



## **MyFood – Control de la caducidad de alimentos**

**Víctor Andrés Cáceres**

Grado de Ingeniería Informática

Desarrollo de Aplicaciones para Dispositivos Móviles

**David Escuer Latorre**

**Jordi Almirall López**

**Carles Garrigues Olivella**

Enero 2023



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>MyFood</i>
<b>Nombre del autor:</b>	<i>Víctor Andrés Cáceres</i>
<b>Nombre del consultor/a:</b>	<i>David Escuer Latorre</i> <i>Jordi Almirall López</i>
<b>Nombre del PRA:</b>	<i>Carles Garrigues Olivella</i>
<b>Fecha de entrega (mm/aaaa):</b>	01/2023
<b>Titulación:</b>	<i>Grado Ingeniería Informática</i>
<b>Área del Trabajo Final:</b>	<i>Desarrollo Aplicaciones Dispositivos Móviles</i>
<b>Idioma del trabajo:</b>	<i>Castellano</i>
<b>Palabras clave</b>	<i>Móvil, Despensa, Alimento</i>
<b>Resumen del Trabajo (máximo 250 palabras):</b> <i>Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.</i>	
<p>En la actualidad se desperdician en el mundo unos 931 millones de toneladas de comida y una de las principales causas es la fecha de caducidad de los alimentos. Cada vez nos vemos más sometidos a espaciar más nuestras compras y guardar más cantidad de productos en diferentes lugares de almacenaje.</p> <p>La encuesta y entrevistas realizadas (punto 2) revelan que los usuarios tienen problemas para controlar los productos que disponen en casa y su fecha de caducidad, así como la compra de alimentos.</p> <p>MyFood es una App que permite al usuario tener un control de los alimentos que tiene en casa y su caducidad. Permite añadir productos a la despensa mediante el código de barras o manualmente y avisar diariamente de los productos caducados o casi caducados mediante un correo electrónico.</p> <p>La finalidad del TFG ha sido crear una App sencilla y funcional que permita evitar tirar la comida por motivos de fecha de vencimiento al usuario y mejorar las compras.</p> <p>Como conclusión cabe mencionar la dificultad de adaptar una idea personal tanto a nivel de diseño como funcionalidad a las necesidades del usuario.</p>	

Como líneas futuras se plantea el desarrollo de un botón OCR para capturar los datos del alimento a partir de una captura de la imagen de la etiqueta y las notificaciones de la barra de tareas. Además, se valora la posibilidad de migrar la App también a otras plataformas como Apple o Windows Mobile.

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

Currently, some 931 million tons of food are wasted in the world and one of the main causes is the expiration date. We are increasingly subjected to space out our purchases more and store more products in different storage places.

The survey and interviews conducted (point 2) reveal that users have problems controlling the products they have at home and their expiration date, as well as the purchase of food.

MyFood is an App that allows the user to have control of the food they have at home and their expiration. It allows you to add products to the pantry by barcode or manually and notify daily of expired or almost expired products by email.

The purpose of the TFG has been to create a simple and functional App that allows to avoid throwing away the food for reasons of expiration date to the user and improve purchases.

In conclusion, it is worth mentioning the difficulty of adapting a personal idea both at the level of design and functionality to the needs of the user.

As future lines, the development of an OCR button is proposed to capture food data from a capture of the label image and taskbar notifications. In addition, the possibility of migrating the App also to other platforms such as Apple or Windows Mobile is valued.

# Índice

1. Introducción .....	1
1.1 Contexto y justificación del Trabajo .....	1
1.2 Objetivos del Trabajo .....	5
1.3 Enfoque y método seguido .....	8
1.4 Planificación del Trabajo .....	9
1.5 Breve resumen de productos obtenidos .....	14
1.6 Breve descripción de los otros capítulos de la memoria.....	14
2. Usuarios y Contexto de Uso .....	16
2.1 Investigación / Tomas de Datos.....	16
2.1.1 Justificación y proceso de planteamiento .....	16
2.1.2 Desarrollo y resultados de la encuesta.....	16
2.1.3 Desarrollo y resultados de la entrevista profunda .....	16
2.1.4 Conclusiones obtenidas .....	17
2.2 Perfil de Usuario.....	17
2.2.1 Ficha nº 1 .....	18
2.2.2 Ficha nº 2.....	19
2.2.3 Ficha nº 3.....	20
2.2.4 Ficha nº 4.....	21
3. Diseño conceptual.....	23
3.1 Escenarios de uso.....	23
3.2 Flujo de interacción .....	26
4. Prototipado .....	27
4.1 Sketches escaneados .....	27
4.2 Prototipo horizontal de alta fidelidad.....	28
5. Evaluación .....	38
5.1 Recopilación de preguntas .....	38
5.2 Tareas que los usuarios deben realizar .....	38
5.3 Preguntas referentes a las tareas.....	39
5.4 Análisis de resultados e incluir mejoras .....	40
6. Diseño Técnico.....	41
6.1 Definición de los casos de uso .....	41
6.1.1 Diagrama UML de Actores .....	41
6.1.2 Diagrama UML Flujo.....	42
6.1.3 Listado casos de uso.....	42
6.2 Diseño de la arquitectura .....	52
6.2.1 Diagrama UML Base de datos .....	52
6.2.2 Diagrama UML Entidades y clases.....	54
6.2.3 Diagrama explicativo de la arquitectura del sistema .....	59
6.2.4 Librerías externas.....	60
7. Implementación .....	61
7.1 Entorno de desarrollo empleado.....	61
7.2 Herramientas.....	61
7.3 Servicio PHP y MySQL .....	61
7.4 API Openfood.....	61
7.5 Análisis del estado del proyecto .....	61

8. Pruebas.....	63
8.1 Prueba de aplicación.....	63
8.2 Pruebas Unitarias.....	63
8.3 Pruebas Funcionales de Integración .....	63
9. Valoración económica.....	65
9.1 Estrategia de Monetización.....	65
9.2 Plan de Marketing y Distribución .....	65
10. Conclusiones y Líneas Futuras.....	66
10.1 Conclusiones del Trabajo .....	66
10.2 Reflexión de Logros .....	66
10.3 Análisis Seguimiento de Planificación y Metodología.....	66
10.4 Líneas Futuras .....	67
11. Glosario .....	68
12. Bibliografía.....	70

## Lista de figuras

<i>Figura 1: : Reparto toneladas de comida desperdiciada en el mundo</i> .....	1
<i>Figura 2: Captura 1 App "Rastreador de caducidad"</i> .....	2
<i>Figura 3: Captura 2 App "Rastreador de caducidad"</i> .....	2
<i>Figura 4: Captura 3 App "Rastreador de caducidad"</i> .....	2
<i>Figura 5: Captura 1 App "Nosh"</i> .....	3
<i>Figura 6: Captura 2 App "Nosh"</i> .....	3
<i>Figura 7: Captura 3 App "Nosh"</i> .....	3
<i>Figura 8: Captura 1 App "Control Alimentos y Compras"</i> .....	4
<i>Figura 9: Captura 2 App "Control Alimentos y Compras"</i> .....	4
<i>Figura 10: Captura 3 App "Control Alimentos y Compras"</i> .....	4
<i>Figura 11: Diagrama de Gantt de planificación de tareas</i> .....	13
<i>Figura 12: Ficha perfil - Jovenes menores de 25</i> .....	18
<i>Figura 13: Ficha perfil - Jovenes entre 25 y 35</i> .....	19
<i>Figura 14: Ficha perfil - Adultos entre 36 y 45 años</i> .....	20
<i>Figura 15: Ficha perfil - Mayores de 45 años</i> .....	21
<i>Figura 16: Flujo de interacción de la App</i> .....	26
<i>Figura 17: Sketch escaneado 1</i> .....	27
<i>Figura 18: Sketch escaneado 2</i> .....	27
<i>Figura 19: Sketch escaneado 3</i> .....	27
<i>Figura 20: Sketch escaneado 4</i> .....	27
<i>Figura 21: Prototipo Login</i> .....	28
<i>Figura 22: Prototipo Registrarse</i> .....	29
<i>Figura 23: Prototipo "¿Olvido su contraseña?"</i> .....	30
<i>Figura 24: Prototipo Pantallas principales navegación</i> .....	31
<i>Figura 25: Prototipo Despensa</i> .....	32
<i>Figura 26: Prototipo Ver producto Despensa</i> .....	33
<i>Figura 27: Prototipo Lista de compra</i> .....	34
<i>Figura 28: Prototipo Caducidad</i> .....	35
<i>Figura 29: Prototipo Recetas</i> .....	36
<i>Figura 30: Prototipo Configuración</i> .....	37
<i>Figura 31: Diagrama UML de Actores</i> .....	41
<i>Figura 32: Diagrama de Flujo</i> .....	42
<i>Figura 33: Modelo Entidad Relación base de datos SQLite</i> .....	52
<i>Figura 34: Definición tablas base de datos SQLite</i> .....	53
<i>Figura 35: Modelo Entidad Relación base datos MySQL</i> .....	53
<i>Figura 36: Definición tablas base de datos MySQL</i> .....	54
<i>Figura 37: Diagrama de clases MVP 1</i> .....	55
<i>Figura 38: Diagrama de clases MVP 2</i> .....	56
<i>Figura 39: Diagrama de clases MVP 3</i> .....	57
<i>Figura 40: Diagrama de clases MVP 4</i> .....	58
<i>Figura 41: Diagrama de clases MVP 5</i> .....	59
<i>Figura 42: Diagrama arquitectura del sistema</i> .....	60

## Lista de tablas

<i>Tabla 1: Comparativa App "Rastreador de caducidad"</i> .....	2
<i>Tabla 2: Comparativa App "Nosh"</i> .....	3
<i>Tabla 3: Comparativa App "Control Alimentos y Compras"</i> .....	4
<i>Tabla 4: Obetivos no funcionales</i> .....	6
<i>Tabla 5: Objetivos funcionales</i> .....	8
<i>Tabla 6: Planificación PEC1 - Plan de Trabajo</i> .....	10
<i>Tabla 7: Planificación PEC2 - Diseño y Arquitectura</i> .....	10
<i>Tabla 8: Planificación PEC3 - Implementación</i> .....	11
<i>Tabla 9: Planificación de la Entrega final</i> .....	11
<i>Tabla 10: Planificación Defensa virtual</i> .....	12
<i>Tabla 11: Escenario contexto de uso - Lista de compra</i> .....	23
<i>Tabla 12: Escenario contexto de uso - Despensa</i> .....	24
<i>Tabla 13: Escenario contexto de uso - Productos caducados</i> .....	24
<i>Tabla 14: Escenario contexto de uso - Sugerencia recetas</i> .....	25
<i>Tabla 15: Escenario contexto de uso - Información nutricional</i> .....	25
<i>Tabla 16: Caso de uso CU-001</i> .....	43
<i>Tabla 17: Caso de uso CU-002</i> .....	43
<i>Tabla 18: Caso de uso CU-003</i> .....	43
<i>Tabla 19: Caso de uso CU-004</i> .....	44
<i>Tabla 20: Caso de uso CU-005</i> .....	44
<i>Tabla 21: Caso de uso CU-006</i> .....	45
<i>Tabla 22: Caso de uso CU-007</i> .....	45
<i>Tabla 23: Caso de uso CU-008</i> .....	46
<i>Tabla 24: Caso de uso CU-009</i> .....	46
<i>Tabla 25: Caso de uso CU-010</i> .....	46
<i>Tabla 26: Caso de uso CU-011</i> .....	47
<i>Tabla 27: Caso de uso CU-012</i> .....	47
<i>Tabla 28: Caso de uso CU-013</i> .....	48
<i>Tabla 29: Caso de uso CU-014</i> .....	48
<i>Tabla 30: Caso de uso CU-015</i> .....	48
<i>Tabla 31: Caso de uso CU-016</i> .....	49
<i>Tabla 32: Caso de uso CU-017</i> .....	49
<i>Tabla 33: Caso de uso CU-018</i> .....	49
<i>Tabla 34: Caso de uso CU-019</i> .....	50
<i>Tabla 35: Caso de uso CU-020</i> .....	50
<i>Tabla 36: Caso de uso CU-021</i> .....	50
<i>Tabla 37: Caso de uso CU-022</i> .....	51
<i>Tabla 38: Caso de uso CU-023</i> .....	51
<i>Tabla 39: Caso de uso CU-024</i> .....	51
<i>Tabla 40: Caso de uso CU-025</i> .....	52



# 1. Introducción

## 1.1 Contexto y justificación del Trabajo

MyFood es una propuesta de App Mobile que pretende evitar que los alimentos que disponemos en casa queden incomedibles por mal estado y asimismo perder dinero. Se trata de un tema relevante dado al agitado y veloz ritmo de vida que se tiene en estos tiempos. Esto provoca que se preste poca atención a la comida que hay en casa, pero nuestra App permite que se actúe con mayor organización y control.

Según un artículo de la BBC [1], en el año 2019 se desperdiciaron un total de 931 millones de toneladas de alimentos en el mundo, un 17% del total. De esta cifra se sabe que el 61% proviene de los hogares familiares, un 13% de organizaciones comerciales y un 26 % de los servicios de hostelería. Para evitar tal caos, uno de los factores a tener en cuenta es verificar la fecha de vencimiento de los productos.



Figura 1: Reparto toneladas de comida desperdiciada en el mundo

Dentro del conjunto de App's existentes se puede encontrar un sinfín de aplicaciones orientadas a controlar la fecha de caducidad y compra de nuestros alimentos. Algunos ejemplos con sus puntos a favor y a mejorar son:

## Rastreador de caducidad

Descarga: [Rastreador de caducidad - Aplicaciones en Google Play](#)

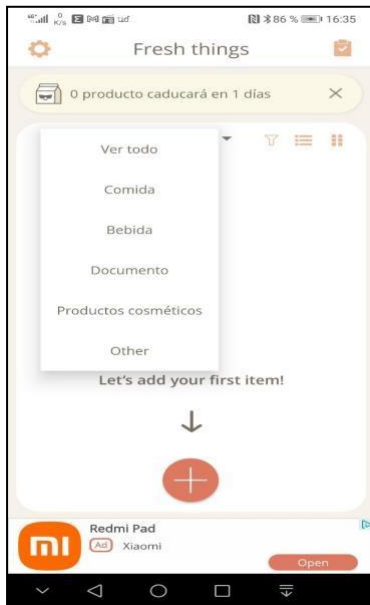


Figura 2: Captura 1 App "Rastreador de caducidad"

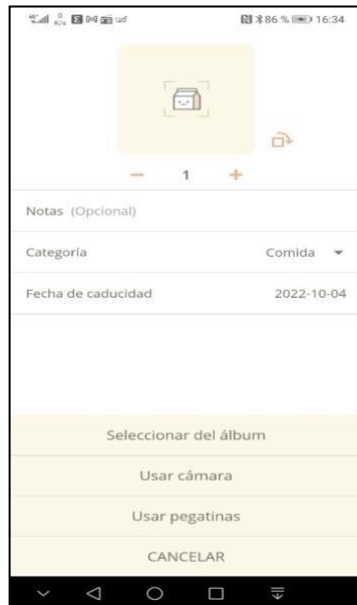


Figura 3: Captura 2 App "Rastreador de caducidad"

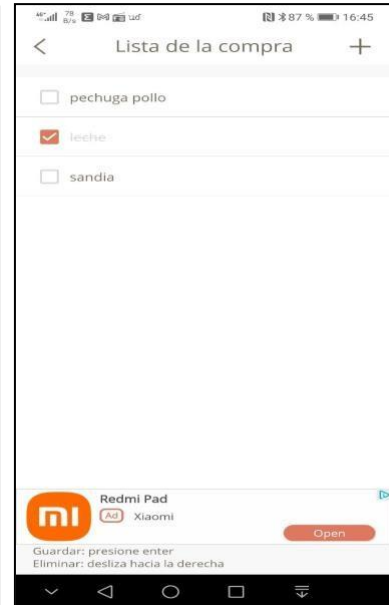


Figura 4: Captura 3 App "Rastreador de caducidad"

Puntos a favor	Puntos a mejorar
Fácil de usar e intuitiva. Buena Experiencia de Usuario (UX)	No lee el tipo de producto por código de barras.
Se pueden incorporar otros elementos que no sean alimentos como: bebidas, cosméticos, etc.	La fecha de caducidad se debe insertar a mano.
Permite insertar una imagen del producto mediante cámara, desde la galería del teléfono o con un icono a modo pegatina.	Se permite insertar una fecha de caducidad anterior a la fecha actual.
Permite enviar una notificación en el teléfono cuando queda un tiempo definido para que caduque el producto.	No muestra una barra de progreso con el tiempo transcurrido desde su compra hasta la fecha de caducidad.

Tabla 1: Comparativa App "Rastreador de caducidad"

Descarga: [nosh - Reduce food waste - Aplicaciones en Google Play](#)

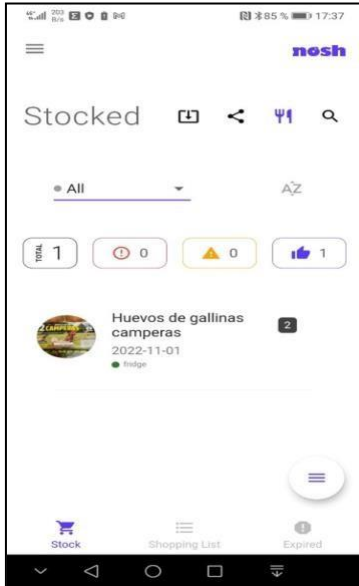


Figura 5: Captura 1 App "Nosh"

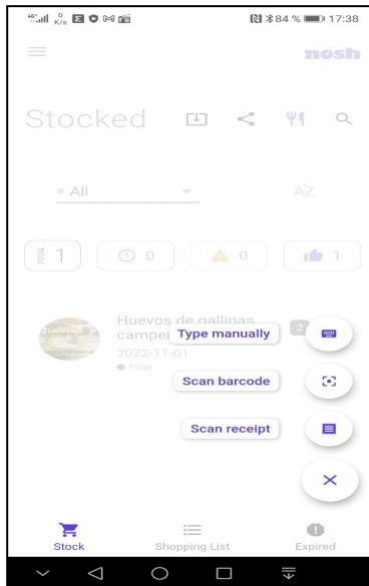


Figura 6: Captura 2 App "Nosh"

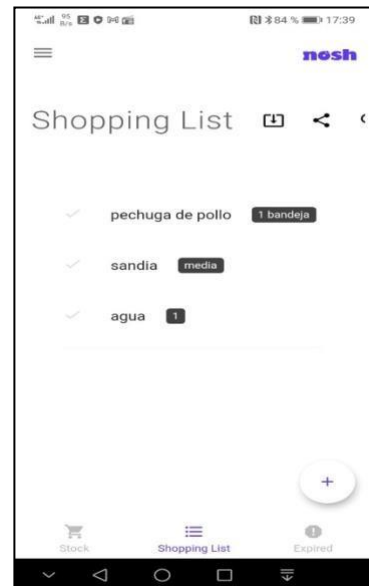


Figura 7: Captura 3 App "Nosh"

Puntos a favor	Puntos a mejorar
Permite crear una lista de la compra para recordar que se debe coger.	La fecha de caducidad se debe insertar a mano.
Permite obtener un gráfico de que productos son los más consumidos, para llevar así un control de salud.	No permite exportar la lista de productos a formato Excel
Fácil usabilidad (UX).	No envía notificaciones cuando un producto esta caducado o a punto de caducar.
Divide el tiempo de caducidad de los productos en 3 franjas: menos de 1 día, menos de 3 y bueno por al menos 4 días.	5 categorías de productos predefinidas. No se pueden añadir nuevas.

Tabla 2: Comparativa App "Nosh"



## Control de Alimentos y Compras

Descarga: [Control de Alimentos y Compras - Aplicaciones en Google Play](#)



Figura 8: Captura 1 App "Control Alimentos y Compras"



Figura 9: Captura 2 App "Control Alimentos y Compras"

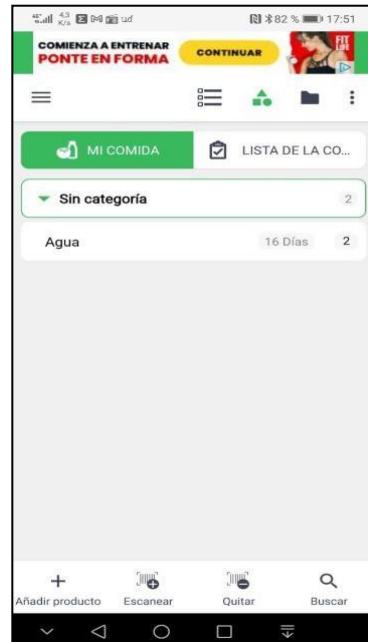


Figura 10: Captura 3 App "Control Alimentos y Compras"

Puntos a favor	Puntos a mejorar
Reconoce el tipo de producto por código de barras	La fecha de caducidad se debe insertar a mano.
Permite exportar la lista de los productos a Excel	No realiza un calculo de la cantidad malgastada en productos caducados.
Permite clasificar los productos o alimentos en categorías.	No muestra una barra de progreso con el tiempo transcurrido desde su compra hasta la fecha de caducidad.
Se puede localizar un producto en la base de datos bien con el micro o con el código de barras.	No envía ningún tipo de aviso cuando un producto ha caducado.

Tabla 3: Comparativa App "Control Alimentos y Compras"

En definitiva, se busca formar una App a partir de las aplicaciones anteriores con lo mejor de cada una y algunas ideas que se echan en falta. Se busca principalmente que se pueda insertar mediante la lectura del código de barras el producto, que puedan exportarse los datos a formato Excel y permita notificar al usuario cuando un producto haya caducado o está a punto. Asimismo, también la fecha de caducidad será introducida por medio de una imagen para ahorrar tiempo.

## 1.2 Objetivos del Trabajo

En el desarrollo de la App se ha definido una serie de objetivos que se deberán cumplir. Se pueden separar en dos tipos. Por un lado, los no funcionales, que hacen referencia a factores que no tengan que ver con el funcionamiento de la aplicación. Como, por ejemplo: el tipo de plataforma, cumplir con los mínimos de calidad de tiempo de respuesta y carencia de errores, etc. Y, por otro lado, se dispone de los requisitos funcionales que hacen más peso a lo que hace la App.

- Objetivos no funcionales

Identificador	Descripción
Diseño y presentación	
ONF1	La App debe estar orientada a todo tipo de usuarios, incluidos daltónicos y epilépticos.
ONF2	La App debe mostrar un orden lógico de la información que maneja de iconos, color de texto y estructura.
ONF3	La App debe conservar las proporciones y el aspecto para cualquiera de los tamaños de móvil.
Rendimiento y escalabilidad	
ONF4	El tiempo de respuesta de carga de datos no debe exceder de 5 segundos.
Retroalimentación	
ONF5	Cada vez que exista un error o se confirme un paso se informara mediante un pop up al usuario

Usabilidad y humanidad	
<b>ONF6</b>	Fácil UX (Experiencia de Usuario). Una persona con pocos conocimientos debe ser capaz de utilizar la aplicación.
<b>ONF7</b>	El menú principal de la aplicación no tendrá más de 5 opciones.
Operacionales y de entorno	
<b>ONF8</b>	La App debe ser capaz de acceder y utilizar la cámara del dispositivo móvil.
<b>ONF9</b>	El sistema debe ser capaz de escanear y reconocer un código de barras
<b>ONF10</b>	El sistema debe ser capaz de reconocer una fecha plasmada en una imagen
<b>ONF11</b>	La App debe estar desarrollada para un sistema operativo Android
<b>ONF12</b>	El almacenamiento de datos de la aplicación se llevará a cabo mediante una base de datos SQLite y MySQL.
Cumplimiento	
<b>ONF13</b>	La App debe ser desarrollada en 2 meses

Tabla 4: Obetivos no funcionales

- Objetivos funcionales

Identificador	Descripción
<b>OF1</b>	Añadir un nuevo producto
<b>OF1.A1</b>	Escanear el código de barras del producto para más fácil inserción del nombre del producto
<b>OF1.A2</b>	Insertar el precio del producto

<b>OF1.A3</b>	Insertar la cantidad del producto (paquete, litros, pieza, kg)
<b>OF1.A4</b>	Insertar la fecha de caducidad mediante una captura de imagen o calendario desplegable
<b>OF1.A5</b>	Añadir una imagen mediante la cámara, galería o pegatina
<b>OF2</b>	Borrar un producto
<b>OF3</b>	Consultar todos los productos comprados.
<b>OF3.A1</b>	Precio total de la compra realizada
<b>OF3.A2</b>	Precio malgastado en productos caducados
<b>OF3.A3</b>	Precio de cada uno de los productos
<b>OF3.A4</b>	Mostrar productos caducados en color rojo
<b>OF4</b>	Consultar lista de productos por caducidad.
<b>OF4.A1</b>	Productos caducados
<b>OF4.A2</b>	Productos a punto de caducar
<b>OF4.A3</b>	Productos de 1 a 10 días de caducar
<b>OF5</b>	Consultar lista de productos a comprar
<b>OF6</b>	Añadir un producto a lista la lista de compra
<b>OF7</b>	Borrar un producto de la lista de compra
<b>OF8</b>	Consultar recetas
<b>OF9</b>	Sugerir recetas
<b>OF10</b>	Login
<b>OF11</b>	Registrarse
<b>OF12</b>	Recuperar contraseña
<b>OF13</b>	Acceso a App

<b>OF14</b>	Añadir Lugar de almacenaje
<b>OF15</b>	Borrar Lugar de almacenaje
<b>OF16</b>	Añadir Unidad de Cantidad
<b>OF17</b>	Borrar Unidad de Cantidad
<b>OF18</b>	Notificar cuando un producto ha caducado
<b>OF19</b>	Notificar cuando un producto está a punto de caducar

*Tabla 5: Objetivos funcionales*

### 1.3 Enfoque y método seguido

Tal como se ha comentado antes, existen un sinfín de aplicaciones en el mercado para controlar la fecha de caducidad de los alimentos. Algunas de ámbito gratuito y otras de pago, pero se hace difícil adaptar ninguna al usuario debido a que son de código cerrado.

Es por ello que tras analizar y comparar varias aplicaciones se ha optado por conservar los aspectos positivos que han gustado e incorporar funcionalidades que se piensan que pueden mejorar y dar una mayor facilidad de uso al usuario.

Se ha elegido que la implementación se lleve a cabo en una aplicación nativa en Android por su fácil acceso a las funcionalidades del dispositivo y porque Android es un sistema operativo de móvil más asequible en el mercado y más extendido [2]. Este hecho ayuda a tener un mejor control de la cámara del dispositivo para la lectura del código de barras y la lectura mediante imagen de la fecha de caducidad.

Respecto al modelo de desarrollo seguido, se ha optado por una metodología ágil basada en continuas iteraciones. En la que en cada iteración se revisan todas las fases del ciclo de desarrollo. Los motivos de su elección son los que siguen:

- El software no es crítico. Se trata de una aplicación sencilla de uso doméstico.
- Se emplean ciclos de desarrollo cortos.
- Solo existe una persona como recurso humano de desarrollo.
- Cambios continuos en el código. Los cuáles obligan a volver a atrás y modificar elementos.



#### 1.4 Planificación del Trabajo

El desarrollo del proyecto será implementado por el estudiante Víctor Andrés Cáceres como único recurso humano, obteniendo el soporte y dirección del consultor de TFG.

En la realización del trabajo se utilizarán las siguientes herramientas:

- Ordenador MacBook Pro con sistema operativo macOS Ventura versión 13.0.1
- Android Studio Delphin 2021.3.1: Se trata de un IDE de desarrollo para aplicaciones nativas en Android.
- Kotlin 1.7.10: Lenguaje de programación que mejor se acopla a Android Studio [3], muy parecido a java y con notables ventajas.
- SQLite: base de datos de formato universal.
- VCS Git: Sistema que funcionará con Android Studio para mantener un registro y control del código de la App.
- GrantPro: software gratuito para la planificación de proyectos.
- Draw.io: Software para la creación de diagramas UML.
- Marvel: Para la generación de prototipos.
- Gimp: Programa de edición gráfica para el trabajo con las imágenes incluidas en el trabajo,
- infinity.net sitio online para scripts php y base de datos MySQL.
- Cron-job.org sitio web para programar un script php online cada x tiempo.
- Emulador scrpcy para más comodidad en la ejecución y depuración en un dispositivo móvil físico.
- Google Colab para realizar un pequeño programa de web scraping en Python.
- Camtasia: Herramienta para grabar la pantalla del ordenador y realizar la presentación visual virtual.

La planificación temporal del trabajo está formada por un conjunto de 976 horas, considerando 8 horas cada día de lunes a domingo, festivos incluidos. La cuál viene determinada por el inicio y final de cada una de las PEC's impuestas.

Las tareas y subtareas a realizar durante el trabajo y su planificación en horas vienen determinadas a continuación:

Desde el 28 de septiembre hasta el 12 de octubre se realizará la PEC 1 que consiste en realizar el plan de trabajo y se refleja con las siguientes tareas y subtareas que se desglosan a continuación:

<b>PEC 1 – Plan de Trabajo (120 h)</b>	
<b>Tarea</b>	<b>Duración (horas)</b>
<b>Escoger Temática del TFG</b>	<b>8</b>

<b>Plan de Trabajo</b>	96
<b>Contexto y justificación del trabajo</b>	56
<b>Objetivos del trabajo</b>	56
<b>Enfoque y método del trabajo</b>	24
<b>Planificación del trabajo</b>	24
<b>Sumario productos obtenidos</b>	16
<b>Descripción de los otros capítulos de la memoria</b>	16
<b>Pequeña App “Hello world” en Android Studio</b>	16

Tabla 6: Planificación PEC1 - Plan de Trabajo

Desde el 13 de octubre hasta el 9 de noviembre se realizará la PEC 2 que se fundamenta en el diseño y arquitectura de la App con las siguientes tareas y subtareas que se desglosan a continuación:

<b>PEC 2 – Diseño y Arquitectura (224 h)</b>	
<b>Tarea</b>	<b>Duración (horas)</b>
<b>Fases DCU</b>	224
<b>Análisis</b>	64
<b>Diseño</b>	120
<b>Diseño conceptual</b>	120
<b>Prototipado</b>	120
<b>Evaluación</b>	40
<b>Diseño técnico App</b>	224
<b>Definición casos de uso</b>	64
<b>Diseño arquitectura</b>	160

Tabla 7: Planificación PEC2 - Diseño y Arquitectura

Desde el 10 de noviembre hasta el 21 de diciembre se realizará la PEC 3 que tratará de la implementación de la App. Posee las siguientes tareas y subtareas que se desglosan a continuación:

<b>PEC 3 – Implementación (330 h)</b>	
<b>Tarea</b>	<b>Duración (horas)</b>
<b>Primera construcción</b>	56
<b>Test primera construcción</b>	40
<b>Elaboración juego de pruebas</b>	8
<b>Análisis de calidad (QA)</b>	32
<b>Segunda construcción</b>	56
<b>Test segunda construcción</b>	40
<b>Revisión juego de pruebas anterior</b>	8
<b>Análisis de calidad II (QA2)</b>	32
<b>Construcción final y entrega App Beta</b>	120

*Tabla 8: Planificación PEC3 - Implementación*

Desde el 22 de diciembre hasta el 9 de enero se realizará la Entrega final en la que se deberá proporcionar una memoria, la App, un video explicativo y la revisión por parte de los profesores. Posee las siguientes tareas que se muestran a continuación:

<b>Entrega final (258 h)</b>	
<b>Tarea</b>	<b>Duración (horas)</b>
<b>Últimos retoques memoria</b>	152
<b>Ultimos retoques Desarrollo App</b>	153
<b>Elaboración vídeo de presentación</b>	152
<b>Revisión profesorado</b>	104

*Tabla 9: Planificación de la Entrega final*

Desde el 23 de enero hasta el 27 de enero se realizará la Defensa virtual, en la que se responderá a una serie de preguntas por parte de los profesores acerca del trabajo realizado. Posee la siguiente tarea que se muestra continuación:

<b>Defensa virtual (40 h)</b>	
<b>Tarea</b>	<b>Duración (horas)</b>
<b>Defensa virtual</b>	40

*Tabla 10: Planificación Defensa virtual*

A continuación, mostramos las tareas anteriores mediante un diagrama de Gantt:

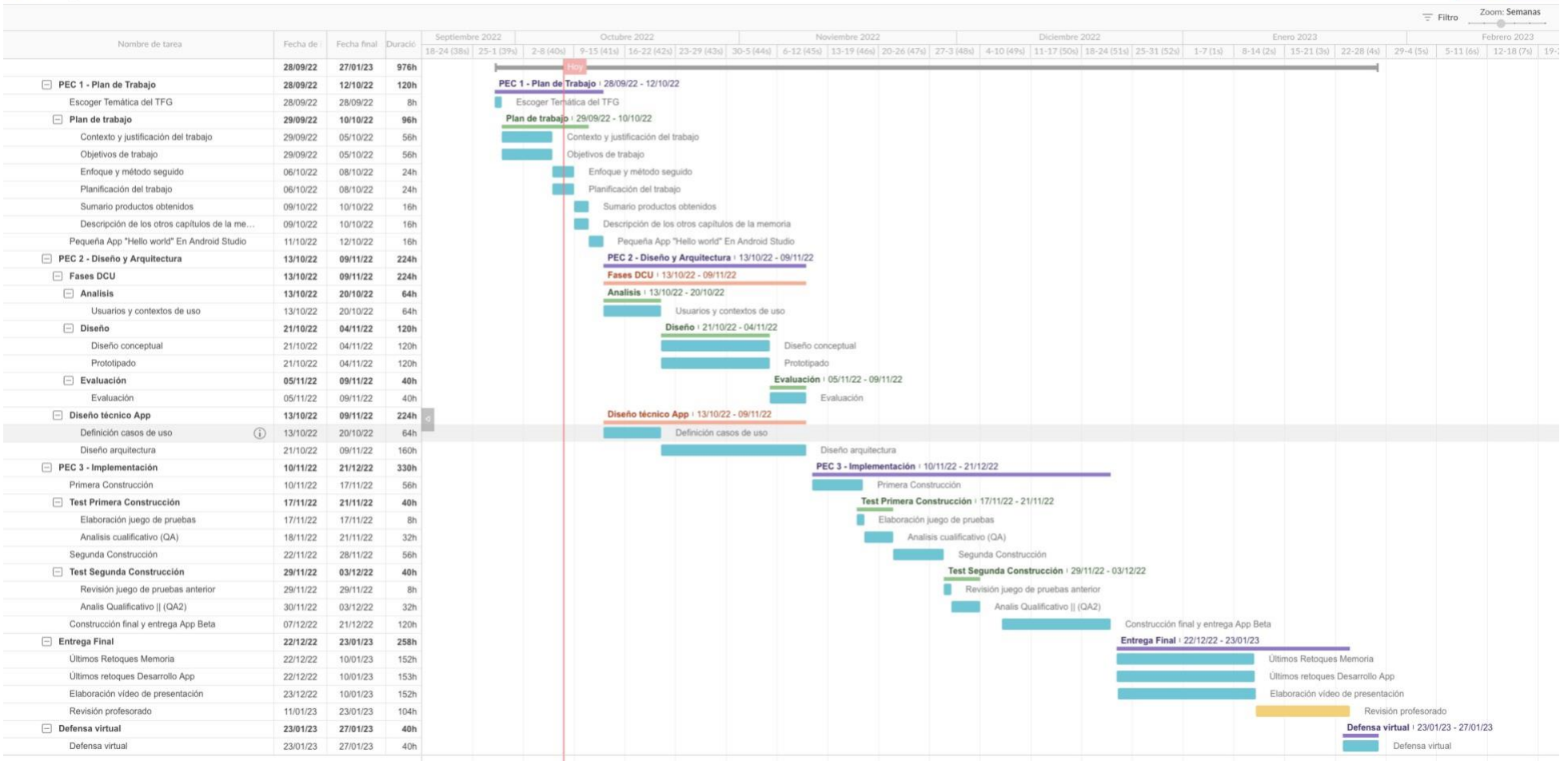


Figura 11: Diagrama de Gantt de planificación de tareas

## 1.5 Breve resumen de productos obtenidos

- Memoria del Trabajo Final de Grado:  
Documentación completa elaborada por el estudiante del trabajo realizado.
- Vídeo Demostrativo y Presentación:  
Video grabado por el estudiante demostrando y presentando el trabajo realizado
- Prototipo de diseño:  
Prototipo y documentación de la aplicación.
- Modelo de Diseño:  
Diseño documentado de la aplicación.
- Modelo de Diseño con Interacción.  
Diseño interactivo de la aplicación que facilitará la comprensión de los distintos componentes y transiciones.
- Aplicación Final:  
Paquete de código listo para producción con la aplicación desarrollada.
- Manual de usuario de la aplicación:  
Breves instrucciones de como el usuario debe usar la aplicación.

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

- Usuarios y Contextos de Uso:  
En este capítulo se lleva a cabo un análisis de los usuarios voluntarios sobre aplicaciones relacionadas con la App a conseguir, atendiendo a las tareas, objetivos y funcionalidad. Además, se define cada uno de estos usuarios mediante fichas de persona y escenario, el resumen de entrevista llevado a cabo, el contexto de uso de la aplicación, etc.
- Diseño conceptual:  
Se recoge la definición de la arquitectura de la información de la aplicación, el esquema y la navegación por los diferentes contenidos y pantallas. Asimismo, se definen las características gráficas de la interfaz, considerando la información recogida de los análisis, modelado y diseño conceptual.
- Prototipado:  
Consta de los prototipos realizados para la App. Se parte de la idea de realizar los primeros esbozos con el fin de obtener un reflejo lo más parecido a la realidad de la aplicación que queremos alcanzar.

- **Evaluación:**  
En esta sección se realizará una evaluación del prototipo utilizando técnicas específicas que serán valoradas por los usuarios voluntarios.
- **Diseño Técnico:**  
En este capítulo se incluyen los casos de uso y flujos de información. Además, abarca todo lo relativo a la construcción de la App: modelo de datos, diagramas de clases, librerías empleadas, etc.
- **Implementación:**  
Se describirá el entorno de desarrollo que se ha utilizado y los puntos más importantes de la implementación del proyecto.
- **Pruebas:**  
En esta sección se explica cuales son las diferentes pruebas realizadas para lograr tener un producto final de calidad.
- **Valoración económica:**  
Se definirá que estrategia seguir para ganar adeptos a la aplicación y conseguir beneficios. Así como un plan de marketing que permita dar a conocer al producto y el modo de distribución.
- **Conclusiones y Líneas Futuras:**  
Consta de 4 partes. En primer lugar, se resumen las conclusiones en el desarrollo general del proyecto. En el segundo punto, se reflexiona sobre los logros alcanzados con respecto a los objetivos planteados. A continuación, en el tercero se analiza el seguimiento realizado en la planificación y metodología empleada. Y finalmente, en el último punto se reflexiona sobre posibles líneas de futuro para la aplicación y su entorno.
- **Glosario:**  
Definición de términos y acrónimos más importantes utilizados en la memoria.
- **Bibliografía:**  
Contiene un listado numerado de referencias bibliográficas que se han utilizado en la memoria del proyecto. Siguiendo a continuación la bibliografía necesaria para la realización del producto.

## 2. Usuarios y Contexto de Uso

### 2.1 Investigación / Tomas de Datos

#### 2.1.1 Justificación y proceso de planteamiento

Para tantear a los usuarios con respecto al tema de la gestión de alimentación y la realización de la compra se ha elegido un método de indagación cuantitativo de tipo encuesta y uno cualitativo como es la entrevista profunda. La encuesta proporciona una idea general del modo de actuar del usuario, mientras que la entrevista profunda nos permite conocer más personalmente y abiertamente al individuo. Es por ello que una combinación de ambas proporciona un resultado más completo.

#### 2.1.2 Desarrollo y resultados de la encuesta

Hemos realizado 11 preguntas tipo test a 55 personas (la mayoría mujeres) mediante la herramienta gratuita online freeonlinesurveys [4] (ver documentos Anexo I y Anexo II) sobre el modo de organizar la compra de los alimentos y su uso. Como, por ejemplo: la frecuencia de compra, si existe algún sistema de comunicación en caso de realizarse entre varios o si leen la información nutricional de los productos. Previamente se ha notificado que la encuesta es anónima y rápida de hacer al usuario.

En la encuesta realizada se ha podido observar que la mayoría de la gente tiene un buen nivel tecnológico. Muchas viven solas, pero se organizan la compra mediante una lista previa con frecuencia de una semana. Asimismo, las personas con más edad son las que más compran por internet, mientras que las más jóvenes (menos de 35) acostumbran a realizar sus compras en un supermercado o hipermercado.

También se puede observar algunas dificultades con el manejo de la fecha de vencimiento de los productos alimentarios, aunque la mayoría suele controlarlo bien. Además, suelen planificar al detalle, el consumo de lo que van a comer durante toda la semana. Es de interés también observar que la gran mayoría de usuarios conocen la diferencia entre fecha de consumo preferente y fecha de vencimiento.

#### 2.1.3 Desarrollo y resultados de la entrevista profunda

Por otro lado, hemos entrevistado virtualmente a diversas personas de diferentes rangos de edades por redes sociales con preguntas abiertas sobre sus hábitos de compras, gestión, etc. con el objetivo de conocer un poco más al usuario. En



general todos tienen un buen manejo tecnológico y tienen alguna dificultad con la fecha de caducidad.

Las personas más jóvenes acostumbran a comprar lo que necesitan el mismo día, sin necesidad de planificar a la larga. No obstante, las personas más mayores planifican más detalladamente la compra una vez por semana. Dependiendo la zona donde viven suelen realizar la compra en pequeños comercios o supermercados. Aunque la gran mayoría lo hace en este último. En el caso de las personas que viven solas, existe más espacio en la despensa y frigorífico para guardar más cantidad de alimentos.

Los más jóvenes no suelen complicarse la vida y van a lo práctico, de esta forma casi no realizan recetas. Acostumbran a comer sano y a vivir la mayoría solos, aunque algunos aún no se han emancipado. Por el contrario, las personas mayores acostumbran a estar casados y compran alimentos para toda la familia. En todo momento, cuidan con mucho esmero la alimentación para los suyos. También realizan más recetas y prueban más variedades culinarias.

Asimismo, las personas de más edad miran con más esmero los valores nutricionales en las etiquetas de los alimentos y son más conscientes de conceptos como la diferencia entre fecha de vencimiento y de consumo preferente.

#### 2.1.4 Conclusiones obtenidas

En general, se puede apreciar que los usuarios tienen problemas para controlar la fecha de caducidad de los alimentos, en consecuencia, una app como la nuestra sería adecuada. También se observa que la gran mayoría adquiere sus productos en supermercado o hipermercado, por consiguiente, un lector de código de barras sería adecuado en nuestra aplicación.

La frecuencia de compra es mayoritariamente de una vez a la semana, este hecho nos marca la temporalidad habitual de compra y consumo. Además, se observa que muchos de ellos elaboran recetas de cocina, por lo que iría bien que el sistema nos proporcionase ideas con los alimentos que todavía no se nos han caducado. Sabemos, asimismo, que muchos de ellos leen la etiqueta nutricional de los alimentos. Esto nos lleva a pensar en la posibilidad de incorporar dicha información por cada alimento en el aplicativo.

#### 2.2 Perfil de Usuario

Tras la realización de los métodos de indagación efectuados podemos elaborar cuatro perfiles como arquetipo de usuario que utilizarán la aplicación y que podemos ver en las siguientes fichas.

## 2.2.1 Ficha nº 1



### JOVENES MENORES DE 25

#### Presentación

*El colectivo está formado por jóvenes de menos de 25 años con un nivel de manejo tecnológico alto. Suelen ser personas que se encuentran desarrollando sus estudios universitarios y acostumbran a vivir solos. Les gusta comer fuera y cuando van a comprar suelen improvisar sus compras en el momento que lo necesitan. Bastante a menudo tienen problemas con la caducidad de los productos, en especial la fruta, que siempre se les estropea*

#### Motivaciones

Viajar, comer fuera,  
las redes sociales...

#### Frustraciones

El empleo, la vivienda...

#### Demografía

menos de 25 años  
soltero  
estudiante de carrera  
español  
vive solo

#### Metas

sacarse la carrera y  
encontrar un buen trabajo

#### Habilidades



#### Problemas con fecha de vencimiento

con frecuencia

#### Frecuencia de compra

1 vez a la semana

#### Organización

compran sin planificar, según necesitan

#### Lugar de compra

supermercado o hipermercado

#### Leen etiqueta de compra

no siempre

#### Diferencian entre fecha de vencimiento y preferente de consumo

la mayoría sí

Figura 12: Ficha perfil – Jóvenes menores de 25

#### Contexto de Uso:

La aplicación se utilizará en casa, que es el lugar donde se almacenan los productos y en el supermercado o hipermercado, donde los jóvenes suelen planificar más sus compras. En ambos casos hemos de tener en cuenta que el usuario puede estar cogiendo con una mano productos de una estantería y con la otra utilizando el móvil. Esto provoca que la inserción de datos deba ser cómoda y rápida. Por otra parte, en el supermercado se es susceptible a distracciones por lo que la información deberá ser clara y sencilla. El dispositivo será un dispositivo Smartphone, por ser el más cómodo de transportar.

#### Análisis de tareas:

Los usuarios necesitarán poder añadir y borrar productos, consultar la lista de productos con sus precios. Deberán poder también poder consultar los productos caducados y a punto de caducar para su control. Asimismo, se podrá tachar los productos ya consumidos.

#### Listado de características necesarias:

- Lista productos caducados
- Lista productos a punto de caducar
- Lista productos consumidos
- Lista de compra
- Notificación en barra de tareas del móvil cuando un producto ha caducado.

## 2.2.2 Ficha nº 2



### JOVENES ENTRE 25 Y 35

#### Presentación

*Los jóvenes entre 25 y 35 años son personas con trabajos más estables que los menores de 25 y acostumbran a vivir en pareja, pero aún no tienen hijos, en la mayoría de casos. Sus habilidades y conocimientos con las tecnologías son elevados. Les gusta las excursiones, el baile y la comida sana. También, acostumbran a comprar 1 vez a la semana en supermercados e hipermercados, según necesitan. No suelen hacer recetas complicadas y suelen llevarse algún táper o fruta para comer en el trabajo.*

#### Demografía

entre 25 y 35 años  
en pareja  
informáticos, psicólogos,  
fisioterapeutas...  
español  
vive en compañía

#### Metas

comprarse una vivienda y  
tener hijos

#### Habilidades



#### Motivaciones

excursiones, baile, la  
comida sana

#### Frustraciones

el dinero, el deporte

#### Problemas con fecha de vencimiento

a veces

#### Frecuencia de compra

1 vez a la semana

#### Organización

*compran planificando y a la hora de preparar la comida con lo que tienen en casa, improvisan según necesitan*

#### Lugar de compra

supermercado o hipermercado

#### Leen etiqueta de compra

casi siempre

#### Diferencian entre fecha de vencimiento y preferente de consumo

la mayoría sí

Figura 13: Ficha perfil – Jóvenes entre 25 y 35

#### Contexto de Uso:

La aplicación se utilizará en casa, que es el lugar donde se almacenan los productos, el trabajo y en el supermercado o hipermercado, donde suelen planificar más sus compras. En el primer y último caso hemos de tener en cuenta que el usuario puede estar cogiendo con una mano productos de una estantería y con la otra utilizando el móvil. Esto provoca que la inserción de datos deba ser cómoda y rápida. Por otra parte, en el supermercado se es susceptible a distracciones por lo que la información deberá ser clara y sencilla, mientras que en el trabajo la inserción debe ser rápida para introducir los datos en poco tiempo y volver a la tarea.

Utilizarán la App cuando hacen la compra en el supermercado. Y después en casa, cuando estén colocando los productos en la despensa o revisándolos e incluso en el tiempo libre en el trabajo. El dispositivo será un dispositivo Smartphone, por ser el más cómodo de transportar.

#### Análisis de tareas:

Los usuarios necesitarán poder añadir y borrar productos, consultar la lista de productos con sus precios. Deberán poder también poder consultar los productos caducados y a punto de caducar para su control. Asimismo, se podrá tachar los productos ya consumidos y consultar la información nutricional del producto.

Listado de características necesarias:

- Lista productos caducados
- Lista productos a punto de caducar
- Lista productos consumidos
- Lista de compra
- Notificación en barra de tareas del móvil cuando un producto ha caducado.
- Información nutricional del producto.

### 2.2.3 Ficha nº 3



#### Demografía

entre 36 y 45 años  
casado y con hijos  
informáticos, psicólogos,  
amas de casa...  
español  
vive en familia

#### Metas

disfrutar del día a día con  
la familia

#### Habilidades



## ADULTOS ENTRE 36 Y 45

#### Presentación

*Los adultos entre 36 y 45 años son personas con trabajos estables que acostumbran a vivir en pareja y tener hijos. Sus habilidades y conocimientos con las tecnologías son un poco inferior a los anteriores, pues se desenvuelven bien con ellas, pero no con las tecnologías más recientes. Se sienten cómodos con los temas familiares y las relaciones con las personas. Si tienen tiempo acostumbran a realizar recetas un poco más complicadas en casa.*

#### Motivaciones

la familia, las  
personas, viajar

#### Frustraciones

las enfermedades

#### Problemas con fecha de vencimiento

frecuentemente

#### Frecuencia de compra

1 vez a la semana, aunque también cada 2 o 4 días

#### Organización

*compran planificando y a la hora de planificar que comer con lo que tienen en casa, improvisan según necesitan*

#### Lugar de compra

supermercado o hipermercado, aunque también algunos compran por Internet

#### Leen etiqueta de compra

aproximadamente el 50% de ellos sí

#### Diferencian entre fecha de vencimiento y preferente de consumo

la mayoría sí

Figura 14: Ficha perfil – Adultos entre 36 y 45 años

Contexto de Uso:

La aplicación se utilizará en casa, que es el lugar donde se almacenan los productos, en el trabajo, durante el tiempo de ocio y en el supermercado o hipermercado, donde suelen planificar más sus compras. En el primer y último caso hemos de tener en cuenta que el usuario puede estar cogiendo con una mano productos de una estantería y con la otra utilizando el móvil. Esto provoca que la inserción de datos deba ser cómoda y rápida. Por otra parte, en el supermercado se es susceptible a distracciones por lo que la información deberá ser clara y sencilla. El dispositivo será un dispositivo Smartphone, por ser el más cómodo de transportar.

## Análisis de tareas

Los usuarios necesitarán poder añadir y borrar productos, consultar la lista de productos con sus precios. Deberán poder también poder consultar los productos caducados y a punto de caducar para su control. Asimismo, se podrá tachar los productos ya consumidos y consultar la información nutricional del producto. También podrán consultar sugerencias de recetas con los productos que tengan en la lista de productos que no hayan caducado.

### Listado de características necesarias

- Lista productos caducados
- Lista productos a punto de caducar
- Lista productos consumidos
- Lista de compra
- Notificación en barra de tareas del móvil cuando un producto ha caducado.
- Información nutricional del producto.
- Consultar recetas sugeridas.

### 2.2.4 Ficha nº 4

## MAYORES DE 45



#### Demografía

más de 45 años  
casado y con hijos  
comerciantes, camareros,  
amas de casa...  
español  
vive en familia

#### Metas

disfrutar del día a día con  
la familia y tener buena  
salud

#### Habilidades



#### Presentación

*El grupo formado por personas de más de 45 años se trata de un conjunto que posee menos habilidades TIC conforme aumenta la edad. Planifican en menor medida que los anteriores y tienen un gran sentido del ahorro. Asimismo, conforme aumenta la edad, la forma física decae. Es por ello y por la falta de tiempo también, que aprovechan algunos para realizar las compras por Internet. Acostumbran a realizar recetas elaboradas cuando tienen tiempo libre y viven en familia*

#### Motivaciones

la familia, las personas, la  
comida casera

#### Frustraciones

las enfermedades, la falta de concentración,  
la forma física...

#### Problemas con fecha de vencimiento

pocas veces

#### Frecuencia de compra

1 vez a la semana, aunque también cada 2 o 4 días

#### Organización

*compran planificando y a la hora de planificar que comer con lo que tienen en casa, improvisan según necesitan*

#### Lugar de compra

supermercado o hipermercado y rara vez también por Internet

#### Leen etiqueta de compra

aproximadamente el 50% de ellos sí

#### Diferencian entre fecha de vencimiento y preferente de consumo

la mayoría sí

Figura 15: Ficha perfil - Mayores de 45 años

## Contexto de Uso:

La aplicación se utilizará en casa, que es el lugar donde se almacenan los productos el trabajo, tiempo de ocio y en el supermercado o hipermercado, donde suelen planificar más sus compras. En el primer y último caso hemos de tener en cuenta que el usuario puede estar cogiendo con una mano productos de una estantería y con la otra utilizando el móvil. Esto provoca que la inserción de datos deba ser cómoda y rápida. Por otra parte, en el supermercado se es susceptible a distracciones por lo que la información deberá ser clara y sencilla.

Utilizarán la App cuando hacen la compra en el supermercado. Y después en casa cuando estén colocando los productos en la despensa o revisándolos e incluso en el en el trabajo durante un breve descanso. El dispositivo será un dispositivo Smartphone, por ser el más cómodo de transportar.

## Análisis de tareas

Los usuarios necesitaran poder añadir y borrar productos, consultar la lista de productos con sus precios. Deberán poder también poder consultar los productos caducados y a punto de caducar para su control. Asimismo, se podrá tachar los productos ya consumidos y consultar la información nutricional del producto. También podrán consultar sugerencias de recetas con los productos que tengan en la lista de productos que no hayan caducado.

## Listado de características necesarias

- Lista productos caducados
- Lista productos a punto de caducar
- Lista productos consumidos
- Lista de compra
- Notificación en barra de tareas del móvil cuando un producto ha caducado.
- Información nutricional del producto.
- Consultar recetas sugeridas

### 3. Diseño conceptual

#### 3.1 Escenarios de uso

Se define a continuación los escenarios en los que los diferentes perfiles harán uso de la aplicación y su contexto.

<b>ESCENARIO 1 – LISTA DE COMPRA</b>	
Perfil	Cualquiera de los 4 perfiles
Donde	Perfil Menores de 25: En la universidad o en casa Los otros perfiles: En casa o el trabajo
Cuando	En cualquier momento
Por qué	Al usuario se le puede olvidar comprar un alimento que necesite.
Objetivos	Tener una lista de alimentos a comprar en el momento de estar en el supermercado o hipermercado.
Tareas	Añadir, consultar, modificar y eliminar un producto.
Necesidades de información	Confirmación de que la tarea se ha realizado. Lista de productos para comprar.
Funcionalidades	Añadir, consultar, editar y eliminar en la base de datos de productos de compra.
Como desarrolla las tareas	Mediante la pantalla y el teclado de un dispositivo móvil Android.

Tabla 11: Escenario contexto de uso - Lista de compra

<b>ESCENARIO 2 – DESPENSA</b>	
Perfil	Cualquiera de los 4 perfiles
Donde	En el supermercado y en casa
Cuando	Cuando el usuario se encuentra cogiendo los productos y escaneándolos
Por qué	Es útil llevar un control de los productos alimentarios que disponemos en casa para que no se estropeen,
Objetivos	Tener un registro de los alimentos que se tienen o tendrán en casa, su precio y el total de lo que hemos gastado.

Tareas	Añadir, consultar, modificar y eliminar un producto.
Necesidades de información	Confirmación de que la tarea se ha realizado. Información del precio, cantidad y total gastado. Lista de productos en despensa.
Funcionalidades	Añadir, consultar, editar y eliminar en la base de datos de productos de despensa. Escanear producto. Capturar fecha de vencimiento de un alimento mediante una imagen. Que la app obtenga el nombre del producto a partir del código de barras escaneado mediante un servicio externo.
Como desarrolla las tareas	Mediante la pantalla, el teclado y la cámara de un dispositivo móvil Android.

Tabla 12: Escenario contexto de uso - Despensa

<b>ESCENARIO 3 – PRODUCTOS CADUCADOS</b>	
Perfil	Cualquiera de los 4 perfiles
Donde	En cualquier lugar
Cuando	En cualquier momento
Por qué	Para saber que productos se encuentran incomedibles.
Objetivos	Saber que productos de los que se disponen en casa se encuentran caducados.
Tareas	Consultar productos caducados
Necesidades de información	Información del precio total de alimentos adquiridos caducados Lista de productos caducados.
Funcionalidades	Consulta en base de datos de productos caducados. Notificación mediante correo y mensaje en barra de tareas del móvil de que producto ha caducado.
Como desarrolla las tareas	Mediante la pantalla de un dispositivo móvil Android.

Tabla 13: Escenario contexto de uso - Productos caducados

<b>ESCENARIO 4 – SUGERENCIA RECETAS</b>	
Perfil	Cualquiera de los 4 perfiles



Donde	En casa
Cuando	Cuando el usuario se disponga a cocinar y no sepa que hacer.
Por qué	Porque el usuario muchas veces no tiene una idea de que realizar con los productos alimentarios de que dispone en casa
Objetivos	Obtener sugerencias de recetas que lleven ingredientes disponibles en despensa.
Tareas	Obtener ingredientes de la receta que se encuentren en la lista de despensa y no estén caducados. Obtener directrices para realizar la receta.
Necesidades de información	Ingredientes, cantidades y directrices de la receta.
Funcionalidades	Cruzar datos de ingredientes de recetas disponibles con los productos alimentarios en despensa disponibles no caducados y proporcionar los resultados.
Como desarrolla las tareas	Mediante la pantalla de un dispositivo móvil Android.

Tabla 14: Escenario contexto de uso - Sugerencia recetas

<b>ESCENARIO 5 – INFORMACIÓN NUTRICIONAL</b>	
Perfil	Cualquiera de los 4 perfiles
Donde	En cualquier lugar
Cuando	En cualquier momento
Por qué	Para llevar una dieta o simplemente saber que se come.
Objetivos	Obtener la composición de un producto alimentario.
Tareas	Saber de que nutrientes esta compuesto un alimento. Saber que cantidad de cada nutriente tiene el alimento. Buscar por tipo de nutriente: básicos, vitaminas, minerales...
Necesidades de información	Componentes del alimento con su cantidad y unidad de medida.
Funcionalidades	Que la app adquiera mediante un servicio externo la composición del alimento.
Como desarrolla las tareas	Mediante la pantalla de un dispositivo móvil Android.

Tabla 15: Escenario contexto de uso - Información nutricional

### 3.2 Flujo de interacción

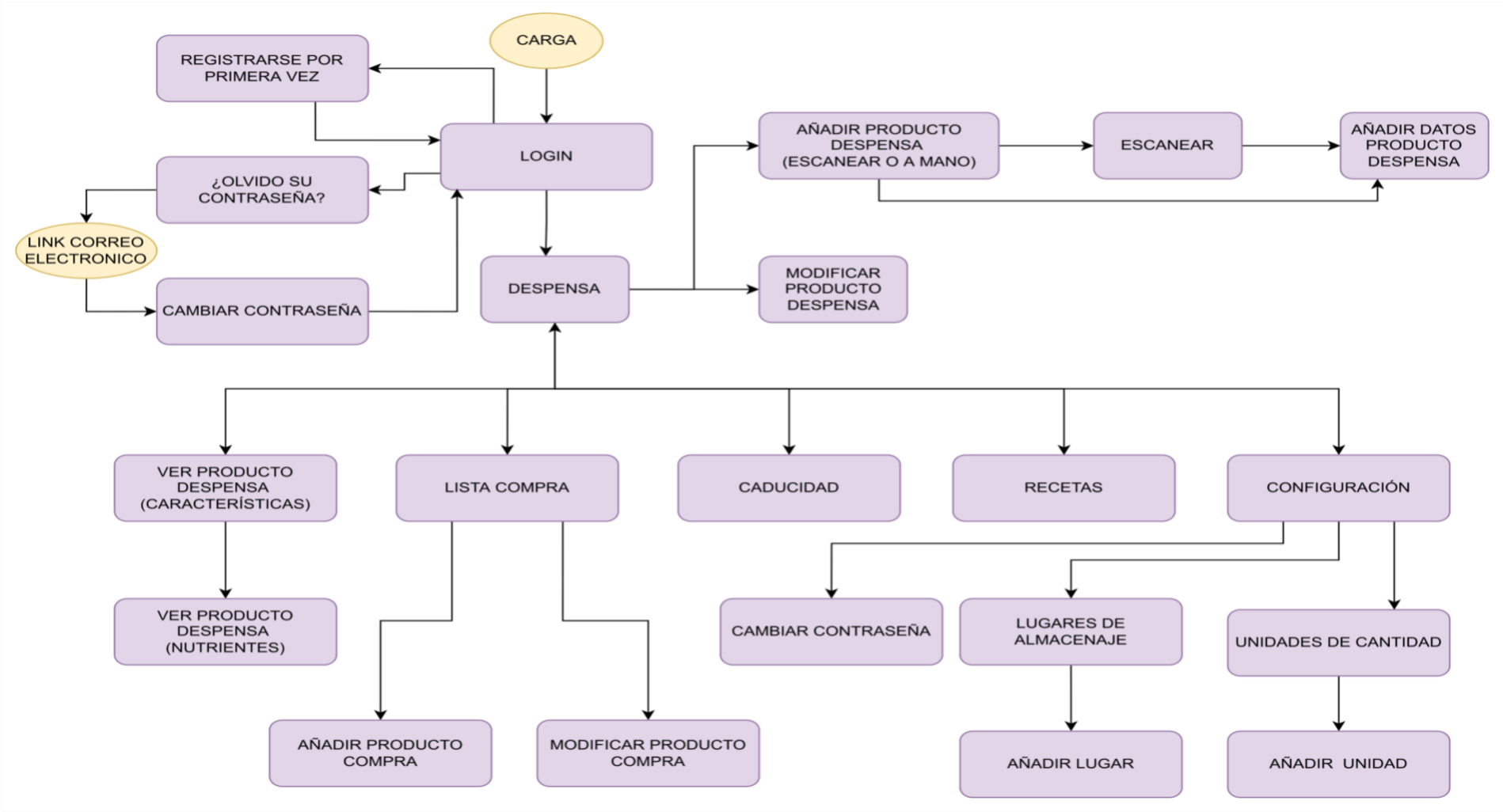


Figura 16: Flujo de interacción de la App

# 4. Prototipado

## 4.1 Sketches escaneados

Se muestran a continuación los primeros sketches a mano alzada de lo que será la App.

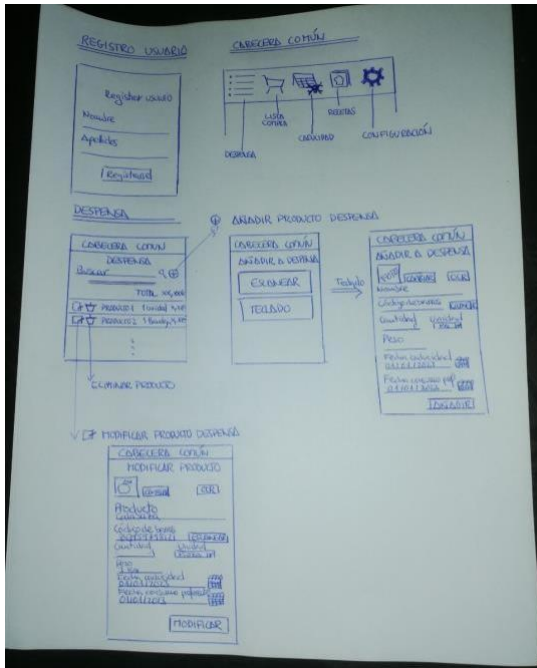


Figura 17: Sketch escaneado 1

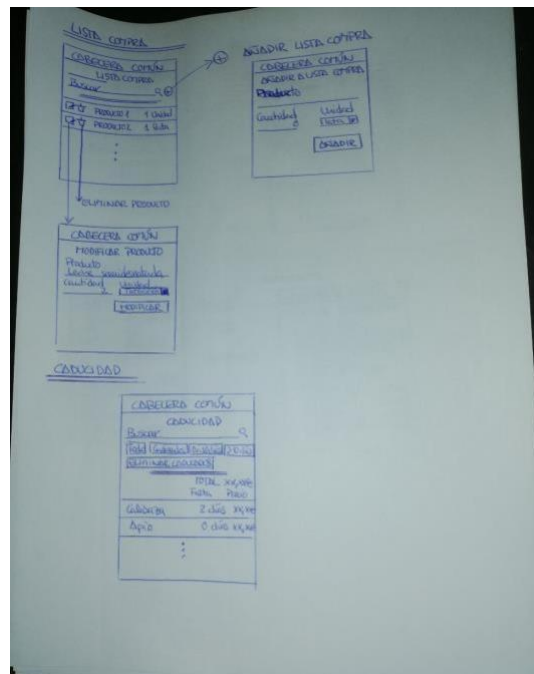


Figura 18: Sketch escaneado 2

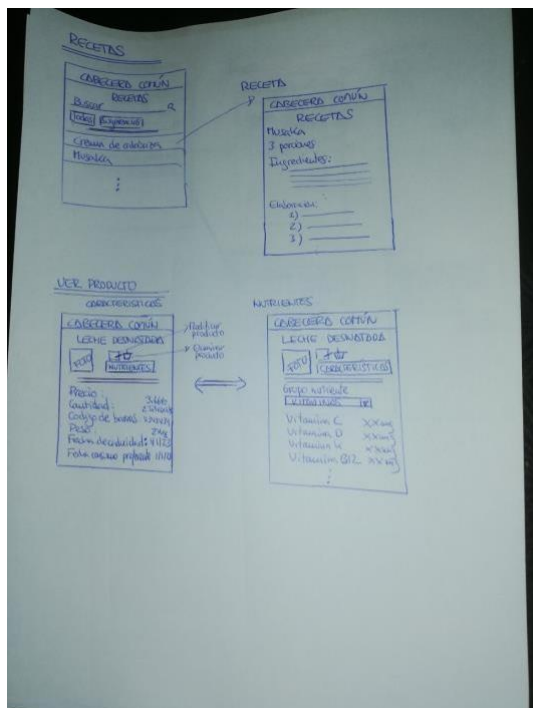


Figura 19: Sketch escaneado 3

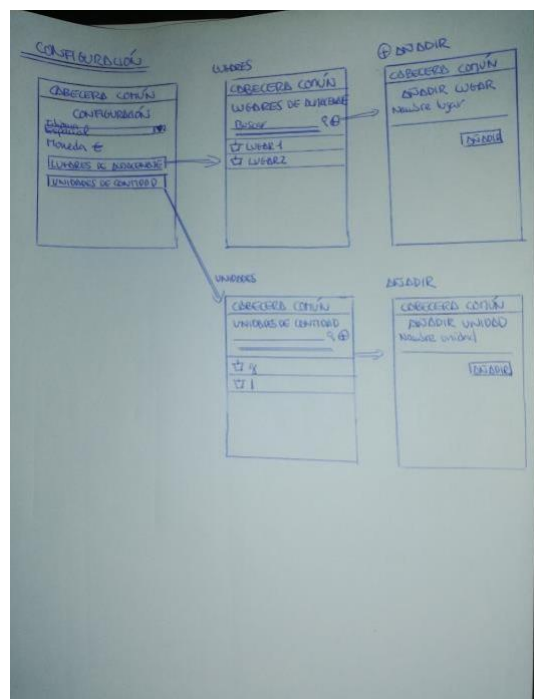


Figura 20: Sketch escaneado 4

## 4.2 Prototipo horizontal de alta fidelidad

Para la realización del prototipado horizontal se ha empleado la herramienta de uso gratuito online marvel. Puede verse su aspecto y navegación en el siguiente enlace:

<https://marvelapp.com/prototype/18818654>

### Log in

La primera pantalla que se observa nada más cargar la aplicación es la de log in. En ella será necesario introducir nuestro nombre de usuario y contraseña para entrar en la App para acceder a su uso.

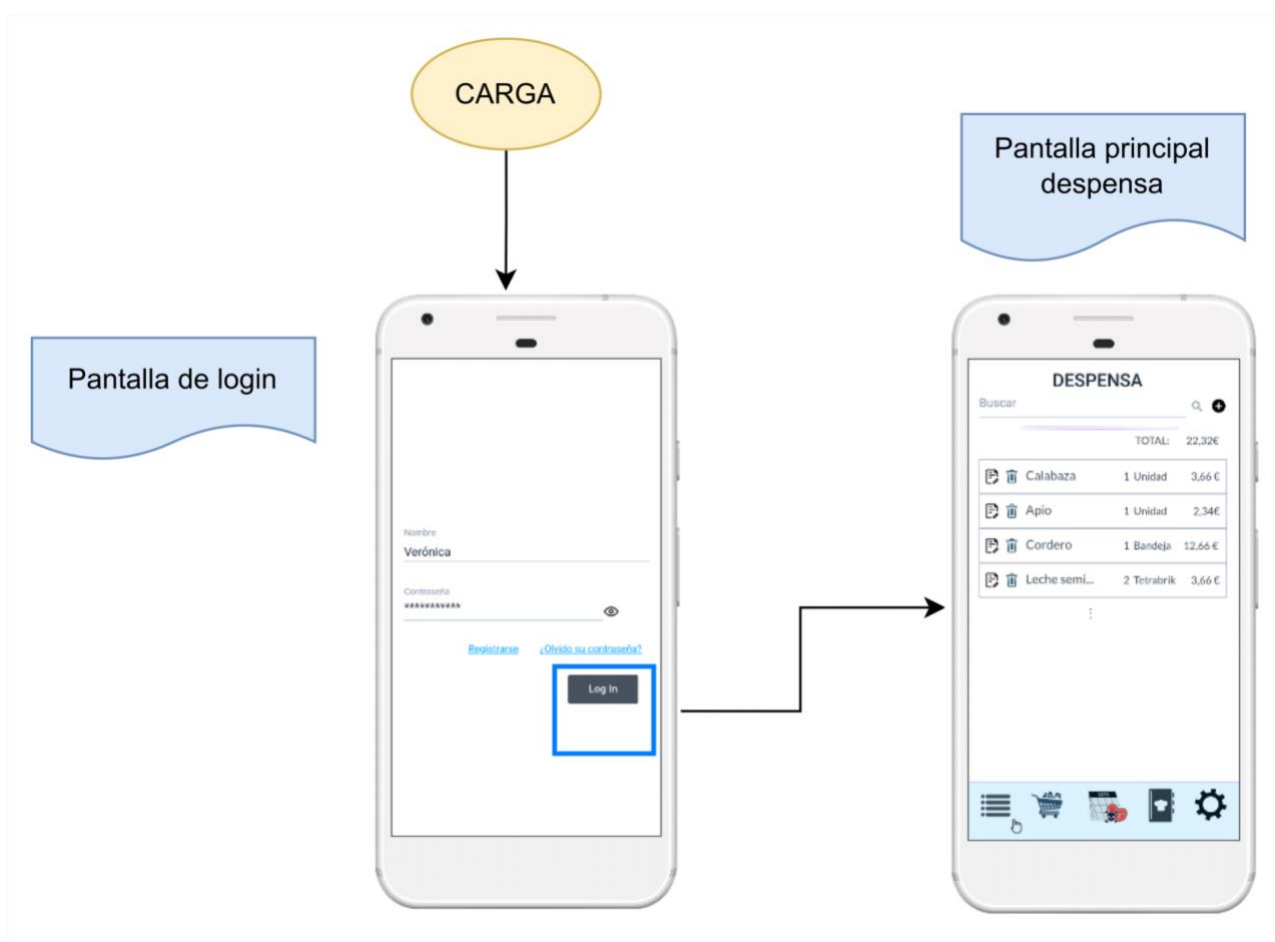


Figura 21: Prototipo Login

### Registrarse

Si es la primera vez que se utiliza la App es obligatorio el registro del usuario en la base de datos. Este paso crea un identificador de usuario que es almacenado en el dispositivo y en la base de datos externa una fila con los campos del

identificados, nombre, apellidos, email y contraseña. Después de realizar el paso, se vuelve a la pantalla de log in para acceder a la App.

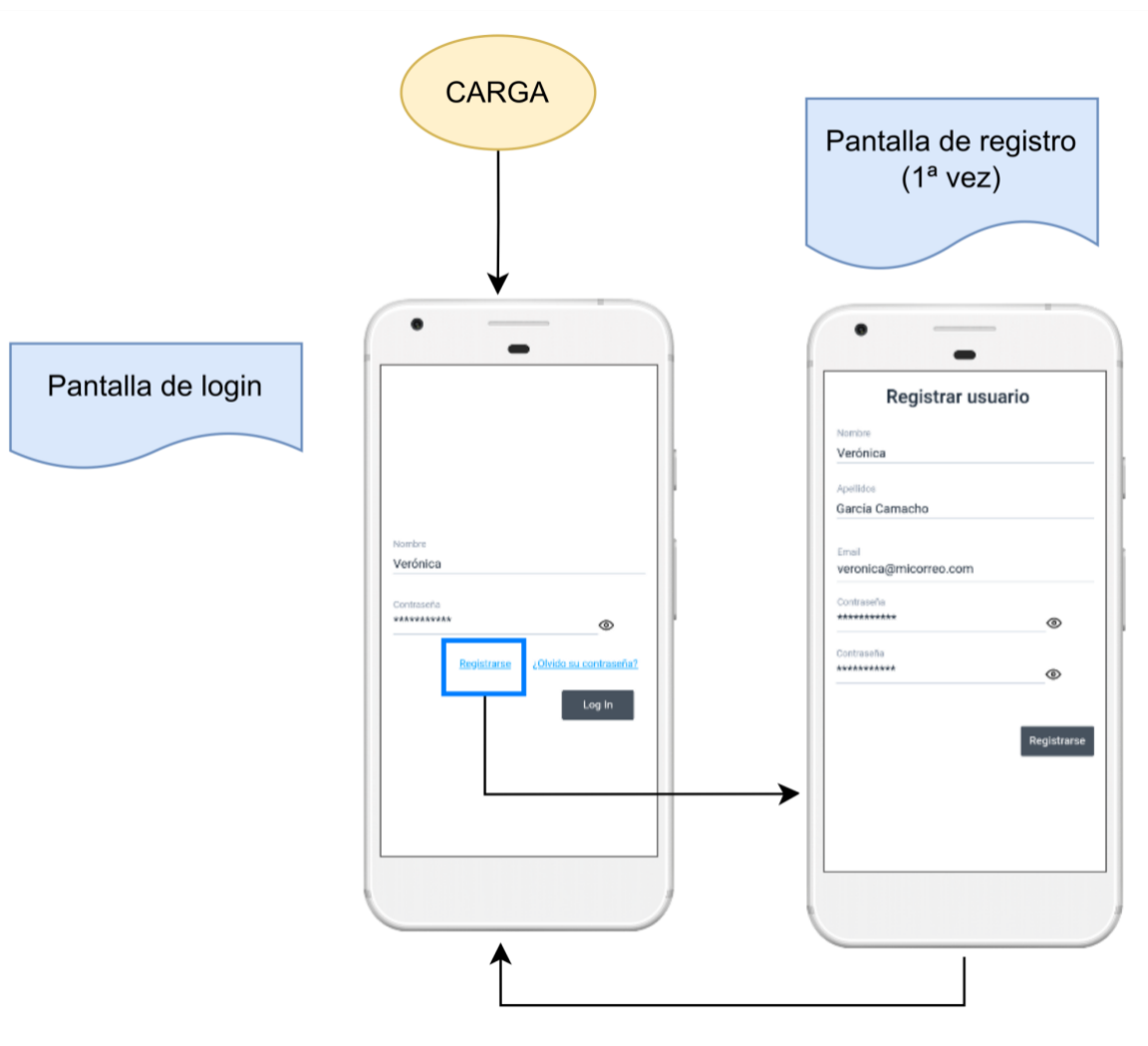


Figura 22: Prototipo Registrarse

### ¿Olvido su contraseña?

En la pantalla de log in es posible haber olvidado la contraseña de acceso. Es por ello que el sistema dispone de un proceso de restablecimiento de contraseña que consiste en enviar un enlace a la cuenta de correo indicada para poderla cambiar.

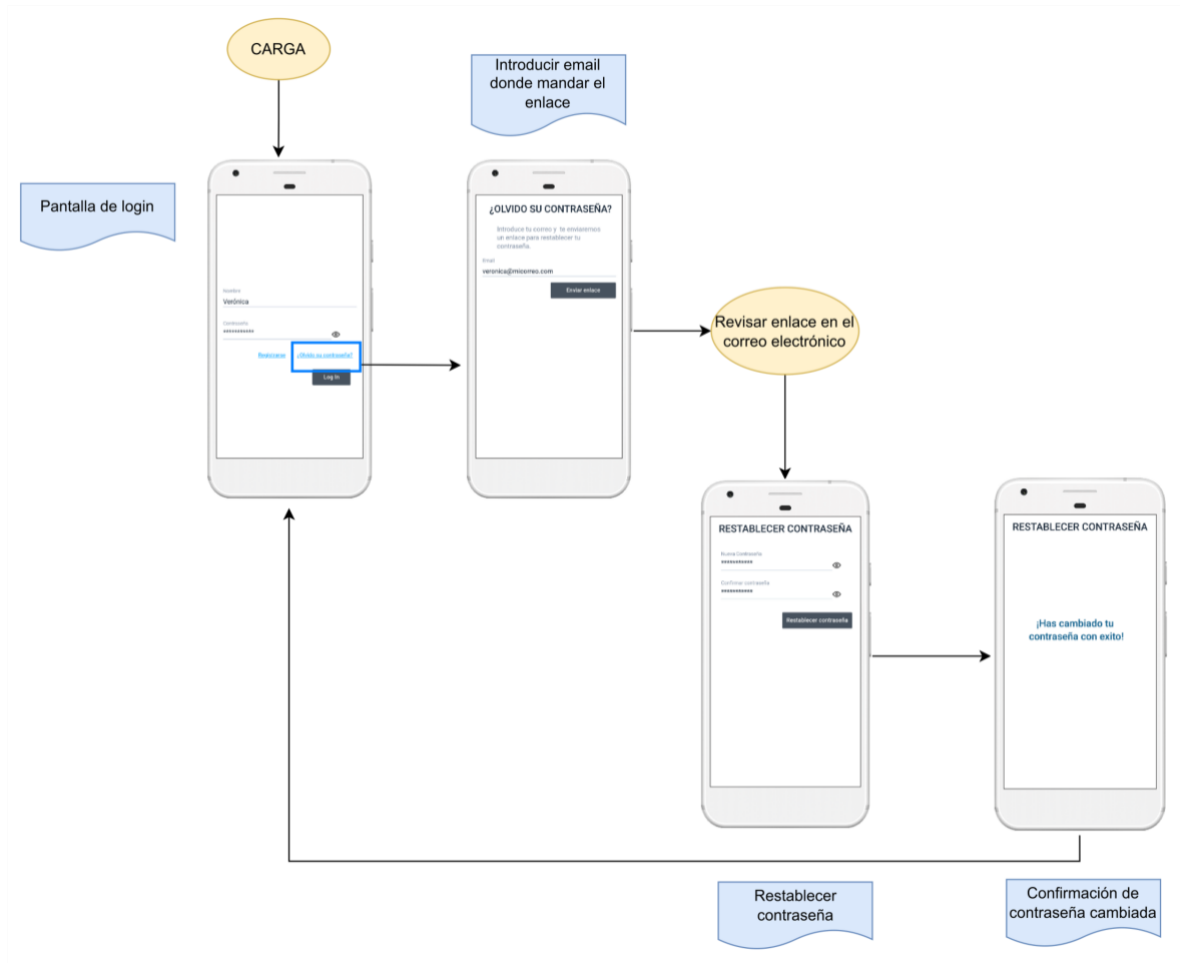


Figura 23: Prototipo "¿Olvido su contraseña?"

### Pantallas principales de navegación

Gracias al menú de la cabecera de abajo se puede navegar entre las diferentes pantallas principales: despensa, lista de compra, caducidad, recetas y configuración. La navegación no tiene un único sentido, ya que desde cualquiera de ellas se puede llegar a otra de las pantallas principales.

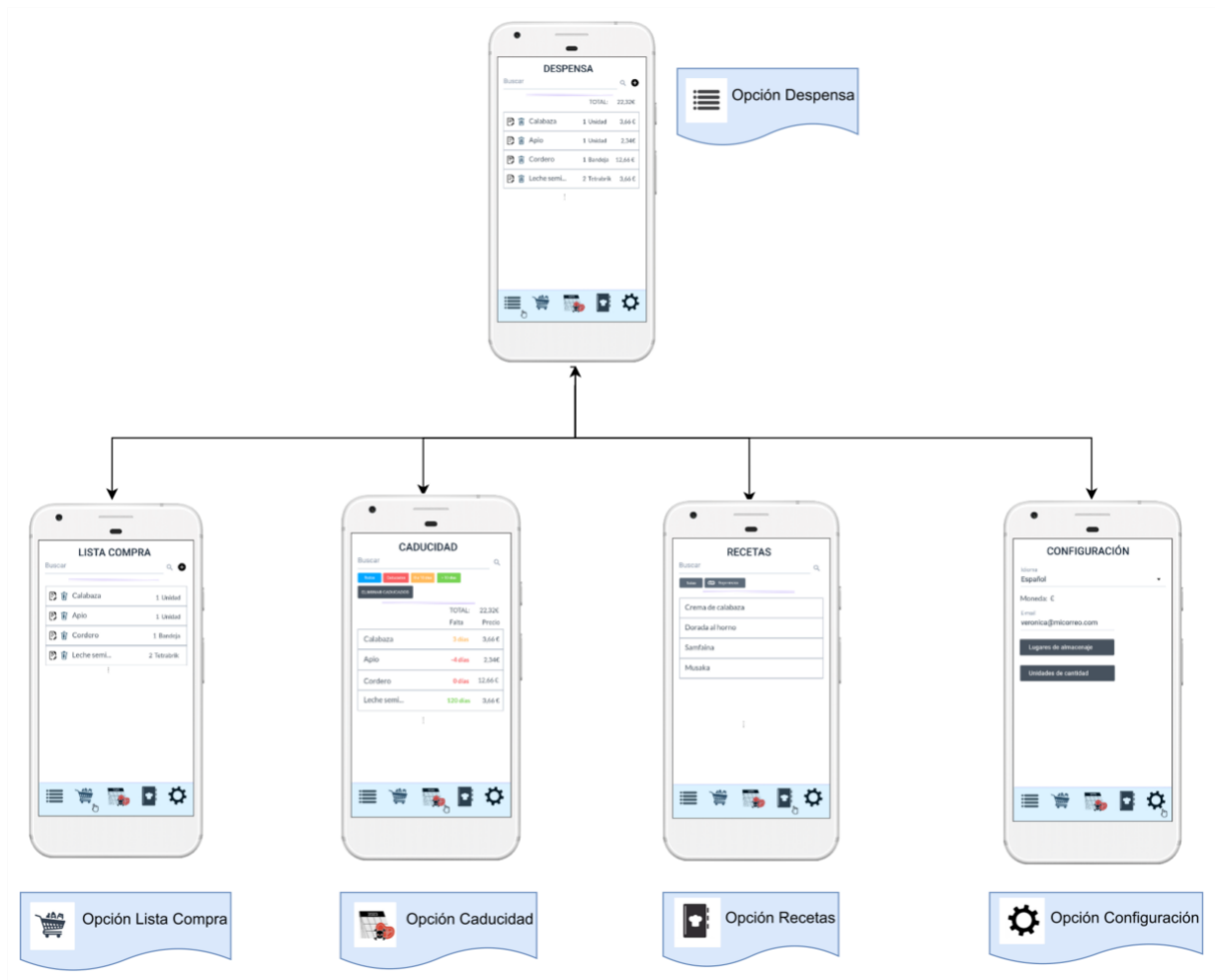


Figura 24: Prototipo Pantallas principales navegación

## Despesa

El apartado de despesa consta en un primer momento de la pantalla principal de despesa con buscador de productos en la lista, donde podemos realizar cuatro acciones:

- Añadir un producto a la despesa.
- Modificar un producto en la despesa.
- Eliminar un producto en la despesa.
- Ver un producto de la lista de despesa.

Al añadir un producto se pasa a una pantalla intermedia que pregunta si se quiere escanear un código de barras para identificar el producto o insertar directamente los datos a mano. Si se escoge escanear, rellena el nombre del producto y el código de barras y pasa a la pantalla de inserción de datos para continuar añadiendo los datos a mano, sino pasa directamente a ella y se introduce todo a mano.

Una vez dentro de la pantalla “Añadir a despensa”, se puede cambiar la foto del producto, facilitar la inserción de datos en los campos mediante una lectura OCR de texto de etiqueta en una captura de imagen o escanear el código de barras si no se ha realizado antes.

La pantalla de modificar es similar a la de añadir, solo que carga los datos existentes permitiendo modificarlos.

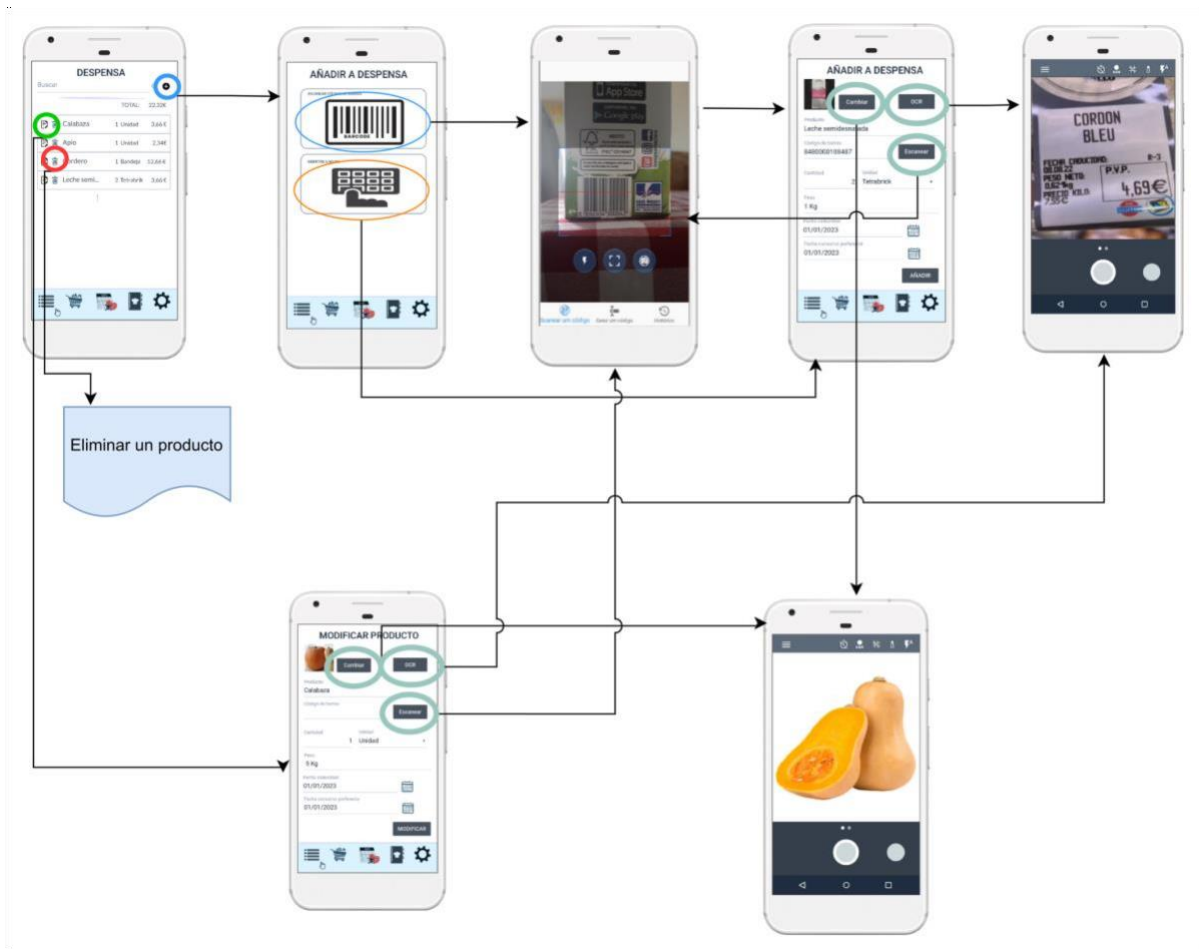


Figura 25: Prototipo Despensa

Si se toca un producto de la lista podemos ver sus características y nutrientes. También podemos ver la foto del producto y modificarlo o eliminarlo directamente.



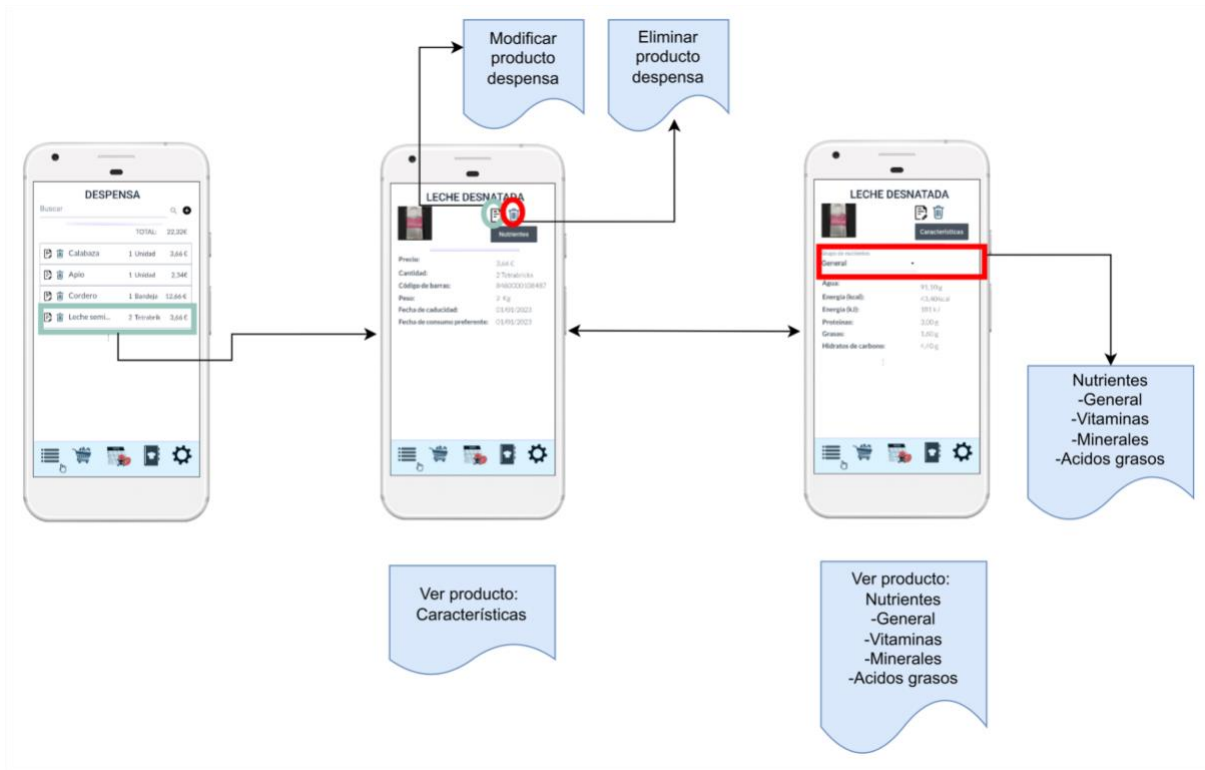


Figura 26: Prototipo Ver producto Despensa

## Lista de compra

La lista de la compra, consta de una pantalla con un buscador de los productos de la lista existente. Se puede añadir un nuevo producto dando paso a una nueva pantalla donde introducir el nombre y su cantidad requerida o modificarlo entrando en otra pantalla donde se cargan los datos existentes para cambiarlos.

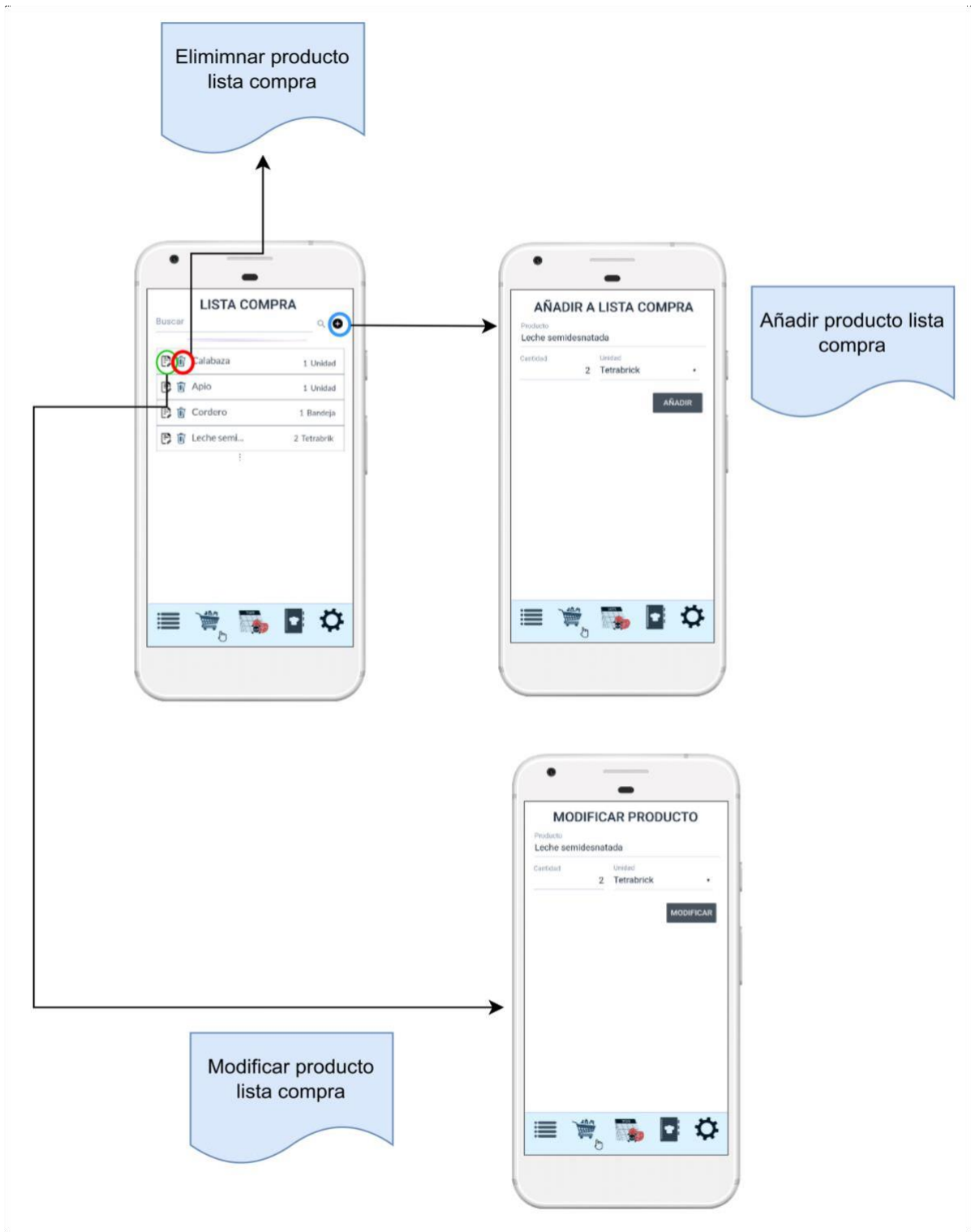


Figura 27: Prototipo Lista de compra

## Caducidad

El apartado de caducidad consta de una pantalla donde se aprecia el tiempo que le queda a cada producto para caducar. Se clasifican los productos en tres tipos:

- Caducados

- 0 a 10 días
- Más de 10 días

Asimismo, se dispone una opción para listar todos los productos existentes y un buscador para encontrar un producto concreto. Otro botón del que se dispone es el de eliminar los productos que están caducados una vez que ya los tenemos localizados y apartados.



Figura 28: Prototipo Caducidad

## Recetas

La sección de recetas permite listar todas las recetas de la base de datos en la cual se puede localizar mediante el buscador una concreta. También existe una opción de sugerencias que recoge todas las recetas que contengan ingredientes que se encuentren en la despensa.

Si se accede a una receta de la lista se ve sus porciones, ingredientes y elaboración.



Figura 29: Prototipo Recetas

## Configuración

En la configuración se aprecia la opción de cambiar el idioma de la interfaz de la aplicación, la moneda, el correo donde se notificará los productos caducados y la contraseña de acceso. Además, se puede añadir nuevos sitios de almacenaje o nuevas unidades de cantidad del producto. Para ello, accedemos a sus pantallas correspondientes.

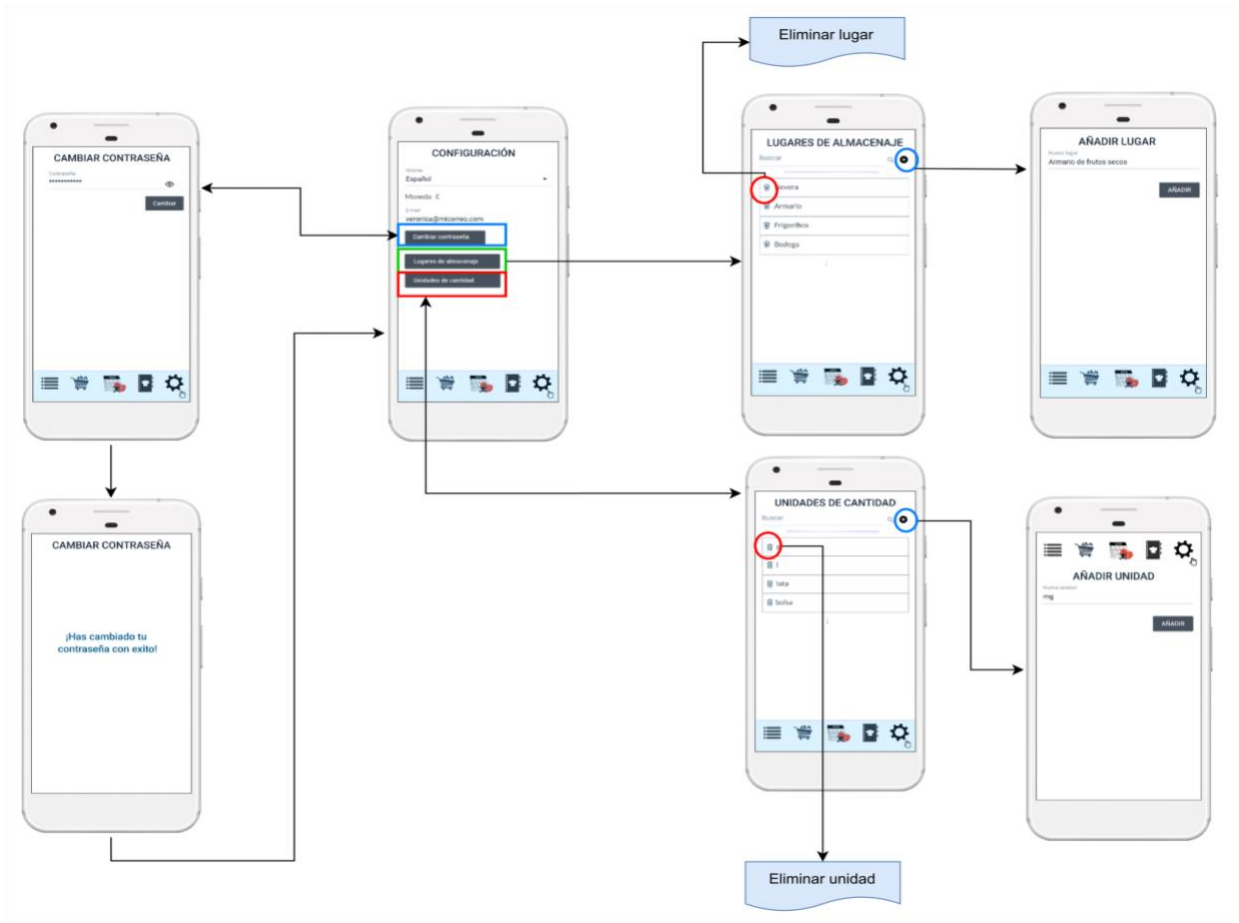


Figura 30: Prototipo Configuración

Además de las pantallas existentes cabe destacar la existencia de pop ups para dar retroalimentación al usuario cada vez que se lleve a cabo una acción para informar o pedir confirmación.

## 5. Evaluación

### 5.1 Recopilación de preguntas

- ¿Encuentra fácilmente las funcionalidades?
- ¿Cuánto tarda en encontrar lo que quiere hacer?
- ¿Resulta cómodo introducir la información a la hora de añadir o modificar un producto?
- ¿La App tiene pasos innecesarios cuando queremos llevar a cabo una tarea?
- ¿Es intuitiva?
- ¿Tarda demasiado en realizar una tarea?
- ¿Cómo encuentra la organización de los diferentes botones? ¿Es fácil y sencillo de manejar?
- ¿Encuentra la App no cerrada?, ¿Se puede navegar hacia todas las funcionalidades?
- ¿Ha encontrado una notificación con el/los producto/s caducado/s?
- ¿Mejoraría alguna funcionalidad? ¿Como?

### 5.2 Tareas que los usuarios deben realizar

- Tarea 1 – Entrar en la App.
- Tarea 2 – Añadir un producto a la despensa.
- Tarea 3 – Buscar un producto en la despensa
- Tarea 4 – Modificar un producto de la despensa.
- Tarea 5 – Eliminar un producto de la despensa.
- Tarea 6 – Ver características e información nutricional de un producto.
- Tarea 7 -- Añadir un producto a la lista de compra.
- Tarea 8 -- Buscar un producto en la lista de compra.
- Tarea 9 – Modificar un producto en la lista de compra.
- Tarea 10 – Eliminar un producto de la lista de compra.
- Tarea 11 – Consultar la caducidad de los productos de la despensa.
- Tarea 12 – Buscar un producto en la lista de productos con la caducidad de la despensa.
- Tarea 13 – Buscar una receta en la lista de recetas
- Tarea 14 – Consultar lista de recetas sugeridas con los productos en despensa.
- Tarea 15 – Consultar información de elaboración de una receta.
- Tarea 16 – Cambiar idioma de la App
- Tarea 17 – Cambiar contraseña de acceso
- Tarea 18 – Añadir un lugar de almacenaje
- Tarea 19 – Eliminar un lugar de almacenaje
- Tarea 20 – Añadir una unidad de cantidad
- Tarea 21 – Eliminar una unidad de cantidad.

### 5.3 Preguntas referentes a las tareas

- Tarea 1 – Entrar en la App.
  - ¿Se accede rápido?
  - ¿Si se te ha olvidado la contraseña, tienes opción a recuperarla?
  - ¿Es fácil registrarse?
  
- Tarea 2 – Añadir un producto a la despensa.
  - ¿Es rápido añadir un producto?
  - ¿Puedes subir una foto sin problemas?
  - ¿El botón OCR permite leer información textual en una imagen de interés?
  - ¿Puede escanear un producto y obtener el nombre de este y su código de barras?
  
- Tarea 3 – Buscar un producto en la despensa
  - ¿Puedes encontrar un producto que hayas añadido previamente en la lista de despensa?
  
- Tarea 4 – Modificar un producto de la despensa.
  - ¿Puedes modificar los datos de un producto existente en la lista de despensa?
  
- Tarea 5 – Eliminar un producto de la despensa.
  - ¿Puedes eliminar un producto existente en la lista de despensa?
  
- Tarea 6 – Ver características e información nutricional de un producto de la despensa.
  - ¿Puedes ver las características e información nutricional de un producto de despensa?
  - ¿Puedes cambiar el grupo de nutrientes y visualizar las cantidades de estos en el producto?
  
- Tarea 7 -- Añadir un producto a la lista de compra.
  - ¿Es posible añadir un producto a la lista de compra?
  
- Tarea 8 -- Buscar un producto en la lista de compra.
  - ¿Es posible encontrar en la lista de compra un producto añadido previamente?
  
- Tarea 9 – Modificar un producto en la lista de compra.
  - ¿Es posible modificar la cantidad o el nombre de un producto existente en la lista de compra?
  
- Tarea 10 – Eliminar un producto de la lista de compra.
  - ¿Es posible eliminar un producto existente de la lista de compra?
  
- Tarea 11 – Consultar la caducidad de los productos de la despensa.
  - ¿Puedes ver los días que faltan para que caduque y el precio del producto según la temporalidad escogida?
  - ¿Puedes ver el total gastado de los productos seleccionados?

- Tarea 12 – Buscar un producto en la lista de productos con la caducidad de la despensa.
  - ¿Puedes encontrar un producto existente en la lista de despensa desde el buscador de la página de caducidad.
- Tarea 13 – Buscar una receta en la lista de recetas
  - ¿Es posible buscar una receta existente en la lista de recetas del sistema?
- Tarea 14 – Consultar lista de recetas sugeridas con los productos en despensa.
  - ¿Puedes obtener una lista de sugerencias de recetas con al menos uno o más productos de los disponibles en despensa.
- Tarea 15 – Consultar información de elaboración de una receta.
  - ¿Puedes consultar la información de elaboración e ingredientes de una receta?
- Tarea 16 – Cambiar idioma de la App
  - ¿Es posible cambiar el idioma de la App?
- Tarea 17 – Cambiar contraseña de acceso
  - ¿Si cambias la contraseña y vuelves acceder de nuevo, no te deja entrar en la App hasta que introduces la nueva contraseña?
- Tarea 18 – Añadir un lugar de almacenaje
  - ¿Es posible añadir un nuevo lugar de almacenaje de producto?
- Tarea 19 – Eliminar un lugar de almacenaje
  - ¿Es posible eliminar un lugar de almacenaje existente de un producto?
- Tarea 20 – Añadir una unidad de cantidad
  - ¿Se puede añadir una nueva unidad de cantidad de producto?
- Tarea 21 – Eliminar una unidad de cantidad.
  - ¿Es posible eliminar una unidad de cantidad existente de un producto?

#### 5.4 Análisis de resultados e incluir mejoras

Tras analizar los resultados obtenidos en los test se han incluido algunas mejoras. En primer lugar, se ha cambiado el menú principal que en los sketches escaneados se situaban en la parte superior a la parte inferior por ser más cómodo. También se ha añadido una pantalla de log in con un apartado para registrarse y otra para restablecer la contraseña. Asimismo, se han añadido una serie de botones en la lista de productos con caducidad para poder localizar la temporalidad que falta en el vencimiento de cada uno de los productos de que disponemos en despensa.



# 6. Diseño Técnico

## 6.1 Definición de los casos de uso

### 6.1.1 Diagrama UML de Actores

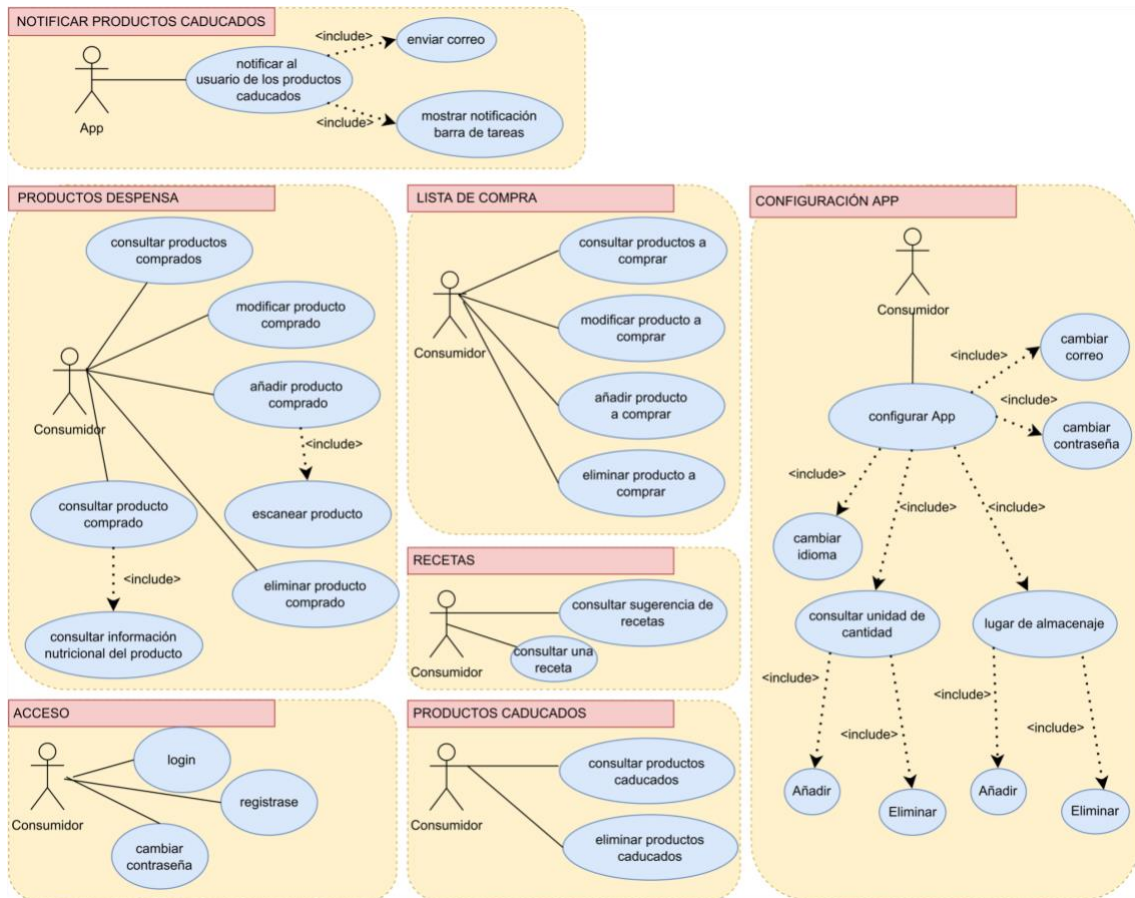


Figura 31: Diagrama UML de Actores

## 6.1.2 Diagrama UML Flujo

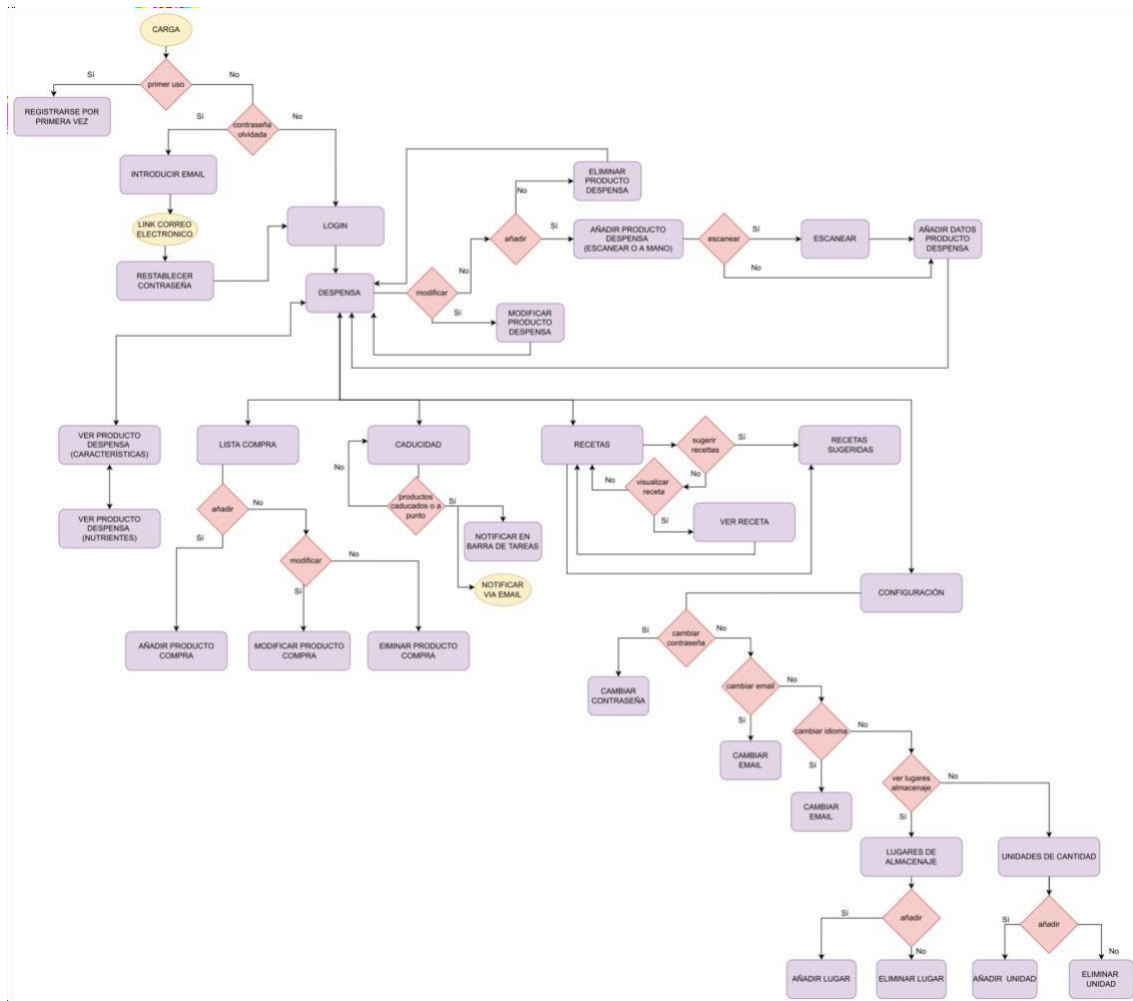


Figura 32: Diagrama de Flujo

## 6.1.3 Listado casos de uso

<b>Identificador</b>	<b>CU-001</b>
<b>Nombre</b>	Login Aplicación
<b>Descripción</b>	Introducir credenciales para entrar y poder utilizar la aplicación.
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	Ninguna
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>El consumidor introduce su nombre y contraseña y pulsa el botón “Log in” en la App, se envían los datos a la base de datos MySQL..</li> <li>La base de datos externa MySQL recibe el nombre y contraseña y responde si son correctos .</li> <li>El consumidor tiene acceso a la App</li> </ul>

<b>Postcondiciones</b>	El consumidor tiene acceso a la App.
------------------------	--------------------------------------

Tabla 16: Caso de uso CU-001

<b>Identificador</b>	<b>CU-002</b>
<b>Nombre</b>	Registrar Consumidor
<b>Descripción</b>	Registrar consumidor en el sistema
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	Ninguna
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor introduce su nombre y apellidos y pulsa el botón “Registrar” en la App, se envían los datos a la base de datos MySQL..</li> <li>• La base de datos externa MySQL recibe el nombre y apellidos . Crea un nuevo registro con el nombre y apellidos del consumidor y un identificador asignado y lo devuelve a la App.</li> <li>• La App recibe el identificador y lo almacena en un fichero de configuración interno.</li> <li>• La App informa del resultado del envío.</li> </ul>
<b>Postcondiciones</b>	Se crea un identificador de usuario en el fichero de configuración de la App.

Tabla 17: Caso de uso CU-002

<b>Identificador</b>	<b>CU-003</b>
<b>Nombre</b>	Contraseña olvidada
<b>Descripción</b>	El consumidor ha olvidado su contraseña y se restablece
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	Ninguna
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor introduce su correo y pulsa el botón para enviar un enlace a su cuenta de correo</li> <li>• La base de datos externa MySQL recibe el correo y comprueba que se encuentra en la base de datos. Acto seguido envía un enlace a la cuenta de correo del usuario para restablecer su contraseña .</li> <li>• El consumidor pincha en el enlace he introduce dos veces su nueva contraseña, luego pulsa para cambiarla.</li> <li>• La contraseña se ha restablecido.</li> </ul>
<b>Postcondiciones</b>	Se restablece una nueva contraseña para el usuario.

Tabla 18: Caso de uso CU-003

<b>Identificador</b>	<b>CU-004</b>
<b>Nombre</b>	Buscar Producto Despensa
<b>Descripción</b>	Buscar un producto concreto almacenado en la despensa.
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de despensa.

<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor introduce el nombre del producto y pulsa la lupa.</li> <li>• En la lista de la pantalla de despensa se muestra solo el producto introducido su cantidad y su precio.</li> <li>• Se actualiza el precio Total con el precio del producto.</li> </ul>
<b>Postcondiciones</b>	Se muestra el nombre del producto buscado con su cantidad, unidad y precio

Tabla 19: Caso de uso CU-004

<b>Identificador</b>	<b>CU-005</b>
<b>Nombre</b>	Añadir Producto Despensa
<b>Descripción</b>	Añadir un nuevo producto a la lista de despensa.
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de despensa.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor pulsa el icono añadir</li> <li>• El sistema pregunta si quiere ir directamente a introducir los datos o quiere escanear el código de barras.</li> <li>• Escanear código de barras</li> <li>• El sistema abre la pantalla de añadir producto despensa.</li> <li>• El consumidor inserta los datos del producto despensa. <ul style="list-style-type: none"> <li>○ Cambiar Imagen</li> <li>○ Leer texto en imagen (OCR)</li> <li>○ Escanear código de barras</li> <li>○ Insertar resto de campos</li> </ul> </li> <li>• Se envían los datos a base de datos MySQL</li> <li>• La base de datos MySQL introduce los datos</li> <li>• El sistema avisa que se han introducido los datos correctamente y vuelve a la pantalla principal de despensa.</li> </ul>
<b>Postcondiciones</b>	Se ha añadido un nuevo producto a la lista de despensa.

Tabla 20: Caso de uso CU-005

<b>Identificador</b>	<b>CU-006</b>
<b>Nombre</b>	Modificar Producto Despensa
<b>Descripción</b>	Modificar un producto a la lista de despensa existente.
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de despensa.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor pulsa el icono editar</li> <li>• El sistema abre la pantalla de modificar producto despensa.</li> </ul>

	<ul style="list-style-type: none"> <li>• El consumidor modifica los datos del producto despensa <ul style="list-style-type: none"> <li>○ Cambiar Imagen</li> <li>○ Leer texto en imagen (OCR)</li> <li>○ Escanear código de barras</li> <li>○ Insertar resto de campos</li> </ul> </li> <li>• Se envían los datos a base de datos MySQL</li> <li>• La base de datos MySQL modifica los datos</li> <li>• El sistema avisa que se han modificado los datos correctamente y vuelve a la pantalla principal de despensa.</li> </ul>
<b>Postcondiciones</b>	Se ha modificado el producto en la lista de despensa.

Tabla 21: Caso de uso CU-006

<b>Identificador</b>	<b>CU-007</b>
<b>Nombre</b>	Eliminar Producto Despensa
<b>Descripción</b>	Eliminar un producto a la lista de despensa existente.
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de despensa.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor pulsa el icono eliminar-</li> <li>• El sistema pregunta al consumidor si realmente desea eliminar el producto.</li> <li>• El sistema envía una petición de borrar con el id de producto a la base de datos externa MySQL.</li> <li>• La base de datos MySQL recibe la petición y borra el producto de la lista de despensa.</li> <li>• El sistema elimina visualmente de la lista el producto.</li> <li>• El sistema avisa que se ha realizado correctamente la acción.</li> </ul>
<b>Postcondiciones</b>	Se ha borrado el elemento de la lista de despensa.

Tabla 22: Caso de uso CU-007

<b>Identificador</b>	<b>CU-008</b>
<b>Nombre</b>	Ver Producto Despensa
<b>Descripción</b>	Ver un producto a la lista de despensa existente.
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de despensa.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor pulsa un producto de la lista de despensa.</li> <li>• Se envía la petición de búsqueda de características con el id de producto a la base de datos externa MySQL.</li> <li>• Se muestra la pantalla Ver Producto con los datos de las características del producto.</li> <li>• El usuario pulsa el botón Nutrientes.</li> </ul>

	<ul style="list-style-type: none"> <li>• Se envía petición de búsqueda de nutrientes del producto a la base de datos externa MySQL.</li> <li>• Se muestra la pantalla Ver producto con los datos de los nutrientes generales del producto.</li> <li>• Seleccionar un grupo de nutrientes.</li> <li>• Se envía petición de búsqueda de nutrientes del producto a la base de datos externa MySQL.</li> <li>• Se muestra la pantalla Ver producto con los datos de los nutrientes generales del producto.</li> </ul>
<b>Postcondiciones</b>	Ninguna

Tabla 23: Caso de uso CU-008

<b>Identificador</b>	<b>CU-009</b>
<b>Nombre</b>	Buscar Producto Lista Compra
<b>Descripción</b>	Buscar un producto concreto almacenado en la lista de compra.
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de lista de compra.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor introduce el nombre del producto y pulsa la lupa..</li> <li>• En la lista de la pantalla de lista de compra se muestra solo el producto introducido con su cantidad.</li> </ul>
<b>Postcondiciones</b>	Se muestra el nombre del producto buscado con su cantidad y unidad.

Tabla 24: Caso de uso CU-009

<b>Identificador</b>	<b>CU-010</b>
<b>Nombre</b>	Añadir Producto Lista Compra
<b>Descripción</b>	Añadir un nuevo producto a la lista de compra
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de lista de compra.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor pulsa el icono añadir</li> <li>• El sistema abre la pantalla de añadir productos de lista de compra</li> <li>• El consumidor inserta el nombre, la cantidad y la unidad de cantidad del producto y pulsa añadir.</li> <li>• Se envían los datos a base de datos MySQL</li> <li>• La base de datos MySQL introduce los datos</li> <li>• El sistema avisa que se han introducido los datos correctamente y vuelve a la pantalla principal de lista de compra.</li> </ul>
<b>Postcondiciones</b>	Se ha añadido un nuevo producto a la lista de compra.

Tabla 25: Caso de uso CU-010

<b>Identificador</b>	<b>CU-011</b>
----------------------	---------------

<b>Nombre</b>	Modificar Producto Lista Compra
<b>Descripción</b>	Modificar un producto a la lista de lista de compra.
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de lista de compra.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor pulsa el icono editar</li> <li>• El sistema abre la pantalla de modificar producto lista compra.</li> <li>• El consumidor modifica los datos del producto despensa</li> <li>• Se envían los datos a base de datos MySQL</li> <li>• La base de datos MySQL modifica los datos</li> <li>• El sistema avisa que se han modificado los datos correctamente y vuelve a la pantalla principal de lista de compra.</li> </ul>
<b>Postcondiciones</b>	Se ha modificado el producto en la lista de compra.

Tabla 26: Caso de uso CU-011

<b>Identificador</b>	<b>CU-012</b>
<b>Nombre</b>	Eliminar Producto Lista Compra
<b>Descripción</b>	Eliminar un producto a la lista de lista de compra.
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de lista de compra.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor pulsa el icono eliminar-</li> <li>• El sistema pregunta al consumidor si realmente desea eliminar el producto.</li> <li>• El sistema envía una petición de borrar con el id de producto a la base de datos externa MySQL.</li> <li>• La base de datos MySQL recibe la petición y borra el producto de la lista de compra.</li> <li>• El sistema elimina visualmente de la lista el producto.</li> <li>• El sistema avisa que se ha realizado correctamente la acción.</li> </ul>
<b>Postcondiciones</b>	Se ha borrado el elemento de la lista de compra.

Tabla 27: Caso de uso CU-012

<b>Identificador</b>	<b>CU-013</b>
<b>Nombre</b>	Buscar Producto Lista Caducidad
<b>Descripción</b>	Buscar un producto concreto almacenado en la lista de caducidad.
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de caducidad.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor introduce el nombre del producto y pulsa la lupa.</li> </ul>

	<ul style="list-style-type: none"> <li>En la lista de la pantalla de lista de caducidad se muestra solo el producto introducido con los días que quedan para que caduque y su precio.</li> </ul>
<b>Postcondiciones</b>	Se muestra el nombre del producto buscado con su cantidad y unidad.

Tabla 28: Caso de uso CU-013

<b>Identificador</b>	<b>CU-014</b>
<b>Nombre</b>	Seleccionar Tipo Productos Lista Caducidad
<b>Descripción</b>	Seleccionar productos en la lista de caducidad según la temporalidad que quede para que caduque: Todos, ya caducados, 0 a 10 días o más de 10 días.
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de caducidad.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>El consumidor pulsa una opción de temporalidad de caducidad.</li> <li>El sistema envía una petición de productos con esa temporalidad a la base de datos externa MySQL.</li> <li>El sistema muestra visualmente solo los productos con esa temporalidad.</li> </ul>
<b>Postcondiciones</b>	Se muestran los productos caducados con la temporalidad seleccionada.

Tabla 29: Caso de uso CU-014

<b>Identificador</b>	<b>CU-015</b>
<b>Nombre</b>	Eliminar Productos Caducados
<b>Descripción</b>	Eliminar productos caducados de la lista de caducidad y de la lista de despensa.
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de caducidad. Asimismo, la lista de caducidad debe tener productos caducados.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>El consumidor pulsa el botón eliminar caducados.</li> <li>El sistema envía a la base de datos externa MySQL una petición de borrado con productos cuya fecha de caducidad haya pasado</li> <li>El sistema muestra visualmente los cambios en la App.</li> </ul>
<b>Postcondiciones</b>	Se eliminan los productos que están caducados de la lista de despensa.

Tabla 30: Caso de uso CU-015

<b>Identificador</b>	<b>CU-016</b>
<b>Nombre</b>	Buscar Receta
<b>Descripción</b>	Buscar una receta en el sistema
<b>Actores</b>	Consumidor, base de datos externa MySQL



<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de recetas.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor introduce el nombre del título de la receta y pulsa la lupa.</li> <li>• En la lista de la pantalla de recetas se muestra solo el nombre de la receta..</li> </ul>
<b>Postcondiciones</b>	Se muestra el nombre de la receta buscada en la lista de recetas.

Tabla 31: Caso de uso CU-016

<b>Identificador</b>	<b>CU-017</b>
<b>Nombre</b>	Sugerir Recetas
<b>Descripción</b>	Sugerir recetas que contengan uno o más alimentos de la despensa
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de recetas.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor pulsa el botón sugerencias.</li> <li>• El sistema envía una petición a la base de datos externa MySQL para que retorne recetas con productos que se encuentren en la lista de despensa.</li> <li>• Se muestran visualmente los títulos de las recetas que coinciden con los resultados de la búsqueda anterior.</li> </ul>
<b>Postcondiciones</b>	Se muestra los nombres de las recetas sugeridas con uno o más productos en despensa.

Tabla 32: Caso de uso CU-017

<b>Identificador</b>	<b>CU-018</b>
<b>Nombre</b>	Ver Receta
<b>Descripción</b>	Ver una receta concreta de la lista de recetas mostrada
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de recetas.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor selecciona una receta de la lista.</li> <li>• El sistema envía una petición a la base de datos externa MySQL para que retorne los datos de la receta solicitada.</li> <li>• El sistema abre la pantalla de la información de Receta y muestra la información de esta.</li> </ul>
<b>Postcondiciones</b>	Se muestra los nombres de las recetas sugeridas con uno o más productos en despensa.

Tabla 33: Caso de uso CU-018

<b>Identificador</b>	<b>CU-019</b>
<b>Nombre</b>	Buscar Lugar Almacenaje

<b>Descripción</b>	Buscar un lugar de almacenaje del consumidor
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de configuración.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor introduce el nombre del lugar de almacenaje y pulsa la lupa-</li> <li>• En la lista de la pantalla de lugares de almacenaje se muestra solo el nombre del lugar.</li> </ul>
<b>Postcondiciones</b>	Se muestra el nombre del lugar buscado en la lista de lugares de almacenaje-

Tabla 34: Caso de uso CU-019

<b>Identificador</b>	<b>CU-O20</b>
<b>Nombre</b>	Añadir lugar de almacenaje
<b>Descripción</b>	Añadir un lugar de almacenaje para el consumidor
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de configuración.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor pulsa el botón lugares de almacenaje.</li> <li>• El consumidor introduce el nombre del lugar de almacenaje</li> <li>• El sistema envía una petición de inserción con el nombre a la base de datos externa MySQL..</li> <li>• La App vuelve a la pantalla de la lista de lugares de almacenaje.</li> </ul>
<b>Postcondiciones</b>	Se añade un nuevo lugar de almacenaje para el consumidor.

Tabla 35: Caso de uso CU-020

<b>Identificador</b>	<b>CU-O21</b>
<b>Nombre</b>	Eliminar Lugar Almacenaje
<b>Descripción</b>	Eliminar un lugar de almacenaje del consumidor
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de configuración.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor pulsa el icono de eliminar.</li> <li>• Se envía una petición para eliminar el lugar de almacenaje a la base de datos externas MySQL.</li> <li>• Visualmente se elimina el lugar de almacenaje de la lista.</li> </ul>
<b>Postcondiciones</b>	Se elimina el lugar de almacenaje de la lista.

Tabla 36: Caso de uso CU-021

<b>Identificador</b>	<b>CU-O22</b>
<b>Nombre</b>	Buscar Unidad Cantidad
<b>Descripción</b>	Buscar una unidad de cantidad del consumidor

<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de configuración.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor introduce el nombre de la unidad de cantidad y pulsa la lupa-</li> <li>• En la lista de la pantalla de unidades de cantidad se muestra solo el nombre de la unidad.</li> </ul>
<b>Postcondiciones</b>	Se muestra el nombre de la unidad de cantidad buscada en la lista de lugares de almacenaje-

Tabla 37: Caso de uso CU-022

<b>Identificador</b>	<b>CU-O23</b>
<b>Nombre</b>	Añadir unidad de cantidad
<b>Descripción</b>	Añadir una unidad de cantidad para el consumidor
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de configuración.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor pulsa el botón unidades de cantidad.</li> <li>• El consumidor introduce el nombre de la unidad de cantidad</li> <li>• El sistema envía una petición de inserción con el nombre a la base de datos externa MySQL..</li> <li>• La App vuelve a la pantalla de la lista unidades de cantidad.</li> </ul>
<b>Postcondiciones</b>	Se añade una nueva unidad de cantidad para el consumidor.

Tabla 38: Caso de uso CU-023

<b>Identificador</b>	<b>CU-O24</b>
<b>Nombre</b>	Eliminar Unidad de Cantidad
<b>Descripción</b>	Eliminar una unidad de cantidad del consumidor
<b>Actores</b>	Consumidor, base de datos externa MySQL
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de configuración.
<b>Iniciado por</b>	Consumidor
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El consumidor pulsa el icono de eliminar.</li> <li>• Se envía una petición para eliminar la unidad de cantidad a la base de datos externas MySQL.</li> <li>• Visualmente se elimina la unidad de cantidad de la lista.</li> </ul>
<b>Postcondiciones</b>	Se elimina la unidad de cantidad de la lista.

Tabla 39: Caso de uso CU-024

<b>Identificador</b>	<b>CU-O25</b>
<b>Nombre</b>	Notificar producto caducado
<b>Descripción</b>	Notifica por email y en la barra de tareas cuando un producto caduca o esta a punto de caducar.

<b>Actores</b>	Consumidor, base de datos externa MySQL, servicio PHP
<b>Precondiciones</b>	El consumidor debe estar registrado en el sistema y debe estar ubicado en la pantalla de recetas.
<b>Iniciado por</b>	Servicio PHP
<b>Flujo</b>	<ul style="list-style-type: none"> <li>• El servicio PHP detecta cada día a las 00:00 si existen productos caducados en la base de datos MySQL</li> <li>• Envía un correo PHP con los productos caducados y a punto de caducar.</li> <li>• Envía un json a la App para que muestre una notificación en la barra de tareas de los productos caducados y a punto de hacerlo.</li> </ul>
<b>Postcondiciones</b>	Se recibe correo y notificación con los productos caducados y a punto de hacerlo.

Tabla 40: Caso de uso CU-025

## 6.2 Diseño de la arquitectura

### 6.2.1 Diagrama UML Base de datos

El sistema constará de dos tipos de base de datos. Una interna en el propio dispositivo de tipo SQLite y otra de tipo externo MySQL a la que se accederá mediante peticiones GET y POST. Los diagramas UML siguientes reflejan el modelo de datos y sus diagramas Entidad relación.

#### Base de Datos SQLite

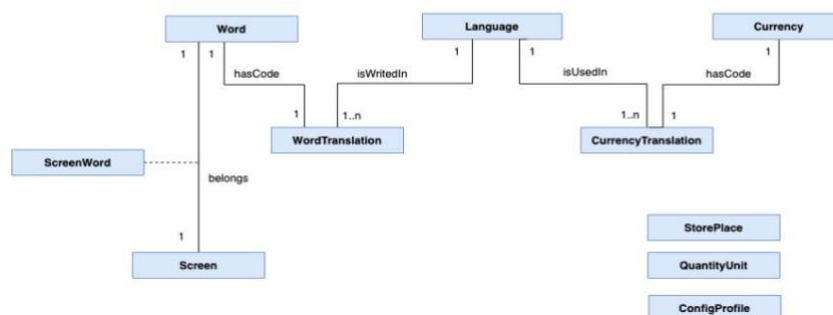


Figura 33: Modelo Entidad Relación base de datos SQLite

Definición de tablas:

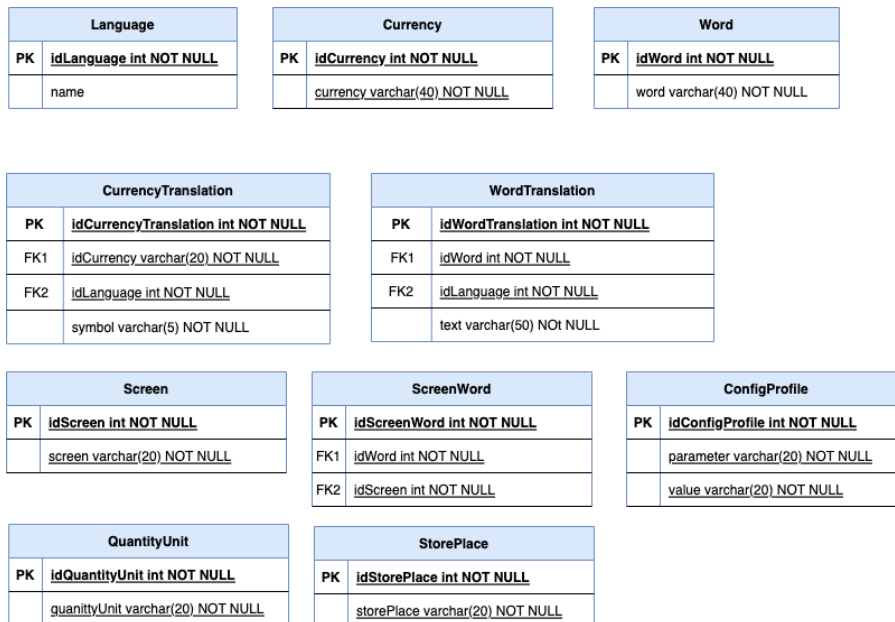


Figura 34: Definición tablas base de datos SQLite

### Base de Datos MySQL

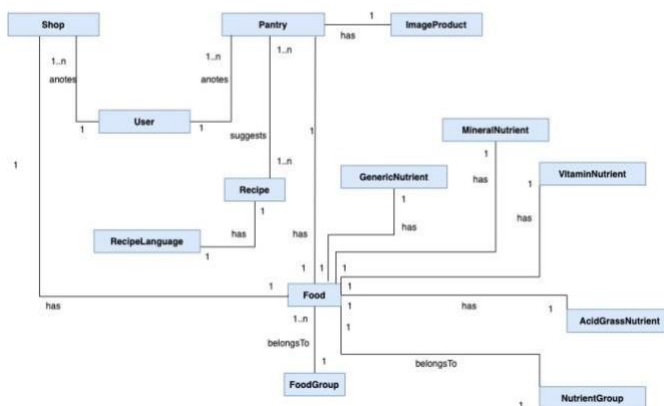


Figura 35: Modelo Entidad Relación base datos MySQL

Definición de tablas:



Figura 36: Definición tablas base de datos MySQL

## 6.2.2 Diagrama UML Entidades y clases

La App utiliza una arquitectura MVP. Se ha optado por esta opción por ser la menos compleja y porque es más fácil de localizar el código. El modelo de clases es el que sigue:



Figura 37: Diagrama de clases MVP 1

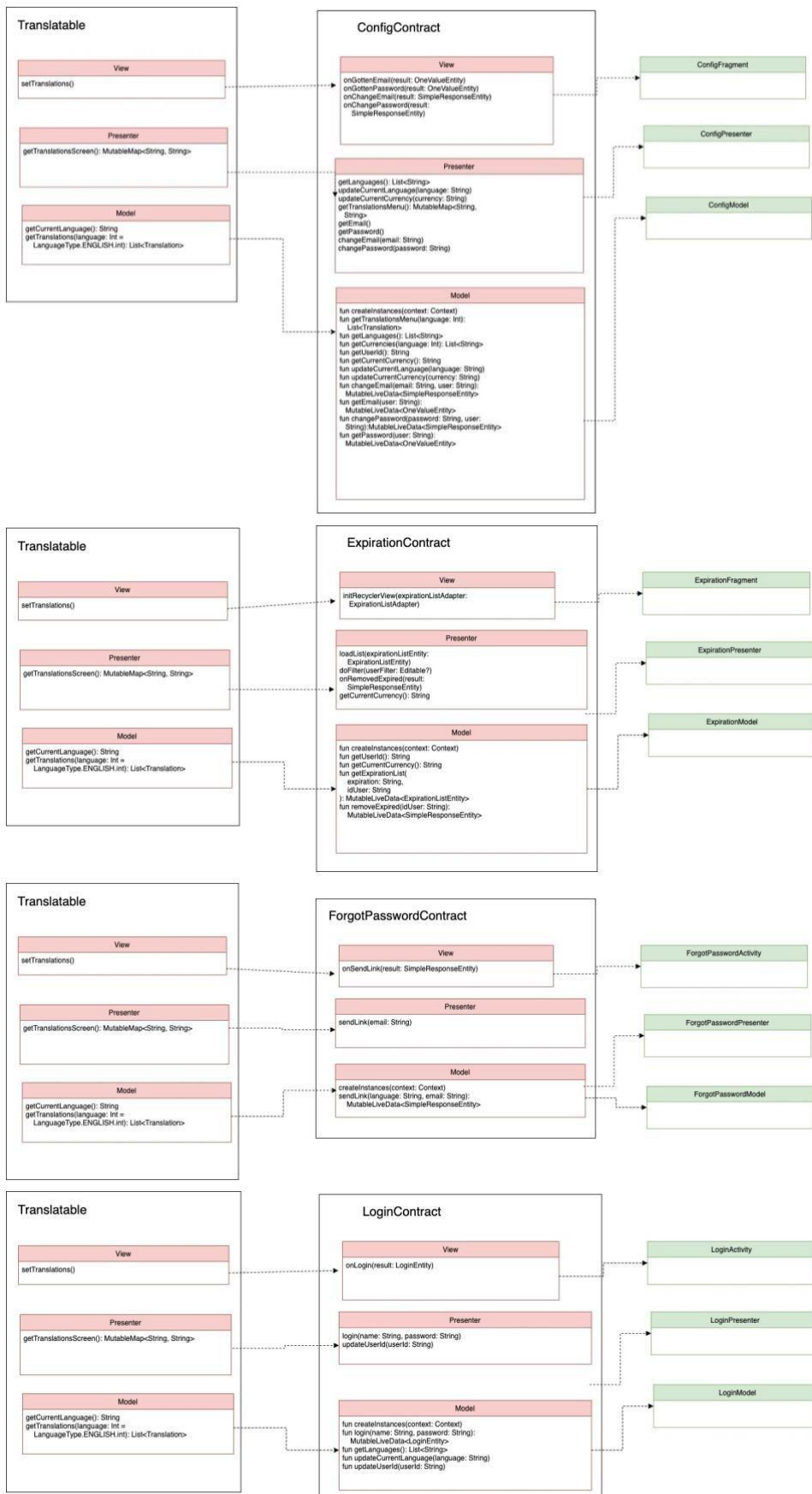


Figura 38: Diagrama de clases MVP 2



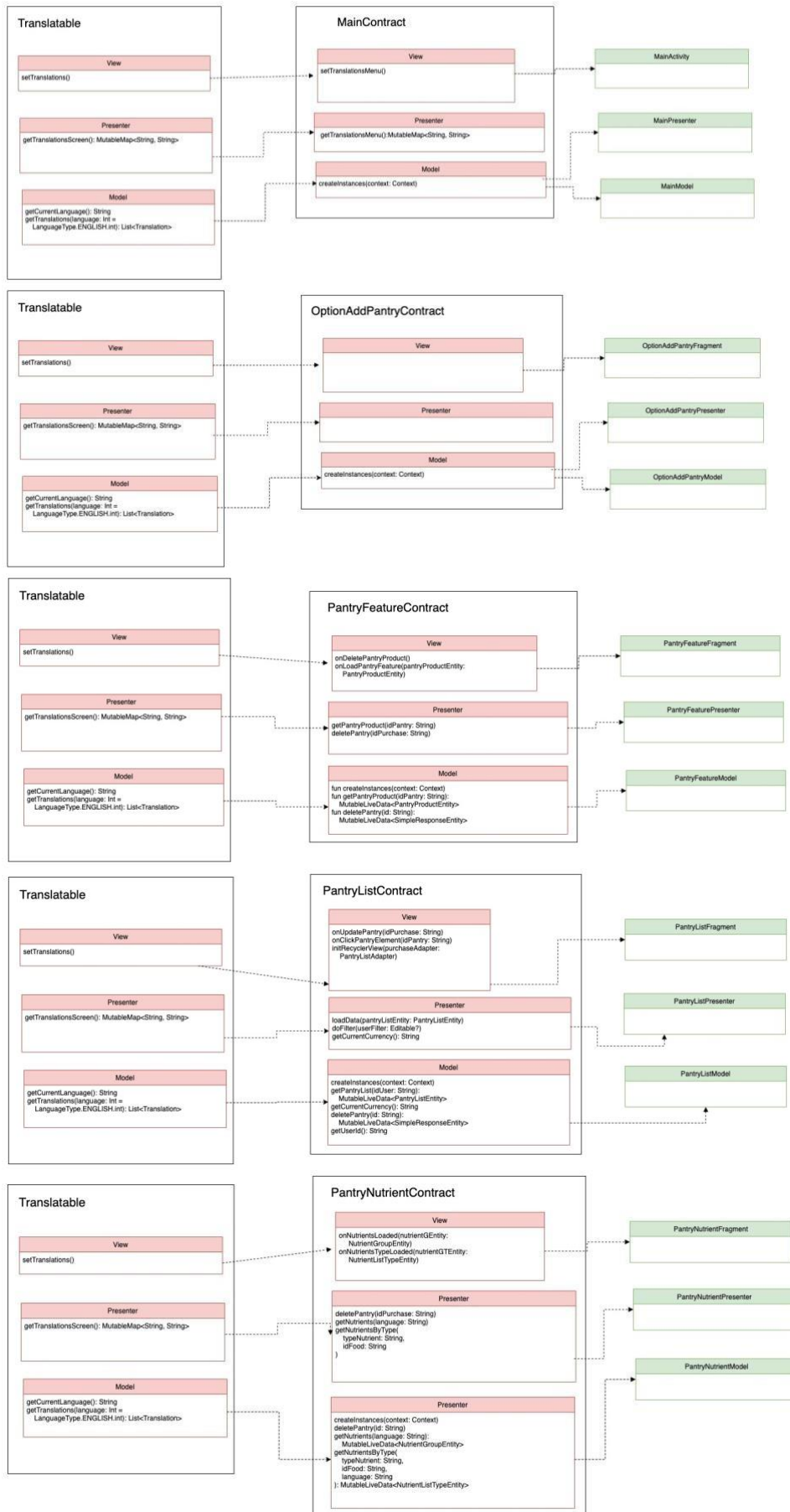


Figura 39: Diagrama de clases MVP 3

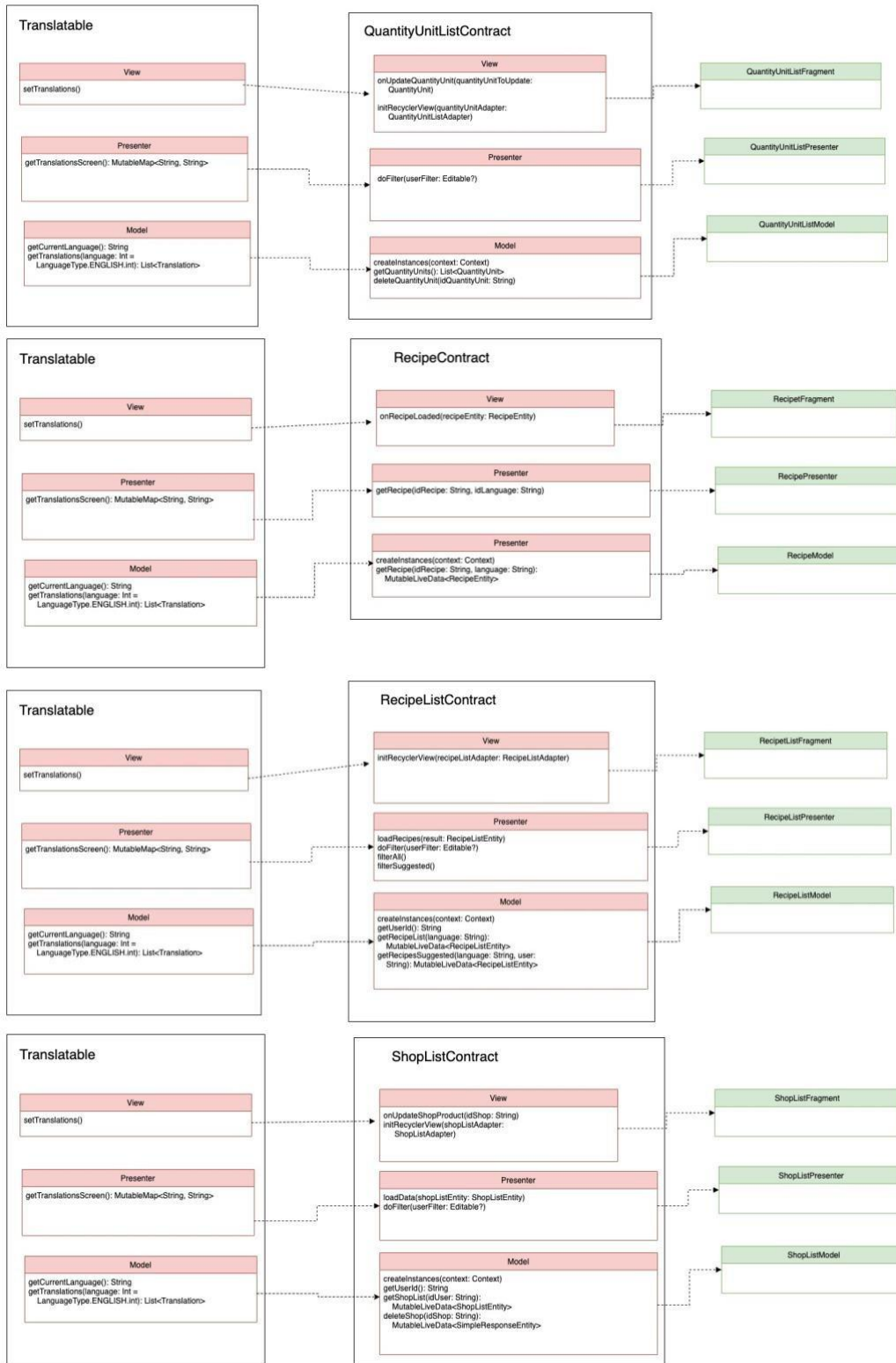


Figura 40: Diagrama de clases MVP 4

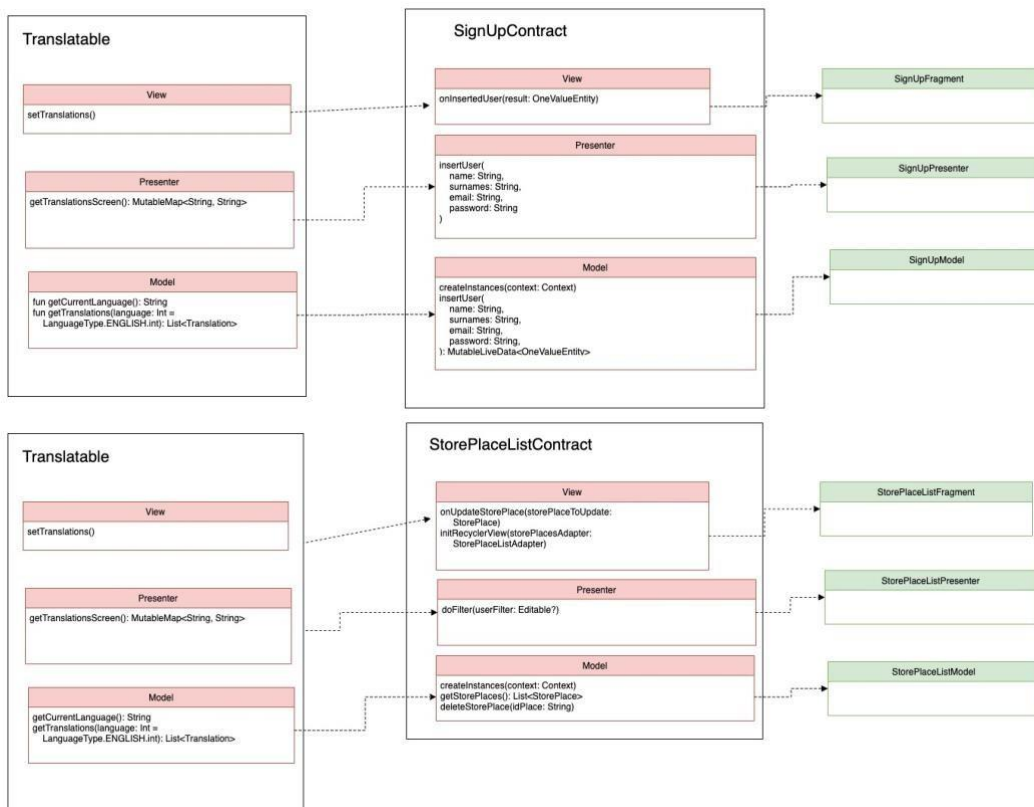


Figura 41: Diagrama de clases MVP 5

### 6.2.3 Diagrama explicativo de la arquitectura del sistema.

A continuación, se muestra un diagrama explicativo de la arquitectura del sistema. El usuario utiliza la App en su dispositivo Android que contiene en su interior una base de datos SQLite para los datos más básicos y una librería para escanear denominada Zxing. Para los datos más gordos obtiene los datos de la base de datos MySQL a través de pequeños scripts en PHP a los que se puede acceder mediante peticiones GET y POST con protocolo HTTP REST.

Asimismo, también se obtiene la información nutricional del producto mediante una API a la que se accede mediante una petición GET en protocolo REST a openfoodfacts y de igual modo para obtener el texto mediante lectura OCR de una imagen mediante la API de OCR Nanonets. Finalmente, disponemos también de un servicio en Cronjob.org para programar diariamente un script PHP que realiza una petición HTTP con protocolo REST a la base de datos MySQL de Infinityfree para lanzar una tarea que envíe un correo a los usuarios con los productos caducados.

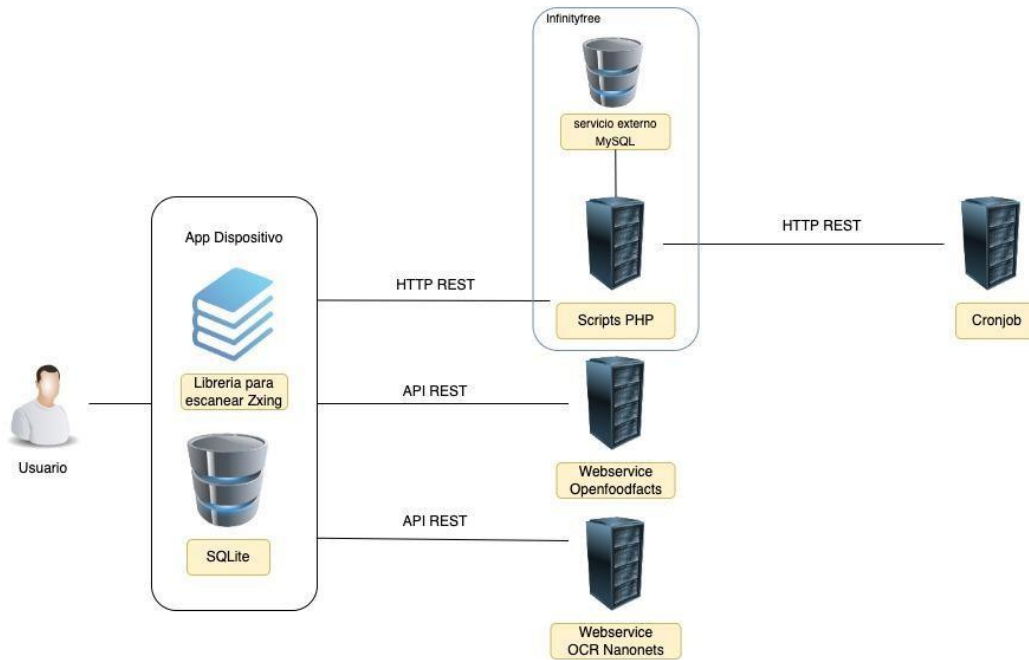


Figura 42: Diagrama arquitectura del sistema

#### 6.2.4 Librerías externas

- Retrofit: Librería que permite realizar peticiones API REST a los scripts php del hosting ifastnet.
- Room: Con ella se obtendrán los datos de la base de datos local del dispositivo SQLite.
- Picasso: Librería que nos permite cargar una imagen url desde internet en un elemento de vista ImageView.
- Zxing: Gracias a ella se podrá escanear códigos de barra mediante la cámara del dispositivo móvil.

## 7. Implementación

### 7.1 Entorno de desarrollo empleado

Para el desarrollo se ha utilizado el IDE de Android Studio Dolphin con la última versión 2021.3.1 y su emulador. El motivo es que es más ligero que el de versiones anteriores y ofrece más funcionalidad. Alternativamente se ha empleado también el emulador scrpcy que permite ejecutar y depurar el código en un dispositivo físico sin necesidad de cable. Asimismo, he utilizado mi teléfono móvil Huawei P20 que aún funciona con versión Android.

### 7.2 Herramientas

Se ha utilizado la herramienta SQLite Studio para poblar la base de datos SQLite, verificar los datos y hacer pruebas de consulta. Paralelamente para probar las peticiones GET y POST realizadas se ha empleado la aplicación Postman por ser de ámbito gratuito y su fácil manejo.

### 7.3 Servicio PHP y MySQL

Se ha optado por un servicio con Ifastnet por ser más consistente y robusto. El hosting también dispone de un programador de tareas (Cronjob), pero no se ha conseguido establecer la tarea mediante ese servicio y es por ello que se ha utilizado la ayuda de otro servicio gratuito en la nube llamado cron-job.org.

### 7.4 API Openfood

También se ha utilizado la API de Openfood que proporciona las características de un producto alimenticio a partir de una petición GET con el código de barras. Esta API está muy bien, ya que nos evita relacionar los posibles códigos de barras con su producto a mano.

### 7.5 Análisis del estado del proyecto

En un principio se ha tenido problemas después de crear una versión de SQLite para el correcto funcionamiento de la App. La versión de base de datos nueva no era reconocida y la aplicación se cerraba. Para ello cada vez que se añadía un nuevo dato se ha incrementado la versión y se ha realizado una carga previa con el método `fallbackToDestructiveMigration`.

Por otro lado, se empezó el proyecto sin ningún tipo de arquitectura, después al ser demasiado confuso se utilizó una arquitectura MVVM con la librería Dagger y corrutinas. Sin embargo, se presentaron problemas en la construcción y se ha optado finalmente por una arquitectura de clases MVP por ser menos compleja y más legible.

Asimismo, se empezó a realizar las peticiones a los scripts PHP mediante la librería externa okhttp, pero más tarde se descubrió la librería Retrofit que resulta más legible y mucho más organizada y además permite serializar los objetos json retornados. También, para más consistencia en el retorno de datos se ha empleado los objetos MutableLiveData que nos permiten pasar datos de un hilo a otro de ejecución.

Durante el periodo de desarrollo se quería poder cambiar la imagen del producto de despensa con una de la propia galería del dispositivo. Dado que la uri retornada no se abría por problema de permisos y no se consiguió realizar de esta forma, se optó por otro camino. Se cambió el campo en que se almacena la imagen de varchar (url) a longblob para guardar la imagen en formato base64. Dicho cambio provocaba que en la petición GET la url fuera demasiado larga y no se aceptase por lo que se tuvo que enviar con una petición POST.

Respecto al servicio PHP y MySQL, al principio empecé con el hosting gratuito de Infinity.net para realizar los scripts que permitiesen proporcionar la API al dispositivo móvil. No obstante, poco antes de finalizar el desarrollo se produjo un apagón en el sistema y no funcionó el servicio durante 2 días. Además, cuando se solventó, el servicio ya no permitía realizar peticiones mediante un dispositivo móvil devolviendo error. Es por ello que se ha contratado los mismos servicios con ifastnet durante un periodo determinado.

Por último, por falta de tiempo no se ha podido finalizar las funcionalidades del botón OCR en la pantalla de añadir producto de despensa, ni las notificaciones de los productos expirados en el móvil. De este modo, se ha hecho no visible el botón OCR en la vista para que no sea visible al usuario y poder desarrollarlo en futuras versiones.

# 8. Pruebas

## 8.1 Prueba de aplicación

Para la realización de las pruebas se utilizará el framework de JUnit5 para las pruebas unitarias y por otro lado para las funcionales de integración el emulador propio de Android Studio Dolphin y el de scrpy para dispositivo físico.

Las pruebas unitarias se ejecutarán directamente desde Android Studio Dolphin, mientras que en las funcionales se probará manualmente mediante la interfaz de usuario de la App.

## 8.2 Pruebas Unitarias

Las pruebas unitarias se han llevado a cabo durante el desarrollo de todo el proyecto para testear que las peticiones en la API se devuelven y se ejecutan correctamente.

## 8.3 Pruebas Funcionales de Integración

Conforme se ha ido finalizando el desarrollo de las distintas secciones de la aplicación se han llevado a cabo pruebas funcionales con el fin de probar cada una de las acciones que pueden realizarse en la app. Se han probado las siguientes acciones:

- Registro de usuario
- Envío de link para resetear la contraseña en caso de olvido.
- Log in de usuario
- Carga de listado de productos despensa
- Escaneo de código de barras y carga producto de la API Openfood
- Cambiar imagen de producto despensa con una de la galería de imágenes del dispositivo.
- Insertar producto despensa
- Modificar producto despensa (desde listado y pantalla de características y nutrientes del producto)
- Eliminar producto despensa (desde listado y pantalla de características y nutrientes del producto)
- Ver características producto despensa
- Ver tipo nutrientes producto despensa
- Carga de lista de productos compra
- Insertar producto compra
- Modificar producto compra
- Eliminar producto compra
- Consultar productos caducados, entre 0 y 10 días de caducidad, más de 10 días a caducar y todos.

- Eliminar productos caducados
- Consultar lista de recetas en el idioma seleccionado
- Consultar lista de recetas sugeridas en el idioma seleccionado
- Consultar una receta
- Cambiar lenguaje y moneda y que se vea reflejado en la App
- Obtener correo y cambiar correo en la pantalla de Configuración
- Obtener contraseña y cambiar contraseña en la pantalla de Configuración
- Consultar lista de unidades de cantidad
- Insertar unidad de cantidad
- Modificar unidad de cantidad
- Eliminar unidad de cantidad
- Consultar lista de lugares de almacenaje
- Insertar lugar de almacenaje
- Modificar lugar de almacenaje
- Eliminar lugar de almacenaje



## 9. Valoración económica

### 9.1 Estrategia de Monetización

La monetización de la App se fundamentará en dar una funcionalidad completa en la que se limitaran la cantidad de productos de despensa y de compra que se puedan almacenar. Dicha limitación se podrá desbloquear en una versión de pago (de coste aún por concretar).

También se contempla la opción de añadir publicidad dentro de la App para obtener financiación y en la cual si se paga desaparecería.

Ofrecer una funcionalidad completa y sin publicidad nos proporciona la opción de convencer al usuario de comprar la aplicación de pago sin limitaciones, brindando la oportunidad de que conozca la potencia real de la misma.

### 9.2 Plan de Marketing y Distribución

Para la estrategia de marketing y distribución se brindará más capacidad de productos y velocidad a aquellos usuarios que hayan hablado y convencido a otros para que se descarguen la App.

Asimismo, se utilizarán banners y otros tipos de publicidad en páginas web para dar a conocer el producto al usuario. El usuario podrá descargar una demo del producto y probarlo durante un periodo de una semana de forma gratuita. Tras ese periodo deberá comprar la App para seguir utilizándola sin publicidad y con funcionalidad completa.

# 10. Conclusiones y Líneas Futuras

## 10.1 Conclusiones del Trabajo

Durante la elaboración del TFG la idea clave a destacar es que se debe realizar un análisis meticuloso previo al desarrollo y conocer bien las herramientas que se van a utilizar para después no perder tiempo investigando o probando mediante prueba y error sus funcionalidades. Las elecciones en cuanto base de datos, librerías, patrones, etc. deben ser robustas y nos deben asegurar una solución para todas las funcionalidades que se plantean. El código debe ser legible y comprensible para alguien que no haya desarrollado el producto.

Ha habido muchos cambios y retrasos durante la etapa de desarrollo del TFG por desconocimiento de patrones o herramientas que han provocado la no finalización de objetivos secundarios como el botón OCR o las notificaciones de la barra de tareas en el móvil.

A pesar de las dificultades que se han presentado, se está satisfecho con el esfuerzo realizado y el producto obtenido. Se han conocido herramientas nuevas, el lenguaje de programación Kotlin, base de datos, scripts de servidor, patrones de diseño, etc. Ha sido un proyecto muy ecléctico formado por muchos elementos diferentes. Asimismo, se ha aprendido a tomar decisiones y solventar problemas cuando el tiempo se agotaba para descartar o incorporar elementos y tener un producto terminado funcional.

## 10.2 Reflexión de Logros

Referente a los logros del trabajo realizado se ha de decir que no se han conseguido implementar todos por falta de tiempo, aunque la App se ha acabado y es funcional. Un ejemplo de ello son objetivos secundarios como el botón OCR y las notificaciones de la barra de tareas del móvil.

La principal razón ha sido los retrasos por cambios e investigación en objetivos principales o imprevistos como el apagón en el antiguo hosting que se estaba utilizando cuando el desarrollo ya estaba bastante avanzado.

## 10.3 Análisis Seguimiento de Planificación y Metodología

Aunque se ha seguido la planificación y metodología prevista en un inicio, ha habido imprevistos en el desarrollo que han provocado retrasos en la finalización de las primeras fases del producto. Como ya se ha comentado, dichos cambios se podían haber evitado con un mejor conocimiento de las herramientas, base de datos, patrones, etc. La razón de ese cambio ha sido obtener un código más legible para alguien que no haya desarrollado la App, robusto, rápido y que sea funcional.

#### 10.4 Líneas Futuras

De cara a líneas futuras, el sistema debería tener un hosting para las peticiones a la API de MySQL más robusto conforme más usuarios utilicen la App. Además, se puede incorporar los objetivos secundarios como el botón OCR para obtener los datos del producto alimenticio a partir de su etiqueta mediante la captura de una imagen y las notificaciones en la barra de tareas del teléfono móvil cuando un producto este caducado o a punto de hacerlo.

Con respecto a la API de Openfood que utilizamos para obtener el producto a partir del código de barras sería conveniente tener nuestra propia base de datos. De esta forma si se retira el servicio de Openfood seguiríamos teniendo la funcionalidad.

También en un futuro se podría pasar la aplicación a otros dispositivos móviles como Apple o Windows Mobile publicando las aplicaciones en Google Play Store, Apple Store, etc.

Una vez pulido y realizado el producto, sería conveniente también realizar un mantenimiento cada determinado tiempo para actualizar posibles productos y funcionalidades de la App.

# 11. Glosario

- App: Application. o App Mobile Término empleado para referirse a las aplicaciones para smarthphones.
- BBC: British Broadcasting Corporation
- UX: User eXperience: Experiencia de Usuario
- Excel: Hoja de cálculo desarrollado por Microsoft.
- IDE (Entorno de Desarrollo Integrado): Entorno de programación.
- TFG: Trabajo Final de Grado
- ONF: Objetivo No Funcional
- OF: Objetivo Funcional.
- Android: Sistema Operativo para dispositivos móviles.
- PEC: Prueba de Evaluación Continua.
- VCS: Version Control System. Sistema de Control de Versiones.
- MySQL: Sistema de gestión de base de datos relacional considerada como la base de datos de código abierto más popular del mundo.
- PHP: Lenguaje de programación de uso general que se adapta esencialmente al desarrollo web.
- PDF: Portable Document Format. Tipo de formato de almacenamiento de archivos digitales portáti.
- GET: Método de petición HTTP que solicita una representación de un recurso específico en el que solo se deben recuperar datos.
- POST: Método de petición HTTP que se utiliza para enviar una entidad o recurso específico, causando a menudo un cambio en el estado o efectos secundarios en el servidor.
- UML: Unified Modeling Language. En castellano lenguaje unificado de modelado, es el lenguaje de modelado gráfico más conocido para visualizar, especificar, construir y documentar un sistema.

- OCR: Optical Character Recognition. Reconocimiento Óptico de Caracteres. Proceso dirigido a la digitalización de textos en la cual se identifican símbolos y caracteres del alfabeto a partir de una imagen.
- MVP: Model-View-Presenter. Patrón Modelo-Vista-Presentador derivado de la arquitectura MVC (Modelo Vista Controlador) utilizado mayoritariamente para construir interfaces de usuario. La lógica de negocio es colocada en el Presentador.
- MVVM: Model-View-ViewModel. Patrón Modelo-Vista-VistaModelo que permite una separación limpia entre la lógica de la aplicación, la interfaz de usuario y el acceso a base de datos. La lógica de negocio es colocada en la parte de VistaModelo.
- Hosting: Servidor web donde se alojan scripts o paginas web y base de datos para que cualquier usuario o App pueda acceder a ella a través de un dominio de Internet.
- Cronjob: Es un servicio que permite ejecutar de manera automatizada tareas o scripts en sistemas operativos o servicios web.
- Framework: Entorno de trabajo. Conjunto estandarizado de conceptos y código utilizado para una problemática particular.
- Dagger: Es un inyector de dependencias para Android
- Corrutinas: Es un patrón de diseño de simultaneidad que se puede usar en Android para simplificar el código que se ejecuta de forma asíncrona.

## 12. Bibliografía

- [1] Fernanda Paúl. Las impactantes cifras que deja el desperdicio de comida en el mundo (y cuáles son sus efectos). BBC News Mundo. 15 marzo 2021.  
Web: <https://www.bbc.com/mundo/noticias-56322961>  
Visitada: 7 octubre 2022
- [2] Web: <https://gs.statcounter.com/os-market-share/mobile/worldwide>  
Visitada: 7 octubre 2022
- [3] Web <https://developer.android.google.cn/kotlin/first?hl=es-419>  
Visitada: 9 octubre 2022
- [4] Web <https://www.freeonlinesurveys.com>  
Visitada: 20 octubre 2022
- Capítulo 17 – Persistencia de datos con Room. Curso Kotlin Para ANDROID. 9 noviembre 2017  
Web <https://cursokotlin.com/capitulo-17-persistencia-de-datos-con-room/>  
Visitada: 1 diciembre 2022
- Tutorial Retrofit 2 en Kotlin con Corrutinas – Consumiendo API [Capitulo 20 v2]. Curso Kotlin Para ANDROID. 18 febrero 2021  
Web <https://cursokotlin.com/tutorial-retrofit-2-en-kotlin-con-corrutinas-consumiendo-api-capitulo-20-v2/>  
Visitada: 9 diciembre 2022
- Testing Room Database in Android using Junit. Geeksforgeeks. 17 marzo 2021.  
<https://www.geeksforgeeks.org/testing-room-database-in-android-using-junit/>  
Visitada: 10 diciembre 2022
- API. OpenFoodFacts  
<https://world.openfoodfacts.org/data>  
Visitada: 29 noviembre 2022
- Room Database with Kotlin. Medium. Çağnur Hacımahmutoğlu Parçal .25 Junio 2020  
<https://medium.com/huawei-developers/room-database-with-kotlin-mvvm-architecture-477c3ad3c264>

Visitada: 10 diciembre 2022

- How To Use Retrofit 2 With Android Using Kotlin. Mushta1q M A. 10 junio 2020  
<https://www.c-sharpcorner.com/article/how-to-use-retrofit-2-with-android-using-kotlin/>

Visitada: 9 diciembre 2022