



## Aplicación móvil para organización de lecturas **Livooks**, una vida completa de libros

**Francisco Villanueva Muñoz**

Máster universitario de Desarrollo de aplicaciones para dispositivos móviles  
Trabajo final de Máster

**Pau Dominkovics Colls**  
**Carles Garrigues Olivella**

28-10-2020



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

## FICHA DEL TRABAJO FINAL

<b>Título del trabajo:</b>	<i>App para organizar lecturas, Livooks</i>
<b>Nombre del autor:</b>	<i>Francisco Villanueva Muñoz</i>
<b>Nombre del consultor/a:</b>	<i>Pau Dominkovics Colls</i>
<b>Nombre del PRA:</b>	<i>Carles Garrigues Olivella</i>
<b>Fecha de entrega (mm/aaaa):</b>	10/2020
<b>Titulación::</b>	<i>Máster universitario de Desarrollo de aplicaciones para dispositivos móviles</i>
<b>Área del Trabajo Final:</b>	<i>Trabajo final de Máster</i>
<b>Idioma del trabajo:</b>	<i>Castellano</i>
<b>Palabras clave</b>	<i>Libros, organizar, lectura</i>
<p><b>Resumen del Trabajo (máximo 250 palabras):</b> <i>Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo.</i></p>	
<p>La lectura es una de las principales aficiones y pasatiempos del ser humano a lo largo de toda su vida. El propósito de este proyecto ha sido obsequiar a los amantes de los libros con un rinconcito donde tener organizadas sus lecturas, e incluso almacenar esas frases o citas que los marcaron para siempre y llevarlas siempre consigo. Así como animarlos mediante retos y estadísticas de su progreso a seguir manteniendo ese hábito tan saludable en un mundo de estrés y falta de tiempo.</p> <p>Para conseguir los objetivos marcados se ha desarrollado una aplicación móvil multidispositivo para el sistema operativo Android utilizando su lenguaje oficial Kotlin. Como servidor Backend se ha utilizado Firebase Firestore y la API pública de Google Books para la búsqueda de libros.</p> <p>Se ha seguido una metodología de trabajo hybrid Agile, combinando el desarrollo tradicional en cascada con los beneficios de la metodología Agile. Pasando por las fases de análisis, diseño, implementación, verificación y mantenimiento, pero dotando al proyecto de flexibilidad e iteración con test de manera continuada a lo largo de todo el proceso.</p> <p>Como resultado se ha obtenido una aplicación funcional, cumpliendo la mayor parte de los objetivos del proyecto. En un futuro está prevista a implementación para el sistema operativo iOS, al igual que más funcionalidades, con lo que la app puede tener gran potencial.</p>	

**Abstract (in English, 250 words or less):**

Reading is one of the main hobbies and pastimes of the human being throughout his life. The purpose of this project has been to give book lovers a little corner where they can organize their readings, and even store those phrases or quotes that marked them forever and always carry them with them. As well as encouraging them through challenges and statistics of their progress to continue maintaining that healthy habit in a world of stress and lack of time.

To achieve the objectives set, a multi-device mobile application has been developed for the Android operating system using its official Kotlin language. Firebase Firestore and the Google Books public API have been used as a backend server for book search.

A hybrid Agile work methodology has been followed, combining traditional cascade development with the benefits of the Agile methodology. Going through the analysis, design, implementation, verification and maintenance phases, but providing the project with flexibility and iteration with continuous testing throughout the entire process.

As a result, a functional application has been obtained, fulfilling most of the project objectives. In the future, it is planned to be implemented for the iOS operating system, as well as more functionalities, with which the app has great potential.

## Índice

1. Introducción.....	1
1.1 Contexto y justificación del Trabajo.....	1
1.2 Objetivos del Trabajo.....	4
1.3 Enfoque y método seguido.....	7
1.4 Planificación del Trabajo.....	10
1.5 Breve sumario de productos obtenidos.....	14
1.6 Breve descripción de los otros capítulos de la memoria.....	14
2. Diseño y análisis.....	15
2.1 Diseño centrado en el usuario (DCU).....	15
2.1.1 Investigación y análisis.....	16
2.1.2 Diseño conceptual.....	20
2.1.3 Diseño del prototipo.....	23
2.1.4 Evaluación.....	30
2.2 Diseño Técnico.....	33
2.2.1 Definición de los casos de uso.....	33
2.2.2 Diseño de la arquitectura.....	42
2.3 Revisión de la planificación.....	46
3. Implementación.....	47
3.1 Herramientas.....	47
3.2 Aspectos destacados del proceso de implementación.....	51
3.3 Funcionalidad y estado de la implementación de la aplicación.....	52
3.4 Plan de pruebas.....	59
3.5 Revisión de la planificación.....	61
4. Conclusiones.....	62
4.1 Resultados obtenidos y aprendizaje.....	62
4.2 Líneas futuras del proyecto.....	63
5. Glosario.....	64
6. Bibliografía.....	66
6.1 Recursos Web.....	66
6.2 Recursos de texto.....	67
7. Anexos.....	68
7.1 Formulario de investigación sobre la lectura.....	68
7.2 Guion de entrevista sobre la lectura.....	73

## Lista de figuras

Figura 1 Gráfico de porcentaje de lectores de en España	1
Figura 2 Cuota de mercado de SO móviles en España - Sept 2020	7
Figura 3 Cuota de mercado Android en España - Sept 2020	8
Figura 4 Metodología en cascada	9
Figura 5 Metodología híbrida	9
Figura 6 Metodología Agile	9
Figura 7 Diagrama de Gantt planificación de tareas TFM	12
Figura 8 Fases diseño centrado en el usuario	15
Figura 9 Aficiones más frecuentes entre los entrevistados.	16
Figura 10 Aficiones más frecuentes entre los entrevistados.	16
Figura 11 Nivel de estudio entre los lectores	17
Figura 12 Dispositivos móviles utilizados	17
Figura 13 Uso principal dispositivos	17
Figura 14 Franja en la que lectores leen	18
Figura 15 Lugar donde lectores leen	18
Figura 16 Porcentaje de lectores que utilizarían aplicación	18
Figura 17 Prototipo de baja fidelidad (Sketching)	25
Figura 18 Prototipo alta fidelidad Pantalla de splash y autenticación	26
Figura 19 Prototipo alta fidelidad pantalla principal Mis Libros	26
Figura 20 Prototipo alta fidelidad listado categoría de libros y detalles	27
Figura 21 Prototipo alta fidelidad búsqueda título o escaneado y detalle	27
Figura 22 Prototipo alta fidelidad pantalla descubrir libros y perfil	28
Figura 23 Prototipo alta fidelidad pantalla retos, logros y evolución	28
Figura 24 Árbol de navegación de la aplicación	29
Figura 25 Búsqueda libro con icono de progreso de búsqueda	30
Figura 26 Diagrama UML de casos de uso "Gestión de usuarios"	33
Figura 27 Diagrama UML de casos de uso "Búsqueda libros"	35
Figura 28 Diagrama UML de casos de uso "Gestionar <i>Mis libros</i> "	37
Figura 29 Diagrama UML casos de uso "Sección descubrir, progreso y perfil"	40
Figura 30 Interrelación del cliente con el servicio Backend y las APIs	43
Figura 31 Representación gráfica arquitectura MVC	43
Figura 32 Asimilación gráfica arquitectura MVC en Android	44
Figura 33 Distribución de las diferentes versiones de Android	47
Figura 34. Esquema del uso de base datos Firestore para la app	48
Figura 35. Registro mediante FirebaseAuth utilizado en Livooks	49
Figura 36. Estructura del proyecto	51
Figura 37. Pantalla de inicio de la aplicación. Splash Screen	53
Figura 38. Pantalla de registro y modelo registro mediante Google y correo	53
Figura 39. Pantalla de recuperación de contraseña	54
Figura 40. Pantalla principal	54
Figura 41. Pantallas de búsqueda tanto manual como mediante escaneo	56
Figura 42. Pantalla detalles del libro buscado	56
Figura 43. Compartir libro con otras personas	57
Figura 44. Categorías libros	57
Figura 45. Pantalla detalles del libro almacecnado	58
Figura 46. Selección de fecha de lectura, añadir y eliminar citas	59
Figura 47. Pirámide del conjunto de test para una aplicación	59

## Lista de Tablas

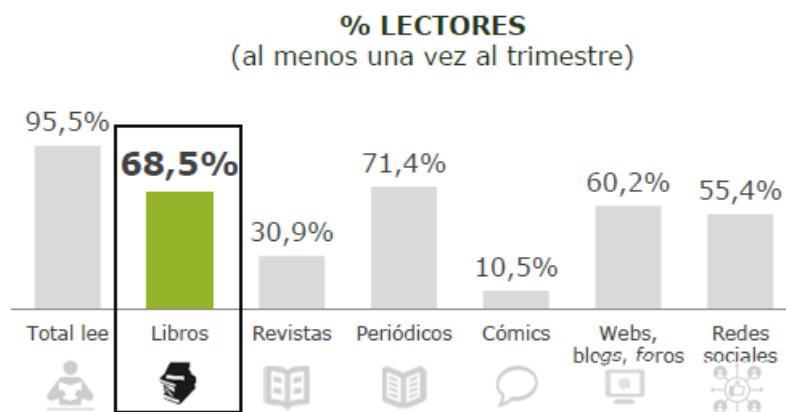
Tabla 1. Análisis de la competencia (Benchmarking)	2
Tabla 2. Planificación de tareas proyecto fin de master	11
Tabla 3. Análisis de riesgos	13
Tabla 4. Tareas a completar por el usuario sometido al Test	32
Tabla 5. Caso 1 - Registro de usuario	34
Tabla 6. Caso 2 – Inicio de sesión	34
Tabla 7. Caso 3 – Recuperar contraseña	34
Tabla 8. Caso 4 – Cierre de sesión	34
Tabla 9. Caso 5 – Buscar libro manualmente	35
Tabla 10. Caso 6 – Buscar libro escaneando código barras	36
Tabla 11. Caso 7 – Detalles libro buscado	36
Tabla 12. Caso 8 – Añadir libro a <i>Mis libros</i>	36
Tabla 13. Caso 9 – Ver los libros de mi biblioteca	37
Tabla 14. Caso 10 – Buscar un libro en una categoría	37
Tabla 15. Caso 11 – Filtrar libros en una categoría	38
Tabla 16. Caso 12 – Ver detalle de un libro de mi biblioteca	38
Tabla 17. Caso 13 – Cambiar de estado (categoría) un libro	38
Tabla 18. Caso 14 – Eliminar un libro	39
Tabla 19. Caso 15 – Compartir libro	39
Tabla 20. Caso 16 – Añadir una cita	40
Tabla 21. Caso 17 – Acceso a la sección descubrir libros	41
Tabla 22. Caso 18 – Acceso al perfil	41
Tabla 23. Caso 19 – Modificar perfil	41
Tabla 24. Caso 20 – Acceso a la sección progreso	41
Tabla 25. Caso 21 – Establecer un reto	41
Tabla 26. Caso 22 – Ver evolución del usuario	42
Tabla 27. Caso 23 – Ver logros	42

# 1. Introducción

## 1.1 Contexto y justificación del Trabajo

**La lectura es una de las principales aficiones y pasatiempos del ser humano.** Según el Barómetro de Hábitos de Lectura, los índices de lectura en España siguen mejorando y el porcentaje de lectores en 2019 alcanzó un 68,7% de la población, que supone un aumento del 1,8 respecto al año anterior [\[1\]](#).

### LECTURA DE LIBROS



**Figura 1 Gráfico de porcentaje de lectores de en España**  
(Fuente: Informe de resultados hábitos de lectura en España 2019)

Por otro lado, **la lectura tiene innumerables beneficios para la salud.** Favorece la imaginación, la concentración y la empatía, e incluso previene la degeneración cognitiva. “Al igual que nos cuidamos y vamos cada vez más al gimnasio, deberíamos dedicar media hora diaria a la lectura”, sostiene el escritor catalán Emili Teixidor, autor de *La lectura y la vida* (Columna) y de la exitosa novela que inspiró la película *Pa negre*. [\[2\]](#)

Partiendo de estas dos premisas, se genera un nicho de mercado inmejorable para responder a cualquier necesidad sobre esta temática tan relevante mediante una aplicación.

Y es que, una de las principales problemáticas que se encuentran los amantes de la lectura es no poder recordar todo lo que han leído a lo largo de su vida. Esas ocasiones en las que buscan en su recuerdo esa frase o cita que les cautivó, de aquel libro que leyeron antaño, pero le es imposible traerlo a su memoria. Igualmente, el lector tiene inquietud por conocer cuál es su evolución a lo largo del tiempo, libros que lee al año, el género que más ha leído y en qué momento temporal de su vida ha sido. Para finalizar, tristemente, las personas se encuentran cada vez más estresadas y les cuesta mantener los hábitos de lectura, necesitando en ocasiones el ánimo de una motivación extra para sacar el hueco de sumergirse en la lectura y absorber sus beneficios.





De esta serie de necesidades nace la idea de crear una aplicación. Los libros que se leen van ocupando un lugar importante a lo largo de nuestra vida, se merecen tener un lugar donde queden organizados. Por ello, se quiere conseguir una **clasificación según el estado en el que se encuentre el libro** (leídos, leyendo, pendientes, e incluso una lista de libros deseados que nos gustaría tener en propiedad). Esta aplicación no pretende ser una base de datos biblioteca de libros en propiedad, se quiere una organización muy personal e íntima de nuestras lecturas.

En la búsqueda de esa personalidad e intimismo se podrán valorar los libros que se han leído, hacer una reseña, e incluso almacenar esas frases o citas que nos marcaron para siempre y llevarlas siempre consigo.

Se proponen retos y consecución de logros, con el objetivo de mantener a los lectores motivados y mejorar sus hábitos de lectura. Así como estadísticas para conocer nuestro progreso a lo largo del tiempo.

**Es por todo ello que la app Livooks pasa a ser de útil a realmente necesaria en nuestro desarrollo cultural.**

Bien es cierto, que hay aplicaciones de mucha calidad a nuestro alcance de forma totalmente gratuita que nos ayudan en la tarea de organizar libros. Como decía Apple en su anuncio *“Hay una aplicación para eso”* [3]. Por lo que identificar nuestra potencial competencia y analizar sus fortalezas y debilidades resulta imprescindible.

	Puntos Fuertes	Puntos débiles
<p><b>Mi Biblioteca</b> [4]</p> 	<p>Gratuita y sin publicidad</p> <p>Biblioteca virtual donde almacenar libros</p> <p>Agrega libros manualmente o escaneando código barras</p> <p>Exportar su biblioteca dentro de un archivo Excel</p>	<p>Mala traducción</p> <p>Se centra sobre todo en organizar libros en propiedad</p> <p>No tiene opción de sincronización para guardarlo en la nube</p> <p>No tiene modo oscuro</p>
<p><b>Goodreads</b> [5]</p> 	<p>Gratuita</p> <p>Agrega libros manualmente o escaneando código barras</p> <p>Gran base de datos con su propia API</p> <p>Recibe recomendaciones personalizadas en función de tus gustos.</p> <p>Incluye red social en la que se puede ver lo que tus amigos están leyendo, discutir libros a través de mensajes, reseñas y grupos.</p> <p>Actualizaciones frecuentes</p>	<p>Mala traducción</p> <p>Muy centrada en red social</p> <p>No tiene modo oscuro</p>

**Tabla 1. Análisis de la competencia (Benchmarking)**

Aunque muchas de las principales funcionalidades pensadas en un primer momento para la aplicación Livooks se encuentran en estas aplicaciones de referencia. Siempre hay espacio para la innovación y la mejora.

No deja de sorprender, que las traducciones de las aplicaciones sean tan poco cuidadas. Es una parte muy importante de las que se quejan muchos usuarios. Por lo que tenemos un punto a favor, ya que, al ser en idioma español, puede ser muy bien acogida por gran parte del público hispanohablante.

Se echa en falta en las aplicaciones analizadas, ese punto de enganchar al lector para que use la app a diario. **Livooks** hará uso del concepto de '**gamificación**', ya que combinará el placer de leer con la superación de retos y la obtención de logros que engancha al usuario y lo motiva a seguir continuando la experiencia. Además, el usuario se sentirá a gusto consigo mismo al ver el progreso en sus hábitos de lectura mediante estadísticas. Incluso una pequeña meta puede conllevar una gran satisfacción.

Como valor añadido de nuestro producto, se ha querido avanzar un paso más allá y no quedarnos solamente con la parte organizativa, ni tratar los libros como una mera biblioteca. Se pretende conseguir un toque de '**emocionalidad**'. Se buscará personalizar la app de tal manera que se visualice como un rinconcito íntimo de nuestras lecturas, donde tendremos como función característica almacenar citas memorables de los libros que vamos leyendo, poniendo cierto orden en nuestra memoria lectiva.

## 1.2 Objetivos del Trabajo

Una vez se ha tenido clara la temática de la aplicación y la necesidad a cubrir. Se marcarán los objetivos a cumplir para que todas las necesidades queden satisfechas, llevando a buen término el proyecto y la aplicación pueda salir a la luz:

- 1) Ofrecer al usuario un lugar donde tener organizados los libros que ha leído, los que está leyendo, los que tiene pendientes y una lista de deseos de libros a adquirir que podrá ser compartida.
- 2) Tener acceso a una base de datos externa (mediante API), donde poder encontrar cualquier libro mediante un buscador.
- 3) Utilizar la solución de Google Firebase para la autenticación de los usuarios (Firebase Authentication)
- 4) Ser capaz de agregar libros escaneando su código de barras o reconociendo el ISBN mediante la cámara del móvil.
- 5) Conseguir dar a la aplicación un sentido emocional. Poder valorar los libros leídos y hacer una reseña, así como almacenar las citas favoritas y poder consultarlas siempre que se quiera.
- 6) Mediante técnicas de gamificación como retos, animar a los usuarios a mejorar sus hábitos de lectura. Pudiendo observar su progreso a lo largo del tiempo.
- 7) Poner en práctica los conocimientos adquiridos a lo largo del master y desarrollar la aplicación desde cero hasta producción, subiéndola a Google Play para su descarga.

En esta etapa se definirán al detalle las funcionalidades que la aplicación debe poseer para alcanzar las metas propuestas y cumplir con el cometido para el que ha sido creada:

### Requerimientos funcionales

- **RF1. El acceso a la aplicación.** Estará limitado a los usuarios registrados, permitiendo la creación de nueva cuenta a quien no lo esté de forma sencilla, bien manualmente o mediante las principales redes sociales, como la cuenta de Google.
- **RF2. Organizar libros en categorías según su estado:** leyendo, leídos, pendientes, lista de deseados. Posibilidad de agregar nuevos libros, eliminarlos o cambiar de categoría.
- **RF3. Almacenar** citas favoritas de nuestras lecturas.

- **RF4. Valoración y reseña.** Posibilidad de valorar el libro y realizar una reseña de lo que nos ha parecido tras marcar como leído.
- **RF5. Buscador y filtro por categorías.** Permitirá a los usuarios encontrar los libros en los que estén interesados utilizando diferentes criterios: nombre, autor, género...etc. añadirlo a la colección y añadirle un estado.
- **RF6. Escaneo código barras.** Las mejores aplicaciones son las que facilitan el trabajo a los usuarios. Para la búsqueda de libros se podrá utilizar la cámara del dispositivo móvil, agilizando así la entrada de datos, mediante el escaneo del código de barras del libro o detector de su ISBN.
- **RF7. Estadísticas.** Llevar estadísticas interesantes sobre nuestras lecturas. Libros leídos al año, comparativa respecto años anteriores...etc.
- **RF8. Retos.** Para ayudar a mejorar los hábitos de lectura y de forma motivadora, se propondrán retos, con los que se conseguirán logros a desbloquear en la aplicación.
- **RF9. Compartir** libros mediante las principales redes sociales o las citas que tenemos guardadas sobre éstos.

Debido al alto riesgo de falta de tiempo para la realización del proyecto se comentan estos requerimientos como opcionales y poderlos implementar en un futuro.

#### **Requerimientos funcionales (Opcionales)**

- **RFO1. Ajustes.** Personalizar notificaciones, modo oscuro de la app.
- **RFO2. Notificaciones.** Notificaciones Push, con las últimas novedades de libros y noticias relacionadas con el mundo literario.
- **RFO3. Club de lectura.** Donde se propondrá un libro mensualmente y los usuarios irán comentado sobre él.
- **RFO4. Personificación del sistema.** Una mascota que nos guíe en los retos y que podamos personalizar.

#### **Requerimientos no funcionales**

- **RNF1. Usabilidad.** La usabilidad de la aplicación debe estar enfocada al usuario final y al dispositivo en el que se usará, es decir, un dispositivo móvil. La navegación, menús, acciones y estética debe ser acorde a la

plataforma en la que se está ejecutando, de forma que sea lo más familiar posible al usuario.

- **RNF2. El funcionamiento** debe ser intuitivo. La curva de aprendizaje será suave. Las metáforas y el uso de iconografía facilitarán la comprensión de las funcionalidades a la vez que fomenta su exploración.
- **RNF3. Uso de test automatizados de pruebas y validación.** Se desarrollarán tests a lo largo de la programación de la app.
- **RNF4. Lanzamiento en producción.** Se tratará y codificará para lanzamiento en producción en la tienda Google Play.
- **RNF5. El sistema será robusto y fiable.** Sin generación de errores que nos puedan hacer perder información.
- **RNF6. Mensajes de error.** El sistema deber proporciona mensaje informativo cuando no sea posible realizar alguna acción o conectar con la API.

### 1.3 Enfoque y método seguido

Resulta muy difícil ofrecer una receta mágica para el desarrollo de aplicaciones móviles y, por tanto, se hace imprescindible adaptar los métodos y los conocimientos adquiridos. **Este proyecto desarrolla un producto nuevo**, valorando las estrategias de diseño más adecuadas y utilizando las buenas prácticas y recomendaciones que se han ido viendo a lo largo del master.

El enfoque de desarrollo será el de una **aplicación nativa multidispositivo**. Con el enfoque nativo se ofrece el mejor rendimiento al acceder a las APIs y funciones del dispositivo, así como una mayor integración, apostando por una experiencia de usuario de calidad. Además, facilita al usuario una experiencia multidispositivo pudiendo usar la app tanto en móvil como en tableta, siempre procurando una transición transparente, consistente y coherente entre ellos.

Una vez seleccionados los dispositivos para los que se va a desarrollar la aplicación, nos encontramos ante la disyuntiva de para qué sistema operativo implementarla. Lo ideal sería hacerlo para las plataformas Android e iOS, ya que como se aprecia en el gráfico entre las dos se cubre más del 99% del mercado, siendo despreciable el porcentaje de los demás sistemas operativos [6]. En este caso, se propone **Android como sistema operativo**, ya que tiene el mayor número de usuarios potenciales, lo cual interesa en esta app gratuita, en el que su objetivo es que pueda beneficiarse de ella el mayor número posible de usuarios. (En un futuro está previsto el desarrollo de la app para la plataforma iOS)

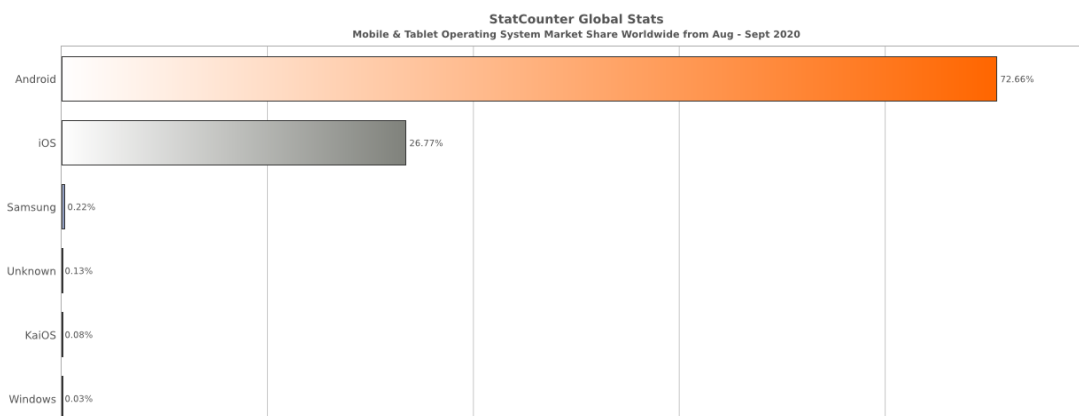
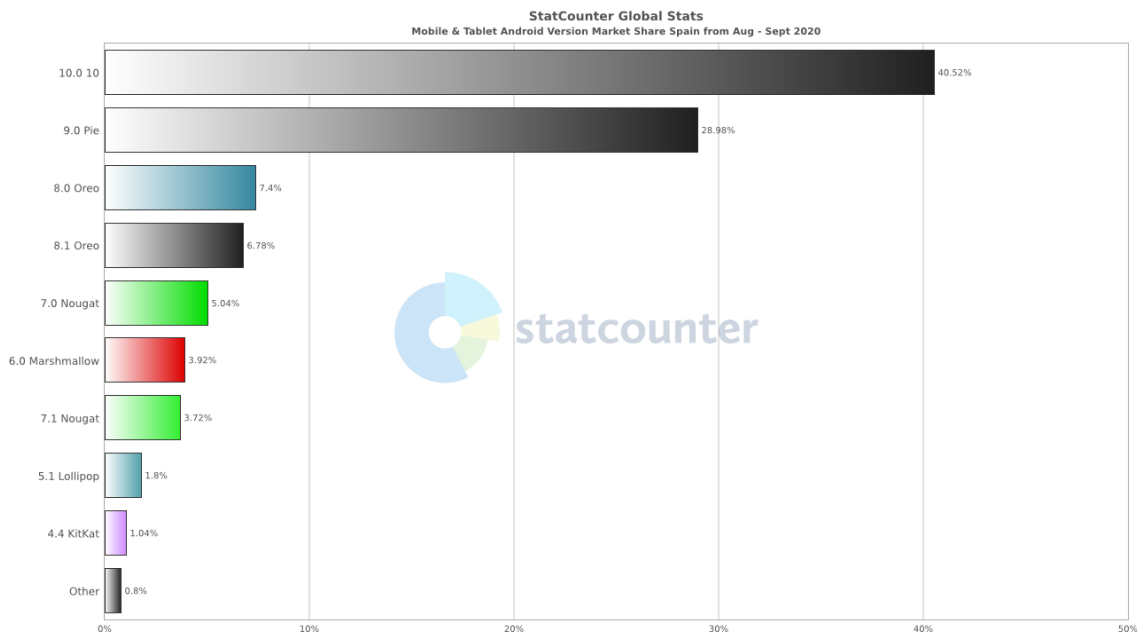


Figura 2 Cuota de mercado de SO móviles en España - Sept 2020  
(Fuente: Statcounter GlobalStats)

Como entorno de desarrollo para poder programar en el sistema operativo elegido, **se utilizará el IDE Android Studio**, ya que es el entorno de desarrollo de aplicaciones nativas Android soportado oficialmente por Google. Igualmente, el lenguaje oficial de programación Android desde 2017 pasó a ser Kotlin en detrimento de Java. Se está creando una enorme comunidad a su alrededor. De hecho, según las estadísticas se encuentra entre los 5 lenguajes más queridos por los usuarios debido a su simplicidad mezclada con potencia [7]. La

interoperabilidad con Java es absoluta, por lo que el riesgo que presenta la adopción de Kotlin es muy bajo si se compara con sus ventajas. Estos hechos han sido los que han hecho decantarme por **Kotlin como lenguaje de programación para llevar a cabo la implementación de la app.**

Debido al alto grado de fragmentación de Android [8] (aunque cada año va siendo menor), se ha tenido que decidir la versión de Android para la que se va a desarrollar la app. Para ello se han tenido en cuenta las funcionalidades de la app y las versiones más recientes de Android, para encontrar un equilibrio entre novedad e índice de mercado.



**Figura 3 Cuota de mercado Android en España - Sept 2020**  
(Fuente: Statcounter GlobalStats)

Como se puede ver en la gráfica, en España ya está implantado Android 10 (API29) en el 40% de los móviles Android y por debajo de la versión 4.4 (KitKat API19) es prácticamente despreciable. **Con lo que se desarrollará para la API29 teniendo como versión mínima la API21** siendo compatible con la gran mayoría de dispositivos teniendo en cuenta las funcionalidades de la app.

Como se ha comentado en el apartado anterior una vez arranque la aplicación será requerido el registro del usuario para lo cual será necesario comunicarse con un servidor remoto con el que intercambiar datos donde almacenar los perfiles de los usuarios. Debido a mi desconocimiento en la parte de servidor, se ha decidido utilizar la **plataforma Google FireBase como servicio de Backend** por su sencillez de uso y gestión de usuarios, incluso a través de cuentas de redes sociales.

Finalmente, los equipos de trabajo, han propiciado la aparición de diferentes metodologías a la hora de abarcar los requerimientos de un proyecto y dar respuesta a sus necesidades. En este caso va a ser una sola persona la que se encarga del conjunto del proyecto con lo que **se ha optado desarrollar una metodología híbrida entre un desarrollo en cascada y un desarrollo Agile (Hybrid Agile)** [9].

Dado que en este caso los requerimientos del trabajo están fijados desde un inicio, encajaría perfectamente con una metodología de desarrollo en cascada, donde se irán realizando entregables secuencialmente. Por otro lado, el equipo está formado por un solo individuo, el cual, conoce la estructura y las necesidades del proyecto, los planes de entrega, así como las técnicas de prototipado y de testeo, muy apto para un desarrollo Agile. Donde el testeo y la iteración será fundamental ya que seguro, habrá que hacer ciertas modificaciones respecto los objetivos fijados en un principio siempre en comunicación con el tutor. La idea es seguir un método centrado en el usuario que no sea tan estático y predictivo como el modelo en cascada. Así se dota de flexibilidad al proyecto, pero siguiendo unos plazos de entrega con unos hitos marcados como veremos más adelante que sí se deberán respetarse.



Figura 4 Metodología en cascada



Figura 6 Metodología Agile

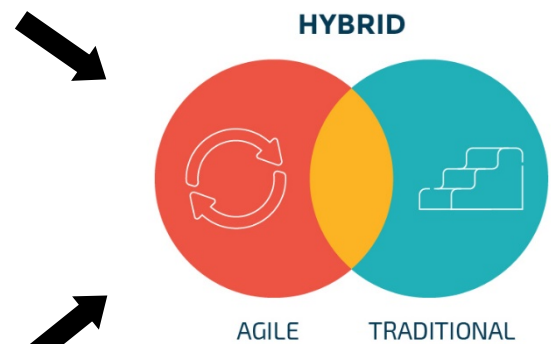


Figura 5 Metodología híbrida

(Fuente: Open International)



## 1.4 Planificación del Trabajo

Una vez se tiene claro el enfoque y la metodología que se va a seguir durante el proyecto, es importante conocer los recursos con los que se cuenta para poder llevarlo a cabo.

En cuanto a los recursos físicos necesarios disponibles se tiene:

- **Equipo portátil HP OMEN i7-7700** con sistema operativo Windows 10. Disco duro SSD y 16 gigas de memoria RAM.
- **Teléfono móvil Motorola G5 Plus** con 4 gigas de memoria RAM y 32 gigas de almacenamiento. La opción de desarrollo activada.

En cuanto a Software necesario disponible se tiene:

- **Paquete de Office2016**, para este proyecto se utilizará Word, MS Project y PowerPoint.
- **IDE Android Studio** actualizado a la última versión.
- **Draw.io**, para la creación de diagramas de flujo y mapa de navegación.
- **Git**, para el control de versiones.
- **Figma**, para prototipado de alta calidad
- **Adobe PhotoShop 2016**, para tratamiento de imágenes.

En cuanto al recurso horas-hombre para la realización de la planificación temporal, vendrá determinado en función del tiempo que se le pueda dedicar a lo largo del cuatrimestre y los hitos parciales de cada una de las PEC.

En cuanto a la dedicación personal se ha establecido lo siguiente:

- **De lunes a viernes: dedicación de 4 horas diarias**
- **Sábados y Domingos: dedicación de 8 horas diarias**

En cuanto al calendario de la asignatura se debe respetar los hitos parciales de cada una de los entregables, que serán los siguientes.

- **PEC 1:** del 16/09/2020 al 07/10/2020
- **PEC 2:** del 08/10/2020 al 28/10/2020
- **PEC 3:** del 29/10/2020 al 09/12/2020
- **Entrega final:** del 10/12/2020 al 30/12/2020

Partiendo de estas premisas se ha elaborado la siguiente planificación temporal de cada tarea utilizando un diagrama de Gantt que nos ayudará a tener una

visión global del proyecto y seguir las tareas de manera eficaz evitando retrasos.

**Según la planificación estimada, el número de horas totales que se dedicará al proyecto serán de 536 horas.**

ACTIVIDAD	INICIO	FIN	DURACION
<b>TRABAJO FIN DE MASTER</b>	<b>16/09/20</b>	<b>30/12/20</b>	<b>536 horas</b>
<b>PEC1 - PLAN DE TRABAJO</b>	<b>16/09/20</b>	<b>07/10/20</b>	<b>112 horas</b>
Selección de la temática y justificación	16/09/20	20/09/20	22 horas
Análisis de la competencia	20/09/20	22/09/20	12 horas
Objetivos del trabajo y requerimientos	22/09/20	26/09/20	20 horas
Enfoque y método seguido	26/09/20	30/09/20	20 horas
Planificación	30/09/20	04/10/20	24 horas
Análisis de riesgos	04/10/20	06/10/20	8 horas
Revisión de la memoria para entrega	06/10/20	07/10/20	6 horas
<b>Entrega PEC1</b>	<b>07/10/20</b>		
<b>PEC2 - DISEÑO</b>	<b>08/10/20</b>	<b>28/10/20</b>	<b>108 horas</b>
Diseño centrado en el usuario	08/10/20	10/10/20	12 horas
Perfiles de usuario	10/10/20	13/10/20	18 horas
Escenarios de uso	13/10/20	17/10/20	16 horas
Mapa de navegación	17/10/20	19/10/20	16 horas
Prototipado	19/10/20	25/10/20	28 horas
Evaluación prototipado	25/10/20	25/10/20	6 horas
Revisión de la memoria para entrega	26/10/20	26/10/20	4 horas
Margen para desviaciones	27/10/20	28/10/20	8 horas
<b>Entrega PEC2</b>	<b>28/10/20</b>		
<b>PEC 3 - IMPLEMENTACION</b>	<b>29/10/20</b>	<b>09/12/20</b>	<b>216 horas</b>
Acceso a API base datos libros	29/10/20	31/10/20	14 horas
Autenticación usuarios mediante Servidor Firebase	31/10/20	03/11/20	18 horas
Codificación Interfaz búsqueda libros	04/11/20	09/11/20	32 horas
Codificación Interfaz escaneo código barras e ISBN	10/11/20	15/11/20	32 horas
Codificación Organización de lecturas	16/11/20	21/11/20	22 horas
Codificación detalles libros, valoración reseña y citas	21/11/20	23/11/20	16 horas
Codificación retos y estadísticas progreso	23/11/20	28/11/20	26 horas
Ejecución iterativa de pruebas	29/11/20	02/12/20	20 horas
Revisión de la memoria para entrega	03/12/20	05/12/20	16 horas
Margen para desviaciones	06/12/20	09/12/20	20 horas
<b>Entrega PEC3</b>	<b>09/12/20</b>		
<b>ENTREGA FINAL</b>	<b>10/12/20</b>	<b>30/12/20</b>	<b>100 horas</b>
Revisión de la memoria para entrega final	10/12/20	12/12/20	12 horas
Elaborar presentación	12/12/20	20/12/20	45 horas
Creación de video presentación	20/12/20	26/12/20	23 horas
Edición manual de usuario	27/12/20	28/12/20	10 horas
Subir la aplicación a la Play Store	28/12/20	30/12/20	10 horas
<b>Entrega final</b>	<b>30/12/20</b>		

**Tabla 2. Planificación de tareas proyecto fin de master**

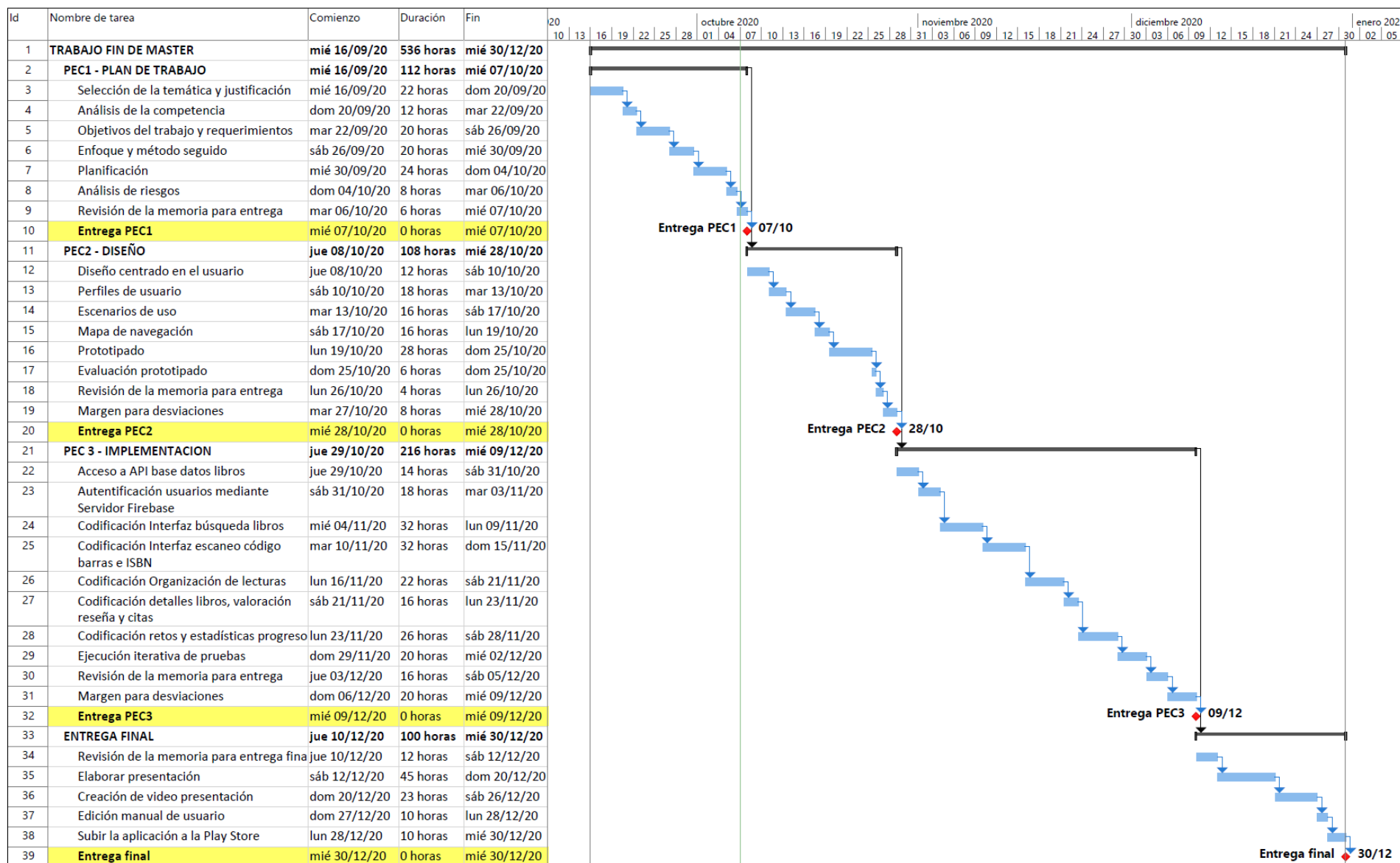


Figura 7 Diagrama de Gantt planificación de tareas TFM

Por desgracia, por muy eficientemente que se haya planificado el proyecto, ésta, podría verse afectada por diversos motivos. La realización de todo proyecto conlleva unos riesgos asociados, que tendrán que ser analizados. Igualmente se debe tener preparado un plan de contingencias para la mitigación de los mismos, si no se quiere que el proyecto fracase. Tras un análisis pormenorizado se han detectado los siguientes riesgos y como mitigarlos:

Riesgo	Descripción	Probabilidad	Impacto	Tipo	Acciones Mitigadoras
<b>Encontrar trabajo</b>	Debido a mi situación actual en paro tengo más tiempo para dedicar al master. El cual podría verse reducido si lo encuentro.	Media	Muy alto	Riesgo mayor	Intentar dar el máximo todos los días para tener un buen colchón en el caso de que tuviera la suerte de trabajar.
<b>Planificación optimista</b>	Error en la planificación, retrasando el cumplimiento de tareas y los hitos marcados del proyecto.	Media	Medio	Riesgo significativo	Control de planificación semanal. Tener siempre colchón para contratiempos. Esfuerzo extra para mitigar desviaciones.
<b>Conocimiento insuficiente</b>	Posibilidad de tratar con tecnología desconocida o poco experimentado en ella.	Media	Medio	Riesgo mayor	Indagar e intentar buscar documentación y tutoriales con los que superar esa barrera. Consultas al tutor en la medida de lo posible.
<b>Problemas de salud</b>	Mientras no haya vacuna estamos expuestos al COVID y el invierno puede ser duro.	Media	Bajo-Medio	Riesgo menor	Seguir las indicaciones de las autoridades sanitarias y respetar confinamiento. Vacuna, si la hubiera.
<b>Traslados temporales</b>	Visitas a casa de mis padres temporalmente ya que vivo fuera de la casa familiar.	Alta	Bajo	Riesgo menor	Llevar el portátil para continuar trabajando y perder las menos horas posibles.
<b>Avería portátil</b>	Avería del medio físico en el que se desarrolla el proyecto	Bajo	Alto	Riesgo Significativo	Disponer de copias de seguridad. Disponibilidad económica para reparación o sustitución lo antes posible.
<b>Pérdida de información</b>	Pérdida de recursos elaborados, ya sea código o documentación.	Bajo	Alto	Riesgo Significativo	Uso de herramientas de versionado como Github. Copias de seguridad casi a diario del proyecto.

**Tabla 3. Análisis de riesgos**  
(Fuente: Elaboración propia)

## 1.5 Breve resumen de productos obtenidos

Para poder alcanzar los objetivos de este trabajo final de grado se deberán obtener los siguientes productos:

- Memoria trabajo fin de master. Versión definitiva de este documento.
- Aplicación multidispositivo nativa para el sistema operativo Android (archivo APK instalable y código de la aplicación en formato comprimido)
- Manual de usuario
- Presentación del proyecto diapositivas formato PowerPoint
- Vídeo con la defensa del trabajo fin de master con demo incluida

## 1.6 Breve descripción de los otros capítulos de la memoria

En este apartado se describen brevemente los capítulos de la memoria que nos ha llevado a conseguir los objetivos de nuestro proyecto a través de la metodología propuesta.

### **Capítulo 2: Diseño**

Este capítulo se dedica al análisis y diseño de la aplicación centrada en el usuario (DCU), así como su arquitectura. Dejando un prototipo de la app preparado para su posterior implementación.

### **Capítulo 3: Implementación**

A partir del prototipo, se prepara este capítulo, donde se desarrolla la codificación de nuestra app. Se detallarán las decisiones tomadas, las APIs utilizadas y los problemas encontrados con la solución adoptada.

### **Capítulo 4: Conclusiones**

Se analizarán los resultados obtenidos comparándolos con los objetivos marcados y se hará una reflexión crítica general sobre el proyecto elaborado y la implementación de mejoras futuras.

### **Capítulo 5: Glosario**

Se ha generado un glosario con las definiciones de términos y acrónimos con la intención de ayudar a la comprensión del documento al mayor número de personas.

### **Capítulo 6: Bibliografía**

Se detallarán las fuentes utilizadas para la elaboración del documento.

### **Capítulo 7: Anexos**

Se expone listado de apartados que son demasiado extensos para incluir dentro de la memoria y tienen un carácter auto contenido.

## 2. Diseño y análisis

### 2.1 Diseño centrado en el usuario (DCU)

Durante la fase de diseño y desarrollo de la aplicación, **se va a colocar al usuario en el centro de todo el proceso**. El objetivo que se pretende lograr con el DCU, es el de conseguir una app que satisfaga las necesidades de sus usuarios, conociendo sus características, necesidades, objetivos y el contexto en el que utilizarán el producto.

Lo lógico es hacer todo lo posible por conocer bien a quienes van a utilizar la app, es decir, una primera fase de **investigación y análisis** de los usuarios. En definitiva, conocer y empatizar con el usuario final, ya que es quien realmente acabará interactuando con el producto. En esta fase temprana, definir correctamente el perfil de usuario, las fichas de persona y los escenarios de uso serán clave para el éxito de la app.

Con esta base sólida de conocimiento sobre el usuario final, se podrá comenzar a tomar **decisiones sobre el diseño de los contenidos** de la app, dando respuesta a los deseos y necesidades investigadas. En esta fase se intentará construir una narrativa en la que el usuario se sienta el protagonista principal. Se comenzará con un prototipo de baja fidelidad para ir escalándolo hasta un alto nivel de detalle respondiendo a la evolución y test con usuarios.

Con el objetivo de detectar rápidamente errores o puntos de mejora en nuestro diseño, se utilizarán técnicas de inspección y test con usuarios. En esta última fase se puede tomar como una **evaluación** para comprobar nuestro diseño encaja realmente con las necesidades del usuario.

Aunque se ha hablado de estas tres fases de forma secuencial, no se deben entender como estancas y sucesivas. La **iteración** juega un papel decisivo, ya que cada fase se enriquece una y otra vez con la respuesta de los usuarios, con lo que modificaremos nuestros diseños y los evaluaremos una y otra vez de forma cíclica hasta obtener el mejor resultado posible.



Figura 8 Fases diseño centrado en el usuario  
(Fuente: Blog UX, la era del usuario)

### 2.1.1 Investigación y análisis

Como se comentó anteriormente, la investigación y análisis nos ayudará a conocer quién y porqué usará la aplicación y cuáles son sus intereses y sus motivaciones. Si no conocemos al usuario de nuestra aplicación, difícilmente podremos dar respuesta a sus deseos y necesidades.

Para tener una primera aproximación a los usuarios se ha optado por la **utilización de encuestas** (cuestionario mediante formulario de google) como método de indagación cuantitativo.

Se puede acceder al cuestionario en el siguiente enlace o ver en el anexo [10]:

<https://forms.gle/7KxP15U6x97iGsyN8>

Se ha comenzado el análisis de las encuestas con las principales aficiones de los encuestados. Como se puede ver en el gráfico **la lectura sigue estando entre las aficiones preferidas**. De hecho, **el 75% de las personas encuestadas leyó más de un libro el año pasado**. Lo cual sugiere que hay un amplio nicho de mercado de usuarios que puede utilizar nuestra app.

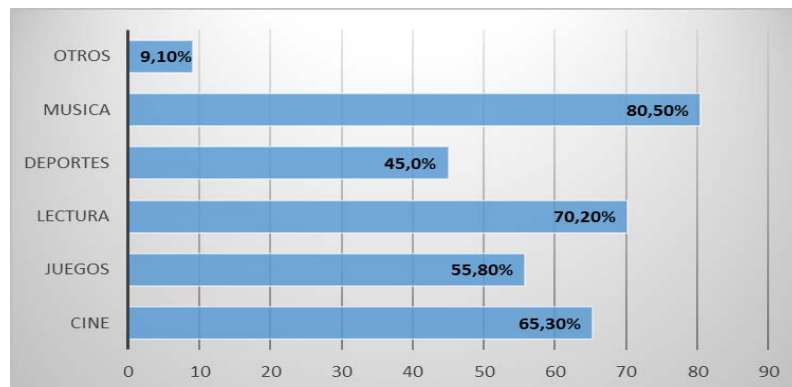


Figura 9 Aficiones más frecuentes entre los entrevistados.  
Fuente (elaboración propia)

Continuando con el análisis, en todas las franjas de edad **las mujeres leen más que los hombres**. Aunque este margen se va estrechando con la edad.

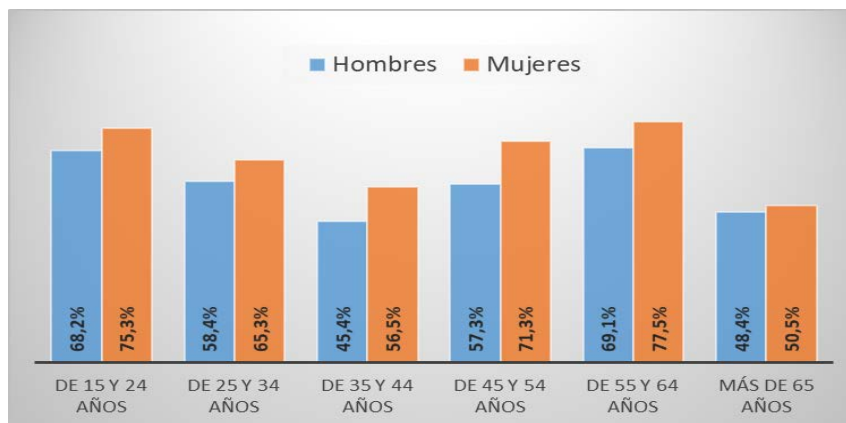
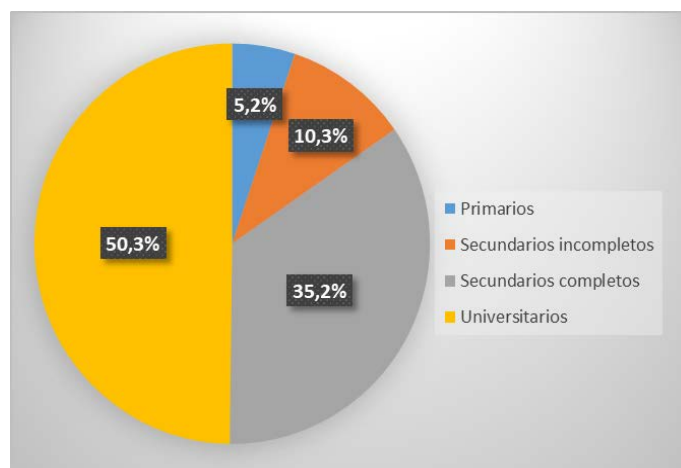


Figura 10 Aficiones más frecuentes entre los entrevistados.  
Fuente (elaboración propia) Los datos representan a los que leen más de un libro al año

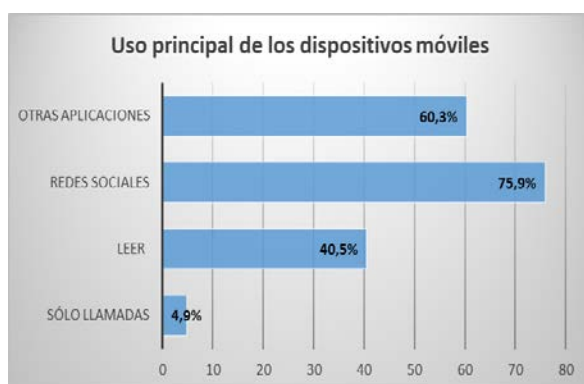
Las encuestas desvelan que **la lectura de libros crece notablemente a medida que aumenta el nivel de estudios finalizados**. Entre los lectores hay mayor proporción de universitarios y secundarios completos que de gente con estudios primarios o secundarios incompletos.



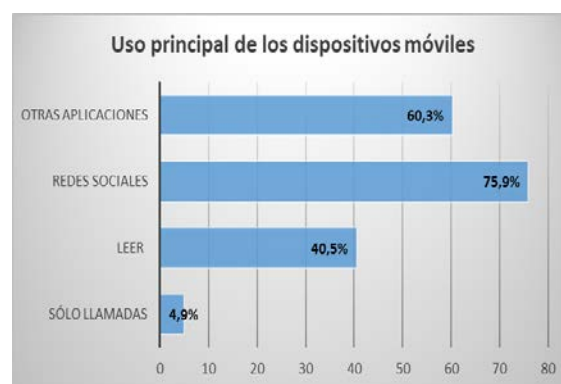
**Figura 11 Nivel de estudio entre los lectores**  
Fuente (elaboración propia)

**Las personas solteras que viven en la ciudad y en casa de sus padres son los que más leen (el 80%)**. El segundo grupo que más lee son los casados o viviendo en pareja sin hijos y por último muy igualados los solteros independientes o matrimonio con niños menores de 18 años.

A pesar de la irrupción de las nuevas tecnologías, **los libros en papel siguen siendo los favoritos (65%)** frente a libro electrónico, el móvil o la tableta. Aunque el 30% de los lectores habituales aseguraron haber leído al menos un libro en formato electrónico en el último año. Y es que, en cuanto al uso de tecnología, **el 95% de los lectores utiliza el móvil para algo más que no sean solo llamadas y el 40% de los lectores utiliza tanto tablet como móvil en su vida diaria**.



**Figura12 Dispositivos móviles utilizados**



**Figura 13 Uso principal dispositivos**

Fuente (elaboración propia)

Es esencial para que nuestra app se distribuya a un gran número de personas, que los lectores utilizan móvil y/o tablet. Según el gráfico se pone de manifiesto que en la mayoría de los casos se puede contar con ello.



La franja del día que más utilizan los lectores para leer es la noche (60%), donde más de 70% los que leen de noche lo hace en la cama.

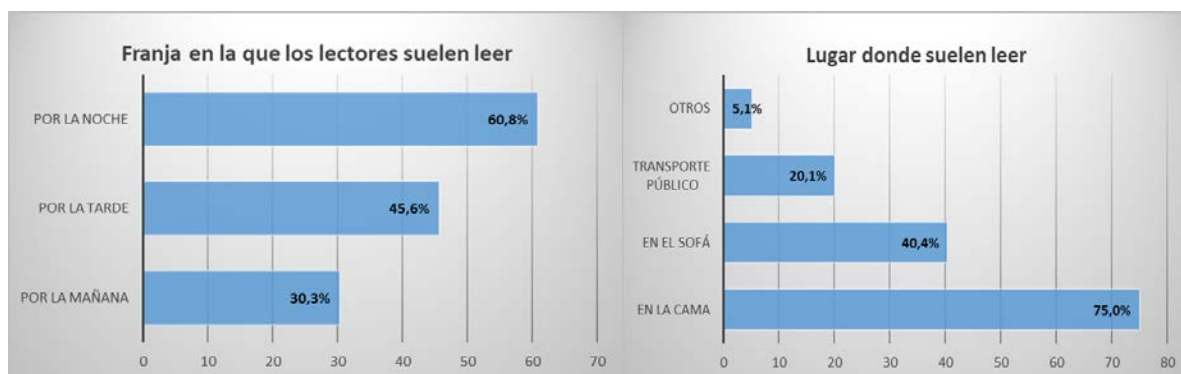


Figura 14 Franja en la que lectores leen

Figura 15 Lugar donde lectores leen

Fuente (elaboración propia)

Teniendo en cuenta este dato, sería muy interesante que la aplicación tuviese un modo oscuro, sino en ésta primera versión, en una futura actualización.

Para finalizar únicamente el 3% de los lectores utilizan una app para organizar sus lecturas, sin embargo, el 55% afirma que le gustaría usar alguna. El principal motivo por el que no utilizan una app para organizar sus lecturas es porque no conocían que hubiera (60%).



Figura 16 Porcentaje de lectores que utilizarían aplicación

Fuente (elaboración propia)

Para profundizar en sus motivaciones e intereses, se ha elegido continuar con las **entrevistas** como método de indagación cualitativo.

Se ha optado por una entrevista semiestructura con unas preguntas básicas, pero con un guion abierto, donde la persona entrevistada tenga margen para expresarse. El guion de la entrevista se puede encontrar en el anexo [\[11\]](#).

Por motivos de prevención frente al COVID, las entrevistas han sido realizadas por teléfono en lugar de en persona.

**Gran parte de los entrevistados declaran usar móvil y/o tablet en su vida diaria** y utilizan prácticamente las mismas aplicaciones, WhatsApp, Facebook, Twitter, Spotify...etc. Ninguno de los entrevistados usa una aplicación para organizar libros, pero comentan que les parece interesante.

Los lectores comentan que lo que les motiva de la lectura es que además de disfrutar leyendo obtienen beneficios tanto a nivel cultural como mentales. **Gran parte de los lectores no leen más por falta de tiempo, pero si tuvieran pequeños retos motivadores seguro que podrían sacar algún hueco y aumentar sus hábitos de lectura.**

Cuando se les pregunta por su libro favorito, todos lo han tenido claro y han hecho un breve resumen sobre él. Pero **la mayoría se lamenta por no recordar ninguna de esas frases o citas que les encantó mientras lo leían.**

Para encontrar el siguiente libro para leer, los entrevistados suele recurrir a internet para ver las últimas novedades y sus críticas, buscar por los libros más leídos por géneros, o pedir consejo a sus amigos o conocidos de un buen libro. **Varios han opinado que sería muy interesante una aplicación de que les recomendará un libro en función de sus intereses, de sus últimas lecturas etc.** Tal y como hace Netflix o HBO recomendando películas. (Este hecho ha motivado una nueva funcionalidad para la app, **descubrir libros**, donde se nos recomendarán libros según nuestros intereses, ver los libros más leídos y las últimas novedades).

**Las personas que tiene organizados los libros en casa son lo que con ver con mejores ojos ven una aplicación para tenerlos organizados virtualmente.** Pero no solo les gustaría tener organizados los que tienen en propiedad sino en general lo que van leyendo y tener planificadas sus lecturas. Los entrevistados afirman que la pereza es el principal motivo de no tener los libros organizados en casa, por lo que, para usar una app, debería ser algo realmente sencillo de hacer, incluso que puedas disfrutar haciéndolo, han comentado.

Personas por encima de 45 años ha comentado que les daría un poco de pereza tener que ponerse a recordar todos los libros que han leído y meterlos en la aplicación. Pero a partir de ahora les parece muy interesante comenzar. En cuanto a lectores más jóvenes, les atrae mucho la idea de ver dentro de unos años cual ha sido su evolución y la cantidad de libros que han leído con sus anotaciones correspondientes.

### 2.1.2 Diseño conceptual

Una vez finalizada la fase de investigación, llega el momento de procesar y sintetizar toda la información obtenida, de manera que todas las decisiones de diseño que se tomen, tengan en cuenta las necesidades detectadas durante esta etapa. Para ello, se ha optado por la utilización de la técnica de perfil de usuario, ficha de persona y escenarios de uso, finalizando con un mapa de navegación de la app.

- **Perfil de usuario**

A partir de los resultados de la encuesta podemos crear un perfil de usuario tipo objetivo, que nos permita agrupar los usuarios de nuestra aplicación según sus características comunes:

**Hombre o mujer entre 18 y 65 años, que le gusta la lectura. Tiene estudios de secundaria o universitarios e inquietudes culturales. Utiliza en su vida diaria móvil y/o tableta. Es una persona organizada y planificadora. Le gusta estar informados sobre últimas novedades literarias y disfrutan con los retos creativos.**

- **Ficha de persona y escenario de uso**

Resulta muy frío y abstracto hablar de porcentaje, estadística y grupos de población, por lo que pasaremos a desarrollar las fichas de persona. Éstas no ayudarán a dotar de realismo a la persona, haciéndola parecer un usuario real, al que le asignaremos las características detectadas durante la fase de investigación y análisis de usuarios y basándonos en el perfil de usuario.

De la misma manera, una vez se ha conseguido dirigir el proyecto hacia las características de los usuarios potenciales de nuestra aplicación, crearemos un escenario, que nos ayude a establecer un contexto donde interactúen el usuario y nuestra app en una situación concreta.

- **Árbol de navegación de la aplicación**

Para finalizar con el diseño conceptual, sería interesante tener un primer diagrama visual de cómo va a estar estructurada nuestra aplicación, para que visualmente y de manera esquemática, podamos tener una idea de lo que se puede hacer y donde se puede ir. Para ello tras los escenarios se representa el árbol de navegación de la aplicación.



*"Los libros me a ha acompañado toda mi vida"*

**Nombre :** Felisa Sierra

**Edad:** 49 años

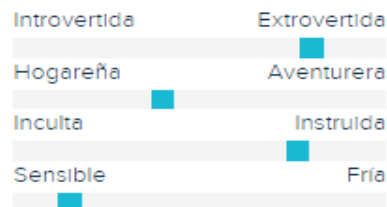
**Trabajo:** Profesora magisterio

**Familia:** Casada y con 2 hijos.

**Población:** Barcelona, España

**Carácter:** Apasionado

## Personalidad



## Metas

- Seguir poder disfrutando del tiempo libre y sus nietos.
- Publicar un libro de cuentos infantiles.
- Comprar una casa más grande donde pueda tener su propia biblioteca.

## Frustraciones

- No poder realizar grandes viajes debido al coronavirus.
- No haber podido inculcar a sus hijos el hábito por la lectura
- Cada vez tienes menos memoria para acordarse de todo lo que ha leído.

## Biografía

Felisa está felizmente casada y tiene 2 hijos ya emancipados, incluso tiene un nieto precioso. Desde que los pájaros volaron del nido Felisa tiene mucho tiempo libre que invierte social y culturalmente. Le encanta viajar, la lectura, la música, y se apunta a cualquier actividad cultural que se realice en su zona.

Felisa no sólo disfruta leyendo, sino que también le gusta escribir. Ha escrito algunos cuentos infantiles, que por no considerarlos suficientemente buenos no se ha atrevido a publicar. Y es que los pequeños son su vocación, de ahí que estudiara magisterio infantil. Forma parte de un club de lectura en la biblioteca del barrio y le apasiona debatir las lecturas con sus compañeros y aconsejar y dejarse aconsejar por el próximo libro a leer.

Felisa lleva una vida relajada y organizada. Ha hecho amigos en muchos sitios debido a su carácter extrovertido y tiene la habilidad de saber conservarlos. Para ello utiliza WhatsApp en su móvil para enviar mensajes y Facebook en su tableta para ponerse al día mediante fotos de sus vidas. Para terminar el día nada hay que le da mas placer que tumbarse en la cama con un buen libro y adentrarse en el fantástico mundo de la lectura.

## Comportamiento habitual

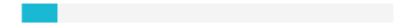
Social



Cultural



Deporte



Viajar

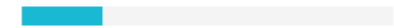


## Tecnología

Redes sociales



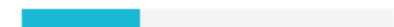
Ordenadores



Móviles



Tablets



## Aplicaciones frecuentes



## Escenario

Felisa llega a casa del trabajo, ha sido un día duro guerreando en el cole con los pequeños, pero se siente bien consigo misma, le gusta su trabajo. Además está muy contenta, es martes y hay sesión de club de lectura. Come con su marido y lo invita a unirse al club, pero éste prefiere quedarse viendo "The Dark", serie a la que está enganchado. Una vez en el club van comentando el libro que acaban de terminar, "El alquimista" de Paulo Coelho. Felisa abre la aplicación Livooks, donde guarda las citas y frases que la han cautivado, como por ejemplo "sólo una cosa convierte en imposible un sueño: el miedo a fracasar" y las expone con sus compañeros para ir opinando sobre ellas. Al terminar la sesión, el coordinador del club reparte el próximo libro a leer y hace una recomendación sobre las últimas novedades. Cuando llega a casa Felisa se da un baño relajante mientras su marido hace la cena. Se va a la cama temprano, quiere comenzar su nuevo libro. Una vez en la cama abre Livooks, señala "El alquimista" como y lo etiqueta como finalizado. Mediante el buscador encuentra su nuevo libro a leer por su título "La sombra del viento" y lo añade a leyendo. Para dejarlo todo organizado busca el libro recomendado por el coordinador y lo deja como pendiente. Se siente a gusto con la facilidad de organizar sus lecturas. Abre su libro y se sumerge en la lectura, siempre con Livooks a mano para almacenar esas citas memorables que quiere que siempre la acompañen.



"Tecnología, juegos y libros que más se le puede pedir a la vida"

**Nombre :** Pablo Velázquez  
**Edad:** 27 años  
**Trabajo:** Ingeniero Informático  
**Familia:** Vive con sus padres.  
**Población:** Sevilla, España  
**Carácter:** Tecnológico

## Personalidad



## Metas

- Ascender en su trabajo de programador junior a senior.
- Tener más tiempo libre para dedicar a sus aficiones.
- Conseguir ganar un campeonato de Magic a nivel nacional.

## Frustraciones

- La gran cantidad de publicidad en las aplicaciones.
- Que Georger JJ Martin no termine de escribir juego de tronos.
- No haber sido seleccionado para Top Gamers.

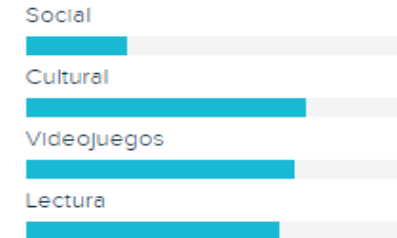
## Biografía

Pablo es hijo único, vive en Sevilla con sus padres en el barrio de Nervión. Situación en la que se encuentra muy a gusto y de momento no se plantea independizarse, quizás cuando encuentre pareja. Trabaja como programador en una empresa de Software. Se desplaza en transporte público al trabajo donde suele aprovechar para leer o jugar con el móvil, sus principales aficiones.

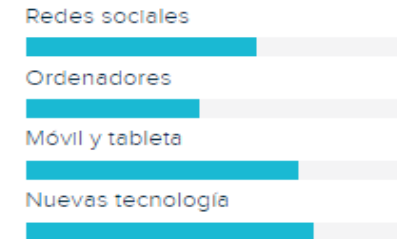
La vida social de Pablo no es muy ajetreada, no suele salir de fiesta y cuando queda con los amigos es para jugar a Magic. Tiene más de 1000 cartas organizadas y etiquetadas que cuestan una fortuna. Es un apasionado de los retos, su predilección son las *Escape Rooms*, tanto en juegos de mesa como en vivo.

A Pablo le encantan las nuevas tecnologías y siempre está dispuesto a probar cualquier novedad del mercado . En casa tiene un ordenador de sobremesa que sobre todo utiliza para jugar, una tableta en la que suele leer los libros que descarga de Internet y se mueve con un Smartphone último modelo lleno de aplicaciones de todo tipo.

## Comportamiento habitual



## Tecnología



## Aplicaciones



## Escenario de uso

Son las 11:15h de la mañana del Sábado y Pablo se acaba de despertar. Ha aprovechado para descansar, ya que la noche anterior se quedó jugando hasta tarde. Desayuna , recoge su cuarto y prepara sus cartas de Magic para la quedada de esta tarde con sus colegas. Como de costumbre, han quedado en *La Posada*, un sitio tranquilo donde tomar algo y poder jugar tranquilamente. Allí se encuentra con Hugo que lleva un libro, el comienzo de la saga "*The Black Company*", Pablo se siente atraído por el título, le pide el libro, lee la sinopsis y le parece interesante. Abre su aplicación *Livooks*, selecciona el icono de escanear el código de barras, se abre la cámara del móvil y apunta al código, el libro se agrega en unos segundos a la aplicación y lo pone como pendiente de leer. Termina y comienza a barajar su mazo, la partida está a punto de comenzar. Esa noche Pablo se va a la cama temprano, se descarga el libro que le recomendó su amigo en la tableta y comienza su lectura, al poco de comenzar hay una frase que llama su atención, la selecciona y la copia, abre *Livooks* selecciona el libro que está leyendo, pega la frase y la guarda. Aparece una nuevo diálogo en la pantalla, "Nueva insignia desbloqueada, El devorador de citas, 100 citas almacenadas". Pablo sonríe orgulloso, deja la app en segundo plano y continúa con su lectura.

### 2.1.3 Diseño del prototipo

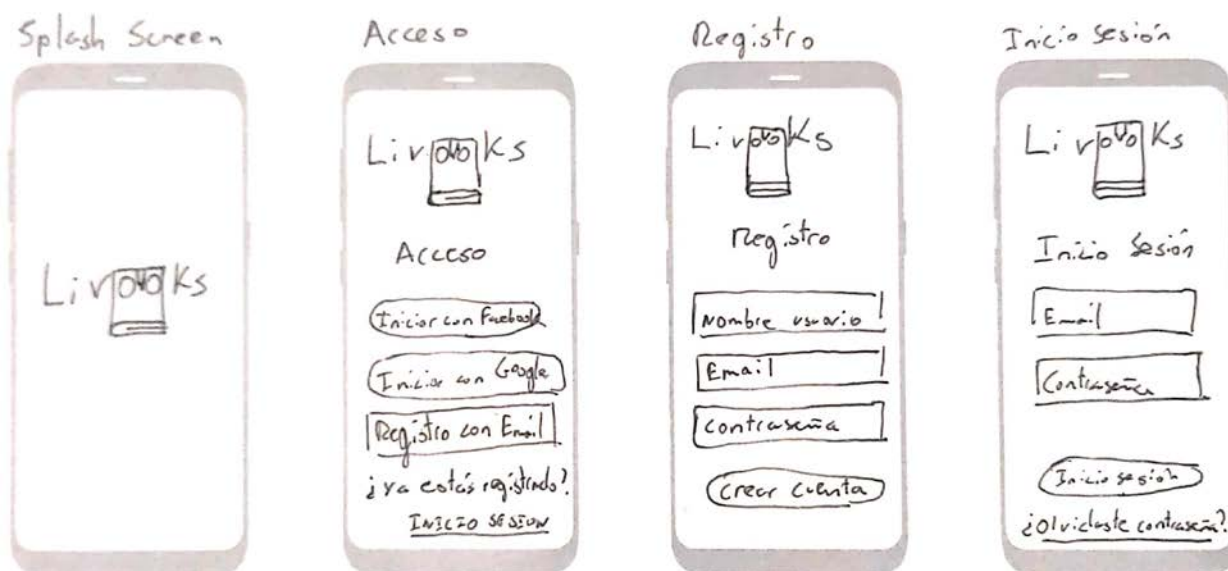
Llegados a este punto ya se tiene una idea más detallada de las funcionalidades de la aplicación, los objetivos que tendrá que permitir alcanzar a los usuarios, qué tipos de usuarios la utilizarán, en qué contexto, etc. El siguiente paso consiste en conceptualizar cómo lo resolverá.

Sin entrar todavía en la fase de producción, con la técnica de prototipado se puede empezar a dar forma al producto, resolviendo los diferentes problemas de diseño que vayan apareciendo.

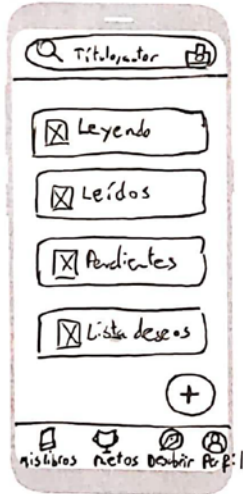
- **Prototipado de baja fidelidad**

Se ha comenzado con un prototipado de baja fidelidad en papel (sketching) que permitirá de una manera rápida y eficaz esbozar las ideas que salen de la cabeza y tomen vida propia, empezando a evolucionar. En esta fase lo más importante es visualizar qué es lo que realmente se quiere conseguir y cómo se quiere hacer.

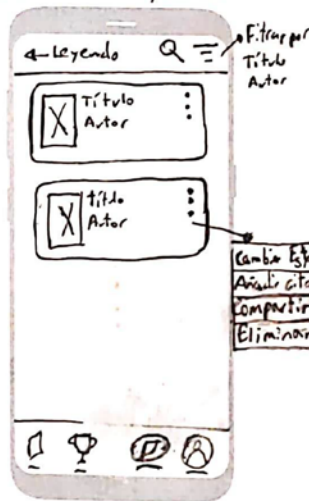
Se han identificado los conceptos, flujos y elementos de la interfaz sin utilizar ningún software para tener mayor libertad. Solamente se han utilizado unas plantillas como guía para dibujar sobre ellas para hacernos una idea de los tamaños y proporciones de los elementos que se colocan en pantalla



Pantalla Inicio



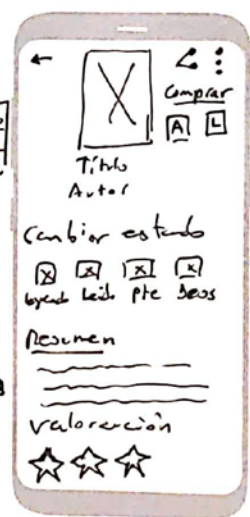
Pantalla Leyendo



Detalles Libro leyendo



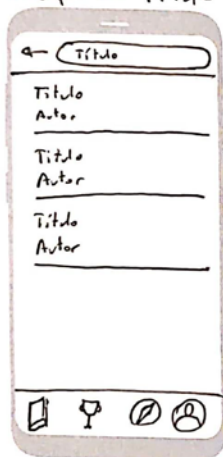
Detalles Libro Deseos



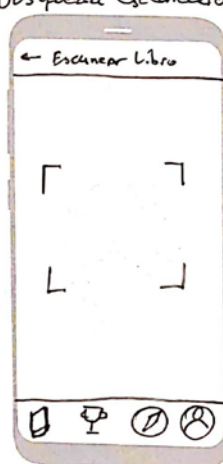
Búsqueda en Pantalla Inicio



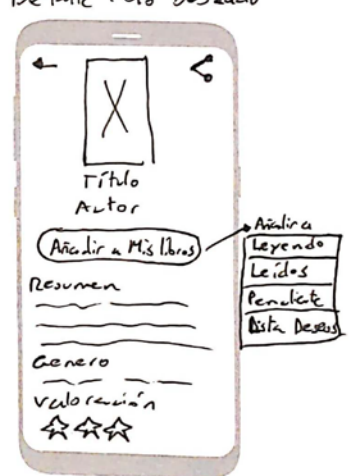
Búsqueda Título



Búsqueda escaneado



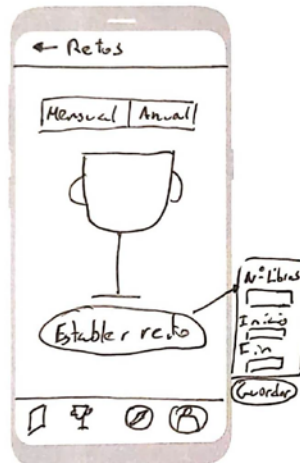
Detalle libro buscado



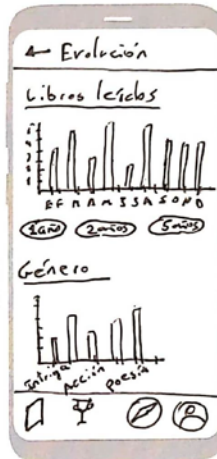
Pantalla progreso



Pantalla Retos



Pantalla Evolución



Pantalla Logros



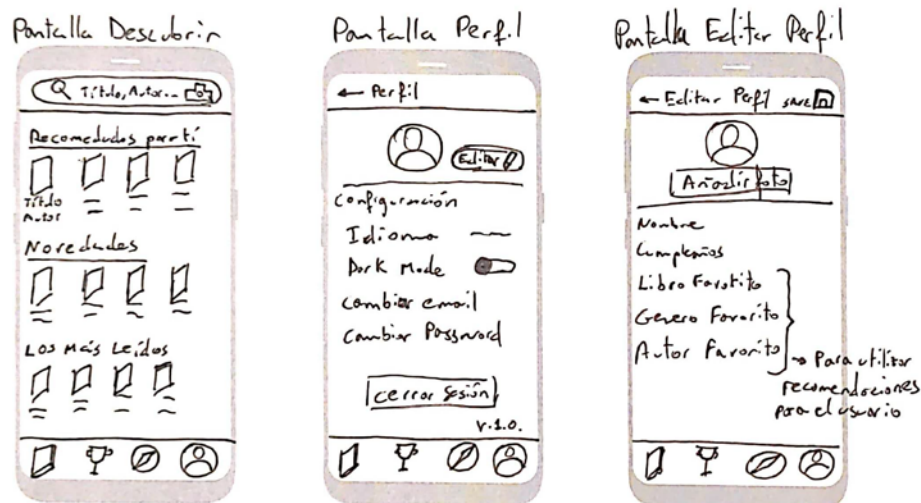


Figura 17 Prototipo de baja fidelidad (Sketching)  
Fuente (elaboración propia)

- **Prototipado de alta fidelidad**

En este momento se tienen las ideas más claras y toca refinar el diseño. A medida que avanzamos con el diseño, también avanzará el nivel de detalle de los prototipos, hasta el punto de que querer dotarlos de interacción y realismo. Con el prototipo de alta fidelidad se ha conseguido testear nuestras ideas más allá del papel.

El prototipo se ha realizado utilizando la herramienta **Figma**, que permite fácilmente realizar un diseño totalmente interactivo. Consiguiendo una calidad tal, que la versión prototipo y la implementada finalmente pueda llegar a ser inapreciable.

Como veremos sobre las pantallas, a la hora de diseñar el prototipo se han teniendo en cuenta los criterios de usabilidad y el seguimiento de la filosofía Material Design de Android, ya que es la plataforma para la que vamos a implementar la app. Asegurando así una buena consistencia y estándares para una mejor experiencia de usuario y una curva de aprendizaje rápida.

Llegado el momento, este prototipo servirá para hacer pruebas con usuarios reales, detectando problemas de concepto, diseño o usabilidad antes de entrar en la fase de implementación.

A continuación, se presentan algunas de las pantallas de la aplicación:

Cada vez que se inicie la aplicación aparecerá una **Splash Screen** con el logo que representa la app. Un logo sencillo, pero eficaz, ya que de un vistazo se pueda intuir que los libros van a ser los protagonistas. La paleta de colores se ha seleccionado como imagen de marca y familiaridad para todas las pantallas. La siguiente pantalla será la de **acceso**, donde se puede iniciar sesión con Facebook o Google, así como el **registro** con email muy breve si no se está registrado, y un **inicio de sesión** típico si ya lo estamos, en el que podremos recuperar nuestra contraseña olvidada.



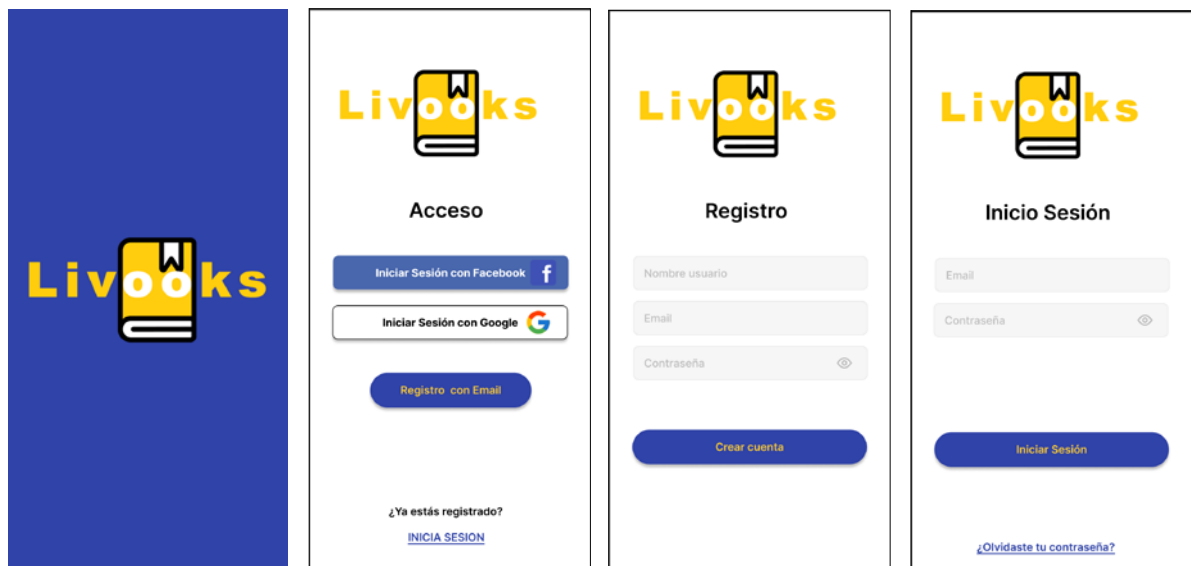


Figura 18 Prototipo alta fidelidad Pantalla de splash y autenticación

Tras el registro o inicio exitoso en la app, aparecerá la pantalla principal, “el corazón de la aplicación”. En ella se dispone de una barra de navegación en la parte baja, situando de izquierda a derecha las secciones más importantes coincidiendo con la zona más cómoda de interacción para el usuario. Esta barra estará presente en todo momento en la app. También se puede ver una toolbar en la parte de arriba donde se pueden buscar libros en la base de datos, bien manualmente, o bien mediante el escaneo del código de barras con la cámara del teléfono. Se ha incluido en esta pantalla un botón fab icon para añadir libros, al cual están habituados los usuarios de android, en el que se puede elegir igualmente buscar libro manualmente o escaneado.

El contenido de esta pantalla principal “**Mis libros**”, esta formado por la clasificación de nuestras “estanterías”. Los libros que estamos leyendo, que hemos leído, que tenemos pendientes de leer, y lo que nos gustaría comprar o que nos regalen en nuestra lista de deseos.



Figura 19 Prototipo alta fidelidad pantalla principal Mis Libros

La siguiente pantalla muestran el listado de los libros de una de las categorías (será igual para todas) en formato CardView para dar un aspecto más atractivo, donde podemos hacer filtrado. La pantalla que la acompaña es la de detalles de uno de los libros del listado, el cual, se puede gestionar. Cambiándolo de categoría, añadiendo cita, fechas. Se ha añadido un menú de puntos flotantes para la eliminación del libro y accesos directos a acciones como compartir.

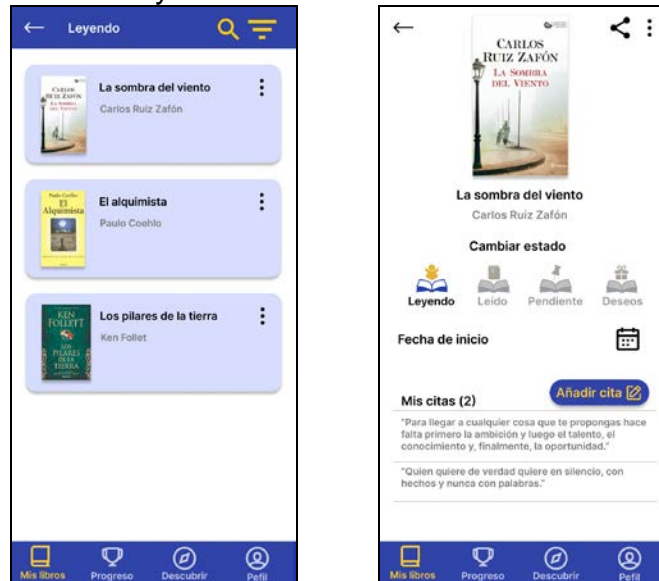


Figura 20 Prototipo alta fidelidad listado categoría de libros y detalles

Las dos primeras pantallas siguientes, representan la búsqueda manual y la de escaneado código de barras de las que se habló en la pantalla principal. En la primera de ellas aparecerá un listado coincidente con la búsqueda y en la segunda se colocará el código de barras en el recuadro de la pantalla, tal como se le indica al usuario como guía. La tercera pantalla es la de detalles del libro buscado, la cual aparecerá al seleccionar un libro de la lista, o bien se escanee su código de barras. En ella se puede ver los detalles del libro y lo más importante el botón para añadirlo a nuestra biblioteca.

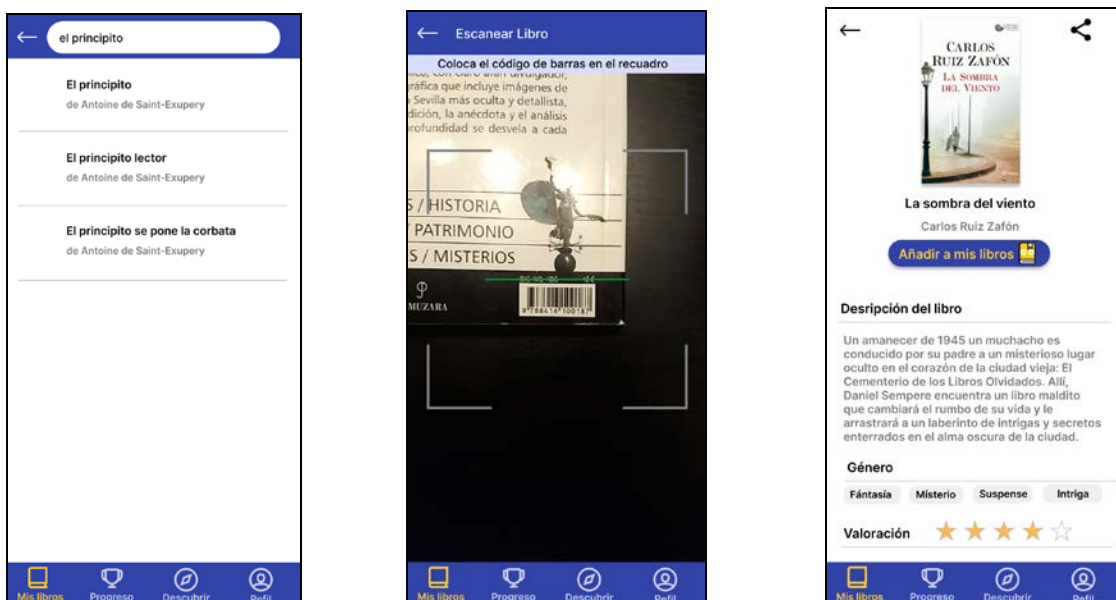


Figura 21 Prototipo alta fidelidad búsqueda título o escaneado y detalle

En la siguiente pantalla se encuentra la sección descubrir, donde se podrá, además de buscar libros, elegir entre las recomendaciones que muestra la app a través de un catálogo seleccionado para ayudar al usuario a elegir su próxima lectura. A la derecha aparece la sección de perfil, donde se pueden cambiar algunas opciones de configuración, así como su edición.

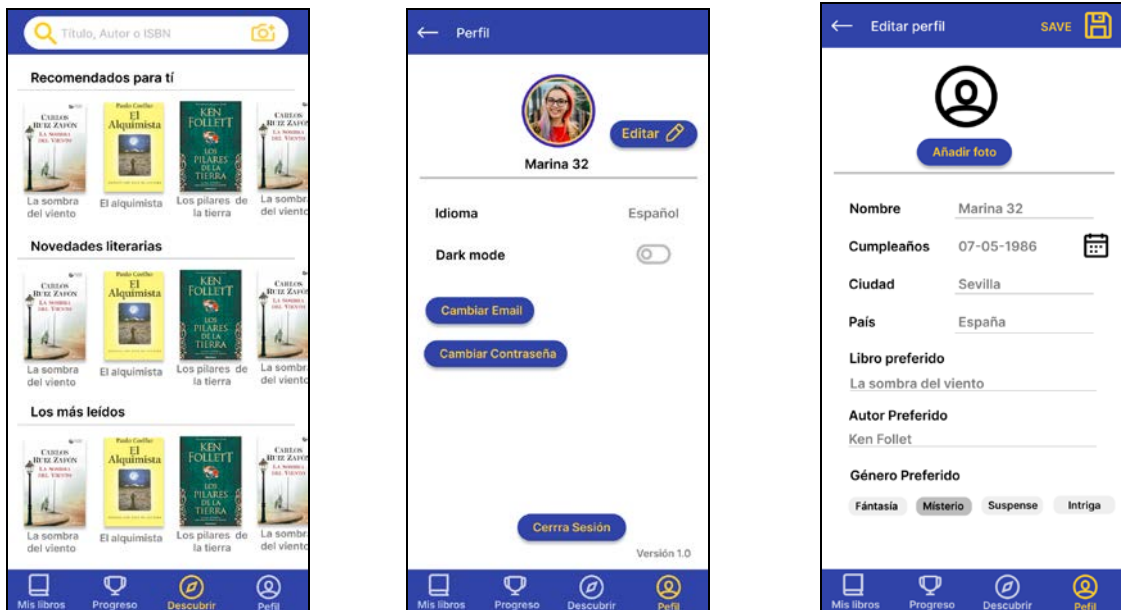


Figura 22 Prototipo alta fidelidad pantalla descubrir libros y perfil

Las siguientes pantallas representan la sección de progreso, donde se tendrá una pantalla principal con el resumen de nuestro progreso, así como diferentes opciones. La primera de retos, pantalla en la que se podrán establecer retos mensuales o anuales de lectura. La segunda de evolución donde de manera gráfica se puede ver el progreso del usuario a lo largo de los años y la tercera de logros, donde se desbloquearán insignias en función de los retos proporcionados por la app que se vayan superando.

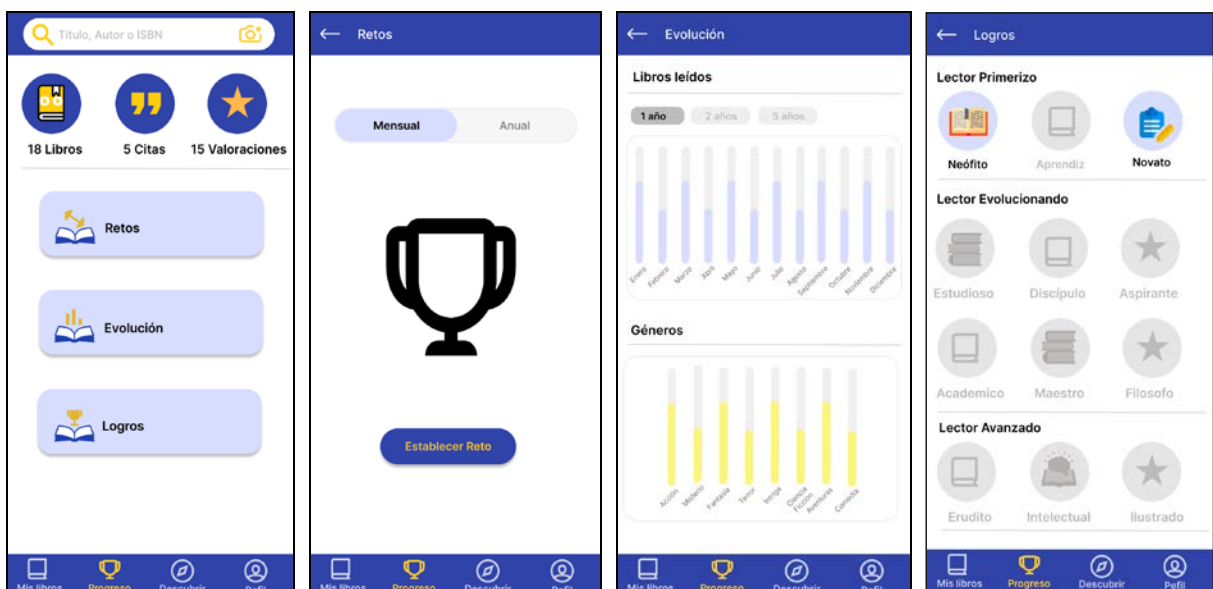


Figura 23 Prototipo alta fidelidad pantalla retos, logros y evolución

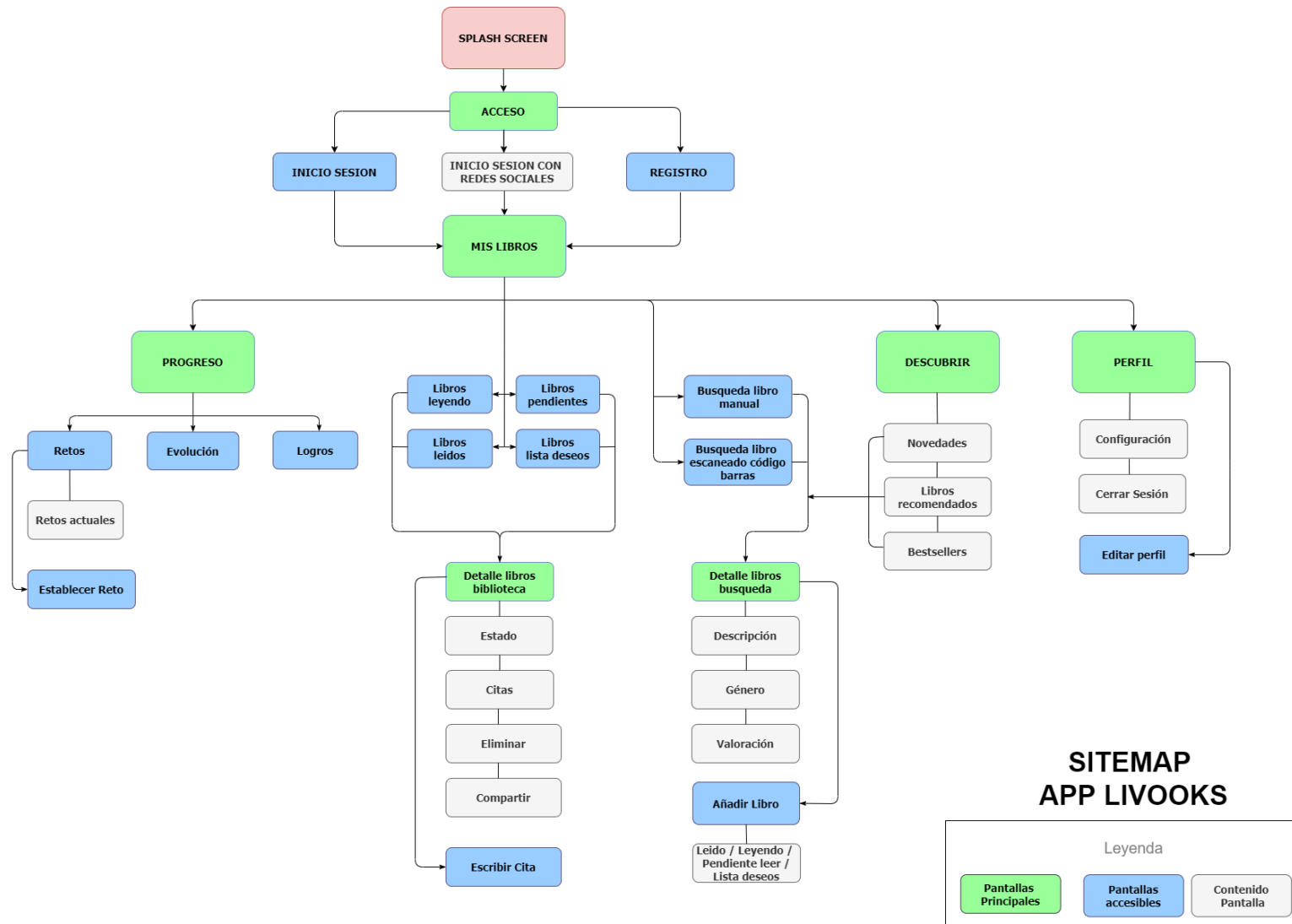


Figura 24 Árbol de navegación de la aplicación

## 2.1.4 Evaluación

Como el título adelanta, se trata de técnicas de evaluación para detectar rápidamente errores o puntos de mejora en el diseño, ya que resulta “virtualmente imposible” diseñar a la primera una interfaz de usuario que no tenga problemas de usabilidad

Detectadas estas mejoras, es donde entra en juego el desarrollo iterativo del DCU, de tal manera que ambos se van autoalimentando hasta obtener el mejor de los productos posibles.

**Ha llegado la hora de comprobar que la solución propuesta, se adapta al público objetivo, para ello se ha utilizado el análisis heurístico y el test de usuarios.**

- **Análisis heurístico**

El análisis heurístico, ha permitido detectar fácilmente los errores más graves, sometiendo al diseño a revisión de acuerdo a un conjunto de reglas y principios de usabilidad establecidos. En este caso se han utilizado las diez reglas heurísticas de Jakob Nielsen [\[10\]](#) para el diseño de interfaces, detectando entre ellas:

**Visibilidad del estado del sistema:** el sistema debe informar en todo momento al usuario de lo que está ocurriendo.

- No se había tenido en cuenta mostrar un icono de progreso cuando realicemos una búsqueda de un libro, que indicará al usuario que se encuentra realizando la acción requerida.



**Figura 25 Búsqueda libro con icono de progreso de búsqueda**  
Fuente (elaboración propia)

**Control y libertad del usuario:** hay ocasiones en que los usuarios pueden elegir funciones por error y necesitarán una “salida de emergencia” claramente marcada. El sistema debe permitir las opciones de deshacer.

- Cuando se elimine un libro se implementará un Snackbar con opción deshacer, por si el usuario lo ha eliminado por error.

**Flexibilidad y eficiencia de uso:** el sistema debe ser claro para los usuarios novatos, sin dificultar el camino a los avanzados.

- Se han implementado diferentes opciones para realizar la misma operación permitiendo atajos al usuario experto.

**Estética y diseño minimalista:** en un móvil cada pedazo de información extra compite con lo importante y resta espacio en pantalla.

- Se ha eliminado información en pantalla innecesaria o irrelevante dejando pantallas más amplias y limpias.

- **Tests con usuarios**

Tras haber detectado con el método anterior los errores más evidentes. Con los tests de usuarios, se quiere obtener información de carácter cualitativo sobre nuestra interfaz: si está bien diseñada, si se entiende bien a la primera, si es fácil conseguir los objetivos que se proponen.

Sin tener ningún conocimiento anterior sobre la aplicación, se plantea a usuario la realización de diferentes tareas sobre el sistema mientras lo observamos. El usuario no tendrá posibilidad de hacer preguntas, ya que se trate de averiguar lo que haría el usuario estando a solas. Las tareas encomendadas e identificadas como críticas son las siguientes:

<b>TAREAS A COMPLETAR</b>
Registrarse en la aplicación
Añadir un libro mediante búsqueda manual a nuestra biblioteca.
Añadir un libro mediante escaneado de su código de barras
Añadir un libro desde la pantalla descubrir libros.
Ver los libros que estamos leyendo/leídos/pendiente o en lista deseo.
Eliminar un libro de cualquiera de las categorías de nuestra biblioteca.
Añadir una cita en uno de nuestros libros almacenados.
Cambiar de categoría uno de nuestros libros almacenados.

Añadir un reto de lectura
Cerrar la sesión

**Tabla 4 Tareas a completar por el usuario sometido al Test**

Han intervenido en los test de usabilidad:

- **Participantes:** se han escogido a dos usuarios potenciales para la realización del test de 45 min aproximada de duración.
- **Facilitador:** en este caso, el desarrollador de la aplicación, animando a los participantes a transmitir lo que está experimentando, pero sin prestar ayuda como se comentó anteriormente.
- **Observador:** en este caso, el desarrollador de la aplicación, analizando todo lo que ocurre y tomando notas para la posterior solución de los problemas que se vayan detectando.

A partir de esta evaluación se ha detectado las siguientes mejoras a aplicar en nuestra app.

- El usuario ha podido añadir una cita sin problemas, pero ha tenido varios errores al escribirla por lo que quería modificarla o eliminarla y crear otra. Ninguna de las posibilidades se había contemplado de antemano, con lo que se ha visto reflejado la desilusión en la cara del usuario. **Corrección necesaria.**
- Al establecer reto mensual o anual el usuario no tenía posibilidad de elegir cuando comenzar. **Corrección necesaria.**

Al ser un proceso iterativo durante la fase de implementación de la aplicación se seguirá sometiendo a evaluación a medida que vayamos avanzando en su desarrollo para poder seguir detectando errores o posibles mejoras de manera temprana.

## 2.2 Diseño Técnico

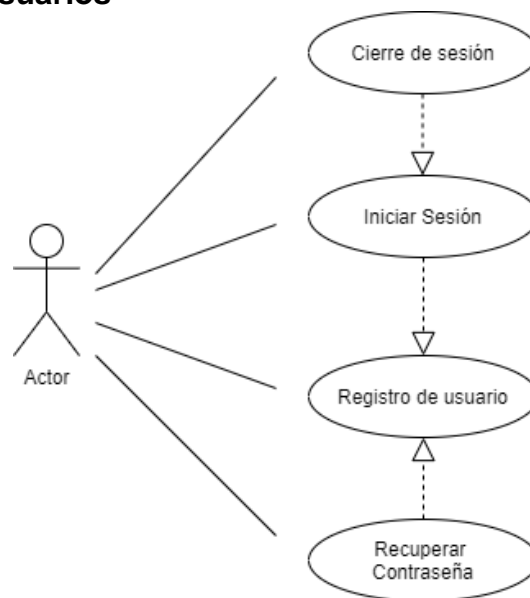
Una vez finalizada la fase de DCU y habiendo conseguido un prototipo de alta fidelidad de nuestra aplicación, es el momento de comenzar su diseño técnico.

### 2.2.1 Definición de los casos de uso

Se pasará a definir formalmente los casos de uso, que servirán, para establecer la interacción entre el sistema y el usuario, alcanzando los objetivos de las funcionalidades de la aplicación derivadas de los escenarios de uso y el prototipo definido durante el DCU.

El diagrama UML correspondiente servirá para identificar de forma gráfica actores y flujo. En este caso se contará con un único actor, que será el usuario que desea tener sus lecturas organizadas. El sistema no se verá influenciado por ninguna otra entidad que no sea el usuario genérico.

- **Gestión de usuarios**



**Figura 26 Diagrama UML de casos de uso “Gestión de usuarios”**  
Fuente (elaboración propia)

Se exponen los diferentes casos de uso identificados, indicando los actores que intervienen, las precondiciones, flujo y postcondiciones.

<b>CU01: Registro de usuario</b>	
<b>Descripción</b>	Darse de alta en la aplicación
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>• El dispositivo debe tener conexión a Internet</li> <li>• Que el email no esté ya registrado</li> <li>• Que la contraseña cumpla requisitos de seguridad</li> </ul>
<b>Flujo</b>	Desde la pantalla de <b>Acceso</b> , el usuario selecciona registro con email. Accede a la pantalla de registro, introduce email, contraseña,



	confirmación de contraseña y pulsa en crear cuenta.
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>• Si el email no está registrado se informa al usuario de que la cuenta ha sido creada correctamente y llevará al usuario al menú principal <b>Mis libros</b>.</li> <li>• Si el correo existe, se notifica al usuario que la cuenta ya existe.</li> </ul>

**Tabla 5. Caso 1 - Registro de usuario**  
(Fuente: Elaboración propia)

<b>CU02: Inicio de sesión</b>	
<b>Descripción</b>	Inicio de sesión en la aplicación mediante credenciales
<b>Actores</b>	Usuario
<b>Precondiciones</b>	Estar registrado en la aplicación (CU01 completado) o bien contar con una cuenta de Facebook o Google
<b>Flujo</b>	Desde la pantalla de <b>Inicio de Sesión</b> , el usuario selecciona iniciar sesión con Facebook, Google o si ya está registrado iniciar sesión con credenciales. Introduce email y contraseña, y pulsa en iniciar sesión.
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>• Si las credenciales son correctas, el usuario queda autenticado y llevará al usuario al menú principal <b>Mis libros</b>.</li> <li>• Si el email o contraseña no son correctos muestra al usuario un mensaje de error.</li> </ul>

**Tabla 6. Caso 2 – Inicio de sesión**

<b>CU03: Recuperar contraseña</b>	
<b>Descripción</b>	Recuperar contraseña olvidada para iniciar sesión
<b>Actores</b>	Usuario
<b>Precondiciones</b>	Estar registrado en la aplicación (CU01 completado)
<b>Flujo</b>	Desde la pantalla de <b>Inicio de Sesión</b> , el usuario selecciona ¿Olvidaste tu contraseña?, y aparecerá un cuadro de restablecer contraseña indicando al usuario que se ha enviado un correo de recordatorio contraseña cuando se pulse sobre él .
<b>Postcondiciones</b>	Si el usuario ha escrito un correo registrado, será enviado a su cuenta un correo de restablecimiento de contraseña.

**Tabla 7. Caso 3 – Recuperar contraseña**

<b>CU04: Cierre de sesión</b>	
<b>Descripción</b>	Cerrar sesión de la aplicación
<b>Actores</b>	Usuario
<b>Precondiciones</b>	En algún momento el usuario inició sesión (CU02 completado)
<b>Flujo</b>	Desde la pantalla principal, seleccionamos el icono de <b>Perfil</b> en la barra de menú. En la pantalla perfil seleccionamos <b>cerrar sesión</b> .
<b>Postcondiciones</b>	Se nos mostrará cuadro de diálogo para confirmar el cierre de sesión, si confirmamos, nos llevará a la pantalla de acceso, sino, nos mantendremos en la pantalla actual.

**Tabla 8. Caso 4 – Cierre de sesión**

- **Busqueda libros y añadir a la biblioteca “Mis libros”**

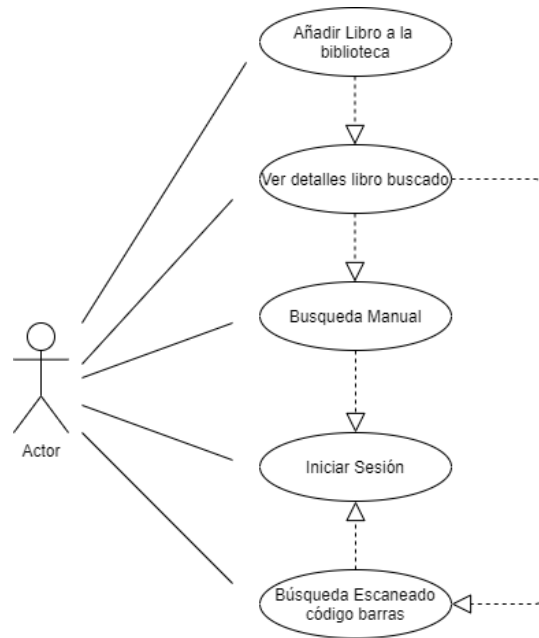


Figura 27 Diagrama UML de casos de uso “Busqueda libros”

<b>CU05: Buscar libro manualmente</b>	
<b>Descripción</b>	Busqueda de libro mediante introduciendo título, autor, ISBN...
<b>Actores</b>	Usuario
<b>Precondiciones</b>	En algún momento el usuario inició sesión (CU02 completado)
<b>Flujo</b>	<p><b>Opción 1</b></p> <ol style="list-style-type: none"> <li>1. Desde la pantalla de principal <i>Mis Libros</i>, el usuario selecciona la <b>barra de busqueda en la toolbar</b>.</li> <li>2. El usuario introduce una cadena de texto del libro a buscar.</li> <li>3. Aparecen listado con los libros coincidentes con los parámetros de busqueda</li> </ol> <p><b>Opción 2</b></p> <ol style="list-style-type: none"> <li>1. Desde la pantalla de principal <i>Mis Libros</i>, el usuario pulsa el <b>botón flotante de añadir libro</b>.</li> <li>2. El usuario selecciona <b>buscar libro por título</b> del cuadro que se abre.</li> <li>3. El usuario introduce una cadena de texto del libro a buscar.</li> <li>4. Aparecen listado con los libros coincidentes con los parámetros de busqueda</li> </ol>
<b>Postcondiciones</b>	Si el libro existe en la base de datos, el usuario visualiza los libros que coincide con los parámetros introducidos. Si no existe aparecerá mensaje comentando que no ha podido encontrar libro.

Tabla 9. Caso 5 – Buscar libro manualmente

<b>CU06: Buscar libro escaneando código barras</b>	
<b>Descripción</b>	Busqueda de libro mediante el escaneo de su código de barras con la cámara del dispositivo móvil.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>• En algún momento el usuario inició sesión (CU02 completado)</li> <li>• El usuario debe contar con una cámara para la captura de imágenes</li> <li>• El usuario debe conceder permisos para el acceso a la cámara</li> </ul>

<b>Flujo</b>	<p><b>Opción 1</b></p> <ol style="list-style-type: none"> <li>Desde la pantalla de principal <b>Mis Libros</b>, el usuario selecciona el icono de la cámara en la barra de búsqueda de la toolbar.</li> <li>Se abre la cámara del teléfono y el usuario coloca el código de barras del libro en el recuadro.</li> <li>Nos lleva a la pantalla de detalles del libro.</li> </ol> <p><b>Opción 2</b></p> <ol style="list-style-type: none"> <li>Desde la pantalla de principal <b>Mis Libros</b>, el usuario pulsa el <b>botón flotante de añadir libro</b>.</li> <li>El usuario selecciona buscar libro mediante <b>escaneado código barras</b> del cuadro de selección que nos sale.</li> <li>Se abre la cámara del teléfono y el usuario coloca el código de barras del libro en el recuadro.</li> <li>Nos lleva a la pantalla de detalles del libro.</li> </ol>
<b>Postcondiciones</b>	Si el libro existe en la base de datos (API), el usuario visualiza la página de detalle del libro. Si no existe aparecerá mensaje comentando que no ha podido encontrar libro. Si el libro ya está agregado a nuestra biblioteca aparecerá un mensaje comentándolo.

**Tabla 10. Caso 6 – Buscar libro escaneando código barras**

<b>CU07: Detalles libro buscado</b>	
<b>Descripción</b>	Ver los detalles del libro buscado
<b>Actores</b>	Usuario
<b>Precondiciones</b>	En algún momento el usuario inició sesión (CU02 completado) Buscó libró manualmente (CU05 completado) o buscó libro mediante escaneado (CU06 completado)
<b>Flujo</b>	<ol style="list-style-type: none"> <li>El usuario pulsa sobre uno de los libros resultado de la búsqueda.</li> <li>Nos lleva a la pantalla de detalles del libro.</li> </ol>
<b>Postcondiciones</b>	El usuario visualiza la página de detalle del libro.

**Tabla 11. Caso 7 – Detalles libro buscado**

<b>CU08: Añadir libro a <i>Mis Libros</i></b>	
<b>Descripción</b>	Añadir un libro a nuestra biblioteca.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>En algún momento el usuario inició sesión (CU02 completado)</li> <li>Buscó libró manualmente (CU05 y CU07 completado) o buscó libro mediante escaneado (CU06 completado)</li> </ul>
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla de detalles del libro, el usuario pulsa en <b>añadir a mis libros</b>.</li> <li>Se abre cuadro para seleccionar en que apartado de mis libros desea añadirlo: <b>leyendo, leídos, pendientes o lista de deseos</b>.</li> <li>Pulsamos en guardar y nos llevará a la pantalla del conjunto de libros de esa categoría.</li> </ol>
<b>Postcondiciones</b>	Si el libro ya existe en nuestra biblioteca, nos mostrará un mensaje. Si no existe, nos permitirá guardarlo en la categoría que queramos. El usuario visualiza la pantalla de la categoría que haya seleccionado.

**Tabla 12. Caso 8 – Añadir libro a *Mis libros***

- Gestionar *Mis libros*

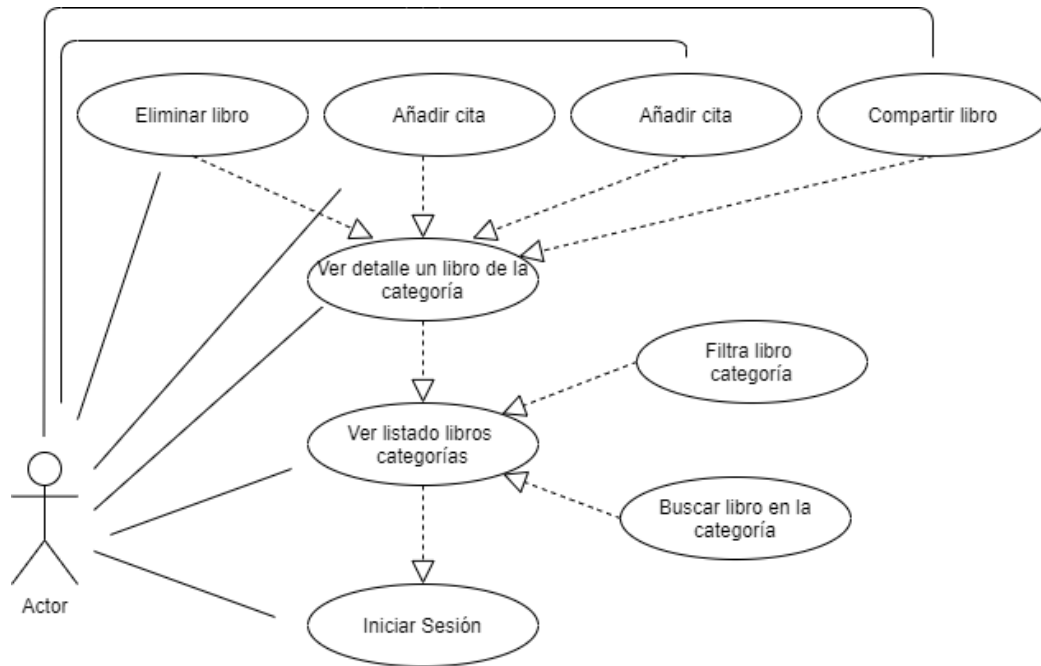


Figura 28 Diagrama UML de casos de uso “Gestionar *Mis libros*”

CU09: Ver los libros de mi biblioteca	
<b>Descripción</b>	Visualizar listado de “Mis libros” en las diferentes categorías.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>• En algún momento el usuario inició sesión (CU02 completado)</li> <li>• Buscó libro manualmente (CU05 y CU07 completado) o buscó libro mediante escaneado (CU06 completado)</li> <li>• Añadió el libro a una de las categorías. (CU08 completado)</li> </ul>
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla de inicio <i>Mis libros</i>, el usuario pulsa en una de las categorías: <b>leyendo, leídos, pendientes o lista de deseos</b>.</li> <li>2. Se abre pantalla correspondiente a la categoría seleccionada mostrando los libros que tenemos guardados en esa categoría.</li> </ol>
<b>Postcondiciones</b>	<p>Si no hay libros en esa categoría, nos mostrará un mensaje.</p> <p>Si hay libros, visualizaremos en la pantalla un listado de los libros que tengamos añadidos a esa categoría.</p>

Tabla 13. Caso 9 – Ver los libros de mi biblioteca

CU10: Buscar un libro en una categoría	
<b>Descripción</b>	Busqueda libro concreto en una de las categorías.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>• El usuario accede a una de las categorías de <i>Mis libros</i> (CU09 completado)</li> </ul>
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla de listado de libros de una categoría pulsamos en el <b>icono de lupa de la toolbar</b>.</li> <li>2. El usuario introduce una cadena de texto del libro a buscar.</li> <li>3. Aparecen los libros coincidentes con los parámetros de búsqueda</li> </ol>
<b>Postcondiciones</b>	<p>Si no hay libros que correspondan con la búsqueda, nos mostrará un mensaje. Si hay libros, los visualizaremos en la pantalla.</p>

Tabla 14. Caso 10 – Buscar un libro en una categoría

<b>CU11: Filtrar libros en una categoría</b>	
<b>Descripción</b>	Filtrar el listado de libros de una de las categorías.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>El usuario accede a una de las categorías de <b>Mis libros</b> (CU09 completado)</li> </ul>
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla de listado de libros de una categoría pulsamos en el <b>icono de filtrado en la toolbar</b>.</li> <li>Se abre cuadro con opciones de filtrado y el usuario las elige: filtrado por título, autor, etc.</li> <li>Aparecen los libros coincidentes con los parámetros de filtrado.</li> </ol>
<b>Postcondiciones</b>	Visualizaremos en la pantalla los libros con el filtrado seleccionado.

Tabla 15. Caso 11 – Filtrar libros en una categoría

<b>CU12: Ver detalle de un libro de mi biblioteca</b>	
<b>Descripción</b>	Visualizar detalles de uno de mis libros de la biblioteca.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>El usuario accede a una de las categorías de <b>Mis libros</b> (CU09 completado)</li> </ul>
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla de listado de libros de una categoría pulsamos en uno de los libros.</li> <li>Se abre pantalla correspondiente a los detalles del libro.</li> </ol>
<b>Postcondiciones</b>	Visualizaremos la pantalla con los detalles del libro que hayamos seleccionado.

Tabla 16. Caso 12 – Ver detalle de un libro de mi biblioteca

<b>CU13: Cambiar de estado (categoría) un libro</b>	
<b>Descripción</b>	Cambiar de categoría o estado uno de los libros de la biblioteca.
<b>Actores</b>	Usuario
<b>Precondición 1</b>	<ul style="list-style-type: none"> <li>El usuario accede al detalle de uno de los libros de las diferentes categorías (CU12 completado)</li> </ul>
<b>Flujo</b>	<p><b>Opción 1</b> Desde la pantalla de detalles del libro, en la sección estado del libro, el usuario selecciona el icono del estado al que quiere actualizar el libro: <b>leyendo, leídos, pendientes o lista de deseos</b>.</p> <p><b>Opción 2</b></p> <ol style="list-style-type: none"> <li>Desde la pantalla de detalles del libro, pulsamos sobre el menú flotante de 3 puntos de la toolbar.</li> <li>Se abre menú desplegable y el usuario selecciona la opción cambiar estado.</li> <li>Seleccionamos el estado al que queremos actualizar: <b>leyendo, leídos, pendientes o lista de deseos</b>.</li> </ol>
<b>Precondición 2</b>	<ul style="list-style-type: none"> <li>El usuario accede a una de las categorías de <b>Mis libros</b> (CU09 completado)</li> </ul>
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla de listado de libros, pulsamos sobre el menú flotante de 3 puntos de libro que queremos cambiar estado.</li> <li>Se abre menú desplegable y el usuario selecciona la opción cambiar estado.</li> <li>Seleccionamos el estado al que queremos actualizar: <b>leyendo, leídos, pendientes o lista de deseos</b>.</li> </ol>
<b>Postcondiciones</b>	El estado del libro se actualiza y se coloca en la categoría correspondiente.

Tabla 17. Caso 13 – Cambiar de estado (categoría) un libro

<b>CU14: Eliminar un libro</b>	
<b>Descripción</b>	Eliminar un libro del listado de libros de una categoría
<b>Actores</b>	Usuario
<b>Precondición 1</b>	<ul style="list-style-type: none"> <li>El usuario accede al detalle de uno de los libros de las diferentes categorías (CU12 completado)</li> </ul>
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla de detalles del libro, pulsamos sobre el menú flotante de 3 puntos de la toolbar.</li> <li>Se abre menú desplegable y el usuario selecciona la opción eliminar libro.</li> <li>Se muestra un mensaje de confirmación al usuario.</li> <li>El usuario confirma la eliminación pulsando en aceptar.</li> <li>El libro es eliminado del perfil del usuario.</li> <li>Se muestra la lista de la categoría del libro eliminado actualizada.</li> </ol>
<b>Precondición 2</b>	<ul style="list-style-type: none"> <li>El usuario accede a una de las categorías de <i>Mis libros</i> (CU09 completado)</li> </ul>
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla de listado de libros en la categoría elegida, pulsamos sobre el menú flotante de 3 puntos de libro que queremos eliminar.</li> <li>Se abre menú desplegable y el usuario selecciona la opción eliminar.</li> <li>Se muestra un mensaje de confirmación al usuario.</li> <li>El usuario confirma la eliminación pulsando en aceptar.</li> <li>El libro es eliminado del perfil del usuario.</li> <li>Se muestra la lista de la categoría del libro eliminado actualizada.</li> </ol>
<b>Postcondiciones</b>	El libro es eliminado del sistema y por lo tanto de nuestro perfil de usuario.

**Tabla 18. Caso 14 – Eliminar un libro**

<b>CU15: Compartir libro</b>	
<b>Descripción</b>	Comparte con otros usuarios en las principales redes sociales
<b>Actores</b>	Usuario
<b>Precondición 1</b>	<ul style="list-style-type: none"> <li>El usuario accede al detalle de uno de los libros de las diferentes categorías (CU12 completado)</li> </ul>
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla de detalles del libro, pulsamos sobre el icono compartir de la toolbar.</li> <li>Se abre menú desplegable con las principales redes sociales, seleccionamos una y le damos a enviar.</li> </ol>
<b>Precondición 2</b>	<ul style="list-style-type: none"> <li>El usuario accede a una de las categorías de <i>Mis libros</i> (CU09 completado)</li> </ul>
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla de listado de libros, pulsamos sobre el menú flotante de 3 puntos de libro que queremos compartir.</li> <li>Se abre menú desplegable y el usuario selecciona la opción compartir.</li> <li>Se abre menú desplegable con las principales redes sociales, seleccionamos una y le damos a enviar.</li> </ol>
<b>Postcondiciones</b>	El libro es compartido con el usuario o usuarios de la red social elegida.

**Tabla 19. Caso 15 – Compartir libro**

<b>CU16: Añadir una cita</b>	
<b>Descripción</b>	Añadir a un libro que estemos leyendo o leído.
<b>Actores</b>	Usuario
<b>Precondición 1</b>	<ul style="list-style-type: none"> <li>El usuario accede al detalle de uno de los libros de las diferentes categorías (CU12 completado)</li> </ul>

<b>Flujo</b>	<p><b>Opción 1</b></p> <ol style="list-style-type: none"> <li>Desde la pantalla de detalles del libro, en la sección <b>Mis citas</b>, el usuario selecciona añadir cita.</li> <li>Se abre cuadro de texto para escribir o pegar cita.</li> <li>Pulsamos en guardar cita.</li> <li>La cita queda almacenada.</li> </ol> <p><b>Opción 2</b></p> <ol style="list-style-type: none"> <li>Desde la pantalla de detalles del libro, pulsamos sobre el menú flotante de 3 puntos de la toolbar.</li> <li>Se abre menú desplegable y el usuario selecciona la opción añadir cita.</li> <li>Se abre cuadro de texto para escribir o pegar cita.</li> <li>Pulsamos en guardar cita.</li> <li>La cita queda almacenada.</li> </ol>
<b>Precondición 2</b>	<ul style="list-style-type: none"> <li>El usuario accede a una de las categorías de <b>Mis libros</b> (CU09 completado)</li> </ul>
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla de listado de libros, pulsamos sobre el menú flotante de 3 puntos del libro al que queramos añadir cita.</li> <li>Se abre menú desplegable y el usuario selecciona la opción añadir cita.</li> <li>Se abre cuadro de texto para escribir o pegar cita.</li> <li>Pulsamos en guardar cita.</li> <li>La cita queda almacenada.</li> </ol>
<b>Postcondiciones</b>	La cita es añadida en la sección <b>Mis citas</b> en la pantalla de detalles de libro.

Tabla 20. Caso 16 – Añadir una cita

- **Secciones descubrir libros, retos y perfil**

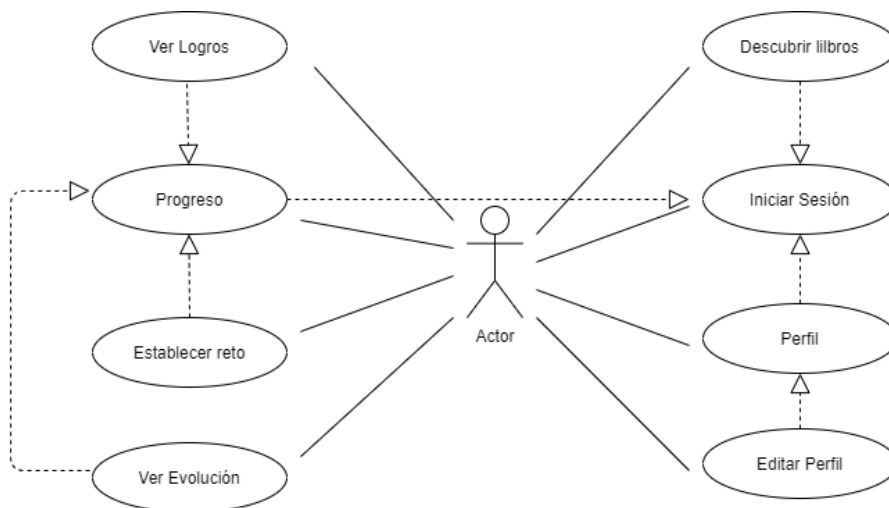


Figura 29 Diagrama UML casos de uso “Sección descubrir, progreso y perfil”

<b>CU17: Acceso a la sección descubrir libros</b>	
<b>Descripción</b>	Acceso a la opción descubrir libros del menú principal
<b>Actores</b>	Usuario
<b>Precondiciones</b>	En algún momento el usuario inició sesión (CU02 completado)
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla principal, seleccionamos el icono de <b>Descubrir</b> en la barra de menú.</li> <li>Nos lleva a la pantalla descubrir libros, aparecerán los libros recomendados, últimas novedades y los más vendidos.</li> </ol>

<b>Postcondiciones</b>	Visualizaremos en la pantalla los libros a descubrir por el lector.
------------------------	---

**Tabla 21. Caso 17 – Acceso a la sección descubrir libros**

<b>CU18: Acceso al perfil</b>	
<b>Descripción</b>	Acceso a la opción perfil del usuario del menú principal
<b>Actores</b>	Usuario
<b>Precondiciones</b>	En algún momento el usuario inició sesión (CU02 completado)
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla principal, seleccionamos el icono de <b>Perfil</b> en la barra de menú.</li> <li>Nos lleva a la pantalla de nuestro perfil, con nuestra configuración correspondiente que podremos modificar.</li> </ol>
<b>Postcondiciones</b>	Visualizaremos en la pantalla nuestro perfil y configuración.

**Tabla 22. Caso 18 – Acceso al perfil**

<b>CU19: Modificar perfil</b>	
<b>Descripción</b>	Modificar los datos del perfil del usuario
<b>Actores</b>	Usuario
<b>Precondiciones</b>	En usuario accede a la pantalla de perfil de usuario (CU18 completado)
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla de perfil de usuario, pulsamos el botón <b>Editar Perfil</b>.</li> <li>Nos lleva a la pantalla de modificar los datos de nuestro perfil</li> </ol>
<b>Postcondiciones</b>	Visualizaremos en la pantalla los datos a modificar de nuestro perfil.

**Tabla 23. Caso 19 – Modificar perfil**

<b>CU20: Acceso a la sección progreso</b>	
<b>Descripción</b>	Acceso a la pantalla de progreso
<b>Actores</b>	Usuario
<b>Precondiciones</b>	En algún momento el usuario inició sesión (CU02 completado)
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla principal, seleccionamos el icono de <b>Progreso</b> en la barra de menú.</li> <li>Nos lleva a la pantalla de nuestro progreso, donde podemos seleccionar ver un resumen de los libros leídos, citas escritas y libros valorados, así como seleccionar diferentes opciones.</li> </ol>
<b>Postcondiciones</b>	Visualizaremos en la pantalla nuestro progreso y diferentes opciones .

**Tabla 24. Caso 20 – Acceso a la sección progreso**

<b>CU21: Establecer un reto</b>	
<b>Descripción</b>	Establecer un reto de lectura
<b>Actores</b>	Usuario
<b>Precondiciones</b>	En usuario accede a la pantalla de progreso (CU20 completado)
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla progreso, pulsamos el botón <b>Retos</b>.</li> <li>Nos lleva a la pantalla de retos</li> <li>Seleccionamos si queremos reto mensual o anual</li> <li>Seleccionamos el número de libros a leer</li> <li>Le damos al botón Comenzar reto.</li> </ol>
<b>Postcondiciones</b>	Establecemos un reto de lectura.

**Tabla 25. Caso 21 – Establecer un reto**



<b>CU22: Ver evolución del usuario</b>	
<b>Descripción</b>	Ver las estadísticas de evolución lectora del usuario
<b>Actores</b>	Usuario
<b>Precondiciones</b>	En usuario accede a la pantalla de progreso (CU20 completado)
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla progreso, pulsamos el botón <b>Evolución</b>.</li> <li>2. Nos lleva a la pantalla de estadísticas de nuestra evaluación</li> <li>3. Seleccionamos si queremos estadística anual, dos años, 5 años.</li> <li>4. Aparecen gráficos estadísticos sobre nuestra lectura de la selección seleccionada.</li> </ol>
<b>Postcondiciones</b>	Visualizaremos pantalla de estadísticas de evolución de lectura.

**Tabla 26. Caso 22 – Ver evolución del usuario**

<b>CU23: Ver logros</b>	
<b>Descripción</b>	Ver los logros conseguidos por el usuario y los pendientes
<b>Actores</b>	Usuario
<b>Precondiciones</b>	En usuario accede a la pantalla de progreso (CU20 completado)
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla progreso, pulsamos el botón <b>Logros</b>.</li> <li>2. Nos lleva a la pantalla de logros</li> <li>3. Pulsando sobre el icono de cada logro se abre cuadro flotante que nos indica las condiciones para conseguirlo.</li> </ol>
<b>Postcondiciones</b>	Visualizaremos pantalla de estadísticas de logros.

**Tabla 27. Caso 23 – Ver logros**

## 2.2.2 Diseño de la arquitectura

Las funcionalidades de la aplicación han quedado establecidas mediante los casos de uso anterior. Llega entonces una de las labores de mayor importancia en el proyecto. Definir la arquitectura del sistema más idónea, la que se adapte mejor, para poder cumplir con las funcionalidades de la aplicación de la manera más eficiente posible.

- **Arquitectura física**

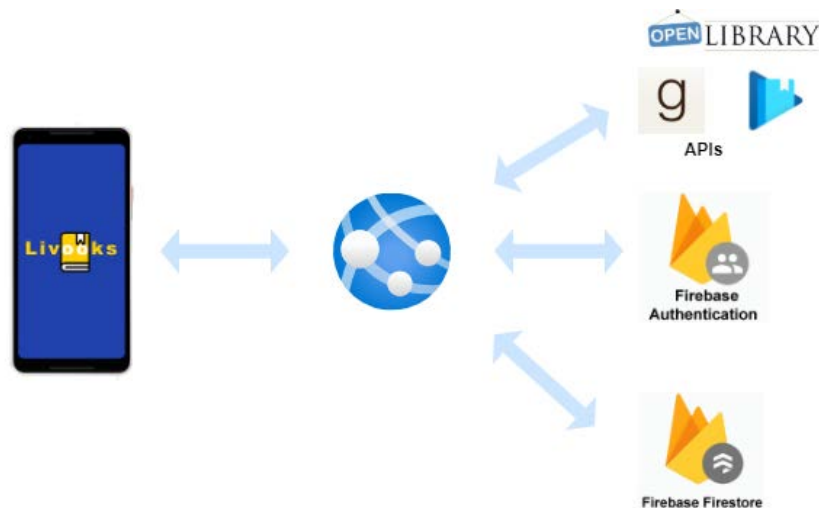
En este caso la aplicación Livooks correrá en dispositivos Android, que se comunicarán con diferentes servidores a través de APIs, lo que se conoce como modelo cliente-servidor. Donde el cliente será nuestra aplicación, la cual solicitará datos a la parte de servidor que será donde esté alojada la base de datos y preparará los recursos necesarios para transmitírselos al cliente.

En la parte de servidor, se utilizará la solución de Google Firebase (**Firestore Authentication**) para la autenticación de los usuarios de manera simple y segura. Ofreciendo varios métodos de autenticación (por ejemplo, mediante correo electrónico y contraseña) y proveedores externos como Google o Facebook, lo cual aprovecharemos en nuestra app.

Se utilizará también la solución de Google FireBase (**Cloud Firestore**) para almacenar y sincronizar datos de los usuarios y entre dispositivos a escala global mediante una base de datos NoSQL alojada en la nube. Esta es la base

de datos más reciente de Firebase para el desarrollo de apps para dispositivos móviles. Aprovecha lo mejor de Realtime Database con un modelo de datos nuevo y más intuitivo por ello nos hemos decantado por esta solución.

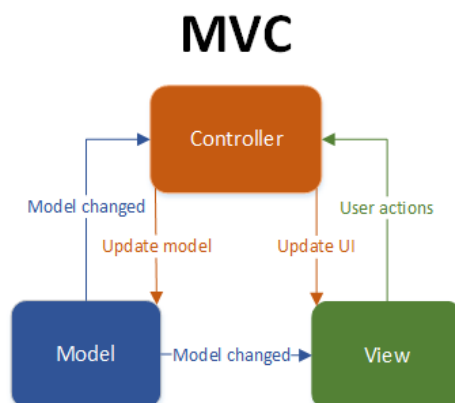
Para finalizar, se ha propuesto la utilización de algunas de las **API públicas Goodreads, Openlibrary y/o Google Books** para consultar los datos relativos a los libros. Dependiendo del resultado de las búsquedas de libros en castellano se utilizará una u otra de forma definitiva en nuestro proyecto.



**Figura 30 Interrelación del cliente con el servicio Backend y las APIs**  
Fuente (Elaboración propia)

- **Arquitectura de diseño del software**

Como patrón de arquitectura de diseño de la aplicación se ha seguido el paradigma **MVC (modelo, vista, controlador)**. La razón de haber elegido este modelo, es que es un modelo muy contrastado para separar la interfaz de usuario de la lógica de negocio, rápido de implementar en una app y menos complejo, con el que el autor de esta memoria tiene más experiencia. Igualmente, una de las grandes ventajas que posee esta técnica es que nos permitirá modificar cada uno de los componentes (modelo, vista, controlador) sin necesidad de modificar los demás, lo que permite que la app sea modular, escalable y que se puedan actualizar fácilmente.

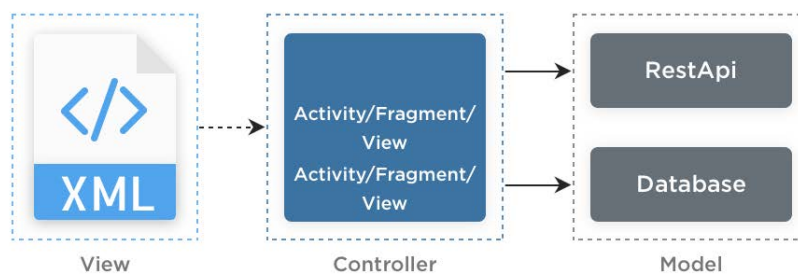


**Figura 31 Representación gráfica arquitectura MVC**  
Fuente (patrones arquitectónicos en Android [\[11\]](#))

**Modelo:** La capa de datos, representa el conjunto de entidades que van a conformar el “cerebro de la aplicación”, gestiona todos los accesos y consultas a bases de datos o APIs de terceros.

**Vista:** Es la capa visual, será nuestra interfaz, la visualización de nuestro modelo en un formato adecuado para interactuar. Desplegaremos botones, campos de texto, imágenes, con el fin de permitir la interacción con el usuario.

**Controlador:** Es el intermediario entre la vista y el modelo. Responde a eventos (usualmente acciones del usuario) e invoca peticiones al ‘modelo’ cuando se hace alguna solicitud sobre la información. Puede por ejemplo tener la lógica de un botón al pulsar sobre él, llevarnos a otra vista, actualizar, decirle al modelo que proporcione más datos. Convierte el modelo a algo que pueda representarse mediante la vista.



**Figura 32 Asimilación gráfica arquitectura MVC en Android**  
Fuente (patrones arquitectónicos en Android [\[12\]](#))

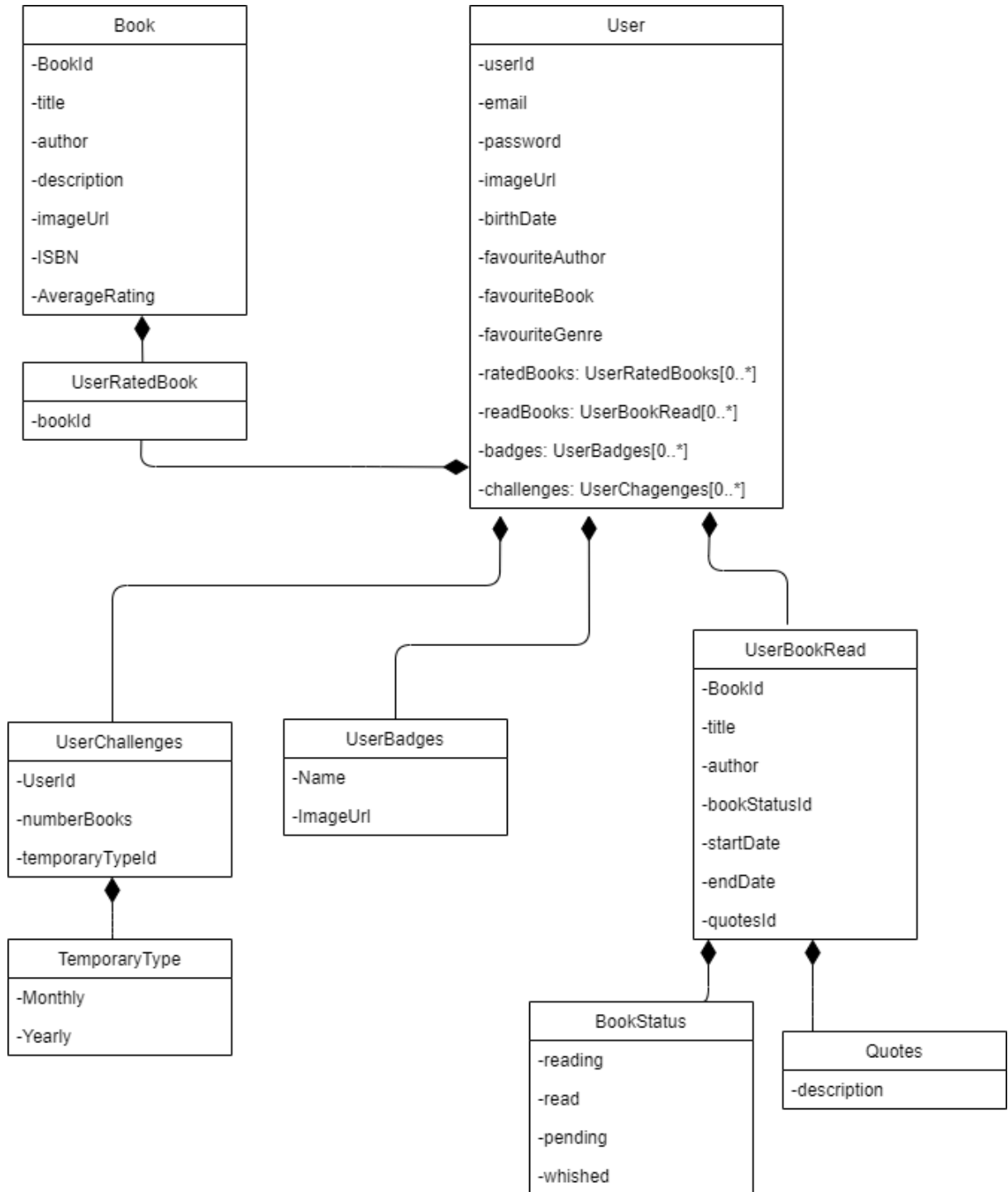
Bien es cierto, que este modelo se va quedando anticuado frente a otros modelos de arquitectura más modernos como el MVP o el MVVC que separan más si cabe los datos y su presentación de la lógica de la aplicación, buscando el principio de “single responsibility” y la facilidad de implementar los test unitarios. Pero al no tener mucha experiencia y con el corto tiempo que tenemos para la implementación de la app habría mucho riesgo de no ser capaz de tenerla lista.

- **Arquitectura de datos**

En este punto se detallan como se almacenarán y estructurarán los datos que el sistema usará para su funcionamiento. Será una primera revisión que seguro que se irá modificando a lo largo del proyecto.

Como se comentó en el diseño de arquitectura física, el servicio que utilizaremos como Backend será Cloud Firestore que es una base de datos NoSQL alojada en la nube. No tendremos ni registros ni tablas, almacena los datos en documentos que contienen campos que se asignan a valores, por lo que no es necesario crear el esquema de datos antes de empezar a utilizarla. Estos documentos se almacenan en colecciones, que usaremos como contenedores para los documentos y que usaremos para organizar los datos y compilar consultas.

No obstante, relacionado en gran medida con la base de datos y a modo de resumen, se ha creado el siguiente diagrama de clases que nos permite representar las clases y objetos que se emplearán en el proyecto, así como la relación de los mismos.



### 2.3 Revisión de la planificación

La planificación ha ido conforme a lo esperado y no ha tenido que ser retocada. Bien es cierto, que se ha consumido todo el tiempo e incluso me han faltado horas donde podría haber pulido alguna pantalla o diagrama realizado, pero igual me hubiera sucedido si hubiera tenido unos días más.

Me he dado cuenta que me cuesta discernir cuando se ha obtenido una calidad suficiente, en la cual no merezca la pena el esfuerzo de tiempo invertido para mejorarla y poder dedicarlo a la siguiente tarea. Esto podría hacer que con el poco tiempo que tenemos para la fase de implementación se pudieran quedar funciones de la aplicación sin realizar, por lo que sabiéndolo de antemano se trabajará en mejorar este aspecto.

### 3. Implementación

Durante los apartados anteriores se ha visto el contexto y necesidad del producto, así como su diseño conceptual y de arquitectura. Con lo que ahora podremos dar un paso firme hacia la implementación de nuestra aplicación para que pase del papel a un producto real.

Veremos las principales herramientas utilizadas para el desarrollo del producto, aquellos aspectos de especial importancia en el desarrollo de la codificación (sin adentrarnos en los detalles de código) y los resultados obtenidos con una muestra visual de su funcionalidad. Para finalizar se comentarán las pruebas realizadas que aseguran la fiabilidad del producto antes de salir a producción.

De forma muy resumida, diremos que en esta fase se trata de escribir el código fuente en el lenguaje de programación escogido utilizando las mejores técnicas y herramientas a nuestra disposición para convertir el prototipo en un producto real.

#### 3.1 Herramientas

Para llevar a cabo la implementación del proyecto se han utilizado una serie de herramientas, algunas imprescindibles para el desarrollo de la aplicación y otras que no han ayudado a ser más eficientes y ahorrar mucho código.

- **IDE Android Studio**

Como vimos en apartado anteriores, la aplicación se ha desarrollado utilizando el IDE Android Studio, en concretamente se ha utilizado la versión 4.1.1, la versión más reciente durante la elaboración de este documento. En un principio se había pensado al API19, como versión mínima para el desarrollo de la aplicación, pero al final se ha realizado para la API21, ya que igualmente, se llega un porcentaje de usuarios superior al 94%, adquiriendo un compromiso equilibrado entre llegar al mayor número potencial de usuarios y soporte para las mejores prestaciones del sistema operativo como el aprovechamiento de Material Design.

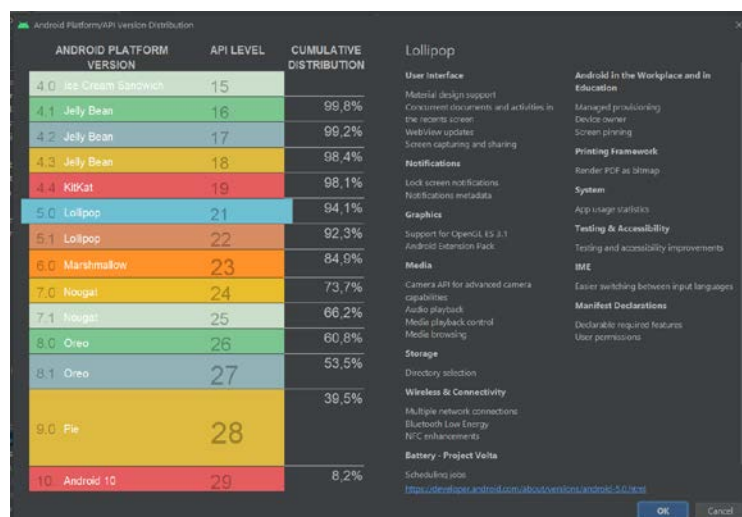


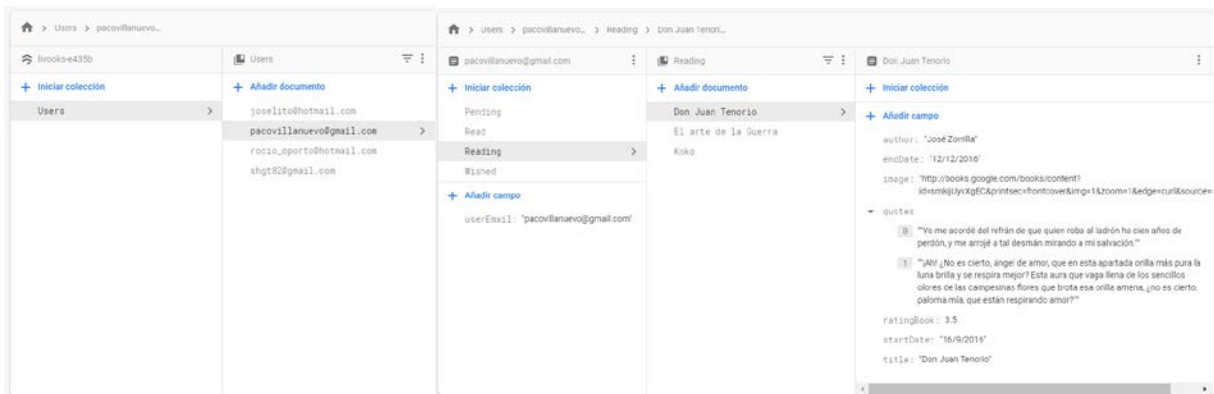
Figura 33 Distribución de las diferentes versiones de Android

- **Firestore**

Esta plataforma de Google ofrece diferentes servicios en la nube fácilmente integrables en aplicaciones web y móviles, que evitan que el desarrollador tenga que implementar la lógica de servidor (Backend). Como vimos brevemente en el apartado de arquitectura de la app, se han utilizado los servicios de autenticación (Authentication) y de base de datos en tiempo real (Cloud Firestore).

### **Firestore Cloud Firestore**

Como ya se comentó, Cloud Firestore es la base de datos más reciente de Firebase para el desarrollo de apps para dispositivos móviles y la recomendada por Firebase cuando se comienza un proyecto nuevo. En la siguiente imagen podemos ver el esquema de uso para la aplicación.



**Figura 34. Esquema del uso de base datos Firestore para la app**

Como se puede observar en la imagen anterior, cada usuario dispone de una colección de diferentes libros organizados en documentos que representan la categorías o estado en el que se encuentra el libro (leyendo, leído, pendiente o lista deseados).

### **FirestoreUI Auth**

Para dar todas las facilidades de registro a los usuarios se ha utilizado Firebase Auth como método de autenticación, pudiendo hacerlo mediante correo electrónico, cuenta de Google, cuenta de Facebook, o cuenta de Twitter. Se ha necesitado para ello abrir una cuenta de desarrollador tanto en Facebook como en Twitter para poder implementar estas alternativas de registro.

Se ha aprovechado la IU de código abierto y totalmente personalizable que nos brinda Firebase para la implementación de la interfaz de registro consiguiendo un resultado bastante atractivo.

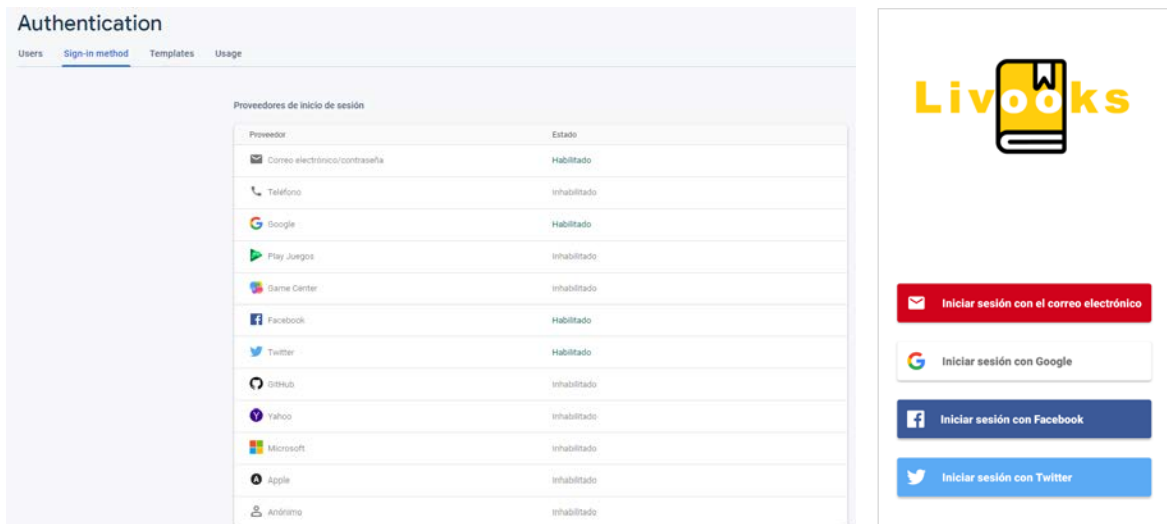


Figura 35. Registro mediante FirebaseUI Auth utilizado en Livooks

- **Librerías**

Se ha realizado apoyo de las siguientes librerías para facilitar la implementación y desarrollo del proyecto:

**Volley** es una librería desarrollada por Google para optimizar el envío de peticiones Http desde las aplicaciones Android hacia servidores externos. Este componente actúa como una interfaz de alto nivel, liberando al programador de la administración de hilos y procesos tediosos de parsing, para permitir publicar fácilmente resultados en el hilo principal. En cuanto a su elección en lugar de Retrofit u OkHttp se ha basado en su facilidad de implementación y la importancia de que Google esté detrás de esta librería.

<https://developer.android.com/training/volley?hl=es>

**Gson** es una librería de código abierto creada por Google, que permite serializar objetos Java para convertirlos en un String. Su uso más frecuente es para convertir un objeto en su representación JSON y a la inversa.

<https://github.com/google/gson>

**Picasso** es una librería para Android que se utiliza para gestionar las imágenes utilizando técnicas de cacheo. Esto consiste en que la librería se encarga de almacenar/recuperar las imágenes de la memoria volátil o interna del dispositivo o de descargarla de internet en caso necesario. También se encarga de eliminarla cuando ya no la utilizamos. Se podría haber elegido igualmente la librería Glide, ya que las dos son muy similares, me he decantado por Picasso por ocupar menos espacio.

<https://square.github.io/picasso/>

**Material.io** Librería para el uso de componentes Material Design, simplifica la creación de interfaces gráficas atractivas y usables para los usuarios.



<https://material.io/develop/android/docs/getting-started>

**ZXing Android Embedded** Librería para el escaneo de código de barras. La cámara se administra en un hilo en segundo plano, para un tiempo de inicio rápido.

<https://github.com/journeyapps/zxing-android-embedded>

- **APIs**

Para la búsqueda de libros se ha utilizado la **API de Google Books**, se han barajado otras librerías como Goodreads u open Library Books, pero se ha seleccionado Google Books por ser la más potente, y la que más libros encuentra tanto en español como en inglés. En un futuro está pensada la implementación de combinación de las diferentes API, que sería lo ideal.

<https://developers.google.com/books>

Básicamente se puede acceder a ella de la siguiente forma:

***<https://www.googleapis.com/books/v1/volumes?q=FRASE A BUSCAR>***

Con el fin enriquecer la experiencia de usuario, se ha optado por automatizar la búsqueda de libros de forma que conforme el usuario vaya escribiendo en el campo de búsqueda el sistema muestre el conjunto de resultados, estableciendo un mínimo número de caracteres necesarios para comenzar a mostrar los resultados.

- **Git**

Software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente. Su propósito es llevar registro de los cambios en archivos de computadora y coordinar el trabajo que varias personas realizan sobre archivos compartidos.

Como no pueda faltar en el desarrollo de cualquier aplicación, se ha implementado Git como control de versiones de la aplicación. No tanto para la coordinación de trabajos por varias personas, que en este caso no tenía lugar, sino para llevar el registro de cambios ante cualquier contratiempo o anomalía se pudiera recuperar la información.

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

- **GitHub**

GitHub es un portal para gestionar las aplicaciones que utilizan el sistema Git y que fue comprada por Microsoft en junio del 2018. La plataforma está creada para que los desarrolladores suban el código de sus aplicaciones y herramientas, y que como usuario no sólo puedas descargar la aplicación,

sino también entrar a su perfil para leer sobre ella o colaborar con su desarrollo.

He utilizado la plataforma GitHub como repositorio para la aplicación pudiendo ser accedida desde el siguiente enlace.

<https://github.com/duende23/Livooks>

### 3.2 Aspectos destacados del proceso de implementación

- **Estructura del proyecto**

La estructura del proyecto ha seguido el patrón genérico ofrecido por Android Studio incluyendo algunos directorios para la organización del modelo de datos, fragmentos, adaptadores, interfaces, providers, y operaciones realizada con base de datos NoSQL Firebase Firestore.

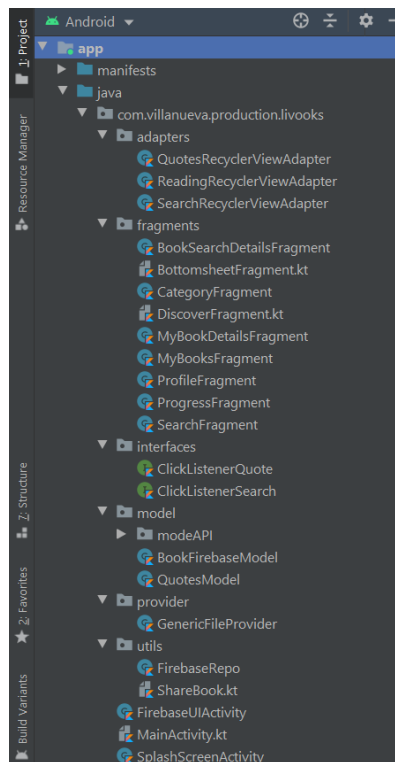


Figura 36. Estructura del proyecto

#### **Model**

Contiene el modelo de datos de la aplicación. Para extraer el modelo de datos de la API Google Books se ha utilizado el plugin *JSON to kotlinClass*, ahorrándonos trabajo.

#### **Fragments**

Este directorio contiene los fragmentos usados en la aplicación.

## **Adapters**

Este directorio contiene los adaptadores utilizados en la app para mostrar RecyclerViews personalizados.

## **Intefaces**

Este directorio contiene las interfaces utilizadas para interactuar con el listado de datos del RecyclerView teniendo bien separadas la parte lógica de la vista.

## **Provider**

Directorio de los proveedores de contenido. Que es una subclase de *ContentProvider* que proporciona acceso estructurado a los datos que administra la aplicación. Todos los proveedores de contenido de la aplicación se definen en un elemento en el archivo de manifiesto. Se ha utilizado para compartir el libro con otros usuarios.

## **Utils**

Directorio que contiene las principales clases y métodos que utilizamos en la app. Cabe destacar un directorio propio para todas las operaciones CRUD relacionadas con la base de datos Firebase Firestore.

- **Adaptación al uso de distintos idiomas**

Aunque bien es cierto que de momento sólo está disponible en español, todo el texto del proyecto se ha incluido en el fichero "strings.xml", siguiendo manual de buenas prácticas de Android Studio, de forma que sea fácilmente proyectable a otro idioma con sólo introducir líneas extras para cada idioma en una futura versión.

- **Seguridad**

Se han aplicado reglas de seguridad en la base de datos para que sólo puedan ser accedidas y modificadas por usuarios registrados en la aplicación.

### 3.3 Funcionalidad y estado de la implementación de la aplicación

Tras el desarrollo conceptual y justificación de la solución tecnológica adoptada durante la memoria para el desarrollo de la aplicación, en este apartado se muestran los resultados obtenidos en la implementación, destacando los puntos más destacados. Bien es cierto, que no ha podido llevarse a cabo el desarrollo de la funcionalidad de retos requerida al inicio del proyecto por la falta de tiempo.

- **Splash Screen**

Punto de inicio de la aplicación, constituido por el logo de la aplicación. Su duración está preestablecida a un tiempo fijo, durante la misma comprobaremos si el usuario está registrado o no y dependiendo de ello la app nos dirigirá a la pantalla de registro o a la pantalla principal.



Figura 37. Pantalla de inicio de la aplicación. Splash Screen

- **Pantalla de registro**

Se ha realizado la codificación para que se pueda iniciar sesión con las principales redes sociales Facebook, Twitter o Google, así como el registro con email muy breve si no se está registrado. Se ha implementado la posibilidad de recuperar nuestra contraseña olvidada en el caso de que no nos acordemos. La forma seleccionada para nuestra aplicación ha sido la de recuperación mediante envío de enlace al correo electrónico, siendo este mensaje de correo personalizable.

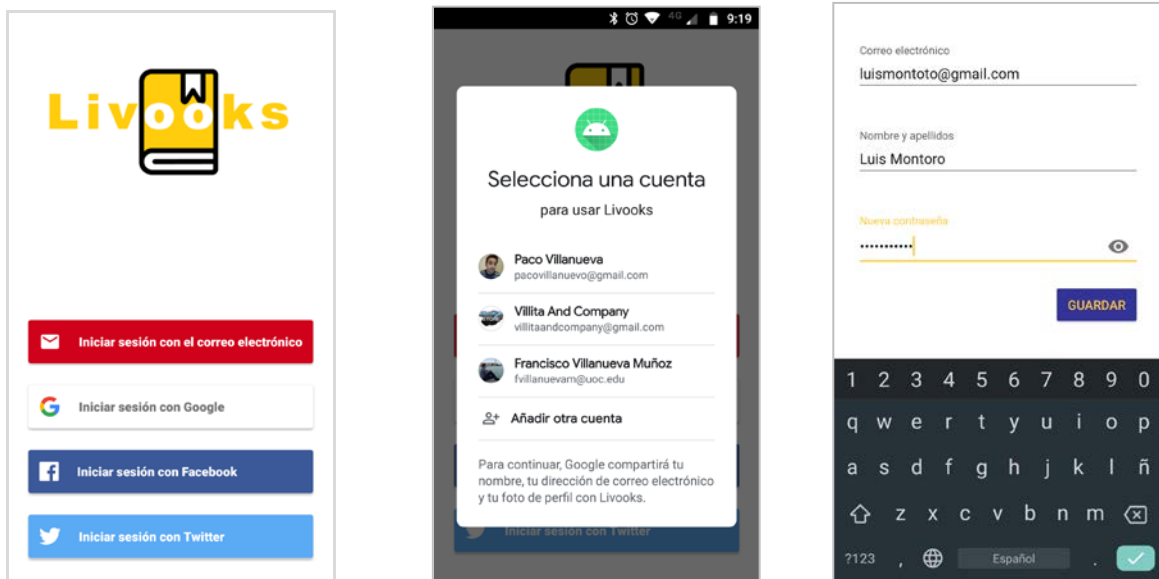


Figura 38. Pantalla de registro y modelo registro mediante Google y correo

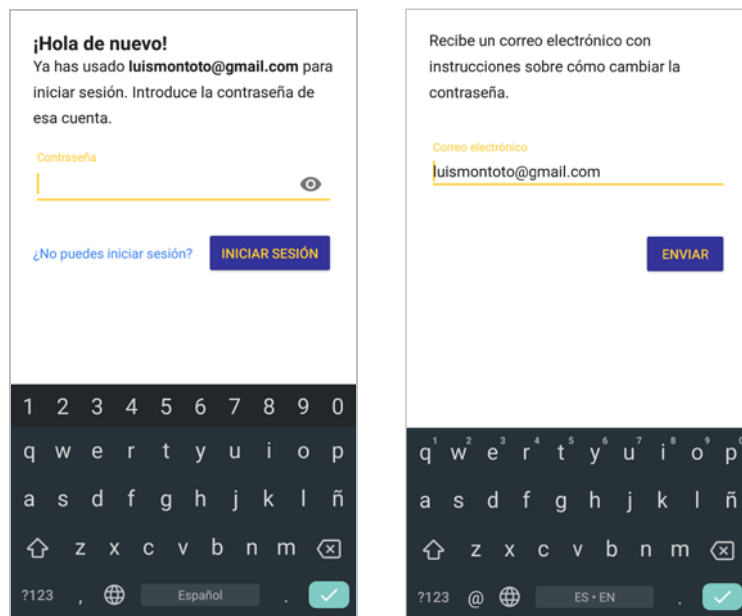


Figura 39. Pantalla de recuperación de contraseña

- **Pantalla Principal**

En la pantalla principal se ha implementado un **Navigation View** para navegar entre las principales secciones de la aplicación que estarán codificadas en diferentes fragmentos y una **Toolbar** personalizada para la búsqueda de libros tanto de forma manual como escaneando el código de barras del libro con la cámara del dispositivo.



Figura 40. Pantalla principal

Las diferentes secciones de la aplicación serán:

**Mis Libros**, será la sección por defecto en la que nos encontramos tras abrir la app y hacer login exitosamente.

**Progreso**, será la sección donde podamos ver nuestro progreso de lectura, así como personalizar retos.

**Descubrir**, será la sección donde podamos ver recomendaciones de libros personalizadas y las últimas novedades de libros.

**Perfil**, será la sección donde podamos editar nuestro perfil, cerrar la sesión y las principales opciones de configuraciones de la aplicación.

- **Pantalla Mis Libros**

Como se comentó anteriormente será el fragmento que aparezca por defecto al loguearnos correctamente en la app. Tendremos las diferentes categorías o estados de los libros que tenemos almacenados pudiendo acceder a cualquiera de ellas.

Se ha implementado una vista **CardView** para cada una de las categorías, para darle un aspecto visual agradable, y se ha incluido el número de libros que tenemos almacenados en cada categoría extraído de la base de datos NoSQL implementada Firestore.

Se ha diseñado un fragmento común para las categorías, el cual modificará su contenido en función de cada una de éstas.

- **Pantalla Búsqueda**

En la pantalla de búsqueda se ha implementado un **RecyclerView** para mostrar los resultados de la búsqueda proporcionada por la API Google Books de la que hablamos anteriormente.

Mientras se cargan los datos desde la API, mostraremos una **ProgressBar** indicando al usuario que se está ejecutando la operación solicitada.

Se han combinado las librerías **Volley** y **Gson** para obtener los resultados de la API de la manera más sencilla posible ahorrándonos trabajo de codificación.

Para la búsqueda de libros mediante su código de barras se ha implementado la librería **ZXing Android Embedded**.

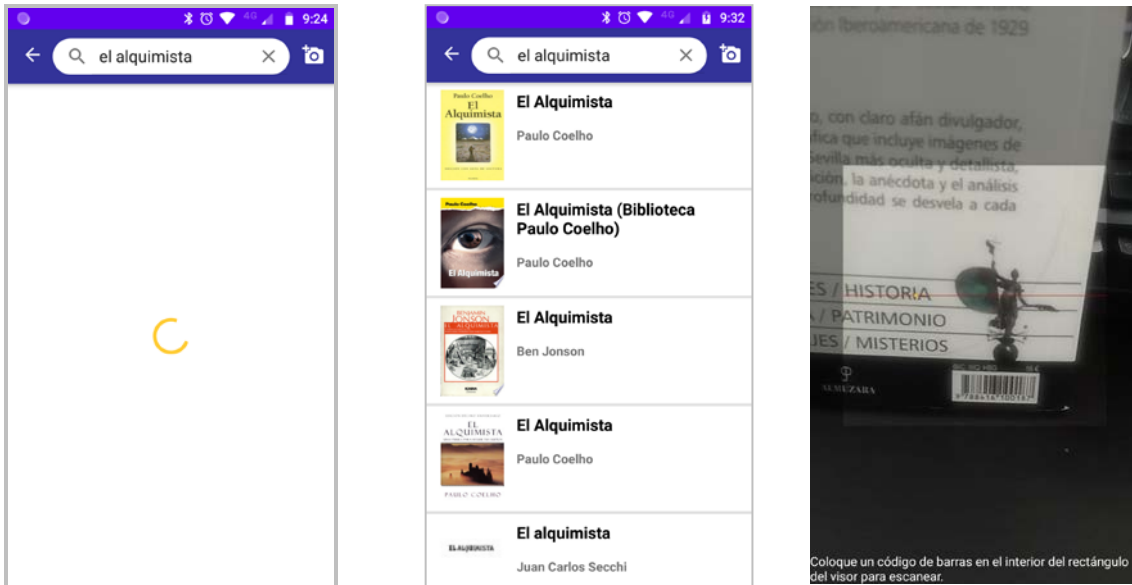


Figura 41. Pantallas de búsqueda tanto manual como mediante escaneo

- **Pantalla detalles libro buscado**

Tras elegir el libro deseado en el resultado de la búsqueda, iremos a la pantalla de detalles del libro, donde se ha implementado una **Collapsing Toolbar** que le da un aspecto profesional junto con la descripción del libro, la valoración de los lectores y el género.

Se ha implementado un botón desplegable **PopUp Menu** para seleccionar la categoría en la que nos gustaría guardar el libro (leyendo, leídos, pendiente de leer o lista de libros deseados para tener en propiedad). Para finalizar se ha implementado un **Rating Bar** para poder ver la valoración media del libro por los lectores.

Cabe destacar, que podemos compartir el libro con otros usuarios, a los que se enviará la imagen y el texto mediante cualquier aplicación de mensajería. Para ello se ha implementado un botón de compartir en la toolbar.

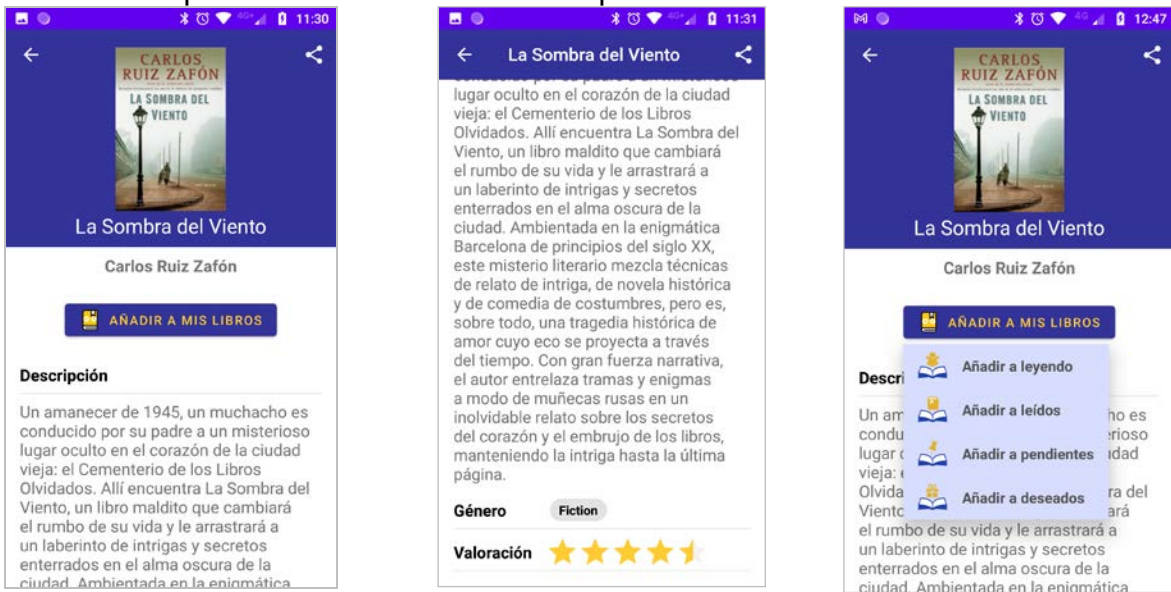


Figura 42. Pantalla detalles del libro buscado

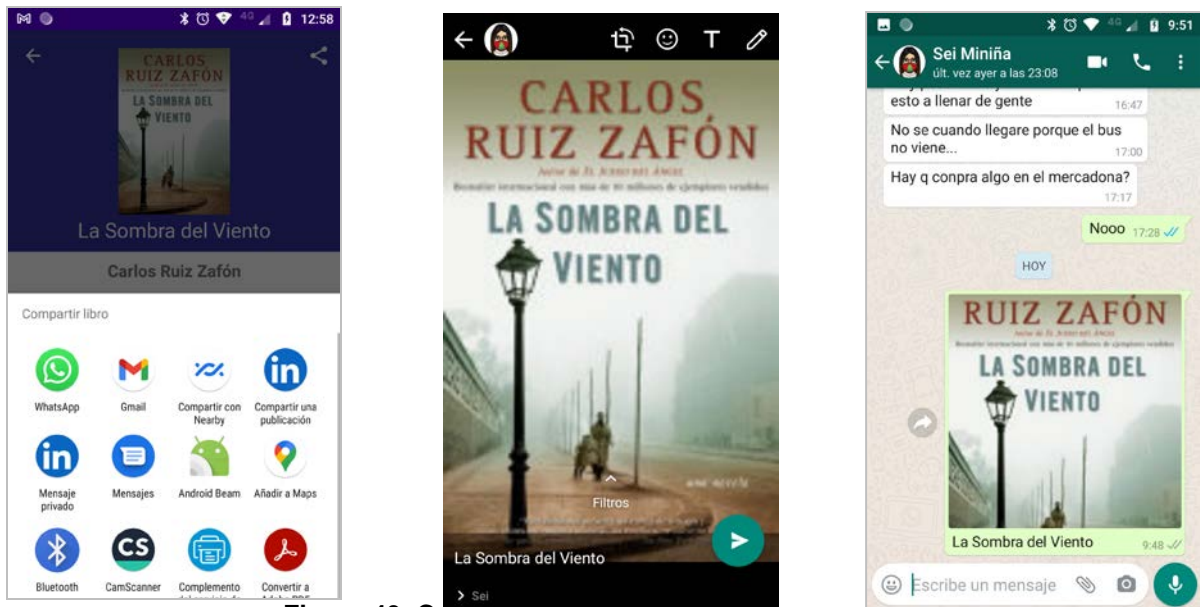


Figura 43. Compartir libro con otras personas

- **Pantalla libros en cada categoría**

Esta pantalla estará formada por un fragmento con un RecyclerView que será rellenado con los libros almacenados pertenecientes a la categoría seleccionada. Aparecerá el listado representando cada libro en un CardView con la imagen del libro, título y autor.

Un aspecto muy importante en esta pantalla es que se ha implementado un botón de menú en cada CardView del libro, donde se desplegara un panel deslizante **Bottom sheet**, donde podemos realizar las siguientes funciones.

- Mover el libro a diferentes categorías
- Compartir libro
- Eliminar libro de la lista

Si pulsamos sobre uno de los libros nos llevará a la pantalla de detalles.

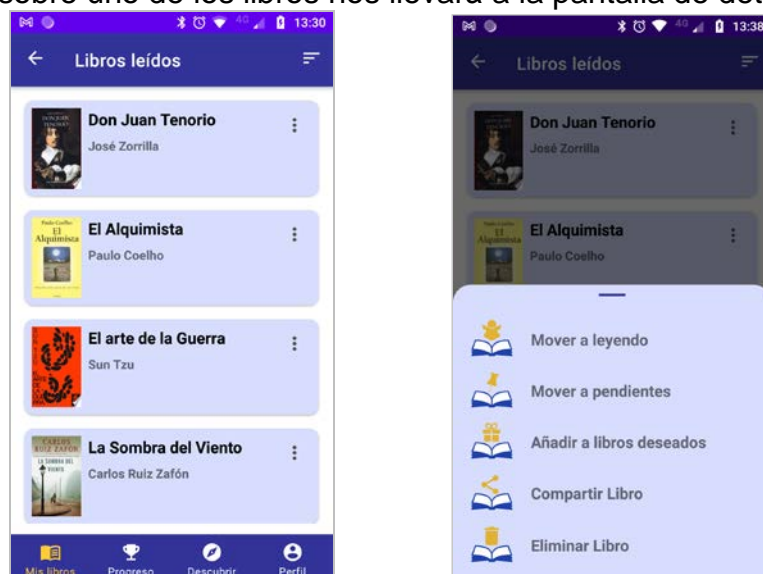


Figura 44. Categorías libros



- **Pantalla detalles libro**

En la pantalla de detalles del libro almacenado seguimos manteniendo la Collapsing Toolbar al igual que en la pantalla detalles del libro buscado, pero en este caso ira acompañada de un menú que desplegará un panel deslizante *Bottom sheet* donde se podrán realizar las mismas funciones que comentamos en el apartado anterior.

En el diseño del fragmento, se han implementado cuatro iconos, para cambiar la categoría o estado del libro de forma más directa, dando una estética muy cuidada.

Como detalles del libro podemos colocar una fecha de inicio de lectura y una fecha de fin de lectura implementadas a través de un **DatePicker** para seleccionar la fecha de una forma muy intuitiva para el usuario.

Se codificado también una ratingBar para tener almacenada nuestra propia valoración del libro de 1 a 5 estrellas.

Para finalizar, como una de las funciones más interesantes, tendremos la posibilidad de insertar una cita del libro que nos haya gustado. Se ha implementado un botón de insertar cita donde aparece un cuadro de diálogo personalizado **AlertDialog** en el que podremos escribir o pegar la cita que nos haya gustado. Una vez guardada la cita, ésta quedará almacenada apareciendo en un listado que se ha implementado mediante un RecyclerView incluyendo el número de citas que tenemos almacenadas que podrán ser eliminadas o editadas.

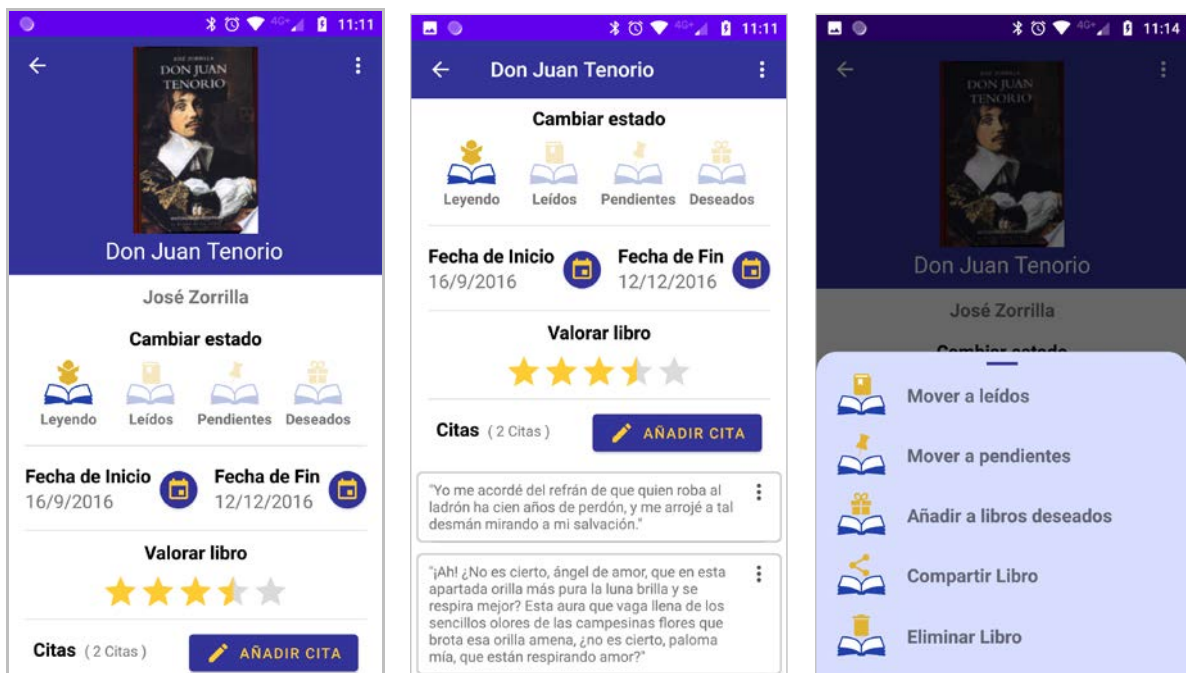


Figura 45. Pantalla detalles del libro almacenado

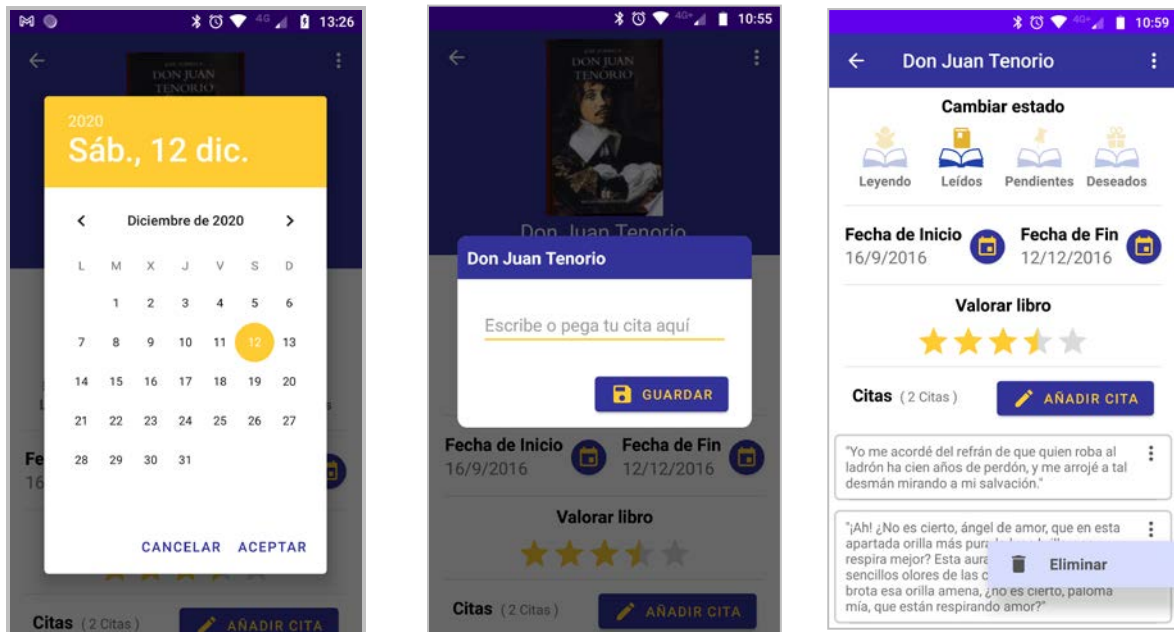


Figura 46. Selección de fecha de lectura, añadir y eliminar citas

Las pantallas de la sección de progreso y descubrir no han podido ser implementadas totalmente por falta de tiempo.

### 3.4 Plan de pruebas

Toda aplicación debería contener un conjunto de pruebas amplio para cada uno de los siguientes tipos:

- Pruebas de unitarias
- Pruebas de integración
- Pruebas de instrumentación o de interfaz de usuario

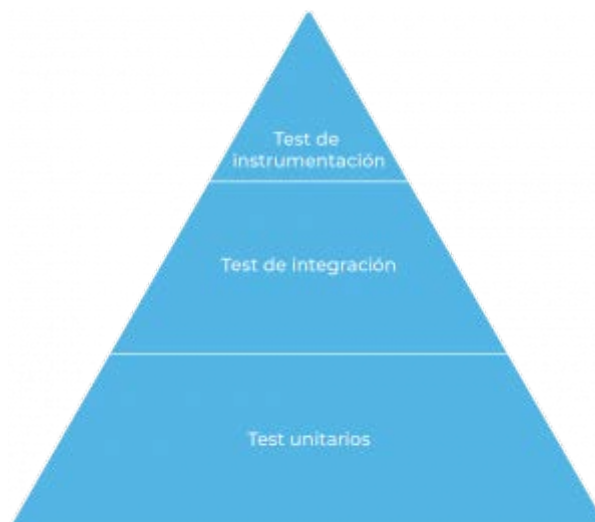


Figura 47. Pirámide del conjunto de test para una aplicación (Fuente: Adictos al trabajo. Pruebas de instrumentación en Android) [13].

Según la pirámide de tests, la base de nuestro proyecto debe estar formada por los test unitarios. Son los más importantes, ya que, aunque cada escalón que subimos en la pirámide de pruebas, aumenta en fidelidad, también aumenta en tiempo de ejecución y esfuerzo para mantener y depurar. Es aconsejable que este tipo de tests engloben el 70% de los tests de nuestra aplicación. Son los más rápidos de hacer y no dependen del emulador o dispositivo real. Las herramientas más utilizadas para realizar test unitarios son **JUnit y Mockito**.

En el siguiente nivel de la pirámide se encontrarían los tests de integración, siendo aconsejable que engloben el 20% de los test de nuestra aplicación. Este tipo de test nos ayudarán a comprobar cómo se comporta nuestro código no sólo de manera individual, sino cuando interactuamos con otras partes del proyecto. La herramienta más común para realizar este tipo de tests es **Robolectric**, con la que no nos hará falta ni emulador, ni dispositivo real (aunque es muy aconsejable).

En el nivel más alto de la pirámide se encontrarían los test de interfaz de usuario o instrumentación (UI Tests), siendo aconsejable no superar el 10 % de los tests de nuestra aplicación. Emulan el comportamiento del usuario y ratifican los resultados de la interfaz de usuario. Las herramientas más utilizadas para realizar este tipo de tests son **Expresso y UI Automator**.

Los desarrolladores se tienen que centrar sobre todo en los test unitarios porque son mucho más fáciles de realizar y rápidos de ejecutar que los de integración o UI Tests, al no necesitar emulador ni dispositivo físico que retardaría las pruebas. Ni si quiera necesitamos ninguna librería de tests de Android. **La idea detrás de las pruebas unitarias es que, si puedes probar que todas las unidades individuales de código funcionan como se espera, podemos confiar que, al agruparlas, la aplicación en su conjunto también funcionará como se espera.** Al fin y al cabo, la aplicación está formada por todas esas unidades pequeñas que funcionan correctamente. Por ello es clave lograr una alta cobertura de pruebas unitarias de nuestra aplicación

Debido a la falta de recurso temporal no ha sido posible la realización de pruebas adecuadas durante el desarrollo del proyecto, sino que se han ido realizando de manera manual tras implementar cada nueva funcionalidad o actualización. Por lo que el desarrollo de las pruebas comentadas son las que deberían existir a futuro en el proyecto.

### 3.5 Revisión de la planificación

Al contrario que en anteriores revisiones de planificación, ésta no ha ido conforme lo esperado y se ha tenido que ir retocando sobre la marcha dejando algunas funcionalidades de la aplicación sin poder realizar, ya no por complejidad que pudieran conllevar, sino por tiempo.

He consumido las 216 horas para la implementación, y un esfuerzo extra consiguiendo 40 horas más, para al menos dejar funcionalidades principales debidamente implementadas.

Del análisis de las causas de este desvío de horas hombre se pueden establecer los motivos principales:

- Falta de experiencia a la hora de realización de una planificación realista y acorde a la envergadura del proyecto, siendo ésta demasiado optimista.
- No tener en cuenta el gran porcentaje de tiempo que conlleva el aprendizaje de nuevas tecnologías (librerías, frameworks, APIs, Rest server) y su implementación. He tardado más tiempo de lo esperado en implementar la API de Google Books y comprender el funcionamiento de base de datos NoSQL Firebase Firestore, con las que nunca había trabajado.

Sinceramente la implementación ha sido un baño de realidad frente al optimismo que me acompañaba al presentar todas las funcionalidades de la aplicación, del que aprenderé a base de experiencia, repetición y retos progresivos.

## 4. Conclusiones

Llegados a este punto, quiere decir, que se ha concluido finalmente la implementación del producto con mayor o menor éxito. Se presentarán las principales conclusiones obtenidas del trabajo realizado, así como las líneas futuras de desarrollo y potencial de la aplicación.

### 4.1 Resultados obtenidos y aprendizaje

En el aspecto global del proyecto, se consideran cumplidos los principales objetivos del proyecto de forma satisfactoria. Bien es cierto, que quedan por implementar algunas funcionalidades que por falta del recurso temporal no han podido llevarse a cabo, como la sección de retos y recomendaciones de libros, que se continuarán como reto personal. A lo largo del proyecto, el nivel de evolución y mejora de competencias ha sido muy apreciable.

- He mejorado notablemente mis competencias, pues nunca había trabajado con librerías utilizadas como Zxing, Volley o Gson. La extracción de datos de APIs o la utilización de una base de datos NoSQL como Firebase Firestore le han dado amplitud a la visión que tenía sobre el desarrollo de una aplicación. Ahora tengo una buena base para poder afrontar nuevos proyectos y desarrollarlos de forma más ágil.
- Uno de los principales retos era realizar la codificación en Kotlin, lenguaje de futuro, y la verdad que las impresiones han sido muy buenas.
- La búsqueda autónoma de información, documentación, tutoriales sobre cualquier nueva tecnología a implementar en el proyecto o problema con el que estemos lidiando.

Realmente ha sido gratificante ver cómo ha ido evolucionando el proyecto, algo que veía en un principio altamente complicado poco a poco se ha ido desarrollando y eliminando dificultades hasta conseguir un producto del que me siento orgulloso y que seguiré mejorando. El desarrollo de este trabajo, desde las fases iniciales hasta la obtención del producto, ha servido para que el autor pudiera tomar conciencia de la complejidad que supone planificar y llevar a cabo un proyecto real de esta naturaleza.

Durante la implementación de la aplicación es donde he encontrado mayor problema como era de esperar a la hora de realizar el TFM. Me he topado con ciertas dificultades que no había previsto o valorado correctamente a la hora de realizar la planificación, obteniendo una lección muy valiosa para afrontar futuros proyectos.

**Como reflexión personal, no puedo estar más satisfecho con el trabajo realizado, no ya por la calidad del mismo, que siendo mi mayor crítico es más que mejorable en algunos aspectos, sino, por el aprendizaje adquirido a lo largo del desarrollo del proyecto. El refuerzo y afianzamiento de conceptos estudiados a lo largo del máster, en un**

**contexto cercano a la realidad de entrega de un proyecto acabado, ha sido una experiencia muy enriquecedora.**

#### 4.2 Líneas futuras del proyecto

Las líneas futuras del proyecto son amplias, ya que la aplicación tiene mucho potencial, destacando lo siguiente:

- Implementación de la sección de retos
- Implementación sección descubrir libros para recomendaciones automáticas y novedades en cuanto a libros.
- Optimizar la app para consumir el menor número de recursos posibles, como ram y batería ya que son componentes críticos que valoran mucho los usuarios a la hora de desinstalar una app.
- Pasar a producción, subir aplicación a Google Play
- Implementación para el sistema operativo iOS y subir a AppleStore

Así, una vez lanzada la aplicación en las principales plataformas, se irán implementando nuevas funcionalidades en versiones posteriores:

- Funciones de socialización
- Selección de idioma
- Modo oscuro
- Permitir la edición de datos de un libro registrado, por ejemplo, cambiar la foto de portada.
- Contemplar todas las opciones que vayan demandando los usuarios escuchando y respondiendo a sus valoraciones.

## 5. Glosario

**App** *f* Término utilizado para referirse a las aplicaciones de dispositivos móviles.

**API** *f* Siglas de *application program interface*. Interfaz expuesta para ser utilizada dentro de una aplicación con el objetivo de dar acceso a librerías o funciones externas a la aplicación.

**Framework** *m*: estructura conceptual y tecnológica de asistencia definida que puede servir de base para la organización y desarrollo de software, y que puede contener programas, bibliotecas, etc.

**Android** *m* Sistema operativo móvil desarrollado por Google

**iOS** *m* Sistema operativo para dispositivos móviles de iPhone.

**IDE (*Integrated development environment*)** *m* Entorno integrado de desarrollo, conjunto de herramientas que dan soporte a la implementación, despliegue, test y depuración de aplicaciones.

**ISBN** *m* Identificador único para libros, previsto para uso comercial.

**DCU (Diseño centrado en el usuario)** *m*. Diseñar situando al usuario en el centro de todo el proceso.

**Firestore de Google** *f* Plataforma en la nube para el desarrollo de aplicaciones web y móvil.

**Firestore Authentication** *f* Permite tanto el registro propiamente dicho (mediante email y contraseña) como el acceso utilizando perfiles de otras plataformas externas (por ejemplo, de Facebook, Google o Twitter), una alternativa muy cómoda para usuarios reacios a completar el proceso.

**Google Play** *f* Plataforma de distribución digital de aplicaciones móviles para los dispositivos con sistema operativo Android.

**Backend** *m* Conjunto de acciones que pasan en una aplicación pero que no vemos como, por ejemplo, la comunicación con el servidor.

**Notificaciones Push** *f* Mensajes instantáneos que recibes en tus dispositivos.

**CRUD** *m* Hace referencia a un acrónimo en el que se reúnen las primeras letras de las cuatro operaciones fundamentales de aplicaciones persistentes en sistemas de bases de datos: Create (Crear registros), Read (Leer registros), Update (Actualizar registros) Y Delete (Borrar registros).

**MVC** *m* Model View Controller, es un patrón de arquitectura de software, que separa los datos y principalmente lo que es la lógica de negocio de una

aplicación de su representación y el módulo encargado de gestionar los eventos y las comunicaciones.

**MVP** *m* Model View Presenter, es un patrón arquitectónico de interfaz de usuario diseñada para facilitar pruebas de unidad automatizada y mejorar la separación de inquietudes en lógica de presentación.

**MVVM** *m* Model View ViewModel, es un patrón de arquitectura de software. Se caracteriza por tratar de desacoplar lo máximo posible la interfaz de usuario de la lógica de la aplicación.



## 6. Bibliografía

### 6.1 Recursos Web

1. **Asociación de editores de Madrid (2020)**. La FGEE presenta el barómetro de hábitos de lectura y compra de libros en España 2019. [Consulta: 22 Septiembre 2020 ] <<https://www.editoresmadrid.org/la-fgee-presenta-el-barometro-de-habitos-de-lectura-y-compra-de-libros-en-espana-2019/>>
2. **La Vanguardia (2012)**. Los beneficios de la lectura. [Consulta: 22 de Septiembre de 2020] < <https://www.lavanguardia.com/estilos-de-vida/20120613/54312096470/los-beneficios-de-la-lectura.html>>
3. **Applesfera (2010)**. Apple registra el "there's an app for that" de sus anuncios. [Consulta: 25 de Septiembre de 2020] < <https://www.applesfera.com/apple/apple-registra-el-theres-an-app-for-that-de-sus-anuncios>>
4. **Mi Biblioteca (2020)**. Aplicación biblioteca personal de búsqueda rápida [Consulta: múltiples días Septiembre de 2020] <[https://play.google.com/store/apps/details?id=com.vgm.mylibrary&hl=es\\_419](https://play.google.com/store/apps/details?id=com.vgm.mylibrary&hl=es_419)>
5. **Goodreads (2020)**. Aplicación para organizar libros y recibir recomendaciones. [Consulta: múltiples días septiembre de 2020] < [https://play.google.com/store/apps/details?id=com.goodreads&hl=es\\_419](https://play.google.com/store/apps/details?id=com.goodreads&hl=es_419)>
6. **Statcounter (GlobalStats 2020)** Cuota de mercado de sistemas operativos móviles y tabletas en España - septiembre 2020 [Consulta: 29 de septiembre de 2020] <<https://gs.statcounter.com/os-market-share/mobile-tablet/spain#monthly-202008-202009-bar>>
7. **Genbeta (2020)** Lenguajes de programación más queridos y odiados [Consulta: 02 de Octubre de 2020]< <https://www.genbeta.com/desarrollo/65-000-desarrolladores-responden-lenguajes-programacion-queridos-odiados-que-paga-mejor-2020> >
8. **Statcounter (GlobalStats 2020)** Cuota de mercado de la versión Android para móviles y tabletas en España - septiembre 2020 [Consulta: 30 de septiembre de 2020] <<https://gs.statcounter.com/android-version-market-share/mobile-tablet/spain#monthly-202008-202009-bar>>
9. **Open International (2019)** Enfoque híbrido: la mezcla perfecta entre lo tradicional y lo ágil [Consulta: 03 de octubre de 2020] < [https://www.openintl.com/es/enfoque-hibrido-la-mezcla-entre-implementacion-tradicional-y-agil/#\\_ftn2](https://www.openintl.com/es/enfoque-hibrido-la-mezcla-entre-implementacion-tradicional-y-agil/#_ftn2) >
10. **Android4All (2018)** Las 10 claves que hacen que cualquier persona pueda usar una app (aunque no tenga ni idea) [Consulta: 15 de octubre de 2020] <<https://andro4all.com/2018/07/10-reglas-jakob-nielsen-usabilidad>>
11. **Medium (2019)** Patrones Arquitectónicos en Android [Consulta: 21 de octubre de 2020] < <https://medium.com/@vespasoft/patrones-arquitect%C3%B3nicos-en-android-ded39f7a2c10>>
12. **Simform (2019)** MVC vs MVP vs MVVM for Android Application Development [Consulta: 23 de octubre de 2020] < <https://www.simform.com/mvc-mvp-mvvm-android-app-development/>>
13. **Adictos al trabajo (2019)** Pruebas de instrumentación en Android con Espresso [Consulta: 6 de Diciembre de 2020] < <https://www.adictosaltrabajo.com/2019/08/12/pruebas-de-instrumentacion-en-android-con-espresso/> >

**Android Studio. Documentación oficial y guías de desarrollo.** [Consulta: múltiples días desde septiembre a Diciembre 2020] <<https://developer.android.com/studio>>

**Firebase. Documentación oficial y guías de desarrollo.** [Consulta: múltiples días desde septiembre a Diciembre 2020] <<https://firebase.google.com/docs?hl=es>>

## 6.2 Recursos de texto

### **Métodos para el desarrollo de aplicaciones móviles**

Autores: Robert Ramírez Vique, Helena Boltà Torrell  
Universitat Oberta de Catalunya

### **Conceptualización**

Autores: Jordi Flamarich Zampalo  
Universitat Oberta de Catalunya

### **Estrategias de desarrollo**


Autores: Jordi Rovira Piñol  
Universitat Oberta de Catalunya

**Android Apprentice: Beginning Programming with Kotlin**, Raywonderlich Tutorial Team: Irina Galata, Joe Howard & Ellen Shapiro (2019). Second Edition.

**Kotlin Apprentice: Beginning Android Development with Kotlin**, Raywonderlich Tutorial Team: Darryl Bayliss, Tom Blankenship, Fuad Kamal & Namrata Bandekar (2019). Second Edition.

## 7. Anexos

### 7.1 Formulario de investigación sobre la lectura



Formulario investigación sobre lectura

\*Obligatorio

Sexo \*

Hombre

Mujer

Edad \*

De 15 y 24 años

De 25 y 34 años

De 35 y 44 años

De 45 a 54 años

De 55 a 64 años

Más de 65 años

Estado civil \*

Soltero/a independiente

Soltero/a viviendo con sus padres

Casado/a o viviendo en pareja

Hijos \*

Sí, menores de 18 años

Sí, mayores de 18 años

No

Hijos \*

- Sí, menores de 18 años
- Sí, mayores de 18 años
- No

Estudios \*

- Estudios básicos
- Estudios Secundarios
- Estudios universitarios
- Otros

Profesión \*

- En paro
- Trabajador por cuenta propia
- Trabajador por cuenta ajena
- Estudiante

Lugar de residencia \*

- Ciudad
- Pueblo
- Campo
- Otro

Aficiones \*

- Cine
- Juegos
- Lectura
- Deporte
- Música
- Otras

Selecciones los dispositivos móviles que utiliza \*

- Móvil
- Tableta
- Ambos
- Ninguno

¿Para que los suele utilizar? \*

- Sólo para llamadas
- Leer
- Redes Sociales
- Otras aplicaciones

Libros que lee al año \*

- Ninguno
- Entre 1 y 5
- De 5 a 10
- Más de 10

Frecuencia con la que suele leer

- Todos los días
- Fines de semana
- Vacaciones
- Muy esporádicamente

Motivo de no leer más a menudo \*

- No me gusta leer
- No tengo tiempo
- No tengo suficiente motivación
- Otra

Canal de adquisición de libros \*

- Biblioteca
- Compra en librería
- Comprar por internet
- Descarga gratuita

¿Qué formato utiliza para leer? \*

- Papel
- Libro electrónico
- Móvil o tablet

¿Qué horario de lectura suele tener? \*

- Por la mañana
- Por la tarde
- Por la noche

Dónde suele leer \*

- En la cama
- En el sofá
- En transporte público
- Otros

¿Utiliza alguna aplicación para organizar sus lecturas? \*

- Sí
- No

Dónde suele leer \*

- En la cama
- En el sofá
- En transporte público
- Otros

¿Utiliza alguna aplicación para organizar sus lecturas? \*

- Sí
- No

¿Cuál es el motivo de no utilizarla? \*

- Las he probado y no me acaban de encajar
- No me aporta nada
- No conozco ninguna

Recuerdas los libros que has leído y citas o frase que te cautivaron? \*

- Sí
- No
- Sólo algunos

¿Usaría una aplicación donde organizar lecturas y almacenar citas importantes que te marcaron en tu vida? \*

- Sí
- No
- Tal vez

Enviar

## 7.2 Guion de entrevista sobre la lectura

<b>GUION DE LA ENTREVISTA</b>
<b>PERSONAL</b>
¿Qué edad tienes?
¿Estás casado/a?
¿Tienes hijos?
¿Dónde resides actualmente?
¿Qué nivel de estudios posees?
¿Cuáles son tus principales aficiones?
<b>RELACION CON LA TECNOLOGIA</b>
¿Usas móvil y/o tablet?
¿Cuáles horas al día sueles dedicarle?
¿Cuáles son las aplicaciones que más utilizas?
Si tuvieras que decirme una única cosa que te gusta cuando utilizas una aplicación en tu teléfono ¿cuál sería?
<b>RELACION CON LA LECTURA</b>
¿Te gusta leer?
¿Qué es lo que te gusta o te motiva de leer libros?
¿Cuántos libros has leído este año?
¿Cuál es tu libro favorito?
¿Podrías decirme alguna frase o cita que te gustó y recuerdes de ese libro?
¿Te sueles acordar de todos los libros que has leído a lo largo de tu vida?
¿Sueles apuntar citas o frases que te hayan gustado de un libro?
¿Sueles apuntar libros que tienes pensado leerte en un futuro?
¿De dónde sacas la idea de leer tu siguiente libro?
¿Dónde sueles conseguirlo, biblioteca, internet, librería?
¿Cómo tienes organizados los libros físicos que tienes en casa?
¿Conoces alguna aplicación para organizar tus libros de manera virtual?
¿Ves interesante una aplicación en la que queden organizados los libros a lo largo de tu vida?