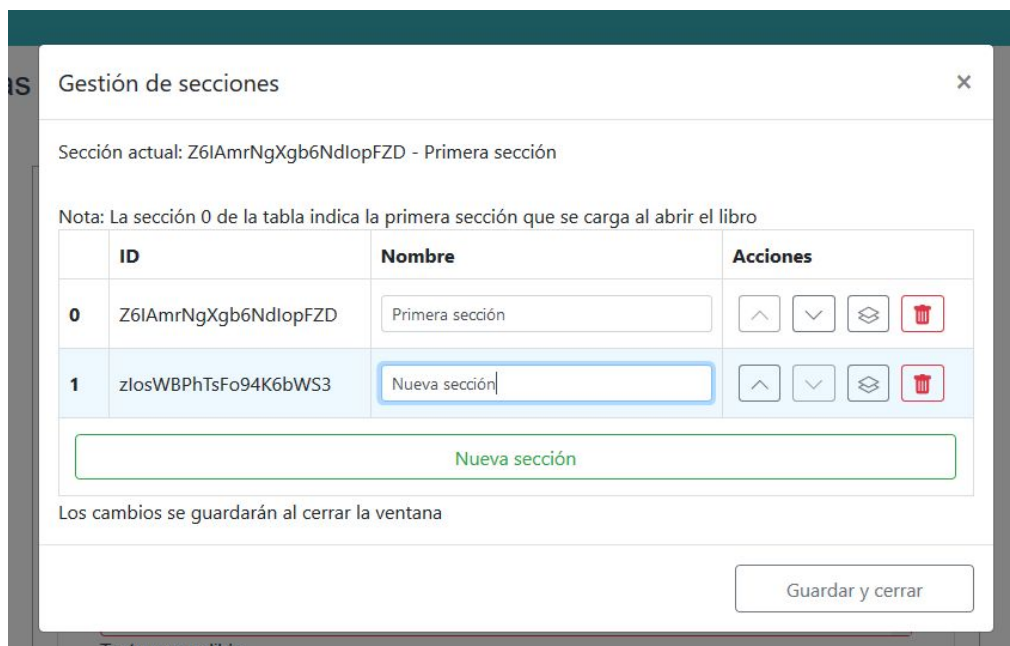


Figura 86. Código de modificar sección.

Con cada acción que modifique las secciones se pone a cierto el booleano ‘changes’. Si al cerrar la sección este booleano está a true la función ‘closeBoton’ de la figura 87 guarda los datos de las secciones almacenadas en ‘sectionsData’ en la base de datos.

```

async closeButton () {
  if (this.checkNames()) {
    // Cada vez que se cierra el modal, actualizamos toda la información de las secciones solo si ha habido cambios en él
    if (this.changes) {
      this.closing = true
      var c = this.extractIDs()
      await this.$emit('update', c)
      for (var i = 0; i < this.sectionsData.length; ++i) {
        await sectionsCollection.doc(this.sectionsData[i].id).get().then(doc => {
          if (doc.exists) {
            sectionsCollection.doc(this.sectionsData[i].id).update({
              name: this.sectionsData[i].name
            })
          }
        })
      }
    }
    await this.$emit('load', this.id)
  }
  await this.$emit('cancel')
} else window.alert('Debes dar un nombre válido a todas las secciones (nombres vacíos o solo con espacios no son válidos)')
},

```

Figura 87. Código de guardar sección.

7.2.5.3. BORRAR SECCIÓN

La funcionalidad de eliminar sección se realiza en el método ‘sectionDelete’ de la figura 88. Este método comprueba que haya al menos más de una sección, ya que no se puede dejar un libro sin secciones, descarga los gadgets de la sección y llama a la función ‘deleteCustomBoxes’ que se encarga de cambiar el modo de las casillas personalizadas que lean de la misma variable que las casillas escritas en la sección a eliminar. A continuación, se elimina la sección y se cambia el índice que marca la sección actual a la siguiente sección o, si no hay, a la primera.

```

async sectionDelete (index) {
  this.busy = true
  this.changes = true
  // No se pueden borrar secciones si es la única presente en el libro
  if (this.sectionsData.length > 1) {
    var c
    var a
    // Borramos los índices de escritura de las custom boxes
    // Para ello, descargamos los gadgets de la sección a borrar
    await sectionsCollection.doc(this.sectionsData[index].id).get().then(doc => {
      a = doc.data().gadgets
    })
    this.$emit('deleteCustomBoxes', a)
    // Borramos la sección
    await sectionsCollection.doc(this.sectionsData[index].id).delete()
    // Si borramos la sección en la que nos encontramos, cargamos la del índice 0
    if (this.sectionsData[index].id === this.id) {
      this.sectionsData.splice(index, 1)
      c = this.extractIDs()
      await this.$emit('update', c)
      await this.$emit('load', this.sectionsData[0].id)
    } else {
      this.sectionsData.splice(index, 1)
      c = this.extractIDs()
      await this.$emit('update', c)
    }
  } else {
    window.alert('No se puede borrar esta sección al no haber ninguna más')
  }
  this.busy = false
},

```

Figura 88. Código de eliminar sección.

7.2.5.4. DUPLICAR SECCIÓN

La función ‘sectionClone’, ver figura 89, es la encargada de clonar una sección. En primer lugar, si la sección a clonar es la sección actual la guarda en la base de datos para asegurar que también se clonen los últimos cambios. Descarga todos los datos de la sección, poniendo el nombre como copia y en caso de haber gadgets de tipo ‘customBox’ se ponen en modo lectura y con esta información se guarda en la base de datos la nueva sección. Se actualiza el array ‘sectionsData’ para incluir la sección clonada.

```
async sectionClone (index) {
  var a
  this.busy = true
  this.changes = true
  // Por si tarda más de la cuenta, ponemos una fila cargando
  this.sectionsData.splice(index + 1, 0, { id: 'Cargando...', name: 'Cargando...' })
  // Si clonamos la sección actual, primero la guardamos
  if (this.sectionsData[index].id === this.id) {
    await this.$emit('saveActual')
  }
  // Descargamos todos los datos de la sección a clonar
  await sectionsCollection.doc(this.sectionsData[index].id).get().then(doc => {
    a = { book_author_ID: doc.data().book_author_ID,
          book_title: doc.data().book_title,
          gadgets: doc.data().gadgets,
          name: doc.data().name + '-copia' }
  })
  // Si hay casillas personalizadas, las ponemos TODAS en modo lectura
  a.gadgets = this.findCustomBoxes(a.gadgets)
  // Creamos una nueva sección con los datos recién descargados
  var b = await sectionsCollection.add({
    book_author_ID: a.book_author_ID,
    book_title: a.book_title,
    gadgets: a.gadgets,
    name: a.name.substring(0, 50)
  })
  // Quitamos la fila de cargando e insertamos los datos de la nueva sección
  this.sectionsData.splice(index + 1, 1)
  this.sectionsData.splice(index + 1, 0, { id: b.id, name: a.name.substring(0, 50) })
  // Actualizamos las secciones de la bd de libros
  var c = this.extractIDs()
  await this.$emit('update', c)
  this.busy = false
},
```

Figura 89. Código de clonar sección.

7.2.5.5. MOVER SECCIÓN

La funcionalidad de mover secciones permite cambiar el orden de lectura de las secciones del libro. Las funciones ‘sectionUp’ y ‘sectionDown’ de la figura 90 aplican este cambio de orden intercambiando las secciones en el array ‘sectionsData’.

Igual que al modificar sección estos cambios se guardan en la base de datos al cerrar la ventana emergente.

```
sectionUp (index) {
  this.busy = true
  this.changes = true
  if (index > 0) {
    var aux = this.sectionsData[index]
    this.$set(this.sectionsData, index, this.sectionsData[index - 1])
    this.$set(this.sectionsData, index - 1, aux)
  }
  this.busy = false
},
sectionDown (index) {
  this.busy = true
  this.changes = true
  if (index < this.sectionsData.length - 1) {
    var aux = this.sectionsData[index]
    this.$set(this.sectionsData, index, this.sectionsData[index + 1])
    this.$set(this.sectionsData, index + 1, aux)
  }
  this.busy = false
},
```

Figura 90. Código de mover sección.

8. CAPÍTULO VIII: EVALUACIÓN DE LA USABILIDAD

En este capítulo se describirán los resultados obtenidos tras realizar una evaluación de la usabilidad a distintos usuarios sobre la aplicación, así como la metodología utilizada.

8.1. METODOLOGÍA

Para realizar una evaluación completa se preparó una serie de tareas que realizaban un recorrido por la aplicación para que los usuarios tuvieran una idea más clara del funcionamiento de la misma y así poder responder a la encuesta. Estas tareas se distribuyeron a veinte personas con conocimientos y edades dispares, entre ellos compañeros de la facultad de informática con conocimientos de aplicaciones web y su funcionamiento, personas entre 20 y 30 años que usan aplicaciones pero no conocen el funcionamiento de ellas, y personas entre 50 y 65 años que utilizan la web principalmente para encontrar información.

La evaluación se ha realizado con Google Forms [30] y consta de 33 preguntas, realizadas con escalas Likert [37], relacionadas con las tareas propuestas, preguntando si ha resultado intuitivo y sencillo la realización de las mismas. La escala utilizada consta de 5 niveles, significando cada uno de ellos lo siguiente:

- 1 - La tarea ha sido muy poco intuitiva o muy difícil de realizar
- 2 - La tarea ha sido poco intuitiva o algo difícil de realizar
- 3 - La tarea no ha sido especialmente sencilla pero tampoco especialmente difícil de realizar.
- 4 - La tarea ha sido sencilla e intuitiva de realizar
- 5 - La tarea ha sido muy sencilla e intuitiva de realizar

En el anexo III se muestran capturas de pantalla de dichas preguntas.

8.2. RESPUESTAS DEL CUESTIONARIO

En este apartado se describen los resultados obtenidos en el formulario.

8.2.1. EVALUACIÓN “CREAR USUARIO”

Respecto a la creación de un usuario el 90% de los encuestados opina que ha sido muy sencillo e intuitivo, otorgando la máxima nota de 5, mientras que a un 10% le ha parecido sencillo pero no absolutamente obvio, estando en este grupo personas que han puntuado con la segunda nota más alta, un 4 (véase figura 91).

Crear un usuario:

20 respuestas

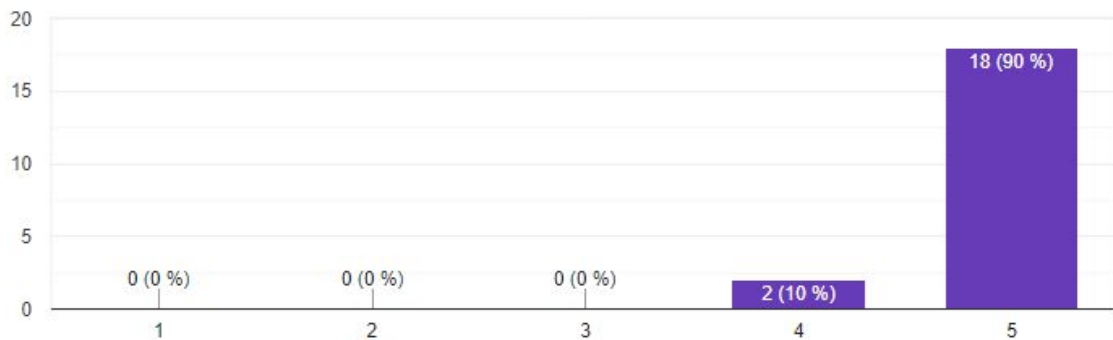


Figura 91. Resultados de crear usuario.

8.2.2. EVALUACIÓN “INICIAR SESIÓN”

En cuanto a iniciar sesión el 100% de los encuestados han respondido con un 5, que significa totalmente sencillo e intuitivo de realizar (véase figura 92).

Iniciar sesión:

20 respuestas

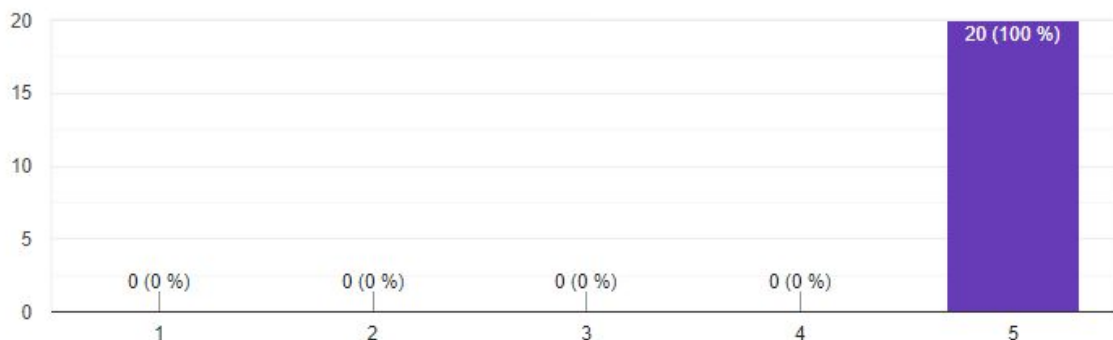


Figura 92. Resultados de iniciar sesión.

8.2.3. EVALUACIÓN “ACCEDER A DATOS DE USUARIO”

En el caso de acceder a los datos de usuario un 80% ha otorgado la máxima nota, opinando que es muy sencillo e intuitivo, un 5% lo ha evaluado con un 4 y un 15% ha elegido calificarlo con un 3, significando que no es muy claro, pero que se puede llegar a entender sin mucha dificultad (véase figura 93).

Acceder a tus datos de usuario:

20 respuestas

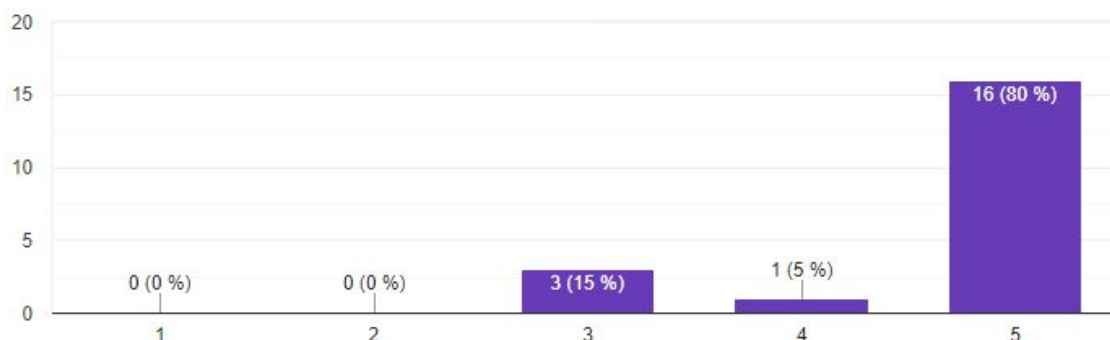


Figura 93. Resultados de acceso a los datos del usuario.

8.2.4. EVALUACIÓN “CREAR BIBLIOTECA ”

Las opiniones obtenidas acerca de crear una biblioteca cuentan con una mayoría del 55% que ha elegido la mejor evaluación con nota de un 5, mientras que un 35% ha marcado la opción de un 4. Del porcentaje restante un 5% ha otorgado un 3, indicando que no ha sido claro pero que se puede realizar sin mucha dificultad, y el último 5% opina que ha sido poco claro y sencillo puntuando la tarea con un 2 (véase figura 94).

Crear una biblioteca:

20 respuestas

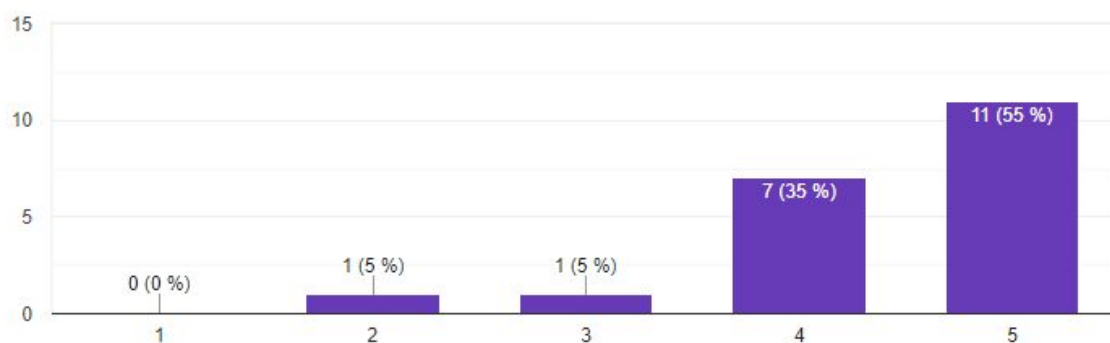


Figura 94. Resultados de crear biblioteca.

8.2.5. EVALUACIÓN “AGREGAR LIBRO A BIBLIOTECA”

Respecto a agregar un libro desde la pantalla de *inicio* a una biblioteca el 60% de los encuestados opina que ha sido muy sencillo e intuitivo con una nota de 5, un 15% ha otorgado una nota de 4, un 20% opina que no ha sido demasiado sencillo ni intuitivo con una nota de 3 mientras que el 5% restante opina que ha sido muy poco claro e intuitivo, eligiendo la mínima nota posible de 1 (véase figura 95).

Agregar un libro desde inicio a tu biblioteca:

20 respuestas

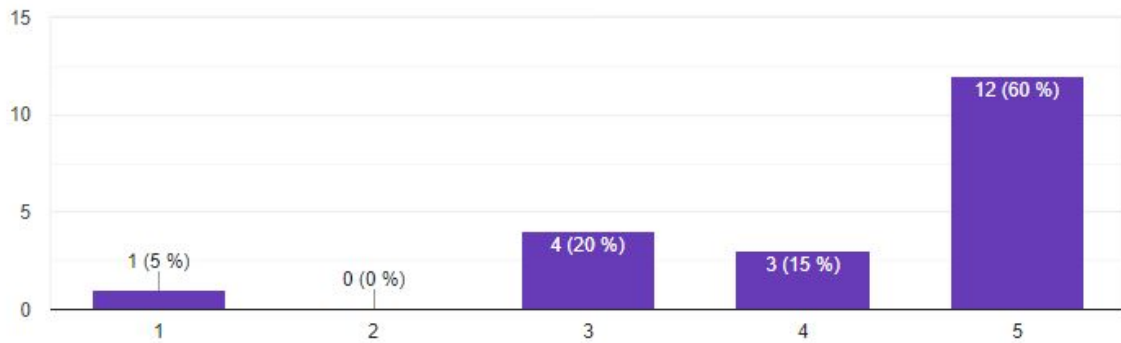


Figura 95. Resultados de agregar libro a biblioteca.

8.2.6. EVALUACIÓN “VER LIBRO AÑADIDO A LA BIBLIOTECA”

Comprobar que el libro agregado a la biblioteca se encuentra dentro de ella le ha parecido muy sencillo e intuitivo, con una nota de 5, al 70% de los encuestados. El 30% restante opina que ha sido sencillo e intuitivo, con una nota de 4 (véase figura 96).

Comprobar que dicho libro se encuentra dentro de tu biblioteca:

20 respuestas

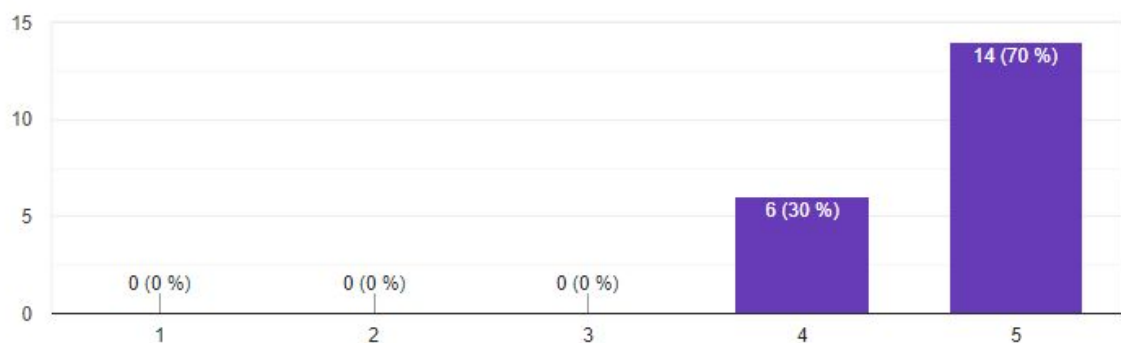


Figura 96. Resultados de buscar el libro añadido en la biblioteca.

8.2.7. EVALUACIÓN “ABRIR LIBRO”

Al 100% de los encuestados les ha resultado muy sencillo e intuitivo abrir el libro de “Piratillas”, el cual se encuentra en la pantalla principal, puntuando la tarea con la máxima nota posible (véase figura 97).

Abrir el libro de "Piratillas":

20 respuestas

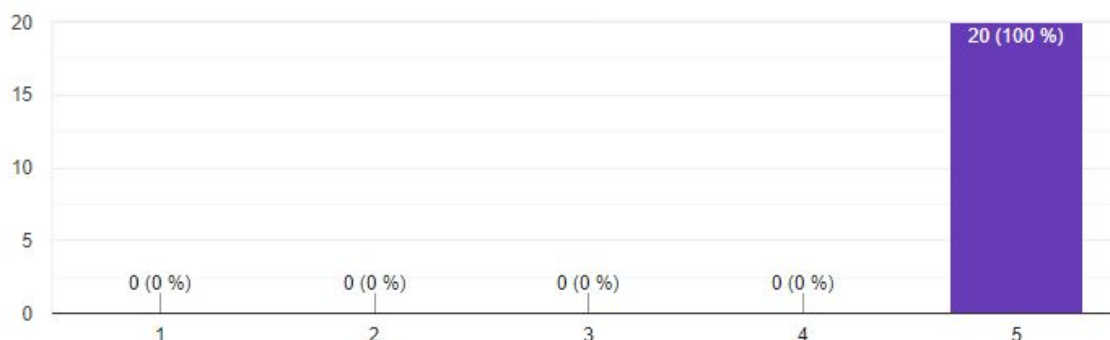


Figura 97. Resultados de abrir el libro 'Piratillas'.

8.2.8. EVALUACIÓN “INTERACCIONES DENTRO DEL LIBRO”

En el caso de la evaluación de las interacciones dentro del libro anteriormente mencionado, un 85% opina que han sido muy sencillas e intuitivas, con una nota de 5, mientras que el restante 15% opina que ha sido sencilla e intuitiva, calificandola con un 4 (véase figura 98).

En el libro de "Piratillas", indica la sencillez de las interacciones dentro del libro:

20 respuestas

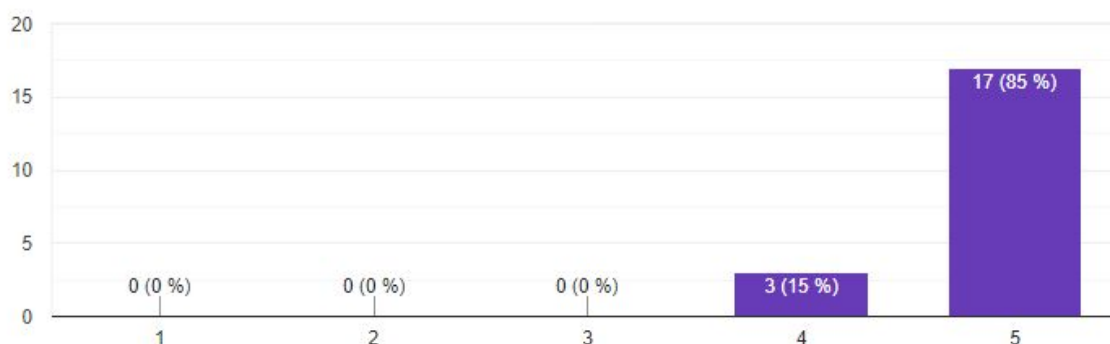


Figura 98. Valoración de la sencillez de las interacciones en el libro.

8.2.9. EVALUACIÓN “LECTURA DEL LIBRO PIRATILLAS”

La lectura del libro de ejemplo “Piratillas” le ha resultado muy poco entretenida a un 5% de los encuestados, a un 40% le ha parecido medianamente entretenida, al 15% entretenida pero mejorable y al 40% restante muy entretenida (véase figura 99).

¿Te ha resultado entretenida la lectura de dicho libro?

20 respuestas

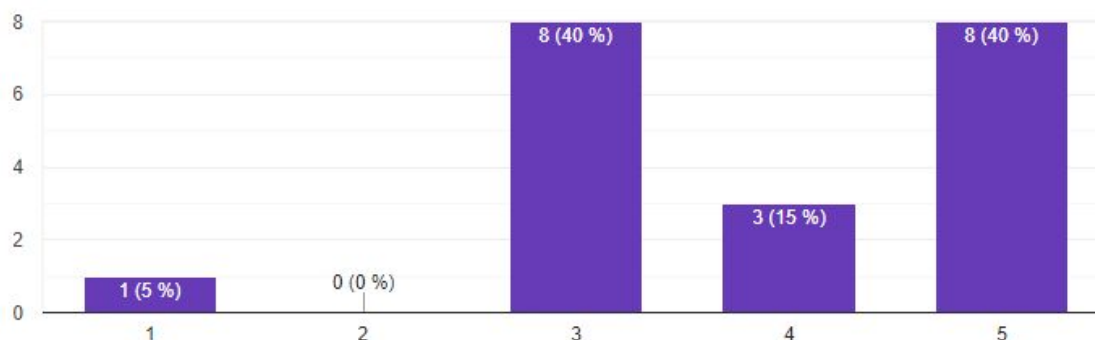


Figura 99. Valoración de la lectura del libro.

8.2.10. EVALUACIÓN “CREAR LIBRO”

En cuanto a la tarea de crear un libro, a un 60% le ha parecido muy sencillo e intuitivo, a un 25% sencillo e intuitivo sin dar la máxima nota, a un 10% no le ha parecido ni muy sencillo ni nada intuitivo con una nota de 3 y al último 5% le ha parecido poco intuitivo y sencillo (véase figura 100).

Crear un libro:

20 respuestas

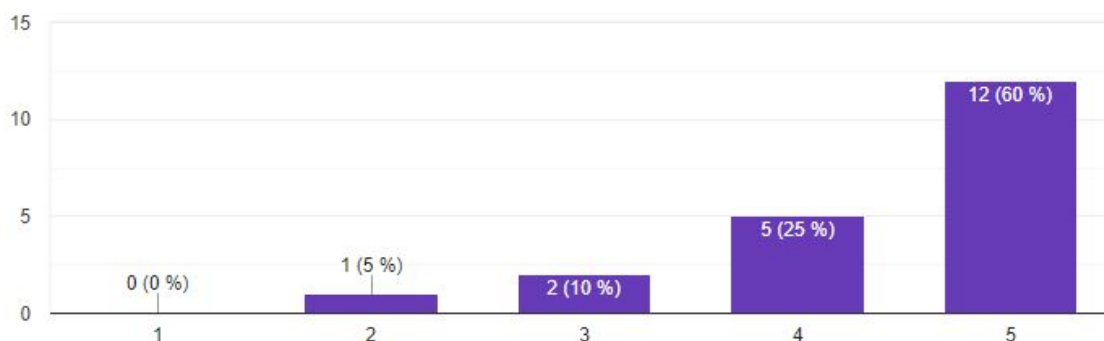


Figura 100. Resultados de la creación del libro.

8.2.11. EVALUACIÓN “AÑADIR LIBRO CREADO A BIBLIOTECA”

La tarea de añadir el libro creado a la biblioteca anteriormente, le ha parecido muy sencillo e intuitivo a un 55% de los encuestados, habiendo otorgado una nota de 5. Al 35% le ha resultado sencillo e intuitivo, con una nota de 4, mientras que al 10% restante lo ha calificado con un 3, significando que no ha sido muy sencillo e intuitivo pero que se puede encontrar la respuesta sin mucha dificultad (véase figura 101).

Añadir el libro creado a tu biblioteca:

20 respuestas

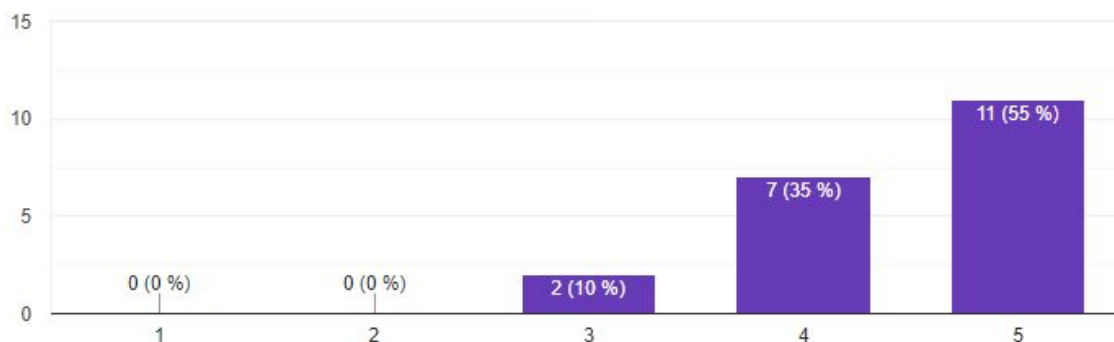


Figura 101. Resultados de añadir el libro creado a la biblioteca.

8.2.12. EVALUACIÓN “AÑADIR PORTADA AL LIBRO”

En el caso de añadir una portada al libro creado, un 65% lo ha puntuado con la máxima nota de 5, un 15% con un 4, un 10% lo ha calificado con un 3, un 5% con un 2 y el 5% restante la ha otorgado la mínima nota de un 1 (véase figura 102).

Añadirle una portada a tu libro:

20 respuestas

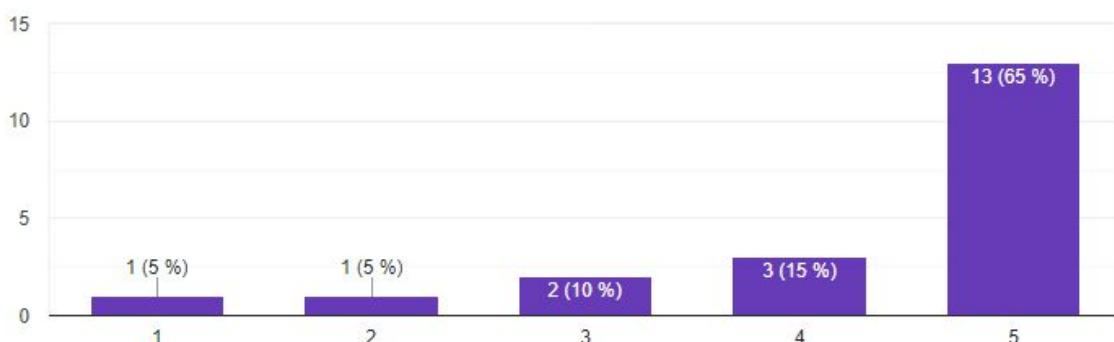


Figura 102. Resultados de añadir portada al libro.

8.2.13. EVALUACIÓN “EDITAR LIBRO CREADO”

Respecto a empezar a editar el libro creado, a un 50% le ha parecido muy sencillo e intuitivo con una nota de un 5, el 30% lo ha puntuado con una nota de un 4, el 15% con un 3 y el último 5% con la mínima nota de un 1, representando que ha sido muy poco sencillo e intuitivo (véase figura 103).

Empezar a editar tu libro:

20 respuestas

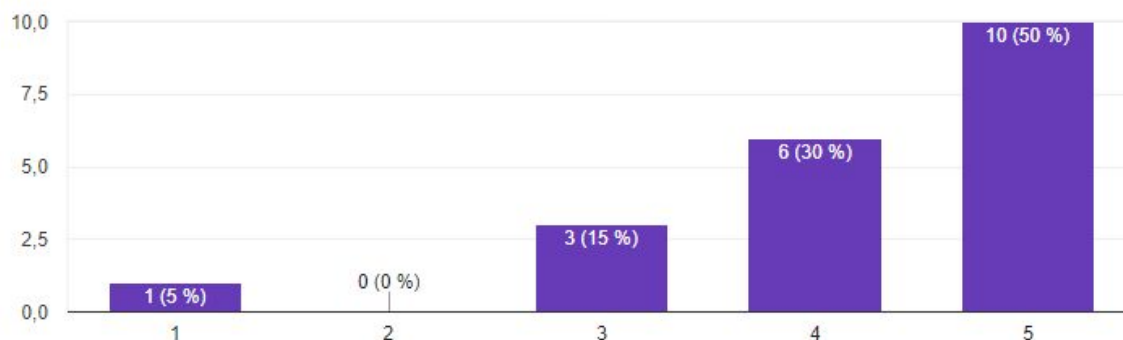


Figura 103. Resultados de iniciar la edición de un libro.

8.2.14. EVALUACIÓN “GADGET TÍTULO”

Añadir el primer gadget de título al libro y escribir algo ha recibido la siguiente evaluación, un 40% lo ha calificado con un 5, un 45% con un 4, un 5% con un 3, otro 5% con un 2 y el 5% restante con un 1 (véase figura 104).

Añadir un título y escribir algo:

20 respuestas

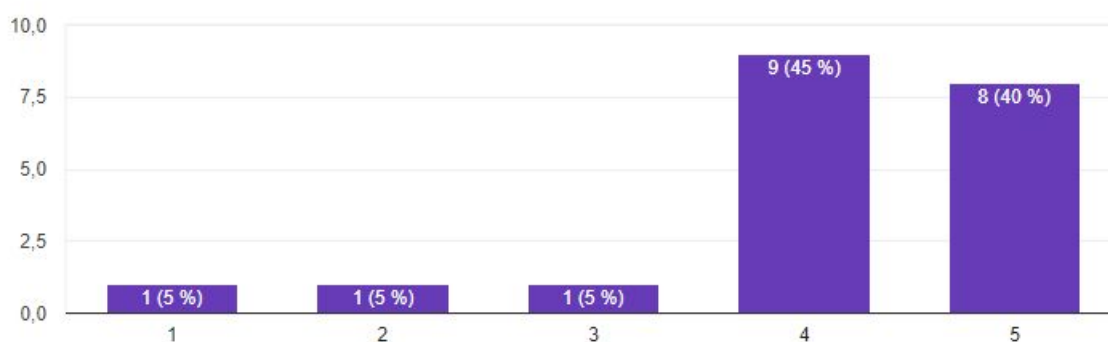


Figura 104. Resultados de escribir título.

8.2.15. EVALUACIÓN “GADGET TEXTO NORMAL”

A un 45% de los encuestados les ha resultado muy intuitivo y sencillo añadir un texto normal, a un 40% sencillo e intuitivo y al 15% restante, calificando con una nota de 3, no les ha parecido muy intuitivo pero tampoco muy difícil (véase figura 105).

Añadir un texto normal y escribir algo:

20 respuestas

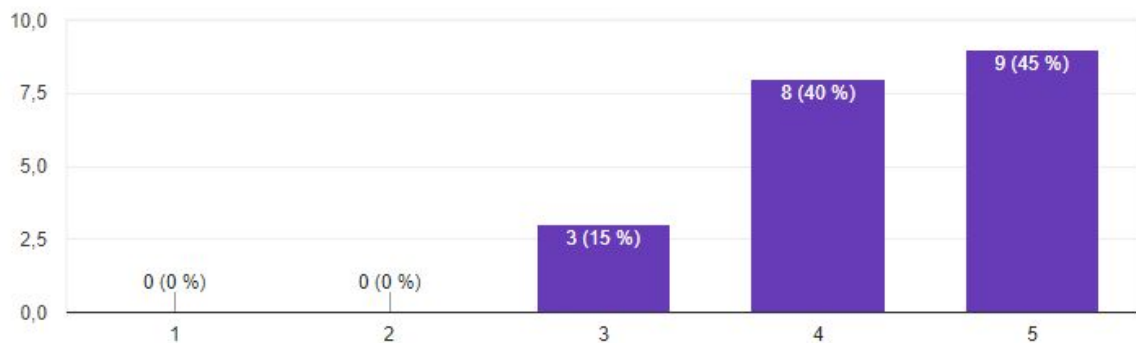


Figura 105. Resultados de escribir texto normal.

8.2.16. EVALUACIÓN “CAMBIAR VALORES GADGETS”

Cambiar los valores de los gadgets de título y texto normal ha tenido una opinión variada, puntuando el 35% de los encuestados con una nota de un 5, el 40% con una nota de un 4, el 20% con un 3 y el último 5% con un 2 (véase figura 106).

Cambiar los valores de los anteriores gadgets:

20 respuestas

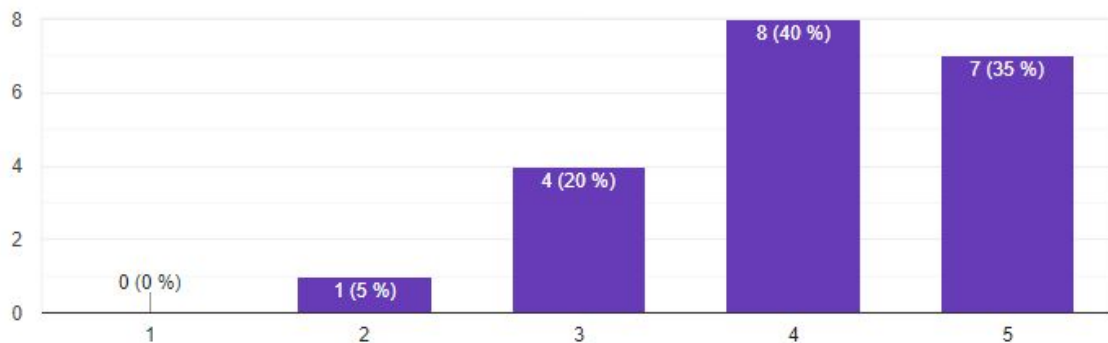


Figura 106. Resultados de modificar gadgets.

8.2.17. EVALUACIÓN “AÑADIR SECCIÓN”

Añadir una nueva sección al libro le ha parecido muy sencillo e intuitivo a un 60% de los encuestados, a un 35%, sin otorgar la nota máxima, le ha parecido sencillo e intuitivo, y a un 5%, con la nota de 3, no le ha parecido muy sencillo pero tampoco excesivamente complicado (véase figura 107).

Añadir una nueva sección al libro:

20 respuestas

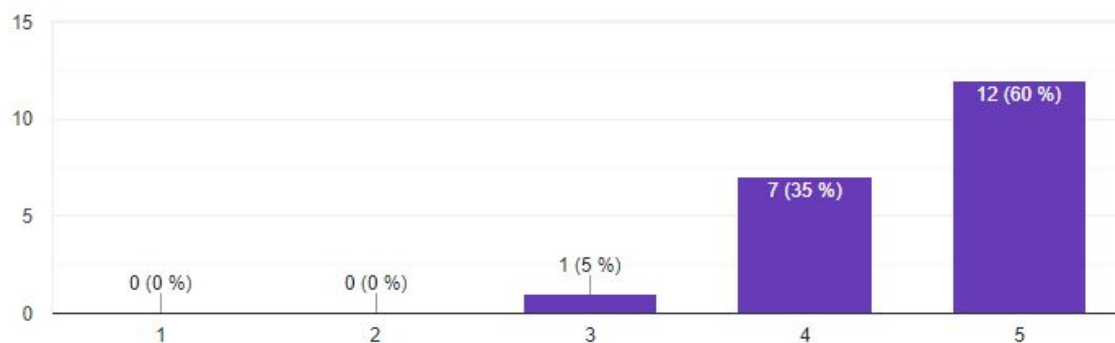


Figura 107. Resultados de añadir nueva sección.

8.2.18. EVALUACIÓN “GUARDAR CAMBIOS DE SECCIONES”

En el caso de guardar los cambios de las secciones, un 65% de los encuestados le ha otorgado la máxima nota de un 5 mientras que el otro 35% lo ha puntuado con un 4 (véase figura 108).

Guardar los cambios de las secciones:

20 respuestas

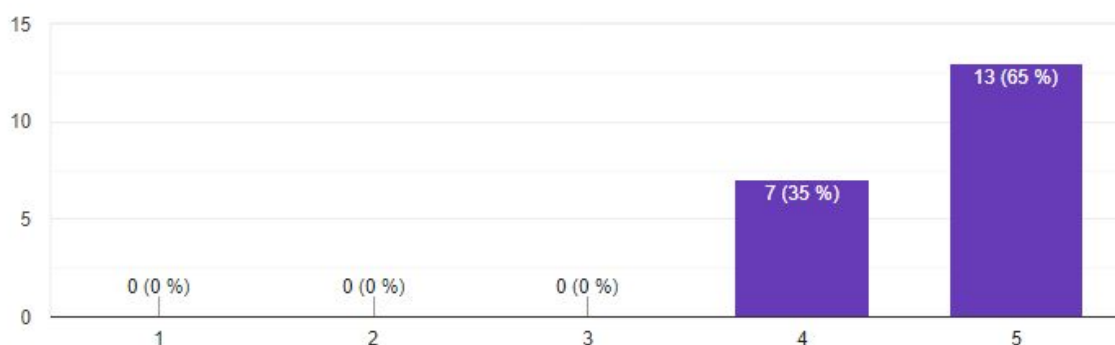


Figura 108. Resultados de guardar los cambios de las secciones.

8.2.19. EVALUACIÓN “DESPLAZAMIENTO ENTRE SECCIONES”

Respecto al desplazamiento o cambio entre secciones, un 75% ha calificado la tarea con un 5, un 20% con un 4 y el último 5% con un 3 (véase figura 109).

Desplazarte entre secciones (cambiar de una sección a otra):

20 respuestas

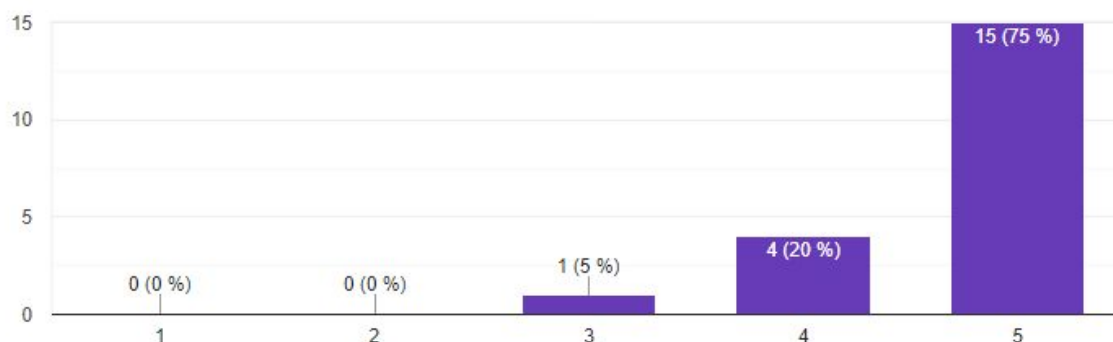


Figura 109. Resultados de desplazarse entre secciones.

8.2.20. EVALUACIÓN “AÑADIR GADGETS Y CONFIGURARLOS”

Añadir uno de los gadgets especificado en la tarea 18 (video, imagen, hipervínculo, texto expandible, anotación o spoiler) y configurarlo ha recibido una nota de 5 por parte del 50% de los encuestados, una nota de 4 por parte del 35%, el 10% lo ha calificado con un 3 y el último 5% con un 2 (véase figura 110).

Añadir uno de los gadgets especificado en la tarea 18 y configurarlo:

20 respuestas

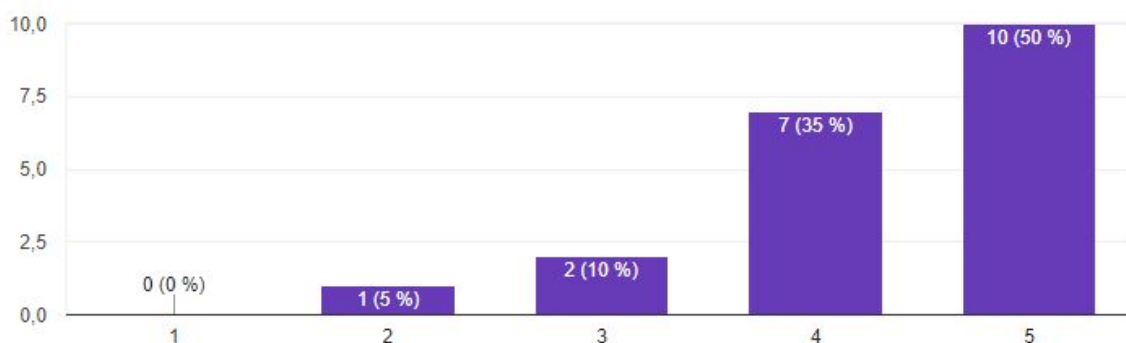


Figura 110. Resultados de añadir un gadget y configurarlo.

8.2.21. EVALUACIÓN “PREVISUALIZAR GADGET”

La tarea de previsualizar un gadget le ha resultado muy sencillo e intuitivo al 45% de los encuestados, al 30% simplemente sencillo, al 20% no muy sencillo pero sin mucha dificultad para realizarla y al otro 5% poco intuitivo (véase figura 111).

Previsualizar un gadget:

20 respuestas

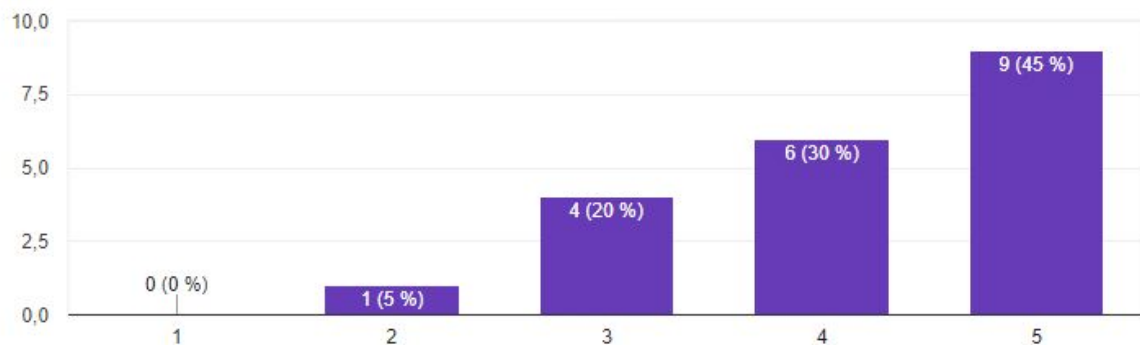


Figura 111. Resultados de previsualizar gadget.

8.2.22. EVALUACIÓN “GADGET SIGUIENTE SECCIÓN”

En cuanto a la tarea de añadir un gadget de siguiente sección y configurarlo, a un 65% le ha resultado muy sencillo e intuitivo con una nota de 5, a un 30% sencillo e intuitivo con una nota de 4 y al último 5% no muy sencillo pero sin mucha dificultad para realizar la tarea, otorgándole una nota de 3 (véase figura 112).

Añadir un gadget de siguiente sección y configurarlo correctamente:

20 respuestas

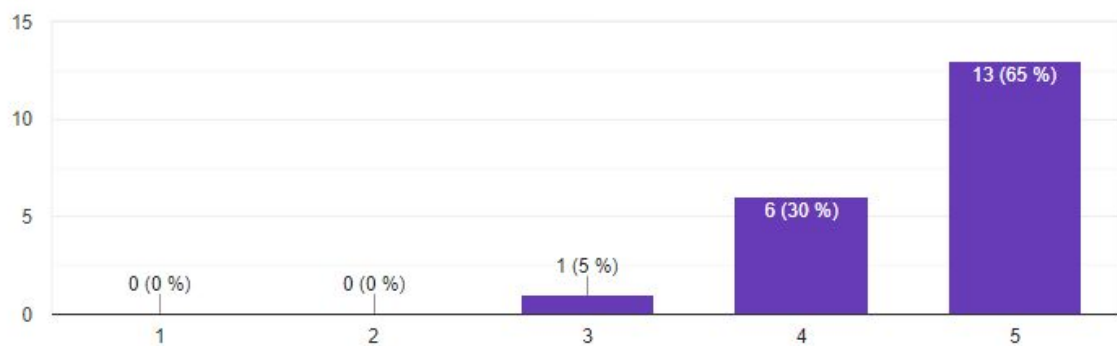


Figura 112. Resultados de añadir un gadget de siguiente sección y configurarlo correctamente.

8.2.23. EVALUACIÓN “PREVISUALIZAR LIBRO”

Respecto a previsualizar el libro completo, un 75% de los encuestados ha puntuado la tarea con una nota de 5, un 15% con un 4, un 5% con un 3 y el otro 5% con la mínima nota de 1 (véase figura 113).

Previsualizar el libro completo:

20 respuestas

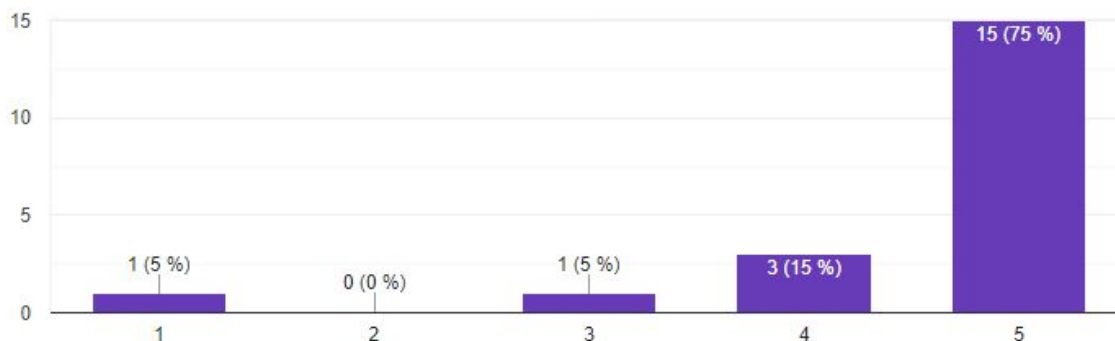


Figura 113. Resultados de previsualizar el libro completo.

8.2.24. EVALUACIÓN “VOLVER A EDICIÓN DE LIBRO”

Volver a la edición desde la previsualización le ha resultado muy sencillo e intuitivo al 70% de los encuestados, al 15% sencillo sin ser totalmente claro y al otro 15% no muy intuitivo pero tampoco muy difícil (véase figura 114).

Volver a la edición desde la previsualización:

20 respuestas

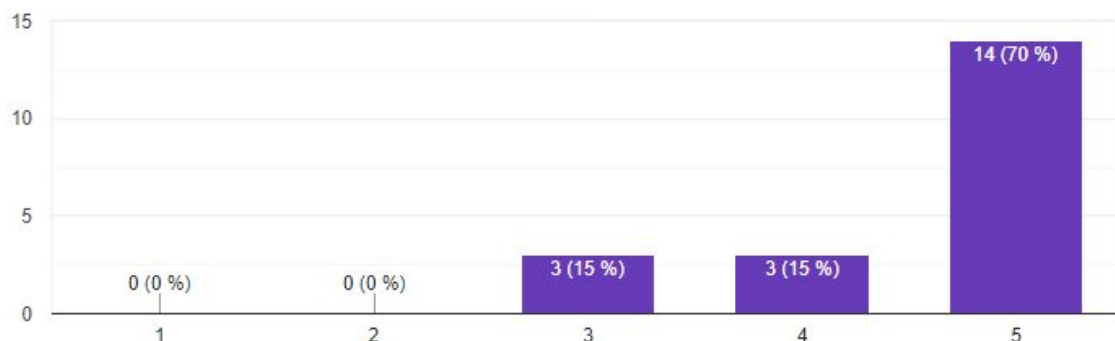


Figura 114. Resultados de volver a la edición desde la previsualización.

8.2.25. EVALUACIÓN “MOVER GADGETS”

A la tarea de mover gadgets de posición, un 50% le ha otorgado una nota de 5, un 25% una nota de 4, un 20% de 3 y el último 5% de 1 (véase figura 115).

Mover los gadgets:

20 respuestas

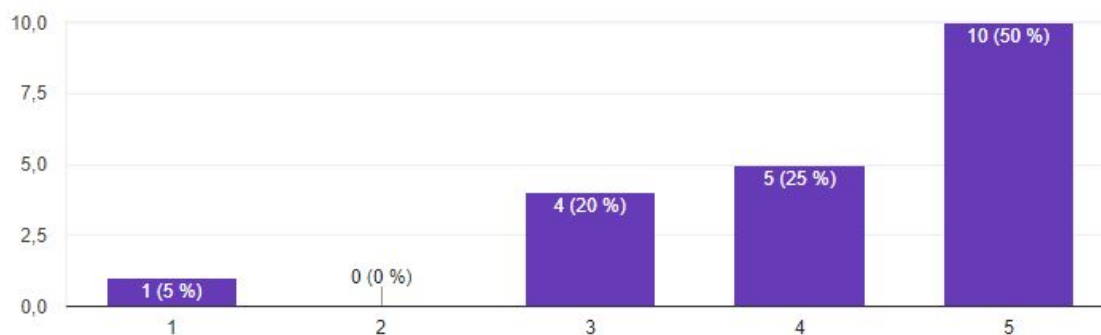


Figura 115. Resultados de mover los gadgets.

8.2.26. EVALUACIÓN “COPIAR GADGETS”

Copiar o clonar un gadget le ha resultado muy sencillo e intuitivo al 65% con una nota de 5, el 25% lo ha calificado con una nota de 4, significando que la resolución de la tarea era sencilla pero no óptima, un 5% con una nota de 3 y el 5% restante opina que ha sido poco sencillo o intuitivo con una nota de 2 (véase figura 116).

Copiar los gadgets:

20 respuestas

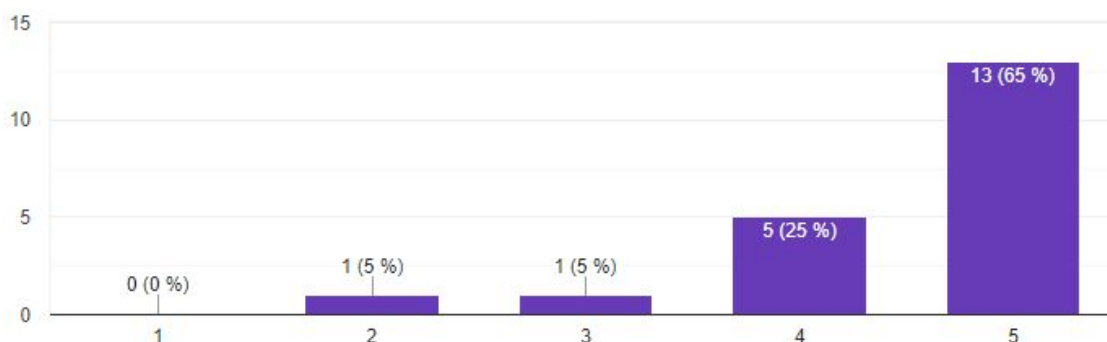


Figura 116. Resultados de copiar los gadgets.

8.2.27. EVALUACIÓN “ELIMINAR GADGETS”

Eliminar un gadget le ha parecido muy sencillo e intuitivo, con una nota de 5, al 80% de los encuestados. El 20% restante opina que ha sido sencillo e intuitivo, con una nota de 4 (véase figura 117).

Eliminar los gadgets:

20 respuestas

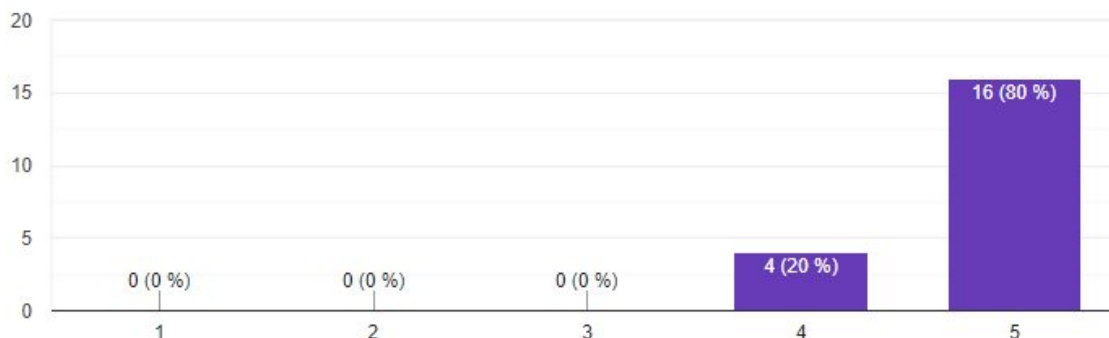


Figura 117. Resultados de eliminar los gadgets.

8.2.28. EVALUACIÓN “COPIAR SECCIÓN”

A la tarea de copiar una sección el 75% de los encuestados la ha calificado con una nota de 5 y el 25% restante una nota de un 4 (véase figura 118).

Copiar una sección:

20 respuestas

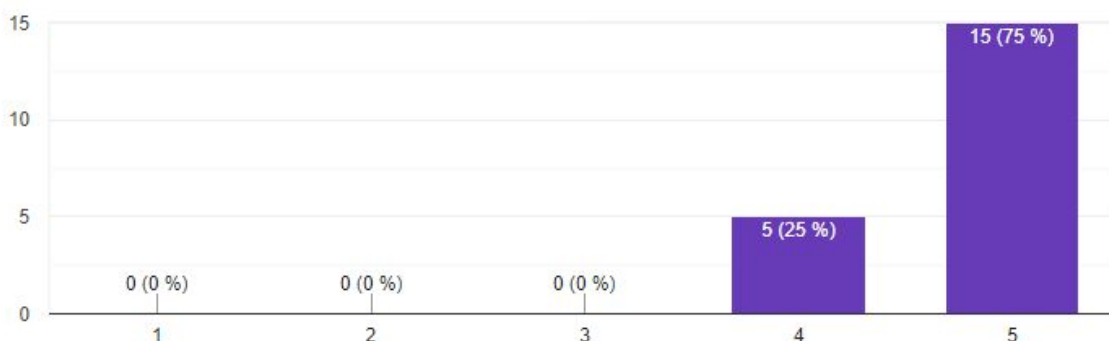


Figura 118. Resultados de copiar una sección.

8.2.29. EVALUACIÓN “INTERACTUAR CON GADGETS”

En el caso de interactuar con los distintos gadgets disponibles un 40% ha otorgado la máxima nota, opinando que es muy sencillo e intuitivo, un 55% lo ha evaluado con un 4 y un 5% ha elegido puntuarlo con un 3, significando que no es muy claro, pero que se puede llegar a entender sin mucha dificultad (véase figura 119).

En general, interactuar con los distintos gadgets disponibles:

20 respuestas

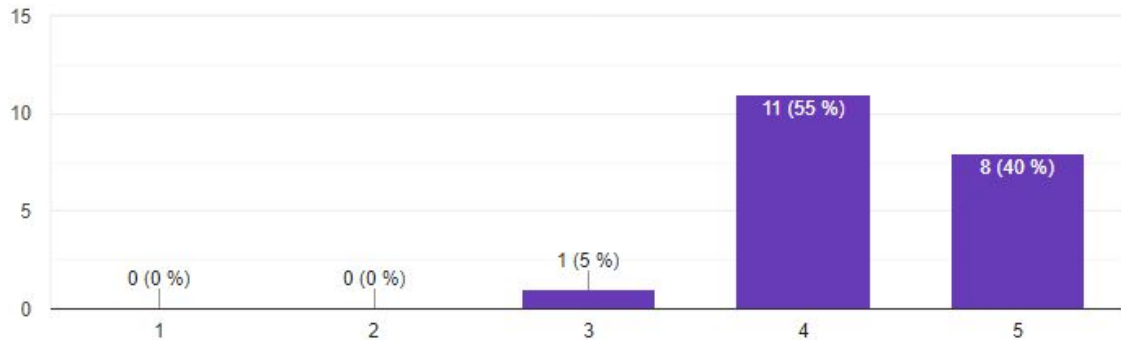


Figura 119. Valoración de la interacción con los distintos gadgets.

8.2.30. EVALUACIÓN “GUARDAR LIBRO”

Guardar el libro le ha resultado muy sencillo e intuitivo al 85% de los encuestados, al 10% sencillo e intuitivo y al último 5% muy poco sencillo o intuitivo (véase figura 120).

Guardar el libro:

20 respuestas

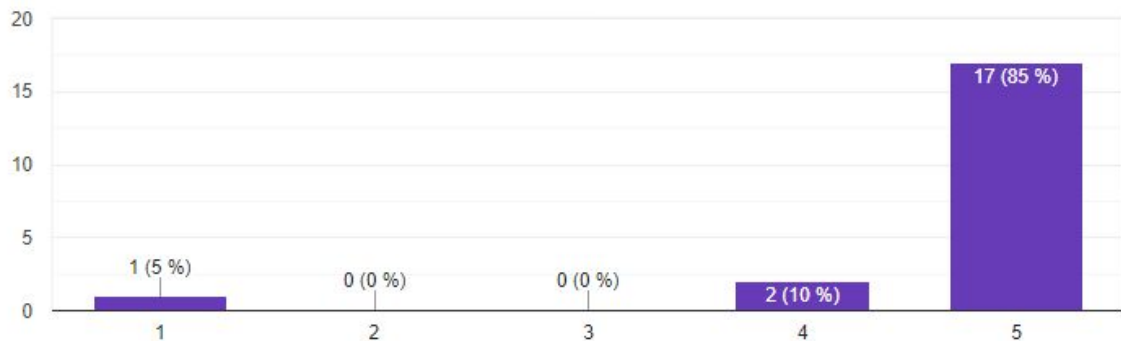


Figura 120. Valoración de guardar el libro.

8.2.31. EVALUACIÓN “PUBLICAR LIBRO”

Respecto a publicar un libro, un 60% de los encuestados lo ha calificado con un 5, un 20% con un 4, un 5% con un 3, un 10% con un 2 y el último 5% con un 1 (véase figura 121).

Publicar el libro:

20 respuestas

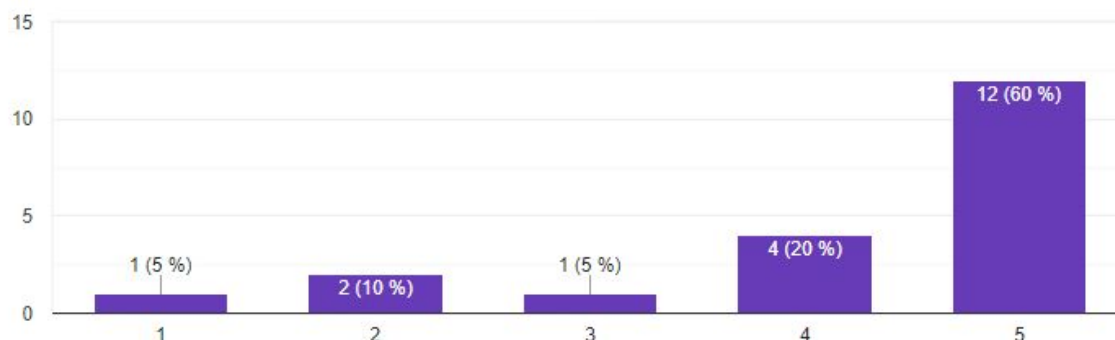


Figura 121. Valoración de publicar libro.

8.2.32. EVALUACIÓN “LEER LIBRO PUBLICADO”

Leer el libro publicado le ha resultado muy sencillo e intuitivo al 85% de los encuestados, habiendo otorgado una nota de 5. Al 15% restante le ha parecido sencillo e intuitivo, con una nota de 4 (véase figura 122).

Leer el libro publicado:

20 respuestas

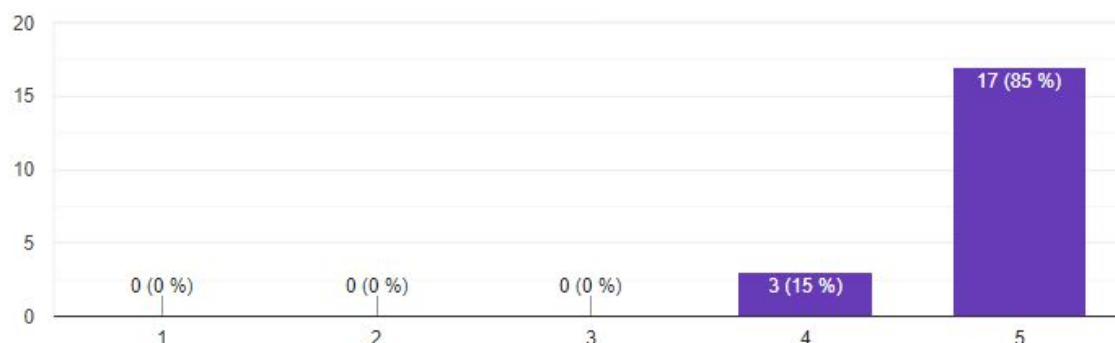


Figura 122. Valoración de leer libro publicado.

8.2.33. EVALUACIÓN “ELIMINAR LIBRO”

Al 95% de los encuestados les ha parecido muy sencillo e intuitivo eliminar un libro con una nota de 5, mientras que al otro 5% no le ha parecido ni muy sencillo ni muy poco intuitivo, otorgando una nota de 3 (véase figura 123).

Eliminar un libro:

20 respuestas

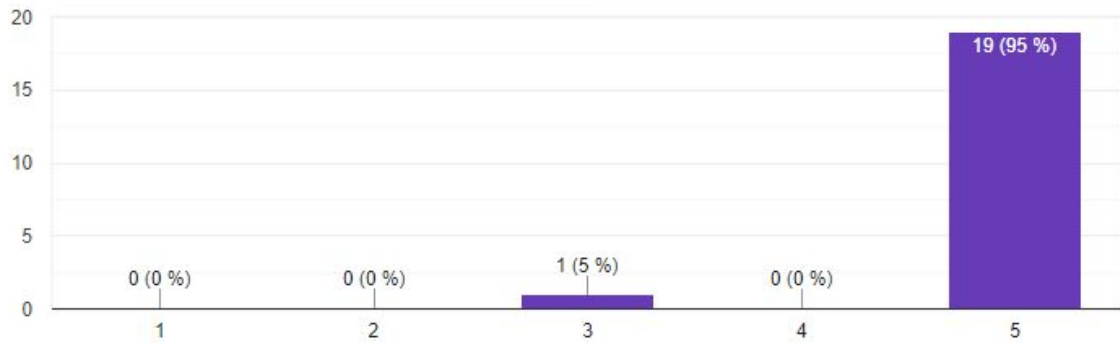


Figura 123. Valoración de eliminar libro.

9. CAPÍTULO IX: CONCLUSIONES Y TRABAJO FUTURO

En este capítulo se recogen las conclusiones obtenidas a lo largo del proyecto y posibles mejoras futuras para la aplicación.

9.1. CONCLUSIONES

En este trabajo se ha desarrollado una aplicación web para la escritura y la lectura de libros interactivos. Para dotar a los libros de este aspecto interactivo, de una manera sencilla para el usuario, se han creado una serie de ‘plantillas’ de estos elementos interactivos llamados gadgets. La aplicación contiene 17 tipos de gadgets distintos, cada uno tiene unos campos propios con los valores a rellenar por el usuario. Además, los datos de todos los libros, las secciones y los gadgets se encuentran almacenados online en una base de datos modelada con Firebase.

Teniendo presente el amplio público al que va dirigida esta aplicación, se ha diseñado la interfaz de una manera simplificada reduciendo el número de elementos accesibles por los usuarios en cada vista y procurando mostrar las opciones disponibles de la manera más intuitiva posible.

Se ha garantizado que los usuarios puedan leer todos los libros publicados, independientemente del autor, y que, si el usuario desea organizar los libros, pueda hacerlo usando bibliotecas.

9.2. TRABAJO FUTURO

Las principales líneas de trabajo futuro son las siguientes:

- **Adaptación a dispositivos móviles:** Adaptar la interfaz de la aplicación web para sistemas operativos móviles (Android e IOs, tanto móviles como tabletas).
- **Libros colaborativos:** Permitir que más de un usuario pueda editar el mismo libro.
- **Componentes sociales:** Añadir elementos como: seguir usuarios, hablar con otras personas, comentar libros, darlos una valoración, valorar comentarios...
- **Mejoras a bibliotecas:** Fusionar bibliotecas, insertar una biblioteca en otra (y que la otra siga existiendo. La biblioteca a insertar puede ser de otro usuario)...
- **Guardar sección:** Se guardará la sección donde cada usuario se ha quedado leyendo un libro, con los gadgets y sus valores incluidos.
- **Administrador:** Añadir para este actor, funcionalidades relacionadas con la gestión de los usuarios y las publicaciones desde su rol.
- **Barra de búsqueda:** Añadir a la aplicación la posibilidad de realizar una búsqueda en la aplicación, ya sea de usuarios, libros o bibliotecas.
- **Gadgets personalizados:** Crear una ventana llamada favoritos en la que el autor pueda agregar, eliminar e insertar en el libro gadgets ya configurados por él.
- **Descargar libro:** Añadir la funcionalidad de descargar uno de los libros escritos para poder leer en otros dispositivos.

- **Modo de visualización en bloques:** Implementar una visualización del progreso tanto en la edición como en lectura del libro donde se muestren las bifurcaciones entre las secciones usando bloques conectados por líneas.
- **Ampliación de gadgets:**
 - **Texto poco a poco:** Un texto definido por el autor se vaya mostrando poco a poco durante la lectura.
 - **Reconstrucción imagen:** Dar una imagen en trozos desordenados para que el lector la ordene y reconstruya.
 - **Puzzle ordenar frase:** Se da una frase desordenada y el lector debe ordenarla para continuar con la lectura.
 - **Sonido frase:** Agregar audios personalizados para la lectura de una frase.
 - **Imagen interactiva:** Se da una imagen y dependiendo de donde clique el lector continúa la historia por un camino u otro.
 - **Tres en raya:** Un juego de tres en raya contra la máquina.
 - **Condicionales:** Dependiendo de decisiones pasadas que se hayan tomado, se continúa la historia por distintas direcciones.

9. CHAPTER IX: CONCLUSIONS AND FUTURE WORK

In this chapter there is a summary of the conclusions obtained through the development of the project and possible improvements for application to do in the future.

9.1. CONCLUSIONES

A web application for the reading and writing of interactive books has been developed for this project. To give the books this interactive function in a simple way, a series of templates have been created called gadgets. The application has 17 different gadgets, each one with their own values and fields to fill in by the user.

Having in account the wide range of people the project is addressed to, the interface has been designed in a simplified manner, reducing the number of elements accessible by the users in each view and trying to show all the available options in a very intuitive way.

It has been guaranteed that users can read all published books and that they can organize them using libraries.

9.2. TRABAJO FUTURO

The main possible improvements to develop in the future are as follows:

- **Adapt the application to view it from mobile devices:** Adapt the application interface so that it can be accessed from mobile devices, Android and IOs.
- **Co-op books:** Allow more than one user to edit the same book.
- **Social components:** Add elements like: follow users, talk to other people, add comments to books, rate books or rate comments.
- **Improvement to libraries:** Fusion libraries, insert a library in another one so that the other one keeps existing, another user libraries can be inserted.
- **Save section:** When a user stops reading a book, save the section they stopped reading so that they can continue from there next time they open the book.
- **Admin:** Add functionality to this actor related to the managements of users or publishing of books.
- **Search bar:** Add the possibility to make a search in the application, be it of users, books or libraries.
- **Custom gadgets:** Add a panel where the writer can add already configured gadgets so that he can use them whenever the need arises.
- **Download book:** Add the functionality to download a published book to read it from other devices.
- **Visualization in blocks:** Implement a visual interface both when editing and reading that shows the bifurcations between sections showing them in blocks with lines.
- **Extra gadgets:**
 - **Show text slowly:** A text written by the writer, appears slowly as the reading goes on.

- **Reconstruction of a picture:** The reader is given a picture split in pieces and he has to build the original one with them.
- **Puzzle of sorting a sentence:** The reader is given a sentence in the wrong order and he has to order it correctly..
- **Sentence with sound:** Add customized voices for the reading of a particular sentence.
- **Interactive picture:** Given a picture, depending where the reader clicks, it will take you one way or another.
- **Tic tac toe:** A game of tic tac toe against the machine.
- **Conditions:** Depending of past decisions the reader made, the book takes you one way or another.

10. CAPÍTULO X: APORTACIONES INDIVIDUALES

En este capítulo se describe el trabajo individual realizado por cada integrante del grupo.

10.1. ANDREA MARTÍNEZ FERNÁNDEZ

Las aportaciones de Andrea Martínez Fernández se centran en dos grupos: contribuciones relacionadas con la escritura de la memoria y contribuciones relacionadas con el desarrollo de la aplicación. A continuación se describen en detalle cada una de estas:

a. Contribuciones en la escritura de la memoria

La memoria se ha realizado y revisado en conjunto por los tres integrantes del equipo, aunque algún capítulo ha sido desarrollado de manera individual.

En el caso de Andrea, ha realizado los siguientes apartados:

- Resumen: explicación de los objetivos del proyecto y de las funcionalidades a desarrollar.
- Estado del arte: estudio de herramientas similares de uso abierto para el público. En este apartado se distinguen dos grupos, las herramientas enfocadas a la escritura y lectura de libros con un enfoque más social y las herramientas que soportan componentes interactivos.
- Diseño de la aplicación: descripción detallada de cada una de las funcionalidades de la aplicación, comentando los aspectos destacados de las vistas y el funcionamiento interno de la aplicación como las interacciones con la base de datos.
- Conclusión: explicación del trabajo realizado, herramientas utilizadas y tecnologías aplicadas.

Algunos capítulos de la memoria como los casos de uso, se llevaron a cabo entre los tres integrantes del grupo tras debatir cómo iba a estar estructurada la aplicación. Otros apartados realizados de esta manera fueron la introducción y el trabajo futuro.

b. Contribuciones en el desarrollo de la aplicación

A diferencia de en la memoria, el trabajo desarrollado en la aplicación ha sido más diferenciado, ya que cada uno se encargó de realizar distintos casos de uso o tareas.

A continuación se enuncian y desarrollan las aportaciones realizadas por Andrea:

Colección de libros en la base de datos

La colección de libros guarda los datos del libro como son la portada y el título entre otros. Esta colección también se relaciona con los usuarios almacenando en cada libro el identificador del usuario propietario.

Desarrollo del módulo de libros

El módulo de gestión de libros abarca las funcionalidades de crear, modificar, mostrar y eliminar libro. Dentro de la creación de libro se incluye la restricción de que dos libros no puedan tener el mismo nombre y se pueden configurar elementos como etiquetas identificativas o una descripción. En el caso de modificar, se pueden cambiar los valores de los elementos introducidos en la creación del libro, así como agregar una portada, la cual se almacena en el storage de firebase.

Storage de firebase

Investigación y primera aproximación a la posibilidad de agregar multimedia a los elementos de la aplicación almacenando estos datos en el storage que proporciona firebase, simplificando su acceso con un enlace.

Gadgets

Los gadgets disponibles a disposición del autor para editar un libro han sido realizados por los tres integrantes del equipo, en el caso de Andrea se ha encargado principalmente de los siguientes:

- Texto expandible: este gadget consiste en un bloque de texto visible en la lectura y otro bloque de texto oculto que se muestra al pulsar sobre el anterior, desplegándose debajo del texto visible.
- Anotaciones: gadget que añade a un bloque de texto la posibilidad de mostrar más información en una ventana emergente al pulsar sobre él, el cual puede contener texto y una imagen.
- Hipervínculos: texto asociado a un enlace web de manera que si se pulsa sobre él abre una nueva ventana con esa dirección.

Diseño de la interfaz

Todo lo relacionado con el diseño de la interfaz de la aplicación web. Esto incluye la decisión de utilizar vue bootstrap como framework de diseño, el diseño principal de todos los botones, pantallas, fuentes del texto o iconos utilizados, la colocación de los distintos elementos dentro de la misma pantalla o configurar manualmente parámetros con css.

Todo esto con el objetivo de obtener un resultado homogéneo entre toda la aplicación, de manera que facilite al usuario familiarizarse con la aplicación y no haya grandes contradicciones ni incongruencias en el diseño.

Despliegue de la aplicación

La investigación de la posibilidad de subir el proyecto a github pages así como su despliegue en la misma generando un enlace url a través del cual la aplicación web está accesible con todas sus funcionalidades.

10.2. ANDREA DEL VADO PUELL

Las aportaciones de Andrea del Vado Puell se pueden separar en dos grupos: contribuciones relacionadas con la escritura de la memoria y contribuciones relacionadas con el desarrollo de la aplicación. A continuación se describen en detalle cada una de estas:

a. Contribuciones en la escritura de la memoria

La memoria se ha realizado y revisado en conjunto por los tres integrantes del equipo, aunque algún capítulo ha sido desarrollado de manera individual.

En el caso de Andrea, ha realizado los siguientes apartados:

- Traducción al inglés: traducción al inglés de los distintos apartados exigidos por la normativa, que incluye la introducción, el resumen, las conclusiones y trabajo futuro.
- Evaluación de usuarios: tras obtener los resultados de la encuesta realizada por distintos usuarios, comentar los diferentes datos obtenidos en las diferentes preguntas propuestas.
- Guía de usuario: para la correcta realización de la evaluación de usuarios, se envió, junto con un listado de tareas, una guía de usuario con información del funcionamiento de la aplicación, en la que se muestra con detalle la finalidad de cada elemento existente en ella.

Algunos capítulos de la memoria como los casos de uso se llevaron a cabo entre los tres integrantes del grupo tras debatir cómo iba a estar estructurada la aplicación. Otros apartados realizados de esta manera fueron la introducción y el trabajo futuro.

b. Contribuciones en el desarrollo de la aplicación

A diferencia de en la memoria, el trabajo desarrollado en la aplicación ha sido más diferenciado, ya que cada uno se encargó de realizar distintos casos de uso o tareas.

A continuación se enuncian y desarrollan las aportaciones realizadas por Andrea:

Crear el proyecto

Andrea se encargó de la creación del proyecto y repositorio de github para así poder todos los integrantes del grupo trabajar desde la misma versión y mantenerse actualizados del estado de la aplicación.

Establecer la conexión con firebase

Para poder acceder a la base de datos, tanto para subida como bajada de datos, se necesita establecer primero una conexión con ella, una vez establecida basta con realizar distintas peticiones desde cualquier parte del proyecto.

Persistencia de los datos

La persistencia de datos se realizó con el store de Vuex, donde se almacenan los distintos datos que se necesitan mantener dentro de la aplicación, como el usuario con sesión iniciada, libro que se está leyendo ahora mismo o la sección en la que se encuentra.

Desarrollo de módulo de usuarios

Todo lo relacionado con los casos de usuarios de este módulo, el cual incluye la creación, modificación, eliminación y posibilidad de mostrar un usuario y lo relacionado con la sesión, iniciar sesión y realizar sesión. Dentro de este módulo se incluyen algunos detalles como comprobaciones de que el nick se tratara de una clave única, eliminar un usuario elimine todos los libros escritos por él, así como sus bibliotecas o que se mantenga la sesión iniciada durante se utiliza la aplicación.

Previsualización de libro desde edición

Dentro de la edición de libro existe la posibilidad de consultar en formato lectura lo que se lleva escrito, así como interactuar con todos los elementos añadidos hasta el momento. A esta función se le llama previsualización del libro.

Vista de lectura de libro

El código relacionado con la vista de lectura de un libro, su estructura interna así como la manera básica de mostrar los distintos gadgets en ese formato, encargándose de los detalles el miembro que realizó cada gadget en particular.

Gadgets

Los gadgets disponibles a disposición del autor para editar un libro han sido realizados por los tres integrantes del equipo, en el caso de Andrea se ha encargado principalmente de los siguientes:

- **Adivinanzas:** El autor puede escribir una adivinanza y su respuesta, configurando el número máximo de intentos y la consecuencia de adivinarla correctamente o equivocarse en todos los intentos, permitiendo cambiar de sección o mantenerse en la sección actual leyendo.
- **Imagen:** El gadget relacionado con subir una imagen al libro para mostrarse entre distintos gadgets.
- **Video:** El gadget de video permite subir un vídeo y elegir si se desea que se auto reproduzca al acceder a esa sección, si se desea que se reproduzca en bucle o ambas.
- **Tarjetas de memoria:** El puzzle de destapar todas las parejas en un número máximo de movimientos, así como su personalización en cuanto a mostrar colores o palabras. También la configuración del resultado de resolver satisfactoriamente el puzzle o quedarse sin movimientos, ambos relacionados con cambios de sección.

10.3. JAVIER SANZ CASTRO

Las aportaciones de Javier Sanz Castro se centran en dos grupos: contribuciones relacionadas con la escritura de la memoria y contribuciones relacionadas con el desarrollo de la aplicación. A continuación se describen en detalle cada una de estas:

a. Contribuciones en la escritura de la memoria

La memoria se ha realizado y revisado en conjunto por los tres integrantes del equipo, aunque algún capítulo ha sido desarrollado de manera individual.

En el caso de Javier, ha realizado los siguientes apartados:

- Modelo de datos: Tanto la estructura del modelo entidad-relación como la descripción de cada campo de la base de datos, incidiendo en cada campo de cada gadget.
- Arquitectura de la aplicación: Como resultado de investigar el funcionamiento de Vue y su estructura interna para poder aprovechar todas las ventajas que ofrece en el desarrollo.
- Tecnología empleada: La tecnología empleada durante el desarrollo de la aplicación. Tanto la empleada para el desarrollo propiamente dicho como la empleada para que el equipo se apoye y organice.

Algunos capítulos de la memoria, como los casos de uso, se llevaron a cabo entre los tres integrantes del grupo tras debatir cómo iba a estar estructurada la aplicación. Otros apartados realizados de esta manera fueron la introducción y el trabajo futuro.

b. Contribuciones en el desarrollo de la aplicación

A diferencia de en la memoria, el trabajo desarrollado en la aplicación ha sido más diferenciado, ya que cada uno se encargó de realizar distintos casos de uso o tareas.

A continuación se enuncian y desarrollan las aportaciones realizadas por Javier:

Módulo gestión de secciones

El módulo de gestión de secciones completo. Al tratarse de uno de los módulos más importantes de la aplicación, se ha asegurado su correcto funcionamiento tanto para evitar bugs como para garantizar que su interacción con el resto de gadgets -prestando especial atención al gadget de casilla personalizada- funcionen correctamente y no suceda ningún fallo que pueda destruir el trabajo de los usuarios.

También se garantiza que el cambio de secciones en la vista de los libros funcione correctamente y se actualice automáticamente cuando el usuario modifica los valores de las secciones.

Además, y con el fin de ahorrar recursos del usuario que ejecute la aplicación, solo se cargan las secciones que se muestran en pantalla.

Editar libro

Toda la estructura acerca de la edición de un libro, tanto en el tratamiento de los gadgets como de las secciones. De los gadgets, se ha implementado su funcionamiento de manera general en el cargado, la modificación y el visionado -aprovechándose de la estructura en forma de componentes de Vue-. De las secciones, se ha implementado el guardado y la descarga de secciones hacia y desde la base de datos. Se ha trabajado en cómo la vista de editar libro permite que el usuario interactúe con la aplicación y qué efectos produce internamente para que todas las operaciones realizadas sobre los gadgets funcionen sin errores.

Módulo de gestión de gadgets

Comprende la gran mayoría de los aspectos pertenecientes a la figura 5 del módulo de gestión de gadgets, con el fin de garantizar que se puedan añadir -prestando atención a las posiciones donde se añaden-, borrar, duplicar, mover y, sobre todo, editar, sin problemas y de una forma interactiva para el usuario que sea fácilmente entendible. Duplicar, además, garantiza que la copia no sea simplemente una referencia al original (JavaScript crea referencias si no se pone cuidado en la forma de hacerlo), sino que sea un objeto completamente independiente y editable por separado.

Módulo de gadgets

Los gadgets disponibles a disposición del autor para editar un libro han sido realizados por los tres integrantes del equipo, en el caso de Javier, se ha encargado principalmente de los siguientes:

- Casilla personalizada: Se ha realizado el gadget de casilla personalizada al completo, tanto en lectura como en escritura, para que sea fácil de entender para el usuario. Además, se han implementado los tres modos de dicho gadget, poniendo énfasis en que sus datos se almacenen correctamente en la base de datos, ya que su funcionamiento es especial y para este gadget se debe trabajar tanto con las colecciones de secciones como de los libros.
- Texto normal: Se trata de un gadget con una estructura interna distinta al resto y que consta de las distintas propiedades de este tipo de textos en otros editores, como la negrita, la cursiva, el subrayado o el tachado. Además, se garantiza que la interfaz se actualice de manera intuitiva con cada movimiento que el usuario realice sobre este tipo de gadget, actualizando los botones en conveniencia.
- Títulos: Corresponde al gadget del título. Como existen tres tipos de títulos, se puede cambiar entre ellos de una forma sencilla e intuitiva apoyándose en la interfaz, que se actualiza marcando el tipo de título que el usuario haya elegido.

- **Números aleatorios:** El gadget de números aleatorios, en el que se garantiza que el usuario obtenga el número entre los intervalos que él desee y que las condiciones que dicho usuario pueda agregar para saltar entre secciones según se cumplan una serie de criterios funcionen correctamente.
- **Tomas de decisiones:** El gadget de toma de decisiones. Se ha implementado que el usuario pueda cambiar de secciones pulsando una sola vez y además permitiendo insertar mensajes personalizados en cada opción, con el fin de resultar más intuitivo.
- **Completar la secuencia:** Corresponde al gadget que permite completar la secuencia. El usuario puede definir la secuencia como él quiera, así como sus soluciones, y se garantiza que cuando se quiera dar saltos entre secciones, estos funcionen correctamente.
- **Pistas:** El gadget de pistas, que permite al usuario resolver una serie de pistas definidas por el autor. Se garantiza su correcto funcionamiento tanto si se cambia de sección como si no, y de que sea intuitivo y sencillo de usar para el usuario.
- **Cambio de sección:** El gadget de cambiar sección para que cada vez que el usuario pulsa, se cargue otra, ha sido realizado por él. Se garantiza además que si un usuario accede a una sección no válida en la lectura, se muestre un mensaje de error para evitar que la aplicación falle o de actualizar los valores automáticamente en la edición si no son válidos.
- **Repetir sección:** Corresponde al gadget de repetir sección que es usado para recargar una sección y así regenerar los gadgets en caso de que se quedaran bloqueados por acabar con su límite de intentos.
- **Spoiler:** Se ha encargado del gadget de spoiler para que el usuario pueda ocultar el contenido que quiera.

Módulo de gestión de bibliotecas

El módulo de gestión de bibliotecas completo, con todos los casos de uso incluidos, así como la gestión de dichas bibliotecas para garantizar su correcto funcionamiento y evitar, en la medida de lo posible, todos los errores que pudieran surgir. Esto también incluye la gestión de conexión por parte del usuario.

Además, aunque no figura en este módulo, se ha encargado de que los libros se inserten correctamente en las bibliotecas desde las vistas que los contienen y de que al pulsar sobre cualquier libro para acceder a él, este se guarde correctamente en el historial del usuario.

También se ha realizado la distinción entre las bibliotecas normales creadas por el usuario como la generada por el sistema para cada uno, teniendo en cuenta que sus características son distintas y, por tanto, la forma en la que funcionan y trabajan también, llegando a tener una vista distinta en determinados aspectos respecto de las normales.

Colecciones en la base de datos

Tanto la colección de bibliotecas como la colección de secciones, con todos sus campos y su forma de almacenarlos. En la colección de bibliotecas se almacena la id del usuario para identificar al propietario y en la de secciones tanto el libro como el autor.

BIBLIOGRAFÍA

[1] «Wattpadd». [En línea]. Disponible en:

<https://www.wattpad.com/home>

[2] «Me gusta escribir». [En línea]. Disponible en:

<http://megustaescribir.com/login>

[3] «Twine». [En línea]. Disponible en:

<https://twinery.org/>

[4] «Vue». [En línea]. Disponible en:

<https://es.vuejs.org/v2/guide/>

[5] «Github pages». [En línea]. Disponible en:

<https://dev.to/rolanddoda/deploy-to-github-pages-like-a-pro-with-github-actions-4hdg>

[6] «YOU, Evan. What is Vuex. URL *https://vuex.vuejs.org*, 2016.». [En línea]. Disponible en:

<https://vuex.vuejs.org/>

[7] «Vue cli». [En línea]. Disponible en:

<https://cli.vuejs.org/guide/>

[8] «Bootstrap». [En línea]. Disponible en:

<https://getbootstrap.com/>

[9] «Bootstrap Vue». [En línea]. Disponible en:

[https://en.wikipedia.org/wiki/Bootstrap_\(front-end_framework\)](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework))

[10] «Vue». [En línea]. Disponible en:

<https://bootstrap-vue.org/docs>

[11] «Babel eslint». [En línea]. Disponible en:

<https://eslint.org/docs/about/>

[12] «EsLint». [En línea]. Disponible en:

<https://antoniomasia.com/que-es-eslint-y-por-que-deberias-usarlo/>

[13] «Firebase». [En línea]. Disponible en:

<https://www.imaginaformacion.com/tutorial/que-es-firebase/>

[14] «Npm». [En línea]. Disponible en:

[https://en.wikipedia.org/wiki/Npm_\(software\)](https://en.wikipedia.org/wiki/Npm_(software))

[15] «GitHub». [En línea]. Disponible en:

<https://github.com/features>

[16] «Git». [En línea]. Disponible en:

<https://git-scm.com/about>

[17] «VSCoDe». [En línea]. Disponible en:

https://en.wikipedia.org/wiki/Visual_Studio_Code

[18] «Sublime». [En línea]. Disponible en:

<https://www.sublimetext.com/>

[19] «Google Drive». [En línea]. Disponible en:

https://www.google.com/intl/es_ALL/drive/using-drive/

[20] «Google Docs». [En línea]. Disponible en:

https://gsuite.google.com/products/docs/?utm_source=docsforwork&utm_medium=et&utm_content=learnmore&hl=es

[21] «Trello Johnson, H. A. (2017). Trello.» *Journal of the Medical Library Association: JMLA*, 105(2), 209. [En línea]. Disponible en: <https://jmla.pitt.edu/ojs/jmla/article/view/49/375>

[22] «Cliente-servidor». [En línea]. Disponible en:

<https://es.wikipedia.org/wiki/Cliente-servidor>

[23] «Vue-fire». [En línea]. Disponible en:

<https://vuefire.vuejs.org/>

[24] «MVVM». [En línea]. Disponible en:

<https://www.codeproject.com/articles/100175/model-view-viewmodel-mvvm-explained>

[25] «Instancia Vue». [En línea]. Disponible en:

<https://es.vuejs.org/v2/guide/instance.html>

[26] «Vue 0.12». [En línea]. Disponible en:

<https://012.vuejs.org/guide/>

[27] «Casa del libro». [En línea]. Disponible en:

<https://www.casadellibro.com/>

[28] «fnac». [En línea]. Disponible en:

<https://www.fnac.es/home/book.aspx#bl=MMLibros>

[29] «google forms». [En línea]. Disponible en:

https://gsuite.google.com/products/forms/?utm_source=formsforwork&utm_medium=et&utm_content=learnmore&hl=es

- [30] «Quest». [En línea]. Disponible en:
<http://textadventures.co.uk/quest>
- [31] «Librojuego». [En línea]. Disponible en:
<https://es.wikipedia.org/wiki/Librojuego>
- [32] «Router». [En línea]. Disponible en:
<https://router.vuejs.org/>
- [33] «Vuefire». [En línea]. Disponible en:
<https://vuefire.vuejs.org/>
- [34] «PortalVue». [En línea]. Disponible en:
<https://portal-vue.linusb.org/>
- [35] «Lint». [En línea]. Disponible en:
[https://en.wikipedia.org/wiki/Lint_\(software\)](https://en.wikipedia.org/wiki/Lint_(software))
- [36] «GitHub Pages». [En línea]. Disponible en:
<https://pages.github.com/>
- [37] «Escala Likert». [En línea]. Disponible en:
<https://www.netquest.com/blog/es/la-escala-de-likert-que-es-y-como-utilizarla>
- [38] «Node.js». [En línea]. Disponible en:
<https://nodejs.org/en/>
- [39] «Instalación git». [En línea]. Disponible en:
<https://git-scm.com/book/es/v2/Inicio---Sobre-el-Control-de-Versiones-Instalaci%C3%B3n-de-Git>
- [40] «SoopBook». [En línea]. Disponible en:
<https://es.slideshare.net/carolinagil1018/soopbook>
- [41] «ZooBurst». [En línea]. Disponible en:
<http://www.app-store.es/zooburst>
- [42] «Squiffy». [En línea]. Disponible en:
<http://textadventures.co.uk/squiffy>

ANEXOS

ANEXO I: GUÍA DE USUARIO

1. BARRA DE NAVEGACIÓN

La barra de navegación es accesible desde todos los lugares de la aplicación y es la que permite ir moviéndose entre funcionalidades (véase figura 124 y 125). Dependiendo si se tiene sesión iniciada o no, aparecen iconos que te permiten crear un usuario o iniciar esa sesión o que te permiten consultar tu usuario y cerrar sesión.



Figura 124. Barra de navegación sin iniciar sesión.



Figura 125. Barra de navegación al iniciar sesión.

1. Acceso para [iniciar sesión](#).
2. Link para [crear un usuario](#) con el que iniciar sesión.
3. Tanto el nombre de la página como el link *Inicio* lleva a la [pantalla con los libros publicados](#).
4. Acceso a los [libros](#) creados por el usuario. Es necesario haber iniciado sesión.
5. Acceso a las [bibliotecas](#) del usuario, es necesario haber iniciado sesión.
6. Consultar la [información de usuario](#).
7. [Cerrar sesión](#).

2. INICIO O VISTA DE LIBROS PUBLICADOS

En la vista de inicio aparecen todos los libros publicados por distintos usuarios (véase figura 126). Se pueden ver tanto si se tiene sesión iniciada como si no. En la figura 127 se identifica cada parte de un libro dentro de esta vista.



Figura 126. Vista de libros publicados.

Los datos de los libros que aparecen son:



Figura 127. Datos de cada libro.

1. Se puede pulsar la imagen y el título para acceder a la [lectura del libro](#).
2. Permite [añadir el libro a una biblioteca](#). Esta opción solo está disponible con sesión iniciada.
3. Información del libro: autor y descripción.

3. GESTIÓN DE USUARIO

En esta sección aparece detallado todo el proceso relacionado con creación de usuario, iniciar sesión o simplemente la consulta y modificación de usuarios.

a. CREAR USUARIO

Cuando se pulsa la opción de crear usuario aparece una ventana emergente con los distintos campos a rellenar para crear la cuenta (véase figura 128).

The image shows a registration form titled "Registro". It contains the following elements:

- 1. "Usuario" input field with a person icon.
- 2. "Nombre" input field with a list of people icon.
- 3. "Email" input field with an envelope icon.
- 4. "Contraseña" input field with a shield and key icon.
- 5. "Confirmar contraseña" input field with a shield and key icon.
- 6. "Imagen de perfil:" section with a "Selecciona una imagen o arrastrala" text and a "Browse" button.
- 7. "Iniciar sesión" button.
- 8. "Crear" button.
- 9. "Cancelar" button.

Figura 128. Vista de registro.

1. Nombre de usuario único con el que se va a iniciar sesión posteriormente y con el que se va a identificar al usuario.
2. Nombre. Este campo es opcional.
3. Introducir una dirección de correo electrónico válido.
4. Contraseña, mínimo de seis caracteres.

5. Volver a escribir la contraseña como en el campo anterior.
6. Introducir imagen de usuario. Este campo es opcional.
7. Cambiar a la ventana de iniciar sesión.
8. Confirmar crear el usuario con esos datos.
9. Volver a la pantalla principal.

b. INICIAR SESIÓN

Una vez pulsada la opción indicada anteriormente para iniciar sesión, aparece una ventana emergente con campos para rellenar (véase figura 129).

The screenshot shows a login form titled 'Iniciar sesión'. It contains the following elements:

- A 'Usuario' input field with a person icon, labeled with a red '1'.
- A 'Contraseña' input field with a shield icon, labeled with a red '2'.
- A 'Crear Usuario' button, labeled with a red '3'.
- An 'Entrar' button, labeled with a red '4'.
- A 'Cancelar' button at the bottom, labeled with a red '5'.

Figura 129. Vista de iniciar sesión.

1. Introducir el nick de usuario con el que se creó la cuenta.
2. Introducir la contraseña.
3. Cambiar el contenido de la ventana para poder crear el usuario.
4. Iniciar sesión y acceder a la aplicación.
5. Salir a la pantalla de *inicio* sin sesión iniciada.

c. VISTA DE INFORMACIÓN DE USUARIO

Al acceder a la vista de usuario aparece la información mostrada en la figura 130.

The screenshot shows the user information screen in the FIKTIN app. The header includes the app name 'FIKTIN' and navigation options 'Inicio', 'Mis libros', and 'Bibliotecas'. The user profile section displays:

- A silhouette icon representing the user, labeled with a red '1'.
- The name 'Pruebas javi', labeled with a red '2'.
- The email 'prueba', labeled with a red '3'.
- The full email 'prueba@tralala.com', labeled with a red '4'.
- 'Modificar' and 'Eliminar' buttons at the bottom, labeled with red '5' and '6' respectively.

Figura 130. Vista de información del usuario.

1. Imagen del perfil. Los usuarios que al crear su cuenta no asignaron ninguna imagen personalizada tendrán una asignada por defecto.
2. Nick de usuario único.
3. Nombre del usuario.
4. Correo electrónico del usuario.
5. [Opción de modificar](#).
6. Opción de eliminar usuario.

d. MODIFICAR USUARIO

En caso de querer modificar algún dato del usuario, aparece una vista con los datos actuales del usuario donde se pueden modificar (véase figura 131).

Cambia los datos que desees modificar

El formulario muestra un ícono de perfil de usuario a la izquierda. A su derecha, hay un campo de texto con un ícono de usuario y el texto 'prueba' (2). Debajo de eso, un campo de texto con un ícono de correo y el texto 'prueba@tralala.com' (3). Luego, dos campos de contraseña con íconos de candado y puntos ocultos; el primero tiene un checkmark verde (4) y el segundo también (5). En la parte inferior, hay dos botones: 'Cancelar' (6) y 'Confirmar' (7). El campo 'Imagen de perfil:' (1) tiene un 'x' rojo a su lado.

Figura 131. Vista de modificar usuario.

1. Imagen actual. Para cambiarla pulsar la **x** y aparece la opción de subir una imagen.
2. Nombre de usuario.
3. Correo electrónico actual.
4. Contraseña.
5. Confirmación de contraseña.
6. Cancelar los cambios que se hayan hecho y volver atrás.
7. Confirmar los cambios y actualizar el usuario.

e. CERRAR SESIÓN

Al pulsar la opción de cerrar sesión, se muestra una ventana emergente con la siguiente información (véase figura 132):

Cerrar Sesión

¿Está seguro que desea cerrar sesión?

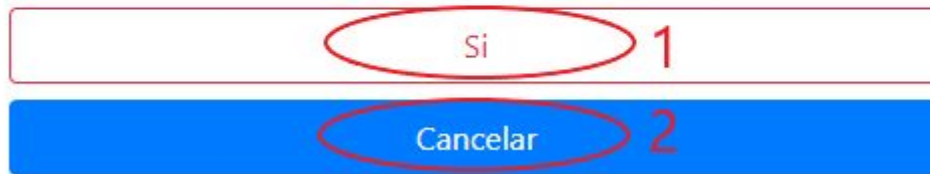


Figura 132. Vista de cierre de sesión.

1. Se cierra la sesión
2. Se deja todo como estaba.

4. BIBLIOTECAS

En esta sección aparece detallada cada una de las partes relacionadas con las bibliotecas.

a. VISTA PRINCIPAL BIBLIOTECAS

En esta pantalla principal aparecen todas las bibliotecas creadas y las opciones relacionadas con ellas (véase figura 133).

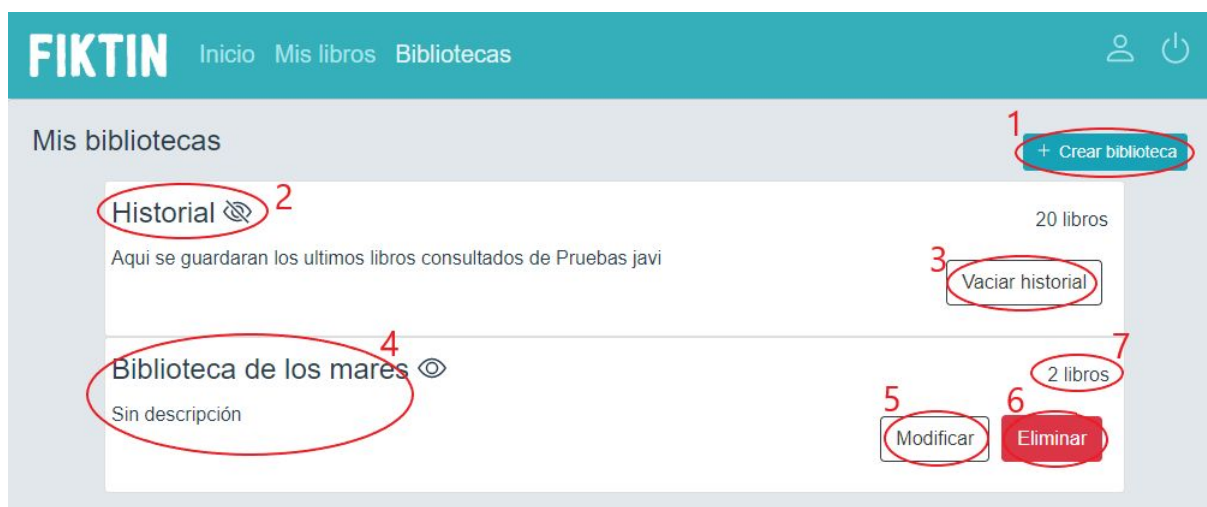


Figura 133. Vista de bibliotecas.

1. Opción para [crear una biblioteca](#) nueva.
2. Historial: biblioteca creada por el sistema donde se almacenan los últimos libros abiertos.
3. Vaciar los libros de la biblioteca historial.
4. Una biblioteca creada por el usuario, aparece el nombre y la descripción en caso de tenerla. Todo el recuadro de la biblioteca es pulsable para [acceder a su contenido](#).
5. [Modificar](#) la información de una biblioteca existente.
6. Eliminar la biblioteca en cuestión.

7. Número de libros que contiene la biblioteca.

b. CREAR BIBLIOTECA

Cuando se pulsa sobre la opción de crear biblioteca se abre una ventana emergente con un formulario para rellenar (véase figura 134).

Crear biblioteca

The screenshot shows a form titled 'Crear biblioteca'. It has two input fields: 'Nombre (0 / 50)' and 'Descripción 0 / 300'. The 'Nombre' field contains the text 'Nombre' and has a red circle with the number '1' next to it. Below it is the text 'Introduce al menos 1 caracter'. The 'Descripción' field is empty and has a red circle with the number '2' next to it. Below the fields is the question '¿Cómo deseas la privacidad de tu biblioteca?' with two radio button options: 'Pública: La podrá ver todo el mundo' (selected) and 'Privada: Solo la podrás ver tú'. The 'Pública' option has a red circle with the number '3' next to it. At the bottom right are two buttons: 'Cancelar' (with a red circle and number '4') and 'Crear' (with a red circle and number '5').

Figura 134. Vista de crear biblioteca.

1. Nombre que se le quiere dar a la nueva biblioteca.
2. Apartado para escribir la descripción de la biblioteca si se desea.
3. Establecer la privacidad de la biblioteca.
4. Cancelar la operación.
5. Crear la biblioteca.

c. MODIFICAR BIBLIOTECA

Al elegir la opción de modificar una biblioteca existente se abre una ventana similar a la de crear biblioteca desde la cual se pueden modificar los datos existentes de dicha biblioteca (véase figura 135).

Modificar biblioteca

The screenshot shows a form titled 'Modificar biblioteca'. It has two input fields: 'Nombre (23 / 50)' and 'Descripción 0 / 300'. The 'Nombre' field contains the text 'Biblioteca de los mares' and has a green checkmark to its right. It has a red circle with the number '1' next to it. The 'Descripción' field is empty and has a red circle with the number '2' next to it. Below the fields is the question '¿Cómo deseas la privacidad de tu biblioteca?' with two radio button options: 'Pública: La podrá ver todo el mundo' (selected) and 'Privada: Solo la podrás ver tú'. The 'Pública' option has a red circle with the number '3' next to it. At the bottom right are two buttons: 'Descartar cambios' (with a red circle and number '4') and 'Guardar' (with a red circle and number '5').

Figura 135. Vista de modificar biblioteca.

1. Nombre de la biblioteca.

2. Descripción.
3. Privacidad.
4. Cancelar la operación.
5. Guardar los cambios.

d. AÑADIR LIBRO A BIBLIOTECA

Los usuarios pueden tener unas bibliotecas personales donde guardar los libros que quieran, cuando se quiere añadir un libro nuevo aparece una ventana emergente como la mostrada en la figura 136.



Figura 136. Vista de añadir libro a biblioteca.

1. Aparecen las bibliotecas disponibles a las que se puede añadir el libro. Se selecciona el recuadro de todas a las que se le quiera añadir.
2. Cancelar operación y volver a la ventana anterior.
3. Guardar los cambios.

e. VISTA CONTENIDO BIBLIOTECA

Al acceder al contenido de una biblioteca se muestran todos los libros que esta contiene, además de dar ciertas opciones para su organización (véase figura 137).



Figura 137. Vista del contenido de una biblioteca.

1. Volver a la vista con todas las bibliotecas.
2. Nombre de la biblioteca a la cual se ha querido acceder a su contenido.
3. Información de los libros que se encuentran en la biblioteca. Se puede acceder a su [lectura](#) desde esta pantalla pulsando en la portada o el título.
4. Bajar el elemento en la lista, al pulsar aparecerá el libro de “Diez negritos” primero.
5. La versión inversa de la opción anterior, subir un elemento en la lista.
6. Eliminar un libro de la biblioteca.

5. LIBROS

En esta sección se ofrece una vista más detallada de las vistas y opciones relacionados con los libros, incluyendo la lectura, escritura y su creación.

a. VISTA MIS LIBROS

La pantalla de *mis libros* accesible a través de la barra de navegación es donde se gestionan todos los libros creados por el usuario (véase figura 138).

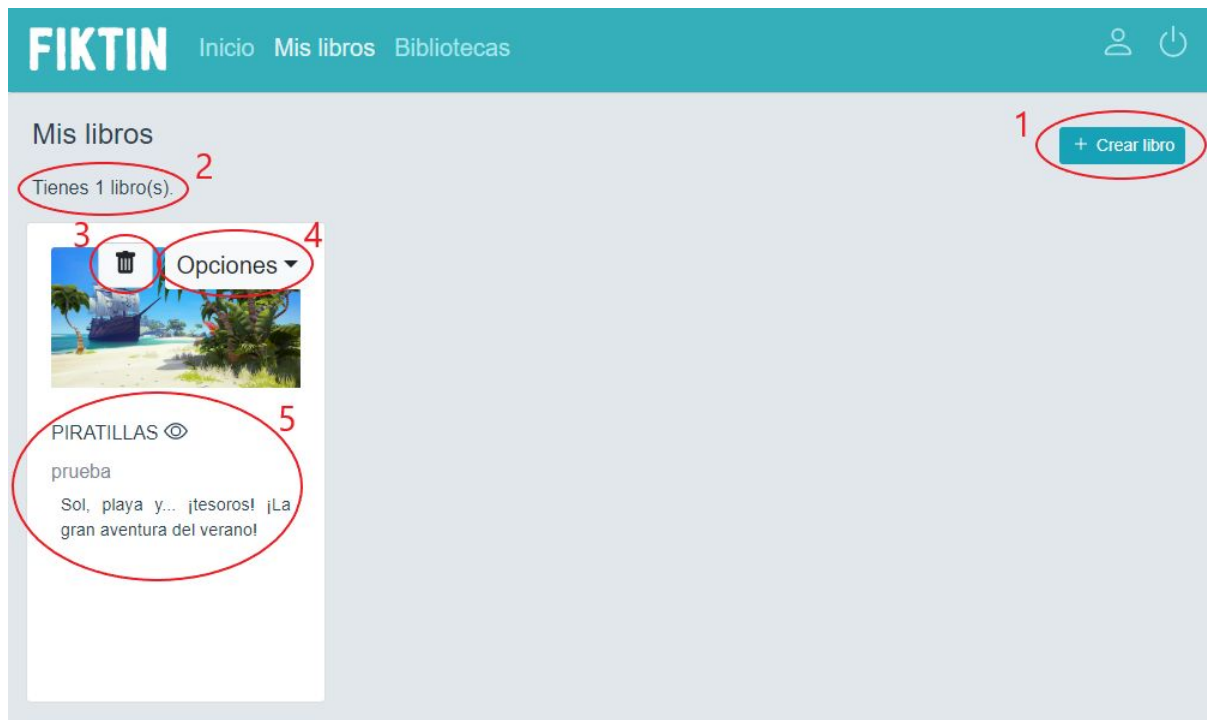


Figura 138. Vista de mis libros.

1. Botón para crear un libro
2. Información de cuántos libros tiene el usuario.
3. Botón de [eliminar libro](#). Tiene una doble confirmación que consiste en pulsarlo otra vez.
4. Menú desplegable con [opciones](#) del libro.
5. Información del libro: título, autor, descripción y si es público o no, indicado por un ojo normal o tachado. Si se pulsa sobre la imagen o el título se accede a la [lectura del libro](#), y a la posibilidad de [escribirlo o editarlo](#).

b. CREAR LIBRO

Cuando se pulsa sobre la opción de crear libro, aparece una ventana emergente con un formulario (véase figura 139).

Crear libro

The image shows a web form titled "Crear libro" with the following elements and annotations:

- Título:** A text input field containing "Título del libro" with a red circle and the number 1 next to it. Below the field is the text "Introduce al menos 3 caracteres".
- Descripción:** A text area containing "Descripción" with a red circle and the number 3 next to it.
- Autor:** Two radio button options: "Nombre de usuario" (selected) and "Nickname". A red circle and the number 2 are next to the "Nombre de usuario" option.
- Etiquetas:** A text input field containing "Añade una etiqueta..." with a red circle and the number 4 next to it.
- Buttons:** Three buttons at the bottom: "Publicar" with a checkbox (circled with 5), "Cancelar" (circled with 6), and "Crear libro" (circled with 7).

Figura 139. Vista de crear libro.

1. Introducir el título del libro.
2. Elegir si se desea que aparezca la información de autor con el nombre de usuario o el nickname.
3. Descripción del libro. Este parámetro es opcional.
4. Etiquetas con las que identificar el libro. Este parámetro es opcional.
5. Elegir si nada más crear el libro se quiere publicar.
6. Cancelar la creación del libro.
7. Crear el libro.

c. OPCIONES DEL LIBRO Y ELIMINAR

Algunas opciones anteriormente mencionadas del libro, son las de eliminar y las que se encuentran en el menú desplegable (véase figura 140 y figura 141).



Figura 140. Vista de eliminar libro. Figura 141. Vista de confirmación de eliminación.

1. Modificar los datos del libro como descripción o imagen.
2. [Agregar el libro a una biblioteca.](#)
3. Botón para eliminar el libro, requiere una doble confirmación.
4. Confirmar la opción de eliminar volviendo a pulsar sobre el botón cuando éste se encuentra en rojo.

d. MODIFICAR LIBRO

Al elegir la opción de modificar aparece una ventana muy similar a la de creación de libro con los parámetros ya existentes, con el extra de poder agregar una imagen (véase figura 142).

Modificar libro

The screenshot shows the 'Modificar libro' interface with the following elements and callouts:

- 1:** Title input field containing 'Piratillas'.
- 2:** Author selection options: 'Nombre de usuario' (selected) and 'Nickname'.
- 3:** Tag input field with placeholder 'Añade una etiqueta...'.
- 4:** Cover image area with a 'Browse' button and an 'Eliminar' (Delete) button.
- 5:** Description input field containing 'Sol, playa y... ¡tesoros! ¡La gran aventura del verano!'.
- 6:** 'Publicar' (Publish) button with a checked checkbox.
- 7:** 'Descartar' (Discard) button.
- 8:** 'Guardar' (Save) button.

Figura 142. Vista de modificar libro

1. Título del libro
2. Elegir cómo aparecerá el autor, si con el nickname de usuario o su nombre.
3. Añadir etiquetas personalizadas al libro.
4. Agregar una imagen de portada o eliminar la existente.
5. Modificar la descripción ya existente.
6. Cambiar si el libro está publicado o no.
7. Descartar los cambios y volver.
8. Guardar las modificaciones.

e. LECTURA DE UN LIBRO

Al acceder la lectura del libro se muestra una pantalla con su contenido y, dependiendo de si se trata un libro del usuario con sesión iniciada o no, aparecen algunas opciones distintas (véase figura 143).



Figura 143. Vista de lectura de libro.

1. Volver a la pestaña desde la que se accedió al libro.
2. Título del libro que se está leyendo.
3. Editar o escribir el libro. Sólo aparece si se trata de un libro del autor. En caso de estar publicado, esta opción está deshabilitada.
4. Publicar o despublicar el libro.
5. Cambiar entre secciones manualmente.
6. Contenido del libro.

f. EDICIÓN DE UN LIBRO

Cuando se accede a la edición de un libro, la pantalla general que aparece es la mostrada en la figura 144.

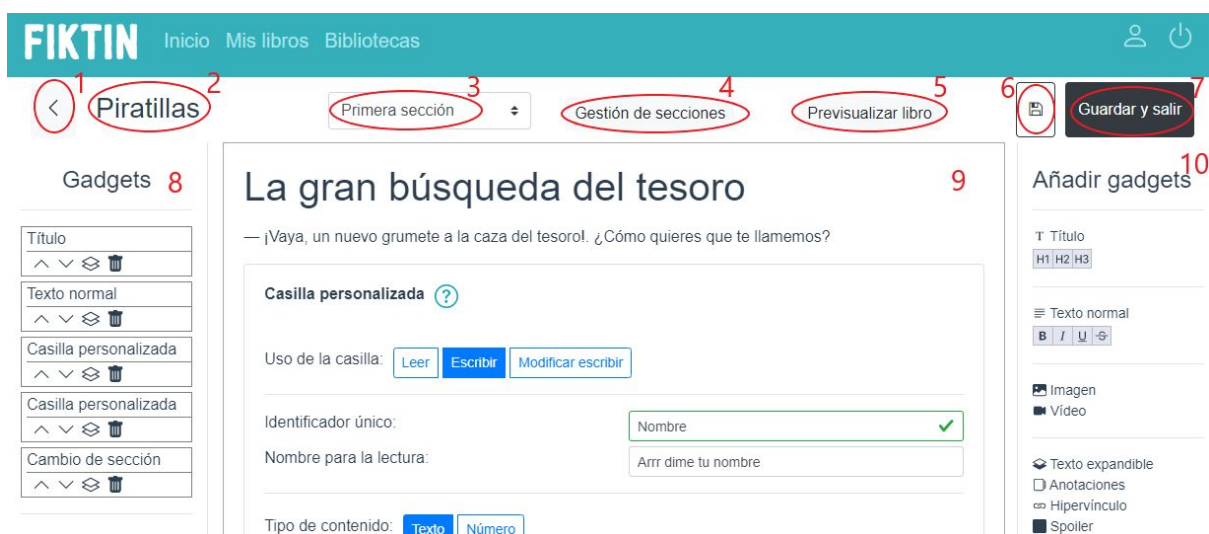


Figura 144. Vista de editar un libro.

1. Volver a la [lectura del libro](#).

2. Título del libro.
3. Navegar manualmente entre secciones.
4. [Gestión de secciones](#), donde se pueden crear, borrar o duplicar.
5. Previsualizar lo que se lleva escrito de libro. Te lleva a una vista de lectura para ver cómo sería el resultado final.
6. Guardar cambios.
7. Guardar y volver a la [lectura del libro](#).
8. Gadgets que se tienen actualmente en el libro. Desde aquí se pueden [gestionar](#): ordenar, eliminar o duplicar.
9. Hoja de escritura, contiene todos los gadgets que el autor vaya [añadiendo](#) y que quedan indicados en el punto anterior (8).
10. Barra lateral con todos los gadgets existentes desde donde se pueden añadir al libro.

i. GESTIÓN DE SECCIONES

Al pulsar sobre la opción de gestión de secciones se abre una ventana como la mostrada en la figura 145.

2 (x)

Gestión de secciones

1 Sección actual: QJ300oRu4rZbaQtKs32A - Primera sección

Nota: La sección 0 de la tabla indica la primera sección que se carga al abrir el libro

	ID	Nombre	Acciones
0	3 QJ300oRu4rZbaQtKs32A	4 Primera sección	6 ↑ 6 ↓ 8 ☰ 8 🗑️
1	NYB3hZ77FdseH2XRtE3i	Nueva sección	5 ↑ 7 ↓ 7 ☰ 7 🗑️

9 Nueva sección

Los cambios se guardarán al cerrar la ventana

10 Guardar y cerrar

Figura 145. Vista de gestión de secciones.

1. Sección en la que se está editando el libro actualmente.
2. Cerrar la ventana. Los cambios se guardan antes de cerrarla.
3. Identificador único de la sección. Evita que en caso de existir dos secciones con el mismo nombre se confundan los saltos o interacciones.
4. Nombre de la sección. Este dato es personalizable.
5. Ordenar secciones: subir una sección en la lista.
6. Ordenar secciones: bajar una sección en la lista.
7. Duplicar una sección con todo su contenido (gadgets).

8. Eliminar una sección. Mínimo tiene que existir una sección, en caso de haber solo una esta función se deshabilita.
9. Crear una nueva sección.
10. Cerrar la ventana guardando los cambios.

ii. AÑADIR GADGETS

La barra lateral derecha en la vista de editar libro es donde se eligen los gadgets que se desean añadir (véase figura 146).

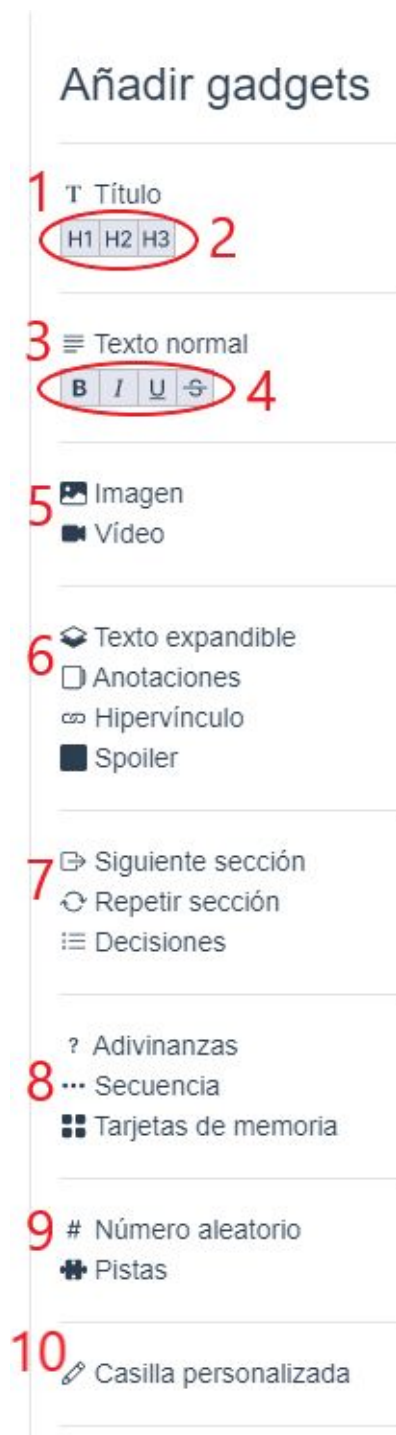


Figura 146. Vista de añadir gadgets.

1. Agregar gadget de tipo [título](#).
2. Modificar el tamaño de los títulos de un gadget seleccionado.
3. Agregar [texto normal](#).
4. Cambiar formato del texto normal elegido.
5. Gadget para añadir elementos de [imagen](#) o [vídeo](#).
6. Gadgets relacionados con opciones especiales de escritura. [Texto expandible](#) permite hacer que se despliegue más texto en un sitio específico. Las [anotaciones](#) permiten al autor escribir sus notas para que aparezcan en una ventana emergente. El [hipervínculo](#) permite agregar un enlace a una página externa. El [spoiler](#) permite hacer que un texto esté oculto y que al pasar el ratón por encima éste se muestre.
7. Gadgets relacionados con cambios de sección simples. [Siguiendo sección](#) es el cambio más simple de todos, con él se elige la sección a la que se quiere ir al terminar una parte del libro. [Repetir sección](#) recarga la sección actual y permite repetir los puzzles. [Decisiones](#) permite al escritor darle al lector distintas opciones a realizar y dependiendo de la que elija le puede llevar a una sección u otra.
8. En este apartado se encuentran los puzzles para realizar interacciones. Con las [adivinanzas](#) se pone un acertijo y el lector tiene que intentar resolverlo. En la [secuencia](#) se dan unos parámetros y se quiere que el lector complete la secuencia. Las [tarjetas de memoria](#) son el juego clásico de ir levantando tarjetas de dos en dos hasta emparejarlas todas.
9. Este apartado también tiene juegos para el lector pero más sencillos. Con el [número aleatorio](#) se define un rango y dependiendo del número que salga que lleve a una sección u a otra. Con las [pistas](#) se le da al usuario varias opciones y tiene que intentar elegir la correcta.
10. La [casilla personalizada](#) permite que el lector introduzca un dato y este vaya apareciendo durante toda la lectura.

iii. GESTIONAR GADGETS

La barra lateral izquierda en la vista de editar libro contiene los gadgets que se han ido añadiendo al libro y es desde donde se puede cambiar su colocación, copiarlos o eliminarlos (véase figura 147).

Gadgets

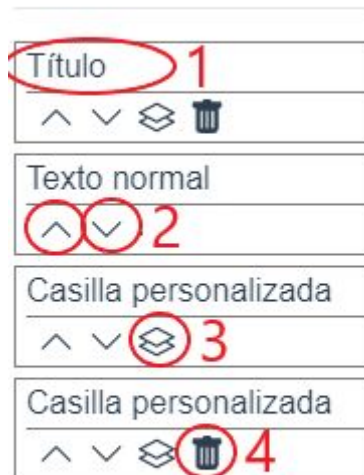


Figura 147. Vista de gestión de gadgets.

1. Tipo de gadget añadido.
2. Subir o bajar su posición en el libro.
3. duplicar un gadget con toda su personalización.
4. Eliminar el gadget del libro.

6. LIBROS

En esta sección se verá más en detalle el funcionamiento de cada gadget y qué significan cada uno de sus parámetros.

a. TÍTULO

El gadget de título es junto con el de [texto normal](#) uno de los más sencillos e importantes. Para su utilización basta con añadir el gadget y **pulsar en la hoja** de edición en la posición en la que se encuentra, es decir, si se encuentra entre dos gadgets distintos hay que pulsar entre esos dos. Esta colocación se observa en la barra de [gestión de gadgets](#).

Existen tres tipos de tamaño de título, para aplicarlos basta con tener el gadget seleccionado, ya sea en la hoja o desde la [gestión de gadgets](#), y pulsar en la opción de tamaño: H1, H2 o H3. Estas opciones cambian el tamaño del título entero, no es posible tener distintos tamaños en un mismo título (véase figura 148).

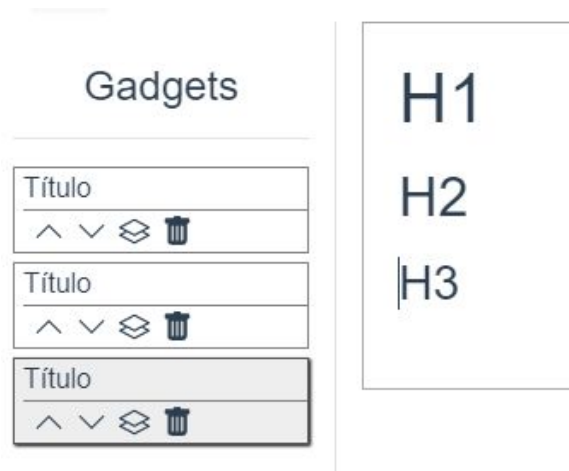


Figura 148. Vista de títulos.

b. TEXTO NORMAL

El texto normal es el gadget más sencillo e importante de la edición de un libro, ya que es el que permite escribir el libro principalmente. Para utilizarlo hay que crear el gadget y pulsar en la hoja de edición en la posición que se encuentra, la cual se puede consultar en la barra lateral de [gestión de gadgets](#). Existe una manera sencilla de personalización del texto que consiste en cambiar el formato del texto con cursiva, negrita, subrayado o tachado. A diferencia de con el [título](#) esta personalización se puede aplicar solo a ciertas palabras del texto (véase figura 149).

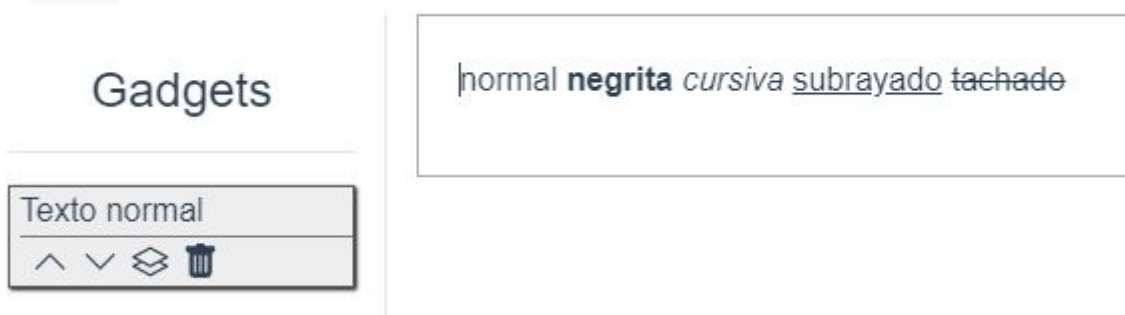


Figura 149. Vista de texto normal.

c. IMAGEN

Cuando se elige el gadget de imagen aparece un recuadro como el siguiente gracias al cual se puede elegir la foto que se desea añadir al libro (véase figura 150).

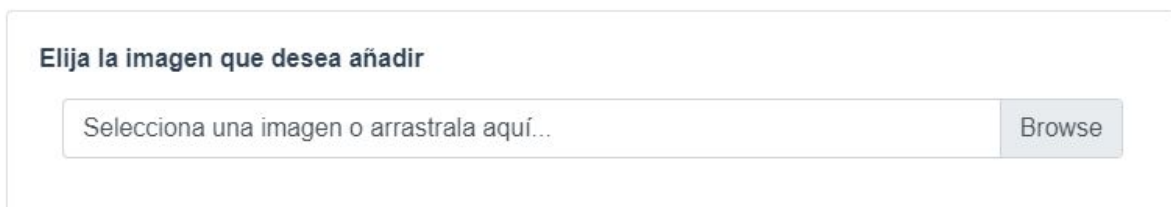


Figura 150. Vista de imagen.

d. VÍDEO

Similar al gadget de imagen es el gadget para introducir un vídeo, la diferencia es que a este se le pueden configurar algunas opciones (véase figura 151).

Elija el video que desea añadir

Selecciona un video o arrástrala aquí... **1** Browse

2 Autorreproducción


3 Bucle

Figura 151. Vista de vídeo.

1. Ventana de selección de vídeo.
2. Al elegir esta opción el vídeo se auto reproduce cuando se accede al libro.
3. Al elegir esta opción el vídeo se reproduce en bucle.

e. ANOTACIONES

Las anotaciones son comentarios introducidos por el autor relacionados a un texto para que se abra una pestaña emergente con la información adicional elegida. Existe la opción de añadir una imagen al final del texto, en cuyo caso cambia la ventana (véase figura 152 y figura 153).

Edite su anotación **4** 

Texto principal


Introduce el texto que quieres que se muestre... **1**

Texto emergente

Introduce el texto que se mostrara al emergentemente... **2**

3 Añadir imagen al final del texto

Figura 152. Vista de anotaciones.

Edite su anotación 


Texto principal

Introduce el texto que quieres que se muestre...

Texto emergente

Introduce el texto que se mostrara al emergentemente...

5 Elija la imagen que desea añadir

Selecciona una imagen o arrastrala aquí... 

6 Cancelar **7** Confirmar

Figura 153. Vista de anotaciones con imagen.

1. Texto que se va a mostrar en la lectura y al que va a ir relacionada la anotación.
2. Texto que aparecerá en la ventana emergente.
3. Opción para dar la posibilidad de agregar una imagen al final del texto, como se muestra en la segunda imagen.
4. Previsualizar el resultado del gadget, cómo se verá en modo lectura.
5. Elegir la imagen que se desea añadir.
6. Cancelar agregar una imagen. Esta opción devuelve la ventana a como se encuentra en la primera imagen.
7. Confirmar la elección de imagen para que esta se guarde.

f. TEXTO EXPANDIBLE

El texto expandible es similar a las anotaciones pero más sencillo, ya que solo tiene la opción de escribir el texto al que irá asociado el texto expandible y la personalización de este texto (véase figura 154).

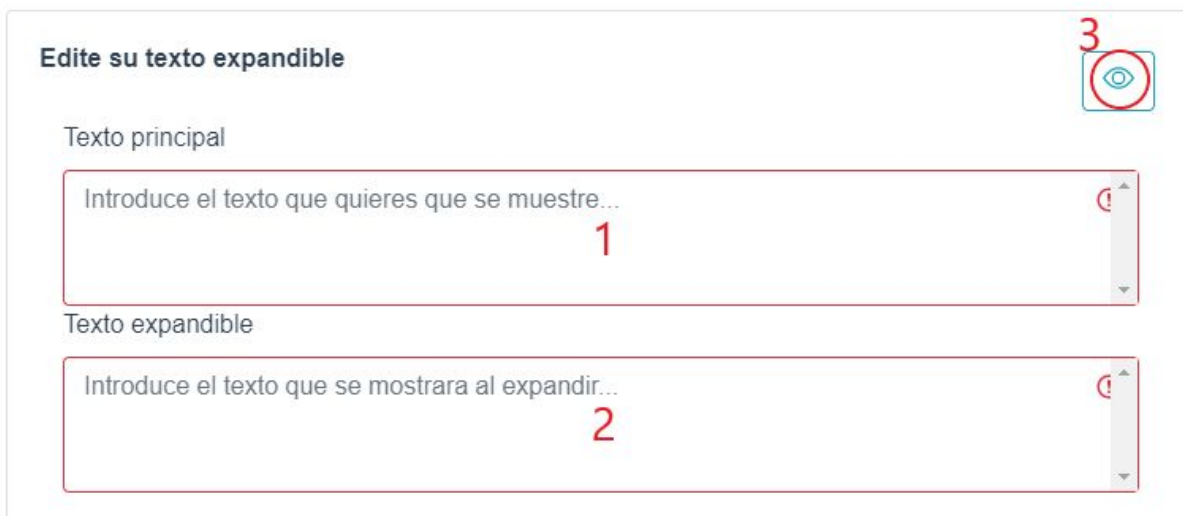


Figura 154. Vista de texto expandible.

1. Texto que aparecerá en el libro.
2. Texto que aparecerá al expandir el gadget.
3. Previsualización del resultado final.

g. HIPERVÍNCULO

Permite añadir al libro enlaces a páginas externas (véase figura 155).



Figura 155. Vista de hipervínculos.

1. Hipervínculo de la página externa.
2. Texto que irá relacionado a este hipervínculo.
3. Previsualización del resultado final.

h. SPOILER

Permite agregar un texto oculto que al pasar el ratón por encima sea legible (véase figura 156).



Figura 156. Vista de spoiler.

1. Texto que aparecerá oculto.
2. Previsualización de cómo se verá durante la lectura.

i. SIGUIENTE SECCIÓN

Este gadget permite hacer un cambio de sección simple (véase figura 157).

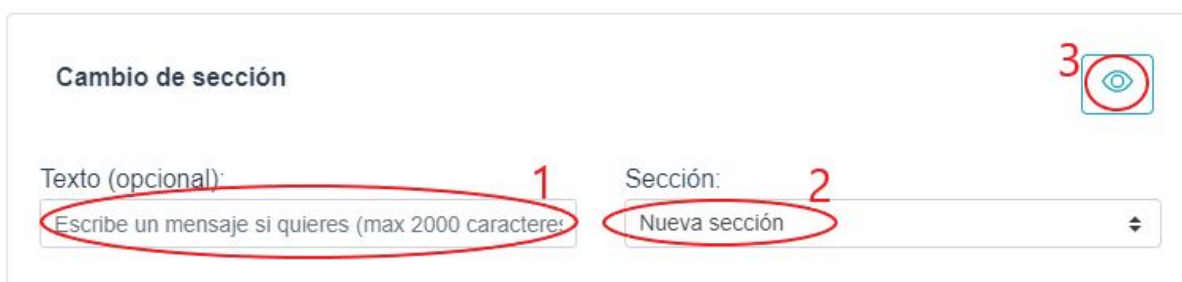


Figura 157. Vista de siguiente sección.

1. Texto que acompañará al botón de cambio de sección
2. Sección a la que enviará el gadget.
3. Previsualización de modo lectura.

j. REPETIR SECCIÓN

Con repetir sección se volverá a recargar la sección actual, permitiendo así que se restablezcan los juegos y se puedan volver a intentar resolver (véase figura 158).



Figura 158. Vista de repetir sección.

1. Texto que acompañará al botón de recargar sección.
2. Previsualización de cómo aparecerá en modo lectura.

k. DECISIONES

Con las decisiones se permite que el autor le de distintas opciones a elegir al lector y, que dependiendo de la que elija, le lleve a una sección o a otra, pudiendo llevar todas las elecciones al mismo sitio (véase figura 159).

Toma de decisiones 4

Nº de opciones: 2 1

Decisión 0: 2 3 Sección:

Decisión 1:

Figura 159. Vista de toma de decisiones.

1. Número de opciones entre las que se le darán a elegir al lector.
2. Texto de la opción en cuestión.
3. Sección a la que enviará la opción en cuestión.
4. Previsualizar el gadget en modo lectura.

I. ADIVINANZA

La adivinanza le permite al autor poner un acertijo que el lector deberá resolver en un número máximo de intentos. En caso de resolverla le llevará a una sección definida, y en caso de no resolverla se podrá enviar al lector a otra sección o permitir que siga leyendo (véase figura 160).

Adivinanza 6

Cambio de sección al acertar: 1

2

Nº de intentos: 1 3

Pregunta: 4

Respuesta: 5

Figura 160. Vista de adivinanza.

1. Sección a la que saltará el libro si se acierta la adivinanza.
2. Botón para elegir si se quiere seguir leyendo o que cambie de sección al fallar. En [secuencia](#) aparece la versión del botón pulsado.
3. Número de intentos que el lector tendrá para resolver la adivinanza.
4. Adivinanza.
5. Respuesta.
6. Previsualización de la lectura.

m. SECUENCIA

El autor define una secuencia con letras o números y se le pide al lector que la complete. En caso de completarla correctamente le llevará a otra sección del libro y en caso contrario permanecerá en la sección actual para seguir leyendo o le llevará a otra sección también (véase figura 161).

The screenshot shows a form titled "Completar la secuencia" with the following elements and annotations:

- 1:** A dropdown menu labeled "Cambio de sección al acertar:" with the option "Nueva sección" selected.
- 2:** A button labeled "Continuar lectura al fallar" circled in red.
- 3:** A dropdown menu labeled "Nueva sección 2" with a downward arrow.
- 4:** A slider control for "Nº de intentos: 1" with a blue dot at the start.
- 5:** A slider control for "Nº de campos dados: 1" with a blue dot at the start.
- 6:** A label "Elemento 1:" circled in red, next to a text input field containing "Texto dado máximo de 20 caracteres".
- 7:** A slider control for "Nº de campos a solucionar: 1" with a blue dot at the start.
- 8:** A label "Solución 1:" circled in red, next to a text input field containing "Solución de máximo de 20 caracteres".
- 9:** A red eye icon in the top right corner, also circled in red.

Figura 161. Vista de completar la secuencia.

1. Sección a la que enviará en caso de resolver el juego.
2. Quitar la opción de al fallar cambiar de sección y seguir leyendo el libro en la sección actual. Una pantalla con la opción cambiada está en [adivinanzas](#).
3. Sección a la que se enviará al lector al no superar el juego.
4. Número de intentos máximos para resolver el juego.
5. Número de campos base que se dan de la secuencia.
6. Elementos de la secuencia base.
7. Número de campos a rellenar de la secuencia.

- Soluciones de los elementos que debe rellenar el lector.
- Previsualización de modo lectura.

n. TARJETAS DE MEMORIA

El clásico juego de levantar tarjetas dadas la vuelta para encontrar todas las parejas. Consiste en ir girando tarjetas de dos en dos, en caso de ser una pareja se quedarán dadas la vuelta, en caso contrario se volverán a girar. Si se resuelve el puzzle en el número límite de movimientos que se da, se continuará la lectura por un sitio, en caso contrario podrá seguir leyendo o cambiar de sección (véase figura 162).



Figura 162. Vista de tarjetas de memoria.

- Sección a la que se irá en caso de resolver el puzzle.
- Botón para cambiar entre la opción de quedarse en la sección si no se resuelve o si ir a otra sección. Un ejemplo de la ventana si se elige cambiar de sección al fallar está en [secuencias](#).
- Número de parejas que hay que encontrar.
- Número máximo de movimientos para resolver el puzzle.
- Existe la posibilidad de customizar las tarjetas como veremos a continuación. En caso de no elegir esta opción, tienen unos colores puestos por defecto.
- Previsualizar el resultado final.

En caso de elegir la opción de customizar tarjetas existen dos tipos de customización: por colores o con palabras.

En la figura 163 se puede ver la vista en caso de elegir la opción de customizar por colores.

Tarjetas de memoria 👁️

Cambio de sección al acertar: Nueva sección ⌵

Nº de parejas: 2 ●

Nº de movimientos: 6 ●

Tipo de customización Colores **1** ⌵ **2**


Pareja 1 **3**

Pareja 2 Elige algún color

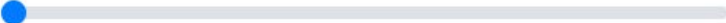
Figura 163. Vista de tarjetas de memoria por colores.


1. Opción de personalización de tarjetas.
2. Cancelar y volver a utilizar los valores por defecto.
3. Elección de color de cada pareja.

Para los colores en una pareja de tarjetas tienen las dos el mismo color, mientras que si se elige personalizarlo con palabras cada una puede tener una palabra o parte de una palabra (véase figura 164).

Tarjetas de memoria 

Cambio de sección al acertar:

Nº de parejas: 2 

Nº de movimientos: 6 

Tipo de customización 1 2

Pareja 1 3 4


Pareja 2

Figura 164. Vista de tarjetas de memoria por palabras.


1. Tipo de personalización elegido.
2. Cancelar y utilizar los colores por defecto.
3. Texto de la primera tarjeta de una pareja.
4. Texto de la segunda tarjeta de la misma pareja.

o. NÚMERO ALEATORIO

Con el gadget de número aleatorio se puede definir un rango del que se sacará un número totalmente aleatorio para tomar una decisión por el lector de salto de sección. Se pueden definir unas condiciones a partir de las cuales se enviará a una sección u a otra (véase figura 165).

Número aleatorio 6 

Limite inferior: 1 Limite superior: 2

Nº de condiciones: 2 

3 Condición	4 Valor	5 Sección
<input type="text" value="Menor o igual"/>	<input type="text" value="5"/>	<input type="text" value="Nueva sección 2"/>
<input type="text" value="Mayor"/>	<input type="text" value="5"/>	<input type="text" value="Primera sección"/>

Figura 165. Vista de número aleatorio.

1. Primer número del intervalo a definir por el escritor.
2. Último número del rango del que se sacará el número aleatorio.
3. Condiciones para cambiar de sección.
4. Valor aplicado a las condiciones.
5. Cambio de sección si se cumple la condición definida.
6. Previsualización del gadget.

p. PISTAS

Dada una serie de “pistas” o fragmentos de texto definidas por el autor se le pide al lector que elija cuáles son las correctas para continuar con la lectura (véase figura 166).

Figura 166. Vista de completar pistas.

1. Cambio de sección si se acierta la respuesta.
2. Se puede elegir cambiar de sección al fallar o quedarse leyendo en la sección actual. Un ejemplo con la ventana de cambio de sección al fallar está en [secuencia](#).
3. Número de pistas que definirá el autor.
4. Valor de cada pista.
5. Número de respuestas que deberá introducir correctamente el lector.
6. Solución correcta de cada respuesta.
7. Previsualización del gadget.

q. CASILLA PERSONALIZADA

De todos los gadgets anteriormente vistos, la casilla personalizada sería el más complejo. Esto se debe a que para que funcione adecuadamente se necesitan **mínimo dos gadgets** de casilla personalizada, una de escritura y otra de lectura. La razón para esto es que hay que diferenciar entre distintos tipos de casilla personalizada dependiendo de su uso, existe la de escritura, lectura o modificar escritura.

La primera sería la de escritura. Sirve para permitir al lector introducir el dato personalizado que va a ir apareciendo a lo largo de la lectura. En la figura 167 se muestra la vista de edición la casilla.

Casilla personalizada ? ¹

Aviso: Si el nombre de la casilla no es válido, no se verá en la lectura

Uso de la casilla: ²

Identificador único: ³ ⓘ

Nombre para la lectura: ⁴

Tipo de contenido: ⁵

Valor por defecto: ⁶

Figura 167. Vista de casilla personalizada en modo escritura.

1. Ayuda.
2. Panel de elección de modo para la casilla.
3. Nombre único para representar a esta casilla, gracias a él podremos acceder al contenido definido en ella desde las casillas de lectura.
4. Título o texto con el que se le indica al lector qué necesita introducir.
5. Tipo de valor que se le pide que introduzca al lector.
6. Valor por defecto, en caso de no querer rellenar nada el lector, aparecerá ese valor.

Un ejemplo de cómo se vería esta casilla en la lectura del libro es la mostrada en la figura 168.

Arrr dime tu nombre: ¹ ²

Figura 168. Vista de casilla personalizada en modo escritura y en lectura del libro.

1. Información definida por el autor. Punto 4 del apartado anterior.
2. Valor por defecto y donde el lector escribirá su información.

El segundo tipo de casilla personalizada es el de lectura, donde se muestra el dato introducido por el lector. En la ventana de lectura aparece como texto normal resaltado en azul, mientras que en la ventana de edición esta casilla aparece como se muestra en la figura 169.

Casilla personalizada ?

Uso de la casilla: **1** Leer Escribir Modificar escribir

Texto anterior: **2** Texto anterior (opcional)

Leer de casilla: **3** casilla 1

Texto posterior: **4** Texto posterior (opcional)

Figura 169. Vista de casilla personalizada en modo lectura.

1. Uso que se le dará a la casilla.
2. Texto anterior al valor definido en la casilla personalizada.
3. Identificador de la casilla asociada.
4. Texto que aparece posteriormente al valor de la casilla personalizada.

El último tipo de casilla personalizadas, son las que te permiten cambiar el valor introducido una vez avanzada la historia, en vez de definir una nueva casilla se le permite al lector cambiar el valor a todas las asociadas. Esta casilla en cuanto a apariencia es igual que la casilla de escritura normal (véase figura 170).

Casilla personalizada ?

Uso de la casilla: **1** Leer Escribir Modificar escribir

Modificar la casilla: **2** casilla 1

Nombre para la lectura: **3** Modificar

Figura 170. Vista de casilla personalizada en modo de modificar escritura.

1. Uso de la casilla.
2. Casilla a la que se le quiere modificar el valor.
3. Texto que aparecerá acompañado del formulario.

ANEXO II: GUÍA DE INSTALACIÓN

Para la instalación del proyecto es necesario tener el framework de Vue. Este requiere de Node.js [38] en su versión 8.9 o superior que se encuentra en su página web. A continuación se instala Vue CLI con el comando:

```
npm install -g @vue/cli
```

Descargar o clonar el proyecto de GitHub. Para descargar se puede hacer desde el siguiente enlace: <https://github.com/avadop/Fiktin>

Si se desea clonar el proyecto es necesario instalar la herramienta Git [39] y ejecutar en la consola el siguiente comando:

```
git clone https://github.com/avadop/Fiktin.git
```

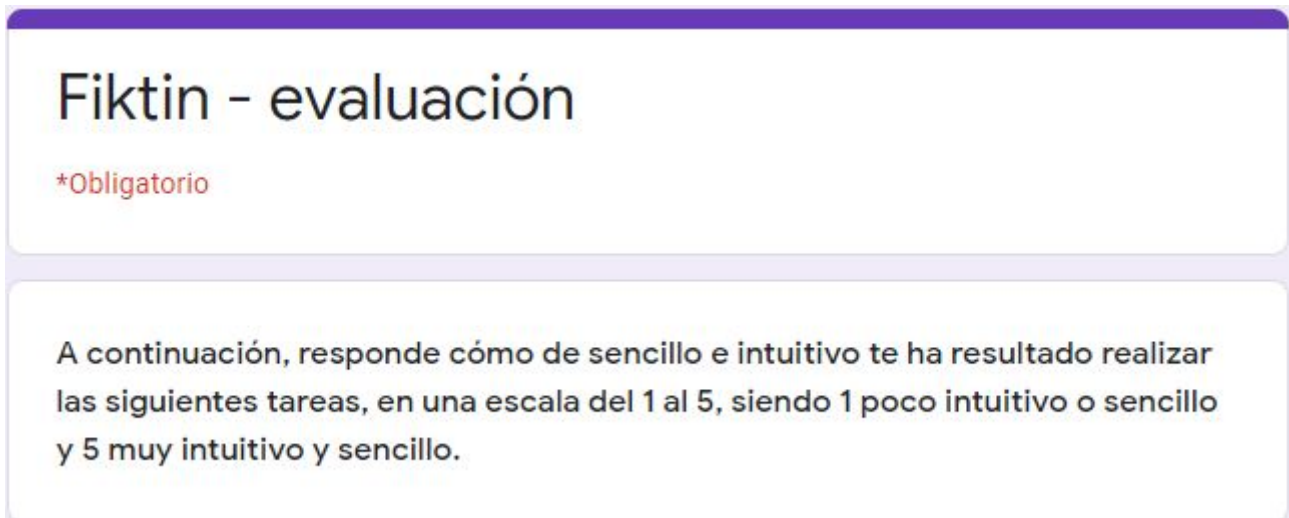
Desde la carpeta donde se encuentra el proyecto hay que abrir una consola y lanzar los siguientes comandos:

```
npm install  
npm run serve
```

Cuando se terminen de ejecutar los comandos aparecerá en la consola el enlace local de la página web que a de ser copiado en el navegador para acceder a la aplicación.

ANEXO III: EVALUACIÓN DE LA USABILIDAD

En este tercer anexo se recogen las capturas de pantalla de las preguntas planteadas en la encuesta para la evaluación de usabilidad.

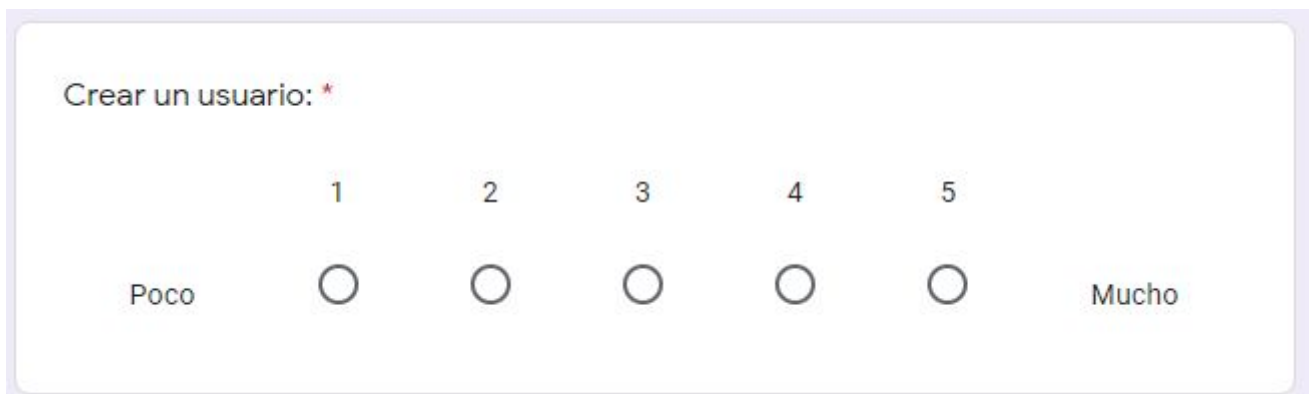


Fiktin - evaluación

*Obligatorio

A continuación, responde cómo de sencillo e intuitivo te ha resultado realizar las siguientes tareas, en una escala del 1 al 5, siendo 1 poco intuitivo o sencillo y 5 muy intuitivo y sencillo.

Figura 171. Introducción de la evaluación.

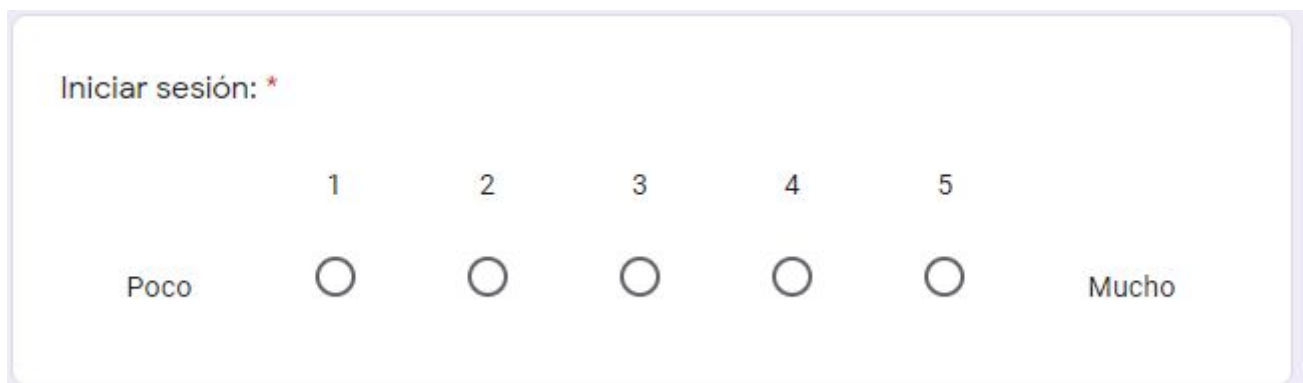


Crear un usuario: *

1 2 3 4 5

Poco Mucho

Figura 172. Pregunta “Crear un usuario”



Iniciar sesión: *

1 2 3 4 5

Poco Mucho

Figura 173. Pregunta “Iniciar sesión”

Acceder a tus datos de usuario: *

Poco 1 2 3 4 5 Mucho

Figura 174. Pregunta “Acceder a datos de usuario”

Crear una biblioteca: *

Poco 1 2 3 4 5 Mucho

Figura 175. Pregunta “Crear biblioteca”

Agregar un libro desde inicio a tu biblioteca: *

Poco 1 2 3 4 5 Mucho

Figura 176. Pregunta “Agregar libro a una biblioteca”

Comprobar que dicho libro se encuentra dentro de tu biblioteca: *

Poco 1 2 3 4 5 Mucho

Figura 177. Pregunta “Ver libro añadido en biblioteca”

Abrir el libro de "Piratillas": *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 178. Pregunta "Abrir libro Piratillas"

En el libro de "Piratillas", indica la sencillez de las interacciones dentro del libro: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 179. Pregunta "Interacciones dentro del libro"

¿Te ha resultado entretenida la lectura de dicho libro? *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 180. Pregunta "Lectura de Piratillas"

Crear un libro: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 181. Pregunta "Crear un libro"

Añadir el libro creado a tu biblioteca: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 182. Pregunta “Añadir libro creado a biblioteca”

Añadirle una portada a tu libro: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 183. Pregunta “Añadir portada al libro”

Empezar a editar tu libro: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 184. Pregunta “Editar libro”

Añadir un título y escribir algo: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 185. Pregunta “Gadget título”

Añadir un texto normal y escribir algo: *

1 2 3 4 5

Poco Mucho

Figura 186. Pregunta “Gadget texto normal”

Cambiar los valores de los anteriores gadgets: *

1 2 3 4 5

Poco Mucho

Figura 187. Pregunta “Cambiar valores de los gadgets”

Añadir una nueva sección al libro: *

1 2 3 4 5

Poco Mucho

Figura 188. Pregunta “Añadir una nueva sección”

Guardar los cambios de las secciones: *

1 2 3 4 5

Poco Mucho

Figura 189. Pregunta “Guardar cambios secciones”

Desplazarte entre secciones (cambiar de una sección a otra): *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 190. Pregunta “Desplazarse entre secciones”

Añadir uno de los gadgets especificado en la tarea 18 y configurarlo: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 191. Pregunta “Añadir gadget”

Previsualizar un gadget: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 192. Pregunta “Previsualizar gadget”

Añadir un gadget de siguiente sección y configurarlo correctamente: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 193. Pregunta “Gadget siguiente sección”

Previsualizar el libro completo: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 194. Pregunta “Previsualizar libro”

Volver a la edición desde la previsualización: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 195. Pregunta “Volver a edición”

Mover los gadgets: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 196. Pregunta “Mover gadgets”

Copiar los gadgets: *

	1	2	3	4	5	
Poco	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Mucho

Figura 197. Pregunta “Copiar gadgets”

Eliminar los gadgets: *

1 2 3 4 5

Poco Mucho

Figura 198. Pregunta “Eliminar gadgets”

Copiar una sección: *

1 2 3 4 5

Poco Mucho

Figura 199. Pregunta “Copiar sección”

En general, interactuar con los distintos gadgets disponibles: *

1 2 3 4 5

Poco Mucho

Figura 200. Pregunta “Interacción con gadgets”

Guardar el libro: *

1 2 3 4 5

Poco Mucho

Figura 201. Pregunta “Guardar libro”

Publicar el libro: *

1 2 3 4 5

Poco Mucho

Figura 202. Pregunta “Publicar libro”

Leer el libro publicado: *

1 2 3 4 5

Poco Mucho

Figura 203. Pregunta “Leer libro publicado”

Eliminar un libro: *

1 2 3 4 5

Poco Mucho

Figura 204. Pregunta “Eliminar libro”

