

# Documentación del proyecto integrado



Iván Fernández Méndez

CFGS Desarrollo de Aplicaciones Multiplataforma

# **Índice**

<b>1- Introducción.....</b>	<b>4</b>
<b>2- Estudio de Viabilidad.....</b>	<b>4</b>
<b>2.1- Descripción del Sistema Actual.....</b>	<b>4</b>
<b>2.2- Descripción del Nuevo Sistema.....</b>	<b>5</b>
<b>2.3- Identificación de Requisitos del Sistema.....</b>	<b>6</b>
<b>2.3.1- Requisitos de información.....</b>	<b>6</b>
<b>2.3.2 Requisitos de funcionalidad.....</b>	<b>9</b>
<b>2.4- Descripción de la Solución.....</b>	<b>11</b>
<b>2.5- Planificación del Proyecto.....</b>	<b>12</b>
<b>2.5.1- Equipo de trabajo.....</b>	<b>12</b>
<b>2.6.- Estudio del coste del proyecto.....</b>	<b>13</b>
<b>3- Análisis del sistema de información.....</b>	<b>14</b>
<b>3.1- Identificación del entorno tecnológico.....</b>	<b>14</b>
<b>3.2.2- Esquema de la Base de Datos.....</b>	<b>15</b>
<b>3.3- Identificación de Usuarios Participantes y Finales.....</b>	<b>15</b>
<b>Los usuarios pensados son:.....</b>	<b>15</b>
<b>3.4- Identificación de subsistemas de análisis.....</b>	<b>16</b>
<b>3.5- Establecimiento de requisitos.....</b>	<b>17</b>
<b>3.6- Diagrama de Análisis.....</b>	<b>20</b>
<b>3.7 Definición de interfaces de usuario.....</b>	<b>20</b>
<b>3.7.1 Especificación de principios generales de interfaz.....</b>	<b>21</b>
<b>Paleta de colores.....</b>	<b>21</b>
<b>Tipografía.....</b>	<b>21</b>
<b>Logotipo.....</b>	<b>22</b>
<b>3.7.2 Especificación de formatos individuales de la interfaz de pantalla.....</b>	<b>23</b>
<b>3.7.3 Identificación de perfiles de usuarios.....</b>	<b>44</b>
<b>3.7.4.- Especificación de formatos de impresión.....</b>	<b>44</b>
<b>3.7.5.- Especificación de navegabilidad entre pantallas.....</b>	<b>45</b>
<b>4 Construcción del Sistema.....</b>	<b>46</b>
<b>5 Conclusión.....</b>	<b>49</b>
<b>6 Glosario de términos.....</b>	<b>51</b>
<b>7 Bibliografía.....</b>	<b>52</b>

## *Lista de Cambios*

Fechas	Cambios
10/4/23	Se opta por quitar el atributo descripción de la tabla categoría ya que no es necesario y se cambia el atributo categoría de la tabla publicación por una lista para así poder tener dos categorías por cada publicación
20/4/23	Se añade la funcionalidad de seguir a otros usuarios
26/4/23	Se añaden 2 tablas más para, la tabla like para guardar los datos de que persona le han dado like a que publicación y la tabla sigue que guarda que usuario sigue a qué usuario
5/5/23	Se añaden el campo baneado a las tablas publicación, usuario y comentario para que sirve de identificador para identificar si están baneados
12/5/23	Se ha añadido el campo nickname en la tabla comentario para ganar en eficiencia en la aplicación
2/6/23	Se decide añadir la funcionalidad de buscar usuarios

## *1- Introducción*

El proyecto que se plantea en esta documentación trata acerca de una aplicación para compartir recetas. Esta se compone de varias funcionalidades:

- Servicio de autentificación: para guardar datos relevantes y publicar recetas
- Apartado de búsqueda: por texto y por categorías
- Publicación de nuevas recetas

Tecnologías y lenguajes pensados para utilizar:

- Android Studio como IDE para desarrollo en móviles
- Kotlin como lenguaje de programación
- Firebase como herramienta de gestión de bbdd, autenticación, y almacén de archivos

## *2- Estudio de Viabilidad*

### *2.1- Descripción del Sistema Actual*

El sistema actual para buscar recetas y compartirlas se basa en recurrir a foros o webs de tops.

Estas páginas tienen varios inconvenientes. La búsqueda de una receta no es eficiente y puedes llegar a necesitar contrastar varias recetas para verificar si no hay errores, o para encontrar la receta que quieras exactamente. Esto requiere un mayor esfuerzo para el usuario y hace que no sea común consultar recetas de esta forma.

Los problemas de búsqueda repercuten directamente en la publicación de recetas. A menos público para compartir menos interés en publicar.

## *2.2- Descripción del Nuevo Sistema*

Creo que la solución a los problemas del sistema actual pueden resolverse y así proporcionar una manera más cómoda e intuitiva para ver y publicar recetas

### *¿Por qué una aplicación de móvil?*

Hoy en día es más común tener un smartphone que un ordenador. Gracias a que son más cómodos y su curva de aprendizaje es menos severa.

Estas características hacen al smartphone la mejor opción para distribuir esta idea y así llegar a más gente.

## *2.3- Identificación de Requisitos del Sistema*

### *2.3.1- Requisitos de información*

Mis requisitos de información están clasificados en colecciones ya que así es como trabaja firebase

Usuario		
Descripción	Esta colección se encarga de guardar los datos de los usuarios registrados en la aplicación así como algunas estadísticas del usuario como, por ejemplo, el número de seguidores, seguidos y publicaciones	
Lista de campos	<u>id_usuario</u>	VARCHAR
	nickname	VARCHAR
	nombre	VARCHAR
	biografia	VARCHAR
	admin	BOOLEAN
	num_seguidores	INT
	num_siguiendo	INT
	num_publicacion	INT
	baneado	BOOLEAN
Volumen de información	No hay un límite de usuarios definido	

Categoría		
Descripción	Esta colección se encarga de almacenar los datos de las categorías de las recetas	
Lista de campos	id_categoria	VARCHAR
	nombre	VARCHAR
Volumen de información	No hay un límite de categorías definido	

Publicaciones		
Descripción	Esta colección se encarga de almacenar los datos de las publicaciones de recetas hechas por los usuarios	
Listas de campos	<u>id_publicacion</u>	VARCHAR
	titulo	VARCHAR
	descripcion	VARCHAR
	ingredientes	ARRAY(VARCHAR)
	pasos	ARRAY(VARCHAR)
	imagen	VARCHAR
	num_likes	INT
	id_usuario	VARCHAR
	id_categoria	ARRAY(VARCHAR)
	baneado	BOOLEAN
Volumen de	No hay un límite de recetas definido	

información	
-------------	--

Comentario		
Descripción	Esta colección se encarga de almacenar los datos de los comentarios de las publicaciones hechos por los usuarios	
Listas de campos	<u>id_comentario</u>	VARCHAR
	texto	VARCHAR
	id_usuario	VARCHAR
	id_publicacion	VARCHAR
	baneado	BOOLEAN
	nickname	VARCHAR
Volumen de información	No hay un límite de comentarios definido	

### **2.3.2 Requisitos de funcionalidad**

<b>Registro de nuevos usuarios</b>	
<b>Descripción</b>	<b>La aplicación permitirá registrar un nuevos usuarios</b>

<b>Inicio de sesión automático de usuarios</b>	
<b>Descripción</b>	<b>La aplicación permitirá iniciar sesión automáticamente a un nuevos usuarios al ejecutar la aplicación</b>

<b>Búsqueda de recetas</b>	
<b>Descripción</b>	<b>La aplicación permitirá buscar recetas ordenadas por el número de likes</b>

<b>Publicación de recetas</b>	
<b>Descripción</b>	<b>La aplicación permitirá publicar nuevas recetas</b>

<b>Ver el perfil</b>	
<b>Descripción</b>	<b>La aplicación permitirá registrar un nuevos usuarios</b>

<b>Editar el perfil</b>	
<b>Descripción</b>	<b>La aplicación permitirá editar la mayoría de los campos de tu perfil</b>

<b>Dar me gusta a una publicación</b>	
<b>Descripción</b>	<b>La aplicación permitirá puntuar con un like a las publicaciones que te gusten</b>

<b>Seguir a un usuario</b>	
<b>Descripción</b>	<b>La aplicación permitirá poder seguir a otros usuarios</b>

<b>Buscar Usuarios</b>	
<b>Descripción</b>	<b>La aplicación permitirá realizar búsquedas de usuarios</b>

### **2.3.3- Otros Requisitos**

#### **Sistema operativo**

El smartphone deberá usar el sistema operativo android en su versión 7.0 Nougat (API 24)

#### **Conectividad**

El dispositivo deberá contar con una conexión a internet

## **Permisos**

El usuario deberá aceptar los permisos necesarios para que la aplicación funcione

### ***2.4- Descripción de la Solución***

La característica principal de mi solución propuesta es que sea lo más intuitiva posible para así diferenciarse del sistema actual y ser más atractiva y cómoda para el usuario.

Para lograrlo, la aplicación estará diseñada para dispositivos móviles. El framework que utilizaré, como ya he explicado anteriormente, es Android Studio ya que la mayoría de dispositivos móviles usan este sistema operativo. La aplicación en varias páginas con las que se podrá acceder a las distintas funcionalidades expuestas anteriormente a través de botones en la parte inferior.

La base de datos que utilizaré será Firestore. Este es uno de los módulos que tiene firebase y que permite almacenar información en una base de datos noSQL. Otros módulos que utilizaré serán:

- Authentication: se encarga de la autenticación y seguridad del usuario. Permite implementar varios métodos de log in y todo el guardado y encriptado de contraseñas lo hace Firebase internamente de manera muy cómoda.
- Storage: permite subir archivos a la nube y acceder a ellos mediante enlaces. Esta herramienta se usará para las imágenes de las recetas.

## *2.5- Planificación del Proyecto*

### *2.5.1- Equipo de trabajo*

La aplicación tiene como objetivo servir de proyecto de fin de cursos para CFGS Desarrollo de Aplicaciones Multiplataforma impartido en el I.E.S. Julio Verne de Sevilla.

El equipo de trabajo estará conformado por una única persona, quien se encargará de la documentación del proyecto, el desarrollo del mismo y su defensa, que en este caso correrá a cargo Iván Fernández Méndez.

### *2.5.2- Planificación temporal*

El desarrollo del proyecto tiene un plazo fijo de 3 meses para finalizarlo. Este desarrollo estará dividido en plazos que servirán de control para controlar el progreso del mismo. Estos plazos son:

Plazo	Tarea
31/03/2023	Análisis funcionales y diseño de la aplicación
09/04/2023	Desarrollo de la documentación
23/04/2023	Diseño e implementación de la bbdd
06/05/2023	Desarrollo de la página de perfiles y de registro e inicio de sesión del usuario
21/05/2023	Desarrollo de la página de visualizado de recetas y publicación
04/06/2023	Desarrollo de la página de búsqueda

## *2.6.- Estudio del coste del proyecto*

La aplicación al ser destinada como proyecto integrado carece de coste monetario como tal. Sin embargo, el objetivo de esta aplicación es de hacer de red social de recetas y que esté subido a algún servidor de descargas para móviles como google play, necesitaré pagar 25 dólares (que al cambio a euros son 20.75€) para darme de alta como desarrollador.

A este precio se le deberá de sumar las tarifas de firebase ya que si quiero que mi aplicación sea de uso público llegará un momento que agote el almacenamiento que ofrece de manera gratuita. Debido a que este no lo voy a agotar hasta pasado un tiempo, he decidido no incluirlo en el precio final.

El precio total sería de 20.75€

### 3- Análisis del sistema de información

#### 3.1- Identificación del entorno tecnológico

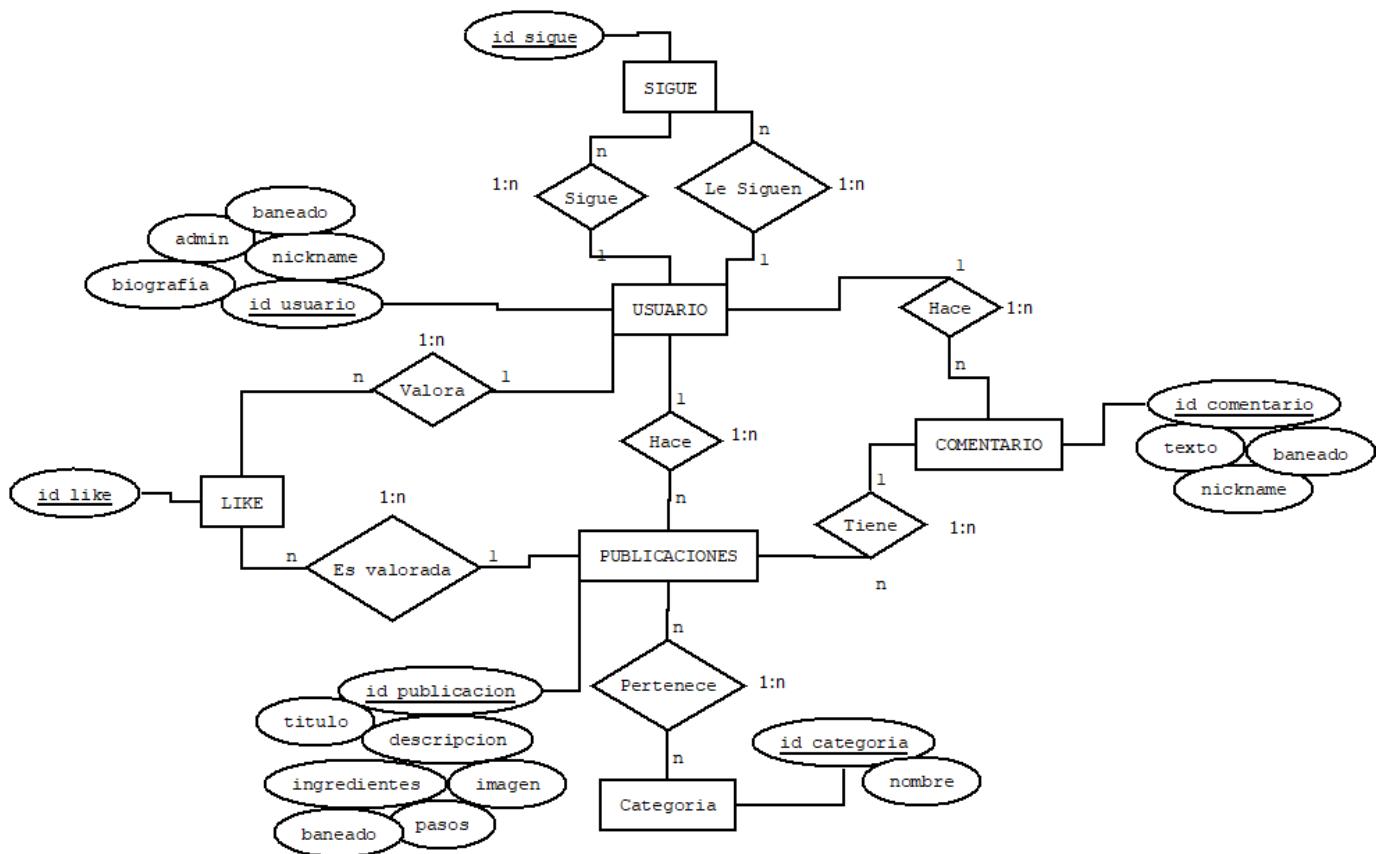
Para el desarrollo solo utilizaré un ordenador de sobremesa con las siguientes características:

- Sistema Operativo: Windows 11
- RAM: 16 GB
- Procesador: Ryzen 5
- Disco duro: SSD de 512GB
- Tarjeta Gráfica: Nvidia 1650 GTX

Para la ejecución de la aplicación y test usaré un Xiaomi Redmi 11 Lite con Android 13 y MIUI 14

#### 3.2- Modelo de Datos

##### 3.2.1- Modelo Entidad-Relación



### *3.2.2- Esquema de la Base de Datos*

En la consola de firestore todo se hace por interfaz gráfica por lo que no hay un script como tal de ejecución.

### *3.3- Identificación de Usuarios Participantes y Finales*

Los usuarios pensados son:

- *Usuario Sin Registrar*

Funciones:

- Registrarse

- *Usuario Registrado*

Funciones:

- Iniciar Sesión
- Ver publicaciones
- Ver perfiles de usuarios
- Publicar
- Editar su perfil

- *Usuario Administrador*

Funciones: Todas las de un usuario registrado más la capacidad de borrar cuentas y publicaciones

### **3.4- Identificación de subsistemas de análisis**

Los subsistemas de mi aplicación son los siguientes:

- ***Gestión de inicio de sesión y registro***

Integra todas las funciones de registro e inicio de sesión tales como la creación de nuevos usuarios. Este está relacionado con el subsistema de perfiles debido a que para mostrar un perfil primero deberá estar autenticado

- ***Gestión de visualizado de publicaciones***

Engloba las funciones para mostrar listas de publicaciones y la página de la misma publicación. Así como también las funciones para interactuar con la publicación. Está conectado al subsistemas de búsqueda de publicaciones, las publicaciones encontradas harán uso de este subsistema para mostrarse en la interfaz del usuario.

- ***Gestión de búsqueda de publicaciones***

Comprende todas las funciones de búsqueda de publicaciones. Hace uso del subsistema de visualización de publicaciones para mostrar los resultados de la búsqueda

- ***Gestión y creación de nuevas publicaciones***

Integra todas las funciones de creación de una nueva receta. Para lograrlo hace uso del subsistema de perfiles para adjuntar datos a la publicación y relacionarla con un usuario que será el autor

- *Gestión de perfiles*

Este subsistema engloba toda la gestión del perfil, tanto la edición como la visualización de este. Está relacionado con los subsistemas de inicio de sesión y registro y creación de publicaciones.

- *Gestión de seguimiento de usuarios*

Integra todas las gestiones de la funcionalidad de seguir a otro usuario. Este subsistema hace uso del subsistema de perfiles para relacionar los 2 usuarios

- *Gestión de búsqueda de usuarios*

Engloba las funcionalidades de búsqueda de usuarios y hace uso de la gestión de perfiles

### **3.5- Establecimiento de requisitos**

#### ***Gestión de inicio de sesión y registro***

Siempre que se quiera acceder a la aplicación te pedirá que te identifiques para ello hay 3 métodos

- **Registro por email y contraseña:** Permitirá a los usuarios registrarse con una cuenta de email y una contraseña.
- **Registro e inicio de sesión por cuenta de google:** Esta será la manera más rápida de identificarse. Hará uso de una de tus cuentas de google que estén con la sesión iniciada y no pedirá nada más
- **Inicio de sesión por email y contraseña:** Pedirá un email y una contraseña y comprobará si los credenciales son correctos en tal caso te autorizará la entrada a la aplicación con el usuario autenticado

### *Gestión de visualizado de publicaciones*

- **Visualización de listas de publicación:** Permitirá visualizar una serie de datos de publicaciones que al tocar en ellos te llevarán a la página de esta publicación
- **Visualizado de la página de la publicación:** Mostrará todos los datos de la publicación y ofrecerá la oportunidad de comentar y de dar like
- **Dar like/ quitar like:** Permite incrementar a 1 el contador de likes de la publicación si esta no tenía el like del usuario anteriormente dado y viceversa a la hora de quitarlo.
- **Comentar:** Los usuarios harán un comentario a la publicación

### *Gestión de búsqueda de publicaciones*

- **Búsqueda por título:** Se podrá buscar una receta a través de un fragmento del nombre del título.
- **Búsqueda por categoría:** Permite buscar las recetas que pertenezcan a una categoría en concreto

### *Gestión y creación de nuevas publicaciones*

- **Creación de publicaciones:** Los usuarios podrán crear una nueva publicación con los datos que quieran
- **Edición de una publicación:** Permitirá al propietario de una publicación eliminarla
- **Eliminación de publicaciones:** Dará la capacidad solo a los administradores de borrar las publicaciones ajenas y a los usuarios normales solo las propias

## *Gestión de perfiles*

- **Visualización de perfiles:** Los usuarios podrán ver los perfiles de otros usuarios y los suyos
- **Edición de perfiles:** Los propietarios de un perfil tendrán la capacidad de editar su perfil
- **Eliminación de un perfil:** Los administradores podrán borrar cualquier perfil y los usuarios normales solo los propios

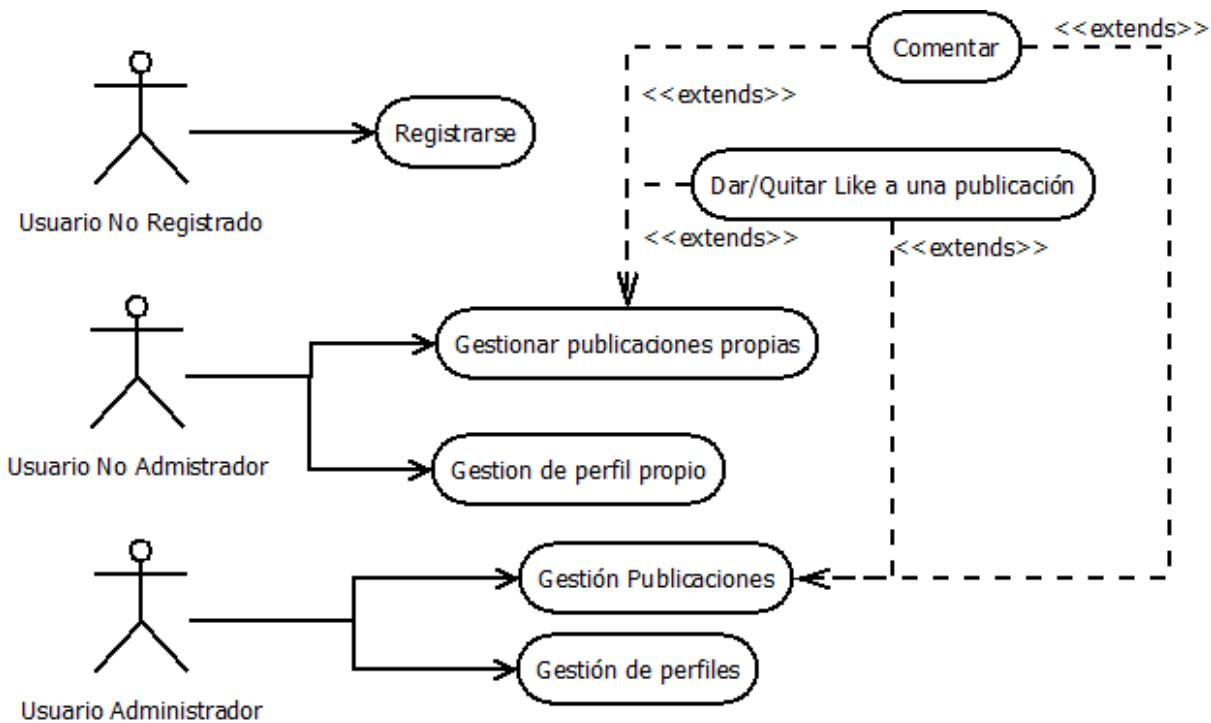
## *Gestión de seguimiento*

- **Listar publicaciones de usuarios que sigues:** Los usuarios podrán ver las publicaciones más populares de las personas a las que siguen
- **Búsqueda por categoría:** se podrá buscar una receta entre las de los usuarios a los que sigues a través de una categoría
- **Búsqueda por título:** se podrá buscar una receta entre las de los usuarios a los que sigues a través de un fragmento del título

## *Gestión de búsqueda de usuarios*

- **Se listará los usuarios por número de seguidores**
- **Se podrá buscar un usuario por un fragmento de su nickname**

### 3.6- Diagrama de Análisis



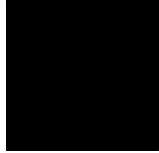
### 3.7 Definición de interfaces de usuario

La interfaz de usuario está pensada con un diseño simple, minimalista y con colores sólidos. Intenta imitar un ambiente de cocina.

También en cuanto a utilidad se han tomado muchas decisiones para que sea fácil de entender y no tanto para la estética. Esto es debido a que mucha gente que utilizará la app son muy adultos por lo que no se pueden utilizar un diseño de interfaz tan simbólico como instagram, tik tok u otras apps de redes sociales ya que cabe la posibilidad de que no lo entiendan. Es por esto que muchos iconos se han cambiado por texto.

### **3.7.1 Especificación de principios generales de interfaz**

#### **Paleta de colores**

 #FCF2DB	Este es el color que se utiliza en el fondo de toda la aplicación y para el color de texto de los elementos de fondo negro
 #000000	Se utiliza para el fondo y el texto de los elementos principales. Como botones y cajas de texto
 #C39800	Se utiliza para marcar los elementos seleccionados
 #FF0000	Se utiliza para indicar que un usuario, una publicación o un comentario está baneado

#### **Tipografía**

Las fuentes utilizadas en el proyecto son las mismas que se han utilizado en este documento. Estas son:

- **Pacifico:** Para señalar tanto los títulos de las publicaciones como los nicknames de los usuarios
- Roboto: Se utiliza para el resto del texto de la aplicación

*Logotipo*



### **3.7.2 Especificación de formatos individuales de la interfaz de pantalla**

La pantallas de la aplicación se puede clasificar en 4 partes:

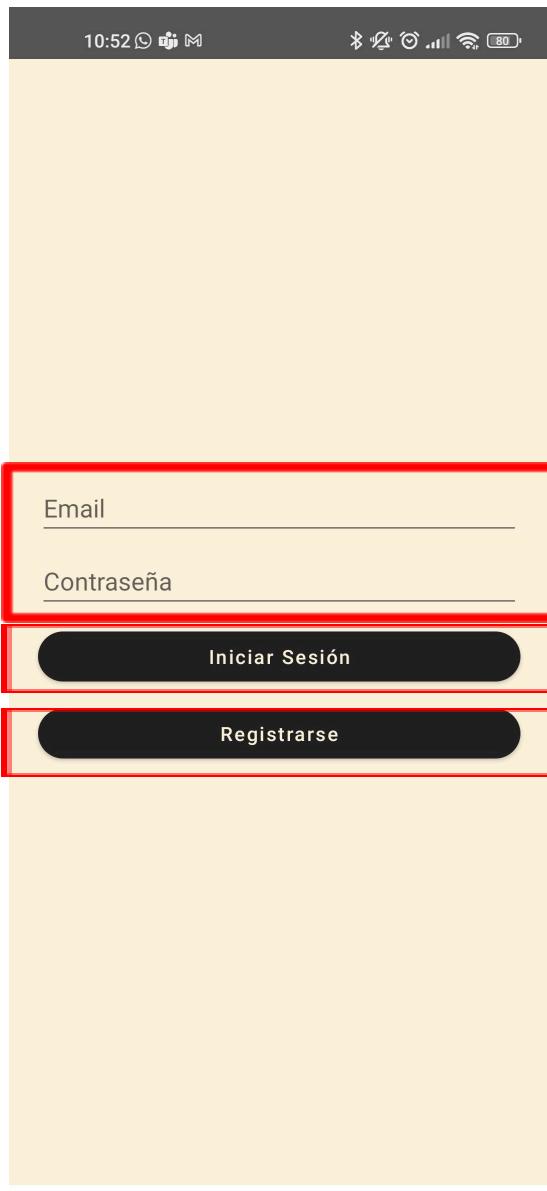
#### **1 Autentificación**

- Pantalla “no autenticado”: Pantalla que aparece al iniciarse la aplicación y si detecta que no hay ningún usuario logueado en la aplicación



- Botón que te lleva a la pantalla iniciar sesión
- Botón que te lleva a la pantalla de registro
- Botón para el login por google

- Pantalla “Login”: Pantalla para iniciar sesión con una cuenta de email y contraseña



Formulario de autentificación para el inicio de sesión

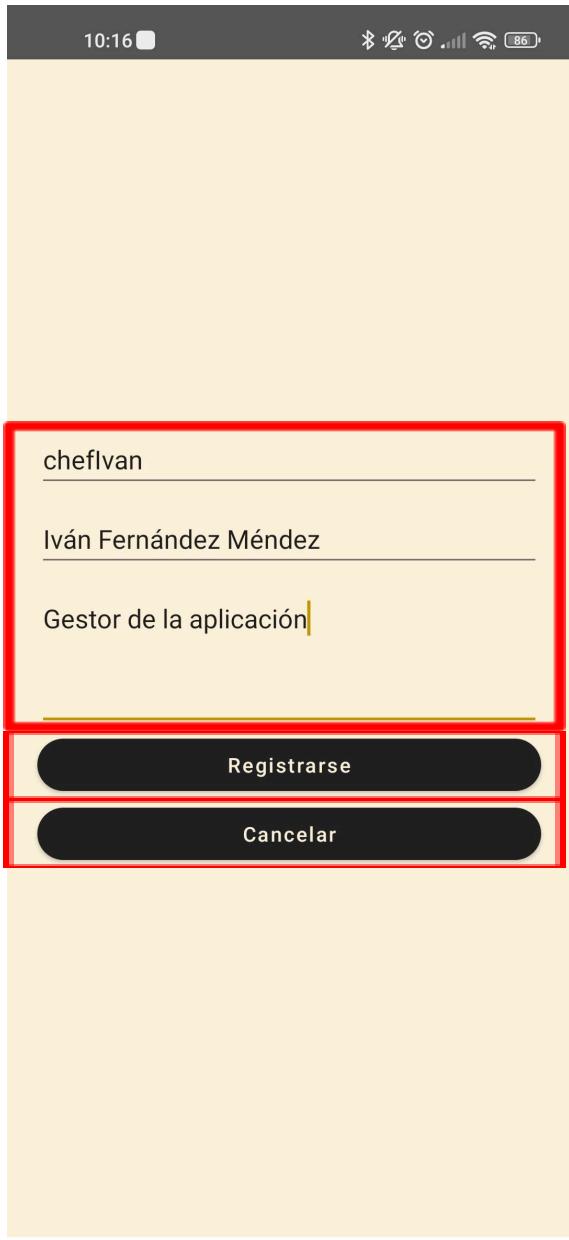
Botón para iniciar sesión con los datos del formulario

Botón que te lleva la pantalla de registro

- Pantalla “Registro”: Pantalla para registrarse en la aplicación por usuario y contraseña



- Pantalla “Registro con google”: Pantalla para introducir los datos necesarios de un nuevo usuario, esta pantalla solo sale cuando el usuario que ha iniciado sesión por google no está registrado en la base de datos



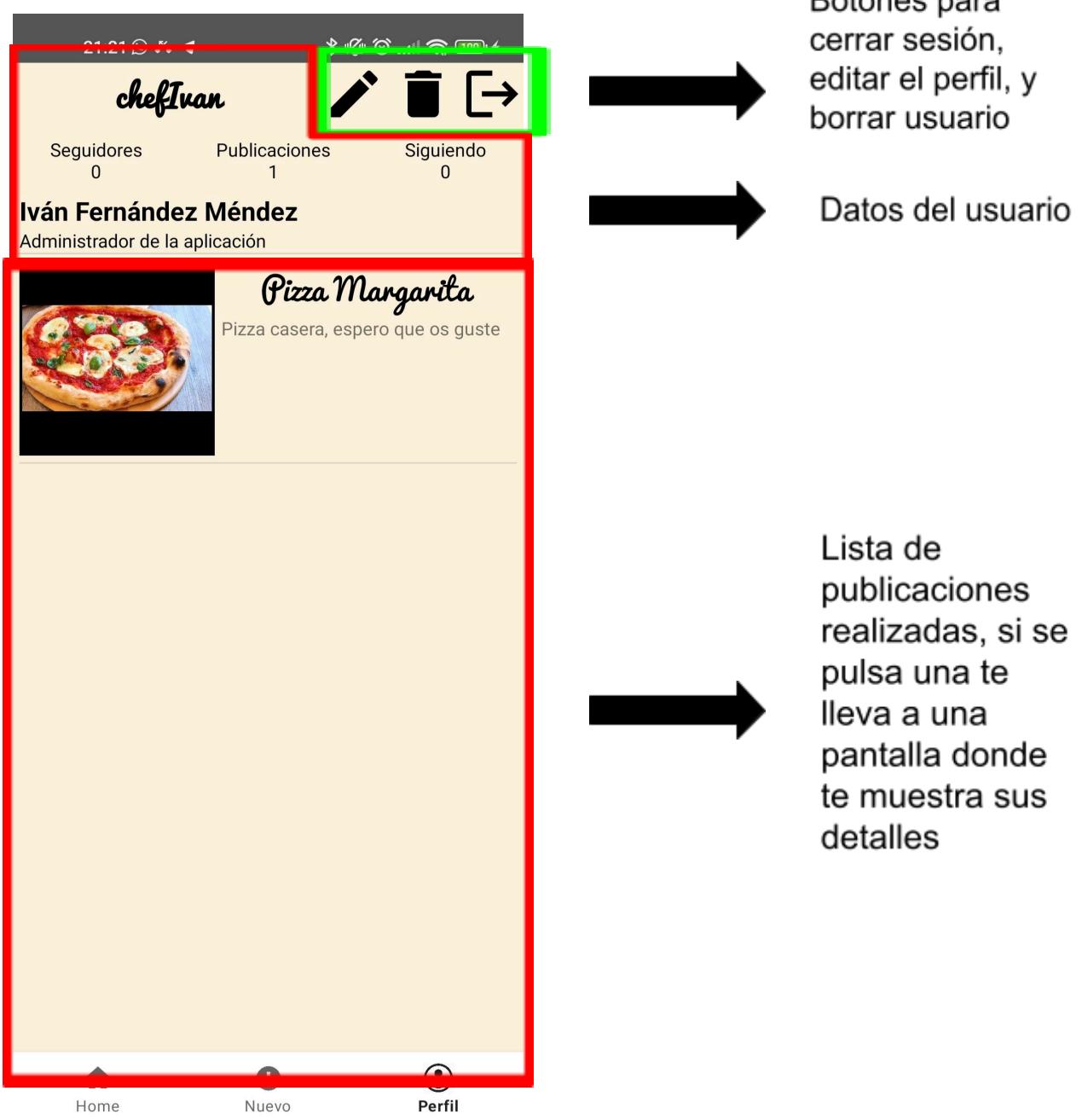
Formulario para crear un usuario por cuenta de google

Botón para crear un usuario con los datos del formulario

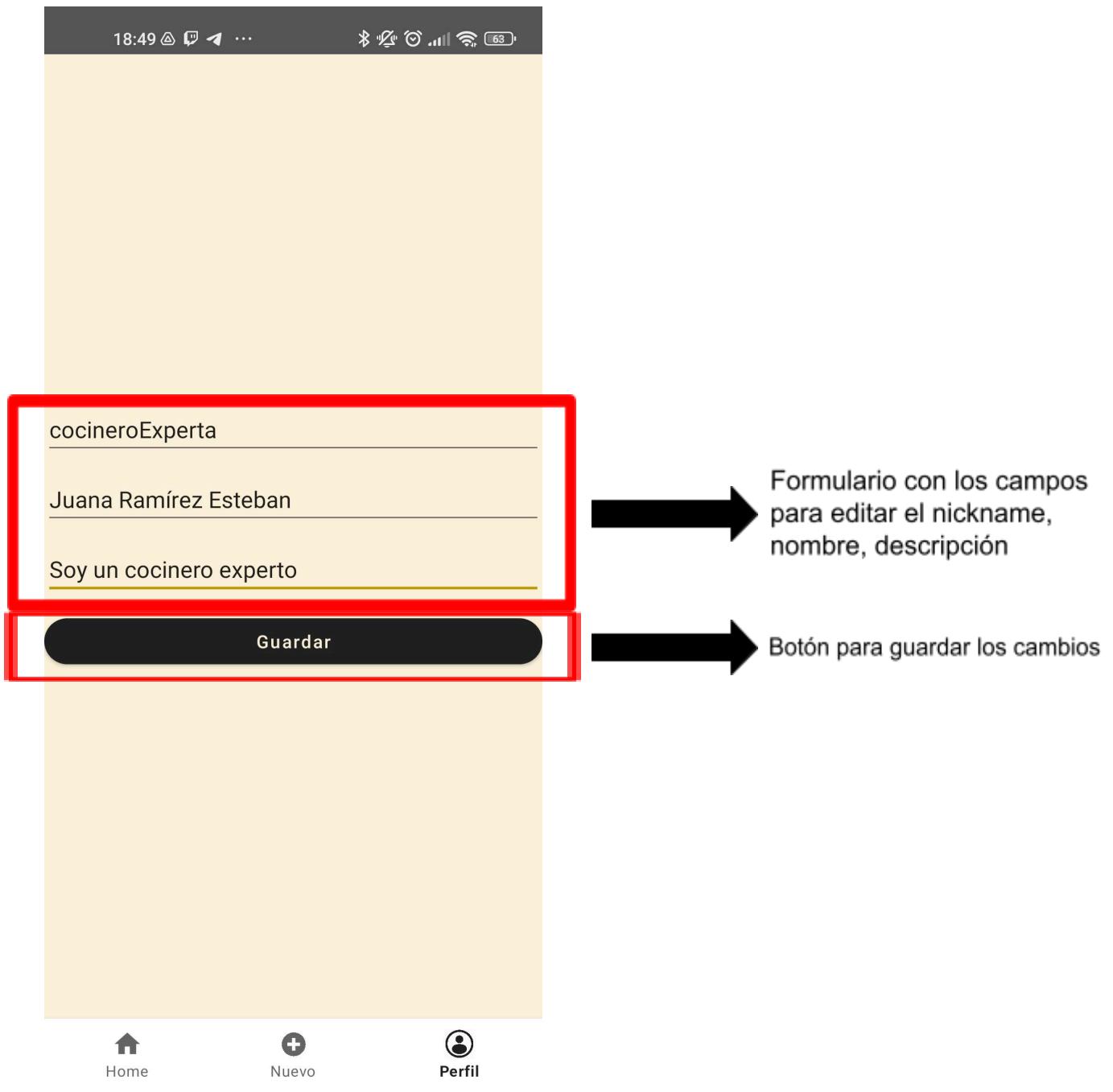
Botón para cancelar la operación

## 2 Perfil

- Pantalla “Perfil”: Pantalla en la que se visualiza tu perfil que permite gestionar tu usuario



- Pantalla “Editar”: Pantalla que te permite editar parte de tu usuario



### 3 Crear una nueva publicación

- Pantalla “Nueva publicación”: Esta es la pantalla que hace de formulario para crear una nueva receta. Esta pantalla es tan grande que para que se muestre bien se ha dividido en 2, en la aplicación están unidas.



**Ingrediente:**

Masa de pizza

Salsa de tomate

Mozarela

Albahaca (Opcional)

Agregar Ingrediente

**Pasos:**

Añade a la masa la salsa y el queso

Añade siquieres la albahaca y al horno durante 15 minutos

Agregar Pasos

Publicar

Lista de ingredientes que se puede aumentar, al lado hay un botón para quitarlo de la lista

Botón para agregar un nuevo ingrediente

Lista de pasos que se puede aumentar, al lado hay un botón para quitarlo de la lista

Botón para agregar un nuevo paso

Botón para publicar

Home

Nuevo

Perfil

## 4 Mostrador de publicaciones

- Pantalla “listado publicaciones”

Esta pantalla tiene 2 versiones, una que muestra las publicaciones más populares y otra que solo muestra las publicaciones de los usuarios a los que sigues.



## Versión siguiendo



Gestor de pestañas para ver las publicaciones más populares o las que sigues

Buscador por título, botón que te lleva a la pantalla de listado de usuarios

Desplegable con las categorías y botón para buscar por la seleccionada

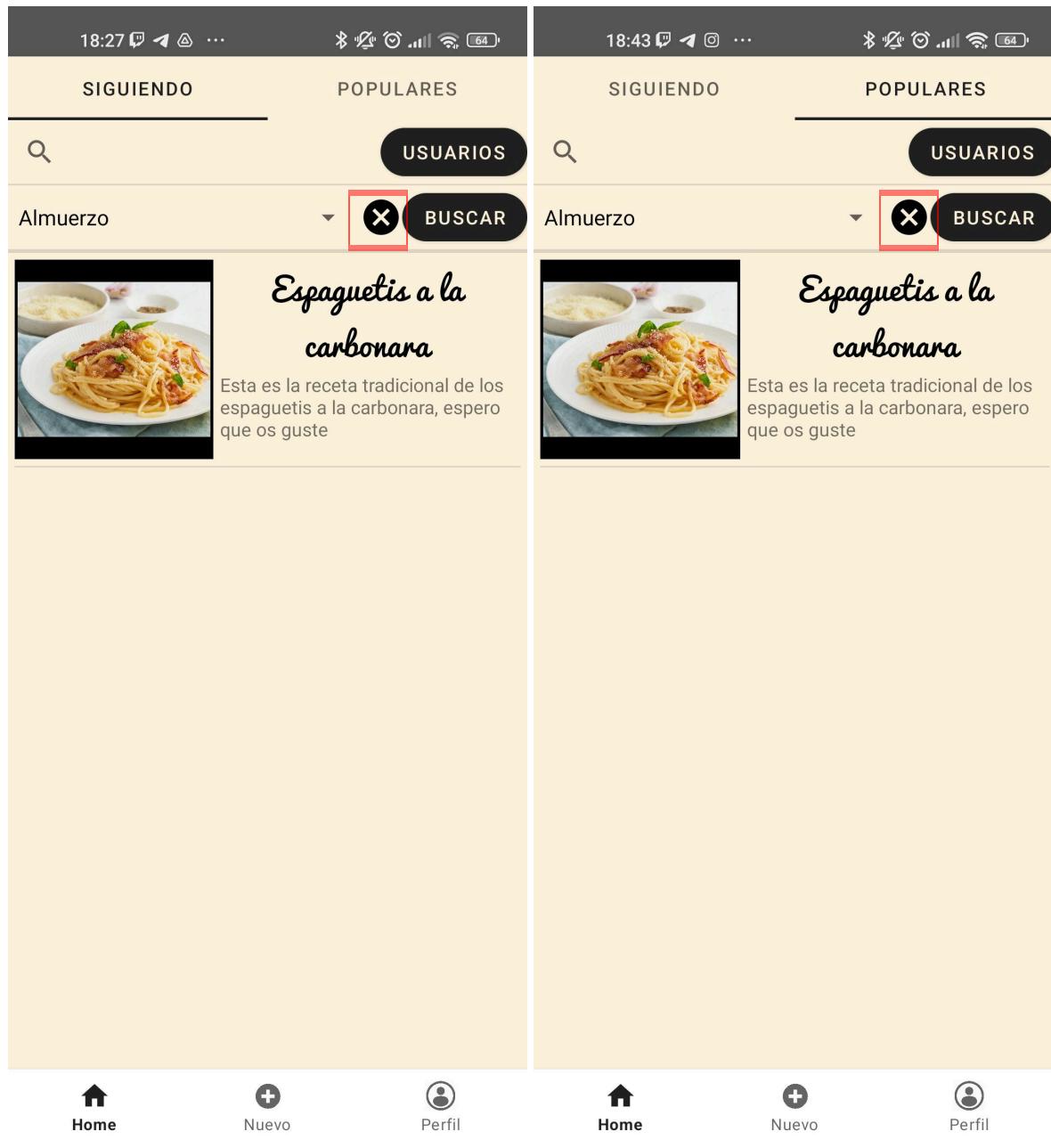
Listado de las publicaciones si se plusa en una te lleva a la pantalla de esa publicación

Home

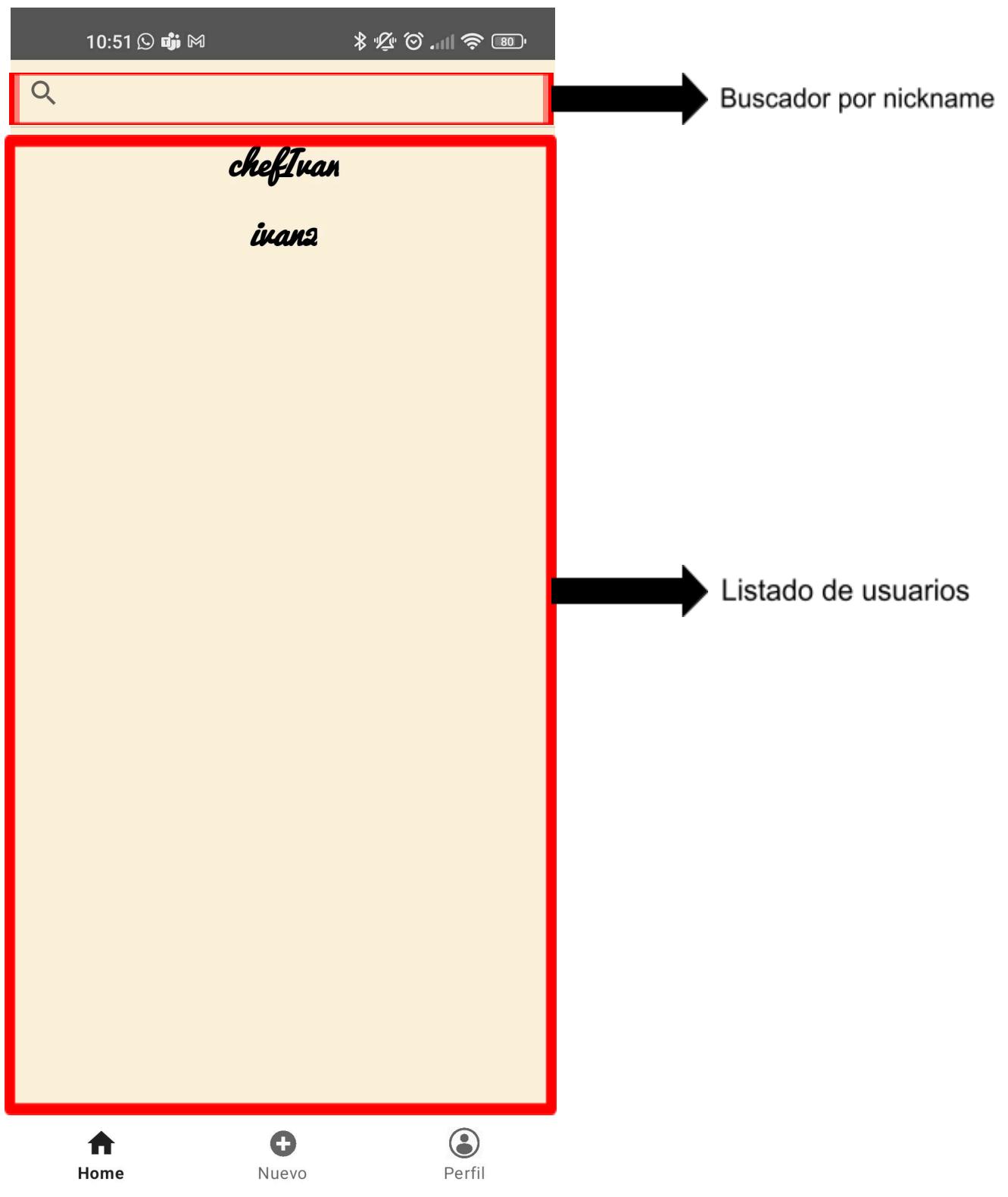
Nuevo

Perfil

Nota: Esta pantalla tiene un botón oculto en las 2 versiones que solo aparece cuando se ha realizado una búsqueda por categoría y sirve para cancelar la búsqueda



- Pantalla “listado usuario” que sirve para buscar usuarios



- Pantalla “autor”

The screenshot shows a mobile application interface for a user profile. At the top, there's a header bar with the time (10:53), signal strength, and battery level (80%). Below the header, the user's name 'ivan2' is displayed in a red-bordered box. To the right of the name are three statistics: 'Seguidores' (1), 'Publicaciones' (2), and 'Siguiendo' (0). A large black button labeled 'Dejar de Seguir' is also in this box. Below this section, the user's name 'Iván Fernández Méndez' is shown in bold. The main content area features two recipe cards. The first card, titled 'Espaguetis a la carbonara', includes an image of the dish, a title, and a descriptive text: 'Esta es la receta tradicional de los espaguetis a la carbonara, espero que os guste'. The second card, titled 'Nachos Caseros', includes an image of a chef's hat, a title, and a descriptive text: 'Nachos caseros faciles y riquísimos para gente que no quiera complicarse mucho'. At the bottom of the screen, there are navigation icons for 'Home' (house icon), 'Nuevo' (plus icon), and 'Perfil' (person icon).

Datos del usuario

Botón para seguir al usuario o dejar de seguirlo

Iván Fernández Méndez

*Espaguetis a la carbonara*

Esta es la receta tradicional de los espaguetis a la carbonara, espero que os guste

*Nachos Caseros*

Nachos caseros faciles y riquísimos para gente que no quiera complicarse mucho

Home

Nuevo

Perfil

- Pantalla “publicación”: Aquí es donde se ve la receta al completo (se ha dividido en 2 imágenes por motivos de espacio)



Los 2 botones dentro de este recuadro azul [ ] son 2 botones que solo aparecen si la publicación es tuya y nos permite borrar y editar publicaciones

Pantalla “editar perfil”: Pantalla pensada para editar una publicación (Se ha dividido en 2 imágenes por motivos de tamaño)





Home

Nuevo

Perfil

## Administración

La aplicación también contiene una parte de administración la cual nos permite gestionar el baneo de publicaciones, usuarios y comentarios además de ver los que ya están baneados, todo esto solo es posible si eres administrador. A continuación se mostrará algunas pantallas donde la vista de administración cambia respecto a la de usuario normal.

The screenshot shows a mobile application interface. At the top, there is a navigation bar with icons for time (10:56), battery level (79%), signal strength, and connectivity. Below the navigation bar is a header with the handle "cocineroExperto". To the right of the handle is a button labeled "DESBANEAR".

The main content area displays the user's status as "Baneado" (Banned). It shows the following statistics: Seguidores (Followers) 0, Publicaciones (Publications) 0, and Siguiendo (Following) 1. Below these stats is a large black "Seguir" (Follow) button.

Below the stats, the user's name is listed as "Juan Ramírez Esteban" with the subtitle "Soy un cocinero experto".

Annotations on the left side of the screenshot include:

- A box labeled "Autor" with a curved arrow pointing to it, indicating the user is an administrator.
- Text below the arrow stating "Texto que indica que el usuario está baneado" (Text that indicates the user is banned).

Annotations on the right side of the screenshot include:

- A curved arrow pointing to the "DESBANEAR" button with the text "Botón para banear/desbanear. Solo es visible si eres administrador" (Button to ban/unban. Only visible if you are an administrator).

At the bottom of the screen are three navigation icons: "Home" (house icon), "Nuevo" (plus sign icon), and "Perfil" (person icon). There is also a horizontal line between the "Nuevo" and "Perfil" icons.

Publicación



Texto que indica que la publicación está baneado

The screenshot shows a mobile application interface. At the top, there is a header bar with the time "10:55", signal strength, battery level at 80%, and other icons. Below the header, the title of the post is "Patatas fritas de bolsa". To the right of the title is a button labeled "DESBANEAR". The post is marked as "Baneado" (banned) by the user "chefIvan".  
**Categoría:**  
- Cerdo  
- Desayuno  
A like icon with the number "0" indicates no likes.  
**Ingredientes:**  
- Son patatas fritas  
**Pasos:**  
1 Abre el paquete de patatas fritas  
  
Below the post content, there is a section titled "Comentarios" (Comments). A text input field says "Escribe un comentario" (Write a comment) with a right-pointing arrow icon. The bottom of the screen features a navigation bar with three items: "Home" (with a house icon), "Nuevo" (with a plus icon), and "Perfil" (with a person icon). A horizontal line is positioned between the "Nuevo" and "Perfil" icons.



Botón para banear/desbanear. Solo es visible si eres administrador



## Listado publicación

Texto que indica que la publicación está baneado

## Comentario



## Ingredientes:

- Espaguetis
  - Beicon
  - Queso rayado parmesano

## Pasos:

- 1 Cocer los espaguetis con agua, aceite y sal
  - 2 Hacer el beicon a la plancha, ponle una pizca de aceite para que no se pegue
  - 3 Apartar los espaguetis cuando esten hechos junto con el beicon
  - 4 Añadir el queso parmesano

## *Comentarios*

## Escribe un comentario

>

*Enrique*

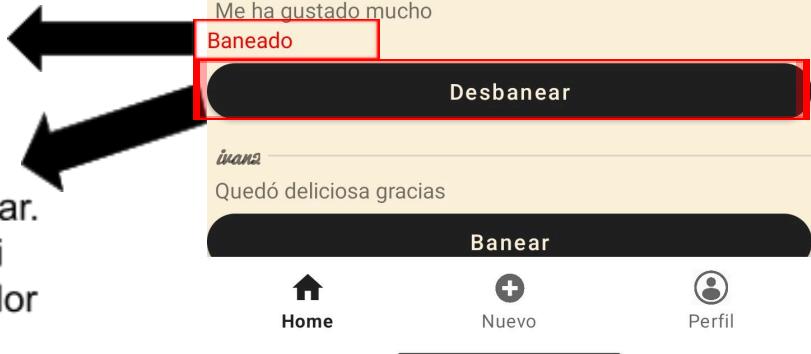
Me ha gustado mucho

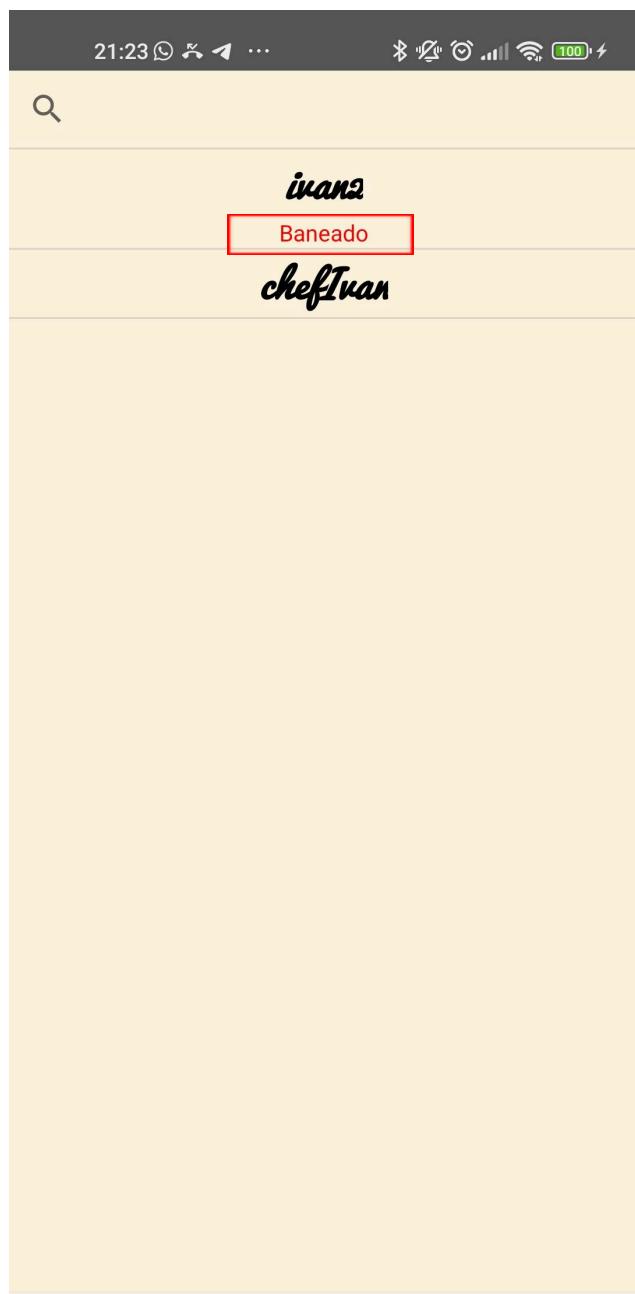
## Baneado

## Desbanear

Texto que indica  
que la  
comentarios está  
baneado

Botón para  
banear/desbanear.  
Solo es visible si  
eres administrador





Texto que indica que el autor está baneado



Home



Nuevo



Perfil

### *3.7.3 Identificación de perfiles de usuarios*

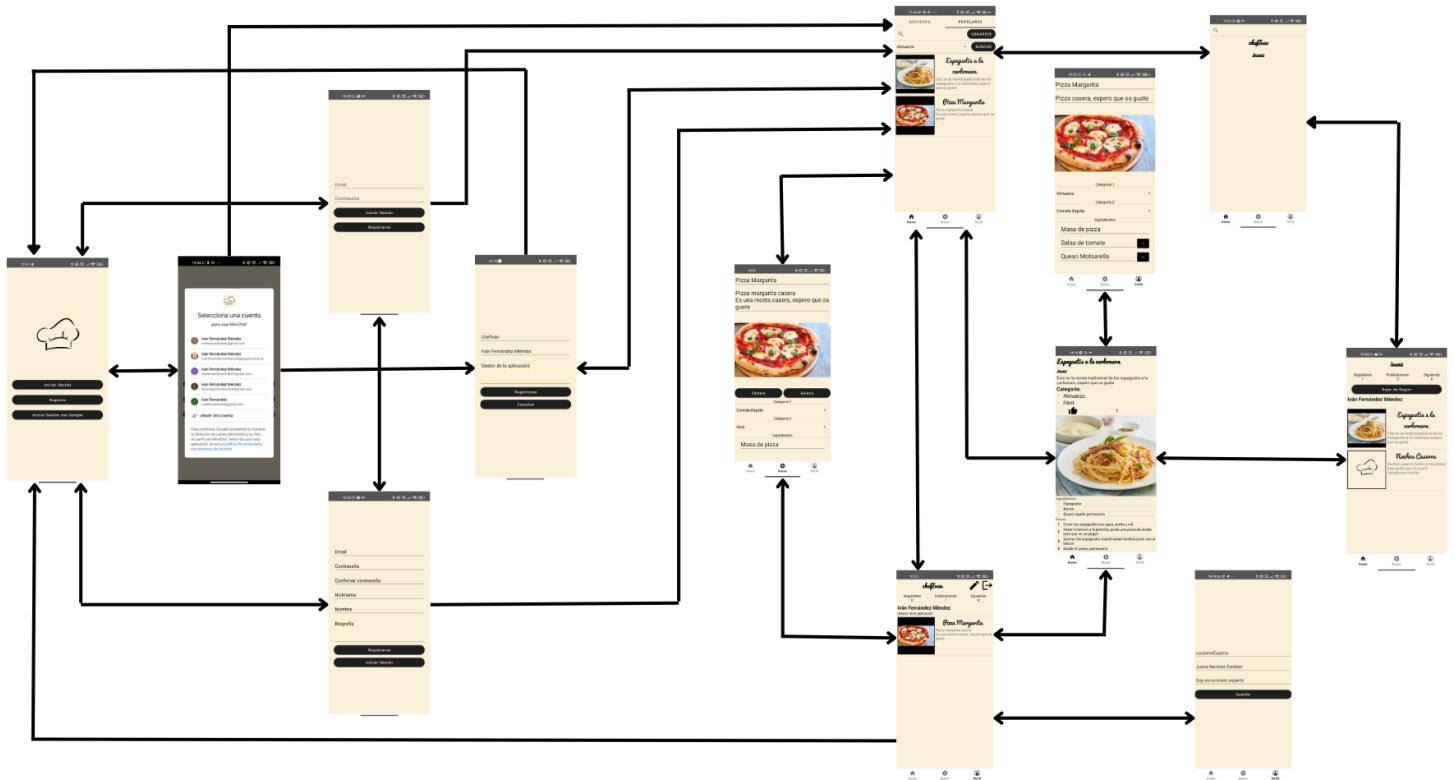
En la aplicación hay 2 tipos de usuarios los cuales pueden hacer cosas diferentes:

- Usuario común: las pantallas a las que puede acceder un usuario son:
  - Perfil
  - Editar
  - Nueva publicación
  - Listado publicaciones (sus 2 versiones)
  - Listado usuarios
  - Autor
- Gestor: se encarga de controlar los baneos ha publicaciones, usuarios y comentarios. Aparte de eso puede hacer lo mismo que un usuario común. Este usuario podrá ver botones y etiquetas que un usuario normal no puede, estos son los especificados en la parte de administración anterior.

### *3.7.4.- Especificación de formatos de impresión*

La aplicación no generará documentos de ningún tipo de formato, solo mostrará la información por pantalla

### 3.7.5.- Especificación de navegabilidad entre pantallas



## **4 Construcción del Sistema**

### **Repositorio GitHub**

<https://github.com/kiruh26/MiniChef>

### **Diseño**

En la aplicación se han utilizado varios patrones de programación. Se ha utilizado el patrón Vista Controlador para gestionar toda la conexión a la base de datos de Firestore. También se ha utilizado el MVVM con el patrón Observer para mantener la gestión de la interfaz separada de la gestión datos y mejorar la gestión eventos de actualización de datos

### **Librerías utilizadas**

```
//Librerías base de Android Studio
implementation 'androidx.core:core-ktx:1.7.0'
implementation 'androidx.appcompat:appcompat:1.6.1'
implementation 'androidx.constraintlayout:constraintlayout:2.1.4'
testImplementation 'junit:junit:4.13.2'
androidTestImplementation 'androidx.test.ext:junit:1.1.5'
androidTestImplementation 'androidx.test.espresso:espresso-core:3.5.1'

//Librerías para el MVVM
implementation 'androidx.lifecycle:lifecycle-livedata-ktx:2.5.1'
implementation 'androidx.lifecycle:lifecycle-viewmodel-ktx:2.5.1'

//Librerías para la navegación entre fragments
implementation 'androidx.navigation:navigation-fragment-ktx:2.5.3'
implementation 'androidx.navigation:navigation-ui-ktx:2.5.3'

//Librerías para el tablayout
implementation 'androidx.viewpager2:viewpager2:1.0.0'
implementation 'com.google.android.material:material:1.9.0'

//Librerías de firebase
implementation platform('com.google.firebaseio:firebase-bom:31.3.0')
implementation 'com.google.firebase:firebase-analytics-ktx'
implementation 'com.google.firebase:firebase-auth-ktx'
implementation 'com.google.firebase:firebase-firebase'
implementation 'com.google.android.gms:play-services-auth:20.4.1'
implementation 'com.google.firebase:firebase-storage-ktx'

//Librería para mostrar las imágenes sin necesidad de descargarla
implementation 'com.github.bumptech.glide:glide:4.15.1'
```

## Funcionalidades No Vistas en Clase

### - Cámara y galería

Para hacer uso de la galería y la cámara es necesario realizar un intentForResult, esto viene a ser una navegación a otra pantalla que espera que vuelva con un resultado. Se hace un intent llamando a la pantalla de galería y otro para la pantalla de la cámara: A continuación el código de cada uno

Galería:

```
private fun abrirGaleria() {
    val intent=Intent()
    intent.type="image/*"
    intent.action=Intent.ACTION_GET_CONTENT

    //startActivityForResult(intent,100)
    openGalleryLaucher.launch(intent.type)
}
```

Cámara:

```
btncamara.setOnClickListener { it: View!
    val intent=Intent(MediaStore.ACTION_IMAGE_CAPTURE)
    intent.putExtra( name: "android.intent.extra.PICTURE_QUALITY", value: 100)
    try {
        val photoFile = createImageFile()
        val photoURI:Uri = FileProvider.getUriForFile(this.requireContext(),
            authority: BuildConfig.APPLICATION_ID+".provider",
            photoFile)
        // Asigna la URI como salida del intent de la cámara
        intent.putExtra(MediaStore.EXTRA_OUTPUT, photoURI)
        startForResult.launch(intent)
    } catch (ex: IOException) {
        // Error al crear el archivo
        ex.printStackTrace()
    }
    // Continúa solo si el archivo fue creado con éxito
    // Obtiene la URI del archivo creado
}
```

A continuación los resultados de los intent y cómo se genera el bitmap tanto para mostrarlo en el previsualizador como para enviarlo a Firebase Storage:

Galería:

```
private val openGalleryLaucher = registerForActivityResult(ActivityResultContracts.GetContent()) { imageUrl ->
    // Lógica para manejar el resultado de la galería aquí
    if (imageUrl != null) {
        val source=ImageDecoder.createSource(requireContext().contentResolver,imageUrl)
        //imagenReceta = MediaStore.Images.Media.getBitmap(requireContext().contentResolver, imageUrl)
        imagenReceta = ImageDecoder.decodeBitmap(source)
        binding.ivFotoReceta.setImageURI(imageUrl)
    }
}
```

Cámara:

```
//Result de la camara
private val startForResult=registerForActivityResult(ActivityResultContracts.StartActivityForResult()){
    result: ActivityResult ->
    if (result.resultCode== RESULT_OK){
        try {
            val imageFile = File(currentPhotoPath)
            if (imageFile.exists()) {
                imagenReceta = BitmapFactory.decodeFile(imageFile.getAbsolutePath)
                this.binding.ivFotoReceta.setImageBitmap(imagenReceta)
            } else {
                Log.e( tag: ":::Error", msg: "El archivo de imagen no existe.")
            }
        }catch (e:Exception){
            Log.e( tag: ":::Error", e.message.toString())
        }
    }
}
```

## - *Librería Glide*

La librería Glide es una librería externa que se tiene que exportar y que nos permite visualizar una imagen a través de un URL como con la proporcionada por Firebase Storage. Todo esto lo realiza sin la necesidad de descargar ningún archivo. La forma en la que se ha utilizado en el proyecto es la siguiente:

```
Glide.with(requireContext()).load(publicacion.imagen!!).into(binding.imagenDetallePublicacion)
```

## **5 Conclusión**

### ***Errores encontrados***

El principal bug encontrado es a la hora de guardar la imagen de la publicación ya que esta te devuelve un bitmap de muy baja calidad. Para resolverlo tuve que crear un archivo temporal en el teléfono que se elimina cuando termina la creación del bitmap (desde un fichero si te deja generarlo a mayor calidad).

Otro bug que tuve es a la hora de publicar una receta ya que si bien no daba ningún tipo de problema y lo generaba todo correctamente, la aplicación y sobre todo el hilo que se encarga de la interfaz se sobrecargaba por culpa de la alta calidad de la imagen. Para resolverlo tuve que usar una “corrutina”, Estos son bloques de código asíncronos que nos permite generar métodos suspendidos (métodos que paran la ejecución hasta que se finalice una operación). De esta forma el hilo que se encarga de la interfaz de usuario (UI), no se ve sobrecargado y se pueden realizar cambios mientras que la aplicación está subiendo una imagen

### ***Futuras mejoras***

Ha habido muchas ideas de funcionalidades que debido al tiempo me han sido imposibles de añadir. Estas son:

- Calculador de kcal: me gustaría añadir un calculador de calorías que nos permita saber cuántas calorías tiene (aproximadamente cada receta). Esto pienso que sería de gran utilidad ya que hoy en día se tiene muy en cuenta las dietas y el control de cuántas calorías comemos al día. Haría que la aplicación se diferenciese de otras.
- También me gustaría que un usuario administrador pueda realizar cambios en la lista de categorías desde la misma aplicación
- Poder recuperar tu contraseña en caso de que se te olvide

- Crear un calendario en el que puedas organizar las recetas que vas a utilizar cada día, como forma de organizar tu dieta
- Poder ver a qué publicaciones les has dado like y así poder acceder a ellas fácilmente

## *Aprendizaje*

- Mejoré mis conocimientos en kotlin:

Kotlin es un lenguaje de programación que dimos en clase de manera superficial. Decidí realizar el proyecto en este lenguaje para aprender más ya que considero que es más eficiente programar en java.

- Uso de nuevo módulos de firebase

Para guardar las imágenes y tener las funcionalidades de autenticación tuve que hacer uso de los módulos Firestore y Authentication que no se dieron en clase.

## *Conclusión tras finalizar el proyecto*

Ahora que he terminado el proyecto he llegado a la conclusión de que podría haber aspirado a menos y poder realizarlo todo con más tiempo y mejor depurado.

En conclusión pienso que he querido abarcar demasiado, no tanto por número de funcionalidades sino por la complejidad de estas. Tampoco tuve en cuenta los errores que podrían haber en el proyecto cuando lo diseñe que alargaron los plazos de las tareas. En un futuro proyecto pienso que no cometeré muchos de estos fallos de diseño.

## ***6 Glosario de términos***

A lo largo del documento se han utilizado palabras para describir cosas muy concretas que puede que no se entienda correctamente su significado. Estas son:

- Like: se usa como sinónimo de dar “me gusta”, es una forma de valorar una publicación
- Ban, Banear, Baneado: en la aplicación banear es la acción de suprimir/prohibir una elemento (publicación, comentario, usuario) cuando este elemento está prohibido se le considera baneado por lo que la aplicación no lo muestra a los usuarios comunes y si es un usuario su entrada a la aplicación está vetada
- Desbanear/ Unban: Sería lo opuesto a los términos Ban/Banear
- Nickname: es el nombre que tienen los usuarios de cara a internet. Es como una especie de mote que te identifica y por el que la gente te conoce en internet

## 7 Bibliografía

- ## - Documentación de kotlin

- ## - Documentación de Firebase

En concreto utilicé la documentación de la implementación de los siguientes módulos en android studio:

- [Authentication](#)
  - [Firestore](#)
  - [Storage](#)

- ## - Documentación de la librería Glide