

Book4Book

Desarrollo de aplicación Android para el intercambio de libros entre usuarios

Javier Sevillano Jiménez

Máster Universitario en Desarrollo de Aplicaciones para Dispositivos Móviles

Pau Dominkovics Coll

Carles Garrigues Olivella

Junio de 2018

00. Índice

01. Contexto y justificación

02. Objetivos

03. Metodología

04. Análisis y diseño

05. Implementación

06. Conclusiones y líneas futuras

07. Demostración

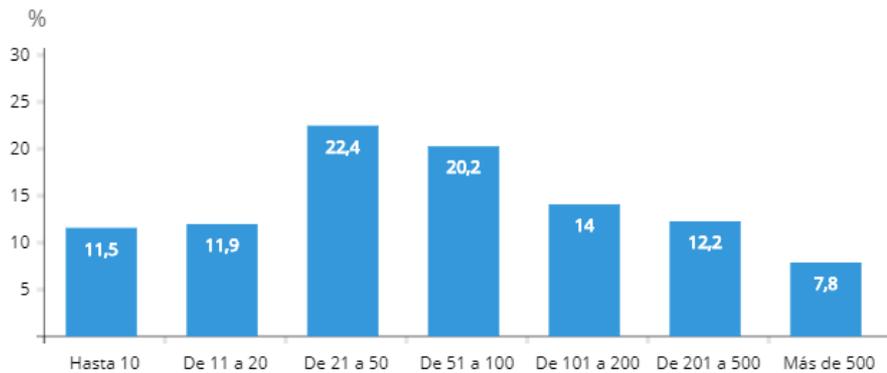
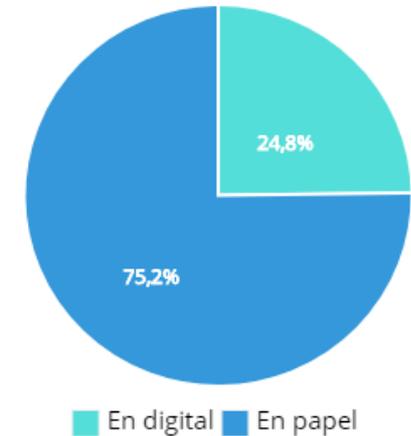
01



Contexto y justificación

01. Contexto y justificación

La lectura como afición goza de buena salud en España. Se lee mucho, según los barómetros y encuestas publicadas, siendo el formato impreso el preferido por los lectores en la inmensa mayoría de los casos.



Los libros en papel ocupan espacio físico y pasan a formar parte de la biblioteca personal. El 54,2% de los lectores acumula más de 50 libros, siendo la media en cada hogar español de 227 ejemplares en 2017.

Muchos de esos libros almacenados quedan en el olvido y no se vuelven a leer, ocupando espacio innecesariamente.

¿Por qué no darles una **segunda oportunidad intercambiándolos con los de otros lectores** en situación similar?



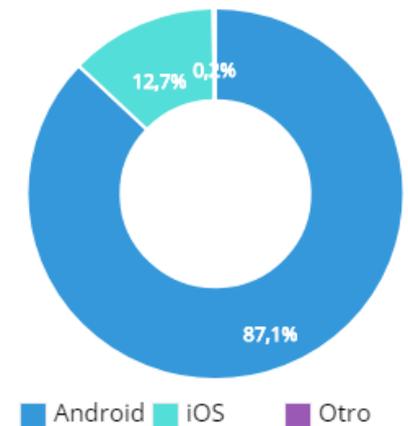
01. Contexto y justificación

Existen herramientas que facilitan el intercambio de libros usados. La mayoría de ellas son sitios web, que ni siquiera tienen un comportamiento responsivo, por lo que resulta complicado utilizarlas en un dispositivo móvil.



Los usuarios se sienten mucho más cómodos usando aplicaciones móviles que sitios web. La gran mayoría del tiempo que pasan utilizando un dispositivo móvil están haciendo uso de una app.

El dominio del duopolio Android-iOS en el universo móvil es total. No obstante, el liderazgo de Android en el mercado Español es indiscutible, con casi 9 de cada 10 dispositivos vendidos en 2017.



01. Contexto y justificación

- Manejando estos datos parece buena idea decantarse por el desarrollo de una aplicación móvil para el sistema operativo Android.
- Siendo, además, una aplicación de tipo colaborativo, cuanto mayor sea la base de personas que la utilicen mayor será también el número de libros que se comparte.



02



Objetivos

02. Objectivos

Generales

- Favorecer la reutilización de libros ya leídos o cuya lectura se ha abandonado.
- Ofrecer a los aficionados a la lectura el acceso a títulos descatalogados o difíciles de encontrar.
- Facilitar el contacto entre personas con afición a la lectura.
- Contribuir al ahorro económico de los usuarios.
- Permitir a los usuarios descubrir nuevos géneros literarios y autores.
- Colaborar en la sostenibilidad del medio ambiente.

Específicos

- Implementar una aplicación móvil que permita el intercambio de libros usados.
- Facilitar la búsqueda de usuarios cercanos geográficamente, para favorecer los intercambios en persona.
- Conseguir intercambios confiables, mediante un sistema de valoración de los usuarios.
- Conseguir un diseño atractivo visualmente y fácil de utilizar por cualquier tipo de usuario.

03



Metodología

03. Metodología

- Requerimientos definidos claramente desde el inicio del trabajo.
- Sin excesivos cambios a lo largo de su ciclo de vida.

Modelo *waterfall*



- Fases ligadas entre sí secuencialmente, la consecución de una depende, de forma irremediable, de la anterior.
- Fundamental el cumplimiento de los hitos intermedios para evitar retrasos finales.

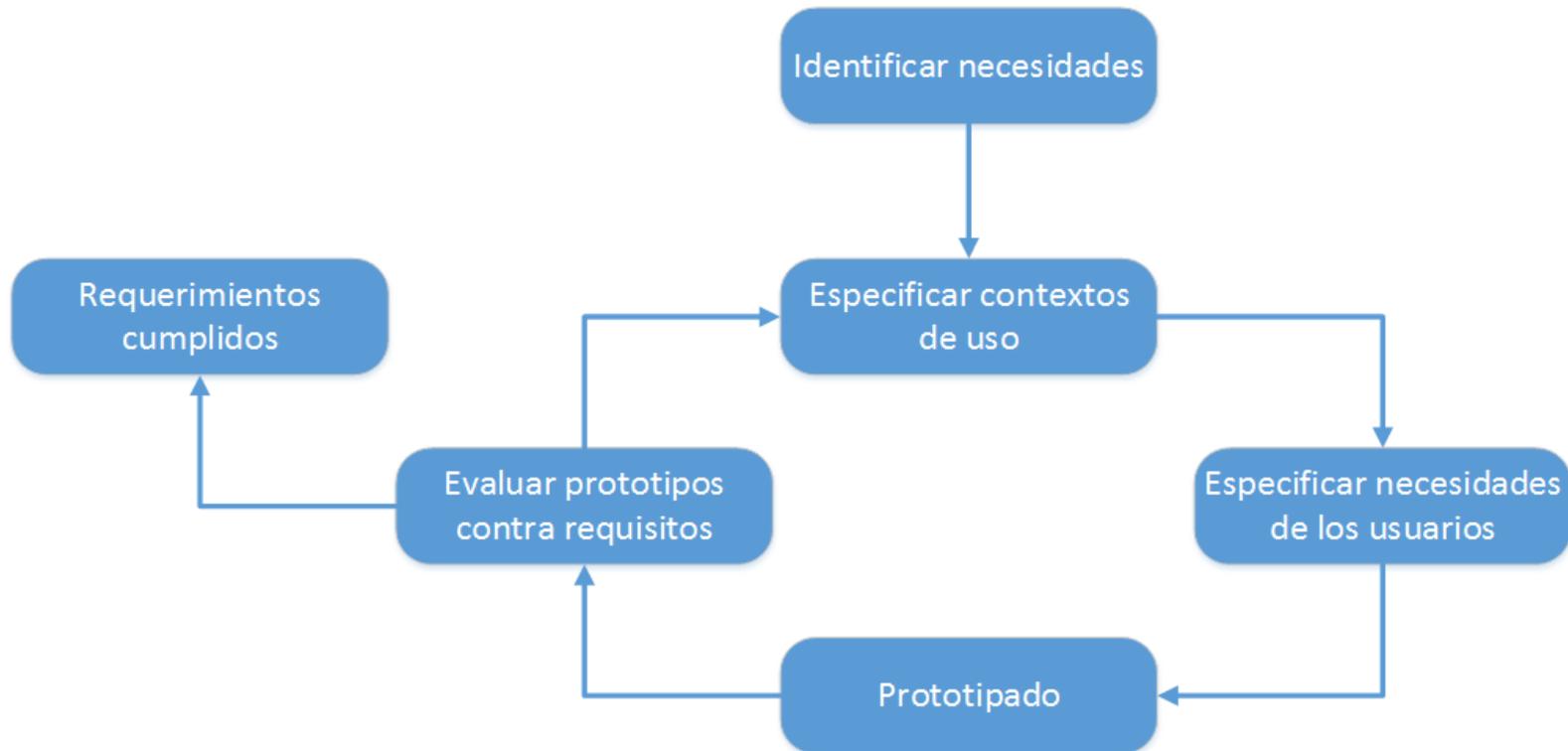
04



Análisis y diseño

04. Análisis y diseño

Se ha empleado un proceso de diseño centrado en el usuario, para dar respuesta a sus necesidades reales y conseguir un producto final de calidad.



04. Análisis y diseño

Especificar contextos de uso

- Determinación de público objetivo.
- Encuesta a usuarios.
- Definición de personas y escenarios.

Especificar necesidades de los usuarios

- Diagrama de casos de uso.
- Descripción de casos de uso.

Prototipado

- Construcción del árbol de navegación.
- Generación de prototipos de baja definición.
- Línea de diseño y subconjunto de prototipos de alta definición.

Evaluación

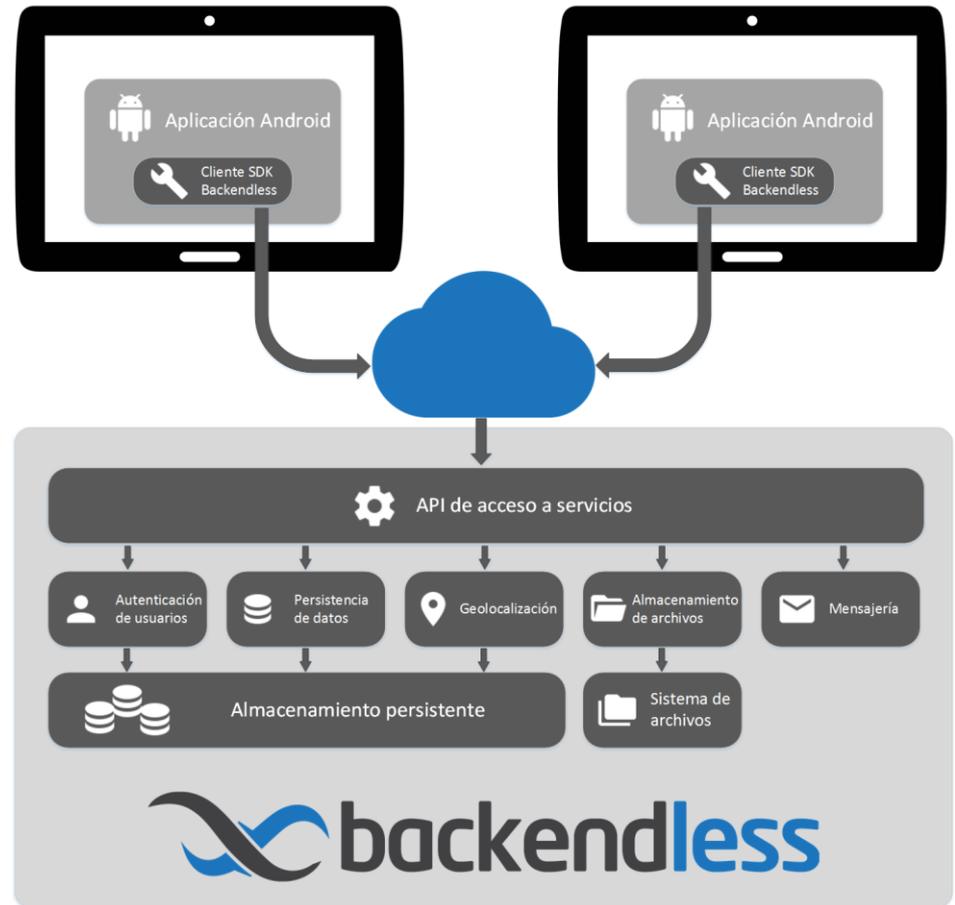
- Evaluación de los prototipos contra los requisitos con la ayuda de los usuarios.
- Iteración del proceso hasta cumplir con los requerimientos.

04. Análisis y diseño

- Arquitectura cliente-servidor.
- Utilización de un servicio *BaaS*, Backendless, para implementar el *backend* de la aplicación.

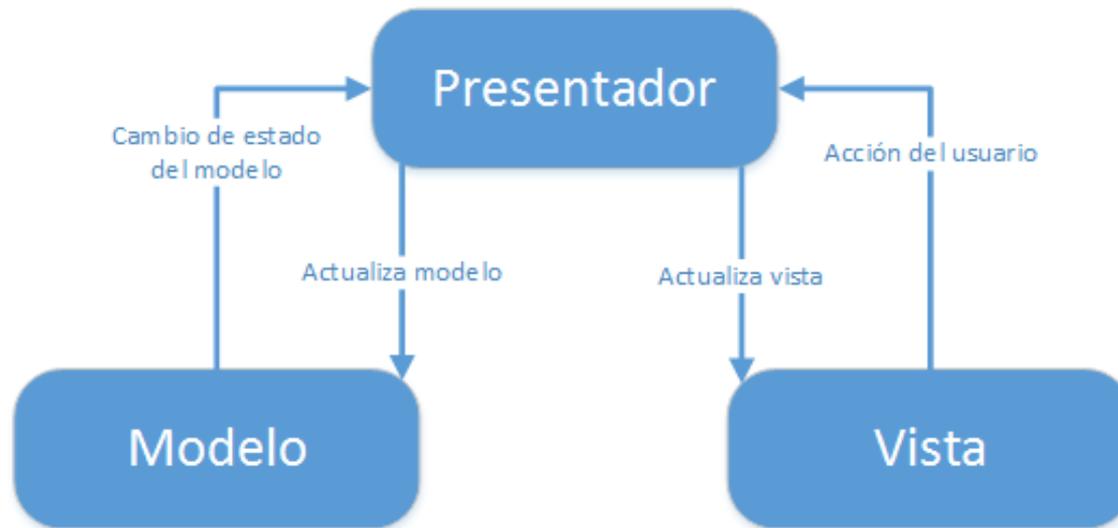


- Evita el despliegue de la infraestructura necesaria para los servidores.
- Ahorra el desarrollo de una API personalizada.
- Permite dedicar menos tiempo al desarrollo del servidor y focalizar esfuerzos en la aplicación.



04. Análisis y diseño

- Separación de responsabilidades entre los componentes mediante el patrón MVP, modelo-vista-presentador.
- Más adecuado que modelo-vista-controlador, al facilitar el desacoplamiento entre vista y controlador en las actividades y fragmentos de Android.
- Modelo: Proveedor de los datos.
- Vista: Muestra los datos del modelo y recoge la interacción con el usuario.
- Presentador: Intermediario entre el modelo y la vista.



05



Implementación

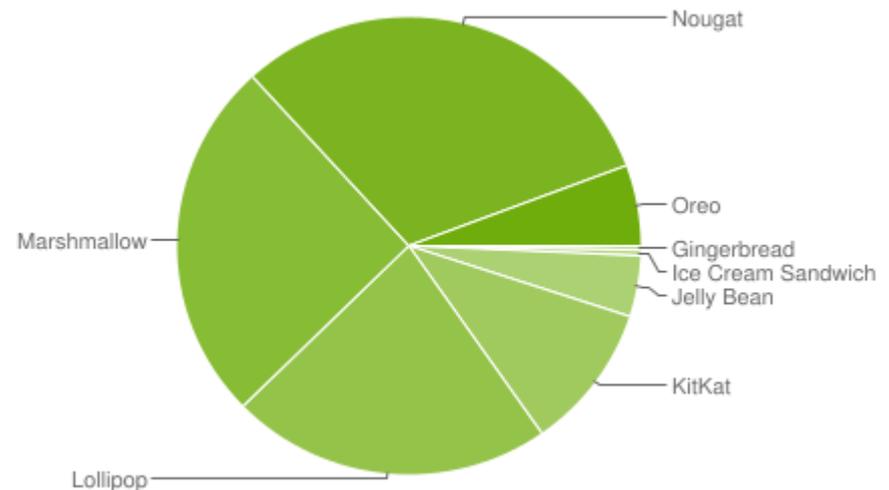
05. Implementación

Desarrollo nativo con el SDK de Android utilizando Android Studio como IDE, para enriquecer la experiencia de usuario y facilitar el acceso a la cámara y la localización del dispositivo.



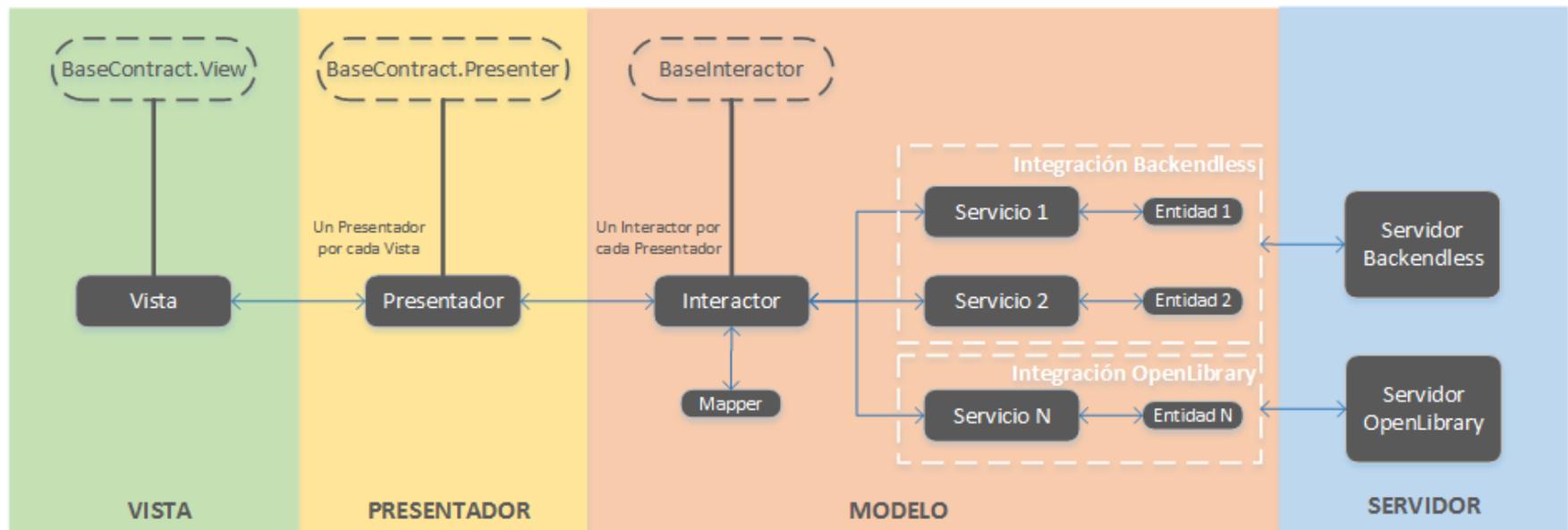
Oreo como versión de Android objetivo (*target*) y KitKat como versión mínima requerida (por dependencia con otras librerías).

Las versiones utilizadas aseguran la compatibilidad con el 95% del conjunto de dispositivos con sistema operativo Android.



05. Implementación

- Implementación de MVP basada en interactores: objetos simples de propósito único que encapsulan la lógica de negocio de un determinado caso de uso.
- Una vista por cada pantalla.
- Un presentador por cada vista.
- Un interactor por cada presentador.
- Servicios que se comunican con el *backend*, uno por entidad.



05. Implementación

Herramientas y APIs utilizadas en el desarrollo de la aplicación:

Backendless SDK

Accede a los servicios ofrecidos por el BaaS Backendless.



OpenLibrary

Obtiene datos de los libros mediante consulta por ISBN.



Retrofit

Cliente HTTP para conectar con la API REST de OpenLibrary.



ZXing Embedded

Escanea el código ISBN de la portada de los libros.



Google Maps

Ubica las búsquedas de usuarios sobre un mapa y determina la ubicación actual del usuario.



Google Places

Permite Introducir la ubicación por defecto del usuario durante el registro.



Picasso

Carga asíncronamente desde el servidor las imágenes de los libros.



Espresso

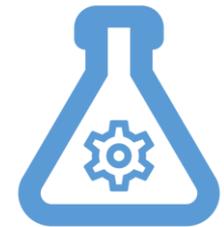
Pruebas automatizadas de la interfaz de usuario de la aplicación.



05. Implementación

El desarrollo ha sido validado a través de diferentes tipos de prueba:

- *Checklist* de pruebas manuales para verificar los aspectos fundamentales de la aplicación.
- Pruebas automatizadas con Espresso para validar determinadas funcionalidades:
 - Autenticación y desconexión de la aplicación.
 - Registro de usuario.
 - Búsqueda de usuarios.
 - Registro y borrado de libros.



Test Results	17s 900ms
edu.uoc.jsev.b4b.ui.LoginActivityTest	17s 900ms
navigateToRegistrationTest	2s 277ms
emailFormatLoginTest	3s 489ms
emptyLoginTest	
validLoginTest	
invalidLoginTest	
TestSuite (edu.uoc.jsev.b4b)	2s 584ms
UtilsTest	1s 514ms
MapUtilsTest	2ms
zoomLevelFromDistance_svgBcn	1ms
zoomLevelFromDistance_svgOdb	0ms
zoomLevelFromDistance_defaultDistance	0ms
distance_svgBcn	
distance_svgOdb	
OpenLibraryTest	1
findBookByIsbn_caيداGigantes	1
Test Results	34s 817ms
edu.uoc.jsev.b4b.ui.RegistrationActivityTest	34s 817ms
fieldsFormatRegistrationTest	6s 436ms
duplicatedRegistrationTest	10s 229ms
emptyRegistrationTest	1s 575ms
validRegistrationTest	13s 688ms
passwordsDontMatchTest	2s 889ms

06



Conclusiones
y líneas futuras

07. Conclusiones

- La metodología *waterfall* ha facilitado que la ejecución del trabajo se haya ajustado a la planificación inicial, sin retrasos en las fechas previstas.
- Objetivos cubiertos en su totalidad. La aplicación desarrollada cumple con los requisitos funcionales y de diseño planteados al inicio del trabajo.
- Principales lecciones aprendidas:
 - Aplicación de los conocimientos adquiridos durante el máster.
 - Separación de responsabilidades en Android utilizando el patrón MVP.
 - Papel fundamental de una buena planificación inicial en la detección temprana y corrección de desviaciones.
 - Importancia de las fases de análisis y diseño en la calidad del producto final.
- La ejecución del trabajo ha supuesto un gran reto en lo personal, al implicar una gran dedicación y esfuerzo y tener que compaginarlo laboralmente.
- Experiencia gratificante y motivadora, al poder materializar una idea propia desde sus etapas iniciales hasta cobrar forma con el producto final.

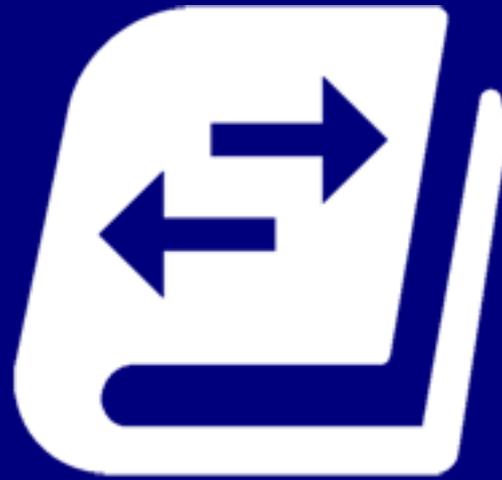
07. Líneas futuras

- Lista de deseos. La aplicación avisaría si algún usuario comparte un libro que se encuentre en ella.
- Implementar parte de la lógica de negocio en el *backend*, para minimizar las llamadas al servidor y permitir operaciones atómicas y más escalables.
- Recordar al usuario, para que no tenga que introducir sus credenciales cada vez que accede a la aplicación.
- Añadir a los libros ofrecidos una valoración o reseña personal.
- Enviar comunicaciones a los usuarios a través de *push*, adicionalmente al email.
- Ofrecer acceso a determinadas funcionalidades sin necesidad de registro previo.
- Adaptar la aplicación para su utilización en tabletas.
- Desarrollar la aplicación para otras plataformas móviles, iOS especialmente.

07



Demostración



book4book

