

BookSwap
Know & Share



TFC DAM 2023-2025

HECHO POR IZAN N. 

21/05/2025



**GENERALITAT
VALENCIANA**

Conselleria de Educaci n, Cultura,
Universidades y Empleo



ÍNDICE

1.RESUMEN DEL PROYECTO	3
1.1 Qué se propone?	4
1.2 De qué está formado?	4
2.JUSTIFICACIÓN Y OBJETIVOS DEL PROYECTO	5
2.1 Para que se propone?.....	5
2.2 A quién se dirige BookSwap?	5
2.3 Qué se pretende conseguir?	5
3.DESARROLLO DEL PROYECTO	6
3.1 Análisis de mercado y modelo de negocio.....	6
3.2 Metodologías utilizadas	7
3.3 Componentes de BookSwap	7
3.4 Problemas, Dificultades y Soluciones en el desarrollo del TFC	10
3.5 Resultados obtenidos	10
3.5.1. LOGIN & REGISTRO	12
3.5.2. PANTALLA HOME	12
3.5.3. PANTALLA MyBOOKS	13
3.5.4. PANTALLA MyEXCHANGES.....	14
3.5.5. MY CHATS	15
3.5.6. SETTING & PROFILE	15
3.5.7. COLECCIONES DE MONGODB.....	16
4.FUTURAS MEJORAS DE PROYECTO	18
1. Validaciones por Correo Electrónico	18
2. Organización de Books, Exchanges, Chats y Messages por fecha y antigüedad	18
3. Filtrar Users, Books, Exchanges, Chats y Messages por palabra a buscar dentro de un buscador.....	18
5.CONCLUSIONES DEL PROYECTO	18
6.BIBLIOGRAFÍAS.....	20

1.RESUMEN DEL PROYECTO

Proyecto de Intercambio y Gestión de Libros para Comunidades Educativas y Sociales

Este proyecto surge de la necesidad de contar con una herramienta eficaz para facilitar el intercambio de libros entre estudiantes, profesores y miembros de una comunidad educativa y/o social. Considerando la importancia del acceso equitativo a recursos educativos y la promoción de una cultura de lectura sostenible, se ha desarrollado BookSwap. Esta aplicación móvil tiene como propósito principal proporcionar una plataforma intuitiva y fácil de usar, que permita a los usuarios ofrecer, buscar e intercambiar libros dentro de su comunidad.

En resumen, BookSwap brinda una solución integral para instituciones educativas y para todos los públicos que desean fomentar el acceso compartido a libros, optimizar el uso de recursos y promover el hábito de la lectura de manera colaborativa y organizada.

Book Exchange and Management Project for Educational and Social Communities

This project arose from the need for an effective tool to facilitate book exchange among students, teachers, and members of an educational and/or social community. Considering the importance of equitable access to educational resources and promoting a sustainable reading culture, BookSwap was developed. The main purpose of this mobile application is to provide an intuitive and easy-to-use platform that allows users to offer, search for, and exchange books within their community.

In short, BookSwap provides a comprehensive solution for educational institutions and all audiences that wish to encourage shared access to books, optimize the use of resources, and promote reading habits in a collaborative and organized manner.

1.1 Qué se propone?

Para dar comienzo a esta memoria del TFC, explicaré cual fue mi elección e idea relacionada.

Personalmente, elegí la creación de una **nueva APP** no relacionada con mi periodo de prácticas, ya que, al estar trabajando en el sector de las aseguradoras y en específico de los bancos, vi muy difícil poder emplear la parte financiera y técnica que todos los trámites necesitan.

He decidido desarrollar una app con múltiples pantallas cuya funcionalidad principal será poder realizar **intercambios de libros** entre gente cercana a tu ubicación **sin ningún coste**. Esta app se llama “BookSwap”.

1.2 De qué está formado?

Como ya he dicho anteriormente, esta app se compone por múltiples pantallas desarrolladas con **React, React-native y Librerías de terceros**, que nos ayudarán a añadir nuestros libros deseados a nuestra “librería de libros”, a tramitar los intercambios con usuarios específicos, a personalizar nuestro perfil de usuario he incluso a poder chatear con los usuarios que se deseen.

Fuera de la parte visual de la app, tenemos **el backend**, el cual constará de una BBDD **NO relacional** (MongoDB) con la cual podremos manejar nuestros datos de manera ágil y sencilla con la ayuda de Sprinboot y nuestra API propia.

2.JUSTIFICACIÓN Y OBJETIVOS DEL PROYECTO

2.1 Para que se propone?

La acumulación de libros ya leídos en los hogares representa un desafío común para muchos lectores: ocupan espacio sin ser reutilizados y, con el tiempo, pierden su valor práctico. Este proyecto surge con el objetivo de ofrecer una alternativa sostenible y funcional a esta problemática, facilitando el intercambio de libros entre personas de diferentes edades e intereses. La implementación de una aplicación interactiva, como BookSwap, permite fomentar una cultura de reciclaje literario, al tiempo que se promueven valores como la sostenibilidad, la inclusión digital y el acceso equitativo a la lectura.

2.2 A quién se dirige BookSwap?

BookSwap está dirigida a un público amplio y diverso, desde niños y jóvenes hasta adultos y personas mayores, abarcando todos los rangos de edad e intereses lectores. La plataforma incluye una variedad de géneros literarios para satisfacer los gustos de cada usuario. Además, durante la creación del perfil y la subida de libros, se podrán personalizar y modificar los datos asociados para facilitar el uso intuitivo y adaptado a cada necesidad, promoviendo así la inclusión digital y la personalización de la experiencia.

2.3 Qué se pretende conseguir?

- **Fomentar la cultura y el acceso a la lectura:**

Estimular el intercambio cultural a través de libros nacionales e internacionales.

Facilitar el acceso a lecturas diversas, independientemente de la situación económica de los usuarios.

- **Impulsar relaciones sociales con intereses comunes:**

Conectar a personas que comparten el gusto por la lectura.

Crear una comunidad colaborativa basada en la confianza y la solidaridad.

- **Promover la sostenibilidad y el reciclaje literario:**
 - Reducir el consumo de libros nuevos mediante el intercambio.
 - Disminuir el impacto ambiental asociado a la producción editorial masiva.
- **Garantizar una experiencia intuitiva y funcional:**
 - Diseñar una interfaz accesible para todos los niveles de experiencia digital.
 - Permitir una navegación fluida y una gestión eficiente del perfil y los libros ofrecidos.

Este proyecto no solo pretende resolver un problema práctico relacionado con el almacenamiento de libros, sino que también promueve valores fundamentales como la sostenibilidad, la inclusión y el acceso igualitario a la cultura. BookSwap busca transformar la forma en que compartimos conocimiento, fortaleciendo los lazos comunitarios y facilitando el acceso a la lectura para todos.

3.DESARROLLO DEL PROYECTO

3.1 Análisis de mercado y modelo de negocio

· **Análisis de mercado:** En años recientes, creció el interés por leer, sobre todo con grupos en redes sociales, como BookTok o Bookstagram. Aunque lo digital avanza, el libro en papel sigue siendo querido entre lectores, sobre todo de 18 a 45 años. Pero muchos guardan libros ya leídos, sin darles otro uso.

A la vez, importa más consumir de forma responsable y sostenible, lo que ayuda a plataformas como BookSwap que impulsan reutilizar.

Además, hay sitios para comprar y vender libros usados (tipo Wallapop o Milanuncios), pero no son solo para libros ni facilitan el trueque. Ahí entra BookSwap: un grupo centrado en libros y en que circulen gratis al intercambiarlos.

BookSwap se ve como algo bueno para el planeta, la cultura y la sociedad, frente a comprar libros como siempre.

- **Modelo de negocio:** Aunque BookSwap empieza como un trueque gratis entre usuarios, hay ideas para ganar dinero a futuro:

- Sistema Premium: usar gratis con lo básico, o pagar para tener más opciones como: buscar mejor, recibir consejos a tu gusto, envíos rápidos o saber dónde va tu intercambio.

- Anuncios enfocados: meter anuncios discretos de editoriales, librerías, eventos de libros o cosas que le gusten al usuario.

- Mercado interno: además de intercambiar, dejar comprar y vender libros entre usuarios de forma segura, cobrando poco por cada venta.

- Asociaciones con casas editoriales o librerías: BookSwap podría ser como una vitrina curiosa para esos sellos pequeños o autores algo nuevos, que buscan dar a conocer sus trabajos en un grupo de gente que ama leer.

3.2 Metodologías utilizadas

- **Metodología Scrum:**

Para el desarrollo de BookSwap, he adoptado una metodología ágil, concretamente la metodología Scrum.

Esta elección se debe a la necesidad de mantener reuniones regulares con el tutor del proyecto y cumplir con entregas parciales de diferentes Sprints.

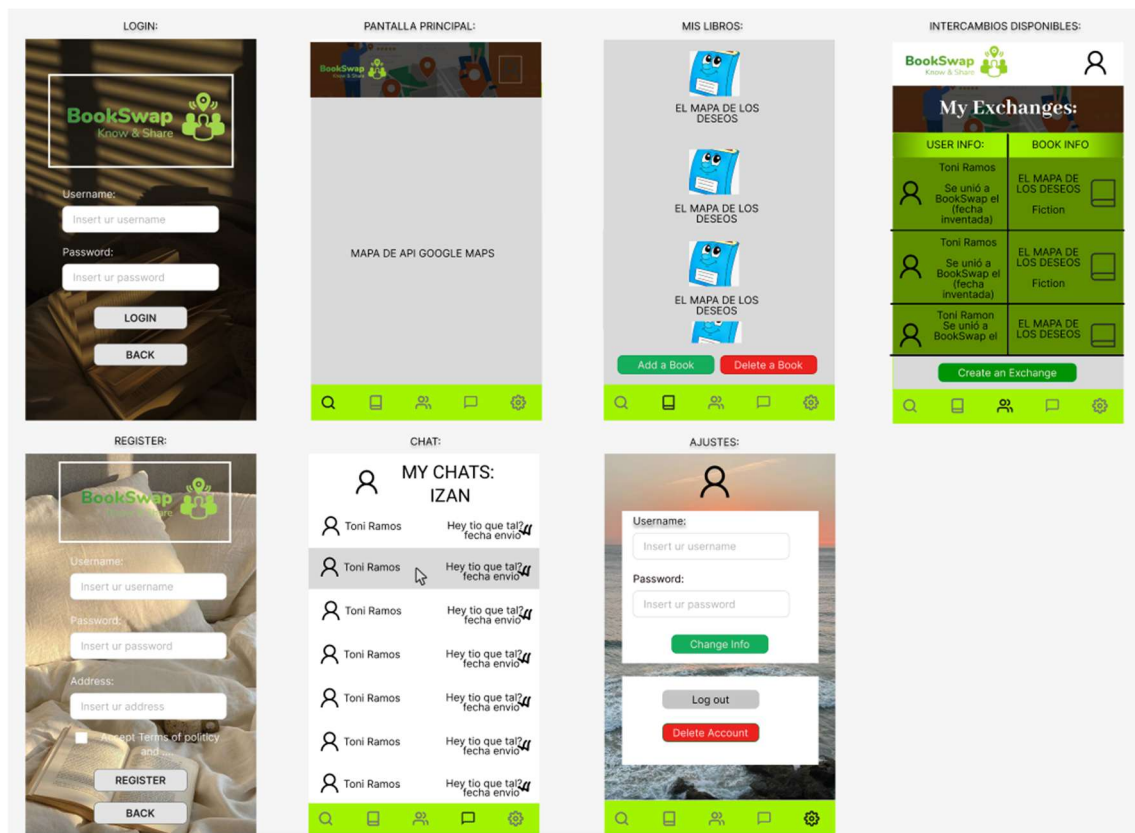
La gestión de tareas se realizó con documentos de .txt que realizaba la función de “Product Backlog”, permitiendo priorizar las tareas críticas para el funcionamiento del proyecto y aquellas que aportan mayor valor global. Esta metodología asegura una adaptación continua y mejora incremental del proyecto.

3.3 Componentes de BookSwap

- **MOCKUP:**

Estos mockups fueron desarrollados desde la web “Figma” al comienzo del TFC. Gracias a estos bocetos pude orientar bien el desarrollo de las pantallas del proyecto, aunque

obviamente hubo cambios de estructuración. Figma me ayudo a optimizar mis métodos de navegación.



· ARQUITECTURA:

Para el desarrollo de BookSwap, opté por utilizar una **arquitectura hexagonal** (también conocida como arquitectura de puertos y adaptadores). Esta decisión se basó en su capacidad para organizar el sistema de forma clara, separando el núcleo de la lógica de negocio (por ejemplo, la gestión de libros e intercambios) de los detalles técnicos como la base de datos o la API web.

Aunque este proyecto es algo pequeño, usar esta arquitectura fue muy beneficioso porque:

- Permite que la lógica de intercambio de libros esté **aislada** y sea **fácil de probar**.
- Hace que sea más sencillo cambiar o agregar tecnologías externas (como otro tipo de base de datos o una interfaz diferente) sin afectar la lógica principal.
- Ayuda a mantener el código limpio y modular, facilitando futuras mejoras o escalabilidad.

· **BACKEND:**

Para el backend, opté por utilizar los conocimientos adquiridos en el curso, implantando así el framework de Java “Spring Boot” para crear la API.

Con Spring Boot se manejan las peticiones por HTTP de manera más sencilla y eficiente de desarrollar, a la vez que nos permite ahorrar tiempo.

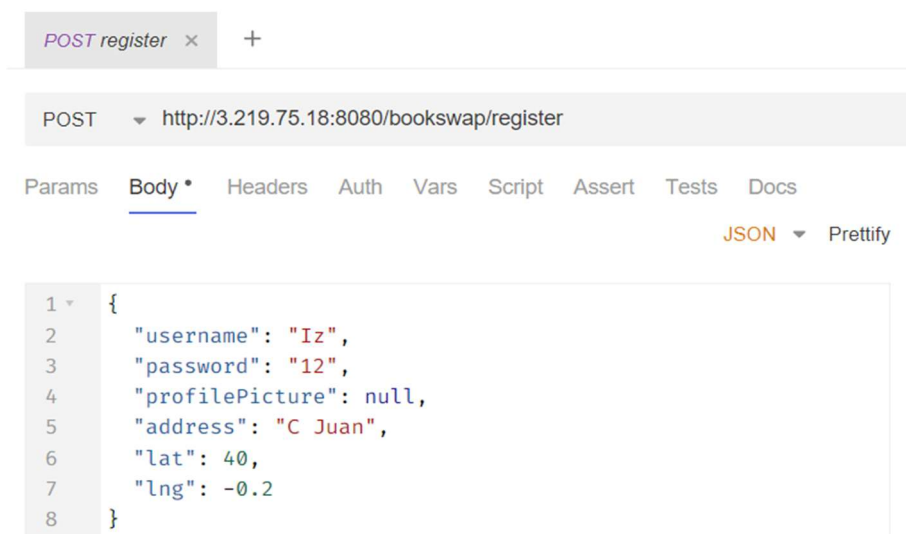
MongoDB: Opté por utilizar una BBDD no relacional para así poder integrarla con Spring Boot y así crear las colecciones de una forma mucho más sencilla. Con ello aproveché la flexibilidad y escalabilidad para manejar los datos.

Despliegue en AWS: Se desplegó la aplicación en un servidor EC2 de AWS, lo que me permitió poder hacer pruebas desde un servidor Mongo externo, con una IP estática que configuré y además permitiéndome así tener una infraestructura robusta y flexible.

· **FRONTEND:** El FrontEnd se desarrollo mediante la web de “Expo Snack” desde la cual pude probar en tiempo real todos y cada uno de los cambios que iba añadiendo al proyecto y además, aprovechando la facilidad de descarga de dependencias y ventajas que ahorran tiempo contra un proyecto de Android Studio.

· **TECNOLOGIAS UTILIZADAS:**

-Postman: Utilizado para probar las peticiones a la API de forma rápida y sin necesidad de desarrollar la aplicación móvil completamente



-**Figma:** Herramienta para crear mockups estéticamente agradables y fáciles de modificar, ayudando a planificar la interfaz de usuario de la aplicación.

3.4 Problemas, Dificultades y Soluciones en el desarrollo del TFC

• **GESTIÓN DEL TIEMPO:**

Durante el desarrollo del proyecto, intenté organizarme lo mejor posible dividiendo las tareas por semanas. Aun así, en algunos momentos me costó seguir el ritmo que tenía planeado, sobre todo cuando surgían problemas técnicos que no esperaba o cuando se me acumulaban las tareas con las prácticas y el segundo trabajo (de fines de semana).

Esta experiencia me ha servido para darme cuenta de lo importante que es planificar con margen y ser flexible cuando las cosas no salen como uno quiere.

• **USO DE API GEOCODING Y MAPA EN EL FRONTEND:**

Para mostrar la ubicación de los usuarios y facilitar los intercambios en BookSwap, utilicé una API de geocodificación que convierte direcciones en coordenadas (latitud y longitud). Luego, esas coordenadas las mostré en un mapa dentro del frontend. Al principio me costó un poco entender cómo conectar todo, pero una vez configurado, resultó muy útil para darle al usuario una mejor experiencia visual y práctica.

• **DIFICULTAD IMPLANTACIÓN MONGO EN AWS:**

Montar MongoDB en Linux dentro de un servicio EC2 de AWS, fue uno de los retos más complicados del proyecto.

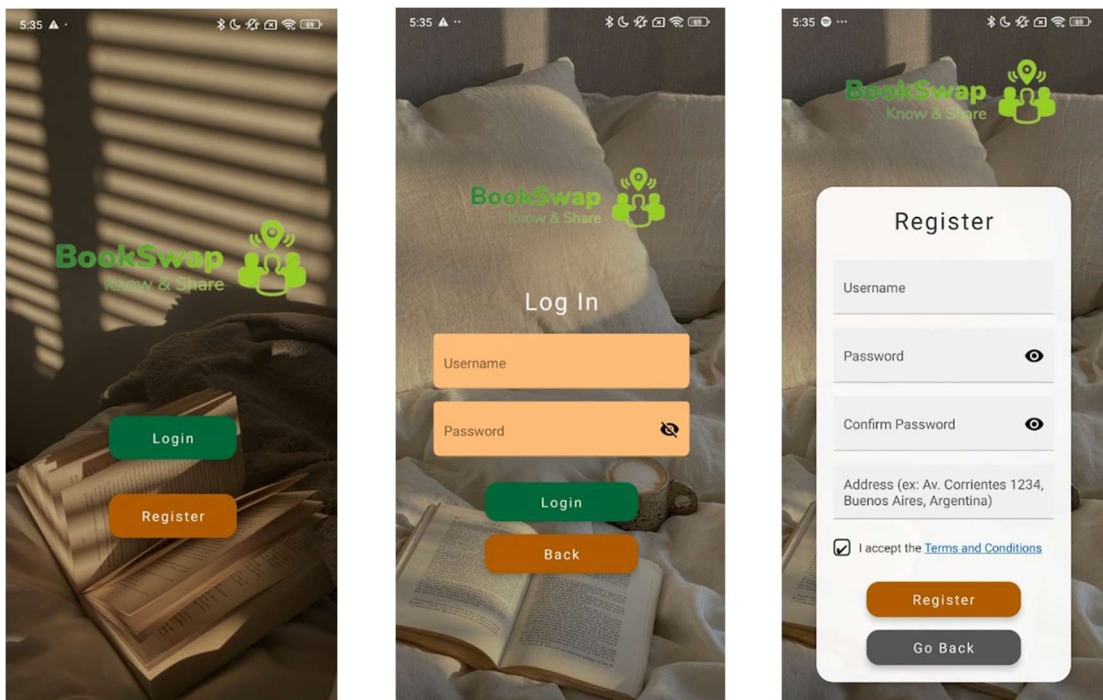
Aunque hay documentación, al principio me costó entender bien los pasos para conectarme desde mi aplicación. Tuve que aprender a manejar cosas como la configuración de seguridad, puertos, y acceso remoto. Después de varios intentos y pruebas, conseguí que funcionara, y fue una buena manera de entender cómo funciona el despliegue en la nube.

3.5 Resultados obtenidos

BookSwap se ha desarrollado con éxito, cumpliendo con el objetivo principal de facilitar el intercambio de libros entre usuarios de forma sencilla y organizada. La aplicación permite gestionar publicaciones, solicitudes de intercambio y ubicaciones de

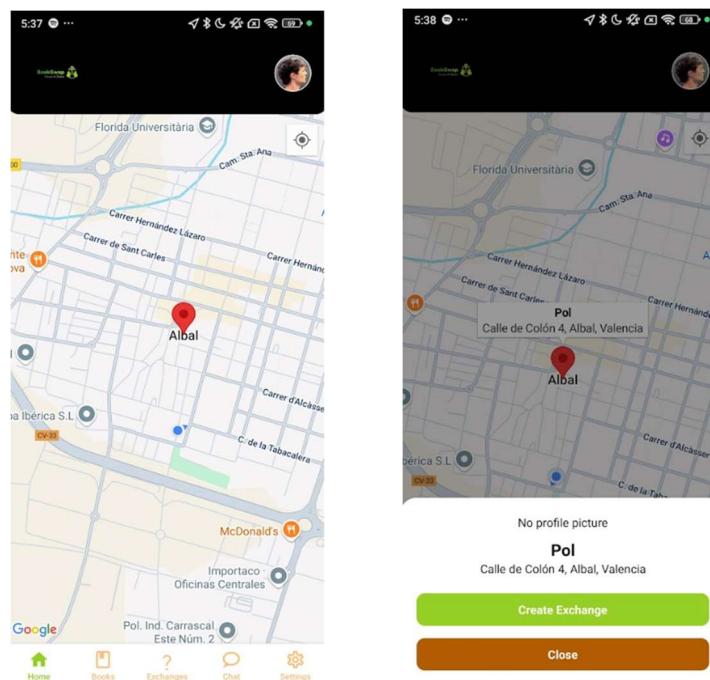
los usuarios desde una misma plataforma. Gracias al uso de buenas prácticas y herramientas modernas, el desarrollo ha sido fluido y he podido aplicar lo aprendido durante el curso. En general, estoy satisfecho con el resultado final, ya que la app funciona correctamente y ofrece una solución útil para fomentar el intercambio de libros de segunda mano.

3.5.1. LOGIN & REGISTRO



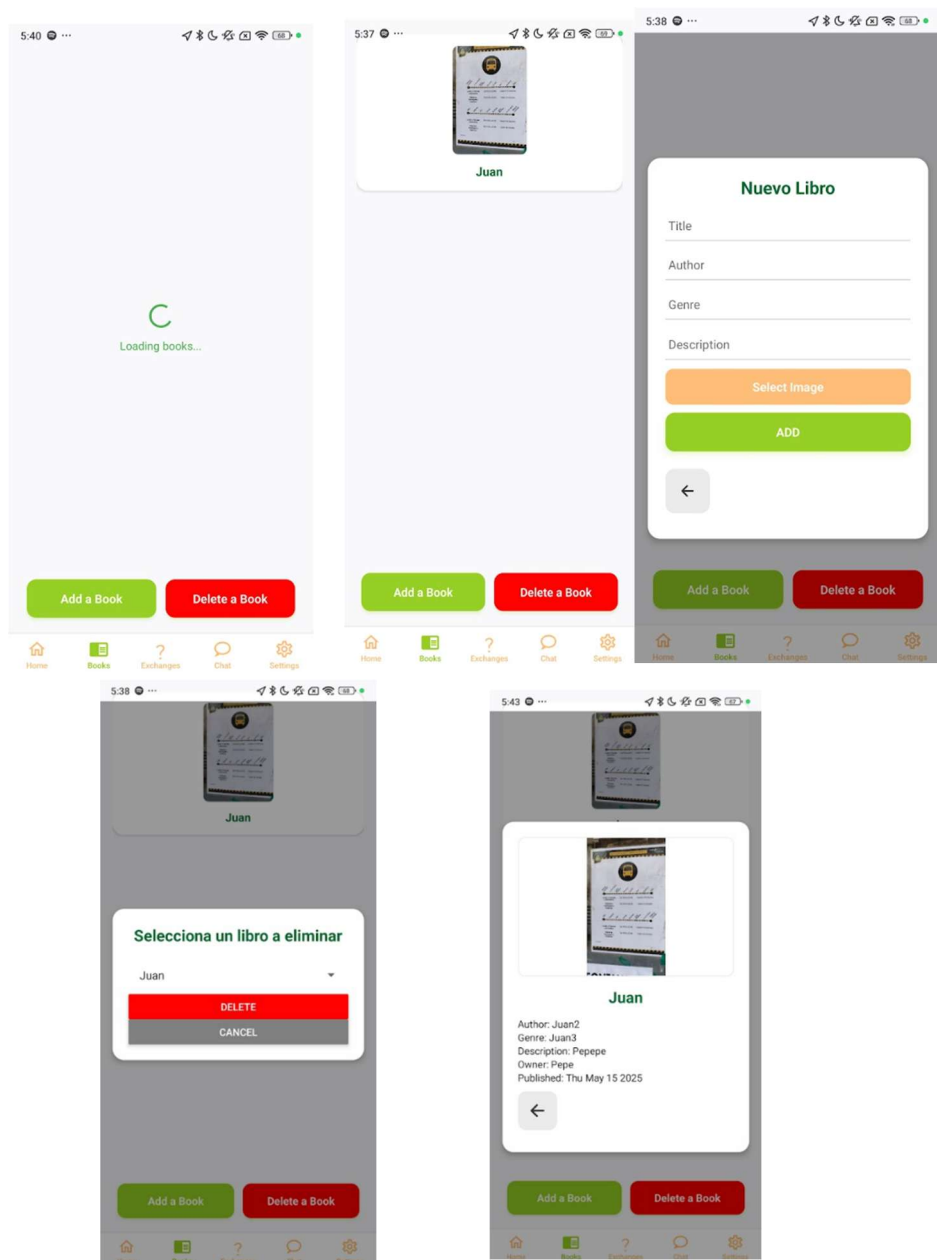
Pantallas donde los usuarios se loguean a la aplicación y/o se registran.

3.5.2. PANTALLA HOME



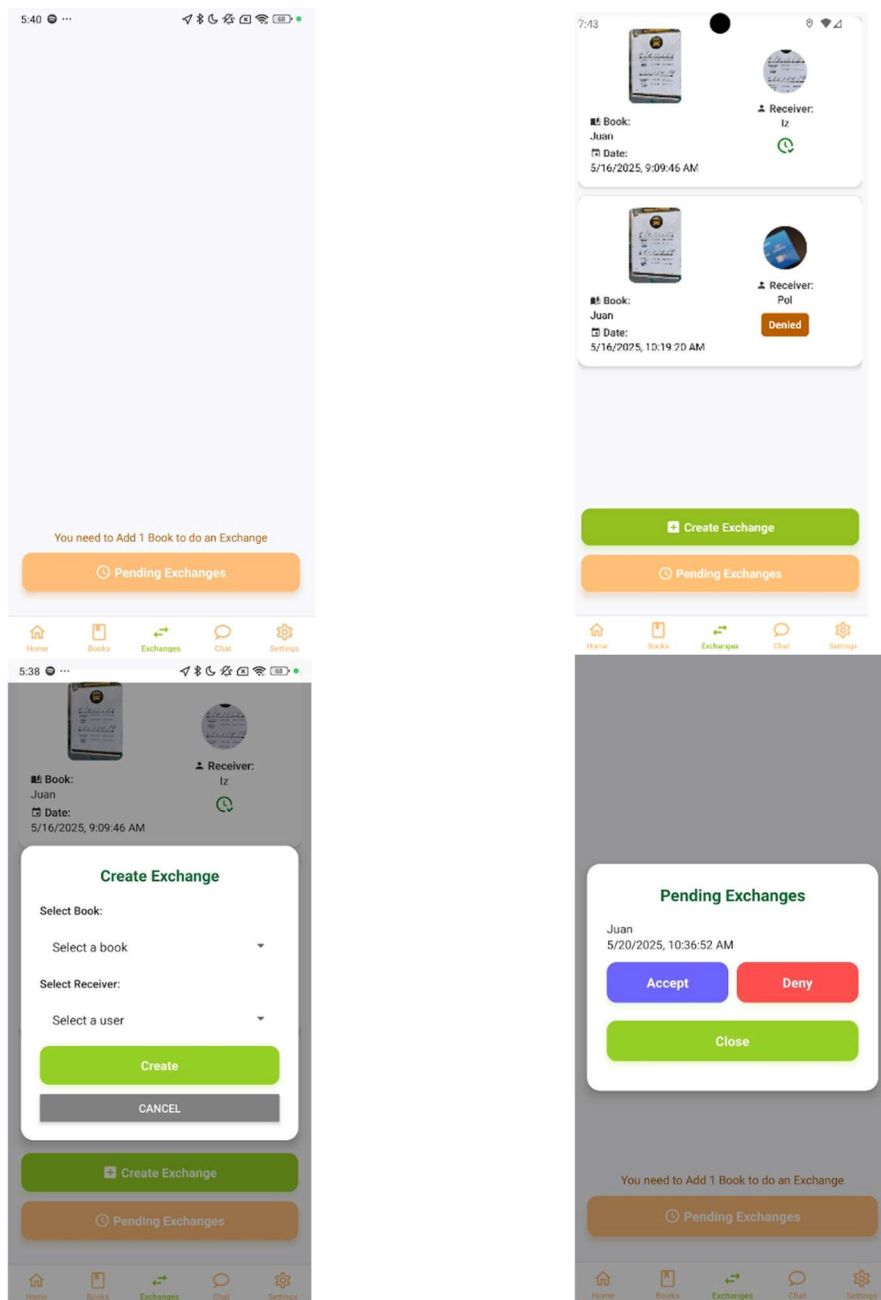
Pantalla que nos mostrará nuestra ubicación actual y la de los usuarios registrados por sus coordenadas (sacados por geocodificación en el backend).

3.5.3. PANTALLA MyBOOKS



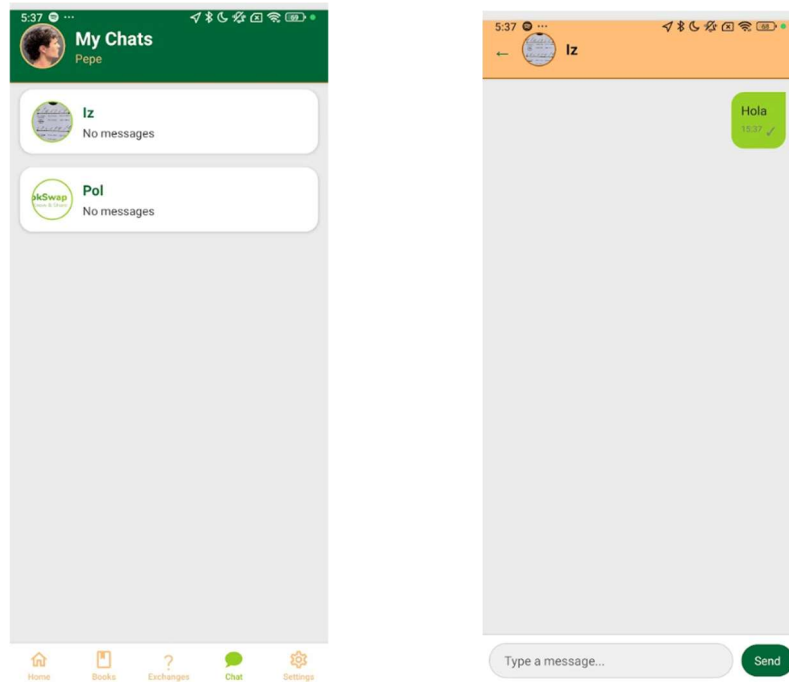
Pantallas en las que los usuarios pueden ver detalles de sus libros, Crearlos y borrarlos.

5.3.4. PANTALLA MyEXCHANGES



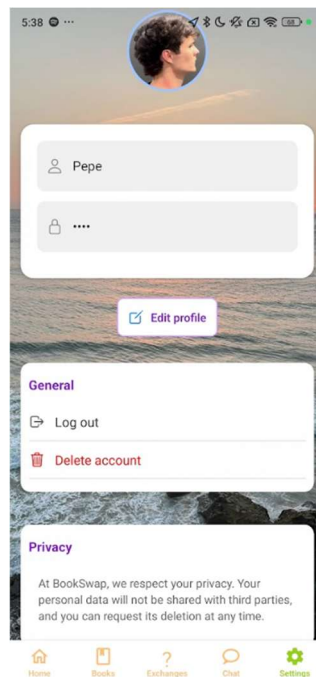
Pantallas en las que los usuarios comprueban sus intercambios creados, su estado y sus intercambios pendientes de aceptar o rechazar.

5.3.5. MY CHATS



Pantalla donde los usuarios podrán chatear con todos los usuarios existentes dentro de la BBDD. Una vez el otro usuario receptor entre al chat, los iconos se pondrán en doble tick como si fuese un WhatsApp.

5.3.6. SETTING & PROFILE



Pantalla de Ajustes donde los usuarios verán su información, podrán editarla, cerrar sesión o incluso borrar su cuenta.

5.3.7. COLECCIONES DE MONGODB

La connection String que uso para ver los datos que se van añadiendo, borrando y/o modificando a la BBDD es:

BookSwap

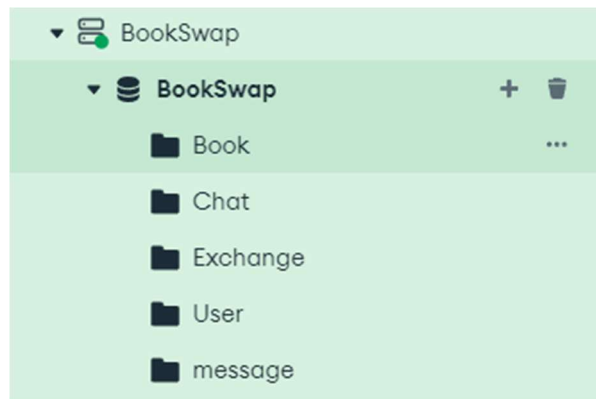
Manage your connection settings

URI ⓘ

Edit Connection String ☐

```
mongodb://admin:*****@3.219.75.18:27017/BookSwap?authSource=admin
```

Y las colecciones que se crean con todas las funcionalidades usadas son:



USER:

```
_id: ObjectId('68248a1995be333218d7b90')
username: "Iz"
password: "6b51d431df5d7f141cbececcf79edf3dd861c3b4069f0b11661a3eeefacbbba918"
profilePicture: "/9j/4AAQSkZJRgABAQAAQABAAQ/4gIoSUNDX1BST0ZJTEUAAQEAIAAIYAAAAAQAABtn..."
address: "C Juan"
lat: 40
lng: -0.2
_class: "bookswap.model.entity.User"
```

Hasheando la contraseña, imagen en base64 y id autoasignado por Eclipse, coordenadas cargadas y añadidas de la API de geocoding.

BOOK:

```
_id: ObjectId('68266a5a79281a739d444395')
title: "Juan"
author: "Juan2"
genre: "Juan3"
description: "Pepepe"
image_url: "data:image/jpeg;base64,/9j/4QxnRXhpZgAATU0AKgAAAQADAEEAAMAAAAABCMwAAAE..."
ownerUsername: "Pepe"
publicationDate: "2025-05-15"
_class: "bookswap.model.entity.Book"
```

Id autoasignado, imagen del libro en base64, fecha del día de hoy añadida automáticamente.

EXCHANGE:

```
_id: ObjectId('682700da805c054d0779722b')
bookId : "68266a5a79281a739d444395"
ownerId : "682633e6a81db64677bced57"
receiverId : "68248a1995be333218d7b90"
status : "accepted"
exchangeDate : "2025-05-16 09:09:46"
_class : "bookswap.model.entity.Exchange"
```

Id autoasignado, asignando los libros y los usuarios por su id correspondiente anulando las similitudes, estado comienza en “pending” y cuando el usuario acepta o rechaza, el estado cambia (mediante endpoint), Fecha y hora actuales de la creación.

CHAT:

```
_id: ObjectId('68270db38d875f5f158a29f8')
participants : Array (2)
  0: "Pepe"
  1: "Iz"
creationDate : "2025-05-16"
_class : "bookswap.model.entity.Chat"
```

Chat con id autoasignado, array de participantes y fecha actual de creación (por si en el futuro se quisiese mostrar por antigüedad).

MESSAGE:

```
_id: ObjectId('68270d688d875f5f158a29f5')
message_id : "fd8e39d7-f300-4605-9bd7-01b6759f49fb"
chatId : "68270d618d875f5f158a29f4"
senderId : "68270d538d875f5f158a29f2"
receiverId : "682633e6a81db64677bced57"
content : "Eyy pepe es prueba esto"
timestamp : "2025-05-16T10:03:20.782665001"
status : "delivered"
_class : "bookswap.model.entity.Message"
```

Mensaje con Id autoasignado, cogiendo el chatId, sender y receiverId como referencia y con un estado “delivered” y cuando el usuario receiverId acceda al chat, el estado del mensaje cambiará a “viewed” y por lo tanto se mostrarán 2 ticks. Fecha actual en tiempo y fecha del envío del mensaje (por si en un futuro se quisieran mostrar los mensajes de una fecha específica).

4.FUTURAS MEJORAS DE PROYECTO

Para un posible futuro desarrollo de BookSwap, diría que podemos focalizarnos en puntos muy distintos y los cuales mejorarían de forma gratificante la Experiencia del Usuario (UX) y su Interfaz (UI).

1. Validaciones por Correo Electrónico

-Registro: Implementar una validación por correo electrónico al registrarse. Los usuarios recibirán un código de verificación en su correo para completar el proceso de registro, aumentando así la seguridad.

-Cambio de Contraseña: Añadir validación por correo electrónico al intentar cambiar la contraseña, asegurando que solo el usuario legítimo pueda realizar este cambio.

2. Organización de Books, Exchanges, Chats y Messages por fecha y antigüedad

Permitir dentro de la interfaz la opción de **ordenar** los Libros y los Intercambios **por antigüedad**, permitiendo así a un usuario confirmar cuando fue la primera vez que hizo un intercambio y sobre que fecha fue.

Dentro de la pantalla MyChats, permitir a un usuario ordenar los chats por chat más reciente/antiguo e igual para los mensajes dentro de una fecha seleccionada dentro del Chat.

3. Filtrar Users, Books, Exchanges, Chats y Messages por palabra a buscar dentro de un buscador

Permitir dentro de la interfaz, la opción de poder **escribir dentro de un textInput** y que, de manera simultánea a la escritura, **encuentre los componentes que coincidiesen** con Título de libro (en caso de Book y Exchange), por Nombre (en caso de los Chats) y en contenido de un mensaje (dentro de la pantalla ChatDetails).

5.CONCLUSIONES DEL PROYECTO

Durante el desarrollo de BookSwap he podido poner en práctica muchos de los conocimientos aprendidos durante el ciclo, tanto a nivel de backend como de frontend. Uno de los principales retos ha sido entender y aplicar la **lógica de desarrollo de los endpoints** y la organización y estructuración de cada uno, que al principio fue todo muy lioso, pero conforme iba desarrollando todas las partes me di cuenta que Spring

Boot me ha ayudado mucho a mantener el código ordenado y modular. También tuve dificultades a la hora de implementar **MongoDB en AWS**, especialmente con la configuración de la conexión y la seguridad, pero después de varios intentos logré que la base de datos fuera accesible de forma estable.

Por lo que respecta al frontend, integrar el **mapa con la API de Geocoding** supuso otro reto importante. Me costó entender cómo transformar una dirección en coordenadas y después mostrarlas en el mapa, pero una vez en funcionamiento, ha mejorado mucho la experiencia de usuario. Además, la **gestión del tiempo** ha sido todo un aprendizaje: tuve que adaptar mis planes en varias ocasiones por problemas imprevistos, pero esto me ha servido también para aprender a ser más flexible y realista con los plazos.

En resumen, el desarrollo técnico de BookSwap me ha servido para consolidar conocimientos, descubrir nuevas herramientas y, sobre todo, **entender** mejor cómo organizar **un proyecto real** de principio a fin.

En lo que respecta a lo personal, he de decir que he **sacado mucho fruto** de este proyecto. Con la experiencia en el mundo laboral, siempre he sabido que hasta que no te chocas de frente contra un proyecto/requerimiento que supone una gran novedad/dificultad, nunca aprendes lo suficiente para sacar las castañas del fuego y dar ese paso arriba a un nuevo escalón de la escalera interminable de la vida.

Estoy muy agradecido de mi trabajo y dedicación y sobre todo de la enseñanza que me brindaron los profesores y la cual me ha hecho ser capaz de desarrollar un proyecto como **BookSwap**.

6.BIBLIOGRAFÍAS

CÓMO Crear Una API KEY de Google MAPS →

<https://www.youtube.com/watch?v=DABAEU9xzH0>

Componentes Icons → <https://github.com/oblador/react-native-vector-icons>

Meta Platforms, Inc. (2023). React Native → <https://reactnative.dev/>

MongoDB Community Server →

<https://www.mongodb.com/try/download/community>

Java Spring Boot guides → <https://spring.io/guides/tutorials/rest> & Contenido Florida Oberta.