

# RooMate: aplicación móvil para compartir piso para estudiantes.

**Álvaro de Porres González**

Máster universitario de Desarrollo de aplicaciones para dispositivos móviles

**Consultor: Pau Dominkovics Coll**

**Nombre Profesor/a responsable de la asignatura: Carles Garrigues Olivella**

27/12/2021



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>RooMate: aplicación móvil para compartir piso para estudiantes</i>
<b>Nombre del autor:</b>	<i>Álvaro de Porres González</i>
<b>Nombre del consultor/a:</b>	<i>Pau Dominkovics Coll</i>
<b>Nombre del PRA:</b>	<i>Carles Garrigues Olivella</i>
<b>Fecha de entrega:</b>	12/2021
<b>Titulación:</b>	<i>Máster universitario de Desarrollo de aplicaciones para dispositivos móviles</i>
<b>Área del Trabajo Final:</b>	<i>Desarrollo de aplicaciones para dispositivos móviles</i>
<b>Idioma del trabajo:</b>	<i>Castellano</i>
<b>Palabras clave</b>	<i>Compartir piso Estudiantes Flutter</i>
<b>Resumen del Trabajo:</b>	
<p>Actualmente, existen multitud de aplicaciones y páginas web que ofrecen un sistema para localizar alojamientos para comprar, alquilar o compartir. Sin embargo, no hay ninguna exclusivamente centrada en estudiantes.</p> <p>En la presente memoria, se describe el proyecto realizado, desde la conceptualización hasta la implementación pasando por el diseño de la aplicación RooMate.</p> <p>RooMate, es una aplicación, que permite publicar anuncios, tanto a los propietarios de los pisos como a los propios inquilinos, exclusivamente dirigidos a estudiantes que buscan un alojamiento para poder acudir a la universidad o su centro de estudios. La diferenciación de esta app del resto de aplicaciones del mercado reside en ciertas funcionalidades como son: información sobre la distancia al centro educativo más cercano, posibilidad de añadir a los inquilinos del piso al anuncio para que el usuario que busque alojamiento sepa con qué tipo de personas convivirá, filtros donde también se puede filtrar por aspectos de los inquilinos (idiomas, estilos de vida, deportes que practican...).</p> <p>La aplicación ha sido desarrollada con Flutter como cliente móvil y con Firebase como backend.</p> <p>El resultado del proyecto es muy positivo, si bien es verdad que ciertas funcionalidades descritas en la toma de requisitos no se han podido llevar a cabo debido a una mala planificación, a unos objetivos algo ambiciosos y al tiempo consumido en la búsqueda de información y documentación en la etapa del</p>	

desarrollo

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

Currently, there are a multitude of applications and websites that offer a system for locating accommodation to buy, rent or share. However, there are none exclusively focused on students.

This report describes the project, from conceptualization to implementation, including the design of the Roomate application.

Roomate is an application that allows to publish ads, both to the owners of the apartments and to the tenants themselves, exclusively aimed at students looking for accommodation to go to the university or their school. The differentiation of this app from other applications on the market lies in certain features such as: information on the distance to the nearest educational center, the possibility of adding the tenants of the apartment to the ad so that the user looking for accommodation knows what kind of people will live with, filters where you can also filter by aspects of the tenants (languages, lifestyles, sports they practice ...).

The application has been developed with Flutter as mobile client and Firebase as backend.

The result of the project is very positive, although it is true that certain features described in the requirements gathering could not be carried out due to poor planning, somewhat ambitious objectives and the time consumed in the search for information and documentation in the development stage.

# Índice

<b>1. Introducción .....</b>	<b>1</b>
<b>1.1. Contexto y justificación del Trabajo .....</b>	<b>1</b>
<b>1.2. Análisis del mercado .....</b>	<b>1</b>
1.2.1. Milanuncios.....	2
1.2.2. Roomster.....	2
1.2.3. Badi .....	3
<b>1.3. Objetivos del Trabajo .....</b>	<b>4</b>
1.3.1. Objetivos de la aplicación .....	4
1.3.2. Requisitos funcionales .....	4
1.3.3. Requisitos no funcionales .....	5
<b>1.4. Enfoque y método seguido .....</b>	<b>5</b>
<b>1.5. Planificación del Trabajo .....</b>	<b>6</b>
<b>1.6. Metodología aplicada.....</b>	<b>6</b>
<b>1.7. Breve resumen de productos obtenidos.....</b>	<b>9</b>
<b>1.8. Breve descripción de los otros capítulos de la memoria.....</b>	<b>9</b>
<b>2. Diseño .....</b>	<b>10</b>
<b>2.1. Diseño centrado en el usuario (DCU) .....</b>	<b>10</b>
2.1.1. Análisis .....	10
2.1.2. Usuarios y contextos de uso .....	15
2.1.3. Diseño conceptual.....	17
2.1.4. Prototipado .....	18
2.1.5. Evaluación .....	30
<b>2.2. Diseño técnico .....</b>	<b>31</b>
2.2.1. Definición de los casos de uso .....	31
2.2.2. Diseño de la arquitectura.....	36
<b>2.3. Revisión de la planificación .....</b>	<b>38</b>
<b>3. Implementación.....</b>	<b>39</b>
<b>3.1. Herramientas para el desarrollo .....</b>	<b>39</b>
3.1.1. Flutter .....	39
3.1.2. Visual Studio Code .....	39
3.1.3. Firebase.....	39
3.1.4. Android Studio .....	40
3.1.5. Google Cloud Platform (GCP).....	40
<b>3.2. Librerías.....</b>	<b>40</b>
<b>3.3. Patrones .....</b>	<b>41</b>
<b>3.4. Pruebas .....</b>	<b>41</b>
<b>3.5. Revisión de la planificación .....</b>	<b>45</b>
<b>4. Conclusiones.....</b>	<b>46</b>
<b>5. Glosario.....</b>	<b>48</b>

<b>6. Bibliografía.....</b>	<b>49</b>
<b>7. Anexos .....</b>	<b>50</b>
<b>7.1. Manual de usuario .....</b>	<b>50</b>

## Lista de figuras

<i>Figura 1. Captura de Milanuncios</i>	2
<i>Figura 2. Captura de Roomster</i>	3
<i>Figura 3. Captura de Badi</i>	3
<i>Figura 4. Diagrama de Gantt</i>	6
<i>Figura 5. Ciclo de vida iterativo e incremental</i>	7
<i>Figura 6. Ciclo de vida ágil.</i>	7
<i>Figura 7. Tablero Scrumban</i>	8
<i>Figura 8. Diagrama de navegación</i>	17
<i>Figura 9. Gráficos de mejora Redbooth</i>	18
<i>Figura 10. Bocetos búsqueda por dirección y por mapa</i>	19
<i>Figura 11. Bocetos de resultado de búsqueda y crear anuncio</i>	19
<i>Figura 12. Bocetos de detalle de anuncio y filtros</i>	20
<i>Figura 13. Bocetos del chat</i>	20
<i>Figura 14. Bocetos crear/editar/visualizar perfil</i>	20
<i>Figura 15. Boceto visualizar piso</i>	21

# 1. Introducción

A continuación, se hace una introducción al proyecto, explicando la motivación de este, los objetivos que se buscan y la planificación que se ha seguido.

## 1.1. Contexto y justificación del Trabajo

Según la Estadística de Estudiantes Universitarios (EEU) del curso 2020-2021<sup>1</sup> publicada por el Ministerio de Universidades, un 30% de los estudiantes de Madrid proceden de una comunidad de residencia distinta, lo que implica que, al menos el 30% de los estudiantes matriculados necesitan adquirir una vivienda en la que residir durante sus estudios, y los pisos compartidos, son una de las opciones principales, debido a que suele tener un precio y unos gastos más reducidos que un alquiler individual. Y como pasa en Madrid, pasa en otras muchas comunidades autónomas.

Existen multitud de aplicaciones para alquilar o compartir piso, como veremos en el siguiente punto, pero no hay una aplicación móvil específica para compartir pisos exclusivamente para estudiantes.

Debido a esta carencia en las tiendas de aplicaciones y en la web, la cantidad de mensajes que se ven en redes sociales preguntando acerca de pisos de estudiantes cerca de la universidad o preguntando si algún estudiante estuviera interesado en ocupar una habitación en un piso es relevante.

Al haber una cantidad importante de webs y aplicaciones especializadas en compartir piso o habitación, lo que se busca con este proyecto es enfocarse principalmente en los estudiantes, haciendo que completen un perfil con sus gustos, hobbies y/o hábitos, etc., para que al alquilar una habitación tengan una idea aproximada de con quién van a convivir teniendo la certeza de que van a vivir con otros estudiantes, lo cual es un requisito que muchos buscan.

Esta aplicación está pensada para ser independiente y encontrarse en las principales tiendas de aplicaciones móviles de forma gratuita, pero también se podría valorar y negociar su incorporación a las páginas web de las propias universidades como trabajo futuro.

## 1.2. Análisis del mercado

Para analizar el mercado, se han buscado aplicaciones con un objetivo parecido, en este caso compartir piso o alquiler de habitaciones, en Google Play,

---

<sup>1</sup>

<https://www.universidades.gob.es/stfls/universidades/Estadisticas/ficheros/PpalesResulEEU.pdf>



la tienda de aplicaciones de Android para comprobar las virtudes y defectos de los competidores.

### 1.2.1. Milanuncios<sup>2</sup>

Es una aplicación que funciona como un tablón de anuncios en los que cabe cualquier cosa. Una de ellas son el alquiler de habitaciones, pero es muy opaco respecto a las otras opciones que veremos más adelante: no se puede saber los gustos de tus compañeros, personalidad, intereses, etc. Mucho menos está orientado a gente joven ni universitaria y los filtros de búsqueda son poco precisos.

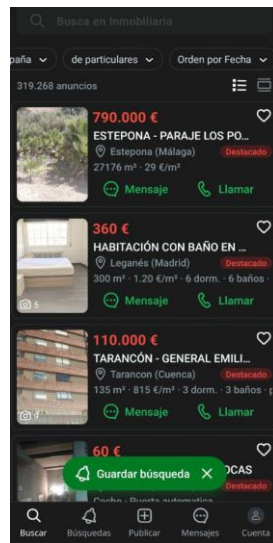


Figura 1. Captura de Milanuncios

### 1.2.2. Roomster<sup>3</sup>

Es una aplicación bastante extendida con presencia en casi 200 países. Con ella se puede definir un estilo de vida al anunciar una habitación y tiene una opción para indicar si es solo para estudiantes. El error que tiene es que no se puede buscar solo por ese tipo de alojamiento y no está del todo orientado a estudiantes.

<sup>2</sup> <https://play.google.com/store/apps/details?id=com.muba.anuncios&gl=ES>

<sup>3</sup> <https://play.google.com/store/apps/details?id=com.roomster&gl=ES>

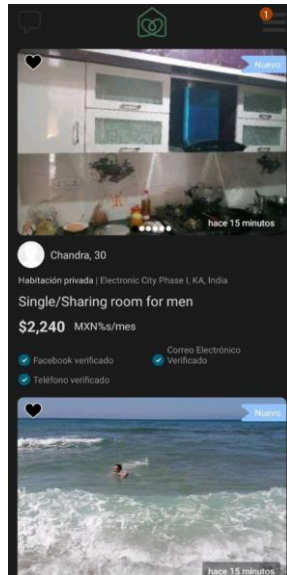


Figura 2. Captura de Roomster

### 1.2.3. Badi<sup>4</sup>

Badi es una aplicación muy bien desarrollada, muy parecida a Roomster, capaz de mostrar habitaciones disponibles, pero con pocas opciones de filtros de búsqueda, y capaz de crear anuncios de habitaciones. También tiene la opción de poner los gustos personales y lo hábitos, pero igualmente, no está orientada a los estudiantes y no pudiendo seleccionar pisos de solo estudiantes.

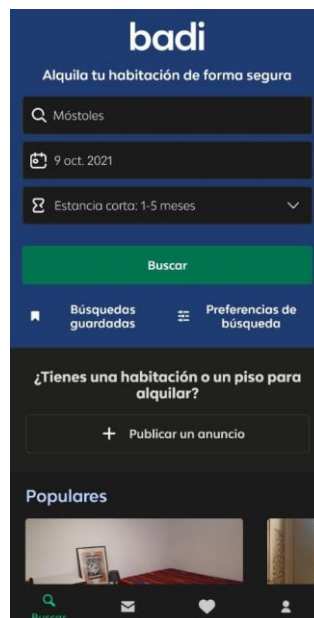


Figura 3. Captura de Badi

<sup>4</sup> <https://play.google.com/store/apps/details?id=es.inmovens.badi&gl=ES>

En general, estas aplicaciones tratan de abarcar todo el mercado y todas las edades. El objetivo de este proyecto es crear una aplicación de referencia para los estudiantes en la que solo encuentren pisos con compañeros estudiantes y más orientada a su estilo de vida y necesidades, como, por ejemplo, la distancia a la universidad más cercana, no encontrar pisos que no sean de estudiantes e impidan una búsqueda clara y concisa o más información sobre sus futuros compañeros, mostrando en el anuncio del alojamiento los gustos, estilos de vida, idiomas... de los inquilinos del alojamiento.

### 1.3. Objetivos del Trabajo

El objetivo principal de este proyecto es proporcionar una herramienta a los estudiantes, principalmente universitarios, para poder compartir piso con otros estudiantes enfocando la aplicación a sus gustos y necesidades.

#### 1.3.1. Objetivos de la aplicación

Se han definido los siguientes objetivos para la aplicación:

- O1: Registro e inicio de sesión del usuario.
- O2: Modificar perfil.
- O3: Geolocalizar al usuario y los pisos/habitaciones.
- O4: Publicación de anuncios de habitaciones.
- O5: Visualización de anuncios.
- O6: Chat integrado.

#### 1.3.2. Requisitos funcionales

A continuación, se muestran los requisitos funcionales de la aplicación:

Requisito Funcional	Objetivo
<b>RF1. Gestión de usuarios</b>	
RF1.1 – Registrar nueva cuenta de usuario	O1
RF1.2 – Inicio de sesión con usuario existente	
RF1.3 – Cerrar sesión	
RF1.4 – Modificar perfil del usuario	O2
RF1.5 – Eliminar cuenta	
<b>RF2. Geolocalización</b>	
RF2.1 – Gestionar permisos de ubicación	O3
RF2.2 – Buscar habitaciones/pisos por ubicación	
<b>RF3. Gestión de anuncios</b>	
RF3.1 – Crear un anuncio propio	O4
RF3.2 – Modificación de anuncio propios	
RF3.3 – Eliminación de anuncios propios	
RF3.4 – Visualizar los anuncios propios en el perfil del usuario	O5
RF3.5 – Visualizar los anuncios por filtros	

RF4. Chat integrado	
RF4.1 – Chat integrado entre el anunciante y el interesado.	O6

Tabla 1. Requisitos

### 1.3.3 Requisitos no funcionales

A continuación, se definen aquellos requisitos que no describen información a guardar, ni funciones a realizar, sino características de funcionamiento:

- *Estética:*
  - La interfaz tiene que ser intuitiva y amigable
- *Usabilidad:*
  - Para un usuario estándar, el aprendizaje de una funcionalidad no puede suponer más de 10 minutos.
  - La aplicación deberá tener una estructura clara, abarcando todas las funcionalidades disponibles
- *Rendimiento:*
  - Cualquier funcionalidad con la que interactúe el usuario no puede tardar en responder más de 5 segundos.
  - La aplicación debe ejecutarse con fluidez en cualquier dispositivo Android con versión 8.0 o superior.
- *Estabilidad:*
  - El sistema no puede dejar de funcionar de manera repentina por un error.
- *Seguridad:*
  - La aplicación debe de mantener los datos almacenados seguros y protegidos.
- *Dispositivos a los que se dirige:*
  - La aplicación está pensada para los sistemas operativos iOS y Android, aunque en un futuro se puede adaptar a Web. En este proyecto se centra en el desarrollo para Android debido al tiempo limitado que se tiene, pero en un futuro se espera poder publicarlo también para iOS

## 1.4 Enfoque y método seguido

Se ha optado por la realización de un proyecto desde cero. Esta decisión vino propiciada mayoritariamente por mi interés en formarme como desarrollador de aplicaciones móviles, concretamente con Flutter, y qué mejor manera que creando un proyecto nuevo y pasar por todas las etapas que ello conlleva.

Sobre la idea de la aplicación, surgió hace tiempo cuando día sí y día también, observaba en redes sociales (Twitter principalmente) cómo estudiantes no cesaban de buscar compañeros o buscar un piso de estudiantes para vivir durante el curso académico. Es por ello por lo que investigué y observé que no hay una aplicación especializada y que podría ser de ayuda para todos esos estudiantes que buscan compañeros de piso.

## 1.5 Planificación del Trabajo

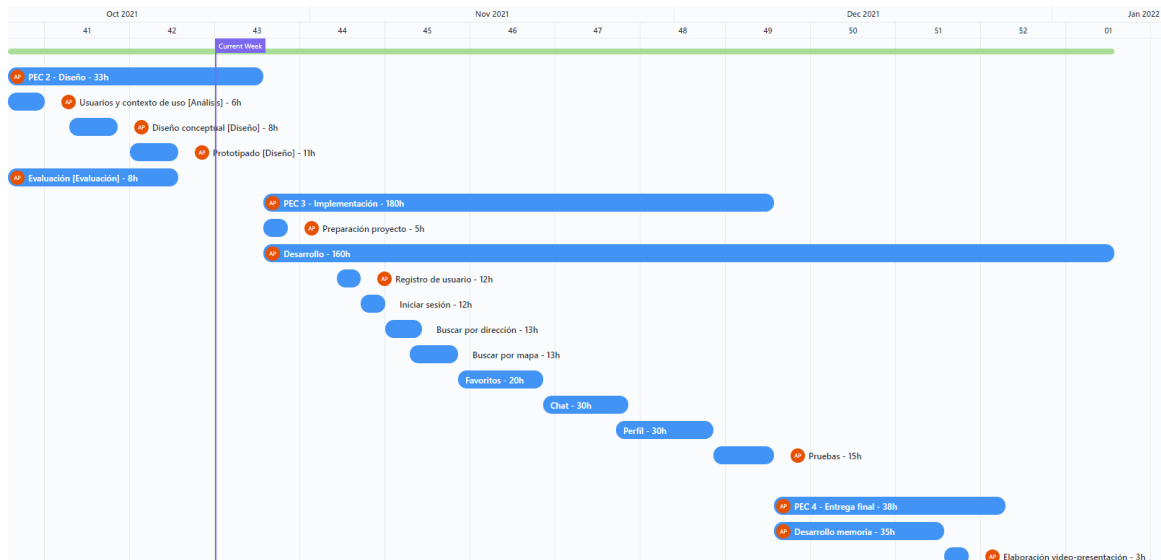


Figura 4. Diagrama de Gantt

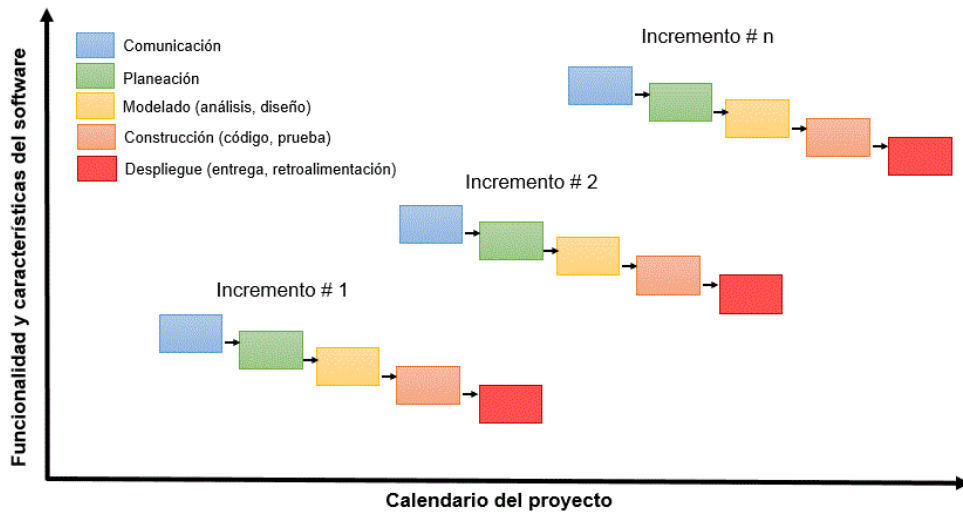
- La dedicación para este proyecto consiste en 2 horas por día laborable y 5 horas por día los fines de semana, aproximadamente.
- Cuando dos tareas coinciden en el tiempo, se repartirán las horas del día, pero dando prioridad a aquella que tenga más horas estimadas.

## 1.6 Metodología aplicada

En el mundo del desarrollo de Software existen muchas posibilidades en lo que a metodología de gestión de proyectos se refiere. Todas estas metodologías se pueden dividir en dos grandes grupos:

- Metodologías tradicionales: Las metodologías tradicionales se deben usar principalmente para aquellos proyectos en los que los requisitos y los resultados se establecen al inicio del proyecto. En este tipo de metodologías los cambios en los requisitos no son bienvenidos, pues conllevaría un coste muy alto tanto económico como temporal. Estas metodologías se caracterizan por realizar una serie de etapas una única vez de forma lineal e independientes entre ellas. Dentro de estas metodologías podemos encontrar ciclos de vida tales como el ciclo de vida línea o cascada o el ciclo de vida en espiral.
- Metodologías ágiles: La gestión ágil de proyectos está diseñada para responder al cambio y entregar el trabajo en ráfagas cortas o “sprints”. En cada entrega o “sprint” se tiene un producto final utilizable. Ejemplos de ciclos de vida dentro de estas metodologías tenemos:
  - **Ciclo de vida iterativo e incremental:** A diferencia del ciclo de vida en cascada, donde las fases del ciclo de vida (requisitos, análisis, diseño, etc.) se realizan una única vez, y el inicio de una fase no comienza hasta que termina la fase que le precede, en un ciclo de vida iterativo e incremental se va liberando partes del

producto (prototipos) periódicamente, en cada iteración, y cada nueva versión, normalmente, aumenta la funcionalidad y mejora en calidad respecto a la anterior. Ver *Figura 5*.



*Figura 5. Ciclo de vida iterativo e incremental*

- **Ciclo de vida ágil:** Podemos definir a los ciclos de vida ágiles como un ciclo de vida iterativo e incremental, con iteraciones cortas (semanas) y sin que dentro de cada iteración tenga por qué haber fases lineales como ocurre en el ciclo de vida en cascada. Ver *Figura 6*.



*Figura 6. Ciclo de vida ágil.*

En este proyecto, se ha optado por usar una Metodología Ágil, debido a que el concepto básico de la aplicación y sus requisitos funcionales básicos están claros desde el principio, pero se prevé que vayan a surgir nuevas funcionalidades y mejoras en la aplicación. Este tipo de metodología permitiría tener un producto tangible desde las primeras semanas y añadir nuevas funcionalidades rápidamente sobre un prototipo funcional.

Hay muchos tipos de metodologías ágiles en el desarrollo de proyectos tales como Scrum, Agile Inception, Kanban o Extreme Programming XP.

En este trabajo de fin de máster, lo que se presupone óptimo es realizarlo siguiendo la metodología Scrum apoyándonos en la organización que aporta Kanban.

Por una parte, en Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales. El proyecto se realiza en “sprints” o ciclos temporales cortos y de duración fija de aproximadamente 2 semanas y en cada iteración o “sprint” se realiza una entrega del producto final con algún incremento de funcionalidad y calidad.

Por otra parte, Kanban ayuda en la gestión de las tareas ofreciéndonos un “WIP” o Work in progress, que limita las tareas que se deben realizar al mismo tiempo.

Combinando estas dos metodologías, se obtienen los tableros Scrumban. Estos tableros sirven para tener una gestión del proyecto visual con pegatinas que indica las tareas por hacer, las que se están haciendo y las hechas, con un máximo en cada una de ellas, el WIP.

Al comenzar cada “Sprint”, el Product Owner, o responsable del proyecto, añade a la primera columna las historias de usuario de dicho “Sprint”. Las historias de usuario son parecidas a los requisitos funcionales de las metodologías tradicionales, pero más simples, concisas y con una estructura común. Después, es el equipo el que se autoorganiza para ir moviendo las historias de usuario entre las columnas. Ver *Figura 7*.



*Figura 7. Tablero Scrumban*

Este proyecto se realizará en tres sprints, en los que al final de cada uno de ellos se tendrá un prototipo de la aplicación funcional. En cada sprint se introducirá una mejora de la calidad de la aplicación, así como nuevas funcionalidades.

La duración total del desarrollo de la aplicación será aproximadamente de 160 horas, otorgando a cada sprint una duración aproximada de dos semanas (54 horas aproximadamente por sprint).

## **1.7 Breve resumen de productos obtenidos**

Al terminar este proyecto se obtendrán los siguientes entregables:

- Código fuente de la aplicación.
- Ejecutable de la aplicación para Android, en formato .apk.
- Memoria del trabajo realizado.
- Presentación en vídeo y PowerPoint del trabajo con una demo.

## **1.8 Breve descripción de los otros capítulos de la memoria**

A continuación, se explicarán el resto de los capítulos de la memoria, donde se tratará aspectos como el diseño, la implementación y las conclusiones.

En el capítulo 2 se comentará la fase de diseño de la aplicación. En él se explica el análisis realizado con entrevistas y encuestas, los usuarios objetivo, los contextos de uso de la aplicación, la fase del prototipado inicial y del de alta fidelidad y el diseño técnico, desde la definición de los casos de uso hasta el diseño de la arquitectura del proyecto.

Dentro del capítulo 3 se detalla la fase de la implementación de la aplicación. Se describe las herramientas utilizadas, los patrones y más detalles sobre la implementación. También se comentarán las pruebas realizadas durante dicha fase y cómo se han abordado los resultados de estas.

En el capítulo 4 se exponen las conclusiones del proyecto.



## 2. Diseño

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

#### 2.1.1. Análisis

La primera etapa en el diseño centrado en el usuario es la etapa del análisis. En esta etapa se buscará encontrar las características y necesidades de los posibles futuros usuarios de la aplicación.

Para obtener dicha información, se han utilizado dos métodos de indagación: encuestas y entrevistas personales.

- **Encuestas**

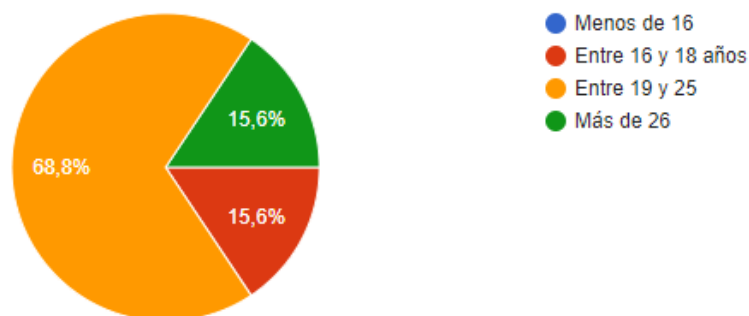
En esta fase, se creó un formulario en Google Forms con preguntas tanto de sexo y de edad, como de aspectos tecnológicos, enviado principalmente a usuarios potenciales en un rango de edad de entre 16 y 30 años, ya que es la edad a la que más gente está estudiando o acaba de estudiar y puede necesitar la aplicación para buscar un piso o una habitación o ha utilizado alguna página web o aplicación y nos puede comentar su perspectiva de aspectos a mejorar.

A continuación, se muestran los resultados de las encuestas realizadas:

- **Datos personales**

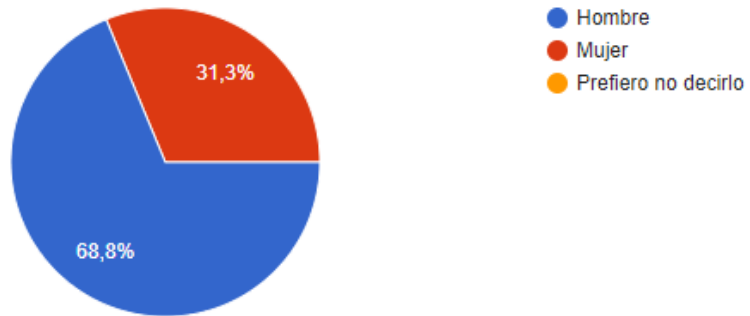
Edad

32 respuestas



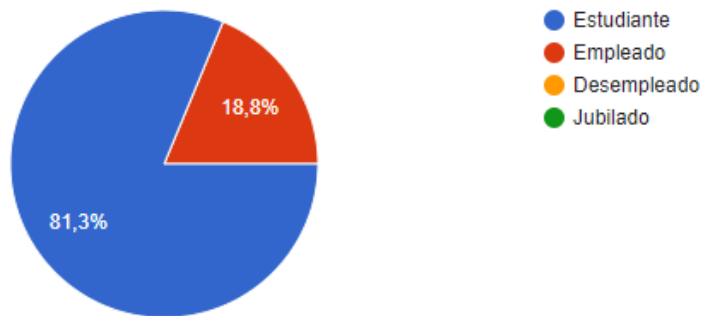
### Sexo

32 respuestas



### Situación laboral actual

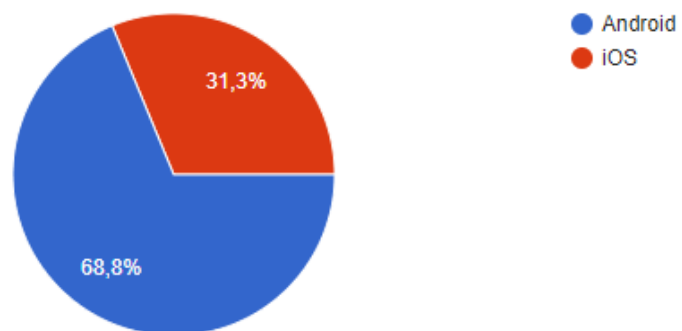
32 respuestas



## ▪ Uso de dispositivos móviles

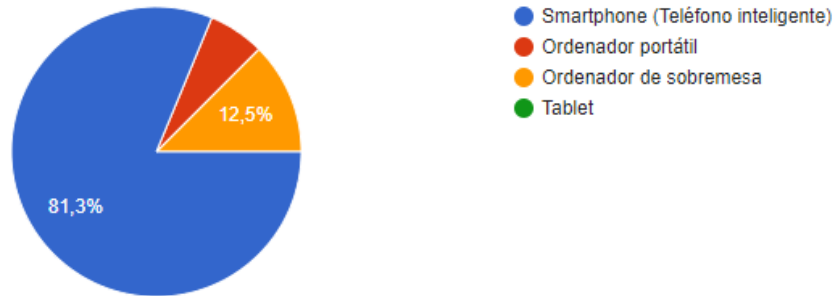
¿Qué sistema operativo móvil utiliza? Si usa varios, indique el que más use

32 respuestas



¿Qué dispositivo utiliza con más frecuencia para comprar online?

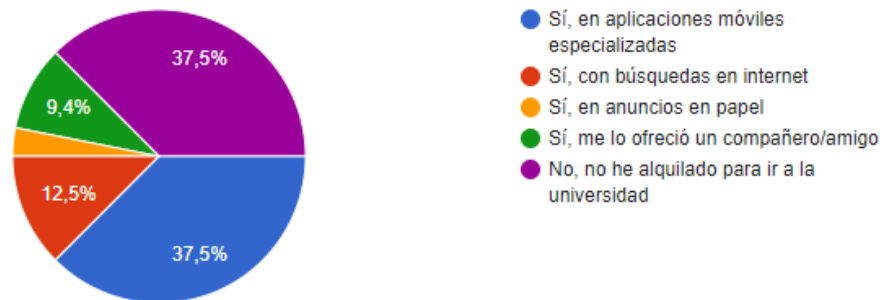
32 respuestas



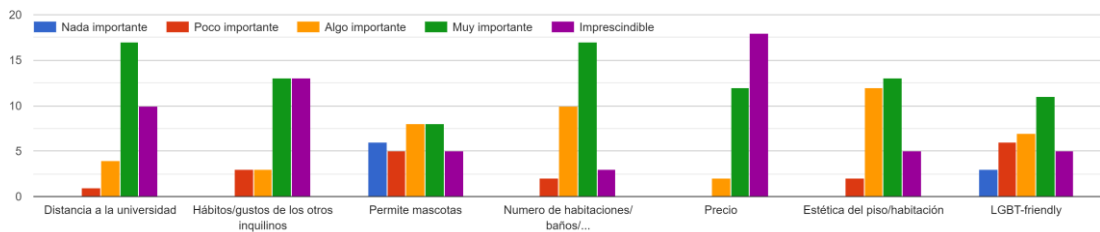
▪ Alquiler y preferencias

¿Has alquilado algún piso o habitación para acudir a la universidad? ¿Dónde lo encontraste?

32 respuestas

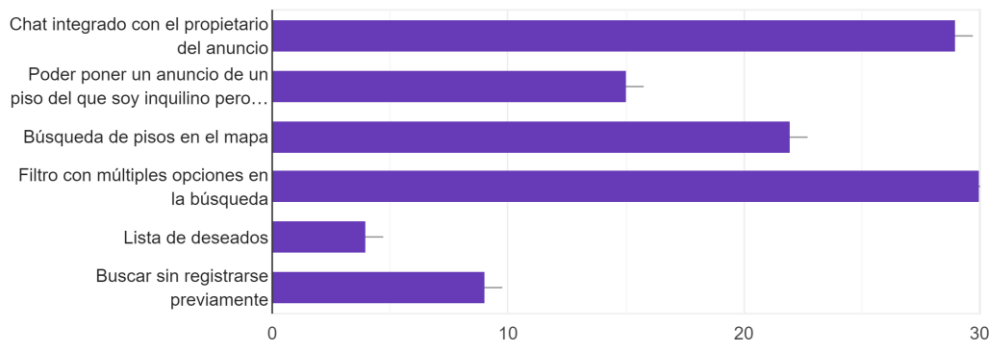


A la hora de buscar piso o habitación para ir a la universidad, ¿qué cualidades le son importantes?



Cuales de las siguientes funcionalidades de una aplicación para alquilar pisos o habitaciones serían importantes para usted

32 respuestas



Con el objetivo de conocer los defectos que pueden tener las demás aplicaciones o páginas webs que se dedican a la compraventa y/o alquiler de pisos y habitaciones, se formuló una pregunta de respuesta corta a los encuestados, en la cual se le pregunta acerca de funcionalidades que echaron de menos o defectos que notaron al usar dichas aplicaciones. Algunas de las respuestas que se repitieron como mínimo dos veces fueron:

- Mejores filtros de búsqueda.
- Conocer los hábitos o gustos de los otros compañeros.
- Encontrar pisos solo de estudiantes es complicado.

Gracias a las entrevistas podemos observar que es una buena estrategia realizar el desarrollo de la aplicación para dispositivos móviles, ya que el 81,3% usa el móvil como su principal dispositivo para realizar compras y el 37,5% de los encuestados que alguna vez ha alquilado habitación o piso lo realizó a través de un teléfono móvil. También podemos observar que el 68,8% de los encuestados utiliza un teléfono Android, por lo que la mejor estrategia es centrarse en el desarrollo para ese sistema operativo, aunque en un futuro se publicará también para iOS, pero debido al escaso tiempo de desarrollo permitido, es mejor centrarse en un sistema operativo.

Como podemos observar, las funciones que no pueden faltar en la aplicación son: chat integrado, filtro con múltiples opciones y buscar pisos en el mapa.

Con relación a las cosas que les parecen importantes a la hora de buscar un piso o habitación, podemos observar que los usuarios consideran importantes casi todas las características, por lo que es importante mostrar las máximas características a los usuarios.

- **Entrevista personal**

Para poder tener más información acerca de las necesidades y gustos de las personas que han alquilado alguna vez un piso o una habitación para ir a la universidad o al instituto, se ha realizado una entrevista personal a una persona que necesito alquilar una habitación para poder estudiar.

El entrevistado es un hombre de 26 años, actualmente vive en Tenerife (Islas Canarias), su lugar de nacimiento y residencia y estudió el Grado en Ingeniería del Software en Madrid, teniendo que alquilar una habitación.

**P:** Cuando viniste a estudiar a Madrid, ¿buscabas un piso completo para ti o una habitación con más compañeros de piso?

**R:** Lo ideal era encontrar un piso de estudiantes con más compañeros porque el alquiler está bastante caro ahí en Madrid y siendo estudiante tampoco me podía permitir vivir solo.

**P:** Y, ¿dónde encontraste el piso en el que finalmente viviste en los años de universidad?

**R:** Pues lo encontré en Idealista, en la parte de compartir que te da la opción de buscar habitación.

**P:** ¿Por qué usaste esa página web? ¿Buscaste en otros sitios?

**R:** Sí, busqué en otros sitios, pero todos los compañeros que habían buscado piso anteriormente me recomendaron Idealista porque es la típica y tiene muchas opciones de pisos.

**P:** ¿Qué fue lo que te llamo la atención de ese piso?

**R:** Pues básicamente lo cerca que estaba a la universidad. Lo que tenía claro es que no quería tener que coger transporte público ni tardar mucho en llegar a la universidad. También vi que el piso ya tenía otros 2 compañeros y el precio no era del todo elevado.

**P:** ¿Echaste algo en falta en la búsqueda de la habitación respecto al motor de búsqueda?

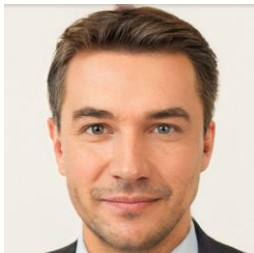
**R:** Bueno la verdad es que no mucha cosa. Si es que es verdad que me habría gustado tener más información de mis compañeros o un chat para hablar con el propietario o con los compañeros, porque solo tenía la opción de llamar por teléfono y a mí no me gusta nada hablar por teléfono.

### 2.1.2. Usuarios y contextos de uso

Los usuarios objetivo de la aplicación serían:

- Usuarios que necesitan una habitación o piso donde vivir para ir a la universidad. La gran parte de usuarios que estarían dentro de este grupo serían jóvenes de entre 17 y 26 años.
- Usuarios que quieren alquilar alguna habitación o piso completo. En este grupo puede haber usuarios que viven dentro del piso y necesita compañero de piso o es el propietario y quiere ganar dinero alquilando habitaciones. En este grupo puede haber gente de todas las edades exceptuando niños.

Respecto a los contextos de uso, al tratarse de una aplicación en la que hay que gastar un poco de tiempo para buscar habitación y comparar, se entiende que el usuario se encontrará en un lugar tranquilo y sentado, previsiblemente en su casa, aunque teniendo en cuenta las características de los teléfonos móviles, que pueden ser usados en cualquier sitio, no está asegurada esta situación. Pasaría lo mismo con los usuarios que ponen a alquilar la habitación o el piso, sería muy probable que lo hicieran desde la casa que quieren poner a alquilar, pues tendrán que hacer las fotos de las habitaciones, medir, etc.

Descripción de la persona	
 <p><b>Nombre:</b> Pablo <b>Edad:</b> 18 <b>Estudios:</b> Bachillerato <b>Profesión:</b> Estudiante <b>Estado civil:</b> Soltero <b>Residencia:</b> Avilés, Asturias</p>	<p>Pablo vive en un pueblo de Asturias con sus padres, en una familia humilde. Ha hecho la prueba de acceso a la universidad y ha elegido una universidad de Santander para hacer Fisioterapia.</p> <p>En verano suele ayudar a sus padres en la lavandería para ir ahorrando poco a poco.</p> <p>Es una persona muy sociable, que le gustan el deporte, sobre todo el fútbol y los videojuegos.</p>

### Descripción de escenario

Es verano, y Pablo está ayudando a sus padres en la lavandería. De repente, su madre le recuerda que es mitad de agosto y que todavía no ha buscado un piso donde vivir para ir a estudiar a la universidad de Santander.

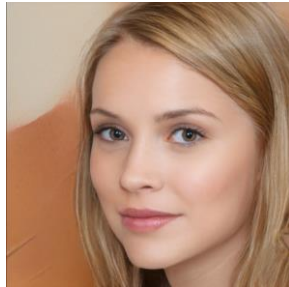
Cuando termina su jornada, Pablo llega a casa y se propone buscar una habitación. Algunos compañeros suyos que están en la misma situación le habían hablado sobre una aplicación para ello. Pablo se la descarga y comienza a buscar piso.

Como Pablo no tiene un gran presupuesto y no tiene coche para moverse, se registra en la aplicación y busca pisos que cumplan los criterios que él quiere: que este cerca de la universidad para ir andando y que sea barato. Para ello hace uso de la búsqueda por mapa.

#### **Necesidades cubiertas:**

- Encontrar una habitación donde vivir
- Habitación cerca de la universidad

### Descripción de la persona



**Nombre:** Noelia  
**Edad:** 20  
**Estudios:** Biología  
**Profesión:** Dependienta  
**Estado civil:** Soltera  
**Residencia:** Madrid

Noelia vive en casa con sus padres y su perro. Actualmente tiene mala relación con sus padres debido a su orientación sexual.

Sus mayores aficiones son las películas, las series y su mascota, que son inseparables.

Actualmente está en 2º de carrera y trabaja a la vez para independizarse cuanto antes.

## Descripción de escenario

Noelia llega a casa como todas las tardes después de la universidad y se encuentra con sus padres. Su relación es mala y como otros muchos días, discuten por su orientación sexual, ya que es lesbiana.

Debido a la mala relación con sus padres y a las continuas discusiones, Noelia decide que es hora de independizarse con lo que ha ido ahorrando, trabajando como dependienta. Decide irse a vivir con otros estudiantes, donde la comprendan y la respeten, y donde pueda ir con su perro.

Noelia se va a su habitación y se descarga una aplicación que había oído hablar por redes sociales para alquilar habitaciones para estudiantes. Se la descarga y se registra. Observa que puede filtrar por pisos que sean LGTB Friendly y que admitan mascotas y que encima, puede ver los gustos y estilos de vida de los otros inquilinos, cosa que le encanta porque ella prefiere una casa tranquila y con gente que le gusten las películas y las series.

### Necesidades cubiertas:

- Encontrar una habitación donde vivir.
- Compañeros con mismos gustos y que aceptan su orientación sexual.

### 2.1.3. Diseño conceptual

Para comenzar a diseñar la aplicación, se ha realizado el siguiente diagrama de navegación, con lo que podemos comenzar a ver las pantallas necesarias y las relaciones entre ellas.

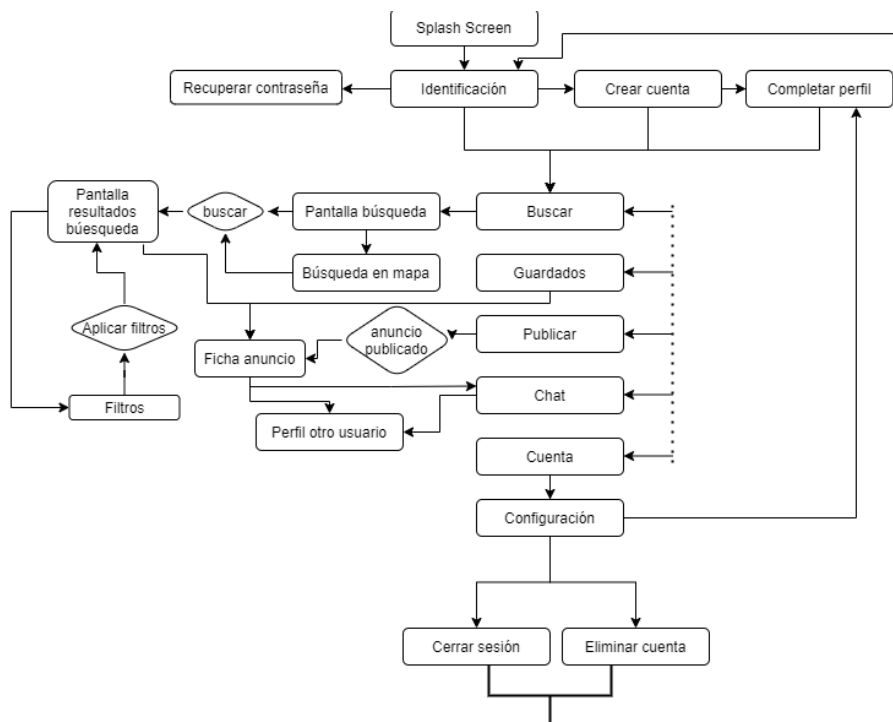


Figura 8. Diagrama de navegación



## 2.1.4. Prototipado

Después de realizar el diagrama de navegación, se procede a realizar el prototipado de la interfaz de la aplicación. Esta etapa se ha realizado en dos pasos: primero se realizaron los bocetos a mano de las principales pantallas para, a continuación, realizar un prototipo de alta fidelidad para ver cómo quedaría la aplicación después del desarrollo, aunque luego siempre se producen cambios, pero con ello se tiene una idea muy aproximada de cómo quedará la aplicación.

A la hora de comenzar a pensar en el diseño, hay que tener en cuenta un aspecto muy importante: la navegación. Para la navegación hay dos grandes opciones: menú hamburguesa o de cajón, o menú fijo inferior.

Para tomar esta decisión se ha buscado información en diferentes fuentes. Por un lado, se ha examinado a la competencia. El resto de las aplicaciones de alquiler de pisos, excepto idealista, tienen un menú fijo inferior, esto es debido a que tienen menos opciones de menú que Idealista, que, sí que necesita un menú más grande y oculto, pues ocuparía mucho en pantalla.

Por otro lado, se han buscado estudios<sup>5</sup> relacionados con esta decisión. Como se puede ver en el siguiente estudio que se realizó con la aplicación de Redbooth, se comprobó los beneficios de cambiar un menú de tipo hamburguesa a uno fijo en la parte inferior. Se mejoraron el número de usuarios activos, aumentó el tiempo de uso de la aplicación y aumentó el número de sesiones.



Figura 9. Gráficos de mejora Redbooth

<sup>5</sup> <https://redbooth.com/blog/hamburger-menu-iphone-app>

Esto se debe a que las funcionalidades de la aplicación siempre están presentes y para el usuario es más fácil encontrar nuevas funcionalidades con el menú fijo abajo. Por el contrario, el menú de hamburguesa directamente ya necesita 1 pulsación extra para acceder a las opciones y está colocado en un sitio que cuesta más acceder: la parte superior izquierda. Actualmente, los teléfonos móviles son cada vez más grandes y cuesta llegar con el pulgar a la parte superior, cuando la mayor parte de las personas utiliza el móvil con una mano. Es por todos estos motivos por los que se ha decidido optar por el menú fijo en la parte inferior.

#### 2.1.4.1. Bocetos a mano

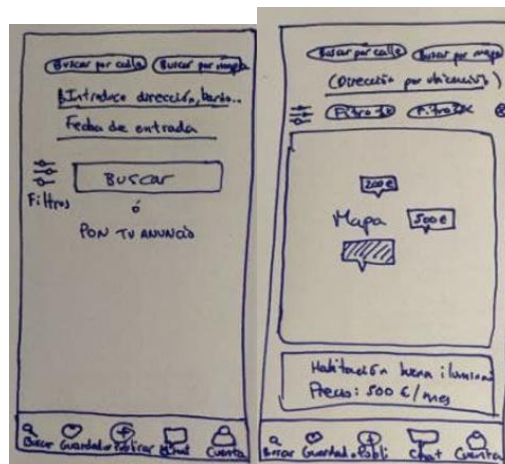


Figura 10. Bocetos búsqueda por dirección y por mapa

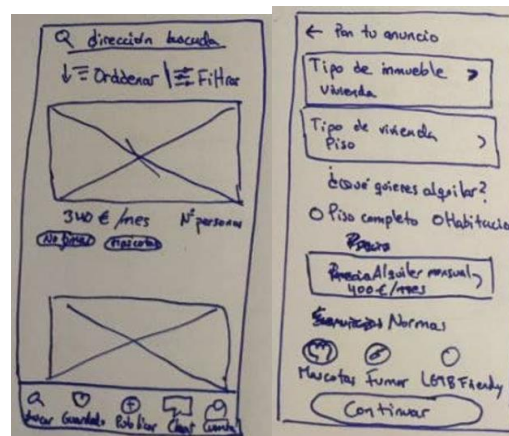


Figura 11. Bocetos de resultado de búsqueda y crear anuncio

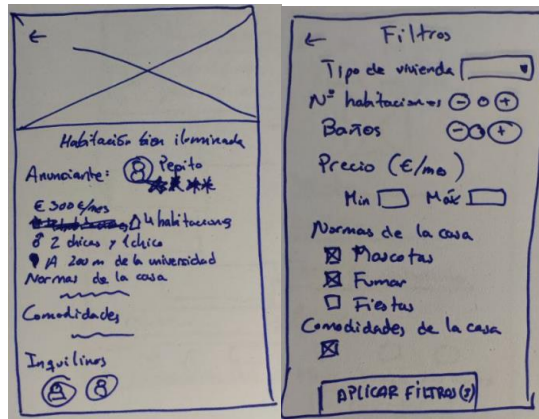


Figura 12. Bocetos de detalle de anuncio y filtros

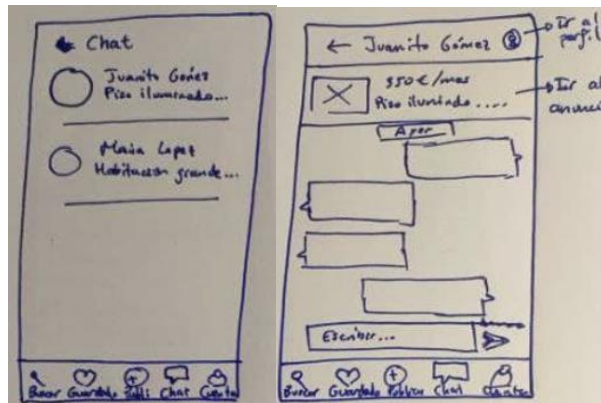


Figura 13. Bocetos del chat

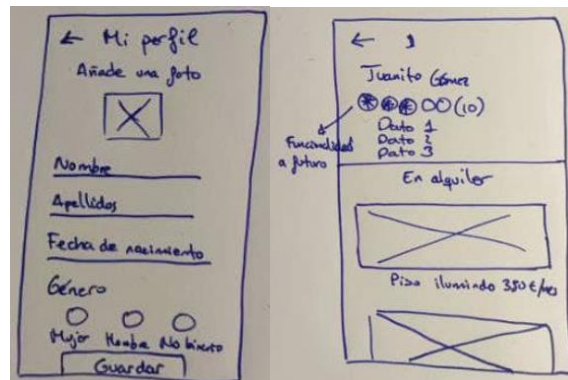


Figura 14. Bocetos crear/editar/visualizar perfil

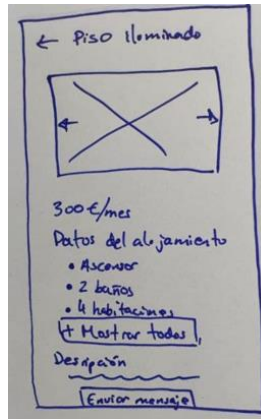


Figura 15. Boceto visualizar piso

#### 2.1.4.2. Prototipo

A continuación, se ha llevado a cabo la realización del prototipo de alta fidelidad mediante el software Axure RP 10, a partir de los bocetos realizados a mano.

Gracias a este prototipo de pantallas, podemos simular cómo será la aplicación en gran medida, aunque luego puedan surgir cambios, pero la interfaz y cómo se relacionan las pantallas en el prototipo es como se espera que esté al terminar el desarrollo.

En el prototipo se ha elegido la fuente Poppins, que aporta unas letras fáciles de leer y un toque redondeado y moderno. Respecto a los colores que identifican la aplicación, se ha elegido el morado. Este color se puede apreciar tanto en botones, formularios, algún texto... Es un color que contrasta bien con el blanco y conjunta bien con el negro.

A continuación, se muestran todas las pantallas realizadas para el prototipo de la aplicación:

## Pantalla de inicio de sesión



The screenshot shows a login screen with the following elements:

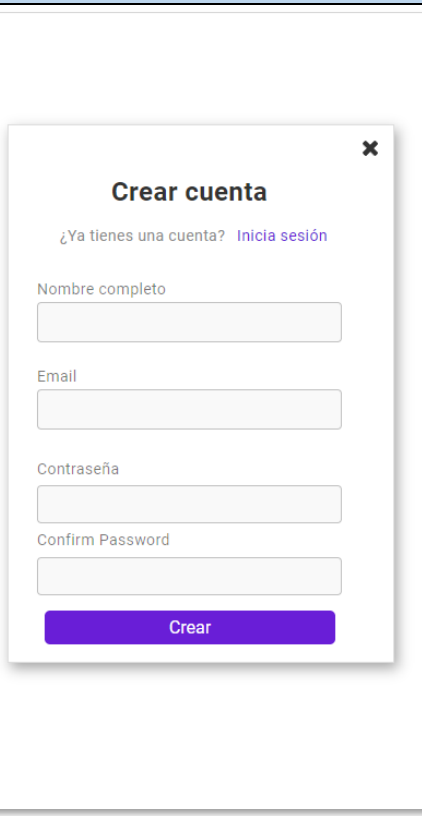
- Iniciar sesión** (Login)
- Link: [¿Eres nuevo? Crea una cuenta](#)
- Form fields: Email, Contraseña (Password)
- Link: [He olvidado mi contraseña](#)
- Button: **Entrar** (Login)
- Social media icons: Google, Facebook, Twitter

Después de la splash screen, esta será la primera pantalla de la aplicación cuando no hayas iniciado sesión anteriormente o sea la primera vez que descargas la aplicación.

Como se explicará más adelante en la evaluación, se observó que era necesario el inicio de sesión con los servicios de Google o Facebook y se añadieron.

En esta pantalla el usuario podrá iniciar sesión, recuperar su contraseña o crear una cuenta nueva.

## Pantalla de creación de cuenta

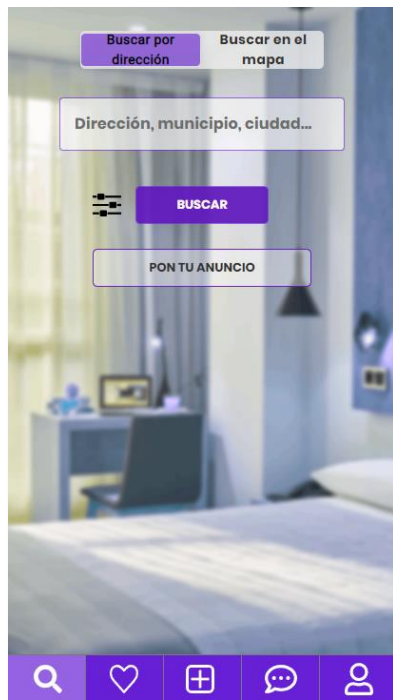


The screenshot shows an account creation screen with the following elements:

- Crear cuenta** (Create account)
- Link: [¿Ya tienes una cuenta? Inicia sesión](#)
- Form fields: Nombre completo (Full name), Email, Contraseña (Password), Confirm Password
- Button: **Crear** (Create)

En esta pantalla el usuario puede crear una nueva cuenta en la aplicación con un nombre completo, usuario y contraseña.

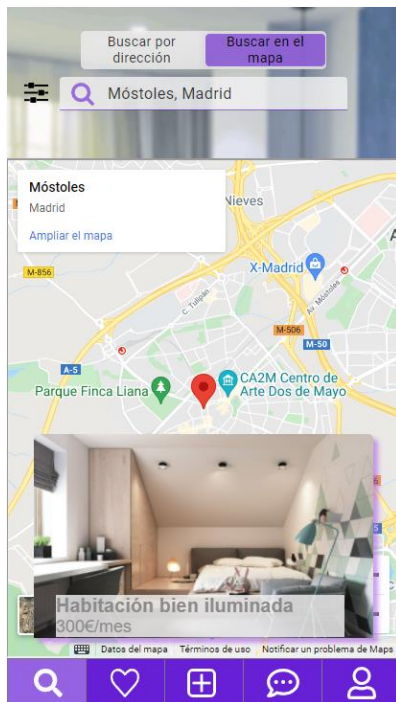
## Pantalla de búsqueda por dirección



Esta es la pantalla que se muestra por defecto tras iniciar sesión en la aplicación. En ella el usuario puede decidir si ingresar una dirección, municipio, ciudad, código postal... para buscar pisos, puede añadir filtros (con el botón de la izquierda del botón buscar), poner un anuncio o cambiar el tipo de búsqueda para buscar en el mapa.

También puede acceder al resto de funciones de la aplicación gracias al menú fijo inferior.

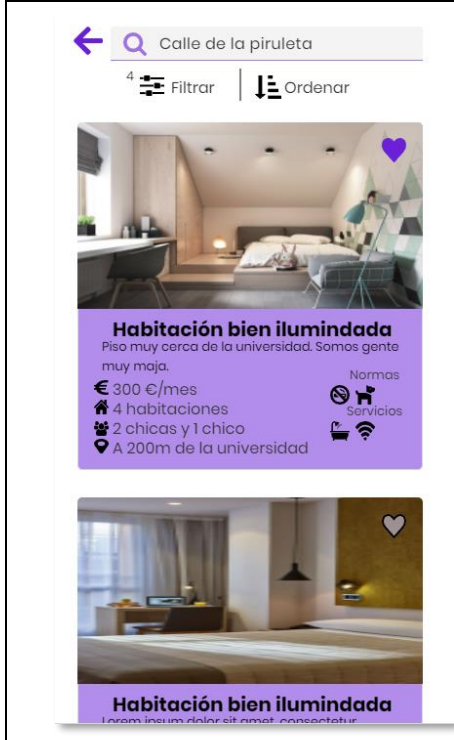
## Pantalla de búsqueda por mapa



En esta pantalla el usuario ha seleccionado "Buscar en el mapa" en la parte superior y aparecerán las casas disponibles en el mapa. Al pinchar sobre una de las casas, aparecerá el anuncio en la parte inferior con el título del anuncio y el precio del alquiler. Si el usuario pulsa sobre el anuncio, accederá a la vista del anuncio, donde saldrán todos los detalles.

El usuario también puede introducir una dirección que se cargará en el mapa y puede hacer uso de los filtros.

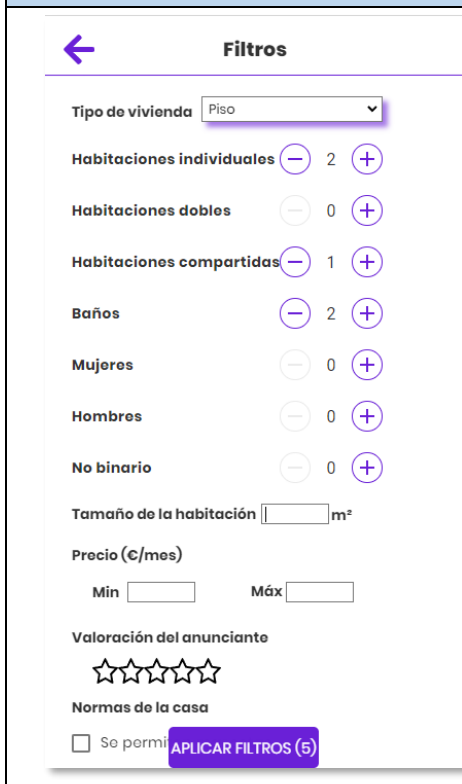
## Pantalla de resultado búsqueda por dirección



Esta es la pantalla que muestras los resultados de la búsqueda. En esta pantalla se pueden filtrar los resultados, aunque ya se hayan filtrado en la pantalla anterior. Al lado del icono de Filtrar aparece el número de filtros aplicados actualmente. También se puede ordenar por precio o distancia a la universidad más cercana.

En los anuncios se encuentra un icono de un corazón, el cual se encuentra gris si no está pulsado o morado en caso de pulsarlo. Esto sirve para guardar un piso en Favoritos.


## Pantalla de filtros




En esta pantalla se pueden elegir todos los filtros que se quieran para realizar búsquedas. En estos filtros se encuentra toda la información disponible de los pisos y de sus inquilinos, pudiendo filtrar por gustos de los inquilinos, normas de la casa, convivencia en la casa, etc.

Para aplicar los filtros, hay un botón fijo en la parte inferior de la pantalla que no se desplaza con el scroll de la pantalla.

## Pantalla de guardados

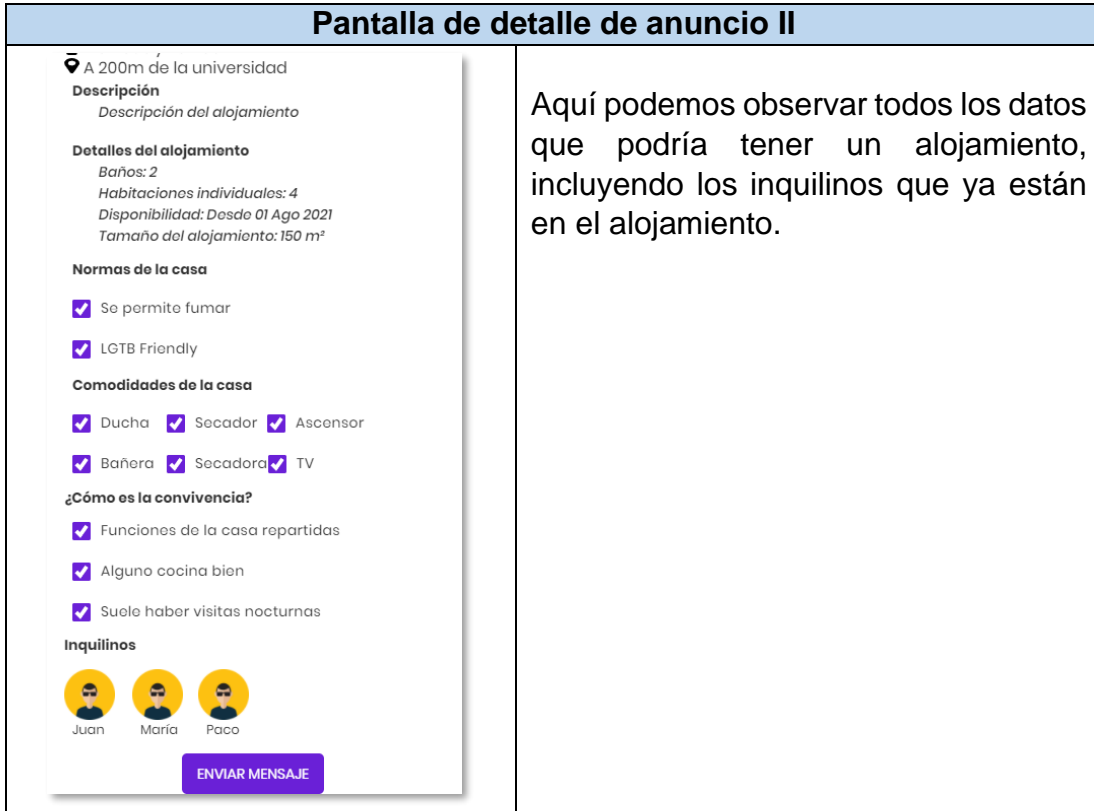
	<p>Se ha decidido cambiar el nombre a esta pantalla a Favoritos, pues es más universal y se utiliza en muchas aplicaciones.</p> <p>En esta pantalla aparecen aquellos alojamientos en los que se haya pulsado el corazón de la esquina superior derecha del anuncio. Así, el usuario puede tener guardados los anuncios que más le llamen la atención para no tener que volver a buscarlos.</p> <p>Si el usuario pulsa en el corazón en esta pantalla, el anuncio desaparecerá.</p>
---	---

## Pantalla de detalle de anuncio

	<p>Esta es la pantalla del detalle del anuncio. En ella, el usuario tiene la opción de guardarlo en Favoritos, leer toda la información, acceder al perfil del anunciante pulsando sobre su icono, enviar un mensaje e incluso, si el anunciante ha incluido a los inquilinos, puedes ver los perfiles de los otros inquilinos. Así el usuario puede ver los gustos, estilos de vida y demás, de sus posibles futuros compañeros</p>
---	--



## Pantalla de detalle de anuncio II



A 200m de la universidad

**Descripción**  
*Descripción del alojamiento*

**Detalles del alojamiento**  
Baños: 2  
Habitaciones individuales: 4  
Disponibilidad: Desde 01 Ago 2021  
Tamaño del alojamiento: 150 m²

**Normas de la casa**

- Se permite fumar
- LGBT Friendly

**Comodidades de la casa**

- Ducha  Secador  Ascensor
- Bañera  Secadora  TV

**¿Cómo es la convivencia?**

- Funciones de la casa repartidas
- Alguno cocina bien
- Suele haber visitas nocturnas

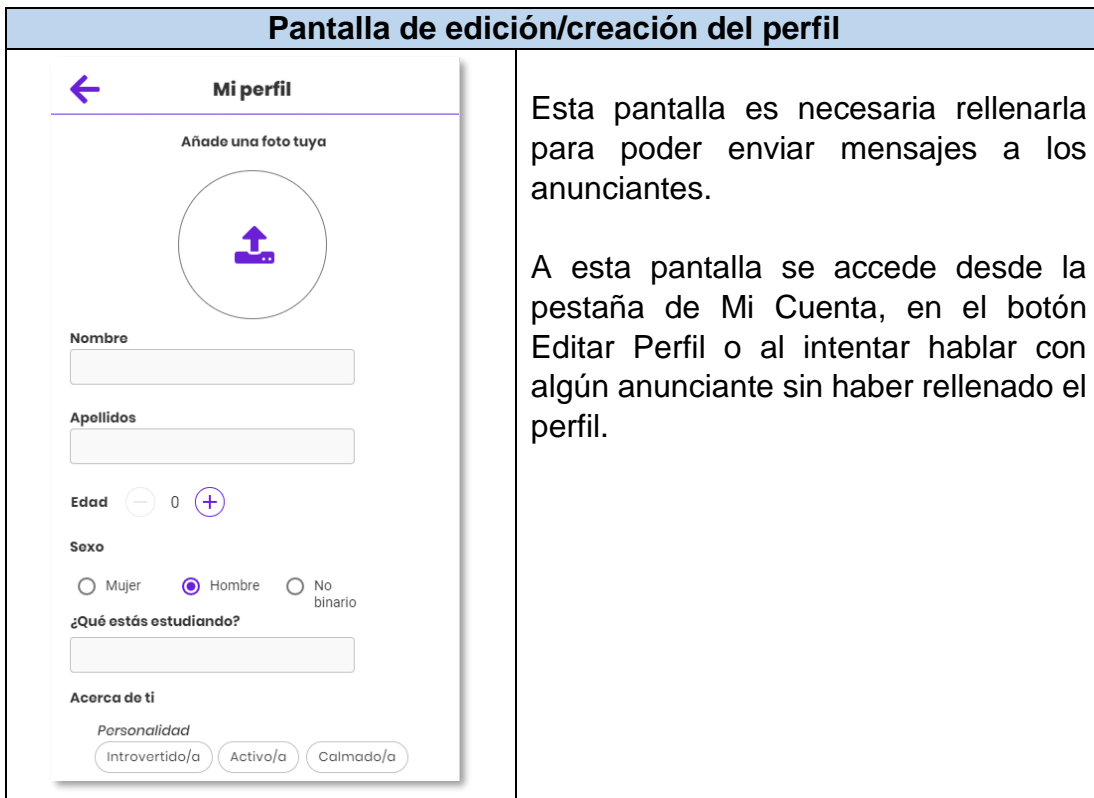
**Inquilinos**

Juan María Paco

ENVIAR MENSAJE


Aquí podemos observar todos los datos que podría tener un alojamiento, incluyendo los inquilinos que ya están en el alojamiento.

## Pantalla de edición/creación del perfil



**Mi perfil**

Añade una foto tuya



**Nombre**

**Apellidos**

**Edad**

**Sexo**

Mujer  Hombre  No binario

**¿Qué estás estudiando?**

**Acerca de ti**


**Personalidad**

Esta pantalla es necesaria rellenarla para poder enviar mensajes a los anunciantes.

A esta pantalla se accede desde la pestaña de Mi Cuenta, en el botón Editar Perfil o al intentar hablar con algún anunciante sin haber rellenado el perfil.

## Pantalla de mis alojamientos

← Mis alojamientos



**Habitación bien iluminada**  
Piso muy cerca de la universidad. Somos gente muy maja.

€300 €/mes  
4 habitaciones  
2 chicas y 1 chico  
A 200m de la universidad

Normas  
Servicios

PUBLICAR MÁS

En esta pantalla se muestran los alojamientos que tenemos en alquiler anunciados en la aplicación.

El usuario puede publicar otro pulsando en el botón inferior: Publicar más.

Pulsando sobre la tarjeta, el usuario puede acceder al anuncio o editarlo pulsando en el símbolo del lápiz, lo cual le llevaría a una pantalla muy similar a la de dar de alta un anuncio.

## Pantalla de alta de anuncio

← Pon tu anuncio

Tipo de vivienda

Ubicación

¿Qué quieres alquilar?

Piso completo  Habitación

¿Eres propietario de la vivienda?

Sí  No

Habitaciones individuales

Habitaciones dobles

Habitaciones compartidas

Baños

Mujeres

Hombres

No binario

Tamaño de la habitación  m<sup>2</sup>

Esta es la pantalla con la que dar de alta un nuevo anuncio. El usuario tendrá que rellenar los datos y pulsar el botón que se encuentra al final del formulario.

Esta pantalla mostrará más o menos campos dependiendo de si el usuario ha marcado que es propietario o no de la vivienda.

En el caso de que no sea propietario, saldrán campos adicionales para rellenar como son: la convivencia en el piso, la limpieza del piso, si suele haber fiestas por la noche...algo más orientado al estudiante y que el propietario no suele saberlo.

## Pantalla de alta de anuncio II

Se permiten fiestas  
 Se permiten fiestas

**Elige las comodidades de la casa**



Ducha  Secador  Ascensor  
 Bañera  Secadora  Piscina  
 TV  Parking  Jardín

**¿Cómo es la convivencia?**

Funciones de la casa repartidas  
 Alguno cocina bien  
 Suele haber visitas nocturnas  
 Noches tranquilas  
 Se suele comer juntos  
 Piso limpio regularmente

**Disponibilidad**  
Desde  Hasta

**Añadir inquilinos (Opcional)**


   
Tú


**PUBLICAR**





Este es el resto del formulario donde se pueden agregar la convivencia en la casa, las comodidades y los inquilinos que usen la aplicación y que ya vivan en ese alojamiento, para que los demás usuarios puedan saber con quién podrían vivir.

## Pantalla de chats

**Chat**

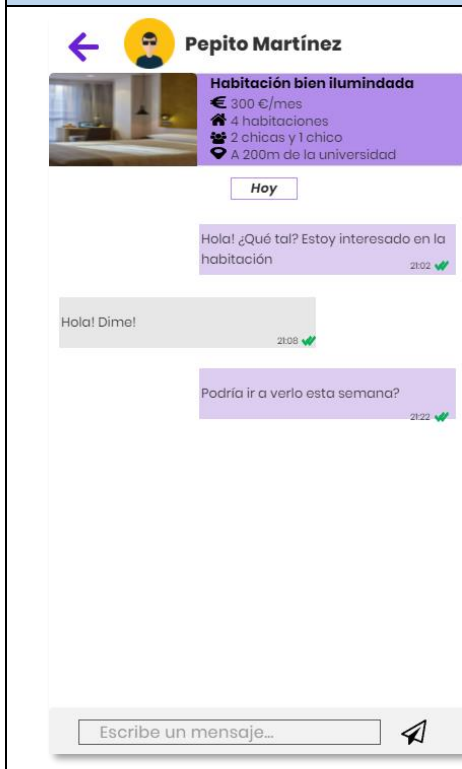
 **Pepito Gómez** Hace 3 horas  
Piso bien iluminado cerca...  
¿Cuándo podría ir al piso?

 **Juanjo Román** Hace 7 horas  
Habitación grande con to...  
¿Puedo ir a ver el piso?

En esta pantalla se muestran las conversaciones abiertas con los anunciantes o con los interesados.

## Pantalla de conversación



En esta pantalla se aprecia la conversación con un anunciante. En la parte superior aparece el nombre del anunciante (pulsando vamos a su perfil) y la habitación o piso en el que el usuario está interesado.

El usuario puede hablar gratuitamente todo lo que quiera con el anunciante y viceversa.

## Pantalla de perfil de usuario



Esta es la pantalla de la vista del perfil de un usuario cuando pulsa el usuario sobre el icono de alguien.

Podemos observar que se muestra una valoración del usuario como su rol. En este caso tiene el rol de inquilino y propietario, puesto que publicó un anuncio e indicó que vive en el alojamiento. Es por ello por lo que al hacer scroll aparecen los pisos de los que es propietario.

Pantalla de perfil de usuario	
<p>Empático/a    LGBT Friendly</p> <p>Estilo de vida</p> <p>Madrugador/a    Hogareño/a    Gamer</p> <p>Idiomas</p> <p>Inglés    Español</p> <p>Deportes</p> <p>Fútbol    Pádel</p> <p>Acerca de Pepito</p> <p>Soy un chico calmado que me encantan los videojuegos, salir con mis amigos y hacer deporte muy amenudo.</p> <p>En alquiler (1)</p> <p><b>Habitación bien iluminada</b>      Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean euismod bibendum in. Normas</p> <p>€ 300 €/mes    Servicios</p> <p>4 habitaciones</p> <p>2 chicas y 1 chico</p> <p>A 200m de la universidad</p>	<p>Aquí podemos observar, cómo se muestra en la propiedad <i>En alquiler</i>, los alojamientos que tiene en alquiler el usuario.</p> <p>Si el usuario pulsa, irá a la página del anuncio.</p>

### 2.1.5. Evaluación

Esta fase consistente en poner a prueba la aplicación con usuarios reales para encontrar mejoras en la aplicación, ha consistido en dos partes.

- **Primera fase:** La primera se ha realizado con un usuario focal al terminar la etapa de los bocetos a mano. Esta parte ha consistido en que el usuario realizase una tarea y nos diese su impresión y opinión de cosas a mejorar.

La tarea para realizar fue: Iniciar sesión en la aplicación, buscar una habitación en el mapa, filtrar los resultados para que solo aparecieran aquellas habitaciones que costaran menos de 300€ y chatear con algún propietario.

El usuario pudo realizar la tarea sin ningún problema, pero nos apuntó que le gustaría poder iniciar sesión con Google, porque siempre inicia sesión con esa cuenta en todas las demás aplicaciones. Esta mejora se tuvo en cuenta a la hora de realizar el prototipo de alta fidelidad.

- **Segunda fase:** La segunda fase se realizó al terminar el diseño del prototipo de alta fidelidad. Se le pidió a un usuario focal que realizara otra tarea para ver qué tal se desenvolvía.

La tarea que se pidió fue: Iniciar sesión con Google, ir a su perfil, editarlo y guardarlo. A continuación, realizar una búsqueda por dirección y guardar un piso como favorito. Por último, visitar sus favoritos y quitarlo de favoritos.

Al final del ejercicio, se pidió al usuario posibles mejoras que hicieran más fácil o intuitiva la aplicación. Comentó que le gustaba

mucho la aplicación y que era muy intuitiva. El único apunte que realizó fue que el corazón para guardar en favoritos no se ve del todo bien.

Por tanto, como mejora, se realizará el cambio del icono para que pueda verse más claramente, haciendo que resalte más con un borde blanco y en un rectángulo.

## 2.2. Diseño técnico

### 2.2.1. Definición de los casos de uso

<b>CU-001: Registrar usuario</b>	
<b>Descripción</b>	Registro de un usuario en la aplicación
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario no registrado
<b>Precondiciones</b>	El usuario tiene un email válido o una cuenta en Google, Facebook o Twitter.
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario accede a la aplicación.</li> <li>- El usuario pulsa sobre el botón de Crear cuenta</li> <li>- El usuario rellena el formulario y pulsa en Crear</li> <li>- La aplicación comprueba que el email no existe en la base de datos</li> <li>- El sistema redirige al usuario a la pantalla de búsqueda</li> </ul>
<b>Postcondiciones</b>	El usuario ha sido registrado, ha iniciado sesión y ha sido redirigido a la pantalla principal
<b>Notas</b>	Si al rellenar el formulario, el usuario existe previamente o no coinciden las contraseñas, se muestra un mensaje de error.

<b>CU-002: Iniciar sesión</b>	
<b>Descripción</b>	Inicio de sesión de un usuario existente
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario registrado
<b>Precondiciones</b>	El usuario tiene una cuenta creada en la aplicación
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario accede a la aplicación</li> <li>- El usuario introduce email y contraseña o pulsa sobre Google, Facebook o Twitter</li> <li>- El usuario accede a la aplicación.</li> </ul>
<b>Postcondiciones</b>	El usuario ha sido <i>logueado</i> satisfactoriamente y redirigido a la página de búsqueda
<b>Notas</b>	Solo es necesario iniciar sesión si has cerrado sesión o si acabas de descargarte la aplicación. Saldrán mensajes de error en caso de no tener credenciales válidas o que no exista cuentas vinculados a los servicios externos.

<b>CU-003: Recuperar contraseña</b>	
<b>Descripción</b>	Crear una nueva contraseña para usuario existente
<b>Prioridad</b>	Media
<b>Actores</b>	Usuario registrado
<b>Precondiciones</b>	Usuario registrado en el sistema
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario pulsa el botón de <i>He olvidado mi contraseña</i></li> <li>- El usuario introduce su email</li> <li>- El sistema comprueba que el email existe en la base de datos y envía un correo para cambiar la contraseña</li> <li>- El usuario introduce las nuevas credenciales</li> </ul>
<b>Postcondiciones</b>	El usuario tiene una nueva contraseña funcional

<b>CU-004: Cerrar sesión</b>	
<b>Descripción</b>	Cerrar la sesión de un usuario
<b>Prioridad</b>	Media
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	Usuario identificado dentro de la aplicación
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario navega a la opción de <i>Mi cuenta</i> y pulsa en <i>Cerrar sesión</i>.</li> <li>- El sistema muestra mensaje de confirmación</li> <li>- El usuario confirma el cierre de sesión</li> <li>- El sistema cierra la sesión del usuario y le redirige a la página de identificación</li> </ul>
<b>Postcondiciones</b>	El usuario no está identificado en la aplicación

<b>CU-005: Eliminar cuenta</b>	
<b>Descripción</b>	Eliminar a un usuario y sus datos del sistema
<b>Prioridad</b>	Alta
<b>Actores</b>	Un usuario registrado
<b>Precondiciones</b>	El usuario debe estar registrado e identificado
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario accede al apartado de <i>Mi cuenta</i>, y pulsa en <i>Eliminar cuenta</i></li> <li>- La aplicación muestra un mensaje de confirmación</li> <li>- El usuario confirma la operación</li> <li>- El sistema borra todos los datos y al usuario de las bases de datos y le redirige a la identificación</li> </ul>
<b>Postcondiciones</b>	El usuario no puede iniciar sesión y sus datos ya no existen

<b>CU-006: Editar perfil</b>	
<b>Descripción</b>	Editar el perfil de un usuario
<b>Prioridad</b>	Alta
<b>Actores</b>	Un usuario registrado
<b>Precondiciones</b>	El usuario debe estar registrado e identificado

<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario accede al apartado de <i>Mi cuenta</i>, y pulsa en <i>Editar Perfil</i>. El usuario edita la información y pulsa en <i>Guardar</i>.</li> <li>- El sistema guarda la información del usuario</li> </ul>
<b>Postcondiciones</b>	Los datos del usuario se han modificado

<b>CU-007: Ver mis alojamientos</b>	
<b>Descripción</b>	Ver los alojamientos puestos en anuncios
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	El usuario debe haber anunciado un alojamiento
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario accede al apartado de <i>Mi cuenta</i>, y pulsa en <i>Ver mis alojamientos publicados</i></li> <li>- El sistema muestra una pantalla con los anuncios publicados por el usuario</li> </ul>
<b>Postcondiciones</b>	El usuario visualiza los anuncios publicados correctamente

<b>CU-008: Ver favoritos</b>	
<b>Descripción</b>	Ver los alojamientos favoritos
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	El usuario ha marcado alojamientos como favoritos
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario accede a la sección de favoritos del menú fijo</li> <li>- El sistema muestra en una lista los alojamientos marcados como favoritos</li> </ul>
<b>Postcondiciones</b>	El usuario ve correctamente sus alojamientos favoritos

<b>CU-009: Buscar alojamientos por dirección</b>	
<b>Descripción</b>	Buscar alojamientos según una dirección, ciudad...
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	N/A
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario se encuentra en la página inicial de búsqueda. El usuario introduce una dirección y pulsa en <i>Buscar</i></li> <li>- El sistema muestra una pantalla con la lista de resultados</li> </ul>
<b>Postcondiciones</b>	El usuario ve correctamente la lista de alojamientos según la dirección

<b>CU-010: Buscar alojamientos en el mapa</b>	
<b>Descripción</b>	Buscar alojamientos en el mapa
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	-
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario se encuentra en la pantalla inicial y pulsa en <i>Buscar en el mapa</i>.</li> </ul>



	- El sistema muestra una pantalla con la ubicación actual del usuario y los resultados cercanos en el mapa
<b>Postcondiciones</b>	El usuario ve correctamente el mapa y los alojamientos disponibles

<b>CU-011: Añadir o quitar de favoritos</b>	
<b>Descripción</b>	Añadir o quitar alojamientos de favoritos
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	-
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario realiza una búsqueda y pulsa sobre el icono de Favorito de un anuncio.</li> <li>- El sistema marca como favorito el anuncio</li> </ul>
<b>Postcondiciones</b>	El alojamiento se ha guardado como favorito y se puede ver en la pantalla de <i>Favoritos</i>

<b>CU-012: Contactar con un anunciante</b>	
<b>Descripción</b>	Contactar por chat interno con un anunciante
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	El usuario tiene completo el perfil
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario accede a un anuncio de un alojamiento y pulsa en <i>Enviar mensaje</i></li> <li>- El sistema crea una conversación con el anunciante</li> </ul>
<b>Postcondiciones</b>	El usuario tiene una conversación creada en la pantalla de <i>Chats</i>

<b>CU-013: Acceder al detalle de un anuncio</b>	
<b>Descripción</b>	Acceder a toda la información de un anuncio
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	Debe haberse creado un anuncio con anterioridad
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario realiza una búsqueda y pulsa sobre un anuncio</li> <li>- El sistema muestra una nueva pantalla con toda la información proporcionada por el anunciante</li> </ul>
<b>Postcondiciones</b>	El usuario puede ver los detalles del anuncio

<b>CU-014: Ver y enviar mensajes</b>	
<b>Descripción</b>	Enviar y recibir mensajes entre usuarios
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	El usuario ha contactado con un anunciante
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario accede a la sección <i>Chats</i> del menú fijo inferior. El usuario accede a una conversación.</li> </ul>

	<ul style="list-style-type: none"> <li>- El sistema muestra los mensajes recibidos y enviados de la conversación</li> <li>- El usuario envía un mensaje</li> </ul>
<b>Postcondiciones</b>	El usuario puede enviar y recibir mensajes

<b>CU-015: Editar un anuncio</b>	
<b>Descripción</b>	Editar un anuncio
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	El usuario ha publicado un anuncio
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario accede a la sección de <i>Ver mis alojamientos publicados</i> en la sección <i>Mi cuenta</i>. El usuario pulsa sobre el icono del lápiz para editar.</li> <li>- El sistema genera una pantalla para editar el anuncio</li> <li>- El usuario edita algunos elementos del anuncio y pulsa en <i>Guardar</i></li> <li>- El sistema guarda la nueva información</li> </ul>
<b>Postcondiciones</b>	El usuario ha modificado correctamente el anuncio
<b>Notas</b>	

<b>CU-016: Eliminar un anuncio</b>	
<b>Descripción</b>	Eliminar un anuncio
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	El usuario ha publicado un anuncio
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario accede a la sección de <i>Ver mis alojamientos publicados</i> en la sección <i>Mi cuenta</i>. El usuario pulsa sobre el icono de la papelera para eliminar el anuncio</li> <li>- El sistema muestra una confirmación</li> <li>- El usuario confirma la operación</li> <li>- El sistema elimina el anuncio</li> </ul>
<b>Postcondiciones</b>	El usuario ha eliminado correctamente el anuncio

<b>CU-017: Filtrar búsqueda</b>	
<b>Descripción</b>	Aplicar filtros a una búsqueda
<b>Prioridad</b>	Alta
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	-
<b>Flujo</b>	<p>Flujo 1</p> <ul style="list-style-type: none"> <li>- El usuario antes de realizar una búsqueda selecciona el icono de los filtros y selecciona algunos. Después, pulsa en <i>Aplicar filtros</i>.</li> <li>- El sistema carga los filtros para cuando pulse en buscar.</li> </ul> <p>Flujo 2</p>

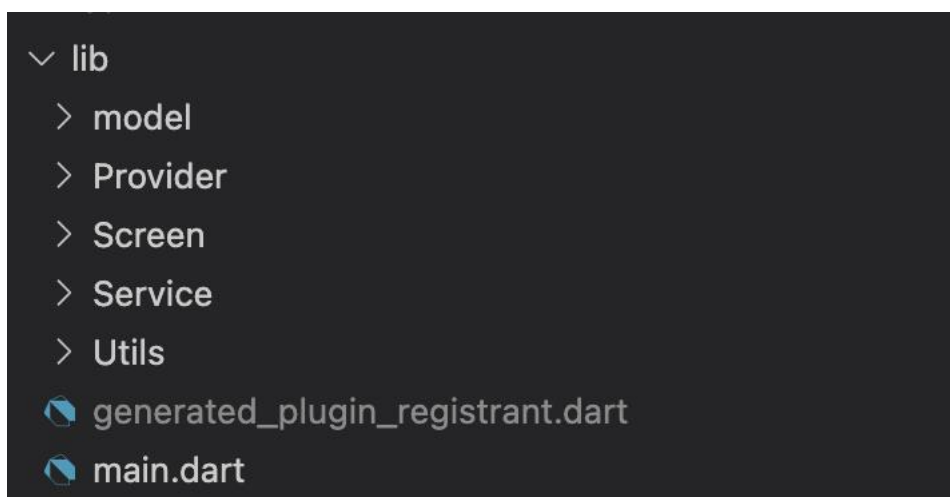
	<ul style="list-style-type: none"> <li>- El usuario realiza una búsqueda por dirección.</li> <li>- El sistema muestra los resultados.</li> <li>- El usuario pulsa sobre el icono de filtros, selecciona algunos y pulsa en <i>Aplicar filtros</i></li> <li>- El sistema muestra los resultados con los filtros seleccionados por el usuario</li> </ul> <p>Flujo 3</p> <ul style="list-style-type: none"> <li>- El usuario pulsa sobre <i>Búsqueda en el mapa</i>.</li> <li>- El sistema carga el mapa y los resultados.</li> <li>- El usuario pulsa sobre el icono de <i>Filtros</i> y selecciona algunos y pulsa en <i>Aplicar filtros</i>.</li> <li>- El sistema muestra los resultados filtrados.</li> </ul>
<b>Postcondiciones</b>	El usuario realiza la búsqueda según los filtros

CU-018: Ordenar búsqueda	
<b>Descripción</b>	Ordenar la búsqueda
<b>Prioridad</b>	Media
<b>Actores</b>	Usuario identificado
<b>Precondiciones</b>	-
<b>Flujo</b>	<ul style="list-style-type: none"> <li>- El usuario realiza una búsqueda por dirección</li> <li>- El sistema muestra los resultados de la búsqueda</li> <li>- El usuario pulsa sobre el icono de <i>Ordenar</i> y selecciona el método de ordenación y pulsa en <i>Aplicar</i></li> <li>- El sistema muestra los resultados ordenados</li> </ul>
<b>Postcondiciones</b>	El usuario ve los resultados ordenados según el criterio elegido

## 2.2.2. Diseño de la arquitectura

### 2.2.2.1. Arquitectura de la aplicación

Respecto a la arquitectura de la aplicación, se ha realizado la siguiente estructura en la aplicación:



- **Model:** Aquí se almacenan los POJOs. Los POJOs son los modelos planos de los objetos que se utilizan en la aplicación. No incluyen ninguna lógica de negocio. Estos objetos son aquellos que nos permitirán transformar la información que proviene de la base de datos y transformarla a una representación con la que se pueda trabajar en la aplicación.
- **Screen:** En este paquete se encuentran todos los Widgets de la aplicación, es decir, todos los elementos visuales o 'vista' de la aplicación.
- **Service:** Aquí se encuentran todos los métodos necesarios para interactuar con la base de datos o con el almacenamiento en la nube donde se almacenan las fotos (del usuario o de los alojamientos). Aquí se hace uso de los modelos que hemos definido previamente. Estos servicios serán invocados desde la vista de la aplicación para recuperar los datos necesarios en cada momento o para guardar o eliminar datos de la base de datos.
- **Utils:** En este paquete se encuentran clases que ofrecen funcionalidades comunes que se pueden utilizar en diferentes puntos de la aplicación.
- **Provider:** En este paquete se han guardado las clases necesarias para implementar tanto el patrón BLOC de Flutter como el patrón Provider. Estos patrones sirven para manejar el estado de la aplicación desde un punto concreto.

#### 2.2.2.2. Diseño de la base de datos

Respecto a la base de datos usada para el proyecto, se ha optado por Cloud Firestore<sup>6</sup> de Firebase. Es una base de datos NoSQL flexible, escalable y en la nube, con el propósito de almacenar datos y sincronizarlos para el desarrollo tanto del cliente como del servidor.

Las bases de datos NoSQL se diferencian de las clásicas bases de datos en:

- No utilizan SQL como lenguaje de consultas.
- No tienen un modelo Entidad Relación.
- No utilizan estructuras fijas como tablas para el almacenamiento de los datos.

Debido a que las bases de datos NoSQL no tienen un modelo Entidad Relación como la tienen las bases de datos SQL, no se ha producido un diagrama Entidad Relación. En su lugar, a continuación, se presentan las colecciones que se van a usar en el proyecto:

---

<sup>6</sup> <https://firebase.google.com/docs/firestore?authuser=0>

**users** = {id: string, email: string, password: string, name: string, lastname: string, age: number, genre: number, profileImage: string, studies: number, personality: array[number], lifestyle: array[number], idioms: array[number], sports: array[number], aboutself: string, favorites: array[string], onRent: array[string], stars: number, chats: array[chat]}

**houses** = {id: string, title: string, description: string, location: string, house: boolean, type: number, room: boolean, owner: boolean, individualRooms: number, doubleRooms: number, sharedRooms: number, baths: number, women: number, mens: number, noBinary: number, size: number, price: number, rules: array [rule], facilities: array[facility], coexistences: array[coexistence], fromDate: timestamp, toDate: timestamp, tenants: array[number], publishDate: timestamp}

**rule** = {id: string, description: string}

**facility** = {id: string, description: string}

**coexistence** = {id: string, description: string}

**type** = {id: string, description: string}

**chat** = {id: string, otherUser: string, room: string, messages: array[message]}

**message** = {id: string, text: string, date: timestamp, sender: string}

### 2.3. Revisión de la planificación

En esta etapa de diseño, el número de horas invertidas ha sido más elevada de la esperada.

Por una parte, se estimó 8 horas para el diseño conceptual, que acabaron siendo 4 horas, pero, por otra parte, el diseño del prototipo de alta fidelidad ha llevado mucho más tiempo del esperado. Este inconveniente ha venido propiciado, entre otras cosas, por el desconocimiento del software utilizado. La estimación inicial del diseño del prototipo fue de 11 horas, que se han convertido en 34 horas, lo cual ha descuadrado la planificación, pero se recuperará invirtiendo más horas por día en las siguientes etapas.

## 3. Implementación

### 3.1. Herramientas para el desarrollo

Para esta parte del trabajo se han utilizado diferentes herramientas, que se exponen a continuación.

#### 3.1.1. Flutter

Flutter es un SDK de Google, de código abierto, para el desarrollo de aplicaciones. Con este framework se pueden realizar aplicaciones tanto para Android, como para iOS, web o aplicaciones para escritorio, desarrollando prácticamente un solo código para todas las plataformas a la vez. Como ventajas de este framework:

Compila en nativo, tanto en Android como iOS.

La creación de interfaces gráficas es muy flexible.

El desarrollo es muy rápido, ofreciendo además el 'hot reload', que permite ver los cambios realizados en la interfaz, simplemente con guardar el archivo.

Como posibles desventajas de utilizar este framework es su lenguaje, Dart. Es un lenguaje desarrollado también por Google pero que necesita una curva de aprendizaje ya que no se utiliza en ningún otro sitio, aunque es muy parecido a Java o C#.

#### 3.1.2. Visual Studio Code

Para comenzar a trabajar con Flutter, se ha utilizado Visual Studio Code, un editor de código desarrollado por Microsoft, el cual se integra muy bien con Flutter y Dart, gracias a que ofrecen los plugins oficiales de Dart y Flutter.

#### 3.1.3. Firebase

Para el backend se ha hecho uso de Firebase. Firebase es una plataforma en la nube para el desarrollo de aplicaciones web y móvil. Esta plataforma ayuda mucho al desarrollo, facilitando la creación de un backend al desarrollador, proporcionando herramientas como bases de datos, autenticación en aplicaciones, almacenamiento en la nube, seguimiento de errores y muchas más. De esta herramienta se ha aprovechado Firebase Auth, Cloud Firestore y Firebase Storage.

- **Firebase Authentication:** Con Firebase Authenticator Podemos realizar la autenticación de los usuarios en la aplicación de forma muy simple. Esta herramienta proporciona servicios de backend, SDK fáciles de usar y bibliotecas de UI ya elaboradas para autenticar a los usuarios en la aplicación.

- **Cloud Firestore:** Esta es la base de datos utilizada en el proyecto. Es una base de datos noSQL escalable y flexible que hace uso de Firebase y Google Cloud.
- **Firestore Storage:** Es un almacenamiento en la nube proporcionado por Firebase, en el cual se almacenan tanto las fotos de los perfiles de los usuarios como las fotos de los anuncios que se publiquen.

#### 3.1.4. Android Studio

Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones para Android y está basado en IntelliJ IDEA. Este entorno ha sido utilizado para simular diferentes dispositivos móviles y realizar pruebas de la aplicación durante el desarrollo.

#### 3.1.5. Google Cloud Platform (GCP)

GCP es una plataforma en la nube que reúne diferentes servicios de tecnología de la información de Google que permite trabajar de una forma mucho más eficiente, ganar flexibilidad y permite una ventaja estratégica. Esta plataforma ha sido utilizada para añadir a nuestro proyecto las APIs:

- Maps SDK for Android
- Maps JavaScript API
- Places API
- Directions API

También ha sido utilizada para poder utilizar el inicio de sesión con Google+. Para ello se ha tenido que configurar la pantalla de consentimiento de OAuth

## 3.2. Librerías

El proyecto se divide en dos grandes elementos: por un lado, tenemos el backend y por otro tenemos la aplicación que será la que utilice ese backend.

Para el desarrollo de la aplicación se han utilizado varias librerías disponibles en el repositorio oficial de Flutter, lo que sirve de apoyo para realizar ciertas funcionalidades. Algunas de las librerías más importantes utilizadas en el proyecto han sido:

- **Image\_picker:** esta librería facilita el poder elegir una imagen de la galería o tomar una foto y añadirla a la aplicación. Se ha hecho uso de esta librería para añadir la foto del usuario y para añadir fotos a un anuncio.

- **carousel\_slider**: esta librería nos permite realizar un carrusel de fotos de una manera muy simple. Utilizada para mostrar las fotos de los anuncios.
- **uuid**: esta librería permite crear identificadores únicos basados en el tiempo. Se ha utilizado para crear los identificadores de los usuarios y de los anuncios.
- **Flutter\_spinbox**: con esta librería se crea un spinbox que se ha utilizado para la publicación de un anuncio, para seleccionar el número de habitaciones, baños, etc...
- **Librerías de Firebase**: Se han utilizado `firebase_auth`, `firebase_storage`, `firebase_core` y `cloud_firestore`. Librerías necesarias para interactuar de forma sencilla con los productos de Firebase.
- **Google\_maps\_flutter**: Librería para usar los mapas de Google en nuestra aplicación. Se usa para la búsqueda en el mapa.

### 3.3. Patrones

A lo largo de todo el desarrollo se ha utilizado la misma estrategia. Una vista o pantalla que realiza llamadas a los servicios de la aplicación. Estos servicios son los que se encargan de llamar al backend y recuperar, editar o eliminar datos, haciendo uso de los objetos declarados en el paquete *model*. Pero para ciertas funcionalidades de la aplicación, se ha hecho uso de un patrón muy común en el desarrollo de aplicaciones en Flutter. Este patrón solo se ha utilizado en ciertas funcionalidades especiales. Para estas funcionalidades era necesario poder acceder al estado de los objetos desde diferentes pantallas y este patrón nos proporciona una forma simple de acceder a él desde cualquier sitio.

Estas funcionalidades son el sistema de mensajes por chats, el menú para navegar por las diferentes pestañas de la aplicación y para todo el tema relacionado con la localización.

El patrón utilizado es el patrón Provider:

- **Provider**: es un patrón de Flutter que proporciona el modelo de datos actual en el lugar donde lo necesitamos actualmente. Contiene algunos datos y notifica a los observadores cuando ocurre un cambio. Para ello la clase que queremos que esté disponible tiene que extender la clase `ChangeNotifier` y los Widgets que queremos que modifiquen o consulten esos datos son los consumidores y puedes acceder a estos datos desde el objeto `Consumer` o `Provider.of()`. Para poder usar el provider hay que instanciarlo en un Widget padre de donde queramos utilizarlo. Se suele instanciar en el Widget padre de todos para poder acceder en toda la aplicación. Este patrón ha sido utilizado en la aplicación para manejar el estado del menú y del chat de la aplicación.

### 3.4. Pruebas



### 3.4.1. Pruebas sin usuarios finales

Durante el proceso de desarrollo se han ido realizando pruebas continuamente con un dispositivo físico y con un dispositivo virtual generado con Android Studio. A continuación, se muestran las pruebas realizadas al final de la etapa de desarrollo:

INICIO DE SESIÓN	
Descripción	Resultado
La interfaz de adapta correctamente para tamaños de pantalla correctamente	Sí
Se puede acceder a crear una cuenta	Sí
Se puede acceder con usuario y contraseña	Sí
Se muestran correctamente los errores si no existe el usuario o el correo	Sí. Fallo: se muestran los datos introducidos, pero es solo visual, hay que volver a introducirlos.
Se puede iniciar sesión con los servicios de Google, Facebook y Twitter	No implementado
Se puede ver la contraseña pulsando en el icono del ojo	Sí
Si ya has iniciado sesión, cuando vuelves a iniciar la aplicación lleva directamente a la página de búsqueda en vez de a iniciar sesión	Sí

REGISTRO	
Descripción	Resultado
Se puede acceder de nuevo a la pantalla de inicio de sesión	Sí
Si no se rellena alguno de los campos se muestra un error	Sí, se muestra un error indicando que todos los campos son obligatorios
Se muestra la contraseña pulsando en el icono del ojo	Sí
Se muestra un error si se intenta dar de alta un email ya existente	Sí, se muestra un mensaje de que el email ya existe
Se crea correctamente una cuenta y entra en la aplicación automáticamente	Sí
Al crear una nueva cuenta, lleva a la pantalla de editar perfil la primera vez	Sí

PERFIL	
Descripción	Resultado
La primera vez que se edita el perfil el botón atrás lleva a la Cuenta del usuario	No. Lleva del nuevo al Login.

Se edita correctamente el perfil del usuario	Sí
Se añade una foto por defecto si el usuario no quiere poner una	Sí
Se añade correctamente la foto subida por el usuario	Sí
Se pueden buscar estudios correctamente según se escribe	Sí
Se muestra error si el nombre o el apellido están vacíos	Sí

<b>BÚSQUEDA</b>	
<b>Descripción</b>	<b>Resultado</b>
Si se pulsa en buscar sin ninguna dirección, se muestran todos los pisos disponibles	Sí
Si se pulsa en el icono de filtros, se abre la pantalla de filtros	Sí
Si se seleccionan filtros y se pulsa en aplicar filtros, se muestran los resultados filtrados	Sí. Fallo: No filtra en la búsqueda en el mapa
Se puede cambiar entre tipo de búsqueda	Sí
Se puede acceder a la pantalla de creación de anuncios desde la búsqueda de dirección	Sí
Se muestran los pisos en la ubicación del usuario	Sí
Si los filtros no arrojan resultados, se muestra un mensaje	Sí, se muestra un mensaje y un botón para volver a la pantalla de buscar
Al pulsar en un marcador del mapa, se muestra un resumen del anuncio	Sí

<b>FAVORITOS</b>	
<b>Descripción</b>	<b>Resultado</b>
Se muestran los anuncios marcados como favoritos del usuario	Sí
Se muestra un mensaje si no tiene favoritos	Sí. Y se muestra un botón para buscar anuncios
Si se pulsa sobre el icono del corazón en un favorito, se elimina de la lista	Sí
Si se pulsa sobre un anuncio de la lista, se muestran sus detalles	Sí

PUBLICAR UN ANUNCIO	
Descripción	Resultado
Validación de datos obligatorios rellenos	Sí. Se muestran un error genérico si algún campo obligatorio no está relleno
Si se rellenan todos los datos obligatorios se crea correctamente el anuncio	Sí
Se pueden añadir fotos	Sí
Se pueden añadir inquilinos al anuncio, buscando por el nombre del usuario	Sí. Fallo: No se actualiza la pantalla correctamente pero sí se añade
Si se intenta añadir un usuario como inquilino ya añadido, se muestra un mensaje	Sí

CHATS	
Descripción	Resultado
Se muestra una lista con los chats activos	Sí
Notificaciones sobre nuevos mensajes	No implementado
Se muestra en el chat individual el anuncio por el que se ha contactado y el perfil del otro usuario	Sí
Se puede acceder al perfil desde el chat	Sí
Se puede acceder al anuncio desde el chat	Sí
Denunciar un chat o a un usuario	No implementado
Eliminar chat	No implementado

CUENTA	
Descripción	Resultado
Se puede acceder a las otras funciones de la aplicación	Sí
Se puede cerrar la sesión del usuario	Sí
Se puede borrar la cuenta del usuario	Sí
Se carga el usuario y su foto	Sí
Se tiene acceso a la pantalla de los anuncios publicados por el usuario	Sí
Al intentar borrar la cuenta, sale un mensaje de confirmación	Sí

ANUNCIO	
Descripción	Resultado
Se puede editar o borrar si el anuncio es del usuario autenticado	Sí

Se puede añadir a favoritos si no es un anuncio del usuario autenticado	Sí
Se muestran todos los datos del anuncio	Sí
Se muestra la distancia a la universidad más cercana	Sí

### 3.4.2. Pruebas con usuarios finales

Durante los últimos compases de la etapa del desarrollo se han realizado pruebas a tres usuarios potenciales de la aplicación. En estas pruebas, se les pidió a estos usuarios que llevaran a cabo ciertas tareas y se les observó para ver si cometían algún error o no conseguían realizar las tareas.

Las tareas para realizar fueron las siguientes:

- Crear una cuenta de usuario y rellenar su perfil.
- Cerrar sesión y volver a iniciar sesión con el usuario que acaban de crear.
- Buscar un piso en Getafe y añadirlo a sus favoritos.
- Crear un nuevo anuncio.
- Buscar un piso con 3 baños y enviar un mensaje al propietario.

Los tres usuarios completaron sin mayor problema todas las tareas, sin cometer muchos fallos y sin demorarse en exceso en las tareas. Como observaciones de los usuarios, uno de ellos comentó que cuando se creaba el anuncio no llevara a la pantalla inicial de búsqueda, sino a la lista de pisos del usuario y al contactar con el propietario de un alojamiento, esperaba que le condujera dentro de la conversación y no a la sección de Chats. Otro de los usuarios nos comentó que no vio fácilmente el botón de los filtros.

Gracias a estas pruebas se cambió la redirección al terminar la creación de un anuncio y al crear un nuevo chat y se cambió el icono de filtros por un botón más explicativo. También, nos confirmó, gracias a la valoración por parte de los usuarios que la interfaz era muy intuitiva, fácil de usar y bonita estéticamente.

## 3.5. Revisión de la planificación

Durante la etapa de desarrollo se estimó una duración de la etapa de 160 horas. Debido a la complejidad de la aplicación y a la errónea estimación de tiempo necesario, las horas dedicadas al final de la etapa ha sido de 230 horas.

Sumado a la complejidad de la aplicación, la poca experiencia con el framework de Flutter y la gestión de los estados, fueron dos de los principales motivos de la diferencia de horas entre la estimación inicial y la real.

Incluso con este aumento de horas dedicadas, han quedado ciertas tareas de prioridad baja sin realizar y ciertos errores o 'bugs'. Estas tareas no hacen referencia a requisitos esenciales de la aplicación ni impiden el correcto funcionamiento de esta (por ejemplo, el inicio de sesión o registro con servicios de terceros como Google, Twitter o Facebook), pero algunas de ellas se

intentaron implementar durante la siguiente fase del proyecto para terminar de pulir el producto.

Durante la última fase, se terminó de dar forma al producto, puliendo detalles, corrigiendo ciertos errores y añadiendo alguna pequeña funcionalidad, dando prioridad a la solución de errores por encima de añadir nuevas funcionalidades.

Las pequeñas funcionalidades añadidas fueron:

- El inicio de sesión y la creación de cuenta mediante Google+.
- Posibilidad de borrar un chat.
- Añadida la función de recuperar contraseña.

Y alguno de los errores corregidos fueron:

- Corregido error al iniciar sesión. Si la contraseña no era correcta, se quedaban los datos introducidos, pero había que volver a escribirlos.
- Corregido comportamiento al quitar de favoritos un anuncio.
- Corregido fallo al entrar a un piso que pertenece al usuario desde el chat. Se mostraba el botón de contactar y de marcar favorito.
- Corregido error visual al entrar al perfil de otro usuario.

## 4. Conclusiones

La mayor conclusión que he sacado con la realización de este proyecto es la importancia vital de realizar una planificación inicial con la mayor precisión posible. A esta conclusión he llegado debido a que la estimación de horas y esfuerzo realizada a comienzos del proyecto fue poco menos que optimista, obligando a realizar un sobre esfuerzo para poder completar el proyecto. Este sobre esfuerzo se produjo debido a la falta de experiencia en proyectos grandes y en la tecnología utilizada.

Otra conclusión importante a la que he llegado es la importancia de contar con la opinión del usuario desde las fases más tempranas del proyecto. Al final, el producto es para ellos y nadie mejor que ellos para decirnos qué quieren, qué ven raro o complicado o qué cambiarían. Esto nos proporciona información muy valiosa y nos puede ahorrar mucho tiempo en el futuro.

Por otro lado, el proyecto me ha servido para darme cuenta del potencial que tiene Flutter y lo mucho que me gusta. Realizando la parte de la implementación, aun teniendo que invertir algunos días más de 12 horas en ello, no ha resultado nada tedioso y me hace plantearme mi futuro profesional. También me ha servido para profundizar tanto en las herramientas de Firebase como en la API de Google Maps.

En el ámbito de los objetivos conseguidos, el resultado es muy positivo. Casi todos los objetivos de prioridad alta han sido implementados, y algunos errores y mejoras de prioridad baja, han sido solucionadas como he comentado en el capítulo anterior.

En líneas generales, siento un gran orgullo por el proyecto realizado. Considero que es una aplicación, atractiva, usable y que puede llegar a cubrir una necesidad. Por supuesto que la aplicación tiene margen de mejora y puede

albergar muchas nuevas funcionalidades, por ello, como trabajo futuro para la aplicación, sería interesante:

- Notificaciones push en el móvil cuando te contacten por un anuncio.
- Alerta de los mensajes nuevos en los chats.
- Opción para desactivar notificaciones.
- Opción para que no te hablen usuarios con el perfil incompleto.
- Adaptar la aplicación para iOS, que al haber utilizado Flutter en el proyecto, solo habría que modificar ciertas cosas, pero prácticamente el desarrollo ya está realizado.

## 5. Glosario

- **Backend:** Pieza de la aplicación que se encarga del almacenamiento y del procesamiento de la información.
- **Bug:** Término inglés para referirse a un error o defecto en el software.
- **Sprint:** En el contexto de las metodologías ágiles, se refiere a un periodo aproximado de entre 2 y 3 semanas, utilizado para organizar el trabajo y conseguir un incremento de valor en el producto que se está creando.
- **SDK:** siglas de Software Development Kit, en español kit de desarrollo de software.
- **Login:** término inglés que hace referencia al inicio de sesión.
- **API:** es una abreviatura de Application Programming Interfaces, que en español significa interfaz de programación de aplicaciones

## 6. Bibliografía

- (s.f.). Obtenido de Programador clic:  
<https://programmerclick.com/article/388054092/>
- (s.f.). Obtenido de Programación y más:  
<https://programacionymas.com/blog/android-mvc-mvp-mvvm>
- Abreu, L. (14 de 05 de 2014). *Why and How to Avoid Hamburger Menus*. Obtenido de Imjabreu : <https://Imjabreu.com/post/why-and-how-to-avoid-hamburger-menus/>
- Axure RP 10. (s.f.). Obtenido de <https://www.axure.com/>
- Garzías, J. (s.f.). Obtenido de <https://www.javiergarzas.com/metodologias-agiles>
- Gómez, K. (s.f.). *Top 5 Metodologías de Desarrollo Software*. Obtenido de MegaPractical: <https://www.megapractical.com/blog-de-arquitectura-soa-y-desarrollo-de-software/metodologias-de-desarrollo-de-software>
- Google. (13 de 10 de 2021). Obtenido de Firebase: <https://firebase.google.com/docs/firestore?authuser=0>
- Google. (s.f.). *Guía de arquitectura de apps*. Obtenido de <https://developer.android.com/jetpack/guide>
- Metodología Scrumban*. (s.f.). Obtenido de Sinnaps: <https://www.sinnaps.com/blog-gestion-proyectos/metodologia-scrumban>
- MONTES ALVAREZ, N. (8 de Abril de 2021). *Patrón Bloc en flutter*. Obtenido de <https://noemontes.dev/?p=285>
- Moreno, P. L. (19 de Mayo de 2020). *Dev Community*. Obtenido de Patrón Provider en Flutter: [https://dev.to/cat\\_yena/patron-provider-en-flutter-4hf1](https://dev.to/cat_yena/patron-provider-en-flutter-4hf1)
- Rancea, B. (15 de 07 de 2021). *3 formas de garantizar una navegación intuitiva y fácil de usar para su tienda de comercio electrónico*. Obtenido de Ecommerce platforms: <https://ecommerce-platforms.com/es/articles/ecommerce-store-navigation>
- Rose, A. (8 de 4 de 2014). *The next web*. Obtenido de UX designers: Side drawer navigation could be costing you half your user engagement: <https://thenextweb.com/news/ux-designers-side-drawer-navigation-costing-half-user-engagement>
- Stern, M. (2014). *Apple on Hamburger Menus*. Obtenido de <http://blog.manbolo.com/2014/06/30/apple-on-hamburger-menus>.
- Universidad Rey Juan Carlos. (2016). *Procesos Software. Apuntes de la asignatura de Procesos Software*.
- Universitat de València. (s.f.). *Metodologías Ágiles vs Metodologías Tradicionales*. Obtenido de Executive Master Project Management: <https://uv-mdap.com/programa-desarrollado/bloque-iv-metodologias-agiles/metodologias-agiles-vs-tradicionales/>



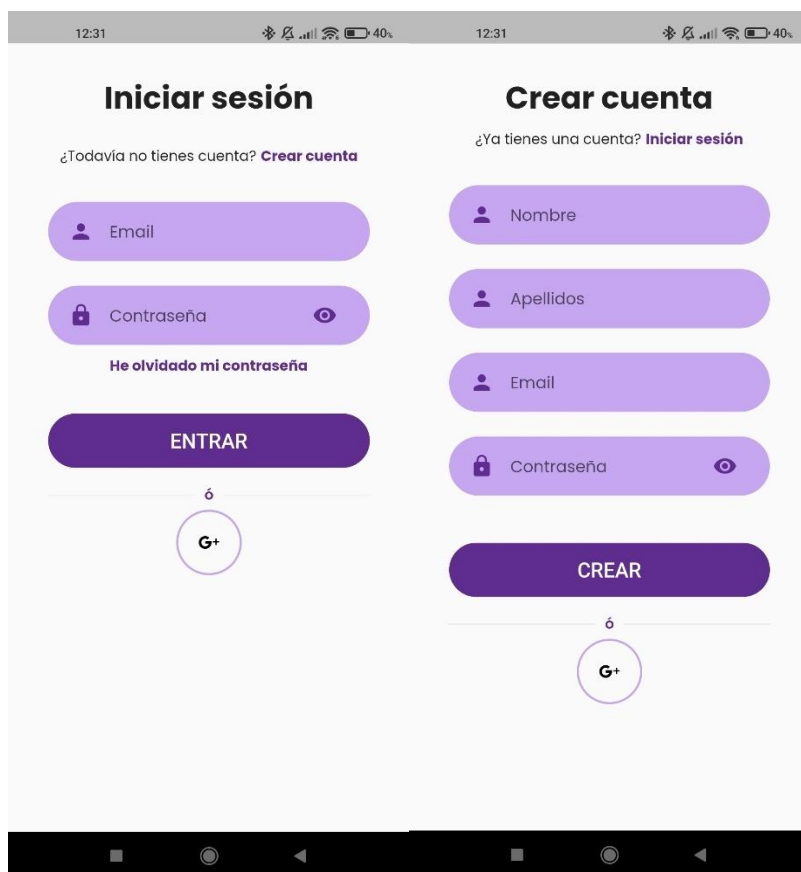
## 7. Anexos

### 7.1. Manual de usuario

- **Inicio de sesión y registro**

Estas pantallas aparecen cada vez que se inicia la aplicación sin estar autenticado previamente en ella. También aparece cuando es la primera vez que inicias la aplicación. Si el usuario ya dispone de una cuenta en la aplicación, podrá introducir su usuario y contraseña y pulsar en Entrar, o iniciar sesión con Google+ si es el método con el que se registró. Si el inicio de sesión es correcto, se redirige a la pantalla de búsqueda por dirección.

En cambio, si el usuario no dispone de una cuenta creada, podrá pulsar sobre el botón 'Crear cuenta' y rellenar los datos correspondientes para crear una nueva. Si los datos son correctos, se redirige al perfil del usuario para que lo complete.



- **Pantallas de búsqueda**

Esta pantalla es la primera que aparece tras iniciar sesión en la aplicación. En la parte inferior de la misma se puede apreciar el menú de la aplicación, que mientras navegamos por las pestañas de este, siempre aparecerá en la parte inferior de la aplicación.

Respecto a la pantalla de búsqueda, se nos permite buscar alojamientos por municipio, ciudad o código postal o en el mapa (Usando el botón superior de la pantalla). Para ello, se introducirá uno de los anteriores criterios en el cuadro de búsqueda y saldrán predicciones según escribimos. A continuación, se seleccionará un resultado deseado y se pulsará en BUSCAR. Con ello, se mostrarán los alojamientos disponibles con esos criterios.

En la pantalla de búsqueda por dirección también se nos permite buscar sin ningún criterio, pulsando directamente en el botón BUSCAR. Esto, nos devolverá todos los alojamientos disponibles en la aplicación.

En la búsqueda en el mapa, al introducir una localidad y seleccionarla en la lista que va apareciendo según escribimos, el mapa nos dirigirá a ese lugar. Si pulsamos sobre cualquier marcador morado (alojamientos publicados), aparecerá una ventana en la parte inferior con la información más relevante.

También, podremos añadir multitud de filtros pulsando en Filtros.

Por último, en la pantalla de búsqueda por dirección, podremos acceder de una manera rápida a la publicación de un alojamiento.

The image displays two side-by-side screenshots of a mobile application's search interface. The left screenshot, taken at 12:41, shows a search screen with a search bar containing the placeholder text 'Municipio, ciudad, CP...'. Below the search bar are three buttons: 'BUSCAR' (highlighted in purple), 'Filtros', and 'PON TU ANUNCIO'. The background of the search screen is a blurred image of a hotel room. The right screenshot, taken at 12:32, shows the search results screen. It features a search bar with the same placeholder text, a 'Filtros' button, and a 'PON TU ANUNCIO' button. The main content is a map view showing various locations and markers, including 'Hospital Universitario Rey Juan Carlos', 'Universidad Rey Juan Carlos', 'Centro de Salud Presentación Sabio', 'Parque Finca Liana', 'Móstoles', 'Hospital Universitario Móstoles Urgencias', 'Hospital Universitario HM Puerta del Sur', and 'El Eslabón'. The bottom navigation bar of the app is visible in both screenshots, with icons for search, home, add, chat, and profile.

51



- **Resultado búsqueda**

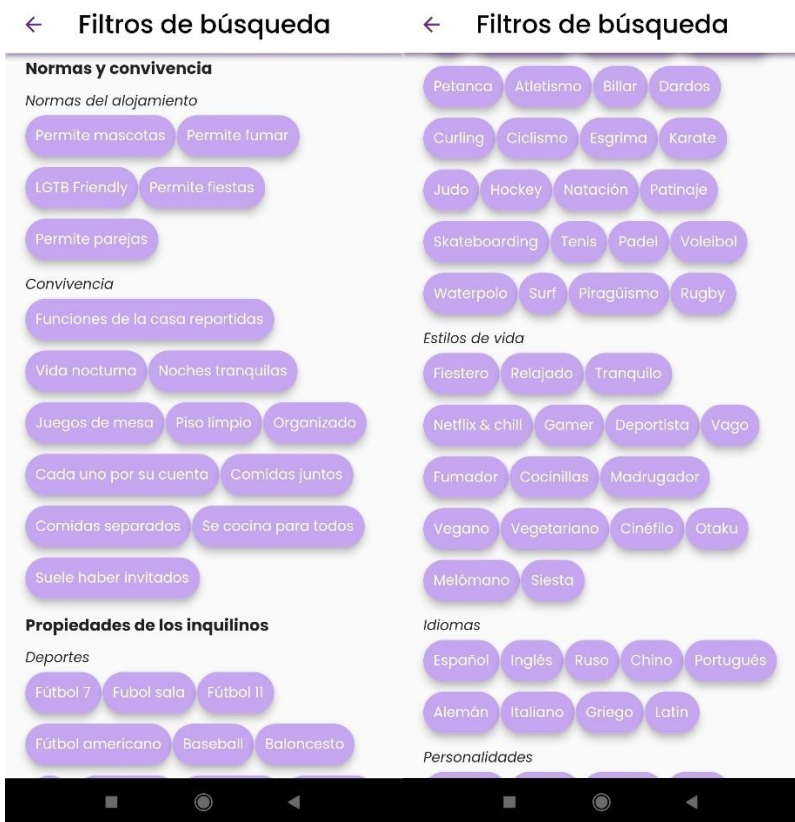
En esta pantalla podemos visualizar los resultados de la búsqueda que hayamos realizado en la pestaña de 'Buscar por dirección'. Pulsando sobre cualquiera de los resultados, accederemos a la visualización del alojamiento y todos sus detalles. También podremos marcar o desmarcar como favorito cualquiera de los anuncios pulsando sobre el icono del corazón de la parte superior derecha de cada anuncio.



- **Filtros de búsqueda**

Esta pantalla nos permite seleccionar multitud de filtros para afinar nuestra búsqueda. Para ello, seleccionaremos los filtros que deseemos y en la parte inferior de la pantalla pulsaremos en Aplicar Filtros y nos mostrará aquellos alojamientos que cumplan con los filtros seleccionados.

La flecha superior izquierda nos dirigirá a la pantalla de búsqueda desde donde hayamos pulsado el botón de filtros.

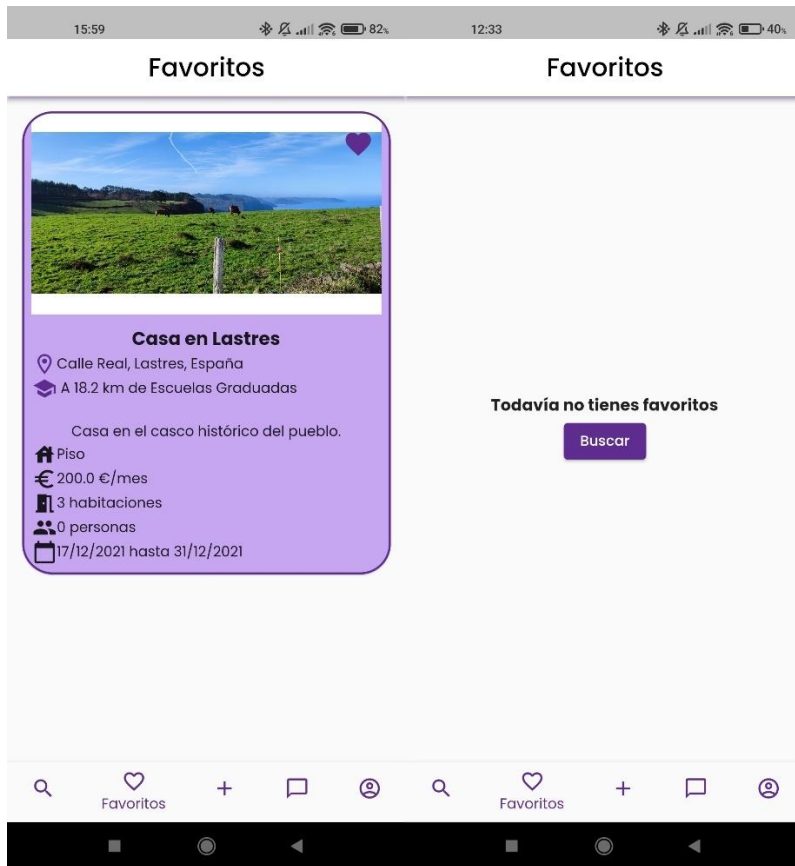




- **Favoritos**

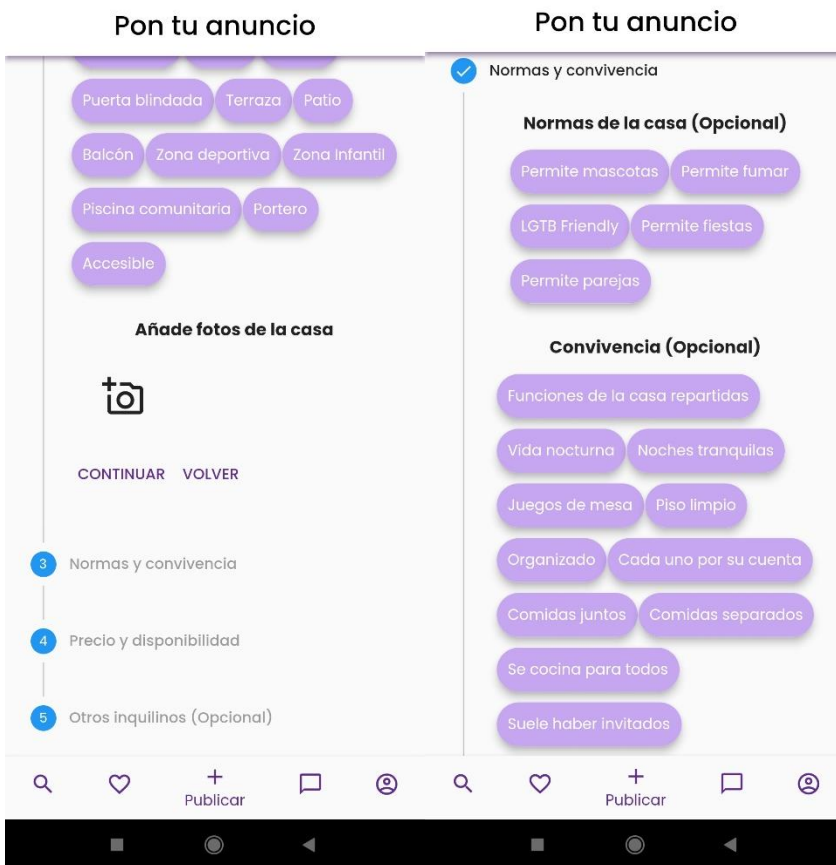
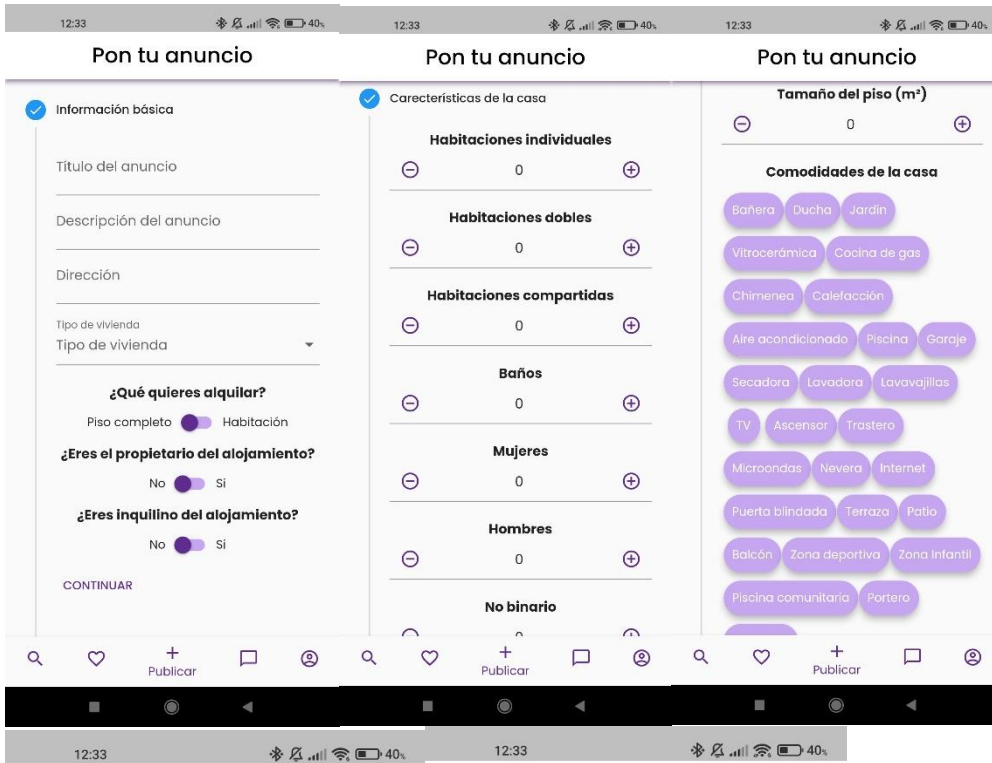
Esta pantalla muestra aquellos alojamientos que el usuario haya marcado como favoritos. En esta pantalla podemos acceder a cualquiera de esos alojamientos o desmarcarlos para que no vuelvan a aparecer en la pantalla.

La pantalla también puede mostrar un mensaje de 'Todavía no tienes favoritos' cuando no hayamos marcado ningún alojamiento como tal. Nos presentará un botón que nos llevará a la pantalla de búsqueda.

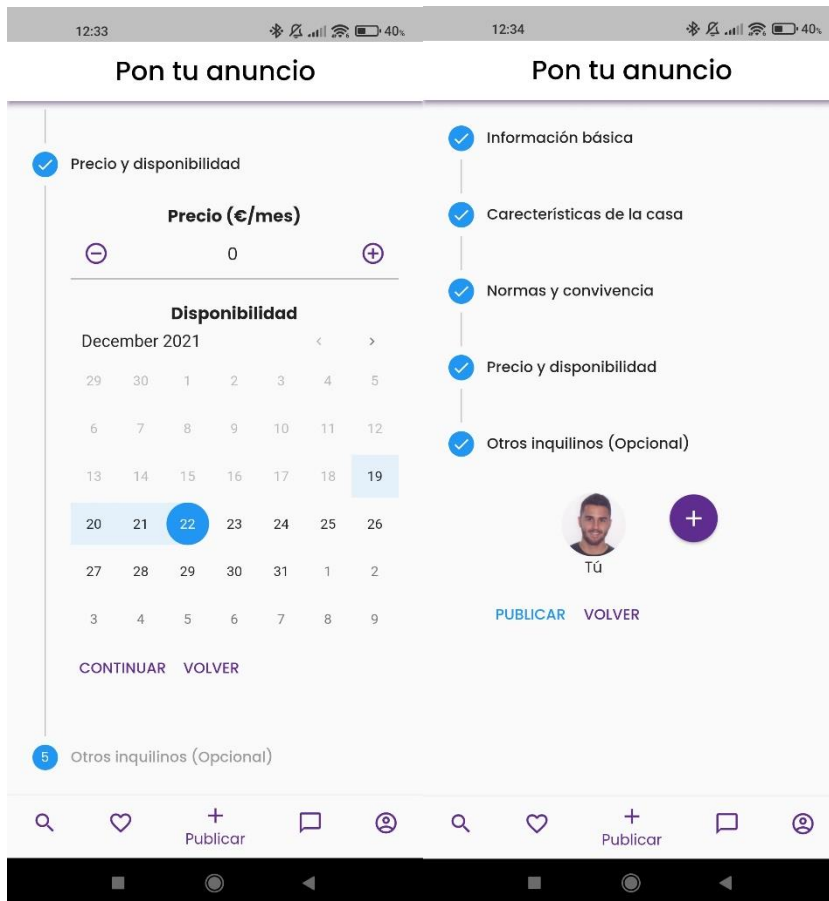


- **Publicar un anuncio**

En esta pantalla se nos permite configurar y publicar un nuevo anuncio en la aplicación. Para ello, habrá que rellenar todos los datos obligatorios, que son aquellos que no tienen la etiqueta de "(Opcional)". Al terminar de rellenar todos los datos, se publicará el alojamiento y aparecerá un mensaje de que se ha creado correctamente. En caso contrario, el anuncio no se creará y se mostrará un mensaje indicando que hay que rellenar todos los datos obligatorios.



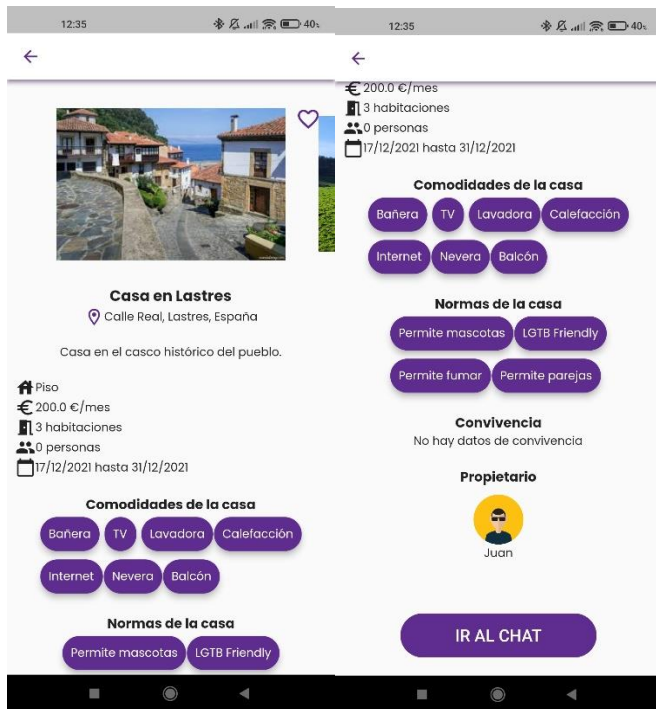




- **Detalle del anuncio**

En esta pantalla se muestran todos los detalles del anuncio que hayamos seleccionado. Podemos marcar el anuncio como favorito, acceder al perfil tanto del propietario como de los inquilinos (si los tiene) y contactar con el inquilino en caso de que el anuncio no sea nuestro.

Si pulsamos sobre la flecha de la parte superior izquierda, volveremos a la pantalla desde donde accedimos al detalle del anuncio.



- **Chats**

Esta pantalla es la cuarta opción del menú inferior. En ella se encuentran los chats activos que el usuario tenga. En caso de no tener ningún chat activo, aparecerá un mensaje indicando la situación.

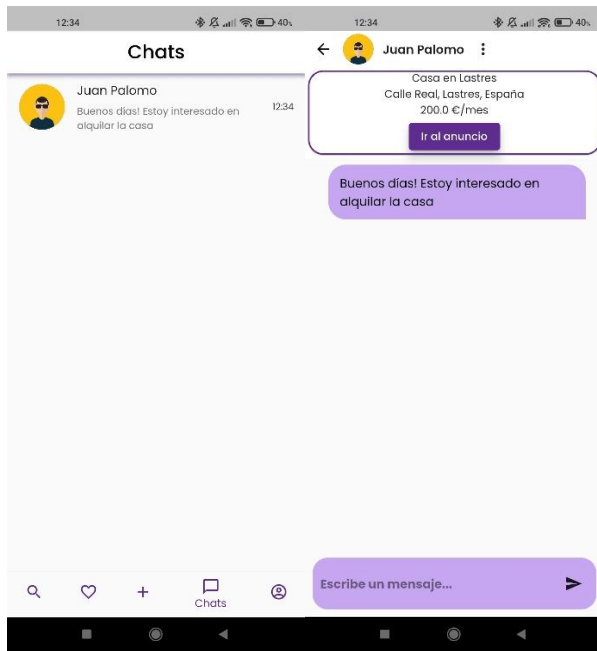
Si pulsamos sobre un chat, se mostrará el interior del chat, con todos los mensajes que se hayan mandado a lo largo de la conversación.

Desde dentro del chat, podemos acceder tanto al usuario (pulsando sobre el nombre o la foto en la parte superior de la pantalla) como al anuncio por el que se ha contactado.

En la parte inferior de la pantalla, se puede pulsar para comenzar a escribir un mensaje y pulsando en el icono de la flecha, se enviará el mensaje al otro usuario y saldrá reflejado en la conversación.

Por último, en la parte superior a continuación del nombre, se aprecian 3 puntos verticales, con los cuales, si los pulsamos, aparecerá un mensaje emergente con el que podremos borrar el chat. Esta acción, borrara el chat para las dos personas involucradas.

Si pulsamos sobre el icono de la flecha de la parte superior izquierda, volveremos a la pantalla de la lista de los chats activos.



- **Cuenta**

Esta pantalla es la quinta y última opción del menú inferior de la aplicación. En ella se puede acceder al resto de funcionalidades principales de la aplicación, tanto las presentes en el menú inferior, como algunas funciones adicionales como editar el perfil y ver los alojamientos publicados.

También, podemos cerrar la sesión o borrar la cuenta definitivamente.



- **Mis anuncios publicados**

En esta pantalla se muestran todos los alojamientos que hayamos publicados. En caso de no haber publicado ninguno, aparecerá un mensaje indicando la situación.

Si tenemos pisos publicados, podemos editarlos o borrarlos.

A esta pantalla se accede desde la pantalla de “Cuenta” y pulsando sobre la opción “Mis alojamientos publicados”. Si pulsamos sobre la flecha superior izquierda, volveremos a la pantalla de “Cuenta”.




- **Perfil de otro usuario**

A esta pantalla podremos acceder desde dos puntos: Desde los detalles de un alojamiento en la sección “Inquilinos” o “Propietario”, o desde dentro de un chat concreto, pulsando sobre la imagen o el nombre del usuario con el que estamos hablando.

Esta pantalla es informativa y no se pueden realizar acciones sobre ella. En ella se muestra toda la información que el usuario haya rellenado.

13:00
72%
13:00
72%

← **Álvaro de Porres**



**Álvaro de Porres**  
Propietario e inquilino

**21 años**  
**Hombre**

**Grado en Ingeniería del Software**

*Deportes que practico*

Fútbol 7   Fútbol sala   Fútbol 11

Ski

*Mi personalidad*

*Mi estilo de vida*

*Mis idiomas*


*Acerca de mí*

Me gusta salir de fiesta y traer gente a casa para pasar un rato jugando a la consola o lo que surja

En alquiler (2)

← **Álvaro de Porres**

En alquiler (2)



**Piso en Móstoles muy cerca de la universidad**

📍 Calle Alonso Cano, Móstoles, España  
A 3.1 km de Universidad Europea de Madrid (Campus Villaviciosa)

El piso se encuentra muy cerca de renfe y metro y de la universidad

🏠 Piso  
€ 250.0 €/mes  
🛏 4 habitaciones  
👤 1 chica y 2 chicos  
📅 15/1/2022 hasta 30/1/2022

