

# Desarrollo de aplicación móvil Android: NutriKNow!

**Noelia Martínez Alonso**

Grado de Ingeniería Informática

Desarrollo de aplicaciones para dispositivos móviles (Android)

**David Escuer Latorre**

**Jordi Almirall López**

**Carles Garrigues Olivella**

01/2021



Esta obra está sujeta a una licencia de  
Reconocimiento-NoComercial-SinObraDerivada  
*3.0 España de Creative Commons*

## FICHA DEL TRABAJO FINAL

<b>Título del trabajo:</b>	Desarrollo de aplicación móvil Android: NutriKNow!
<b>Nombre del autor:</b>	Noelia Martínez Alonso
<b>Nombre del consultor/a:</b>	David Escuer Latorre Jordi Almirall López
<b>Nombre del PRA:</b>	Carles Garrigues Olivella
<b>Fecha de entrega:</b>	01/2021
<b>Titulación:</b>	Grado de Ingeniería Informática
<b>Área del Trabajo Final:</b>	Desarrollo de aplicaciones para dispositivos móviles (Android)
<b>Idioma del trabajo:</b>	Castellano
<b>Palabras clave</b>	Android, alérgenos, alimentación
<b>Resumen del Trabajo:</b>	
<p>Este proyecto consiste en la creación de una aplicación móvil nativa en Android para conocer los alérgenos que potencialmente contiene un producto a partir del escaneo de su código de barras.</p> <p>Además, la aplicación aporta al usuario el desglose de ingredientes, su información nutricional, la presencia de aditivos y la categorización del producto según su grado de procesado.</p> <p>El usuario puede guardar búsquedas y productos como favoritos y recibir notificaciones cuando las características de alguno de estos productos favoritos cambien.</p> <p>La aplicación permite la creación de un perfil del usuario en el que indicar cuáles son los alérgenos de interés para poder adaptar la presentación de la información a las necesidades del usuario.</p> <p>El desarrollo del proyecto sigue una metodología clásica en cascada, por lo que comienza con una fase de investigación en la que se definen los objetivos, la metodología y el plan de trabajo que definirán el curso del proyecto. A continuación, se realiza el diseño centrado en el usuario a partir de la definición de los casos de uso, la navegación y la creación de un prototipo.</p> <p>La implementación de la aplicación utiliza como entorno de desarrollo Android Studio utilizando el API pública de Open Food Facts para la obtención de la información sobre productos y Firebase para la administración de usuarios, envío de notificaciones y gestión de base de datos. El proyecto finaliza con el desarrollo y puesta en marcha de un plan de pruebas y un proceso de pruebas con usuarios potenciales.</p>	

**Abstract:**

This project consists of the creation of a native mobile application on Android to know the allergens that a product potentially contains by scanning its barcode.

In addition, the application provides the user with the list of ingredients, their nutritional information, informs about the presence of additives and the categorization of the product according to its degree of processing.

The user can save searches and products as favourite and receive notifications when the characteristics of any of these favourite products change.

The application allows the creation of a user profile where setting up the allergens of interest to adapt the presentation of the information to the user's needs.

The development of the project follows a classic cascade methodology, so it begins with a research phase in which the objectives, methodology and work plan that will define the course of the project are defined. Next, the user-centered design is carried out from the definition of the use cases, the navigation and the creation of a prototype.

The application implementation uses Android Studio as a development environment, using the Open Food Facts public API to obtain the product's information and Firebase for user administration, notifications and database management. The project ends with the development and implementation of a test plan and a testing process carried out with potential users.

# Índice

1. Introducción .....	2
1.1. Contexto y justificación del Trabajo .....	2
1.1.1. Contexto.....	2
1.1.2. Justificación del trabajo.....	3
1.1.3. Análisis de aplicaciones nutricionales .....	5
1.2. Objetivos del Trabajo .....	7
1.2.1. Objetivos funcionales.....	8
1.2.2. Objetivos no funcionales.....	8
1.2.3. Objetivos funcionales a futuro .....	9
1.3. Enfoque y método seguido .....	9
1.3.1. Entorno de desarrollo .....	10
1.3.2. Metodología.....	10
1.4. Planificación del Trabajo .....	12
1.4.1. Planificación temporal .....	12
1.4.2. Riesgos .....	15
1.5. Breve resumen de productos obtenidos.....	15
1.6. Breve descripción de los otros capítulos de la memoria.....	15
2. Diseño y arquitectura .....	16
2.1. Diseño centrado en el usuario .....	16
2.1.1. Usuarios y contextos de uso.....	16
2.1.1.2. Análisis competitivo.....	18
2.1.1.3. Método de diario .....	18
2.1.1.4. Entrevista.....	20
2.1.1.5. Encuesta .....	21
2.1.1.6. Perfilado de usuarios .....	24
2.1.2. Diseño conceptual.....	25
2.1.2.1. Personas .....	25
2.1.2.2. Escenarios de uso .....	27
2.1.2.3. Flujos de interacción.....	32
2.1.3. Prototipado .....	32
2.1.3.1. Sketches.....	32
2.1.3.2. Prototipo.....	36
2.1.4. Evaluación .....	39

2.1.4.1. Conclusiones y mejoras sobre el prototipo.....	41
2.2. Diseño técnico .....	43
2.2.1. Casos de uso.....	43
2.2.2. Diseño de la arquitectura .....	55
3. Implementación .....	59
3.1. Herramientas y APIs utilizadas.....	59
3.1.1. Herramientas para el desarrollo.....	59
3.1.2. Librerías.....	60
3.1.3. Herramientas de ayuda al desarrollo .....	61
3.2. Desarrollo de la aplicación .....	62
3.2.1. Estructura del proyecto.....	62
3.2.2. Gestión de base de datos .....	64
3.2.3. Gestión de usuarios.....	65
3.2.4. Adaptación a otros idiomas.....	66
3.2.5. Recuperación, carga y almacenamiento de catálogo de API .....	66
3.2.6. Limitaciones de Open Food Facts.....	68
3.2.7. Búsqueda de cambios en productos favoritos y envío de notificaciones .....	69
3.2.8. Interfaz de usuario .....	70
3.2.8.1. DrawerLayout y NavigationView.....	70
3.2.8.2. BottomNavigationView.....	71
3.2.8.3. ListView .....	71
3.2.8.4. ExpandableListView .....	71
3.2.8.5. RecyclerView y Cardview .....	72
3.2.8.6. TabLayout y ViewPager.....	72
3.2.8.7. Floating Action Button .....	72
3.2.8.8. CameraSource, SurfaceView y BarcodeDetector .....	73
3.3. Funcionalidades de la aplicación.....	73
3.3.1. Gestión de usuarios.....	73
3.3.1.1. Inicio y cierre de sesión.....	73
3.3.1.2. Registro de usuario .....	74
3.3.1.3. Configuración de perfiles.....	74
3.3.1.4. Mi cuenta y borrado de cuenta de usuario.....	75
3.3.2. Gestión de la aplicación .....	75
3.3.2.1. Inicio .....	75
3.3.2.2. Ajustes .....	76
3.3.3. Escaneo de productos .....	76

3.3.3.1. Nuevo escaneo .....	76
3.3.3.2. Información de producto .....	77
3.3.3.3. Resumen de la información del producto.....	77
3.3.3.4. Ingredientes del producto .....	77
3.3.3.5. Información nutricional del producto.....	78
3.3.4. Gestión de productos.....	78
3.3.4.1. Búsquedas .....	78
3.3.4.2. Favoritos .....	79
4. Pruebas .....	79
4.1. Dispositivos utilizados en las pruebas.....	79
4.2. Pruebas automatizadas.....	81
4.3. Pruebas manuales.....	82
5. Conclusiones .....	84
5.1. Conclusiones personales.....	84
5.2. Conclusiones objetivas del proyecto.....	84
5.3. Análisis de la gestión del proyecto y planificación actualizada .....	85
5.4. Líneas de trabajo a futuro.....	86
6. Glosario .....	87
7. Bibliografía .....	88
8. Anexos.....	90
8.1. Manual de usuario .....	90
8.2. Manual de instalación.....	90

## Lista de figuras

Figura 1: Infografía de alérgenos	4
Figura 2: Captura de pantalla MyFitnessPal	5
Figura 3: Captura de pantalla Celiacs de Catalunya	5
Figura 4: Captura de pantalla OpenFoodFacts	6
Figura 5: Captura de pantalla Yazio	6
Figura 6: Captura de pantalla Yuka	6
Figura 7: Comparativa entre aplicaciones nutricionales del mercado	7
Figura 8: Fases del modelo en cascada	11
Figura 9: Carga de trabajo por fase	13
Figura 10: Gráfico de distribución de carga por fase	14
Figura 11: Gráfico de distribución de carga en la fase de implementación	14
Figura 12: Diagrama de Gantt	14
Figura 13: Diario de Lara	19
Figura 14: Diario de Ethan	19
Figura 15: Diario de Saray	19
Figura 16: Diario de Sergio	19
Figura 17: Guion para la entrevista	20
Figura 18: Resumen de las respuestas a la entrevista	20
Figura 19: Encuesta - Gráficos	21
Figura 20: Encuesta - Gráficos lugar de residencia y tiempo de uso de dispositivos	22
Figura 21: Encuesta - Gráficos relativos a uso de otras aplicaciones	22
Figura 22: Encuesta - Gráficos relativos a hábitos de compra (I)	22
Figura 23: Encuesta - Gráficos relativos a hábitos de compra (II)	23
Figura 24: Encuesta - Gráficos relativos a intolerancias	23
Figura 25: Encuesta - Gráficos a alimentación saludable	23
Figura 26: Encuesta - Gráficos relativos al etiquetado de los productos	23
Figura 27: Encuesta - Gráficos relativos a las funcionalidades de la nueva aplicación	24
Figura 28: Perfil de usuario descuidado	25
Figura 29: Perfil de usuario cuidadoso	25
Figura 30: Perfil de usuario simplista	25
Figura 31: Ficha de persona con perfil simplista	26
Figura 32: Ficha de persona con perfil cuidadoso	26
Figura 33: Ficha de persona con perfil descuidado	26



Figura 34: EU01 - Escaneo de un nuevo producto en el hogar	27
Figura 35: EU02 - Escaneo de un nuevo producto en el establecimiento	28
Figura 36: EU03 - Acceso a información nutricional de un producto desde escaneo	28
Figura 37: EU04 - Acceso a información sobre alérgenos de un producto desde búsqueda	29
Figura 38: EU05 - Acceso a información sobre aditivos de un producto desde escaneo	29
Figura 39: EU06 - Consulta de productos buscados anteriormente	30
Figura 40: EU07 - Añadir un producto al listado de productos favoritos desde escaneo	30
Figura 41: EU08 - Consulta de productos favoritos	30
Figura 42: EU09 - Consulta de cambios sobre un producto determinado	31
Figura 43: EU10 - Configuración del perfil de usuario	31
Figura 44: Flujo de interacción	32
Figura 45: Recopilación de sketches de la aplicación	36
Figura 46: Logotipo e icono de la aplicación	37
Figura 47: Pantallas del prototipo horizontal de la aplicación	39
Figura 48: Cuestionario pre-test de usuarios	40
Figura 49: Escenario y tareas test de usuarios	40
Figura 50: Cuestionario post test de usuarios	41
Figura 51: Recopilación de mejoras sobre el prototipo	43
Figura 52: Diagrama de casos de uso por paquete	44
Figura 53: Diagrama de casos de uso del sistema	45
Figura 54: CU01 - Registrar usuario	46
Figura 55: CU02 - Iniciar sesión	46
Figura 56: CU03 - Configurar perfil de usuario	46
Figura 57: CU04 - Consultar información de usuario	47
Figura 58: CU05 - Cambiar ajustes	47
Figura 59: CU06 - Cerrar sesión	48
Figura 60: CU07 - Escanear código de barras	48
Figura 61: CU08 - Cargar información de producto	49
Figura 62: CU09 - Ver información priorizada	49
Figura 63: CU10 - Ver información genérica	50
Figura 64: CU11 - Ver información sobre procesamiento y aditivos	50
Figura 65: CU12 - Ver información nutricional	51
Figura 66: CU13 - Ver búsquedas	51
Figura 67: CU14 - Añadir búsqueda	52
Figura 68: CU15 - Borrar búsqueda	53
Figura 69: CU16 - Ver favoritos	53

Figura 70: CU17 - Añadir favorito	54
Figura 71: CU18 - Borrar favorito	54
Figura 72: CU19 - Enviar notificación de cambio en información de producto	55
Figura 73: Diagrama de arquitectura de la aplicación	56
Figura 74: Diagrama de arquitectura por capas	57
Figura 75: Diagrama de clases entidad-relación	57
Figura 76: Diagrama de base de datos	58
Figura 77: Diagrama de la bases de datos interna del dispositivo	59
Figura 78: Distribución de la cuota de dispositivos Android	60
Figura 79: Estructura del proyecto - Clases java	62
Figura 80: Estructura del proyecto - Interfaz gráfica	63
Figura 81: Árbol de base de datos con usuario, perfil, búsqueda y favorito	64
Figura 82: Árbol de base de datos con detalle de producto	64
Figura 83: Firebase - Método de autenticación	65
Figura 84: Firebase - Reglas de acceso a base de datos	66
Figura 85: Firebase - Administración de usuarios	66
Ilustración 86: OpenFoodFacts - Extracto de información de producto	67
Figura 87: Firebase – Notificaciones push en diferentes versiones de Android	69
Figura 88: Menú lateral de la aplicación	70
Figura 89: Barra inferior de la aplicación	71
Figura 90: Diferentes tipos de listas de la aplicación	71
Figura 91: Lista expandible	71
Figura 92: Contenido de card de la aplicación	72
Figura 93: Pestañas de la aplicación	72
Figura 94: Botón flotante	72
Figura 95: Capturas de pantalla en los diferentes dispositivos físicos	80
Figura 96: Tabla comparativa de dispositivos de prueba	81
Figura 97: Resultados de ejecución de tests automatizados	81
Figura 98: Comparativa entre la planificación estimada y la real	85
Figura 99: Captura de pantalla del tablero de Trello	86

# 1. Introducción

En esta primera parte se detallan los aspectos más generales del proyecto, de forma que permita al lector comprender la motivación que existe detrás de este proyecto. Por ello, comprende la definición de los requisitos que este trabajo abarca, las necesidades que se pretenden cubrir y la planificación temporal del proyecto.

## 1.1. Contexto y justificación del Trabajo

### 1.1.1. Contexto

Actualmente, cada vez es más común que, en una interacción social habitual, aparezca una conversación sobre los alérgenos, bien por los continuos cambios en la legislación acerca de los etiquetados de los productos o bien porque cada vez más personas padecen algún tipo de intolerancia a alguno de ellos que hace que tenga que vigilar los alimentos que ingieren. Ambas cosas están altamente relacionadas puesto que, si los etiquetados de los productos van cambiando cada poco tiempo, aquellas personas que deben fijarse en ellos inviertan cada vez más tiempo cuando realizan la compra en el supermercado a causa de la adaptación a la lectura de dichos etiquetados.

Como se ha indicado, en los últimos tiempos hay una mayor incidencia de intolerancias alimentarias. Esta mayor incidencia puede tener muchas y muy diferentes causas, pero existen ciertos factores que pueden desencadenar determinadas intolerancias alimentarias (Atlantida, 2020), como son:

- ✓ **Factores genéticos:** el origen de la intolerancia está marcado por la predisposición de la persona. En este caso se enmarca la celiaquía, mal llamada intolerancia al gluten, que se trata de una enfermedad autoinmune que hace el cuerpo de un individuo celíaco reconozca el gluten como un patógeno y cree anticuerpos contra esta proteína. Un 95% de los celíacos presentan genética positiva referente a la celiaquía (Singlutenismo, 2020).
- ✓ **Aditivos alimentarios:** actualmente los alimentos que consumimos tienen componentes añadidos para preservar y mejorar sus condiciones. Estos aditivos son los conservadores, colorantes, emulsionantes, etc. y, en su mayor parte, son moléculas de síntesis que el organismo no puede eliminar y se van acumulando en diferentes órganos del cuerpo.
- ✓ **Compuestos tóxicos ambientales y orgánicos persistentes:** en este grupo se encuadran los contaminantes ambientales como los metales pesados y los compuestos orgánicos como los pesticidas que se utilizan todos ellos sin tener un análisis efectivo de sus efectos en el organismo.
- ✓ **Alérgenos y tóxicos naturales en los alimentos:** existen componentes presentes en prácticamente todos los alimentos que son tóxicos para el consumo humano como pueden ser las antitripsinas presentes en las legumbres y que tienen un alto impacto sobre las enzimas que se utilizan para digerir las proteínas.
- ✓ **Inmadurez y permeabilidad intestinal:** sobre todo asociados a las intolerancias a alimentos proteicos en los lactantes.
- ✓ **Episodios recurrentes de malnutrición:** relacionada directamente con la anterior.

Como se puede deducir de esta pequeña lista, gran parte de esos factores son inherentes a las condiciones del individuo y el resto tienen gran relación con la forma de vida que lleva la sociedad actual y que no parece que vaya a estar sujeta a un cambio radical a corto plazo. Es por ello que las personas no tienen ningún tipo de control absoluto sobre el posible desarrollo

de una intolerancia y solo pueden minimizar sus efectos sobre su organismo y esto, como ya se ha adelantado, lleva gran parte de su tiempo.

Además, ligado tanto al consumo de aditivos como a una mayor necesidad de vigilar lo que se come para evitar excesos de sal o azúcar, se ha producido un gran auge de productos bio, eco o bajo en calorías que, muchas veces, no tienen las ventajas que se les presuponen a los productos de este tipo. Es por ello que la lectura de la información nutricional y de los porcentajes de aditivos es una práctica cada vez más habitual.

Por otro lado, es una realidad que el uso de smartphones está cada vez más extendido. Es un porcentaje muy bajo de la población el que actualmente no dispone de un teléfono móvil inteligente y que utiliza para facilitar sus tareas diarias: calendarios, diarios, agendas, etc. Como se verá un poco más adelante en el análisis de las aplicaciones actuales que se ha realizado, existen diferentes aplicaciones que tratan de facilitar al usuario acceder a un mayor conocimiento acerca de la composición real de su alimentación diaria.

### **1.1.2. Justificación del trabajo**

Como se indicaba anteriormente, el objetivo principal del proyecto es facilitar el acceso a la información nutricional de los alimentos a los usuarios de la aplicación. Si bien esto incluye a cualquier persona que tenga interés en cuidar su alimentación o en tener mayor control sobre lo que come, el foco está realmente puesto en aquellas personas que tienen alguna limitación alimentaria. Aquí están incluidos todos aquellos con restricciones alimentarias, sea tanto por cuestiones de salud como por decisión propia puesto que, aunque los inicios sean muy diferentes, el día a día cobra gran similitud a la hora de realizar la compra.

En este punto es necesario distinguir entre alergia e intolerancia a un determinado alérgeno. La intolerancia alimentaria es la incapacidad de consumir ciertos alimentos o nutrientes sin sufrir efectos adversos en la salud que suelen limitarse a síntomas digestivos que se prolongan durante unas horas después del consumo del alimento. En cambio, una alergia alimentaria provoca una reacción del sistema inmunitario grave o incluso potencialmente mortal.

Mientras que solo un 2% de los adultos y un 3-7% de los niños padece algún tipo de alergia (Aecosan, 2020), cerca de un 30% de la población sufre algún tipo de intolerancia alimentaria. Pero el dato empeora estudiando a fondo cada una de las intolerancias. Por ejemplo, en el caso de la celiaquía, se estima que un 1% de la población está afectada por enfermedad autoinmune pero un 85% de este porcentaje está sin diagnosticar.

Afortunadamente, a día de hoy se va dando más importancia a todo tipo de restricción alimentaria, pero sí es cierto que queda mucho que mejorar en cuestiones de etiquetado y legislación. El etiquetado, a pesar de estar regulado por ley y de la obligación de contener unos aspectos obligatorios básicos, en la práctica es diferentes según supermercados y marcas por lo que es difícil para el consumidor saber exactamente lo que está consumiendo con detalle.

La información obligatoria que un etiquetado de un producto envasado debe contener (Eletiquetadocuentalmucho Aecosan, 2020) es la siguiente:

- ✓ Denominación: nombre legal o habitual del producto.
- ✓ Ingredientes: ordenados de mayor a menor peso.
- ✓ Alérgenos: destacados en negrita, sin obligación de describir las trazas.
- ✓ Cantidad neta.
- ✓ Fecha de caducidad.
- ✓ Conservación.
- ✓ Empresa y país de procedencia.
- ✓ Información nutricional.

Como se puede observar, el etiquetado de los alimentos nos provee de una gran cantidad de información en envases a menudo de pequeño tamaño lo que dificulta su lectura y la búsqueda de un posible alérgeno. Tal y como se puede llegar a inferir de lo ya comentado, en el caso de que un individuo padezca intolerancia a más de un alérgeno, el tiempo de lectura de las etiquetas se multiplica.

También se debe recordar en este punto cuáles son los principales alérgenos que se deben determinar en el etiquetado y que son los que este proyecto estudia más a fondo (Prevensystem, 2020). Para la creación de estos diagramas se ha utilizado la herramienta online y gratuita Venngage (Venngage, 2020):



**Figura 1: Infografía de alérgenos**

Existen etiquetas formales para declarar la ausencia de un determinado alérgeno, que certifican la realización de análisis periódicos para confirmar la ausencia de ese componente, pero cuyo uso no es obligatorio ni está demasiado extendido. En el caso del gluten, existen dos posibles etiquetas al respecto: sin gluten (indica que el alimento no contiene más de 20mg/kg de gluten) y muy bajo en gluten (indica que el alimento contiene una cantidad de gluten inferior a 100mg/kg).

Partiendo de las reflexiones anteriores, la motivación surge de la intención de facilitar el proceso de detección de alérgenos, así como aportar facilidad de uso y claridad en la presentación de la información nutricional y de ingredientes de los productos para que el proceso de realizar la compra sea más sencillo y rápido, así como el seguimiento de las características de los productos comprados.

También existe una motivación personal que se basa en la propia condición de la alumna que desde hace cinco años padece celiaquía. Sumado al proceso de diagnóstico y aceptación de las limitaciones derivadas de este, se encuentra la dificultad de la realización de la compra semanal, aumentando esta en gran medida cuando no se realiza en un supermercado en el que se conozcan los productos a la venta. En el caso personal de la alumna, una compra completa en un supermercado diferente al habitual puede tomar un 35% más de tiempo que si se realiza en uno conocido, por no hablar de las dificultades que se pueden encontrar al realizar viajes fuera de España y por lo que se ha tratado de buscar un origen de datos que no este limitado geográficamente al territorio español.

Además, existe una motivación educativa y de aprendizaje al tener el objetivo de, por un lado, aplicar todos los conocimientos que se han ido asimilando durante el grado y, por otro, realizar el desarrollo en la plataforma Android que, si bien ha sido siempre de gran interés para la autora de este proyecto, nunca ha dispuesto de tiempo suficiente para dar los primeros pasos en ella. Es por eso que supone un doble esfuerzo, por un lado, acometer un desarrollo que abarca varios desafíos totalmente nuevos (implementar un escáner de códigos de barras, recuperación y visualización de información de recursos externos, gestión de usuarios, gestión

de bases de datos remotas) y, por otro, realizarlo sobre una plataforma del todo desconocida, con lo que se espera un gran aprendizaje durante su desarrollo.

Es por ello que todos estos puntos en su conjunto proporcionan la motivación para el desarrollo de una aplicación que recupere la información de un determinado producto identificado por el escaneo de su código de barras y cuya visualización está determinada por las preferencias que haya seleccionado el usuario en su perfil. Además, las búsquedas se guardan y los productos se pueden categorizar como favoritos para que, en base a la periodicidad seleccionada por el usuario, sea posible recuperar la información y enviar notificaciones de actualización de información si esta ha cambiado según el interés del usuario.

### 1.1.3. Análisis de aplicaciones nutricionales

Actualmente, tras el auge de la alimentación sana y el aumento de las intolerancias, existen gran cantidad de aplicaciones relacionadas con esto y que ofrecen distintos tipos de funcionalidades. Tras realizar una búsqueda de las aplicaciones nutricionales disponibles para Android en Google Play y filtrarla con ayuda del contenido de la red (Psicologiyamente, 2020), se ha instalado, probado y analizado las siguientes aplicaciones nutricionales:

#### ✓ MyFitnessPal:

Tiene un objetivo bastante más amplio, proporcionando también rutinas de ejercicios, aporta un escanear de productos y un sistema de trazado de alimentos consumidos. Su uso está más enfocado al conteo diario de calorías. Muchas de sus funcionalidades necesitan de suscripción premium.

En su página de descarga indican que cuenta con una base de datos de alimentos que supone más de 6,000,000 de alimentos.

<https://play.google.com/store/apps/details?id=com.myfitnesspal.android&gl=ES>

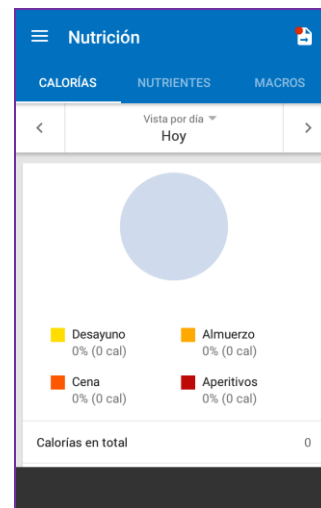


Figura 2: Captura de pantalla MyFitnessPal



Figura 3: Captura de pantalla Celiacs de Catalunya

#### ✓ Celiacs Catalunya:

También tiene un objetivo más amplio, puesto que se trata de la app que proporciona la asociación de celíacos de Cataluña. Provee de geolocalización de establecimientos certificados sin gluten y de un escanear de productos, pero la información que proporciona es esquemática y sin gran detalle y está sujeta a suscripción a la asociación.

En su página de descarga indican que entre sus funcionalidades ofrecen localizar establecimientos donde encontrar alimentos y comida sin gluten, dentro de su lista .

<https://play.google.com/store/apps/details?id=org.celiacscatalunya.mobileapp&gl=ES>

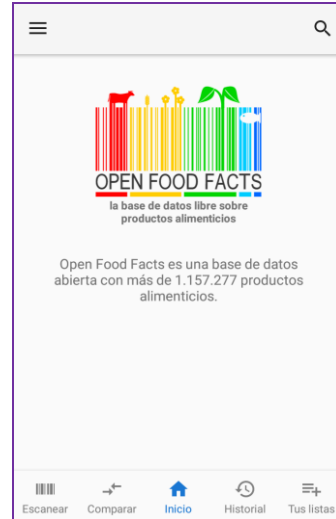
✓ **OpenFoodFacts:**

Aporta un escaneador de productos y gran cantidad de información acerca de cada producto, permitiendo indicar la información en la que el usuario está más interesado.

No es una aplicación comercial, sino que se realiza mediante contribuciones de la comunidad.

En su página de descarga indican que la aplicación permite ver los 1.200.000 productos contenidos en la base de datos abierta y gratuita Open Food Facts, y también añadir imágenes y datos para nuevos productos.

<https://play.google.com/store/apps/details?id=org.openfoodfacts.scanner&gl=ES>



**Figura 4: Captura de pantalla OpenFoodFacts**



**Figura 5: Captura de pantalla Yazio**

✓ **Yazio:**

Aporta un escaneador de productos siendo su foco principal el conteo de calorías para realizar dietas y procesos de adelgazamiento.

Su enfoque es mucho más amplio de lo que se detalla en este proyecto y gran parte de las funcionalidades que ofrece son premium.

En su página de descarga indican que la aplicación permite gestionar un diario de alimentos y registrar la actividad diaria del usuario.

<https://play.google.com/store/apps/details?id=com.yazio.android&gl=ES>

✓ **Yuka:**

Esta aplicación aporta un escaneador de productos centrandó su información en la categorización del producto según su nivel de procesado y su información nutricional.

Tiene alguna funcionalidad premium pero el grueso de sus funciones es gratis.

En su página de descarga indican que proporcionan información sobre 700.000 productos alimentarios de los que se evalúan sus características en función de calidad nutricional, presencia de aditivos y carácter ecológico del producto




<https://play.google.com/store/apps/details?id=io.yuka.android&gl=ES>



**Figura 6: Captura de pantalla Yuka**

Atendiendo a las características que ofrece cada una de estas aplicaciones, se ha elaborado la siguiente tabla comparativa:

	MYFITNESS PAL	CELIACS CATALUNYA	OPENFOOD FACTS	YAZIO	YUKA
Registro requerido	✓	✗	✗	✗	✓
Toda su funcionalidad es gratis	✗	✗	✓	✗	✗
Escaneado de productos	👑	✓	✓	👑	✓
Información sobre alérgenos	👑	✓	✓	👑	✗
Información nutricional	👑	✗	✓	👑	✓
Información sobre procesados	👑	✗	✓	👑	✓
Alertas configurables	✗	✗	✓	✗	👑
Guardado de productos	✓	✗	✓	✗	✓
Información sobre aditivos	👑	✗	✓	👑	✓
Interfaz intuitiva y sencilla	✓	✗	✗	✓	✓

 Funcionalidad presente    
  Funcionalidad ausente    
  Funcionalidad premium

**Figura 7: Comparativa entre aplicaciones nutricionales del mercado**

Los objetivos y funcionalidades disponibles del proyecto se han definido a partir de este análisis y será en la sección correspondiente al diseño en la que se describirá en detalle todo esto.

El objetivo principal se fundamenta en ofrecer todas las funcionalidades analizadas sobre las diferentes aplicaciones en un mismo proyecto de forma gratuita y con una interfaz sencilla y usable. Como se verá en el siguiente apartado, hay características que se quedarán fuera del alcance de este proyecto para su implementación a futuro.

## 1.2. Objetivos del Trabajo

Como ha quedado demostrado con el análisis de algunas de las aplicaciones nutricionales existentes incluido en el apartado anterior, existen multitud de aplicaciones en los markets de las diferentes plataformas que realizan parte de las funcionalidades que abarca este proyecto, pero, o no las proporcionan todas, o lo hacen con una interfaz pobre y poco intuitiva.

El principal objetivo de este proyecto es el desarrollo de una aplicación móvil que, a partir del escaneo del código de barras, proporcione de forma clara y simple la información de interés para el usuario según el perfil que previamente creó y el resto de información disponible sobre el producto para su consulta.



A partir de esta definición, y según lo indicado en la parte de la justificación del proyecto, podemos determinar que este objetivo está compuesto por otros genéricos e inherentes a él como son los siguientes:

- ✓ **OBJ1** – Poner en práctica los conocimientos adquiridos durante el Grado en Informática.
- ✓ **OBJ2** – Ser un proyecto basado en una necesidad real no satisfecha del todo por las aplicaciones actualmente disponibles.
- ✓ **OBJ3** – Aportar sencillez de uso y cumplir con los principios de usabilidad.
- ✓ **OBJ4** – Asegurar la mantenibilidad y adición de mejoras a futuro.

Para conseguir estos objetivos será necesaria la confección de un plan de trabajo adecuado, así como la división de este objetivo de alto nivel en otros de más detalle.

### 1.2.1. Objetivos funcionales

En este subapartado se recogen los objetivos funcionales que definen la funcionalidad de la aplicación resultado de este proyecto. Por lo tanto, el objetivo de alto nivel indicado anteriormente se divide en los siguientes objetivos funcionales:

- ✓ **OF1** – Implementar una aplicación móvil cuya funcionalidad principal sea un lector de códigos de barras que permita, mediante llamadas a un servicio externo, la obtención de la información nutricional de los productos.
- ✓ **OF2** – Crear y gestionar usuarios para el acceso a la aplicación utilizando un gestor de usuarios externo.
- ✓ **OF3** – Crear y editar perfiles para cada usuario con las preferencias de mostrado prioritario de información de cada uno de ellos.
- ✓ **OF4** – Utilizar la cámara del smartphone del usuario para obtener la información del código de barras del producto.
- ✓ **OF5** – Recuperar y mostrar según preferencias del usuario la información correspondiente al producto mediante llamadas a un servicio externo.
- ✓ **OF6** – Guardar y administrar las búsquedas realizadas, guardando en una base de datos externa la información prioritaria para el usuario.
- ✓ **OF7** – Categorizar y gestionar los productos escaneados como favoritos.
- ✓ **OF8** – Actualizar periódicamente la información de las búsquedas guardadas.
- ✓ **OF9** – Enviar notificaciones cuando la información de alguno de los productos favoritos haya cambiado.

En este caso, los objetivos funcionales se han tratado como los clásicos requisitos de la aplicación, es decir, las funcionalidades a las que el usuario tendrá acceso sin tener en cuenta los aspectos técnicos o de visualización.

### 1.2.2. Objetivos no funcionales

Los objetivos no funcionales se refieren a aquellos objetivos que no forman parte de la descripción de las funcionalidades que proporciona el producto final pero que son inherentes a su funcionamiento. Los objetivos no funcionales a los que se presta especial atención durante el desarrollo de este proyecto son los siguientes:

- ✓ **ONF1** – Garantizar que la interfaz de la aplicación es intuitiva y simple, con cierto grado de modernidad siguiente los estándares actuales.

- ✓ **ONF2** – Testear dicha interfaz se testeará para adecuarla a su uso en smartphones, quedando su adaptación a tablets como un desarrollo futuro.
- ✓ **ONF3** – Mostrar la información de forma amigable, utilizando un lenguaje cercano para que cualquier usuario, sea cual sea su origen o formación pueda entender lo que está visualizando.
- ✓ **ONF4** – Asegurar un proceso de registro lo suficiente sencillo como para que los usuarios tengan confianza en el proceso y, a la vez, no sea un proceso tedioso.
- ✓ **ONF5** – Ajustar los tiempos de respuesta de la aplicación en la mayor medida posible a la respuesta de los servicios externos con los que se comunica.
- ✓ **ONF6** – Garantizar que la aplicación se comunique con los servicios externos sin realizar comportamientos extraños.
- ✓ **ONF7** – Asegurar que el sistema sea fiable, sin producir errores no controlados que produzcan el cierre de la aplicación, siendo el resultado final similar a un producto listo para su pase a producción.
- ✓ **ONF8** – Garantizar que los mensajes que la aplicación muestra al usuario tengan un número adecuado para indicarle en todo momento lo que se está realizando y con el detalle suficiente para transmitirle la información acerca de errores producidos o acciones esperadas por su parte.
- ✓ **ONF9** – Seguir las consideraciones generales de seguridad en el desarrollo y funcionamiento de la aplicación sin poner un foco especial en estas cuestiones.

### **1.2.3. Objetivos funcionales a futuro**

Teniendo en cuenta el carácter delimitado en tiempo y esfuerzo de un proyecto de estas características, existen ciertas funcionalidades que no es posible abarcar durante el desarrollo de este proyecto pero que, de acometerse en un futuro, sin lugar a dudas mejorarían la aplicación actual.

Como se verá más adelante, la aplicación tomará como fuente de datos un servicio externo. Esto hace que sea dependiente de la información que éste devuelve. Por ello, en caso de que un código de barras no se encuentre en la base de datos externa, la aplicación muestra un mensaje informando de esta situación y el proceso habrá terminado. Este servicio externo permite la inclusión de nuevos productos en la base de datos y sería una funcionalidad muy interesante de añadir a la actual.

Además, también se podría aprovechar la cámara del smartphone para realizar escaneos de las etiquetas para añadir el contenido extraído de dichos escaneos tras aplicarles un proceso de OCR.

También podrían ser añadidas en un futuro funcionalidades que abarquen el proceso de compartir en redes sociales los productos favoritos de los usuarios o el envío de mensajes de salud mediante notificaciones dentro de la propia aplicación.

### **1.3. Enfoque y método seguido**

En este apartado se especificarán las generalidades del entorno de desarrollo sobre el que se va a trabajar, así como la metodología de desarrollo que se ha elegido para llevar a cabo este proyecto. Para poner en contexto ambas secciones, se indicará el motivo por el que se ha decidido realizar una aplicación nativa y en la plataforma Android.

Según la funcionalidad que se quiere conseguir en el producto final, se necesita un gran control sobre los recursos que proporciona un smartphone, especialmente sobre la cámara. Realizando el desarrollo de una aplicación nativa, este control es total mientras que, si se utiliza alguno de los frameworks al uso que existen actualmente, las funciones a las que se puede acceder están bastante delimitadas. Lo mismo ocurre si se tratan temas como el rendimiento o la estabilidad, estos siempre serán superiores si se trata de aplicaciones nativas (YML, 2020).

El hecho de desarrollar para la plataforma Android tiene una motivación principalmente de tipo práctico. Aunque la alumna si dispone de un dispositivo iPhone para realizar las pruebas, no tiene a su alcance un equipo Macintosh que le permita realizar este desarrollo en plenas garantías. Además, esta decisión también atiende a la cuota de mercado ocupada por cada una de las plataformas: los dispositivos Android suponen el 86% de los móviles vendidos a nivel mundial en el primer trimestre de 2020 (Group, I., 2020). Es por ello que se trata del sistema operativo para móviles de mayor uso.

### 1.3.1. Entorno de desarrollo

Como se ha indicado, el proyecto se basa en el desarrollo de una aplicación nativa para móviles Android. Por lo tanto, se utiliza el IDE oficial para desarrollar este tipo de aplicaciones: Android Studio utilizando como lenguaje de programación Java y su SDK integrado en el propio IDE. Su compilación está basada en Gradle que facilita en gran medida la descarga de las dependencias necesarias durante el desarrollo y la integración con emuladores de Android para realizar pruebas preliminares en local. En este punto hay que recalcar que, para evitar problemas de compatibilidades con dispositivos reales, siempre se realizan pruebas en al menos dos móviles físicos poniendo como versión mínima soportada Android 6.0 (Marshmallow), siendo las versiones de estos móviles físicos Android 6.0 (Sony Xperia F3311) y Android 10 (Samsung Galaxy 9).

Para la obtención de los datos correspondientes a los códigos de barras se utiliza el API REST abierta de **Open Food Facts** que permite el acceso a una base de datos de productos alimentarios en la que la comunidad va contribuyendo y que permite su uso para otras aplicaciones y servicios (Open Food Facts, 2020). Tiene información de gran cantidad de productos de todo el mundo y el detalle de los datos depende, en gran medida, de las aportaciones que recibe. Esta información está libre bajo la licencia ODbL License que permite su adaptación y uso siempre y cuando se atribuya el origen de los datos.

Para el mantenimiento de los usuarios, la gestión de la base de datos y el envío de notificaciones se utiliza Firebase que es fácilmente integrable con Android y proporciona gran variedad de servicios todos ellos bajo Google (Firebase, 2020). Al igual que con el API de datos, en este apartado se nombran los servicios a utilizar y, en los siguientes, se dará más detalles de cada uno de ellos, así como de la problemática de uso:

- ✓ **Authentication:** para la gestión de usuarios.
- ✓ **RealTime Database:** base de datos en tiempo real que permite ser gestionada y sincronizada incluso sin conexión.
- ✓ **Cloud Messaging:** para el envío de mensajes de notificación de forma segura.

### 1.3.2. Metodología

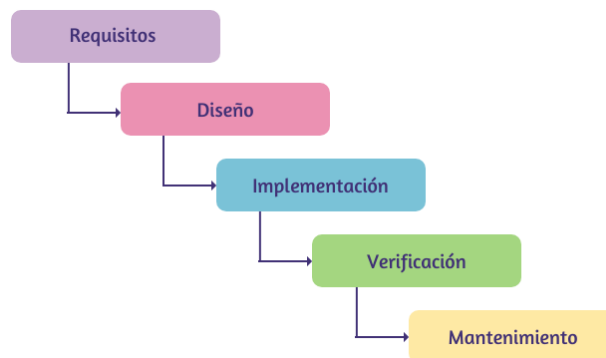
Teniendo en cuenta lo indicado en los anteriores apartados y en los requerimientos en base a las fases de la memoria y documentación del proyecto, se ha optado por aplicar un modelo clásico en cascada para el desarrollo de una nueva aplicación móvil nativa puesto que las diferentes fases del proyecto se encuentran muy bien definidas y, dado que no se extiende en

el tiempo durante una gran cantidad de meses, no supone caer en las limitaciones y riesgos más comunes de este tipo de metodología. Estos pueden ser:

- ✓ Falta de definición de objetivos.
- ✓ Falta de agilidad al resolver problemas en las especificaciones.
- ✓ Mayor complejidad en la resolución de errores.

Todo esto tiene una mayor incidencia cuando se trata de proyectos largos, complejos o sujetos a una gran indefinición. En este caso, el plazo de desarrollo del proyecto es relativamente corto, la funcionalidad esperada está muy acotada y los objetivos son claros y están bien definidos, en gran parte por las limitaciones en los recursos que se pueden destinar a la resolución del proyecto.

Este modelo es secuencial, ejecutando cada fase después de otra y su nombre indica la disposición de cada una de las fases una encima de otra (Cascada, E. and Domínguez, P., 2020). Para la creación de estos diagramas se ha utilizado la herramienta online y gratuita Venngage (Venngage, 2020).



**Figura 8: Fases del modelo en cascada**

Entrando un poco más en detalle en las fases en las que se compone este modelo, podemos indicar lo siguiente:

- ✓ **Fase de requisitos:** es la fase correspondiente al análisis de las necesidades que se quieren cubrir y la definición de los requisitos necesarios. Es una fase vital para el buen desarrollo posterior del proyecto tanto respecto a los objetivos conseguidos como a la estimación de recursos. En este proyecto se corresponde con los puntos de contexto, objetivos, metodología y plan de trabajo.
- ✓ **Fase de diseño:** en esta fase se organiza el sistema en elementos independientes y se definen las relaciones de estos elementos entre sí. Es durante esta etapa cuando se definen los casos y contextos de uso, la arquitectura de la aplicación y se crea un prototipo.
- ✓ **Fase de implementación:** es en esta fase donde se llevan a la práctica los objetivos y requerimientos definidos en las fases anteriores traduciéndose al desarrollo de la aplicación.
- ✓ **Fase de verificación:** una vez terminada la fase de implementación, tiene lugar la fase de pruebas en la que se estudia la calidad del software desarrollado en base a los defectos o errores encontrados en la realización de las pruebas.
- ✓ **Fase de mantenimiento:** se corresponde con la puesta en marcha de la aplicación y la garantía de su correcto funcionamiento.

En este proyecto, este modelo se altera un poco puesto que la fase de implementación y de verificación se fusionan en una única fase para garantizar que todo el desarrollo realizado y entregado funciona correctamente sin necesidad de esperar a que se termine la totalidad del desarrollo. Respecto a la fase de mantenimiento, en este caso se ve sustituida por la entrega final en la que la estudiante debe garantizar que el software desarrollado cumple con los requerimientos definidos en la memoria y que este desarrollo funciona correctamente.

## 1.4. Planificación del Trabajo

Para realizar una planificación de trabajo efectiva hay que tener cuenta dos puntos: por un lado, la planificación temporal y la estimación de esfuerzo en función de la carga de trabajo y, por otro, los riesgos que pueden producirse durante el tiempo de vida del proyecto.

### 1.4.1. Planificación temporal

El primer paso para establecer una planificación temporal de un proyecto es realizar la estimación de recursos disponibles y de esfuerzo necesario para hacer frente a la carga de trabajo que se debe llevar a cabo.





Este proyecto se enmarca en la asignatura Trabajo Final de Grado que según el número de créditos que tiene asignados (12 ECTS) se corresponden con una carga lectiva de unas 300 horas. La duración del proyecto se comprende entre el 16 de septiembre de 2020 y el 1 de enero de 2021. Teniendo en cuenta que todo el trabajo lo debe realizar una única persona, la estudiante, y que son 15 las semanas que componen la duración del proyecto, la carga esperada de horas por semana se establece en un total de 20 horas. Si se tiene en cuenta que la alumna tiene un trabajo a jornada completa y los fines de semana libres, la estudiante puede asumir la siguiente carga semanal:






















- ✓ 2 horas de dedicación con una periodicidad de cuatro días laborables por semana.
- ✓ 3 horas de dedicación cada viernes.
- ✓ 5 horas de dedicación cada sábado y domingo.

Con esta planificación de esfuerzo, se plantea dedicar una hora semanal más de lo previsto en la carga del proyecto y, además, se deja un día libre de asignación para dedicar a posibles contingencias que vayan surgiendo y para poder asumir el trabajo derivado de la otra asignatura que tiene matriculada la alumna para este curso académico.

En la planificación temporal también se indican los hitos planificados en la organización de la asignatura que se traducen en entregas parciales de los componentes entregables del proyecto y que se realizarán de forma periódica en los plazos estipulados por los consultores.

Teniendo esto cuenta, la carga de trabajo del proyecto quedaría distribuida del siguiente modo, a partir de los puntos a trabajar en la memoria y el trabajo a realizar:

	Horas estimadas	Carga total	Fecha de inicio	Fecha de fin	Situación
<b>Plan de trabajo</b>	<b>42</b>	<b>13 %</b>	<b>16/09/2020</b>	<b>30/09/2020</b>	
Contexto y justificación	10	3 %	16/09/2020	19/09/2020	
Objetivos	16	5 %	19/09/2020	25/09/2020	
Enfoque y método utilizado	4	1 %	26/09/2020	26/09/2020	

Planificación y riesgos	12	4 %	26/09/2020	30/09/2020	
<b>Diseño y arquitectura</b>	<b>84</b>	<b>26 %</b>	<b>01/10/2020</b>	<b>28/10/2020</b>	
Usuarios y contextos de uso	16	5 %	01/10/2020	05/10/2020	
Diseño conceptual	20	6 %	05/10/2020	11/10/2020	
Prototipado y evaluación	32	10 %	12/10/2020	23/10/2020	
Diseño técnico y arquitectura	18	5 %	23/10/2020	28/10/2020	
<b>Implementación y pruebas</b>	<b>126</b>	<b>40 %</b>	<b>29/10/2020</b>	<b>09/12/2020</b>	
Implementación	95	30 %	29/10/2020	01/12/2020	
Módulo de integración	15	5 %	29/10/2020	01/11/2020	
Módulo de login	15	5 %	02/11/2020	07/11/2020	
Módulo de perfiles de usuario	10	3 %	07/11/2020	10/11/2020	
Módulo de escaneo	30	8 %	11/11/2020	22/11/2020	
Módulo de información de productos	15	5 %	22/11/2020	28/11/2020	
Módulo de administración de búsquedas	5	2 %	28/11/2020	29/11/2020	
Módulo de gestión de productos favoritos	5	2 %	29/11/2020	01/12/2020	
Testing y pruebas usuario	31	10 %	27/11/2020	09/12/2020	
<b>Entrega final</b>	<b>68</b>	<b>21 %</b>	<b>10/12/2020</b>	<b>01/01/2021</b>	
Mejoras y correcciones	20	6 %	10/12/2020	15/12/2020	
Memoria	24	7 %	16/12/2020	24/12/2020	
Manual de usuario	12	4 %	24/12/2020	28/12/2020	
Presentación	12	4 %	29/12/2020	01/01/2021	
<b>Totales</b>	<b>320</b>	<b>100%</b>	<b>16/09/2020</b>	<b>01/01/2021</b>	



En curso



Borrador



Terminado



Pendiente

**Figura 9: Carga de trabajo por fase**

Los iconos que se muestran en las diferentes tablas de esta memoria están disponibles en Internet para su uso con licencia de atribución (Flaticon, 2020).

Como se puede observar, la fase que más carga de trabajo tiene es la de implementación y pruebas, seguida de la de diseño en la que se definen los puntos particulares a seguir para la implementación y que se han definido en la fase de menos carga de trabajo, la de definición de objetivos y análisis.

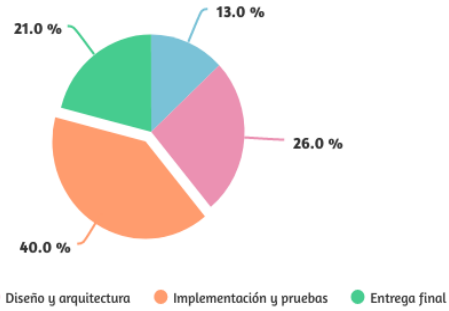


Figura 10: Gráfico de distribución de carga por fase

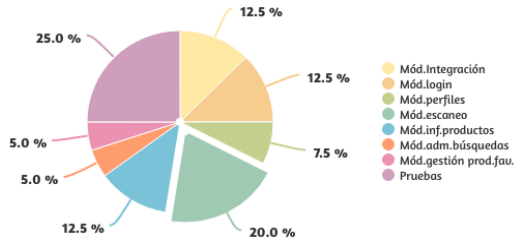


Figura 11: Gráfico de distribución de carga en la fase de implementación

Si atendemos a la distribución de la fase de implementación, la más larga del proyecto, vemos que la carga de horas quedará repartida del siguiente modo:

A continuación, se muestra un diagrama de Gantt que muestra la distribución temporal de la planificación recogida en la tabla anterior de un modo más gráfico. Para la creación de estos diagramas se ha utilizado la herramienta online y gratuita Venngage (Venngage, 2020).

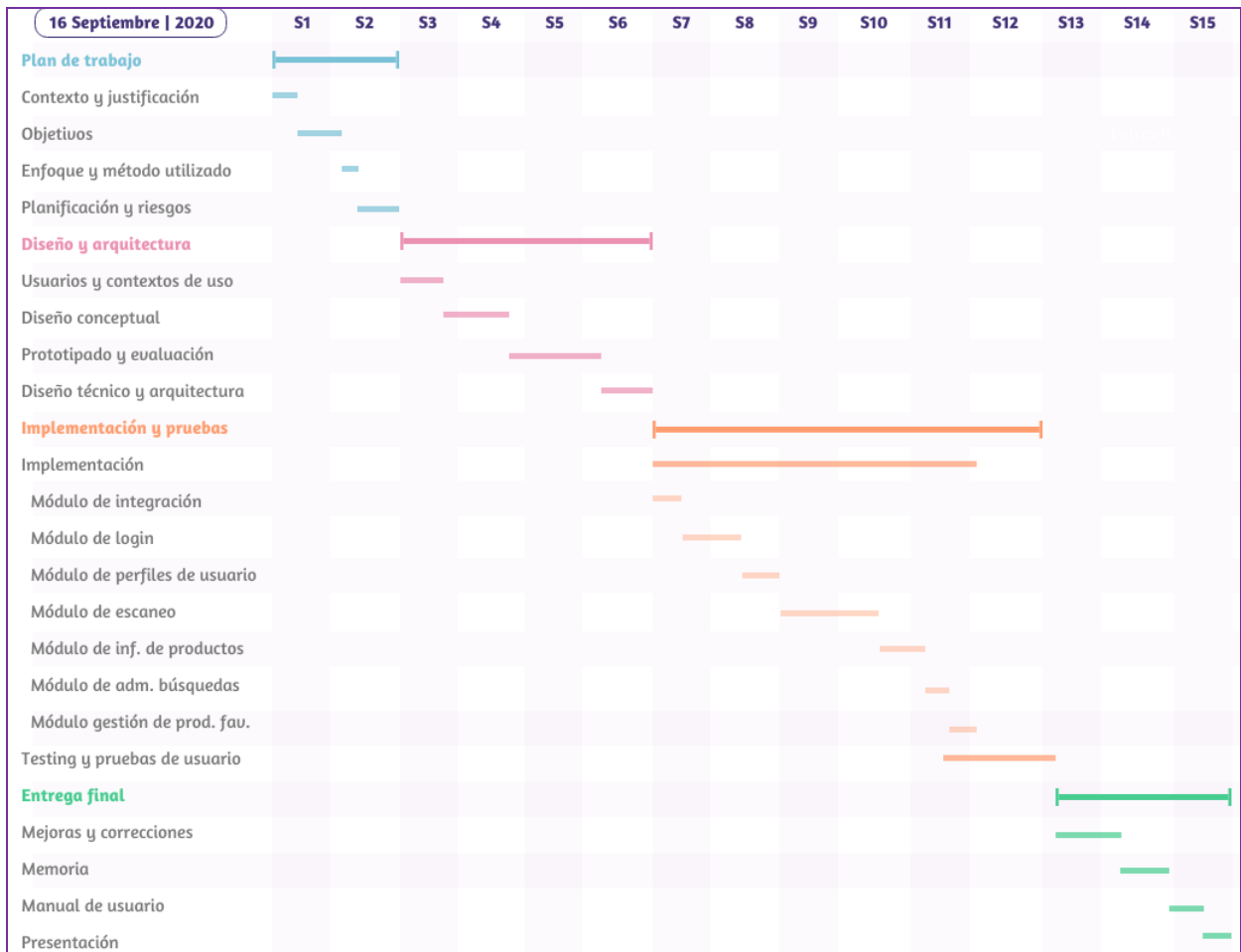


Figura 12: Diagrama de Gantt

Se debe tener en cuenta que el diagrama de Gantt tiene una distribución temporal, por lo que su visualización muestra el espacio temporal en el que se realizan las diferentes tareas. Hay que tener presente que la carga de trabajo por día no es siempre la misma, como se ha indicado anteriormente, a causa de la disponibilidad de la estudiante.

### **1.4.2. Riesgos**

En todo proyecto se pueden producir situaciones imprevistas que pongan en riesgo la finalización exitosa del proyecto, bien porque no se cumple la planificación o porque el producto final no resulta acorde a las especificaciones que se plantearon al inicio.

Existe la posibilidad de que el conocimiento técnico que se adquiriera a lo largo del proyecto no sea suficiente para la consecución de los objetivos definidos. Esto se ha tenido en cuenta en la planificación, debido en gran medida al desconocimiento de la plataforma de desarrollo y a ser el primer proyecto que se acomete en esta tecnología, pero siempre pueden suceder imprevistos que hagan que el curso del proyecto se salga del camino planificado.

También hay que tener en cuenta que tanto la planificación, la estimación de recursos y el diseño de las funcionalidades se realizan desde el desconocimiento de la tecnología y desde la inexperiencia en la realización de este tipo de proyectos en un tiempo tan delimitado. Es posible que todo ello requiera que haya que volver atrás para rehacer todo aquello insuficiente o incorrecto.

Además, existen otros inconvenientes tanto laborales como personales que pueden tener lugar durante el tiempo de duración del proyecto que hagan que el número de horas dedicadas en semanas puntuales sean inferiores a lo planificado.

En caso de que alguna de estas situaciones se produzca y no sea posible realizar un aumento de horas de dedicación para salvar la situación, se podrá plantear la reducción de alcance del proyecto para limitar las funcionalidades que ofrece y asegurar la máxima calidad posible en la implementación. No se trata de la alternativa más conveniente, pero, en caso de que la finalización del proyecto peligre, será más óptimo seguir esta vía.

### **1.5. Breve resumen de productos obtenidos**

Como resultado de este proyecto, se obtendrán los siguientes entregables o productos finales:

- ✓ Aplicación móvil desarrollada para la obtención de la información nutricional a partir del escaneo de un código de barras de un determinado producto, tanto su código fuente como el archivo instalable .apk.
- ✓ Memoria que explique el desarrollo de las diferentes fases de las que consta el proyecto.
- ✓ Manual de usuario que facilite el uso de la aplicación.
- ✓ Presentación en vídeo resumiendo el curso del proyecto y detallando los aspectos más relevantes.

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

La memoria consta de los siguientes capítulos en su estructura:

- ✓ **Introducción:** es el capítulo que da contexto al proyecto, indicando con detalle en qué consiste y cuál es la justificación para llevarlo a cabo. Además, también recoge la definición de los objetivos y requerimientos que marcarán los siguientes pasos hasta llegar al desarrollo. En la misma línea, se define también el plan de trabajo que se seguirá durante el proyecto.



- ✓ **Diseño:** en este apartado se realiza la definición de los contextos de usuario y los casos de uso posibles en la aplicación, mediante la definición de usuarios ficticios. También se lleva a cabo una descripción de la arquitectura de la aplicación, del flujo de navegación que tiene disponible y, por último, se proporciona un prototipo.
- ✓ **Implementación:** en este capítulo se indican los aspectos más técnicos, relativos al desarrollo de la aplicación, desde la definición de su estructura hasta la descripción de todos los elementos que componen sus diferentes funcionalidades.
- ✓ **Pruebas:** en este apartado se detalla el plan de pruebas que se construye a partir de los casos de uso definidos, tanto para las pruebas de la aplicación como para los tests con usuarios y los resultados que se obtienen de todo ello. También recoge los defectos encontrados y las mejoras necesarias después del análisis de los datos obtenidos.
- ✓ **Conclusiones:** en este apartado se analiza el curso del proyecto desde todos los puntos de vista posibles para poder recoger las mejoras posibles tanto en el desarrollo como en la gestión del proyecto. También recoge una descripción de las líneas de trabajo futuras que no han sido tratadas en el proyecto.
- ✓ **Glosario:** este punto funciona como referencia y recoge los términos más utilizados en la memoria que puedan necesitar aclaración.
- ✓ **Bibliografía:** recoge las fuentes consultadas para elaborar la memoria.
- ✓ **Anexos:** en este apartado se incluye el manual de usuario y cualquier otro documento que sea demasiado extenso para ser incluido en la memoria como puede ser el plan de pruebas o las encuestas que se realicen a usuarios.

## 2. Diseño y arquitectura

En esta parte del proyecto, una vez se han establecido los requisitos a cumplir, se organiza el sistema en elementos independientes con las relaciones de interacción entre sí. Es una amplia fase en la que se realizan multitud de tareas para poder detectar las funcionalidades que deberá tener el producto final y, también, la aproximación técnica para llevarlas a cabo.

### 2.1. Diseño centrado en el usuario

En esta fase se recopila información de los usuarios potenciales para conocer información relacionada con el producto que se va a crear, se categorizan los usuarios en diferentes perfiles y, a partir de estos, se definen sus contextos de uso y los diagramas de interacción. Además, también se da forma a la aplicación mediante la creación y evaluación de un prototipo horizontal de alta fidelidad.

En este paradigma de diseño, el usuario se sitúa en el centro del proceso de confección del producto y sus fases son iterativas de forma que se consiga que los productos finales cumplan con los requerimientos de los usuarios a partir de la información recopilada durante este proceso en el que se determinan las características de los usuarios objetivo, sus necesidades y objetivos y, además, el contexto en el que se va a utilizar el producto (X-Wiki UOC, 2020).

#### 2.1.1. Usuarios y contextos de uso

Como se ha comentado anteriormente, el objetivo principal de esta fase es conocer a fondo las características de los usuarios potenciales de la aplicación, así como sus necesidades y el contexto en el que van a hacer uso del producto final. Por todo ello, se explican de forma breve las diferentes técnicas de indagación que existen actualmente, indicando cuáles son las

utilizadas durante el tiempo de vida de este proyecto y, a continuación, se explica cada una de ellas de forma detallada en el marco del trabajo que se desarrolla en esta memoria.

Las técnicas de indagación actúan como iniciadoras del diseño centrado en el usuario y sirven para obtener la información que se necesita acerca de los usuarios potenciales del producto, especialmente en lo relativo a sus necesidades, objetivos y sus contextos de uso. Todas las técnicas de indagación son compatibles entre sí y pueden ser utilizadas en las diferentes fases del proceso de diseño. En este punto es necesario indicar que los contextos de uso en el caso del desarrollo de aplicaciones móviles ya no son estáticos, sino que la movilidad de los usuarios constituye una de sus mayores ventajas e inconvenientes a la vez.

Las principales técnicas de indagación existentes son las siguientes (X-Wiki UOC, 2020) (Garreta Domingo, Muriel y Mor Pera, Enric. 2020):

- ✓ **Observación e investigación contextual:** esta técnica es útil desde el punto de vista de conocer el entorno en el que se desarrollan los usuarios, de forma que el observador se desplaza al lugar donde se hace uso de la aplicación. Teniendo en cuenta el carácter no estático de los entornos de uso de aplicaciones móviles, no parece adecuada en este caso.
- ✓ **Método de seguimiento:** consiste en observar a los participantes mientras llevan a cabo sus actividades cotidianas, de forma que se intervenga lo menos posible con sus acciones para no alterar los resultados. Así, se puede llegar a conocer los patrones de comportamiento, hábitos y rutinas en la vida cotidiana del usuario. Dada su característica intrusiva en la intimidad del participante y teniendo en cuenta los momentos que se están viviendo actualmente en nuestra sociedad, la estudiante no considera adecuado utilizar este método de indagación y prefiere optar por otros que sean más seguros desde el punto de vista sanitario y menos intrusivos en la vida de los participantes.
- ✓ **Método de diario:** en esta técnica es el participante quien reúne la información acerca de la actividad que se investiga. Utiliza muestras pequeñas. Aunque es una técnica que puede obviar datos relevantes y que exigen de un esfuerzo por parte del participante, se opta por ella al tratarse del estudio de una tarea mecánica como es la compra habitual.
- ✓ **Análisis competitivo:** se trata de realizar un análisis de la competencia, formada por aquellos productos que comparten características con la aplicación que se desarrolla en este proyecto. En este caso, se elige utilizar esta técnica limitando su alcance a las aplicaciones de temática similar para ver qué valor añadido se puede conseguir.
- ✓ **Entrevistas en profundidad:** es una técnica que utiliza muestras pequeñas con guiones abiertos para tratar de entender las necesidades y preferencias de los usuarios que se han estudiado. Esta técnica también se utiliza durante el tiempo de vida de este proyecto.
- ✓ **Dinámicas de grupo:** son entrevistas que se llevan a cabo en un grupo de un número alto de personas y en las que el trabajo del moderador tiene una gran importancia para que la información recogida sea de calidad. Al igual que en el caso del método de seguimiento, teniendo en cuenta los momentos sanitarios que estamos viviendo, no se considera recomendable por parte de la alumna la utilización de esta técnica de indagación.
- ✓ **Encuestas:** se llevan a cabo en muestras grandes de usuarios y utilizan formularios estructurados con preguntas con todas las alternativas posibles de respuesta. También se opta por el uso de esta técnica durante el tiempo de vida de este proyecto.

En resumen, se utiliza en una primera iteración la técnica cualitativa del **análisis competitivo** para tratar de dar marco al proyecto y justificar su necesidad. Luego, en posteriores iteraciones se hace uso de las **encuestas** como el principal origen de datos al tratarse de un método cuantitativo, pero, por temor a no llegar a proporcionar datos de una muestra lo suficientemente significativa en la situación que estamos viviendo, se preparan también los

métodos de las **entrevistas** y el **diario**, siempre teniendo en cuenta que los datos obtenidos también son cualitativos y que la muestra que se utiliza es de un tamaño pequeño.

Los motivos que llevan a esta elección son, además, los detallados a continuación:

- ✓ Construir una base lo más sólida posible de cara a la creación de un diseño que sea útil para los usuarios y que cumpla con la funcionalidad pretendida.
- ✓ Evitar riesgos en el ámbito sanitario al estar actualmente viviendo un hecho excepcional, como es una segunda ola durante una pandemia mundial.
- ✓ Escuchar lo que tienen que decir los usuarios en relación a sus necesidades actuales dándoles opción a hacerlo sin necesidad de seguir un guion previamente establecido.

### **2.1.1.2. Análisis competitivo**

Como se ha indicado anteriormente, esta técnica se utiliza durante la fase de justificación y contexto del proyecto para dar marco al mismo y demostrar que no existe actualmente otra aplicación que proporcione todos los servicios que se ofrecen en este caso y de forma gratuita. En su momento se analizaron hasta cinco aplicaciones diferentes de la misma temática y su tabla comparativa se encuentra en el apartado *1.1.3. Análisis de aplicaciones nutricionales*.

Para realizar este análisis, se procedió a la instalación y uso durante unos días de cada una de las aplicaciones estudiadas en los diferentes dispositivos de los que dispone la estudiante, teniendo como resultado las siguientes conclusiones:

- ✓ El uso de una interfaz sencilla y amigable no está resuelto en las aplicaciones actuales.
- ✓ En algunas de las aplicaciones es complicado saber cuál es el foco principal de la funcionalidad que persiguen ofrecer y queda todo difuminado.
- ✓ Casi todas ellas utilizan el registro de usuarios no como un valor añadido de la aplicación sino para poder ofrecer servicios premium en caso de pagar la cuota correspondiente y tener acceso a más funcionalidades.
- ✓ La información que proporcionan sobre los productos tiene diferentes orígenes y está presentada de muy diferentes formas, con distinto contenido.
- ✓ Solo una de ellas dispone de configuración de alertas en función de los intereses del usuario y solo tiene efectos en la forma en la que presenta la información al usuario.
- ✓ Varias aplicaciones intentan fomentar su uso mediante recordatorios como notificaciones push sin aportar ningún tipo de nueva información, lo que es algo molesto para el usuario.
- ✓ Los precios de las suscripciones premium son muy diversos, así como las funcionalidades premium que cada una de las aplicaciones ofrece.

### **2.1.1.3. Método de diario**

Para llevar a cabo este método de indagación, se opta por la selección de cuatro participantes voluntarios para mantener un diario abierto en el que comentan las dificultades que se encuentran durante la realización de la compra habitual y una pequeña lista de deseos. Todos ellos son personas que o tienen algún tipo de restricción alimentaria o cuidan en un nivel elevado su alimentación, lo que hace que el tiempo que invierten al realizar la compra sea mayor que los usuarios que no tienen estas características. También hay que tener en cuenta que la reducción del tiempo que transcurre en el supermercado mientras se hace la compra es un punto muy importante para la población en general. Se ha intentado buscar una muestra de usuarios de diferentes generaciones, pero, lamentablemente, no ha sido posible del todo puesto que hay ciertas actividades que son un poco difíciles de abstraer para personas de

cierta edad que están muy acostumbradas a realizar las cosas de una determinada manera y no llegan a plantearse que ciertos cambios puedan ser posibles.

Los diarios de frases cortas recopilados se adjuntan de forma gráfica a continuación, junto con una breve descripción de cada uno de los participantes. Para la creación de estos diagramas se ha utilizado la herramienta online y gratuita Venngage (Venngage, 2020).



Figura 13: Diario de Lara

Lara es vegana y cuida mucho los ingredientes de todos los productos que consume para asegurarse de que no se trata de productos procesados o de origen animal.



Figura 14: Diario de Ethan

Ethan es celíaco desde que era un niño. Aunque en estos años las cosas han mejorado mucho, sigue invirtiendo mucho tiempo leyendo las etiquetas de sus productos para evitar tener problemas de salud.

Saray ha estado siguiendo dietas durante toda su vida, pero en los últimos tiempos está tratando de mejorar la calidad de los productos que consume, evitando en gran medida el azúcar y los procesados.



Figura 15: Diario de Saray

A Sergio le encanta cuidarse. Su regla principal es minimizar el número de ingredientes que contienen los productos que consume para así asegurarse de que son más saludables.



Figura 16: Diario de Sergio

#### 2.1.1.4. Entrevista

Para realizar la entrevista, en primer lugar, se identifican dos usuarios potenciales cuyas respuestas pueden aportar gran valor a la investigación para, posteriormente, elaborar un pequeño guion para la entrevista. Como se quiere evitar que la conversación durante la entrevista se disperse o se pierda cohesión en el diálogo, se limita el número de preguntas a dieciséis con lo que la duración total de la entrevista no excede los 20-30 minutos. Primero, se realiza una presentación para poner en contexto al entrevistado y comentarle el objetivo de la entrevista. Una vez hecho esto, se continúa con el siguiente guion:

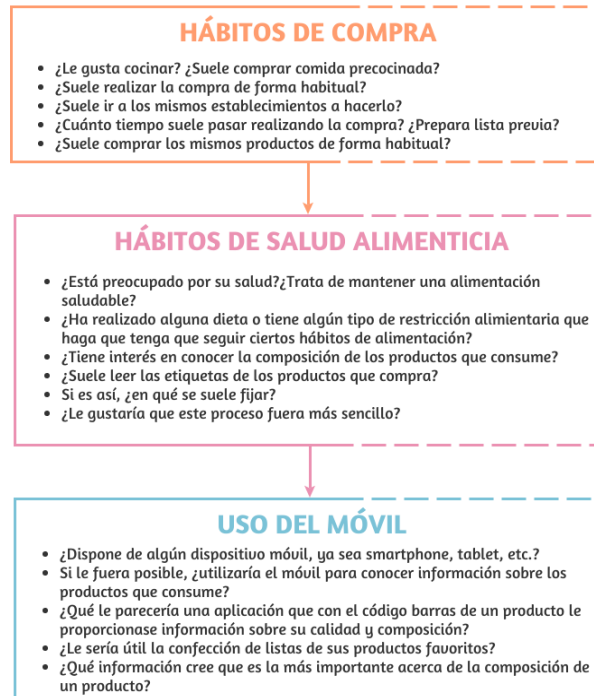


Figura 17: Guion para la entrevista

Las respuestas de los usuarios entrevistados se muestran de forma resumida y con un estilo gráfico realizado con la herramienta online y gratuita Venngage (Venngage, 2020), también utilizado para la elaboración de las preguntas.

**Rosa, 61**

Como ama de casa, destaca que es la encargada de realizar la compra de forma habitual, priorizando siempre los productos frescos para cocinarlos a diario durante la semana. Realizaba la compra siempre siguiendo la misma rutina y en los mismos establecimientos hasta que uno de sus nietos, a los que cuida de forma habitual, ha comenzado con problemas de intolerancia a la lactosa. Ahora, cuando realiza la compra tiene que ir a supermercados que no conoce tanto y siempre acude con mucho miedo a comprar el producto equivocado y que el pequeño Hugo enferme. No utiliza demasiado su teléfono móvil, aunque sus hijos le insisten mucho con ello, pero si hubiera algún mecanismo para poder hacer la compra de una forma más tranquila y segura, sin duda trataría de utilizarlo.

**Carlos, 21**

Como universitario y trabajador a tiempo parcial en una cadena de comida rápida, no está demasiado interesado en su alimentación. Sí que conoce que no la realiza de la forma más saludable que podría hacerlo pero no le gusta cocinar y, además, le parece una pérdida de tiempo. Por lo tanto, cuando realiza la compra la hace de cualquier forma y lo más rápido posible para poder llegar a tiempo a trabajar. Este modo de pensar le ha traído algún susto durante la pandemia al no poder realizar la compra al día como estaba acostumbrado y, a día de hoy, está tratando de mejorar sus hábitos investigando mucho por Internet en su tiempo libre. No cree que el escaneo de códigos de barras pueda ser demasiado útil en su día a día pero sí sacaría provecho de las listas de productos.

Figura 18: Resumen de las respuestas a la entrevista

Una vez concluido el proceso, y si bien la muestra conseguida es muy limitada, las conclusiones obtenidas se listan a continuación:

- ✓ Independientemente de la generación a la que pertenezcan los usuarios, existen motivaciones para que puedan llegar a utilizar la aplicación producto de este proyecto. Es por ello que se sienten positivos y creen que puede tener utilidad para ellos.
- ✓ El aspecto de la confección de perfiles de usuario en los que poder destacar la información más prioritaria para cada uno parece tener una gran acogida en todos los casos.
- ✓ El uso del teléfono móvil por parte de los usuarios de cierta edad es menor que en otras generaciones, pero si la interfaz es sencilla y se sienten cómodos con ella, están motivados a utilizar la aplicación.
- ✓ La información mostrada debe ser clara y entendible por cualquier usuario.

### 2.1.1.5. Encuesta

Como se ha comentado anteriormente, la encuesta, al ser un método de indagación cuantitativo y si la muestra es lo suficientemente grande, proporciona gran cantidad de datos objetivos a partir de las preguntas añadidas en su elaboración.

Para este método se elabora una encuesta en Google Forms compuesta por 21 preguntas de las que todas salvo una de respuesta libre son obligatorias (Google Forms, 2020). Este formulario se distribuye entre la gente conocida de la estudiante para tratar de llegar al máximo número de personas posible, animándolos a compartirla también a su vez.

La encuesta elaborada se encuentra accesible en el siguiente enlace:

[https://docs.google.com/forms/d/e/1FAIpQLSeB5ZQton7gfJrGIF6ChsOPXAsoXvo7aaqTlaRPIPQnLT-eS3Q/viewform?usp=sf\\_link](https://docs.google.com/forms/d/e/1FAIpQLSeB5ZQton7gfJrGIF6ChsOPXAsoXvo7aaqTlaRPIPQnLT-eS3Q/viewform?usp=sf_link)

Además, en el siguiente enlace es posible consultar la totalidad de las respuestas recibidas durante la encuesta en formato Google Docs:

<https://docs.google.com/spreadsheets/u/1/d/e/2PACX-1vSchJcIJsRyuTOP88jgk3fHkOkYpctEiY6R2N52ztD4VCewPQ3avVAUV2dYf65Cu1yLqQOaHNqWkR1/pubhtml>

El tiempo de recopilación de respuestas durante el que la encuesta está abierta es de dos semanas. A partir de ese momento, los datos obtenidos se recogen y se estudian para llegar al análisis que se detalla a continuación.

En estas dos semanas se han recopilado un total de 102 respuestas de participantes con un gran rango de diferencias en lo que a datos demográficos se refiere. Basándonos en el gráfico que se adjunta a continuación, podemos determinar dos franjas de edad diferentes: la que se encuentra entre los 18 y los 40 años y aquella que comprende los mayores de 40 años. También se traduce esto en el grado de manejo de los dispositivos móviles: normalmente a mayor edad, menos manejo o más dificultades se encuentran durante su uso.

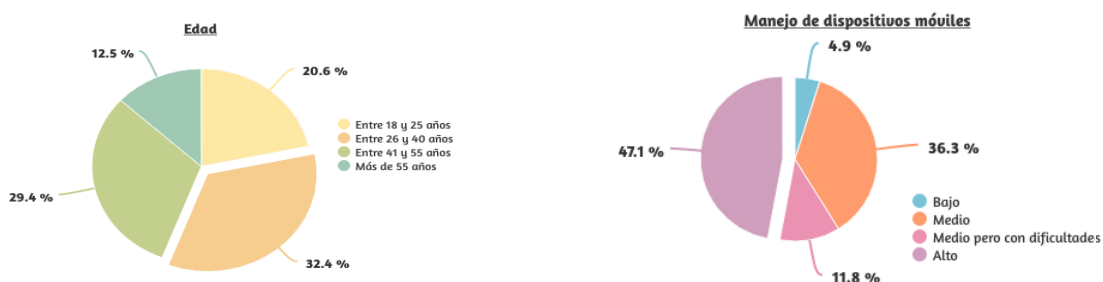
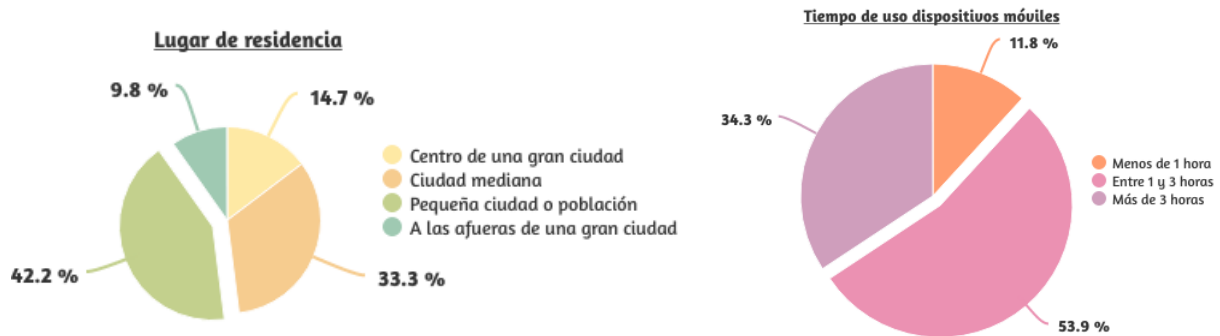


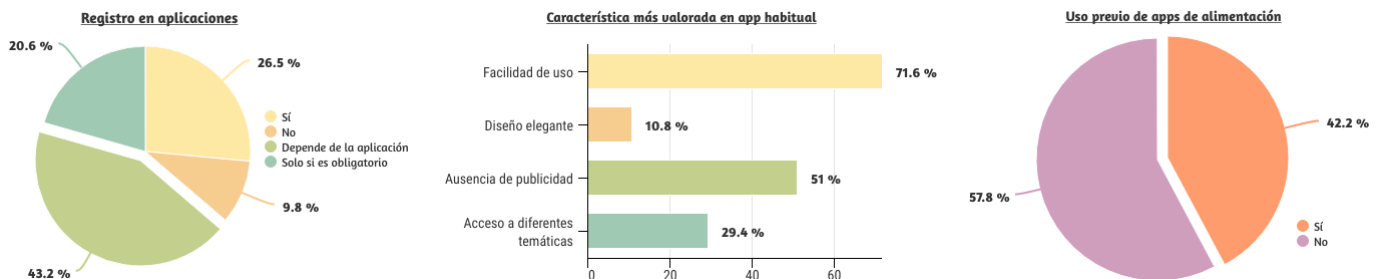
Figura 19: Encuesta - Gráficos

La mayor parte de los participantes en la encuesta viven en poblaciones pequeñas o ciudades medianas y, además, una amplia mayoría indica pasar más de tres horas de su tiempo de ocio utilizando dispositivos móviles. Revisando los porcentajes, tiene más o menos correspondencia con el perfil de edad entre los 18 y 40 años.



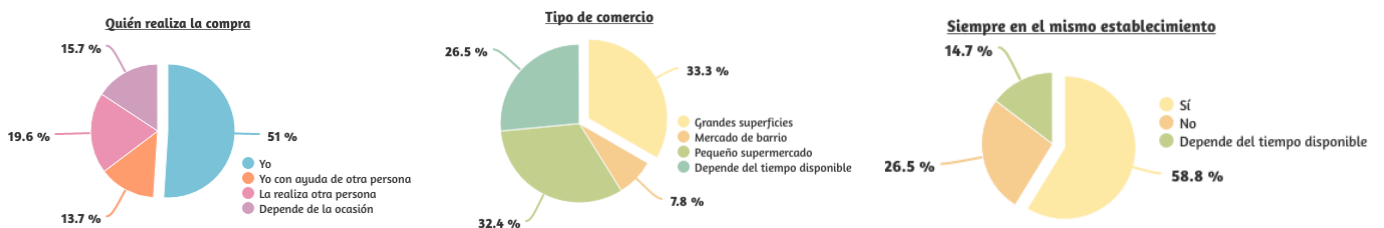
**Figura 20: Encuesta - Gráficos lugar de residencia y tiempo de uso de dispositivos rango de edad y manejo de dispositivos**

Pasando a las preguntas más relativas al uso de aplicaciones, un 42 % de los encuestados indica que el registro en ellas lo realiza en función del interés que tenga en la aplicación, frente a menos de un 10 % que niega radicalmente el registro. Entre las características que más destacan los encuestados de las aplicaciones que utilizan, se encuentra en primer lugar la facilidad de uso, seguida a bastante distancia de la ausencia de publicidad. Siguiendo en esta línea, la mayoría de los encuestados no han utilizado nunca una aplicación relacionada con la alimentación.



**Figura 21: Encuesta - Gráficos relativos a uso de otras aplicaciones**

Respecto a las preguntas acerca de los hábitos de compra, un 51 % de los encuestados son los que realizan la compra en su unidad familiar, llevando a cabo esta más o menos por igual en grandes superficies o pequeños supermercados y reflejando un cierto hábito de comprar siempre en el mismo establecimiento. En ambos casos, hay un alto porcentaje de encuestados que indica que la decisión del lugar donde realizar la compra depende en gran medida del tiempo que tengan disponible para ello.



**Figura 22: Encuesta - Gráficos relativos a hábitos de compra (I)**

Siguiendo en esta línea, la gran mayoría de los encuestados invierte entre 30 minutos y una hora en la realización de su compra habitual y un 74 % de los participantes suelen comprar siempre los mismos productos.

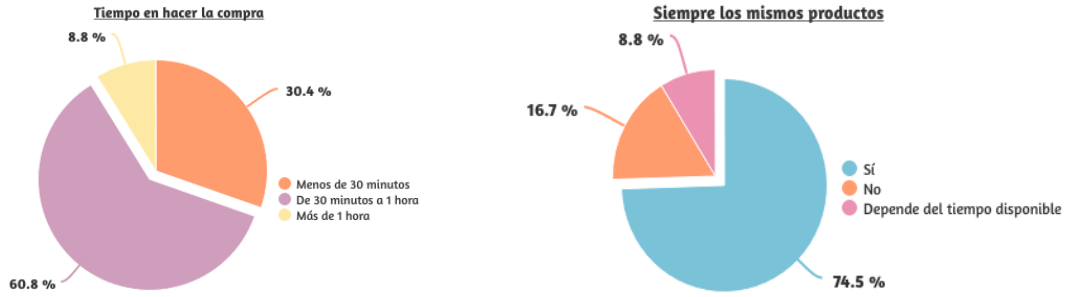


Figura 23: Encuesta - Gráficos relativos a hábitos de compra (II)

Respecto a las preguntas acerca de las intolerancias alimentarias, un 15 % de los participantes indican que padecen una o más de una restricción alimentaria. Respecto al tipo de intolerancia que padecen, la mayoría indica tener celiaquía seguida de la intolerancia a la lactosa.

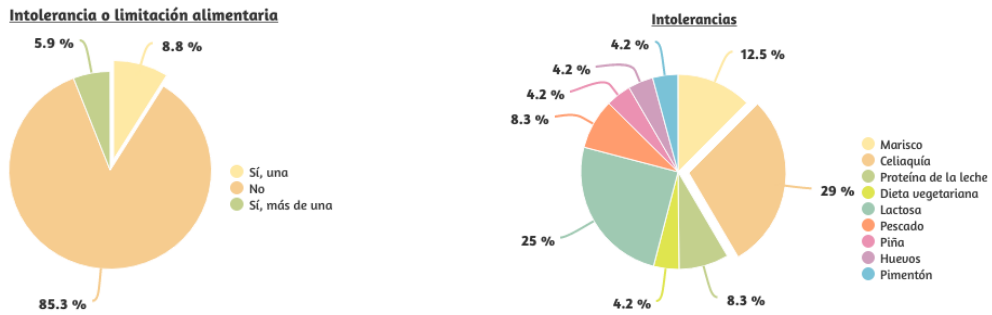


Figura 24: Encuesta - Gráficos relativos a intolerancias

En cuanto a las preguntas sobre mantener una alimentación saludable, casi un 70 % asegura tratar de hacerlo y utiliza tanto publicaciones científicas como blogs de alimentación. Del mismo modo, la información más relevante para determinar si un producto es saludable es, según los encuestados, tanto la información nutricional como el listado de ingredientes.

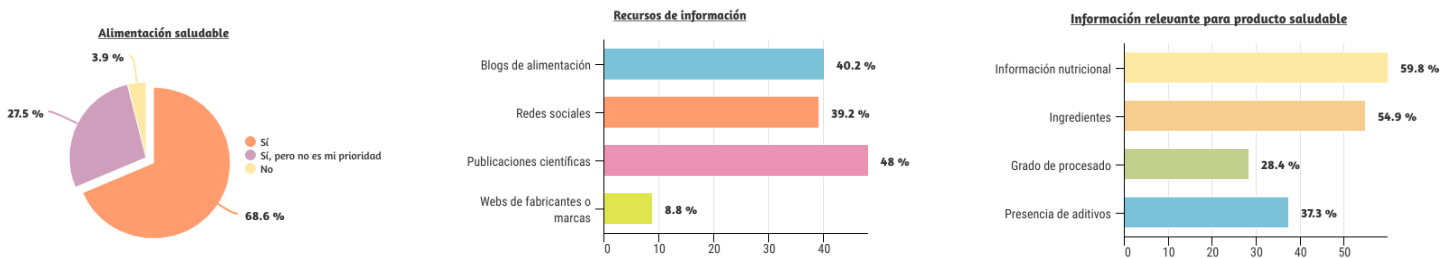


Figura 25: Encuesta - Gráficos a alimentación saludable

Pasando a las preguntas que se centran en el tema del etiquetado de los productos, un 46 % refiere leer los etiquetados indicando que un 62 % no entiende siempre la información que ahí viene indicada. Por ello, un 59 % de los participantes utilizaría otros recursos para facilitarlo.

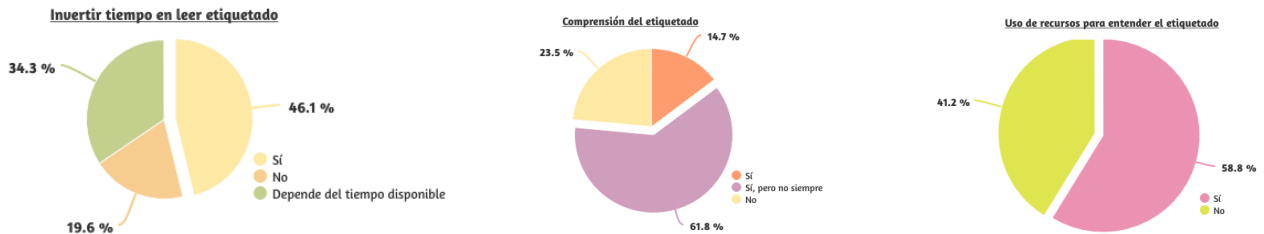
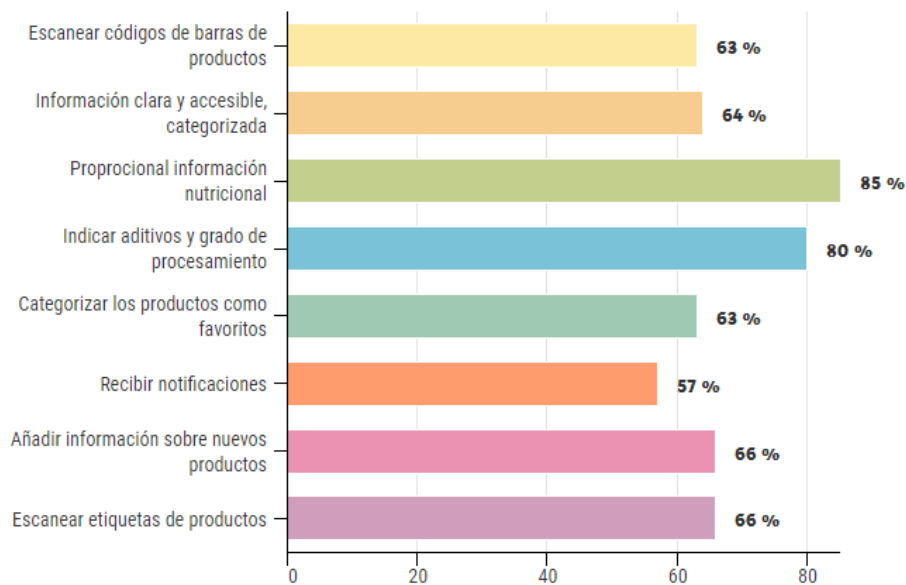


Figura 26: Encuesta - Gráficos relativos al etiquetado de los productos

Por último, estudiando los resultados de las preguntas que se centraban en las funcionalidades que se utilizarían más en una hipotética nueva aplicación, los porcentajes que reflejan una puntuación de 3 o mayor son los siguientes:





**Figura 27: Encuesta - Gráficos relativos a las funcionalidades de la nueva aplicación**

Como se puede ver en estos resultados, la funcionalidad más atractiva para los encuestados sería la de proporcionar la información nutricional seguida de la de proporcionar información sobre aditivos y grados de procesamiento de los productos y la de mostrar la información de una forma clara y accesible, categorizada por importancia. La funcionalidad que menos ha interesado es la relacionada con las notificaciones, lo que tiene sentido ya que, como usuarios de dispositivos móviles, recibimos gran cantidad de notificaciones diariamente.

Entre las conclusiones que se pueden obtener, es posible indicar que la aplicación podría tener una buena acogida, ya que las funcionalidades que ofrece son bastante aceptadas por los encuestados que, en su mayoría, además, no han utilizado antes una aplicación de este tipo, por lo que contaría con el plus de la novedad. Además, en un alto porcentaje refieren hacer la compra de una determinada forma (mismos productos, mismos establecimientos, etc.) dado que el tiempo que tienen disponible no se lo permite con lo que, si la aplicación sirve para limitar el tiempo que invierten en la realización de actividad, podrán tener más tiempo disponible para poder variar más en sus hábitos.

Los resultados de esta encuesta han servido para reafirmar los objetivos que se habían fijado inicialmente, dándole más importancia si cabe a la facilidad de uso y a la claridad en la información, aspectos referidos por una amplia mayoría de los encuestados como atractivos e interesantes en el uso de las aplicaciones en su día a día. También, del mismo modo, dentro de la claridad en la forma en la que se muestra la información habrá que incidir en la categorización de los datos que se muestran para que se vean lo más fácilmente posible y respondan a las necesidades de los usuarios.

#### **2.1.1.6. Perfilado de usuarios**

Una vez llevados a cabo los métodos de indagación indicados en los subapartados anteriores, extraídas las conclusiones y analizados los datos recopilados convenientemente, se ha llegado a la siguiente categorización de los usuarios para poder llevar a cabo el resto de iteraciones del diseño centrado en el usuario en las siguientes agrupaciones de usuarios. Para la elaboración de las infografías que muestran los detalles de las diferentes categorías de usuarios se ha utilizado la herramienta online y gratuita Venngage (Venngage, 2020).



Figura 28: Perfil de usuario descuidado



Figura 29: Perfil de usuario cuidadoso

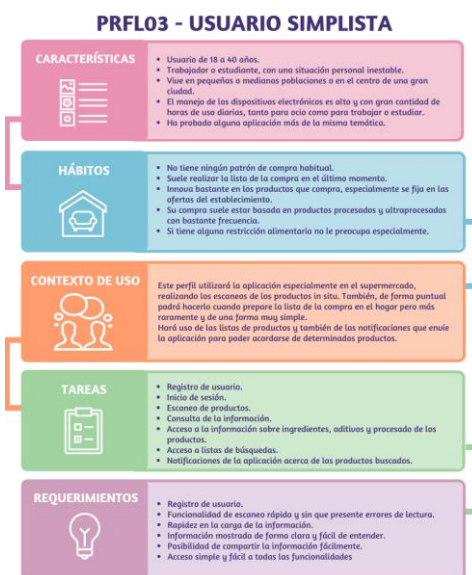


Figura 30: Perfil de usuario simplista

## 2.1.2. Diseño conceptual

Con toda la información recopilada para conocer las preferencias, necesidades y objetivos de los usuarios, es posible comenzar la fase de diseño conceptual para comenzar a esbozar de una forma más tangible los detalles que conforman el producto final. En esta fase, el primer paso consiste en la creación de unos usuarios ficticios que forman parte de alguna de las categorías de usuario creadas anteriormente y que también se sitúan en determinados escenarios de uso.

### 2.1.2.1. Personas

Para acercar la categorización en perfiles de usuario a los escenarios de uso se utiliza la técnica de las personas en las que se crean usuarios ficticios que pertenecen a dichos perfiles de usuario y que conforman descripciones de usuarios potenciales de la aplicación. De este modo, teniendo en cuenta los perfiles de usuario anteriormente definidos, a continuación, se incluye una relación de las siguientes personas ficticias creadas para este proyecto junto con su categorización dentro de uno de los perfiles definidos en el subapartado anterior.



Figura 31: Ficha de persona con perfil simplista



Figura 32: Ficha de persona con perfil cuidadoso



Figura 33: Ficha de persona con perfil descuidado

### 2.1.2.2. Escenarios de uso

En los anteriores subapartados se ha detallado la definición de los perfiles de usuario arquetipo que se identifican como potenciales para el uso de esta aplicación y que, además, hacen posible la generación de tres usuarios ficticios, cada uno formando parte de un perfil de usuario diferente, durante la aplicación de la técnica de las personas. Con toda esta información, en este apartado se explican los escenarios de uso identificados.

Un escenario de uso consiste en la descripción con detalle de una situación concreta que se produce cuando un usuario interacciona con unos objetivos determinado con la aplicación que se desarrolla en este proyecto (Garreta Domingo, Muriel y Mor Pera, Enric. 2020). De esta forma, es factible detectar necesidades que es posible que hayan pasado desapercibidas o descubrir nuevos objetivos. Además, también sirven de punto de partida para definir posteriormente los flujos de interacción de la aplicación.

EU01 – ESCANEADO DE UN NUEVO PRODUCTO EN EL HOGAR	
<b>Perfil de usuario</b>	Cuidadoso (PRFL02). Descuidado (PRFL01)
<b>Persona</b>	Robert
<b>Contexto</b>	Robert tiene a su sobrino en casa para pasar la tarde y, al prepararle la merienda, se da cuenta que cuando hizo la compra no revisó en el establecimiento que no tuviera gluten, por lo que decide utilizar la aplicación para escanear el código de barras del producto que le genera dudas antes de dárselo al chico.
<b>Objetivos</b>	Escanear el código de un producto desde el hogar.
<b>Tareas realizadas</b>	Escanear código de barras.
<b>Necesidades de información</b>	<ul style="list-style-type: none"> <li>✓ Estar registrado previamente.</li> <li>✓ Disponer del código de barras a escanear.</li> </ul>
<b>Funcionalidades</b>	Registro de usuario. Escanear código de barras.
<b>Desarrollo de tareas</b>	<ol style="list-style-type: none"> <li>1. Iniciar sesión.</li> <li>2. Ir a la sección para escanear.</li> <li>3. Escanear código de barras.</li> </ol>

Figura 34: EU01 - Escaneo de un nuevo producto en el hogar

EU02 – ESCANEADO DE UN NUEVO PRODUCTO EN EL ESTABLECIMIENTO	
<b>Perfil de usuario</b>	Simplista (PRFL03)
<b>Persona</b>	Gabriela
<b>Contexto</b>	Gabriela ha olvidado una vez más preparar la lista de la compra, por lo que se encuentra en el establecimiento improvisando lo que necesita comprar para la siguiente semana. Quiere comprar jamón york, pero nunca recuerda cuál es la marca que no tiene lactosa y no termina de aclararse con el etiquetado. Como el día anterior instaló la aplicación en su dispositivo móvil, decide escanear el producto.
<b>Objetivos</b>	Escanear el código de un producto desde el establecimiento.
<b>Tareas realizadas</b>	Escanear código de barras.
<b>Necesidades de información</b>	<ul style="list-style-type: none"> <li>✓ Estar registrado previamente.</li> <li>✓ Disponer del código de barras a escanear.</li> </ul>
<b>Funcionalidades</b>	Registro de usuario. Escanear código de barras.

<b>Desarrollo de tareas</b>	<ol style="list-style-type: none"> <li>1. Iniciar sesión.</li> <li>2. Ir a la sección para escanear.</li> <li>3. Escanear código de barras.</li> </ol>
-----------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------

Figura 35: EU02 - Escaneo de un nuevo producto en el establecimiento

<b>EU03 – ACCESO A INFORMACIÓN NUTRICIONAL DE UN PRODUCTO DESDE ESCANEO</b>	
<b>Perfil de usuario</b>	Cuidadoso (PRFL02)
<b>Persona</b>	Jorge
<b>Contexto</b>	Jorge está en casa preparando la lista de la compra y tiene dudas sobre qué zumo de naranja con lácteos comprar al día siguiente. La última vez compró dos envases diferentes y recuerda que uno le gustó más que el otro, pero tenía más azúcar. Como no se acuerda de cuál era, utiliza la aplicación para escanear su código de barras para consultar su información nutricional.
<b>Objetivos</b>	Consultar la información nutricional de un producto.
<b>Tareas realizadas</b>	Acceder a la información nutricional del producto.
<b>Necesidades de información</b>	<ul style="list-style-type: none"> <li>✓ Estar registrado previamente.</li> <li>✓ Disponer del código de barras a escanear.</li> </ul>
<b>Funcionalidades</b>	Registro de usuario. Escanear código de barras. Mostrado de información.
<b>Desarrollo de tareas</b>	<ol style="list-style-type: none"> <li>1. Iniciar sesión.</li> <li>2. Ir a la sección para escanear.</li> <li>3. Escanear código de barras para mostrar su información.</li> <li>4. Ir a la tercera pestaña de la información mostrada.</li> </ol>

Figura 36: EU03 - Acceso a información nutricional de un producto desde escaneo

<b>EU04 – ACCESO A INFORMACIÓN SOBRE ALÉRGENOS DE UN PRODUCTO DESDE BÚSQUEDA</b>	
<b>Perfil de usuario</b>	Cuidadoso (PRFL02)
<b>Persona</b>	Jorge
<b>Contexto</b>	Jorge tiene un nuevo compañero de trabajo que ha ido a su casa a ver el partido. Le ofrece una cerveza, pero su compañero le indica que es celíaco, por lo que no puede tomarla, y que además tiene intolerancia a la lactosa, por lo que tampoco tomará café. Jorge recuerda que tiene los zumos con lácteos que compró, pero no sabe si tienen lactosa por lo que abre la aplicación para consultar la búsqueda que hizo antes de comprarlos para saber si se lo puede ofrecer a su amigo.
<b>Objetivos</b>	Consultar los alérgenos que contiene un producto.
<b>Tareas realizadas</b>	Acceder a la información sobre alérgenos.
<b>Necesidades de información</b>	<ul style="list-style-type: none"> <li>✓ Estar registrado previamente.</li> <li>✓ Haber escaneado algún código de barras con anterioridad.</li> </ul>
<b>Funcionalidades</b>	Registro de usuario. Escanear código de barras. Listado de

	búsquedas. Mostrado de información.
<b>Desarrollo de tareas</b>	<ol style="list-style-type: none"> <li>1. Iniciar sesión.</li> <li>2. Ir a la sección de búsquedas.</li> <li>3. Seleccionar producto para cargar su información.</li> <li>4. Ir a la primera pestaña de la información mostrada.</li> </ol>

Figura 37: EU04 - Acceso a información sobre alérgenos de un producto desde búsqueda

<b>EU05 – ACCESO A INFORMACIÓN SOBRE ADITIVOS DE UN PRODUCTO DESDE ESCANEO</b>	
<b>Perfil de usuario</b>	Cuidadoso (PRFL02)
<b>Persona</b>	Jorge
<b>Contexto</b>	Jorge ha estado leyendo sobre los diferentes aditivos que componen la mayoría de los productos que se consumen y se alarma pensando que puede tener productos que contengan alguno de ellos en casa, así que comienza a escanear con su móvil los códigos de barras de aquellos productos que le generan más dudas.
<b>Objetivos</b>	Consultar si un determinado producto contiene algún aditivo.
<b>Tareas realizadas</b>	Acceder a la información sobre aditivos del producto.
<b>Necesidades de información</b>	<ul style="list-style-type: none"> <li>✓ Estar registrado previamente.</li> <li>✓ Disponer del código de barras a escanear.</li> </ul>
<b>Funcionalidades</b>	Registro de usuario. Escanear código de barras. Mostrado de información.
<b>Desarrollo de tareas</b>	<ol style="list-style-type: none"> <li>1. Iniciar sesión.</li> <li>2. Ir a la sección para escanear.</li> <li>3. Escanear código de barras para mostrar su información.</li> <li>4. Ir a la segunda pestaña de la información mostrada.</li> </ol>

Figura 38: EU05 - Acceso a información sobre aditivos de un producto desde escaneo

<b>EU06 – CONSULTA DE PRODUCTOS BUSCADOS ANTERIORMENTE</b>	
<b>Perfil de usuario</b>	Cuidadoso (PRFL02). Simplista (PRFL03)
<b>Persona</b>	Gabriela
<b>Contexto</b>	Gabriela necesita saber cuál es la marca que tiene los yogures sin lactosa que tanto le gustan a su pareja, que ya se ha acostumbrado también a comer sin lactosa, por lo que, antes de ir a hacer la compra, abre la aplicación en su móvil y consulta en sus búsquedas cuáles son los yogures que compró el otro día.
<b>Objetivos</b>	Consultar la lista de los productos buscados previamente.
<b>Tareas realizadas</b>	Consultar la lista de productos buscados.
<b>Necesidades de información</b>	<ul style="list-style-type: none"> <li>✓ Estar registrado previamente.</li> <li>✓ Haber escaneado algún código de barras con anterioridad.</li> </ul>
<b>Funcionalidades</b>	Registro de usuario. Escanear código de barras. Listado de búsquedas.
<b>Desarrollo de tareas</b>	<ol style="list-style-type: none"> <li>1. Iniciar sesión.</li> </ol>



2. Ir a la sección de búsquedas.

Figura 39: EU06 - Consulta de productos buscados anteriormente

EU07 – AÑADIR UN PRODUCTO AL LISTADO DE PRODUCTOS FAVORITOS DESDE ESCANEO	
<b>Perfil de usuario</b>	Cuidadoso (PRFL02)
<b>Persona</b>	Jorge
<b>Contexto</b>	Jorge está encantado con el último descubrimiento que ha hecho. Su vecina le recomendó que probase el agua de aloe vera que venden en el supermercado de abajo y le ha gustado mucho. Como no quiere olvidarse de cuál es, escanea su código de barras y añade el producto a su lista de favoritos para el próximo día que compre.
<b>Objetivos</b>	Añadir un nuevo producto al listado de productos favoritos.
<b>Tareas realizadas</b>	Añadir un producto a productos favoritos.
<b>Necesidades de información</b>	<ul style="list-style-type: none"> <li>✓ Estar registrado previamente.</li> <li>✓ Disponer del código de barras a escanear.</li> </ul>
<b>Funcionalidades</b>	Registro de usuario. Escanear código de barras. Añadir producto a favoritos.
<b>Desarrollo de tareas</b>	<ol style="list-style-type: none"> <li>1. Iniciar sesión.</li> <li>2. Ir a la sección para escanear.</li> <li>3. Escanear código de barras para mostrar su información.</li> <li>4. Añadir producto a favoritos.</li> </ol>

Figura 40: EU07 - Añadir un producto al listado de productos favoritos desde escaneo

EU08 – CONSULTA DE PRODUCTOS FAVORITOS	
<b>Perfil de usuario</b>	Cuidadoso (PRFL02). Simplista (PRFL03)
<b>Persona</b>	Gabriela
<b>Contexto</b>	El otro día Gabriela aprovechó que estuvo visitando a una amiga vegana para escanear el código de barras del pan sin lactosa que suele consumir ella y que le recomendó. En su momento lo añadió a su lista de favoritos en la aplicación y menos mal, porque ahora que está en el establecimiento comprando necesita saber cuál era así que utiliza su móvil para consultarlo.
<b>Objetivos</b>	Consultar la lista de productos favoritos añadidos previamente.
<b>Tareas realizadas</b>	Consultar la lista de productos favoritos.
<b>Necesidades de información</b>	<ul style="list-style-type: none"> <li>✓ Estar registrado previamente.</li> <li>✓ Haber escaneado algún código de barras con anterioridad.</li> <li>✓ Haber añadido algún producto a la lista de favoritos.</li> </ul>
<b>Funcionalidades</b>	Registro de usuario. Escanear código de barras. Añadir producto a favoritos. Lista de favoritos.
<b>Desarrollo de tareas</b>	<ol style="list-style-type: none"> <li>1. Iniciar sesión.</li> <li>2. Ir a la sección de favoritos.</li> </ol>

Figura 41: EU08 - Consulta de productos favoritos

EU09 – CONSULTA DE CAMBIOS SOBRE UN PRODUCTO DETERMINADO	
<b>Perfil de usuario</b>	Simplista (PRFL03)
<b>Persona</b>	Gabriela
<b>Contexto</b>	Mientras está en casa, Gabriela recibe una notificación en su móvil que viene de la aplicación indicando que la composición del pan sin lactosa que añadió el otro día a su lista ha sufrido algún cambio. Por ello, pulsa en la notificación para ver cuál es la información actualizada, esperando que no haya dejado de ser sin lactosa, porque le gusta bastante su sabor.
<b>Objetivos</b>	Consultar la información sobre un producto tras la recepción de una notificación de cambio en su composición.
<b>Tareas realizadas</b>	Consultar la información a partir de la notificación push.
<b>Necesidades de información</b>	<ul style="list-style-type: none"> <li>✓ Estar registrado previamente.</li> <li>✓ Haber escaneado algún código de barras con anterioridad.</li> <li>✓ Haber añadido algún producto a la lista de favoritos.</li> <li>✓ Haber recibido una notificación de cambio.</li> </ul>
<b>Funcionalidades</b>	Registro de usuario. Escanear código de barras. Añadir producto a favoritos. Envío de notificaciones. Mostrado de información.
<b>Desarrollo de tareas</b>	<ol style="list-style-type: none"> <li>1. Recibir notificación.</li> <li>2. Pulsar en la notificación.</li> <li>3. Iniciar sesión.</li> <li>4. Acceder a la nueva información.</li> </ol>

Figura 42: EU09 - Consulta de cambios sobre un producto determinado

EU10 – CONFIGURACIÓN DEL PERFIL DE USUARIO	
<b>Perfil de usuario</b>	Cuidadoso (PRFL02). Simplista (PRFL03). Descuidado (PRFL01)
<b>Persona</b>	Robert
<b>Contexto</b>	Tras un nuevo susto, Robert decide configurar su perfil de usuario para añadir la celiaquía de su sobrino como si fuera suya para que le sea más fácil determinar los productos que puede consumir el niño, por lo que utiliza la aplicación para hacer este cambio.
<b>Objetivos</b>	Configurar el perfil del usuario para añadir o modificar restricciones.
<b>Tareas realizadas</b>	Realizar un cambio en el perfil del usuario.
<b>Necesidades de información</b>	<ul style="list-style-type: none"> <li>✓ Estar registrado previamente.</li> </ul>
<b>Funcionalidades</b>	Registro de usuario. Gestión del perfil del usuario.
<b>Desarrollo de tareas</b>	<ol style="list-style-type: none"> <li>1. Iniciar sesión.</li> <li>2. Abrir el menú lateral y seleccionar la opción del perfil del usuario.</li> <li>3. Realizar los cambios necesarios y aceptar.</li> </ol>

Figura 43: EU10 - Configuración del perfil de usuario



### 2.1.2.3. Flujos de interacción

Como último paso del diseño conceptual y para condensar la información recopilada hasta el momento tanto en la fase de diseño orientado al usuario como en esta misma, se incluye el flujo de interacción del sistema. Un flujo de interacción consiste en la representación gráfica de los posibles caminos o decisiones que puede tomar un determinado usuario en su interacción con un sistema, en este caso con una aplicación móvil (Garreta Domingo, Muriel y Mor Pera, Enric. 2020). Como todas las técnicas que se están utilizando en esta fase de diseño, la creación de flujos de interacción puede ayudar a la detección de nuevas necesidades o requerimientos que no se hayan tenido en cuenta previamente porque se pasaron por alto en los anteriores análisis. En las siguientes líneas se adjunta el mapa de interacción realizado con la herramienta online y gratuita Excalidraw (Excalidraw, 2020).

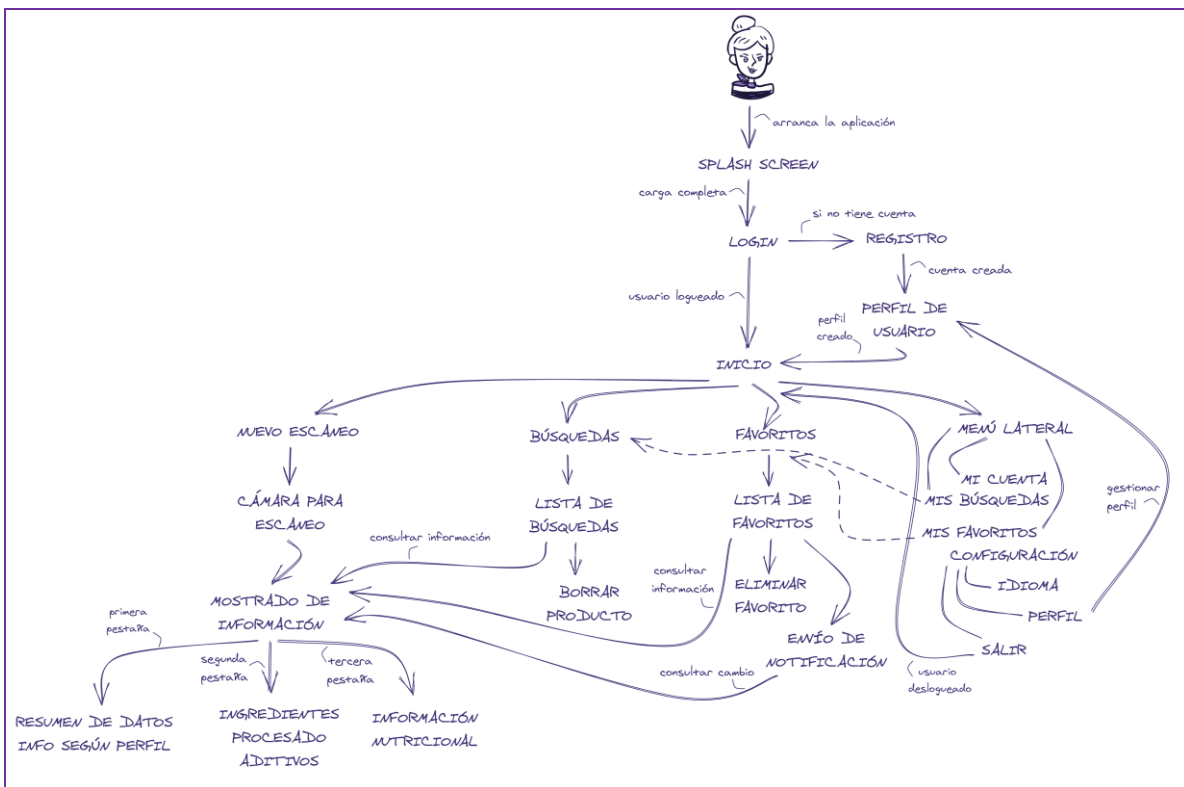


Figura 44: Flujo de interacción

### 2.1.3. Prototipado

El siguiente paso dentro del diseño centrado en el usuario es pasar toda la conceptualización anterior al prototipado. El prototipado consiste en la creación de modelos de diferente grado de fidelidad que son una representación gráfica de lo que será el diseño final de la aplicación. También se trata de un proceso iterativo.

En este caso, se comienza con la elaboración de los prototipos de baja fidelidad o sketches para, posteriormente, elaborar el prototipo horizontal de alta fidelidad que permita realizar una evaluación realista de la experiencia del usuario.

#### 2.1.3.1. Sketches

Como se ha indicado, como primer paso se realizan los sketches de la aplicación como prototipo de baja fidelidad. En este caso, aunque existen gran cantidad de herramientas que permiten su elaboración, se realizan a mano alzada. Además de mostrar todas las pantallas con las que contará la aplicación, se añade una pequeña descripción de cada una de ellas.



### 1. Pantalla de carga

Es la pantalla que se carga cuando la aplicación se está iniciando, siendo la primera pantalla que llega a ver el usuario. En ella se muestra el logotipo de la aplicación y tarda unos segundos en desaparecer de forma automática, cuando todos los recursos necesarios se han cargado en un segundo plano.

### 2. Pantalla de inicio de sesión

Una vez la aplicación ha sido totalmente cargada, la siguiente pantalla que se le muestra al usuario es la de inicio de sesión. En esta pantalla también aparece el logotipo de la aplicación y, en ella, el usuario podrá introducir sus credenciales para comenzar a utilizar la aplicación o crear unas nuevas en caso de que sea la primera vez que entra en la aplicación.



### 3. Pantalla de registro de usuario

En esta pantalla, además de mostrarse también el logotipo de la aplicación, el usuario podrá crear sus propias credenciales. Una vez el proceso se ha completado, la aplicación redirige al usuario a la pantalla de inicio de sesión para que pueda introducirlos e iniciar la sesión de forma apropiada.

### 4. Pantalla de gestión de perfiles

El usuario podrá definir las características propias de su perfil, pudiendo decidir qué alérgenos son los que quiere que la aplicación resalte en la información de los productos y también la periodicidad con la que la aplicación va a comprobar si existe algún cambio en la información de los productos favoritos, para enviarle una notificación push en caso de que esto se produzca.

A esta pantalla se llega la primera vez que se inicia sesión tras crear el usuario y también desde el menú lateral de la aplicación, a través de la opción Mi cuenta, pulsando editar alguna de las opciones disponibles.





## 5. Pantalla de bienvenida

Esta es la pantalla de inicio de la aplicación, una vez se ha iniciado sesión o creado usuario y configurado el perfil, es la que se le muestra al usuario. En ella se puede ver el logotipo de la aplicación, da acceso al menú lateral por primera vez en la navegación y, además, muestra la barra inferior de acceso rápido a las principales funciones: Escanear, Mis búsquedas y Mis favoritos.

## 6. Pantalla del menú lateral

El menú lateral está disponible desde todas las pantallas de navegación de la aplicación con el característico icono de la hamburguesa y muestra opciones para volver a la pantalla de bienvenida, ir a Mi cuenta, ver las búsquedas realizadas por el usuario, ver los favoritos guardados previamente, los ajustes disponibles y cerrar la sesión del usuario.

Este menú se abre de forma lateral de izquierda a derecha, sin ocupar la totalidad de la pantalla.



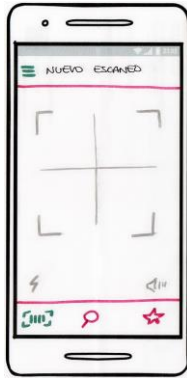
## 7. Pantalla de mi cuenta

En esta pantalla se muestra la principal información del usuario como es su nombre de usuario y el email con el que realizó el registro y luego el listado de los alérgenos configurados y la periodicidad para buscar cambios en la información de los productos. Tanto la lista de alérgenos como esta periodicidad son editables mediante el icono que aparece al lado del título de la sección y, al pulsarlo, la aplicación redirige al usuario a la pantalla de gestión de perfiles. Se accede a ella desde el menú lateral.

## 8. Pantalla de ajustes

A esta pantalla se llega desde el menú lateral y sirve para realizar cambios en el idioma en el que se ve la aplicación. En este caso, solo hay dos idiomas disponibles: español e inglés. Para facilitar la realización de esta funcionalidad de forma visual, al lado de la descripción textual de cada idioma se muestra la bandera correspondiente.





### 9. Pantalla de escanear

Esta pantalla es la pantalla desde la que se escanean los códigos de barras. En ella, el control se pasa a la cámara, que se encargará de realizar el escaneo siempre que detecte que existe un código de barras encuadrado. La captura automática tiene preferencia sobre la implementación de un botón para facilitar la captura de la información por parte del usuario.

A esta pantalla se llega desde la barra inferior de la aplicación, en el icono del código de barras.

### 10. Pantalla de mis búsquedas

Cada vez que se realice un escaneo de un producto, automáticamente aparecerá una búsqueda nueva en esta sección. Para cada producto, se mostrará su nombre o descripción corta y existe la posibilidad de mostrar su detalle, añadirlo a favoritos o eliminarlo de la lista.

A esta pantalla se llega desde la barra inferior de la aplicación, pulsando sobre el icono que representa una lupa, y desde el menú lateral, en la opción Mis búsquedas.



### 11. Pantalla de mis favoritos

En esta pantalla se muestran los productos que han sido marcados como favoritos o bien desde la ficha del detalle de un producto o desde Mis búsquedas. Los productos que se marcan como favoritos, dependiendo de la configuración del perfil del usuario, podrán ser consultados periódicamente en búsqueda de cambios en su información. En esta pantalla se permite ver el detalle de cada producto y eliminarlo de la lista de favoritos.

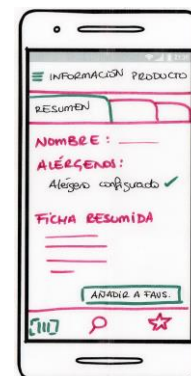
A esta pantalla se llega desde la barra inferior, pulsando sobre el icono que representa una estrella, y desde el menú lateral, en la opción Mis favoritos.



### 12. Pantalla de información de producto: pestaña Resumen

En esta pantalla se muestra la información correspondiente al producto al que corresponde un determinado código de barras. En esta primera pestaña se muestra un resumen con la información más relevante del producto, dando más prioridad a la información destacada en el perfil configurado por el usuario. También aparece un botón para añadir el producto a la lista de favoritos.

A esta pantalla se redirige automáticamente desde la pantalla de escaneo cuando se detecta un código de barras y existe la información del producto o desde las opciones Mis búsquedas y Mis favoritos, pulsando en la lupa.



### 13. Pantalla de información de producto: pestaña Ingredientes



En esta pantalla se muestra la información que corresponde al producto al que corresponde un determinado código de barras. En esta segunda pestaña se muestran los ingredientes recuperados del producto y la información relativa al grado de procesado del producto como es su categorización y el resumen de los aditivos que contiene. También aparece un botón para añadir el producto a la lista de favoritos.

A esta pantalla se llega una vez cargada la información de un producto pulsando la segunda pestaña de la pantalla que se muestra tras un nuevo escaneo o desde Mis búsquedas y Mis favoritos, pulsando en la lupa de un determinado producto.

### 14. Pantalla de información de producto: pestaña Información nutricional

En esta pantalla se muestra la información relativa al producto al que corresponde un determinado código de barras. En esta tercera pestaña se muestra todo lo relativo a la información nutricional del producto cuyo código de barras ha sido escaneado. También aparece un botón para añadir el producto a la lista de favoritos.

A esta pantalla se llega una vez cargada la información de un producto pulsando la tercera pestaña de la pantalla que se muestra tras un nuevo escaneo o desde Mis búsquedas y Mis favoritos, pulsando en la lupa de un determinado producto.



Figura 45: Recopilación de sketches de la aplicación

#### 2.1.3.2. Prototipo

En este punto, una vez creados los sketches de la aplicación para lo que se llevaron a cabo diferentes pruebas de colores, ya se dispone la información suficiente para definir los colores que se utilizarán en la aplicación. Como se ha comentado ya, la finalidad de la aplicación es, por un lado, aportar al usuario conocimiento sobre los productos que consume y, por otro, proporcionarle tranquilidad en este proceso acerca tanto de la información que recibe como de su fiabilidad. Es por este conjunto de razones por las que los colores seleccionados para la aplicación son los siguientes:

- ✓ Verde: es el color que representa la naturaleza por lo que aporta tranquilidad y una sensación refrescante. Se asigna a los botones e iconos que están activos o tienen una relevancia importante en las pantallas.
- ✓ Lila: se trata de un color que proporciona también tranquilidad y, al utilizarse con una baja gama, aporta elegancia y equilibrio y es utilizado para los fondos tanto de las partes destacadas de la aplicación como para el fondo del icono de esta.
- ✓ Morado: este color se asocia con la sabiduría y el conocimiento y es usado para darle color a la tipografía que se muestra a lo largo de la aplicación.

Teniendo esta información en cuenta se diseñan tanto el logotipo como el logo de la aplicación, utilizando esta gama y dando prioridad a los trazos curvos. En ellos se muestra el nombre de la aplicación que se elige combinando la palabra Nutri, de nutricional, con Know que significa saber en inglés y, con las mayúsculas también se juega con la palabra Now, ahora en inglés. La admiración final busca darle un toque informal al nombre y las hojas con las que se adornan darle un significado tanto de frescura y tranquilidad como representar la finalidad de la aplicación, que es ayudar a mantener una alimentación saludable. Del mismo modo, estas hojas son las que representan las notificaciones que recibe la aplicación, en color blanco por definición de la plataforma.



**Figura 46: Logotipo e icono de la aplicación**

Una vez que se eligen los colores y se dispone del diseño de tanto el logotipo como el icono de la aplicación, se procede a la creación del prototipo. Este prototipo, como ya se ha comentado, es de tipo horizontal y de alta fidelidad, lo que quiere decir que abarca la funcionalidad de toda la aplicación y que, además, su aspecto se corresponde en gran medida con el aspecto final de la aplicación. Para la realización de este prototipo se utiliza la herramienta gratuita de creación de prototipos JustInMind (JustInMind, 2020) utilizando las extensiones de Material Design (Material Design, 2020). A continuación, se adjuntan las pantallas creadas teniendo en cuenta los sketches anteriores y, además, es posible acceder al prototipo funcional, en el estado posterior a su evaluación, utilizando el enlace que genera la herramienta:

<https://www.justinmind.com/usernote/token.action?token=P2Z9C1>

En este punto se debe aclarar que el prototipo que se adjunta es una primera aproximación al diseño final, es posible que el ojo experto encuentre alguna diferencia entre este diseño y el modelado final, pero dado el carácter iterativo de esta fase, se producen continuos refinamientos y mejoras, teniendo en cuenta también las limitaciones de la plataforma en la que se desarrolla. Así, se han tomado las siguientes decisiones que discrepan de los resultados de fases anteriores:

- ✓ Pese a la reticencia demostrada por un porcentaje de los participantes en la encuesta a crearse cuentas para poder utilizar las diferentes aplicaciones, el registro de usuario se ha mantenido como obligatorio para poder ofrecer la creación de perfiles personalizados, siendo este uno de los valores añadidos que aporta la aplicación.
- ✓ Se utilizan componentes como el menú lateral y la barra inferior con la intención de ahorrar espacio y poder limitar la inclusión del botón para volver atrás al máximo posible, puesto que aporta poco valor y reduce el espacio disponible en la pantalla.
- ✓ La presentación de la información del producto que se muestra no es la final. Lo ideal sería el uso de componentes como las cards, pero en la plataforma de prototipado no permiten la libertad que luego aportan en el desarrollo final, por lo que finalmente no se han utilizado.





1. Carga



2. Inicio de sesión



3. Registro de usuario



4. Gestión de perfiles



5. Bienvenida



6. Menú lateral



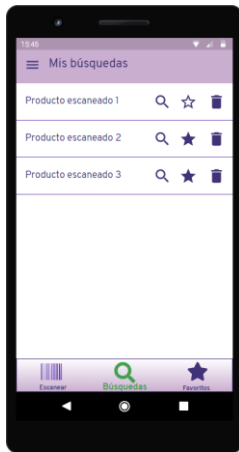
7. Mi cuenta



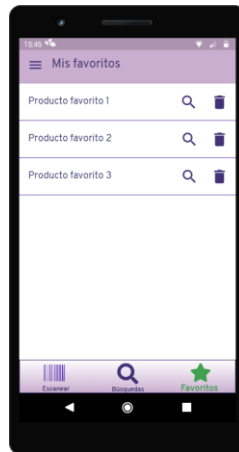
8. Ajustes



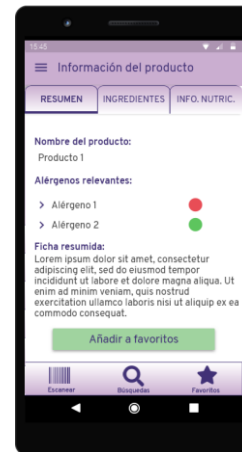
9. Nuevo escaneo



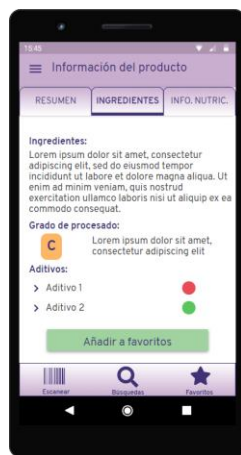
10. Mis búsquedas



11. Mis favoritos (y notificación push recibida)



12. Información del producto Pestaña 1



13. Información del producto Pestaña 2



14. Información del producto Pestaña 3

Figura 47: Pantallas del prototipo horizontal de la aplicación

#### 2.1.4. Evaluación

Como en cada una de las fases detalladas del proceso del diseño centrado en el usuario, el prototipado que se realiza debe ser evaluado de forma conveniente por diferentes usuarios para recibir sus opiniones acerca de las mejoras que podrían ser necesarias en el diseño.

Por lo tanto, el primer paso para llevar a cabo un **test de usuarios** sobre el prototipo es determinar cuáles son los aspectos de la aplicación que se evalúan (X-Wiki UOC, 2013), utilizando en este caso las reglas de Shneiderman para aplicaciones móviles:

- ✓ Coherencia y consistencia: el comportamiento de la aplicación ha de ser el mismo para las acciones que sean similares.
- ✓ Ayudas para usuarios experimentados: con el fin de incrementar la facilidad de uso, se deben añadir atajos que sean de un entendimiento común para usuarios expertos.
- ✓ Retroalimentación: por cada acción que realice, el usuario debe obtener feedback acerca de lo que ha ocurrido y de los resultados obtenidos.
- ✓ Mostrar el trabajo pendiente: se debe indicar el inicio y el final del proceso que se realiza.



- ✓ Gestión de errores: se debe mostrar la información necesaria para saber lo que ha ocurrido y ofrecer alternativas para solventar los posibles errores que se produzcan.
- ✓ Las acciones deben ser iniciadas siempre por los usuarios, sin que haya navegaciones automáticas sin explicación.
- ✓ Las interfaces han de ser claras y simples, sin abrumar al usuario con grandes cantidades de información y proporcionando facilidad de uso al sistema en general.

Una vez definidos los aspectos evaluables, el siguiente paso consiste en la elaboración de los cuestionarios que el usuario tendrá que cumplimentar para obtener información demográfica relevante sobre cada uno de ellos y, también, para conocer su uso de la tecnología, sus hábitos respecto a etiquetado de productos y alimentación saludable y lo habitual que está a utilizar aplicaciones de esta temática. Estas preguntas se presentan en un formato gráfico realizado con la herramienta online y gratuita Venngage (Venngage, 2020), ya utilizado previamente:



**Figura 48: Cuestionario pre-test de usuarios**

Tras la definición del test previo a comenzar el proceso de evaluación propiamente dicho, se deben definir el escenario que se le presentará a cada usuario y las tareas que deben realizar, que se resumen a continuación de forma gráfica con la herramienta online y gratuita Venngage (Venngage, 2020), ya utilizado previamente en este documento:

**Aran**

Aran acaba de instalarse la aplicación, apenas utiliza su dispositivo móvil y, además, no tiene demasiada práctica con las tecnologías. Por todo ello necesita tu ayuda para realizar las siguientes tareas:

- Crear un usuario nuevo e iniciar sesión.
- Configurar el perfil del usuario para añadir un nuevo alérgeno y evitar que la aplicación envíe avisos en ningún momento.
- Escanear un nuevo producto y añadirlo a favoritos tras haber revisado la información nutricional.
- Eliminar el último producto tanto de la lista de favoritos como des la lista de búsquedas.
- Acceder a la información de un producto desde la pantalla de búsquedas.

**Figura 49: Escenario y tareas test de usuarios**

También se debe elaborar un pequeño conjunto de preguntas acerca de la satisfacción con el proceso para que el usuario responda tras la realización del test. En este caso, este test se ha hecho con preguntas valorables de 0 a 5 y alguna respuesta corta, también se muestra de forma gráfica a continuación:

**DATOS DE SATISFACCIÓN**

- ¿Te ha parecido atractivo el diseño de la aplicación (colores, tipo de letra)? Valóralo de 0 a 5.
- Si la puntuación ha sido menos de 3, ¿qué cambiarías para que sea más atractivo?
- ¿Te ha parecido simple el uso de la aplicación? Valóralo de 0 a 5.
- En caso de que no te parezca simple, ¿qué mejorarías?
- Valora de 0 a 5 la utilidad de la aplicación.
- ¿De qué funcionalidad podrías prescindir?
- Valora de 0 a 5 la facilidad para encontrar las tareas dentro de la aplicación.
- Si no te ha resultado fácil, ¿qué cambiarías?
- ¿Crees que usarías la aplicación en tu vida personal?
- Indica cualquier mejora o cambio que consideres relevante.

**Figura 50: Cuestionario post test de usuarios**

En este caso, como no se van a recopilar datos personales de los usuarios ni hay datos confidenciales aportados, no es necesario la formalización de consentimientos o acuerdos de confidencialidad de ningún tipo.

Este test de usuario se lleva a cabo con tres usuarios, correspondientes a cada uno de los perfiles de usuario arquetipo que se definieron anteriormente. Uno de los usuarios trabaja en el sector de las TIC, por lo que su experiencia con el uso de aplicaciones móviles es muy extensa, otra utiliza en gran medida las redes sociales y, por último, el tercer usuario es una persona de mediana edad con un uso básico de los dispositivos móviles. En los tres casos es posible terminar todas las tareas, solo presentándose dificultades en el caso del tercer usuario, pero siendo estas relacionadas con el uso de dispositivos móviles en general.

En cada uno de los tests, como primer paso se les da una breve explicación del objetivo del test y de las funcionalidades que recoge la aplicación de la que van a probar el prototipo. También se explica qué es un prototipo en este contexto. Posteriormente, responden a las preguntas correspondientes al pre-test dando lugar a respuestas largas y a la improvisación de nuevas preguntas en el contexto del resto, en función del curso de la conversación. A continuación, se les explica los escenarios y tareas que deben desarrollar, partiendo de la aplicación en la pantalla previa al login como inicio del escenario global y, luego, realizando los pasos necesarios desde el final de la tarea anterior. Durante el desarrollo de este proceso, la estudiante va observando el curso del proceso sin intervenir salvo que las dificultades que se encuentren sean insalvables y anotando los comentarios que realicen los usuarios que se consideren relevantes. Finalmente, se realiza la batería de preguntas post-test de usuarios para que valoren cómo ha sido para ellos la experiencia.

#### **2.1.4.1. Conclusiones y mejoras sobre el prototipo**

En general, la acogida del prototipo fue bastante buena. Los comentarios acerca del diseño y de la eficacia y simplicidad de uso fueron bastante repetidos en los tres tests de usuarios que se llevaron a cabo. No obstante, tanto a partir de las observaciones de la estudiante como de los propios usuarios, se realizan diferentes mejoras en el prototipo creado para optimizarlo lo máximo posible:

- ✓ La animación que se había añadido en la pantalla de carga inicial se ha eliminado puesto que ninguno de los usuarios creía que era necesaria y, además, consideraban que desviaba la atención del usuario del propio desarrollo de la carga de la aplicación.

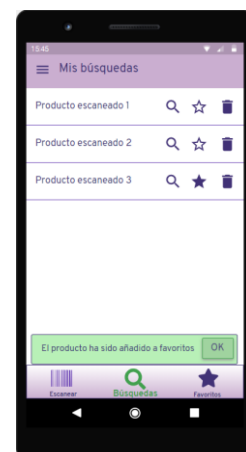
- ✓ El hecho de que la pantalla de configuración de perfil en un inicio no tuviera ni menú ni barra inferior despistó a dos de los tres usuarios que pensaron que estaban haciendo algo mal. Por ello, se ha determinado que esta pantalla no dispondrá de acceso al menú ni a la barra inferior cuando se muestre tras la creación del usuario, pero sí tendrá estos componentes cuando se trate del acceso a partir de la pantalla de Mi cuenta, para dar uniformidad a todas las pantallas a las que se accede una vez se ha iniciado sesión correctamente. Esto también hace que, en esta pantalla en particular, sea necesario el uso de un scroll vertical para acceder a toda la información que se debe mostrar.



- ✓ La disposición inicial del botón añadir producto favorito desde las pestañas de la información de producto no gustó a ninguno de los usuarios que participaron en el test y, a uno de ellos le generó dudas de su utilidad. Por ello, se ha cambiado este botón genérico por uno flotante cuyo uso está muy de moda en todas las aplicaciones modernas. Una vez añadido el producto en cuestión a la lista de favoritos, además de mostrar un mensaje indicándolo, el botón se deshabilitará.

- ✓ Dos de los usuarios echaron en falta la aparición de mensajes indicando la acción que se ha llevado a cabo en las pantallas de Mis búsquedas, Mis favoritos e Información del producto. Siguiendo esta idea, se han incorporado snackbars que informan de lo que está ocurriendo en la aplicación

- ✓ También en estas pantallas de listados, tras ver las dificultades de alguno de los usuarios para entrar a ver el detalle, se ha ampliado esta funcionalidad de estar disponible solo en la lupa para también poder hacerlo pulsando cualquier punto de la línea del producto.





- ✓ El título de la pantalla de bienvenida generaba dudas en uno de los usuarios, por poderse entender como que se está dirigiendo directamente al usuario conectado. Es por ello que se ha modificado de “Bienvenida” a simplemente “Inicio”. Además, se incorpora un icono para darle más énfasis a la idea de que se trata de la pantalla inicial de la funcionalidad de la aplicación.

- ✓ También, revisando aspectos de usabilidad, se decide en una tercera revisión cambiar el menú lateral al lado derecho para que así sea posible incluir un icono para ir atrás en aquellas pantallas que lo necesiten. Además, como se trata de la última revisión en la evaluación del prototipo, aunque cambia su disposición en casi todas las pantallas, no se sustituyen todas las capturas de pantalla mostradas en este documento, solo las dos últimas. Para consultar el aspecto final después de las revisiones iterativas es suficiente con consultar en enlace mostrado al inicio de esta sección. Con la adición del icono de ir atrás, desaparecen también los botones de Ir a inicio que aparecían en alguna de las pantallas para evitar redundancias.



**Figura 51: Recopilación de mejoras sobre el prototipo**

## 2.2. Diseño técnico

Una vez finalizada la fase del diseño orientado al usuario y cuando ya se ha extraído toda la información necesaria relativa al diseño de la interfaz y de las funcionalidades que debe contener la aplicación que se desarrolla, comienza la fase correspondiente al diseño técnico.

En esta fase el objetivo es realizar una abstracción de las funcionalidades recuperadas a partir de la fase anterior y que se bosquejaron en la fase de análisis de requisitos que resulta en la definición de los casos de uso. Además, también se realiza en esta fase la definición de la arquitectura general de la aplicación, desde su arquitectura física a su división en módulos lógicos para facilitar la implementación y a la definición de la estructura de la base de datos que la aplicación utilizará como origen de la información.

### 2.2.1. Casos de uso

A partir de los objetivos funcionales definidos en el apartado 1.2.1. *Objetivos funcionales* y una vez definidos los detalles en cuanto a diseño centrado en el usuario y usabilidad a los que están sujetos estos objetivos, es posible aumentar el nivel de abstracción y definir los casos de uso que conforman el sistema y sus funcionalidades. Estos casos de uso, además, reflejan las interacciones de los principales actores del sistema y las funcionalidades disponibles.

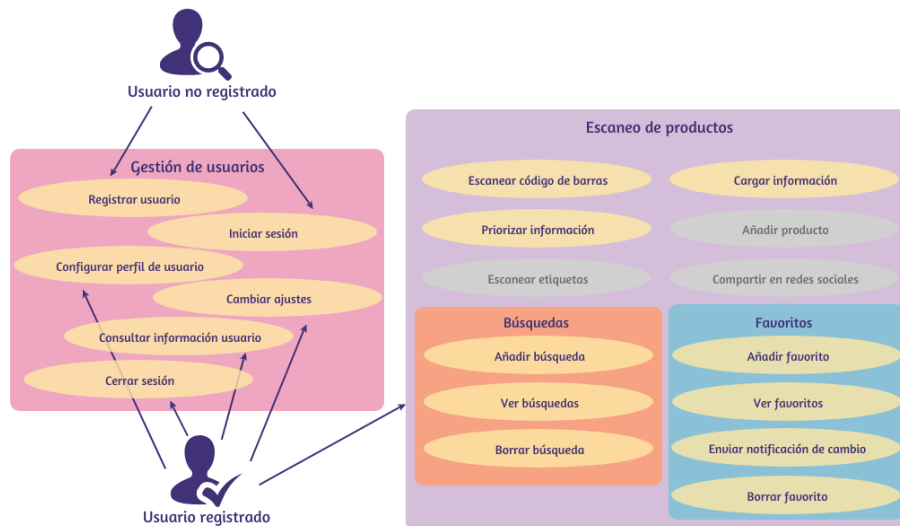
En este caso, aunque durante la fase del diseño centrado en el usuario se encontraron hasta tres perfiles arquetipo para el sistema, el nivel de abstracción que se utiliza los incluye a todos

dentro del mismo actor, que podrá ser un Usuario autenticado o un Usuario no autenticado. Como el sistema a modelar tampoco dispone de un sistema de roles que haga que los comportamientos puedan ser más específicos, encontramos que este nivel de abstracción es suficiente para este sistema.

Como primer paso, se definen los paquetes más genéricos que engloban los diferentes casos de uso y que los agrupan por similitud en las funcionalidades que describen:

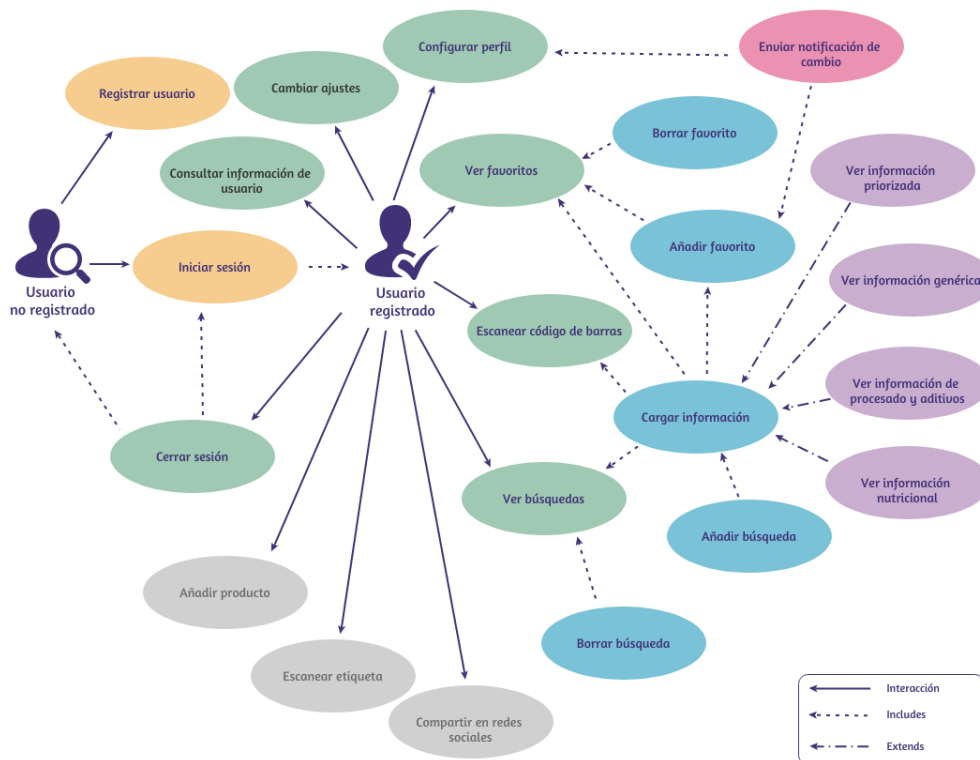
- ✓ Gestión de usuario: en este paquete se incluyen aquellos casos de uso en relación directa con el ciclo de vida del usuario y las configuraciones que le afectan.
- ✓ Escaneo de producto: en este paquete se engloban los casos de uso que se relacionan de forma estrecha con la información del producto desde el escaneo.
- ✓ Búsquedas: este paquete es un subpaquete dentro del correspondiente al escaneo que trata los aspectos referentes a las búsquedas de producto y su mantenimiento.
- ✓ Favoritos: al igual que ocurre con el paquete anterior, este es un subpaquete dentro del paquete de escaneo que permite tratar de forma más específica los productos favoritos y las funcionalidades relacionadas.

Luego, si se profundiza un nivel más en el nivel de detalle, podemos ver los casos de uso que se definen para el sistema que se modela, su clasificación dentro de cada paquete y su interacción con los dos actores principales. También se incluyen en estos casos de uso, en otro color, las funcionales que se han definido a futuro y que no se acometen durante el ciclo de este proyecto por cuestión de tiempo, solo por darle un sentido completo a lo tratado en esta memoria y ubicarlos en cuanto al paquete de casos de uso al que pertenecen, sin mostrar más información al respecto. Los diagramas que se muestran en este apartado se han realizado todos con la herramienta online y gratuita Venngage (Venngage, 2020).



**Figura 52: Diagrama de casos de uso por paquete**

Como se puede observar en el siguiente diagrama en el que ya se detallan más las interacciones, si atendemos a la relación de los actores con los casos de uso, el usuario no autenticado solo tiene acceso al caso de uso de iniciar sesión y al de registrar usuario. Una vez se ha registrado como usuario del sistema, el actor se convierte en un usuario autenticado y podrá posteriormente iniciar sesión y tener acceso al resto de funcionalidades del sistema hasta que decida cerrar sesión y volver a ser un usuario no autenticado.



**Figura 53: Diagrama de casos de uso del sistema**

Como se puede ver en el diagrama, existen relaciones de inclusión y de extensión en estos casos de uso puesto que hay funcionalidades que requieren de haber realizado antes ciertas acciones (inclusión) y otras que particularizan una funcionalidad ya realizada. Tras tener la descripción gráfica clara, se incluye a continuación la descripción detallada de cada uno de los casos de uso.

En los casos de uso que se recogen a continuación, para evitar descripciones repetitivas en el flujo de cada uno de ellos, se indica en las precondiciones los pasos previos que se deben dar antes de llegar a la pantalla en la que se desarrolla la funcionalidad que se describe.

CU01 – REGISTRAR USUARIO	
<b>Prioridad</b>	Media
<b>Propósito</b>	El usuario realiza su registro para poder iniciar sesión en la aplicación y acceder a todas las funcionalidades.
<b>Actores</b>	Usuario no autenticado.
<b>Precondiciones</b>	El usuario ha llegado a la pantalla Registro de usuario desde la pantalla de Iniciar sesión.
<b>Iniciado por</b>	Usuario no autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. En la pantalla Inicio de sesión, selecciona la opción Registro de usuario.</li> <li>2. El usuario rellena los datos de usuario y pulsa el botón. <ol style="list-style-type: none"> <li>a. Si faltan datos por rellenar, se muestra un error y no se avanza de pantalla.</li> </ol> </li> <li>3. El sistema registra al usuario. <ol style="list-style-type: none"> <li>a. Si el email ya existe en el sistema, se muestra un error y no se avanza de pantalla.</li> </ol> </li> </ol>

	4. Se redirige a la pantalla Iniciar sesión.
<b>Postcondiciones</b>	El nuevo usuario ha sido creado y puede iniciar sesión.
<b>Objetivos relacionados</b>	OF2

Figura 54: CU01 - Registrar usuario

<b>CU02 – INICIAR SESIÓN</b>	
<b>Prioridad</b>	Media
<b>Propósito</b>	El usuario inicia sesión en la aplicación para poder acceder a todas las funcionalidades que ofrece.
<b>Actores</b>	Usuario no autenticado.
<b>Precondiciones</b>	El usuario ha llegado a la pantalla Iniciar sesión.
<b>Iniciado por</b>	Usuario no autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. En la pantalla Inicio de sesión, rellena los datos de usuario y pulsa el botón. <ol style="list-style-type: none"> <li>a. Si faltan datos por rellenar, se muestra un error y no se avanza de pantalla.</li> </ol> </li> <li>2. Se redirige a la pantalla Inicio.</li> </ol>
<b>Postcondiciones</b>	El usuario puede acceder a todas las funcionalidades.
<b>Objetivos relacionados</b>	OF2

Figura 55: CU02 - Iniciar sesión

<b>CU03 – CONFIGURAR PERFIL DE USUARIO</b>	
<b>Prioridad</b>	Alta
<b>Propósito</b>	El usuario configura por primera vez las opciones disponibles en su cuenta de usuario tras el registro y primer inicio de sesión o desde el menú lateral de la aplicación.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha llegado a la pantalla Configuración de perfiles bien desde iniciar sesión por primera vez tras registrarse o bien desde la pantalla Mi cuenta o desde el menú lateral de la aplicación.
<b>Iniciado por</b>	Sistema tras registro de usuario. Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. En la pantalla Configuración de perfiles, el usuario rellena las características que quiera configurar y pulsa el botón.</li> <li>2. Se redirige a la pantalla Inicio.</li> </ol>
<b>Postcondiciones</b>	El usuario ha configurado su perfil con sus características.
<b>Objetivos relacionados</b>	OF3

Figura 56: CU03 - Configurar perfil de usuario

<b>CU04 – CONSULTAR INFORMACIÓN DE USUARIO</b>	
<b>Prioridad</b>	Baja
<b>Propósito</b>	El usuario consulta la información de su cuenta de usuario.

<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión y ha llegado a la pantalla Mi cuenta desde el menú lateral de la aplicación.
<b>Iniciado por</b>	Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla Inicio, el usuario abre el menú lateral de la aplicación y selecciona Mi cuenta.</li> <li>Se redirige a la pantalla Mi cuenta.</li> </ol>
<b>Postcondiciones</b>	El usuario accede a la información de su cuenta.
<b>Objetivos relacionados</b>	OF2

Figura 57: CU04 - Consultar información de usuario

CU05 – CAMBIAR AJUSTES	
<b>Prioridad</b>	Baja
<b>Propósito</b>	El usuario modifica los ajustes definidos para la aplicación.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión y llega a la pantalla Ajustes desde el menú lateral de la aplicación.
<b>Iniciado por</b>	Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla Inicio, el usuario abre el menú lateral de la aplicación y selecciona Ajustes.</li> <li>Se redirige a la pantalla Ajustes.</li> <li>El usuario modifica los ajustes que desee y pulsa el botón.</li> <li>Se redirige a la pantalla Inicio.</li> </ol>
<b>Postcondiciones</b>	La aplicación se muestra con los ajustes elegidos.
<b>Objetivos relacionados</b>	OF2, OF8

Figura 58: CU05 - Cambiar ajustes

CU06 – CERRAR SESIÓN	
<b>Prioridad</b>	Media
<b>Propósito</b>	El usuario cierra sesión en la aplicación y deja de tener acceso a las funcionalidades que ofrece.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión y accede a la opción Salir del menú lateral de la aplicación.
<b>Iniciado por</b>	Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>Desde la pantalla Inicio, el usuario abre el menú lateral de la aplicación y selecciona Salir.</li> <li>El sistema cierra la sesión del usuario.</li> <li>Se redirige a la pantalla Iniciar sesión.</li> </ol>
<b>Postcondiciones</b>	El usuario cierra su sesión y deja de acceder a las funcionalidades.



<b>Objetivos relacionados</b>	OF2
-------------------------------	-----

Figura 59: CU06 - Cerrar sesión

<b>CU07 – ESCANEAR CÓDIGO DE BARRAS</b>	
<b>Prioridad</b>	Alta
<b>Propósito</b>	El usuario escanea el código de barras de un producto para poder acceder a la información de dicho producto.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear.
<b>Iniciado por</b>	Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Escanear.</li> <li>2. El usuario escanea un código de barras utilizando la cámara de su dispositivo móvil. <ol style="list-style-type: none"> <li>a. En caso de que la aplicación no pueda acceder a la cámara del dispositivo, se muestra un mensaje de error.</li> </ol> </li> </ol>
<b>Postcondiciones</b>	El sistema tiene la información del código de barras escaneado.
<b>Objetivos relacionados</b>	OF1, OF4

Figura 60: CU07 - Escanear código de barras

<b>CU08 – CARGAR INFORMACIÓN DE PRODUCTO</b>	
<b>Prioridad</b>	Alta
<b>Propósito</b>	El usuario escanea el código de barras de un producto y el sistema recupera la información correspondiente. También se puede llegar a cargar la información del producto desde los listados de búsquedas y de favoritos.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear, escaneando el código de barras mediante el uso de la cámara del smartphone.
<b>Iniciado por</b>	Sistema tras escanear un código de barras.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Escanear.</li> <li>2. El usuario escanea un código de barras utilizando la cámara de su dispositivo móvil.</li> <li>3. El sistema carga la información correspondiente al producto escaneado. <ol style="list-style-type: none"> <li>a. En caso de que la información de ese producto no esté disponible, se mostrará un mensaje de error.</li> </ol> </li> </ol>
<b>Postcondiciones</b>	La información del producto representado por el código de barras escaneado previamente se carga en el sistema.

<b>Objetivos relacionados</b>	OF1, OF4, OF5
-------------------------------	---------------

**Figura 61: CU08 - Cargar información de producto**

<b>CU09 – VER INFORMACIÓN PRIORIZADA</b>	
<b>Prioridad</b>	Alta
<b>Propósito</b>	El usuario escanea el código de barras de un producto y el sistema recupera su información, mostrándola priorizando los alérgenos configurados en el perfil del usuario. También se puede llegar a ver la información del producto desde los listados de búsquedas y de favoritos.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear, escaneando el código de barras mediante el uso de la cámara del smartphone. Previamente, ha configurado en su perfil de usuario los alérgenos de su interés.
<b>Iniciado por</b>	Sistema tras escanear un código de barras. Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Escanear.</li> <li>2. El usuario escanea un código de barras utilizando la cámara de su dispositivo móvil.</li> <li>3. El sistema carga la información correspondiente al producto escaneado. <ol style="list-style-type: none"> <li>a. En caso de que la información de ese producto no esté disponible, se mostrará un mensaje de error.</li> </ol> </li> <li>4. El sistema redirige a la pantalla Información del producto.</li> </ol>
<b>Postcondiciones</b>	La información del producto se muestra en la aplicación priorizada según las preferencias configuradas por el usuario en su perfil.
<b>Objetivos relacionados</b>	OF1, OF4, OF5

**Figura 62: CU09 - Ver información priorizada**

<b>CU10 – VER INFORMACIÓN GENÉRICA</b>	
<b>Prioridad</b>	Media
<b>Propósito</b>	El usuario escanea el código de barras de un producto y el sistema recupera sus datos, mostrando la información genérica del producto. También se puede llegar a ver el contenido del producto desde los listados de búsquedas y de favoritos.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear, escaneando el código de barras mediante el uso de la cámara del smartphone.
<b>Iniciado por</b>	Sistema tras escanear un código de barras. Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Escanear.</li> </ol>

	<ol style="list-style-type: none"> <li>2. El usuario escanea un código de barras utilizando la cámara de su dispositivo móvil.</li> <li>3. El sistema carga la información correspondiente al producto escaneado. <ol style="list-style-type: none"> <li>a. En caso de que la información de ese producto no esté disponible, se mostrará un mensaje de error.</li> </ol> </li> <li>4. El sistema redirige a la pantalla Información del producto.</li> </ol>
<b>Postcondiciones</b>	La información genérica del producto se muestra en la aplicación.
<b>Objetivos relacionados</b>	OF1, OF4, OF5

**Figura 63: CU10 - Ver información genérica**

<b>CU11 – VER INFORMACIÓN SOBRE PROCESAMIENTO Y ADITIVOS</b>	
<b>Prioridad</b>	Media
<b>Propósito</b>	El usuario escanea el código de barras de un producto y el sistema recupera su información, mostrando la información relativa al grado de procesado y los aditivos que contiene. También se puede llegar a ver la información del producto desde los listados de búsquedas y de favoritos.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear, escaneando el código de barras mediante el uso de la cámara del smartphone.
<b>Iniciado por</b>	Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Escanear.</li> <li>2. El usuario escanea un código de barras utilizando la cámara de su dispositivo móvil.</li> <li>3. El sistema carga la información correspondiente al producto escaneado. <ol style="list-style-type: none"> <li>a. En caso de que la información de ese producto no esté disponible, se mostrará un mensaje de error.</li> </ol> </li> <li>4. El sistema redirige a la pantalla Información del producto.</li> <li>5. El usuario selecciona la segunda pestaña de esta pantalla.</li> </ol>
<b>Postcondiciones</b>	La información acerca del grado de procesado y los aditivos que contiene el producto se muestra en la aplicación.
<b>Objetivos relacionados</b>	OF1, OF4, OF5

**Figura 64: CU11 - Ver información sobre procesamiento y aditivos**

<b>CU12 – VER INFORMACIÓN NUTRICIONAL</b>	
<b>Prioridad</b>	Media
<b>Propósito</b>	El usuario escanea el código de barras de un producto y el sistema recupera su información, mostrando la información nutricional. También se puede llegar a ver la información del producto desde los listados de búsquedas y de favoritos.

<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear, escaneando el código de barras mediante el uso de la cámara del smartphone.
<b>Iniciado por</b>	Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Escanear.</li> <li>2. El usuario escanea un código de barras utilizando la cámara de su dispositivo móvil.</li> <li>3. El sistema carga la información correspondiente al producto escaneado. <ol style="list-style-type: none"> <li>a. En caso de que la información de ese producto no esté disponible, se mostrará un mensaje de error.</li> </ol> </li> <li>4. El sistema redirige a la pantalla Información del producto.</li> <li>5. El usuario selecciona la tercera pestaña de esta pantalla.</li> </ol>
<b>Postcondiciones</b>	La información nutricional del producto se muestra en la aplicación.
<b>Objetivos relacionados</b>	OF1, OF4, OF5

**Figura 65: CU12 - Ver información nutricional**

<b>CU13 – VER BÚSQUEDAS</b>	
<b>Prioridad</b>	Baja
<b>Propósito</b>	El usuario consulta el listado de las búsquedas que ha realizado previamente. Se puede acceder desde la opción Búsquedas o desde el menú lateral de la aplicación, seleccionando Mis búsquedas.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear, escaneando el código de barras mediante el uso de la cámara del smartphone.
<b>Iniciado por</b>	Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Búsquedas.</li> <li>2. El sistema redirige a la pantalla Búsquedas mostrando el listado disponible. <ol style="list-style-type: none"> <li>a. En caso de que el listado esté vacío, se mostrará un mensaje de error.</li> </ol> </li> </ol>
<b>Postcondiciones</b>	La aplicación muestra el listado de búsquedas.
<b>Objetivos relacionados</b>	OF1, OF6

**Figura 66: CU13 - Ver búsquedas**

<b>CU14 – AÑADIR BÚSQUEDA</b>	
<b>Prioridad</b>	Baja

<b>Propósito</b>	El usuario escanea al menos un código de barras y el sistema lo añade a su lista de búsquedas.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear, escaneando el código de barras mediante el uso de la cámara del smartphone.
<b>Iniciado por</b>	Sistema tras escanear un código de barras.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Escanear.</li> <li>2. El usuario escanea un código de barras utilizando la cámara de su dispositivo móvil.</li> <li>3. El sistema carga la información correspondiente al producto escaneado. <ol style="list-style-type: none"> <li>a. En caso de que la información de ese producto no esté disponible, se mostrará un mensaje de error.</li> </ol> </li> <li>4. El sistema añade el producto escaneado al listado de búsquedas.</li> </ol>
<b>Postcondiciones</b>	El sistema añade una nueva búsqueda a la lista.
<b>Objetivos relacionados</b>	OF1, OF6

**Figura 67: CU14 - Añadir búsqueda**

<b>CU15 – BORRAR BÚSQUEDA</b>	
<b>Prioridad</b>	Baja
<b>Propósito</b>	El usuario consulta el listado de las búsquedas que ha realizado previamente para poder eliminar alguna de la lista. Se puede acceder desde la opción Búsquedas o desde el menú lateral de la aplicación, seleccionando Mis búsquedas.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear, escaneando el código de barras mediante el uso de la cámara del smartphone.
<b>Iniciado por</b>	Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Búsquedas.</li> <li>2. El sistema redirige a la pantalla Búsquedas mostrando el listado disponible. <ol style="list-style-type: none"> <li>a. En caso de que el listado esté vacío, se mostrará un mensaje de error.</li> </ol> </li> <li>3. El usuario selecciona el registro que quiere borrar pulsando el icono que simula una papelera.</li> <li>4. El sistema borra el producto seleccionado y muestra un mensaje indicándolo.</li> </ol>
<b>Postcondiciones</b>	La aplicación muestra el listado de búsquedas con un registro menos.

<b>Objetivos relacionados</b>	OF1, OF6
-------------------------------	----------

**Figura 68: CU15 - Borrar búsqueda**

<b>CU16 – VER FAVORITOS</b>	
<b>Prioridad</b>	Alta
<b>Propósito</b>	El usuario consulta su lista de productos favoritos. Se puede acceder desde la opción Favoritos o desde el menú lateral de la aplicación, seleccionando Mis favoritos.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear, escaneando el código de barras mediante el uso de la cámara del smartphone. Luego, desde la pantalla de Mis búsquedas o desde la misma pantalla Información de producto lo ha añadido a favoritos.
<b>Iniciado por</b>	Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Favoritos.</li> <li>2. El sistema redirige a la pantalla Favoritos mostrando el listado disponible. <ol style="list-style-type: none"> <li>a. En caso de que el listado esté vacío, se mostrará un mensaje de error.</li> </ol> </li> </ol>
<b>Postcondiciones</b>	El sistema muestra la lista de productos favoritos.
<b>Objetivos relacionados</b>	OF1, OF7

**Figura 69: CU16 - Ver favoritos**

<b>CU17 – AÑADIR FAVORITO</b>	
<b>Prioridad</b>	Alta
<b>Propósito</b>	El usuario añade uno de los productos escaneados a su lista de favoritos. Se puede añadir un nuevo favorito desde la pantalla Información de producto o desde el listado de búsquedas.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear, escaneando el código de barras mediante el uso de la cámara del smartphone.
<b>Iniciado por</b>	Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Escanear.</li> <li>2. El usuario escanea un código de barras utilizando la cámara de su dispositivo móvil.</li> <li>3. El sistema carga la información correspondiente al producto escaneado. <ol style="list-style-type: none"> <li>a. En caso de que la información de ese producto no esté disponible, se mostrará un mensaje de error.</li> </ol> </li> </ol>

	<ol style="list-style-type: none"> <li>4. El sistema redirige a la pantalla Información del producto.</li> <li>5. El usuario pulsa el botón Añadir favorito.</li> <li>6. El sistema muestra un mensaje indicando que se ha añadido un nuevo producto favorito.</li> </ol>
<b>Postcondiciones</b>	La lista de productos favoritos contiene un registro más.
<b>Objetivos relacionados</b>	OF1, OF7

**Figura 70: CU17 - Añadir favorito**

<b>CU18 – BORRAR FAVORITO</b>	
<b>Prioridad</b>	Alta
<b>Propósito</b>	El usuario consulta el listado de productos favoritos que ha realizado previamente para poder eliminar alguna de la lista. Se puede acceder desde la opción Favoritos o desde el menú lateral de la aplicación, seleccionando Mis favoritos.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear, escaneando el código de barras mediante el uso de la cámara del smartphone. Luego, desde la pantalla de Mis búsquedas o desde la misma pantalla Información de producto lo ha añadido a favoritos.
<b>Iniciado por</b>	Usuario autenticado.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Favoritos.</li> <li>2. El sistema redirige a la pantalla Favoritos mostrando el listado disponible. <ol style="list-style-type: none"> <li>a. En caso de que el listado esté vacío, se mostrará un mensaje de error.</li> </ol> </li> <li>3. El usuario selecciona el registro que quiere borrar pulsando el icono que simula una papelera.</li> <li>4. El sistema borra el producto seleccionado y muestra un mensaje indicándolo.</li> </ol>
<b>Postcondiciones</b>	La lista de productos favoritos muestra un registro menos.
<b>Objetivos relacionados</b>	OF1, OF7

**Figura 71: CU18 - Borrar favorito**

<b>CU19 – ENVIAR NOTIFICACIÓN DE CAMBIO EN INFORMACIÓN DE PRODUCTO</b>	
<b>Prioridad</b>	Media
<b>Propósito</b>	El usuario recibe una notificación en su dispositivo móvil indicando un cambio en la información de uno de sus productos favoritos.
<b>Actores</b>	Usuario autenticado.
<b>Precondiciones</b>	El usuario ha iniciado sesión, ha llegado a la pantalla Inicio y ha accedido a la opción Escanear, escaneando el código de barras

	mediante el uso de la cámara del smartphone. Antes o después ha accedido a la pantalla Configuración de perfiles para activar la periodicidad en las notificaciones. Luego, desde la pantalla de Mis búsquedas o desde la misma pantalla Información de producto lo ha añadido a favoritos.
<b>Iniciado por</b>	Sistema tras añadir un nuevo producto favorito.
<b>Flujo</b>	<ol style="list-style-type: none"> <li>1. Desde la pantalla Inicio, el usuario selecciona la opción Escanear.</li> <li>2. El usuario escanea un código de barras utilizando la cámara de su dispositivo móvil.</li> <li>3. El sistema carga la información correspondiente al producto escaneado. <ol style="list-style-type: none"> <li>a. En caso de que la información de ese producto no esté disponible, se mostrará un mensaje de error.</li> </ol> </li> <li>4. El sistema redirige a la pantalla Información del producto.</li> <li>5. El usuario pulsa el botón Añadir favorito.</li> <li>6. El sistema muestra un mensaje indicando que se ha añadido un nuevo producto favorito.</li> <li>7. Cuando se cumple la periodicidad configurada por el usuario, el sistema busca cambios en la composición del producto. <ol style="list-style-type: none"> <li>a. En caso de que no se detecten cambios, el sistema no realiza ninguna acción.</li> </ol> </li> <li>8. El sistema envía una notificación al dispositivo móvil del usuario.</li> </ol>
<b>Postcondiciones</b>	El sistema envía una notificación al dispositivo del usuario indicando un cambio en la composición de uno de sus favoritos.
<b>Objetivos relacionados</b>	OF1, OF7, OF8, OF9

Figura 72: CU19 - Enviar notificación de cambio en información de producto

### 2.2.2. Diseño de la arquitectura

La arquitectura de este sistema se compone de dos partes bien diferenciadas. Por un lado, la parte frontend que se corresponde con la aplicación Android que se desarrolla y, por otro lado, la parte backend que se establece con el uso de las herramientas que ofrece Firebase (Firebase, 2020). Al utilizar Firebase, la intención es limitar el uso del almacenamiento del dispositivo y, también, simplificar el desarrollo para poder centrar esfuerzos en la implementación de la aplicación Android. Como ya se comentó anteriormente, los servicios que se van a utilizar dentro de toda la gran variedad que provee Firebase son los siguientes:

- ✓ **Authentication:** gestiona la autenticación de usuarios de forma rápida y segura.
- ✓ **RealTime Database:** provee almacenamiento y sincronización de datos en tiempo real.
- ✓ **Cloud Messaging:** permite el envío de mensajes de notificación de forma simple y segura.

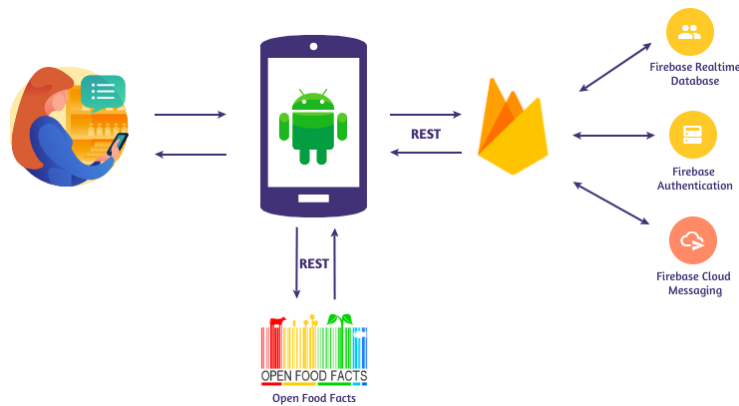
Además, se realizan invocaciones al API REST externa de Open Food Facts (Open Food Facts, 2020) para la obtención de los siguientes datos:

- ✓ Información relativa a los productos a partir de los códigos de barras escaneados.
- ✓ Información relacionada con los aditivos.
- ✓ Información respecto a alérgenos.



Por todo ello, se ha decidido que el modelo de arquitectura que sigue el sistema es el modelo-vista-controlador (MVC) que es el más usado en aplicaciones móviles y cuya característica más relevante es que conlleva la separación de la estructura de la aplicación en capas de presentación, lógica y datos, de forma que la estructura sigue el siguiente esquema:

- ✓ **Modelo:** se trata de la representación lógica de los datos necesarios para el funcionamiento de la aplicación, almacenando los datos obtenidos de las llamadas REST.
- ✓ **Vista:** es la representación visual del modelo y está formado por las interfaces XML que se desarrollan y que se traducen en las diferentes pantallas visuales de la aplicación.
- ✓ **Controlador:** se trata de la lógica encargada de la recepción y tratamiento de los eventos que llegan de las interfaces gráficas de la aplicación.



**Figura 73: Diagrama de arquitectura de la aplicación**

Como se ha indicado, en la capa de presentación se incluyen las vistas de la aplicación que se corresponden con cada una de las pantallas que se han definido en este documento en fases anteriores del diseño.

Siguiendo con esta estructura, y como ya se adelantó en el detalle de la planificación del trabajo, la capa lógica se ha estructurado siguiendo la siguiente clasificación por módulos:

- ✓ **Módulo de login:** se encarga de la gestión de usuarios.
- ✓ **Módulo de perfiles de usuario:** realiza el mantenimiento del perfil de usuario.
- ✓ **Módulo de escaneo:** su función es obtener el código de barras utilizando la cámara del dispositivo del usuario.
- ✓ **Módulo de información de productos:** su funcionalidad se dirige a la gestión de la información que se recibe de los productos y su preparación para ser mostrada.
- ✓ **Módulo de administración de búsquedas:** se encarga de mantener las búsquedas de productos que realiza el usuario.
- ✓ **Módulo de gestión de productos favoritos:** se encarga de gestionar los productos favoritos que almacena el usuario.
- ✓ **Módulo de integración:** es el módulo que se encarga de las interacciones con los sistemas externos que son invocados por la aplicación.
- ✓ **Módulo de seguridad:** es un módulo transversal para asegurar la aplicación de las medidas básicas de seguridad en el sistema.

Por último, la capa de datos se distribuye entre el acceso a los datos propiamente dichos que provienen de la base de datos y entre el acceso a los servicios que se encargan de realizar invocaciones a los servicios externos que proporcionan diferentes funcionalidades a la aplicación.

Por lo tanto, teniendo en cuenta todo lo anterior, el esquema general de la arquitectura de la aplicación quedaría como se muestra en el gráfico que se adjunta a continuación y que ha sido también realizado con la herramienta online y gratuita Venngage (Venngage, 2020):



Figura 74: Diagrama de arquitectura por capas

Una vez definida la arquitectura de la aplicación, es posible realizar un diagrama de clases entidad-relación de alto nivel que represente las diferentes entidades que conforman el sistema que se desarrolla y sus relaciones. Para la realización de este diagrama se ha utilizado la herramienta MagicDraw (MagicDraw, 2020):

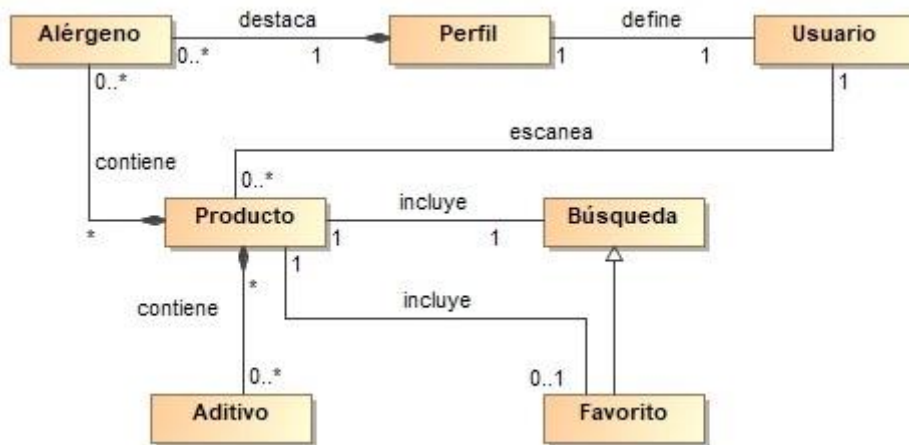


Figura 75: Diagrama de clases entidad-relación

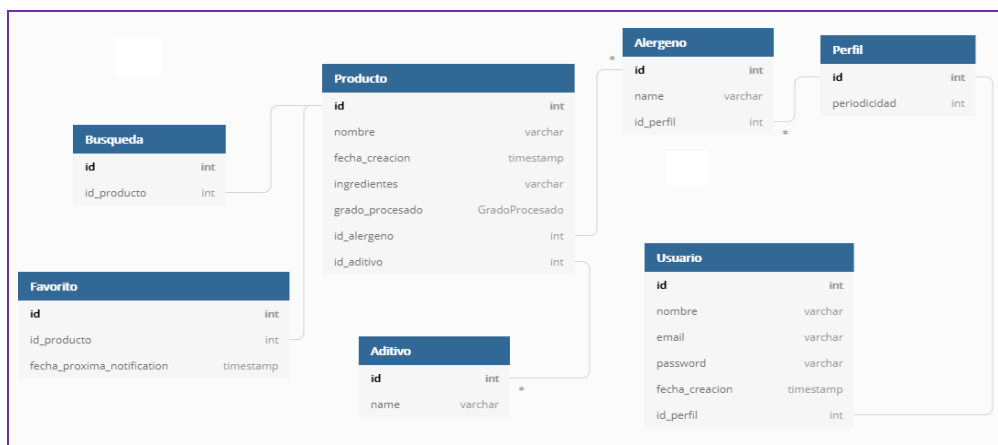
Entrando en el detalle del diagrama, se aporta una breve descripción de cada entidad, para una mejor comprensión del diagrama de base de datos que se muestra más adelante:

- ✓ **Usuario:** es la entidad que se encarga de almacenar la información del usuario.
- ✓ **Perfil:** se trata de la entidad que guarda la información del perfil creado, relacionando al usuario con la configuración definida y con los alérgenos a destacar.

- ✓ **Alérgeno:** es la entidad que modela cada uno de los alérgenos que se pueden encontrar en los productos y que también se pueden asignar a un determinado perfil.
- ✓ **Producto:** se trata de la entidad que modela la información del producto, almacenando un subconjunto de los datos recuperados. También está relacionada con el usuario.
- ✓ **Aditivo:** esta entidad representa cada uno de los aditivos que puede contener un determinado producto.
- ✓ **Búsqueda:** es la entidad que modela cada uno de los registros que se generan al escanear un producto.
- ✓ **Favorito:** es la entidad que almacena un determinado producto con la categorización de favorito que permite que la búsqueda de información con cierta periodicidad.

Tal y como se ve en el diagrama, un usuario define un único perfil que puede destacar ninguno o muchos alérgenos, en relación de composición. Además, un usuario puede escanear ninguno o muchos productos que a su vez puede contener o ninguno o muchos alérgenos o aditivos, también en relación de composición. Cada producto escaneado se incluye como búsqueda y puede o no incluirse también como favorito. También, la entidad llamada favorito hereda de búsqueda ya que tiene la misma estructura y, adicionalmente, permite saber si ha sido revisado en busca de posibles cambios en su composición.

Como se ha comentado, para el almacenamiento de datos se utilizará el servicio que provee Firebase Realtime Database (Firebase, 2020), que consiste en una base de datos con formato NoSql que usa una estructura de nodos JSON pero, aun así, dada la estructura jerárquica que sigue el modelo de datos que se define para esta aplicación, se realiza el diagrama de base de datos. Así, se adjunta el siguiente diagrama de base de datos realizado con la herramienta online y gratuita DbDiagram.io (dbdiagram.io, 2020) que contiene una abstracción con la estructura de las principales tablas creadas para el desarrollo y funcionamiento de la aplicación:



**Figura 76: Diagrama de base de datos**

En este diagrama se han incluido las tablas más importantes que tienen relevancia en el sistema, los atributos principales señalando de estos las claves primarias resaltadas en negra, las claves foráneas con la relación y un campo auxiliar para detallar en cada tabla cuáles son los que conforman su relación, sin que se traduzca necesariamente en campos que aparezcan en la tabla final de la base de datos, y la multiplicidad de estas relaciones representada con un asterisco (\*) en el lado que la relación sea múltiple y sin indicar nada cuando es relación de multiplicidad única. También hay que dejar claro en este punto que solo se almacena la información relevante en cuanto a la comprobación de cambios en momentos posteriores del ciclo de la aplicación, en ningún caso se guarda la totalidad de la información que se recupera.

Además, por los motivos que se detallan más adelante al analizar en profundidad el API de Open Food Facts (Open Food Facts, 2020), se ha decidido incorporar una base de datos interna en el dispositivo del usuario para almacenar la información correspondiente a los catálogos del API y minimizar el número de llamadas REST realizadas para el funcionamiento de la aplicación. La estructura final de esta base de datos se adjunta a continuación:

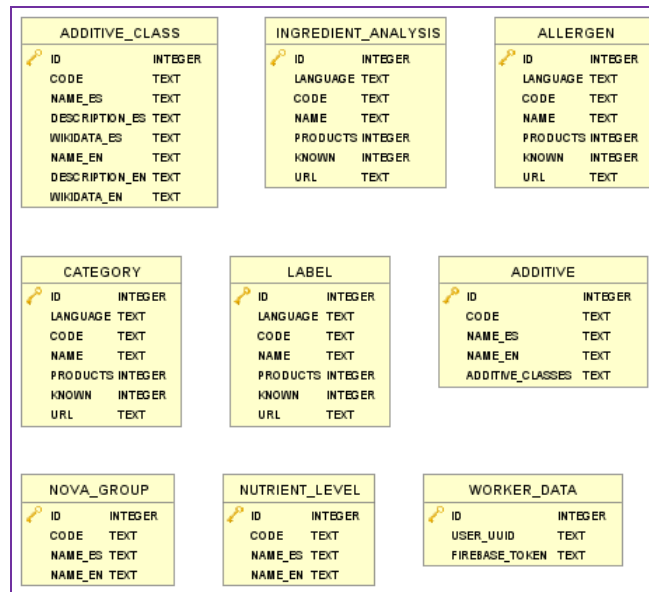


Figura 77: Diagrama de la bases de datos interna del dispositivo

Como se puede observar, se utilizan tablas simples, sin relacionar entre ellas y con un número de atributos y descripciones que trata de emular la información básica que devuelve el API para, si es necesario en un futuro, poder aumentar la información que se utiliza de cada uno de los catálogos.

### 3. Implementación

Una vez definidos tanto los objetivos del sistema como todo el conjunto de especificaciones de diseño, tanto centrado en el usuario como en técnico y de arquitectura, se realiza la implementación de la aplicación propiamente dicha. Para ello, se toman como base normas básicas de diseño como es asegurar el máximo desacoplamiento y reutilización de componentes, así como la utilización de estándares tanto de desarrollo como en cuanto a las herramientas y APIs y herramientas externas utilizadas. Además, también se ha tratado de orientar el desarrollo a la implementación de tareas asíncronas en lo posible, para no empeorar la experiencia de usuario. A continuación se detallan tanto las herramientas utilizadas como los aspectos más técnicos relativos al desarrollo de la aplicación y sus funcionalidades implementadas.

#### 3.1. Herramientas y APIs utilizadas

Durante el desarrollo se ha hecho uso de una gran variedad de herramientas, librerías y APIs de desarrollo para facilitar el desarrollo y acortar tiempos de implementación, al no tener que desarrollar funcionalidades ya existentes como propias:

##### 3.1.1. Herramientas para el desarrollo

- ✓ **Android SDK:** es el kit de desarrollo oficial de Android y su descarga viene integrada con la del IDE Android Studio. A la hora de elegir la versión del SDK para que la aplicación fuera compatible, se ha optado por establecer como versión mínima la versión 23 (Android

Marshmallow 6.0) y la versión de compilación en la versión 29 (Android 10). Con esto, tal como se puede observar en la siguiente imagen (Xakata, 2020), se asegura una cuota de dispositivos Android de alrededor del 85 %:

VERSIÓN DE ANDROID	PORCENTAJE
ICE CREAM SANDWICH (4.0)	0,2%
JELLY BEAN (4.1 - 4.3)	1,7%
KITKAT (4.4)	4%
LOLLIPOP (5.0 - 5.1)	9,2%
MARSHMALLOW (6.0)	11,2%
NOUGAT (7.0 - 7.1)	12,9%
OREO (8.0 - 8.1)	21,3%
PIE (9.0)	31,3%
ANDROID 10 (10.0)	8,2%

**Figura 78: Distribución de la cuota de dispositivos Android**

- ✓ **Android Studio:** se trata del IDE oficial de desarrollo para Android (Android Developers, 2020). Se caracteriza por su facilidad de uso y por la gran cantidad de documentación que hay disponible en relación a su utilización. Además, como ya se ha comentado, permite la descarga de diferentes versiones del SDK de Android para realizar el desarrollo.
- ✓ **Java:** es el lenguaje elegido para el desarrollo para Android. Existen dos posibilidades: Java, un lenguaje orientado a objetos y de los más utilizados actualmente, o Kotlin, un lenguaje de tipado estático. Se ha optado por el desarrollo en Java puesto que se trata de un lenguaje ya conocido previamente y así la deuda técnica existente se limita a la utilización de la plataforma Android.
- ✓ **Git y Bitbucket:** se trata de un sistema de versionado gratuito que implementa un control de cambios muy útil cuando se trata de proyectos de desarrollo que constan de una gran cantidad de código (Git, 2020). Se ha creado un repositorio Git alojado en Bitbucket (Bitbucket, 2020) para prevenir la pérdida del desarrollo realizado por causas ajenas a la estudiante, como puede ser un fallo en el equipo en el que se realiza la implementación o algún error de desarrollo que comprometa el desarrollo anterior. Además, en caso de ser necesario, esto también permite compartir el código realizado de una forma más directa.

### 3.1.2. Librerías

- ✓ **Firestore:** tal como se ha comentado, se utiliza esta herramienta (Firestore, 2020) para proporcionar funcionalidades a la aplicación y disminuir el tiempo y recursos invertidos de otro modo en su desarrollo. Es por ello que se utilizan Firestore Real Database para la gestión de la base de datos, Firestore Authentication para la creación de usuarios y acceso a la aplicación y Firestore Cloud Messaging para el envío de notificaciones push.
- ✓ **SQLite:** se trata de una librería basada en C que proporciona un pequeño y autocontenido motor de base de datos local para dispositivos móviles (SQLite.org, 2020). La motivación del uso de esta librería reside en la estructura en la que el API de producto devuelve los datos ya que utiliza códigos para la información relevante que deben cruzarse con su traducción en los catálogos correspondientes información que se almacena en una base de datos local de SQLite para evitar múltiples llamadas reiteradas al API de producto.

- ✓ **Retrofit:** es una librería que provee un cliente HTTP seguro tanto para Java como para Android para realizar llamadas REST a APIs externas que convierte dichas APIs en interfaces Java (Retrofit, 2020). Dada su facilidad de uso, se ha elegido para realizar todas las llamadas necesarias al API de Open Food Facts.
- ✓ **Material Design:** es un sistema de diseño open source que permite a los desarrolladores implementar soluciones digitales de gran calidad visual (Material Design, 2020). Proporciona tanto recomendaciones de desarrollo como estándares a seguir y componentes listos para usar. En este caso, se han utilizado los componentes que proporciona para formularios (botones, cuadros de texto), para las barras de navegación y los estilos que permite que sean aplicados a otro tipo de componentes. También se ha utilizado para obtener alguno de los iconos mostrados en la aplicación.
- ✓ **Android Iconify y FontAwesome:** se trata de una librería no oficial pero muy extendida, que permite la utilización de iconos vector entre los que elegir para incorporar a las aplicaciones Android desarrolladas, así como un modo fácil e intuitivo de personalizarlos (JoanZapata/android-iconify, 2020). Se ha utilizado concretamente para la utilización de iconos de la librería FontAwesome (Font Awesome, 2020).

### 3.1.3. Herramientas de ayuda al desarrollo

- ✓ **DbVisualizer:** es un visor de bases de datos que permite ver el contenido de las bases de datos Sqlite, entre otros tipos, y también extraer un diagrama de su estructura (DbVisualizer, 2020). Es la herramienta utilizada para obtener el diagrama de la Figura 79.
- ✓ **SoapUI:** se trata de una herramienta que proporciona un modo sencillo de testear APIs REST para poner conocer su funcionamiento y requerimientos, así como estudiar la estructura y contenido de las respuestas que proporciona (SoapUI, 2020). Se ha utilizado para testear el funcionamiento del API de Open Food Facts.
- ✓ **Code Beautify:** es una herramienta online que proporciona una gran variedad de funcionalidades para convertir de un tipo de lenguaje a otro (Codebeautify, 2020). En este caso, la funcionalidad utilizada ha sido el conversor de JSON a Java para tener una primera orientación de la estructura de los modelos Java a los que traducir las estructuras JSON de las respuestas del API de producto.
- ✓ **Open Food Facts API Documentation:** es la web en la que el proyecto open source que proporciona la información de producto tiene documentados todos los servicios y respuestas (Open Food Facts API Documentation, 2020). Ha sido una documentación clave para desarrollar con garantías la recuperación de la información y su presentación.
- ✓ **Easy App Icon:** es la herramienta online que se ha utilizado para la generación de los iconos de la aplicación en las diferentes formas y dimensiones necesarias para cada uno de los tamaños de dispositivo disponibles (EasyAppIcon, 2020).
- ✓ **Android Asset Studio:** se trata de la web a la que se ha dado uso para la creación de los iconos para mostrar cuando el dispositivo reciba una notificación push (Android Asset Studio, 2020). La imagen que se utilice para mostrar en la notificación debe tener fondo transparente y ser de color completamente blanco.
- ✓ **Flaticon:** es la web que se ha utilizado como fuente para la mayoría de los iconos que se muestran en la aplicación (Flaticon, 2020). Los iconos que esta web proporciona son en su gran mayoría de licencia gratis o están bajo una licencia con la que la simple acreditación es suficiente. Para la utilización de tanto estos iconos como los de Material Design se utilizó el generador de Vector Assets que tiene integrado el IDE Android Studio.

## 3.2. Desarrollo de la aplicación

En este apartado y sus subsecciones se detallan los aspectos más relevantes relacionados con el desarrollo de la aplicación y que van desde explicar la estructura elegida para la disposición interna del código del proyecto como a especificar el funcionamiento de diferentes bloques funcionales pasando por el detalle de los componentes principales entre los usados para construir la interfaz gráfica de acuerdo al prototipo realizado en fases anteriores.

### 3.2.1. Estructura del proyecto

Para el desarrollo de la aplicación se ha utilizado la estructura por defecto que provee Android Studio en la creación de un proyecto Android y se ha personalizado para el uso concreto de esta aplicación una estructura más clara mediante el uso de paquetes más específicos, siguiendo tanto el árbol de navegación definido como los casos de uso del sistema.

Siguiendo el patrón Modelo-Vista-Controlador que se seleccionó como el más adecuado en este caso, se ha estructurado el proyecto de la siguiente forma, teniendo las clases java que dotan de funcionalidad a las vistas recopiladas en el directorio /java, mientras que todos aquellos recursos gráficos se encuentran localizados en la carpeta /res:

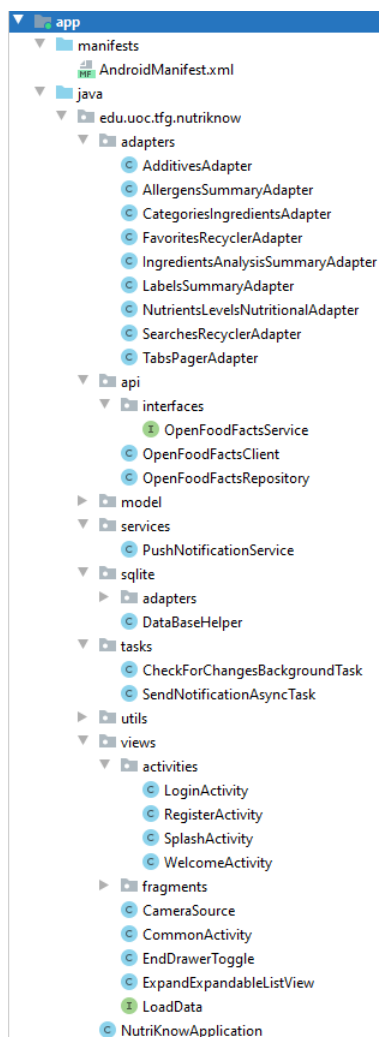


Figura 79: Estructura del proyecto - Clases java

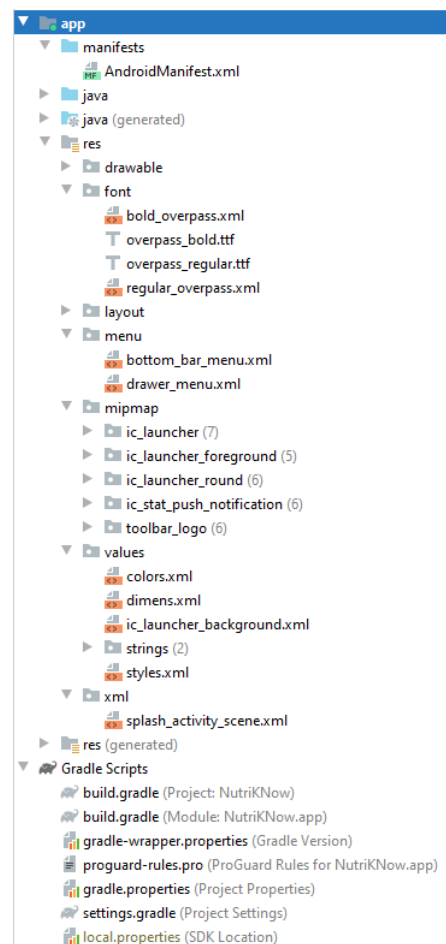
- ✓ /manifests: es el directorio que contiene el xml que define los activities que contiene la aplicación, cuál actúa de principal, los servicios que utilizan y, además, los permisos y features que el sistema necesita.
- ✓ /adapters: aquí se ubican las clases que interactúan entre las vistas y los modelos para utilizar los datos que estos últimos contienen convertidos a un formato que las vistas puedan utilizar para mostrarlos.
- ✓ /api: en esta carpeta se han recogido aquellas clases que tienen relación con el API del producto. En este caso se trata de la interfaz Retrofit para la comunicación, de la clase cliente y del repositorio que se encarga de recuperar la información de los catálogos.
- ✓ /model: aquí se recogen todas aquellas clases que actúan como modelos o entidades de la aplicación. Entre ellas están aquellas que almacenan en memoria la información del usuario relativa a su perfil o sus productos, búsquedas y favoritos y también las que se utilizan para recoger la información del API de producto.
- ✓ /services: aquí se localizan aquellas clases que actúan como servicios, siendo en este caso la clase que escucha continuamente si se ha recibido una notificación push en foreground para reenviarla y que el usuario la pueda ver e interactuar con ella.
- ✓ /sqlite: esta carpeta contiene todas las clases relacionadas con la base de datos interna del dispositivo, incluidos los adaptadores de cada tabla.
- ✓ /tasks: aquí se ubican aquellas clases que se ejecutan en segundo plano para no cargar el sistema demasiado. En este caso, son las clases que envían notificaciones push y que se

encargan de realizar la búsqueda de cambios en los productos favoritos. Pese a que el repositorio que recupera los catálogos también se ejecuta en segundo plano, no se ha incluido en este directorio por ser más clara su relación con el API de producto.

- ✓ /utils: en este directorio se han incluido todas aquellas clases con métodos estáticos que se utilizan en diferentes puntos del sistema y que se han creado para evitar la repetición de código.
- ✓ /views: en esta carpeta se ubican todas las clases en relación directa con las vistas de la aplicación. El sistema solo tiene cuatro actividades diferenciadas en el directorio /activities, que heredan todas ellas de CommonActivity para agrupar comportamientos comunes. El resto de las pantallas, al estar contenidas dentro de una de estas actividades (WelcomeActivity), se ha implementado en fragments autocontenidos que se localizan en el subdirectorio /fragments. En esta carpeta también se incluye la interfaz utilizada para que los fragments recuperen datos de forma customizada y la clase Application para saber si la aplicación está en background en todo momento. Además, también se han incluido aquellas clases necesarias para el funcionamiento de diferentes componentes como el menú lateral, la lista expandible o el acceso a la cámara del dispositivo.

A continuación, se dará un conjunto de detalles generales de la organización de los directorios relativos a las vistas:

- ✓ /drawable: en este directorio aparecen todos aquellos archivos utilizados en las diferentes plantillas visuales como pueden ser los gradientes, selectores, iconos en formato vector, etc.
- ✓ /font: aquí se localiza las fuentes personalizadas utilizada en los estilos del sistema.
- ✓ /layout: en este directorio se ubican todos los xml que conforman las vistas de la aplicación desde las actividades a los fragments, pasando por los layouts propios de la toolbar, las cards o las filas de las diferentes listas que se muestran en las diferentes secciones.
- ✓ /menu: es el espacio dedicado para los layouts que definen la estructura del menú lateral y de la barra inferior que se visualizan en el sistema.
- ✓ /mipmap: dentro de este conjunto de carpetas se encuentran los recursos gráficos necesarios para la aplicación ordenados por sus dimensiones dependiendo del tamaño del dispositivo utilizado.
- ✓ /values: aquí se recogen todos aquellos xml que definen parámetros generales de la aplicación como son los literales, los estilos o los colores.
- ✓ /xml: en este caso, este directorio contiene la animación que se ha utilizado en la pantalla de inicio de la aplicación (splash screen).



**Figura 80: Estructura del proyecto - Interfaz gráfica**

Además de los directorios indicados, podemos encontrar un apartado para los scripts de gradle que recogen todos aquellos detalles específicos de la aplicación como son las dependencias



que utilizan o los módulos que importan y el repositorio en el que encontrarlos, las versiones del sdk y la versión de Java a utilizar en el proyecto.

### 3.2.2. Gestión de base de datos

Como se ha comentado anteriormente, para la gestión de la base de datos se ha decidido utilizar Firebase Realtime Database (Firebase, 2020). El primer paso para la utilización de esta funcionalidad es la creación del proyecto de la aplicación en Firebase, la selección de un nombre de base de datos, la configuración de reglas de acceso y la descarga de la información identificativa. Una vez incorporado este fichero al proyecto y añadido el módulo a utilizar en Gradle, ya se puede comenzar a utilizar este gestor de base de datos.

Firebase Realtime Database genera y gestiona bases de datos de tipo NoSQL y esto hace que haya parte de la información que no se pueda trasladar a un diagrama de base de datos jerárquico, al menos de un modo realmente representativo. Por eso en el apartado de *Diseño de la arquitectura* se resumió esta representación a las principales tablas. A continuación se aporta un ejemplo del árbol que resulta de la configuración de un usuario, el escaneo de un producto y la adición de este producto a la lista de favoritos.



Figura 81: Árbol de base de datos con usuario, perfil, búsqueda y favorito



Figura 82: Árbol de base de datos con detalle de producto

Como se puede observar en los árboles proporcionados, cada usuario tiene relacionado un perfil que determina los alérgenos sobre los que tiene interés, la periodicidad para las notificaciones y su fecha de creación. Luego, cuando el usuario escanea un producto, se genera automáticamente un nodo de búsqueda y, en caso de añadir el producto a favoritos, otro de producto favorito. Ambos nodos tienen prácticamente la misma información, pero el nodo de

favoritos tiene además la fecha de la próxima búsqueda de cambios, calculada en base a la periodicidad definida por el usuario en su perfil.

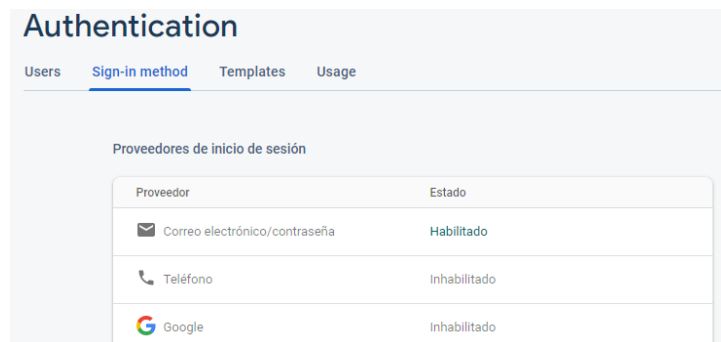
En cuanto al nodo de producto, se puede ver que se guarda un subconjunto de la información que se recupera del API de producto (Open Food Facts, 2020). Esta información se ha organizado en base a su temática y a su posicionamiento en la vista de la aplicación. Se almacena información relativa a los alérgenos y trazas declaradas para el producto, tanto en relación con los configurados como los contenidos en general por los productos, el análisis de ingredientes relativo a si contiene aceite de palma o es vegetariano o vegano, la información nutricional de los ingredientes, el grado de procesado y el contenido en sal o grasas.

Tal como se ha indicado, la información almacenada es solo un subconjunto de la recuperada, puesto que la decodificación de estos datos requiere de múltiples llamadas REST que se han tratado de minimizar limitando la información que se muestra.

### 3.2.3. Gestión de usuarios

Al igual que con el módulo anterior, para realizar la gestión de usuarios con Firebase Authentication es suficiente con la creación del proyecto y la utilización de los snippets de código generados a tal efecto y que se pueden encontrar en su documentación.

En este caso, la autenticación de los usuarios se ha decidido que se realice mediante el método de email y contraseña, sin realizar ningún tipo de comprobación de si el correo es real o no, porque tampoco es algo que sea objeto de la funcionalidad de este sistema, ya que no se realizan comunicaciones por correo electrónico en ningún caso.



**Figura 83: Firebase - Método de autenticación**

Tampoco es necesaria para esta aplicación la utilización de roles de usuario puesto que, tal como se definió durante la fase de establecimiento de objetivos y requerimientos, todos los usuarios tendrán disponibles las mismas funcionalidades y serán de la misma categoría.

Además, esta gestión de usuarios se utiliza de forma muy relacionada con la gestión de base de datos anteriormente comentada ya que las reglas de acceso que se definen en la base de datos dependen en gran medida de que el usuario este autenticado con Firebase para poder leer y escribir los datos que correspondan. En caso de que el usuario no se autentique o pierda la sesión, no tendrá acceso de escritura a la información de la base de datos.



Figura 84: Firebase - Reglas de acceso a base de datos

Firebase también proporciona una consola para poder administrar los usuarios creados y poder actualizarlos, borrarlos o cambiar los proveedores de autenticación que se utilizan. De igual modo, se ha añadido un mecanismo en la aplicación para poder borrar la cuenta del usuario de forma completa, eliminando también la información relativa a ese usuario de la base de datos.



Figura 85: Firebase - Administración de usuarios

### 3.2.4. Adaptación a otros idiomas

Como primera aproximación a un sistema multidioma, se ha implementado en esta aplicación la posibilidad de ver su contenido tanto en español como en inglés por medio de los ficheros string.xml para cada uno de los idiomas. En un primer acceso y hasta que el usuario se loguea, la aplicación se muestra con el idioma por defecto en el que el usuario tenga configurado su dispositivo. En el momento que el usuario inicia sesión, el idioma de la aplicación será aquel que haya elegido el usuario en algún acceso anterior o, en su defecto, se configurará el idioma por defecto de su dispositivo.

Además, desde la pantalla de Ajustes, será posible para el usuario cambiar el idioma de la aplicación en cualquier momento.

### 3.2.5. Recuperación, carga y almacenamiento de catálogo de API

Como se ha comentado en puntos anteriores de esta memoria, la información de producto que se recupera del API de producto (Open Food Facts, 2020) llega codificada. Esto quiere decir que, sin una traducción previa, esta información no tendría demasiado sentido para el usuario, tal y como se puede ver en el siguiente pantallazo:

```

"misc_tags": [
  "en:nutrition-fruits-vegetables-nuts-estimate-from-ingredients",
  "en:nutrition-all-nutriscore-values-known",
  "en:nutriscore-computed"
],
"categories_hierarchy": [
  "en:plant-based-foods-and-beverages",
  "en:plant-based-foods",
  "en:cereals-and-potatoes",
  "en:breakfasts",
  "en:cereals-and-their-products",
  "en:breakfast-cereals",
  "en:flakes",
  "en:cereal-flakes",
  "en:extruded-flakes",
  "en:mueslis",
  "en:mixed-cereal-flakes",
  "en:mueslis-with-nuts"
],
"unknown_nutrients_tags": [],
"last_edit_dates_tags": [
  "2020-04-06",
  "2020-04",
  "2020"
],
"ingredients_text": "copos integrales de _avena_, aceite de girasol, inclusiones 7%
semillas de girasol, semillas de calabaza, coco desecado, azúcar, melaza, antioxidan
"labels_prev_tags": [
  "es:bajo-contenido-de-azucar",
  "es:bajo-o-sin-azucar"
],
"nutrition_data_prepared_per_debug_tags": [],
"traces_tags": [
  "en:milk",
  "en:nuts",
  "en:soybeans"
],

```

**Ilustración 86: OpenFoodFacts - Extracto de información de producto**

En las partes resaltadas de este extracto se puede ver como ejemplo la forma en la que tanto las trazas de alérgenos como las categorías vienen representadas. Para un usuario medio y sin conocimientos de inglés, presentar la información de esta forma no tendría ningún sentido.

Es por todo ello por lo que, tras un estudio exhaustivo de la información proporcionada por este servicio REST, se genera la necesidad de cruzar los códigos más relevantes con sus traducciones, recuperadas en sucesivas llamadas a otros servicios del API que también están soportadas para varios idiomas, pero ubicadas en varios puntos de API diferentes. Para ilustrar un poco cómo sería el proceso, se detalla a continuación el flujo de llamadas para completar los datos a mostrar para cada producto:

1. Obtener la información del producto, para lo que es necesario invocar al servicio REST:

*<https://es.openfoodfacts.org/api/v0/product/{barcode}>*

2. Obtener las traducciones para los códigos que aparecen en el campo label\_tags de la información de producto. Para mantener el multiidioma son necesarias dos llamadas, una por idioma a presentar:

*[https://\[es/en\].openfoodfacts.org/labels.json](https://[es/en].openfoodfacts.org/labels.json)*

3. Obtener las traducciones para los códigos de los campos allergens\_tags y traces\_tags, también es necesaria una llamada por idioma:

*[https://\[es/en\].openfoodfacts.org/allergens.json](https://[es/en].openfoodfacts.org/allergens.json)*

4. Obtener las descripciones de las categorías que vienen en el campo categories\_hierarchy, también con llamadas independientes por idioma:

*[https://\[es/en\].openfoodfacts.org/categories.json](https://[es/en].openfoodfacts.org/categories.json)*

5. Lo mismo ocurre con los códigos correspondientes a los niveles de nutrientes (campo nutrient\_levels), con una única llamada se recuperan todas las descripciones necesarias:

*[https://world.openfoodfacts.org/data/taxonomies/nutrient\\_levels.json](https://world.openfoodfacts.org/data/taxonomies/nutrient_levels.json)*

6. También es necesario recuperar la información del campo nova\_groups\_tags, de igual modo con una única llamada:

*[https://world.openfoodfacts.org/data/taxonomies/nova\\_groups.json](https://world.openfoodfacts.org/data/taxonomies/nova_groups.json)*

7. También se deben recuperar, por separado, las traducciones necesarias tanto para los aditivos (`additives_tags`) como para las clases de aditivos (`additives_classes` dentro de los aditivos recuperados), para lo que son necesarias dos invocaciones independientes:

<https://world.openfoodfacts.org/data/taxonomies/additives.json>

[https://world.openfoodfacts.org/data/taxonomies/additives\\_classes.json](https://world.openfoodfacts.org/data/taxonomies/additives_classes.json)

8. Por último, se deben realizar dos llamadas para recuperar el análisis de ingredientes que soporte multiidioma:

[http://\[es/en\].openfoodfacts.org/ingredients-analysis.json](http://[es/en].openfoodfacts.org/ingredients-analysis.json)

Con este ejemplo queda patente que para poder representar de forma completa la información correspondiente a un producto, son necesarias, en el peor de los casos en el que todos los campos mencionados vengan informados, un total de 13 llamadas REST, con el consiguiente gasto de datos para el usuario que eso conllevaría, ya que, además, todas estas llamadas extra recuperan una gran cantidad de registros por cada una de ellas.

Viendo esta situación, se justifica plenamente el uso de la base de datos interna (Sqlite, 2020) del dispositivo para el almacenamiento de esta información y que, en el momento del escaneo, solo sea necesaria una llamada REST y la información de los catálogos a cruzar esté ya disponible en el dispositivo del usuario, sin tener que hacer ninguna llamada extra. Así, también es posible consultar al API cada vez que el usuario quiera ver el detalle de una de sus búsquedas para que la información de la que disponga sea la más actualizada en cada petición.

Para conseguir esto, cada vez que un usuario abra la aplicación se evalúa si ya existe toda la información de los catálogos en la base de datos local y, en caso contrario, se llama de forma asíncrona, mediante un `Executor`, al repositorio que se encarga de realizar todas las invocaciones al API y de convertir las respuestas que reciba a los modelos creados a tal efecto y a almacenar su contenido en la base de datos local. Como se puede inferir por lo comentado, este es un proceso que lleva un tiempo prolongado en realizarse, varias decenas de segundos, y el usuario solo tendrá que esperar a que la información se termine de cargar en caso de que termine las operaciones de registro y login antes de que haya finalizado el proceso de carga. En todo caso, esta espera, dependiendo del estado de la red y del dispositivo utilizado, se reducirá a, en el peor de los casos, la mitad del tiempo estimado en llevarse a cabo.

Así, cuando el usuario quiera consultar la información de un producto ya buscado o realizar un nuevo escaneo, la información que se recupere se traducirá en un primer momento al modelo que representa de forma fiel la respuesta del API y, antes de incluirlo en el modelo de Producto que se utiliza para almacenar sus datos relevantes en la base de datos, tal como se ha comentado en la sección de *Gestión de base de datos*, y manejar su información en la aplicación, se realiza el cruce de los códigos con la información previamente almacenada en la base de datos interna del dispositivo.

### 3.2.6. Limitaciones de Open Food Facts

Como se ha indicado previamente, Open Food Facts (Open Food Facts, 2020) es un proyecto open source mantenido por la comunidad tanto en el aspecto del desarrollo como en la información que maneja y recupera. Esto significa que la información no es estándar para todos los productos y, por ello, hay diferencias de la información indicada para cada uno de ellos tanto en su grado de completado como en la forma de estar incluida esta información. Así, para los productos españoles en general, no suelen ofrecer traducciones ni para sus nombres ni para sus ingredientes. Existen también diferencias en la información guardada en los países de origen de los productos que no se ha visto con la suficiente relevancia como para realizar una nueva llamada REST para la recuperación de los catálogos de países. Esto, quizás, podría plantearse en un desarrollo futuro sobre esta misma aplicación.

Teniendo en cuenta todo esto, la visualización de la información se ha implementado de forma que sea lo suficientemente dinámica para poder ocultar aquellos campos que, para un determinado producto, se encuentren vacíos, pero se trata de algo muy complicado de comprobar para todas las posibles casuísticas aunque, como se indica en el apartado de *Pruebas*, se ha intentado que estas sean lo más exhaustivas posible.

Del mismo modo, también podrá darse la situación de que los productos escaneados no aparezcan en el banco de datos del que extrae el API la información. Aquí es donde entraría en juego la línea de desarrollo futuro que se abrió en el apartado de *Objetivos funcionales a futuro* para permitir añadir productos nuevos utilizando los servicios que ya proporciona el API de Open Food Facts de forma que el banco de datos vaya completando cada vez más la información de la que dispone utilizando diferentes fuentes.

### 3.2.7. Búsqueda de cambios en productos favoritos y envío de notificaciones

Uno de los objetivos que diferencia esta aplicación de las ya existentes es la funcionalidad que realiza la búsqueda de cambios en aquellos productos configurados como favoritos y, de haber alguno, envía notificaciones al dispositivo del usuario para avisar de esta situación.

Para realizar esta funcionalidad, se ha optado por el desarrollo de un servicio en segundo plano de ejecución periódica y cuyo lanzamiento se realiza la primera vez que se autentica con un determinado usuario. Al realizarlo así, no se consumen recursos extra del dispositivo del usuario cuando lo está utilizando ni la experiencia de usuario se ve impactada.

Así, el servicio se ejecutará cada 24 horas, siempre que el dispositivo esté conectado a una red WIFI, e irá realizando comprobaciones sobre los productos favoritos que estén configurados para buscar actualizaciones en la fecha que coincida con el día actual. El servicio entonces recuperará tanto de la base de datos como del API de producto la información relacionada con el producto favorito que se esté analizando. Con estos dos sets de información se harán comprobaciones, tanto de tamaño como de contenido entre ellos y sólo de cierta información considerada relevante como son los alérgenos, las trazas, el grado de completado, el análisis de ingredientes, los aditivos, la calificación nova group y nutriscore y el nivel de nutrientes.

Si en alguna de estas comprobaciones se detecta algún cambio, se realizará el envío de una notificación informando del producto en el que se ha encontrado alguna diferencia y, además, se marcará el producto como con información nueva para que, desde la interfaz gráfica, sea posible identificar cuál es en caso de que se haya perdido la notificación enviada.

Para el envío de notificaciones es necesaria la configuración del servicio Firebase Cloud Messaging (Firebase, 2020) para el que se ha implementado un envío asíncrono de notificaciones utilizando el propio endpoint del servicio y un listener que, si detecta que la aplicación está en primer plano y se recibe una notificación, la vuelve a enviar al dispositivo para que el usuario pueda verla.

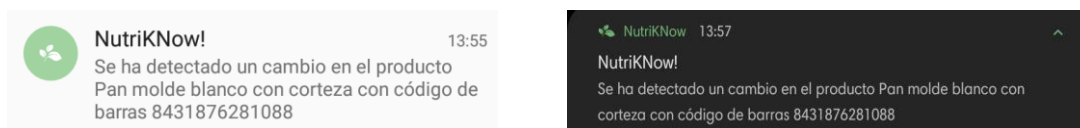


Figura 87: Firebase – Notificaciones push en diferentes versiones de Android

De forma general, en todos los casos, se actualizará la fecha de nueva actualización del modelo de producto favorito en base de datos con la fecha que corresponda en base a la periodicidad que tenga el usuario configurada en su perfil en el momento de la actualización.

Como limitación de este servicio, se debe comentar que este servicio en segundo plano comprobará la información del último usuario conectado a la aplicación en ese dispositivo, para que el envío de notificaciones sea lo más personalizado posible.

### 3.2.8. Interfaz de usuario

En el apartado *Funcionalidades de la aplicación* se detallará toda la parte funcional de las diferentes pantallas de la aplicación, pero existen ciertos elementos que tienen la entidad suficiente para ser incluidos en una sección aparte y explicar tanto su funcionamiento como las razones que justifican su uso, así también se simplifica la sección de funcionalidades para centrarse más en los aspectos técnicos de cada actividad o fragment y no en los detalles de los componentes que los conforman, que ya estarán explicados previamente.

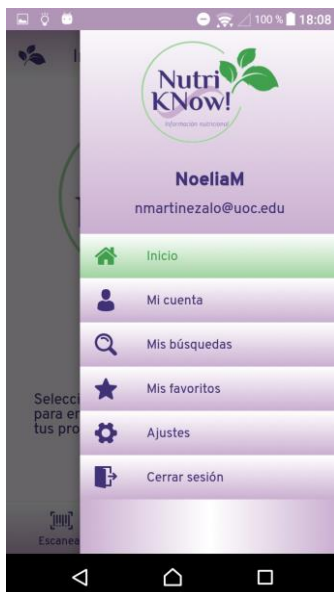
También es importante indicar en relación a la interfaz de usuario que se ha tratado de implementar una interfaz gráfica lo más responsive posible, realizando el dimensionamiento utilizando medidas relativas para que, según el tamaño de cada dispositivo utilizado, se ajusten las dimensiones que mejor se adecúen a cada caso particular. Del mismo modo, las pruebas se han realizado en dispositivos de diferentes dimensiones para asegurarse de que el contenido, aunque se ajuste al tamaño de cada uno, es perfectamente visible en todos ellos.

#### 3.2.8.1. DrawerLayout y NavigationView

Estos dos componentes conforman el menú lateral de la aplicación, decidiendo en base a la opción del menú pulsada, el fragment a presentar al usuario dentro de la actividad general llamada *WelcomeActivity*. En el caso del componente *NavigationView* se ha utilizado el componente que proporciona Material (Material Design, 2020) para aprovechar la facilidad de personalización en cuanto a estilos de la que se dispone utilizando esta librería.

Además, para darle un funcionamiento más personalizable al menú, se ha incorporado al proyecto una clase proporcionada por la comunidad open source de Android que se llama *EndDrawerToggle* y que permite que el menú aparezca en el lado derecho para dejar el lado izquierdo libre para el icono de ir a la pantalla anterior.

Cabe destacar que en esta sección se mostrarán los pantallazos realizados con el dispositivo Sony Xperia F311 con Android 6.0, ya que la visualización de estos componentes se ajusta perfectamente a su tamaño. A continuación se adjunta una captura de pantalla de este menú:



- ✓ **Inicio:** es la opción para volver a la pantalla de inicio desde cualquier punto de la aplicación (*HomeFragment*).
- ✓ **Mi cuenta:** permite consultar la información del usuario y también borrar la cuenta (*UserAccountFragment*). También da acceso al fragment que permite modificar el perfil del usuario (*ProfileSetupFragment*).
- ✓ **Mis búsquedas:** accede a la lista de búsquedas realizadas por el usuario (*SearchesFragment*).
- ✓ **Mis favoritos:** esta opción navega a la lista de productos favoritos del usuario (*FavoritesFragment*).
- ✓ **Ajustes:** pulsando esta opción se carga el fragment para modificar los ajustes de la aplicación (*SettingsFragment*).
- ✓ **Cerrar sesión:** realiza el cierre de sesión del usuario conectado en ese momento.

Figura 88: Menú lateral de la aplicación

### 3.2.8.2. BottomNavigationView

Este componente es similar al anterior pero el número de opciones recomendadas está limitado por la guía de diseño de Material Design (Material Design, 2020), estableciéndolo a más de 3 y menos de 5 opciones. En este caso, son tres las opciones que se muestran, siendo una de ellas la que enlaza con la funcionalidad de **Escanear** (ScanFragment) y las dos restantes opciones análogas a las presentadas en el menú anterior: **Búsquedas** para cargar las búsquedas disponibles del usuario (SearchesFragment) y **Favoritos** para cargar los productos favoritos del usuario (FavoritesFragment).



Figura 89: Barra inferior de la aplicación

### 3.2.8.3. ListView

En la aplicación desarrollada se muestran diferentes tipos de listas, personalizadas en cuanto a estilos y funcionalidad de diferentes formas dependiendo de la pantalla concreta donde se esté mostrando cada lista. Por ejemplo, la lista que se muestra en la pantalla de la información del usuario no tiene un estilo definido ni funcionalidad más allá de la de mostrar datos, la lista de la pantalla de la configuración del perfil, permite añadir y borrar registros de forma dinámica y tiene un estilo muy diferenciado al resto de listas mostradas en la aplicación y, por último, las listas que se muestran en las pantallas en las que se recoge la información del producto no tienen funcionalidad definida pero sí personalización en cuanto a la información y los iconos que muestran. Para estas listas se utilizan los componentes que proporciona el SDK de Android y se cargan mediante el uso de adaptadores y layouts específicos en cada caso.

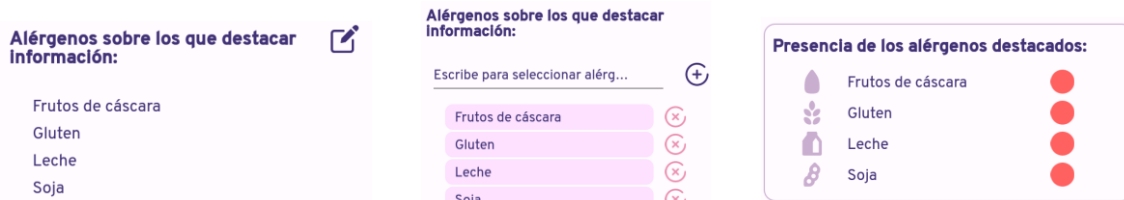


Figura 90: Diferentes tipos de listas de la aplicación

### 3.2.8.4. ExpandableListView

Existe un tipo de lista especial que se utiliza para mostrar información relacionada con otros datos en varios niveles. Estas listas son llamadas expandibles puesto que suelen permitir el mostrado u ocultación de información a voluntad del usuario. En este caso, se ha utilizado para mostrar la información de los aditivos en función de las clases a las que pertenece cada uno de ellos. Tras realizar multitud de pruebas, se ha optado por mostrar la lista siempre expandida y no permitir que la información pueda recogerse puesto que se ha podido comprobar que el rendimiento del componente deja bastante que desear. También, como en el caso anterior, utilizan adaptadores definidos con los layouts que correspondan en cada caso.

Para realizar la implementación de este tipo de lista, se ha utilizado una clase proporcionada por la comunidad open source de Android (ExpandExpandableListView) que permite un mayor control sobre el componente, así como definir el espacio que ocupa la lista de una manera más óptima.



Figura 91: Lista expandible



### 3.2.8.5. RecyclerView y Cardview

En los últimos tiempos, las aplicaciones modernas cada vez utilizan más las cards para el mostrador de información. Este tipo de componente ofrece al desarrollador un alto número de personalizaciones en cuanto a los layouts que pueden mostrar. La estructura de una card se compone típicamente de un título y un texto de detalle, pero existen plantillas de cards realmente complejas. En este caso, las cards se utilizan en conjunción con adaptadores que les proporcionan los datos a mostrar y dentro de una estructura llamada RecyclerView que es una versión actualizada y más optimizada de un ScrollView integrado con una ListView. Este componente se utiliza en los fragments que muestran la información de las búsquedas y la lista de productos favoritos del usuario. En este caso, se han utilizado los componentes propios de Android y la información que se muestra en cada card se detalla a continuación:

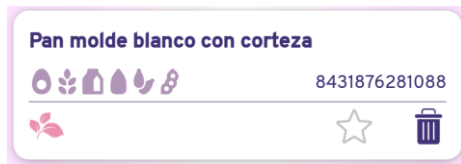


Figura 92: Contenido de card de la aplicación

- ✓ Nombre del producto, actuando como título.
- ✓ Código de barras del producto, siendo el detalle.
- ✓ Un conjunto de iconos que representan los alérgenos que contiene el producto en cuestión.
- ✓ Una barra inferior en la que se localizan las acciones disponibles.
- ✓ Un flag con el logo de la aplicación que indica que el producto ha sido escaneado recientemente o que se han detectado cambios en su composición.

### 3.2.8.6. TabLayout y ViewPager

Como se ha comentado anteriormente, la información que devuelve el API de producto cada vez que se realiza una consulta puede ser mucha, en función de los datos aportados para cada uno de los productos al banco de datos. Por ello, ha sido necesario organizar la información de una forma lo suficientemente estructurada para que no abrume al usuario, sea fácil de leer y pueda verse la información relevante de un vistazo. La estructura que mejor cumple estos requisitos es el TabLayout que se ha incorporado al ProductInfoFragment y que establece una estructura de pestañas en las que un adaptador indica el número de pestañas a mostrar (lo ideal es que sean tres como mínimo y no más de cinco) y el fragment a visualizar en cada una de ellas. También usa un ViewPager que le envía la información que se necesita mostrar.

Para el componente TabLayout, también se ha utilizado el proporcionado por Material Design (Material Design, 2020) dada su facilidad de integración y las posibilidades de personalización mediante contenido y estilos que ofrece.



Figura 93: Pestañas de la aplicación

### 3.2.8.7. Floating Action Button



Figura 94: Botón flotante

Este componente se puede encontrar en muchas de las aplicaciones de uso diario actualmente. Se trata de un botón que se localiza en la parte inferior de la pantalla y a la derecha de la misma. Normalmente desaparece mientras se hace scroll y vuelve a aparecer al deja de navegar. En esta aplicación no se ha implementado esta funcionalidad debido a la cantidad de información mostrada en cada uno de los fragments donde aparece.

En la implementación, se ha incluido en el fragment que recopila la información de producto (ProductInfoFragment) pero la disposición de los mostrados en cada una de las pestañas, hace que pueda visualizarse en todas ellas. La funcionalidad que tiene asignada este botón es la de añadir o borrar un determinado producto de la lista de favoritos, en función de su estado previo, por eso tiene sentido que esté presente en cada una de las pestañas. En este caso también se ha utilizado el componente que proporciona Material Design.

### 3.2.8.8. CameraSource, SurfaceView y BarcodeDetector

Al tratarse de una aplicación que realiza escaneado de códigos de barras, se hace necesario poder controlar la cámara del dispositivo y reflejar la imagen que se está obteniendo en algún punto de la pantalla. Para ello, se utiliza una SurfaceView, que permite la utilización de un tipo especial de canvas para ver la imagen que se está capturando con la cámara del dispositivo.

El acceso a la cámara se realiza mediante la implementación existente de CameraSource. En este caso, como también era necesaria la utilización del flash del dispositivo en caso de que disponga de él, se ha tenido que incorporar una clase proporcionada por la comunidad Android, puesto que la última versión de esta librería no permite acceder a esta funcionalidad.

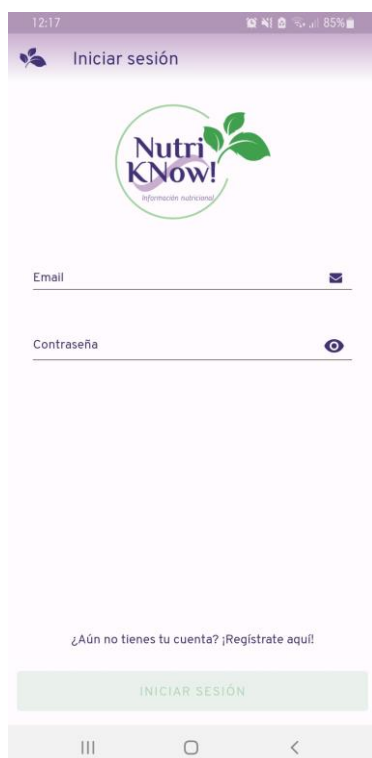
Por último, también es necesario incorporar un mecanismo para que, en el momento que se detecte la presencia de un código de barras, el sistema capture la imagen sin necesidad de ningún tipo de interacción por parte del usuario. Este es el objetivo de BarcodeDetector.

## 3.3. Funcionalidades de la aplicación

En este apartado se detalla cada una de las ventanas (activity o fragment) de la aplicación desde el punto de vista técnico comentando los datos más relevantes de su implementación. Para mostrar las pantallas en esta sección se utilizan las capturas de pantalla realizadas con el dispositivo Samsung S9 con Android 10 ya que es un dispositivo de mayores dimensiones.

### 3.3.1. Gestión de usuarios

#### 3.3.1.1. Inicio y cierre de sesión



La actividad LoginActivity se encarga de gestionar la autenticación de un usuario, siempre que haya sido creado previamente. Los objetos TextInputEditText son los que provee Material Design (Material Design, 2020) y, además, se han añadido validaciones para garantizar que el texto que se introduce en el campo Email sea una dirección de correo electrónico y que la contraseña tenga seis o más caracteres.

Si todo esto se cumple, el botón Iniciar sesión se habilita y se realiza la llamada a Firebase Authentication (Firebase, 2020) para realizar la autenticación en el servicio con el método signInWithEmailAndPassword y también se realiza la carga en memoria de la información de base de datos del usuario. Se informa al usuario del resultado de esta llamada mediante un Snackbar tanto en el caso exitoso como en el erróneo. Como también se ha comentado anteriormente, es en esta actividad en la que se inicia en la carga en segundo plano de los catálogos del API de producto.

Para cerrar sesión, se debe utilizar la opción a tal efecto del menú lateral, que tras cerrar la sesión con el método signOut de Firebase, redirige de nuevo a esta pantalla.

### 3.3.1.2. Registro de usuario

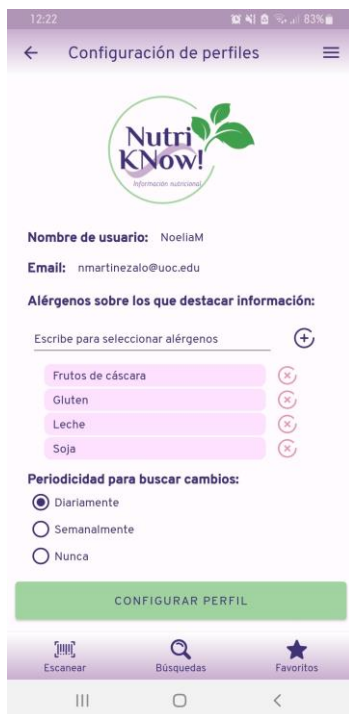
En caso de la activity RegisterActivity el objetivo es la creación de un usuario del sistema. Esta activity no requiere de los datos de catálogo que comenzaron a cargarse en la activity de inicio de sesión por lo que no espera a su carga completa.

Al igual que en el caso anterior, los TextInputEditText también son de Material Design y tienen asociadas validaciones similares a las anteriores: tanto el nombre de usuario como la contraseña debe ser mayor de seis caracteres, el contenido del campo email debe ser una dirección de correo electrónico y que el contenido de los dos campos referentes a la contraseña coincidan. Si las validaciones no se cumplen, se mostrará un mensaje debajo del campo en cuestión para informar de esta situación.

Si se cumplen las validaciones, el botón Crear usuario se habilita y se crea el usuario en Firebase mediante el método createUserWithEmailAndPassword. Además, también se crea un nodo en Firebase Realtime Database para guardar la información del usuario.



### 3.3.1.3. Configuración de perfiles



El fragment ProfileSetupFragment se encarga de la configuración del perfil del usuario. Se muestra autocontenido dentro de la activity WelcomeActivity y se puede llegar a esta pantalla bien como primer paso después de crear un usuario e iniciar sesión por primera vez o bien desde la pantalla Mi cuenta una vez iniciada sesión. La diferencia entre los dos accesos es que con el primero no están accesibles ni el menú lateral ni la barra inferior y con el segundo la pantalla se muestra tal como se ve en el pantallazo proporcionado. Este fragment sí necesita de la información de catálogo del API de producto, por lo que el usuario deberá esperar a su carga completa.

Este fragment dispone de un AutoCompleteTextView gestionado mediante un ArrayAdapter para cargar los alérgenos destacados y que muestra las coincidencias en una capa flotante a medida que el usuario escribe. El alérgeno seleccionado se puede añadir a otra lista cuyo layout permite borrar registros. Dicha lista no permite duplicados y solo permitirá añadir un registro cuando el texto coincida totalmente con alguno de la lista de la capa flotante.

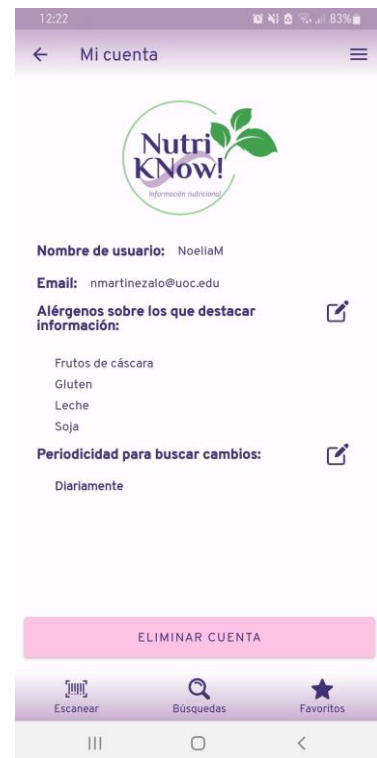
También es posible configurar la periodicidad de la búsqueda de cambios de los productos favoritos. Para ello se ha utilizado un RadioGroup con las tres opciones disponibles. Al pulsar el botón Configurar perfil, se crea un nodo Profile dentro de la base de datos del usuario conectado con los alérgenos y la periodicidad y si existen productos favoritos, se actualiza su fecha de nueva búsqueda de cambios con la calculada a partir de la periodicidad seleccionada.

### 3.3.1.4. Mi cuenta y borrado de cuenta de usuario

Al fragment `UserAccountFragment` se llega utilizando la opción `Mi cuenta` del menú lateral. Esta pantalla se correspondería con la pantalla de solo lectura de la detallada anteriormente para la configuración del perfil. Este fragment, como también el resto, se muestra autocontenido dentro de la actividad `WelcomeActivity` para tener acceso tanto al menú lateral como a la barra inferior.

Muestra una lista con los alérgenos destacados por el usuario, también gestionada con un `ArrayAdapter`, y la opción seleccionada para la periodicidad. En este fragment no se permite la edición de ninguno de estos aspectos, simplemente se da acceso mediante el icono de edición a la pantalla de Configuración de perfiles en la que, como se ha visto anteriormente, sí se puede editar.

También se permite realizar el borrado de la cuenta de usuario. Al pulsar el botón `eliminar cuenta`, se muestra un `AlertDialog` para que el usuario confirme si está seguro de querer borrar la cuenta. Si acepta, se elimina el usuario en `Firebase` con el método `delete` y toda la información relacionada de la base de datos.



### 3.3.2. Gestión de la aplicación

#### 3.3.2.1. Inicio



El fragment `HomeFragment` es un fragment muy sencillo que se limita a actuar como pantalla de inicio de la aplicación, mostrando el logo de esta y un mensaje indicando cómo empezar a utilizarla. Este mensaje actualmente es mínimo, pero en un futuro podría tener más contenido incluyendo más recomendaciones e incluso, consejos de salud.

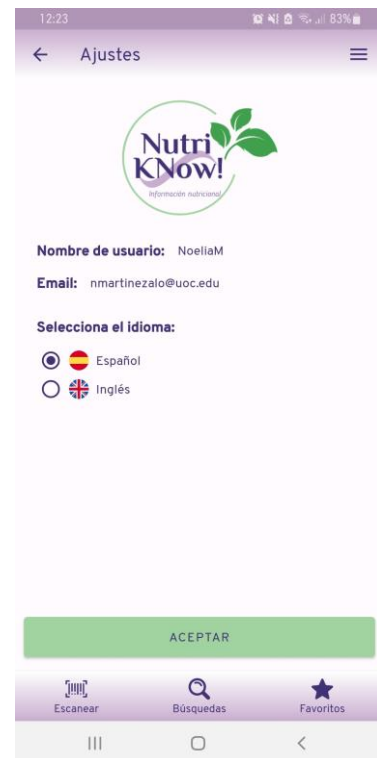
Para dar un poco más detalle de la actividad `WelcomeActivity`, se puede remarcar que, como se ha dicho en anteriores secciones, contiene la implementación de tanto el menú lateral como la barra inferior. Además, también contiene funciones comunes utilizadas por todos o varios fragments como es la obtención la información de producto, diversas redirecciones a otras pantallas o, incluso, la definición de variables globales utilizadas en diversos puntos de la aplicación. Del mismo modo, también hace uso de varios métodos comunes que están definidas en la actividad de la que hereda (`CommonActivity`) como es la carga de datos de la base de datos local del dispositivo.

### 3.3.2.2. Ajustes

El fragment `SettingsFragment` recoge la funcionalidad del cambio de idioma de la aplicación y se llega a ella mediante la opción Ajustes del menú lateral. En un futuro esta funcionalidad podrá ampliarse a otros aspectos configurables, como podría ser la utilización del modo oscuro que ya muchos dispositivos permiten o el cambio en la gama de colores de la aplicación en función de las preferencias del usuario.

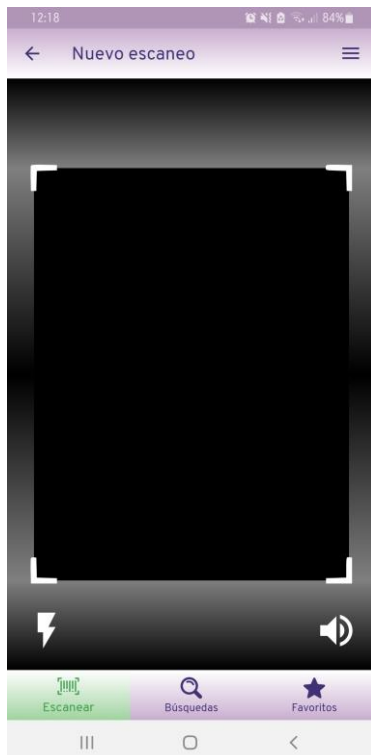
Actualmente los idiomas configurables son dos: español e inglés. Como ya se ha comentado, en un primer momento, el idioma en el que se muestra la aplicación es aquel en el que el usuario tiene configurado por defecto en el dispositivo. Cuando el usuario inicia sesión, en la actividad `WelcomeActivity` se guarda el idioma por defecto en el nodo Perfil correspondiente de la base de datos para mostrar siempre la aplicación en este idioma.

Si el usuario navega a esta opción y cambia el idioma, además de traducirse la aplicación a dicho idioma, también se guarda este en su perfil para, en próximos accesos, mostrar directamente todos los textos en dicho idioma.



### 3.3.3. Escaneo de productos

#### 3.3.3.1. Nuevo escaneo



El fragment `ScanFragment` se encarga de facilitar el escaneo de códigos de barras de productos. Como se detalló en el punto referente al objeto `CameraSource`, este fragment muestra una `SurfaceView` en la que se refleja lo que se captura a través de la cámara del dispositivo del usuario. Se permite la activación y desactivación tanto del flash del dispositivo como la emisión de un sonido cuando la detección se produzca.

Como la `SurfaceView` tiene asociado también un `BarcodeDetector`, en el momento en el que detecta un código de barras, lo captura y se detiene la toma de imágenes. Una vez se ha obtenido el código de barras, se procede a la invocación mediante Retrofit del API de producto para obtener la información correspondiente a ese código de barras.

En caso de que el producto no exista aún en el código de barras, se muestra un mensaje indicándolo. En caso contrario, se carga la información del producto, proceso que tarda un tiempo variable en segundos en función de la cantidad de información disponible para dicho producto.

### 3.3.3.2. Información de producto

Al fragment `ProductInfoFragment` se llega tras escanear un producto o consultar la información de uno existente, con la funcionalidad `TabLayout` y al `Floating Action Button`.

### 3.3.3.3. Resumen de la información del producto

El fragment `SummaryInfoFragment` es el fragment mostrado en la primera de las pestañas definidas en el fragment `ProductInfoFragment` y aglutina la información resumida del producto, así como la información relevante relativa a los alérgenos destacados por el usuario en su perfil.

En todas las pestañas se muestra en primer lugar una sección destacada con el nombre del producto y el grado de completado en el que se encuentra su información. Luego, se muestran diversas listas gestionadas con layouts y adapters específicos y customizados para cada caso.

La información que se puede mostrar en esta pantalla es:

- ✓ Listados de alérgenos y trazas relevantes.
- ✓ Información resumida del producto (marcas, tiendas, países, etc.).
- ✓ Análisis de ingredientes (vegetariano, vegano, aceite de palma).
- ✓ Reconocimientos conseguidos por el producto.



### 3.3.3.4. Ingredientes del producto



El fragment `IngredientsInfoFragment` es el fragment mostrado en la segunda de las pestañas definidas en el fragment `ProductInfoFragment` y comprende la información del producto relativa a su grado de procesado y presencia de aditivos. También este fragment muestra la información destacada del nombre de producto y grado de completado.

La información que, como máximo, se muestra en esta pantalla adicionalmente es:

- ✓ Descripción de ingredientes (normalmente sin posibilidad de internacionalización en la aplicación).
- ✓ Listado completo de alérgenos y trazas presentes.
- ✓ Calificación grupos NOVA en función del número de ingredientes y la presencia de aditivos.
- ✓ Lista de aditivos y las clases a las que pertenecen (detallada en `ExpandableListView`).
- ✓ Listado de categorías a las que pertenece el producto.



### 3.3.3.5. Información nutricional del producto

El fragment `NutritionalInfoFragment` es el fragment mostrado en la tercera de las pestañas definidas en el fragment `ProductInfoFragment` y detalla la información nutricional del producto siempre que esté disponible, así como su evaluación `Nutri-Score` y sus niveles de nutrientes. También este fragment muestra la información destacada del nombre de producto y grado de completado.

La información que, como máximo, se muestra en esta pantalla adicionalmente es:

- ✓ Lista de niveles de nutrientes clasificados por contenido en grasas, azúcares y sal.
- ✓ Evaluación `Nutri-Score` en función de la presencia de elementos favorables o desfavorables desde el punto de vista nutricional.
- ✓ Información nutricional indicada por cada 100g y en comparativa con el peso en el que venga comercializado el producto, si está disponible.



### 3.3.4. Gestión de productos

#### 3.3.4.1. Búsquedas



En el fragment `SearchesFragment` se muestra el listado de escaneos (búsquedas) realizados por el usuario utilizando los recursos detallados en el punto `RecyclerView` y `CardView` gestionados por medio de un adaptador personalizado para esta pantalla. A este fragment se puede acceder mediante la opción `Búsquedas` en la barra inferior o la opción `Mis búsquedas` del menú lateral de la aplicación.

Como se ha comentado, cada card muestra los alérgenos presentes en el producto, un flag indicando si es de escaneo reciente o tiene algún cambio y entre las acciones disponibles por registro está añadir a la lista de productos favoritos (una vez añadido, el icono se deshabilita y cambia su apariencia) y borrar búsqueda (que se encarga de borrar de la base de datos de `Firestore` cualquier nodo que haga referencia al producto en cuestión, tanto de productos, búsquedas o favoritos).

Además, pulsando en cualquier punto de la card se puede acceder a la información del producto, acción que dará por consultado el producto, hará que desaparezca el flag de nuevo producto y también actualizará la fecha de búsqueda de cambios si el producto se añadió previamente a favoritos.

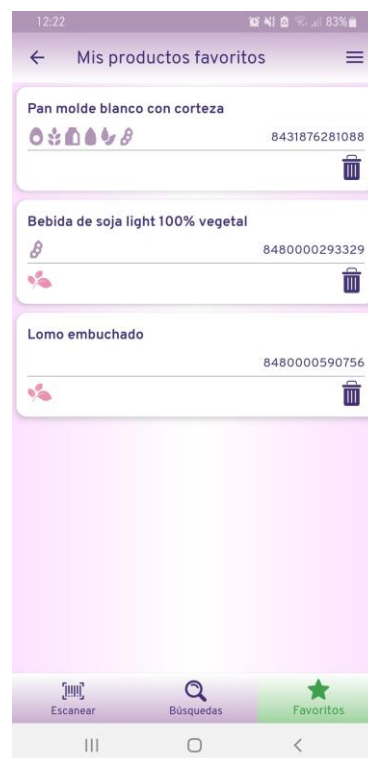
### 3.3.4.2. Favoritos

En el fragment `FavoritesFragment` se muestra el listado de productos favoritos del usuario utilizando los recursos detallados en el punto *RecyclerView y CardView* gestionados por medio de un adaptador personalizado para esta pantalla. A este fragment se puede acceder mediante la opción Favoritos en la barra inferior o la opción Mis favoritos del menú lateral de la aplicación.

El diseño de las cards es el mismo que el comentado para la pantalla anterior, con la diferencia de que en este caso solo se permite la acción de borrado del listado de favoritos que borra el producto únicamente del nodo de favoritos.

Además, pulsando en cualquier punto de la card se puede acceder a la información del producto, acción que dará por consultado el producto, hará que desaparezca el flag de nuevo producto y también actualizará la fecha de búsqueda de cambios si el producto se añadió previamente a favoritos.

En ambas pantallas, si la lista de búsquedas o favoritos está vacía, se muestra una pantalla similar a la de inicio pero adecuando el mensaje que contiene a la ventana en la que se encuentra.



## 4. Pruebas

Las pruebas de este sistema se han distribuido en dos vías, detalladas en las siguientes subsecciones. Por un lado, se han realizado pruebas automatizadas, siendo estas limitadas al tratarse de una aplicación que necesita del uso de la cámara del dispositivo para escanear los productos, una situación difícil de simular con un emulador. Por otro lado, se han realizado pruebas manuales, primero de forma modular a medida que el desarrollo avanzaba y finalmente de forma integrada con los diferentes dispositivos de los que dispone la estudiante, así como hasta cuatro usuarios diferentes de su entorno que probaron la aplicación siguiendo el plan de pruebas con sus dispositivos personales. Todas las pruebas se realizaron teniendo conexión a internet puesto que, como esta aplicación recupera continuamente información del API de producto, la falta de conexión no es una situación adaptable en este caso.

### 4.1. Dispositivos utilizados en las pruebas

En cuanto a los dispositivos que se han utilizado durante las pruebas, las pruebas modulares e integradas de la estudiante se han realizado con los siguientes smartphones, para poder probar con diferentes procesadores y tamaños de pantalla diversos:

- ✓ Sony Xperia F311 con Android 6.0 (Marshmallow)
- ✓ Samsung Galaxy S9 con Android 10 (Q)

También, siempre que la cuenta de usuario con la que se accediera tuviera ya productos escaneados, se ha utilizado el siguiente emulador para probar el funcionamiento de la aplicación, así como su adecuación a la visualización en pantallas pequeñas:

- ✓ Emulador Nexus One con Android 7.1 (Nougat)

Tal como se ha comentado, las pruebas integradas manuales se han extendido a otros usuarios del entorno de la estudiante, con dispositivos y habilidades tecnológicas diferentes:



- ✓ Nexus 6P con Android 8.1 (Oreo)
- ✓ Samsung Galaxy S9 con Android 10 (Q)
- ✓ Xiaomi mi6 con Android 9 (Pie)
- ✓ Samsung Galaxy S20 con Android 10 (Q)



Figura 95: Capturas de pantalla en los diferentes dispositivos físicos

En las siguientes líneas también se puede consultar una comparativa entre las características de los diferentes dispositivos físicos utilizados para las pruebas:

	SONY XPERIA F311	SAMSUNG GALAXY S9	NEXUS 6P	XIAOMI MI6	SAMSUNG GALAXY S20
<b>Versión</b>	6.0	10.0	8.1	9.0	10.0
<b>Pantalla</b>	5"	5.8"	5.7"	5.15"	6.2"
<b>Resolución</b>	720 x 1280	1440 x 2960	1440 x 2560	1080 x 1920	1440 x 3200
<b>Densidad</b>	294 ppi	568 ppi	515 ppi	428 ppi	566 ppi
<b>Procesador</b>	MediaTek MT6755	Exynos 9 Octa 9810	Snapdragon 810	Snapdragon 835	Exynos 990
<b>Memoria</b>	2 GB	4 GB	3 GB	6 GB	8 GB
<b>Almacenamiento</b>	18 GB	64 GB	128 GB	128 GB	128 GB
<b>Cámara</b>	13 MP	12.2 MP	12.3 MP	12.2 MP	64 MP

Figura 96: Tabla comparativa de dispositivos de prueba

## 4.2. Pruebas automatizadas

Para la realización de las pruebas automatizadas de esta aplicación se ha utilizado el framework Espresso (Espresso, 2020). Espresso es un framework open source que permite el desarrollo y ejecución de pruebas automatizadas sobre la interfaz gráfica de Android.

En total, se han desarrollado 23 casos de prueba en tres clases diferentes, divididas por la actividad principal que se ejecuta. También se han realizado de tal modo que tanto la base de datos como el registro de usuarios no sufre ninguna alteración al terminar su ejecución, quedando ambos sistemas como si el proceso de testing no hubiera tenido lugar.

Para la ejecución de estos tests, de los que se adjunta una captura de pantalla del resultado de ejecución, se deben tener en cuenta las siguientes consideraciones:

Test Case	Execution Time
edu.uoc.tfg.nutriknow.LoginActivityTest	37 s 968 ms
login_with_profile_OK	9 s 153 ms
login_incorrect_password_KO	2 s 732 ms
login_without_profile_OK	11 s 541 ms
go_to_register_user_OK	3 s 696 ms
login_incorrect_email_KO	1 s 779 ms
login_KO	9 s 67 ms
edu.uoc.tfg.nutriknow.RegisterActivityTest	43 s 598 ms
register_KO	13 s 990 ms
register_OK	9 s 655 ms
register_incorrect_username_KO	1 s 373 ms
register_not_matching_passwords_KO	6 s 20 ms
register_incorrect_password_KO	4 s 108 ms
register_incorrect_email_KO	2 s 748 ms
register_incorrect_repeat_password_KO	5 s 704 ms
edu.uoc.tfg.nutriknow.WelcomeActivityTest	1 m 52 s 579 ms
welcome_navigation_view_profile_setup_OK	17 s 879 ms
welcome_init_OK	8 s 502 ms
welcome_navigation_view_my_account_OK	13 s 499 ms
welcome_bottom_navigation_view_searches_OK	9 s 120 ms
welcome_bottom_navigation_view_favorites_OK	8 s 912 ms
welcome_navigation_view_home_OK	10 s 671 ms
welcome_navigation_view_settings_OK	12 s 917 ms
welcome_navigation_view_my_searches_OK	9 s 610 ms
welcome_navigation_view_logout_OK	11 s 391 ms
welcome_navigation_view_my_favorites_OK	10 s 78 ms

Figura 97: Resultados de ejecución de tests automatizados

- ✓ Los tests automatizados se deben ejecutar una vez se ha instalado la aplicación en el dispositivo de pruebas, para asegurarse de que el catálogo se ha cargado correctamente. Una posible mejora sobre el proceso de testing sería que la ejecución espere hasta que el proceso asíncrono de carga de datos termine para comenzar con las validaciones.
- ✓ Se han incluido casos de prueba para la activity de login, tanto con resultado correcto como erróneo, de registro de usuario, teniendo en cuenta las posibles situaciones que pueden darse, y para la activity WelcomeActivity, probando las navegaciones tanto del menú lateral como de la barra inferior. No ha sido posible realizar el testeo del fragment ScanFragment puesto que, en las pruebas en emulador, por un bug interno de la librería estándar utilizada, los modelos de detección de códigos de barras no se descargan y la ejecución se embucla.

### 4.3. Pruebas manuales

Las pruebas manuales, tal como se ha comentado en la introducción a esta sección, se han realizado agrupadas en dos grandes grupos. Primero, se realizaron las pruebas propias del desarrollo y, luego, aquellas realizadas por potenciales usuarios del entorno de la estudiante.

En cuanto a las pruebas por módulos a medida que la implementación iba avanzando y de las que se ha hecho cargo la estudiante, se han realizado de forma ordenada e incremental. Primero se han probado las funcionalidades más pequeñas y, a medida que el desarrollo se iba completando, la complejidad de las pruebas iba aumentando. Por lo que estas pruebas se han enfocado primero de forma unitaria y luego se han ido integrando diferentes funcionalidades y navegaciones hasta que el desarrollo se completa. De estas diferentes pruebas se han obtenido las capturas de pantalla incluidas en la sección *Implementación*.

Junto con la implementación de la aplicación se realizó el desarrollo de los tests automatizados que se explican en la sección anterior. Una vez que tanto el desarrollo como ambos tipos de pruebas finalizaron con resultado favorable, se prepara la siguiente fase que consiste en pruebas con otros usuarios. Para hacerlo de una manera más estable y con trazabilidad de las versiones, se ha decidido subir la aplicación a Google Play (Play.google.com, 2020).

Para ello, es necesario crear una página de información de la aplicación y crear una prueba interna, con una versión de la aplicación firmada. Cuando se ha añadido toda la información de la aplicación y la primera versión, la aplicación entra en un proceso de revisión por Google que, en este caso y dada la situación que se está viviendo a nivel mundial, ha tardado unos cinco días en tener un resultado favorable. Además, como la aplicación hace uso del permiso de acceso a la cámara, ha sido necesaria la creación de una web que contenga la política de privacidad de la aplicación para enlazar desde su página y que se ha alojado haciendo uso de Firebase Hosting (Firebase, 2020). Dicha política se puede consultar en el siguiente enlace:

<https://nutriknow-privacy-policy.web.app/>

Con la aplicación subida y revisada y la política de privacidad enlazada, se ha generado el siguiente enlace de acceso para distribuir entre los diferentes testers de la aplicación:

<https://play.google.com/apps/internaltest/4699707697721224395>

Esta prueba ha permanecido abierta durante todo el proceso de prueba con usuarios y se han realizado diferentes iteraciones a medida que los testers iban aportando sugerencias y mejoras en el funcionamiento de la aplicación hasta llegar a la versión 1.3 que es la última creada.

Para orientar el proceso de pruebas de los usuarios externos que no conocen a fondo la finalidad de la aplicación y de forma previa a la creación del manual de usuario, se les ha aportado el siguiente guion de pruebas con las tareas que debían realizar y que cumplen casi en su totalidad las funcionalidades definidas en el apartado *Casos de uso*. La intención de este guion es que se realicen unas pruebas lo más amplias posibles por diferentes usuarios y que sea un proceso que cubra al máximo el desarrollo realizado:

1. Descargar, instalar y abrir la aplicación a partir del enlace indicado anteriormente.
2. Registrarse como usuario de la aplicación.
3. Iniciar sesión.
4. Configurar el perfil añadiendo al menos un alérgeno para destacar.
5. Acceder a Mi cuenta y editar el perfil para añadir otro alérgeno más.
6. Escanear un producto alimenticio y consultar su información.

7. Escanear un producto no alimenticio y observar que no se encuentra en el banco de datos.
8. Consultar la búsqueda y añadir el producto a la lista de favoritos.
9. Consultar la lista de favoritos y eliminar el producto de dicha lista.
10. Cerrar sesión en la aplicación.

A partir de este proceso de prueba, se ha recibido feedback de cada uno de los usuarios acerca de cómo fue la experiencia. A nivel global, no se detectó ningún fallo de impacto que hiciera que la aplicación se cerrase, pero se comunicaron ciertos detalles que se indican:

- ✓ Al añadir productos a la lista de favoritos cuyo código de barras tiene una longitud diferente al resto, en un primer momento la lista de favoritos se ordena de forma errónea.
- ✓ Se produce un error puntual al escanear un determinado producto por un formato de dato erróneo que no produce el cierre, pero sí detiene el detector de códigos de barras.
- ✓ Cuando el proceso de autenticación falla, una vez se muestra el snackbar con el error correspondiente, se habilita de forma errónea el botón de Iniciar sesión.
- ✓ En la pantalla de Configuración de perfiles se sugiere que pulsar intro en el cuadro de texto autocompletable para los alérgenos cuando ya se ha seleccionado una opción del contenido sugerido, añade automáticamente el alérgeno a la lista inferior.
- ✓ También se sugiere un cambio en el texto que aparece en el botón en esa misma pantalla para que indique Configurar perfil tras la creación del usuario y Guardar datos cuando se modifica un perfil ya creado con anterioridad.
- ✓ En la lista de búsquedas se sugiere un cambio de funcionalidad en el botón de Añadir favorito para que permita tanto añadir como borrar de la lista de favoritos sin necesidad de tener que cambiar de pantalla para borrar un favorito añadido de forma errónea.

Todo el feedback recibido ha sido tenido en cuenta y se han realizado las modificaciones pertinentes para incorporar todos los aspectos indicados a la aplicación final.

## 5. Conclusiones

En este apartado se explican las conclusiones que se obtienen de todo el proceso por el que ha atravesado este proyecto tanto a nivel personal como a nivel de objetivos de proyecto. Además, también se han incluido otras dos subsecciones con la intención de detallar cómo ha sido la experiencia de la gestión completa de un proyecto, cómo se ha adaptado la planificación a los posibles contratiempos que hayan podido suceder y cuáles son las líneas de trabajo futuro que ya se adelantaron en la descripción de los objetivos del proyecto.

### 5.1. Conclusiones personales

Una vez se ha llegado a la etapa final del proyecto, es posible afirmar que se han cumplido los objetivos personales planteados por la alumna. El principal objetivo que se había propuesto era el desarrollo de una aplicación móvil para plataformas Android que fuese funcional partiendo de un conocimiento nulo de desarrollo en nativo para Android por parte de la alumna. Pese a contar con experiencia previa tanto en la gestión de proyectos como en el desarrollo de aplicaciones webs y móviles utilizando diferentes frameworks no nativos durante su trayectoria profesional, el principal reto consistía en la adaptación a todas las exigencias que requiere el desarrollo nativo para Android y conseguir un producto estable, funcional y con una apariencia lo suficiente atractiva para poder ser considerada comercial y todo ello dentro del reducido espacio de tiempo en el que se enmarca este proyecto.

El proceso no ha sido fácil puesto que ha requerido una gran labor de adaptación al nuevo entorno de desarrollo y a herramientas desconocidas hasta el momento. Desde el punto de vista académico, se ha podido poner en práctica multitud de conocimientos adquiridos durante todo el grado como es la realización de un diseño atractivo desde la definición de objetivos y planificación de todo el ciclo del proyecto, la creación de los sketches y el prototipo, las evaluaciones con usuarios, la preparación de los contextos, escenarios y casos de uso y todo el aprendizaje en el plano técnico. Hasta el momento, por ejemplo, la estudiante solo había trabajado con Firebase para el envío de notificaciones desde un framework basado en JavaScript. Durante la realización de este proyecto, se ha podido trabajar en el desarrollo nativo mediante la utilización de Android Studio y Espresso, ampliar los conocimientos en Firebase a su uso para autenticación, gestión de base de datos, hosting web y envío de notificaciones desde la propia aplicación, utilizar bases internas del dispositivo con SQLite y profundizar en el conocimiento de las características que proporciona Material Design.

### 5.2. Conclusiones objetivas del proyecto

Al igual que lo comentado en el apartado anterior, se ha conseguido cumplir la totalidad de los objetivos marcados al inicio de este proyecto. Se planteó la creación de una aplicación para escanear códigos de barras que haga más accesible la información que contiene siendo fácil de utilizar y mostrando toda la información de una forma clara. Además, esa aplicación debería permitir el envío de notificaciones cuando se detecten cambios en la composición de los productos favoritos, funcionalidad compleja que, tras la comparativa realizada entre las diferentes aplicaciones existentes en el mercado, aportaba un gran valor añadido.

Debido a la naturaleza del proyecto y a las inquietudes personales de la estudiante, se optó por la creación de la aplicación como nativa para Android utilizando Android Studio para el desarrollo, Firebase para la gestión de base de datos, usuarios y notificaciones, SQLite para la base de datos local del dispositivo que almacena los datos comunes que necesita el servicio de recuperación de información de producto de Open Food Facts.

Antes de llegar a la implementación y una vez realizada la definición de los objetivos, se realizó el diseño de la aplicación a partir del uso de las diferentes herramientas que pone a disposición de la alumna los conocimientos adquiridos acerca del diseño centrado en el usuario para llegar a la realización de un prototipo de alta fidelidad que fue evaluado de forma iterativa por diferentes usuarios potenciales, lo que llevó a la realización de diferentes mejoras sobre su apariencia final. Luego, en un plano ya meramente técnico, se definieron los casos de uso y las estructuras de entidades y sus relaciones y el diseño de la base de datos.

A partir de ese momento, comienza la fase de implementación en la que, si ya en las fases anteriores se había invertido un gran número de horas, se concentra el grueso del tiempo empleado en la realización de este proyecto. Se adopta un enfoque incremental, para ir desarrollando las funcionalidades de forma progresiva y poder ir haciendo pruebas aisladas de todas ellas para garantizar su funcionamiento. El mayor desafío en este punto consistió, además de lograr implementar todas las funcionalidades definidas, en conseguir durante el desarrollo el diseño propuesto en el prototipo y que además fuera responsive en diferentes tamaños de dispositivo. Una vez terminado el desarrollo y las pruebas incrementales, se realizó una batería de pruebas extensa tanto por la estudiante como por usuarios de su entorno cercano con diferentes habilidades técnicas. A raíz de estas pruebas y del feedback recibido, se desarrollan diferentes mejoras sobre la primera versión que se incorporan a la última versión de la aplicación.

### 5.3. Análisis de la gestión del proyecto y planificación actualizada

Durante la primera fase del proyecto se define una planificación temporal con las diferentes tareas que requiere el proyecto en la que se establecen las horas dedicadas a cada una de dichas tareas. Como en toda planificación, se trabaja con horas estimadas que, durante el curso del proyecto se ven alteradas, especialmente durante la fase de implementación. Gracias a la experiencia previa de la estudiante tanto en la planificación de proyectos como en el sector de la informática y el desarrollo, las fechas de fin de cada una de las entregas no se ven alteradas y es posible entregar en tiempo y forma pero sí hay modificaciones en el número de horas dedicadas a la realización del proyecto, causado en gran parte por el alto número de funcionalidades planteadas y la gran exigencia personal en cuanto a los resultados de la alumna. También, durante el proyecto surgen distintos inconvenientes tanto laborales como personales, pero aumentando la dedicación se consigue que estos no tengan impacto sobre el resultado final. A continuación se muestra una comparativa sin entrar en gran detalle dentro de cada fase entre la planificación estimada y la real:

	PLANIFICACIÓN ESTIMADA				PLANIFICACIÓN REAL			
	Horas	Carga	Inicio	Fin	Horas	Carga	Inicio	Fin
<b>Plan de trabajo</b>	<b>42</b>	<b>13 %</b>	<b>16/09/2020</b>	<b>30/09/2020</b>	<b>30</b>	<b>8 %</b>	<b>16/09/2020</b>	<b>30/09/2020</b>
<b>Diseño</b>	<b>84</b>	<b>26 %</b>	<b>01/10/2020</b>	<b>28/10/2020</b>	<b>94</b>	<b>26 %</b>	<b>01/10/2020</b>	<b>28/10/2020</b>
<b>Implementación</b>	<b>126</b>	<b>40 %</b>	<b>29/10/2020</b>	<b>09/12/2020</b>	<b>165</b>	<b>46 %</b>	<b>29/10/2020</b>	<b>09/12/2020</b>
Implementación	95	30 %	29/10/2020	01/12/2020	120	34 %	29/10/2020	26/11/2020
Testing	31	10 %	27/11/2020	09/12/2020	45	13 %	20/11/2020	09/12/2020
<b>Entrega final</b>	<b>68</b>	<b>21 %</b>	<b>10/12/2020</b>	<b>01/01/2021</b>	<b>72</b>	<b>20 %</b>	<b>10/12/2020</b>	<b>01/01/2021</b>
<b>Totales</b>	<b>320</b>	<b>100%</b>	<b>16/09/2020</b>	<b>01/01/2021</b>	<b>362</b>	<b>100%</b>	<b>16/09/2020</b>	<b>01/01/2021</b>

Figura 98: Comparativa entre la planificación estimada y la real

Tal como se adelantó, se puede observar que existen cambios en el número de horas dedicadas a cada tarea, siendo en la fase de planificación un poco menor la dedicación a la planteada en un inicio y aumentándola en las siguientes fases del proyecto. También, la mayor cantidad de horas invertidas en el desarrollo hace que se puedan comenzar antes las pruebas integradas de la aplicación y poder dedicar más tiempo del planificado en un primer momento.

Para ayudar en el proceso de gestión del proyecto y que no se quedase sin atender ninguna tarea necesaria, se ha creado un tablero en Trello (Trello, 2020) en el que se incluyeron todas las tareas y se fueron actualizando sus estados a medida que se iban avanzando:

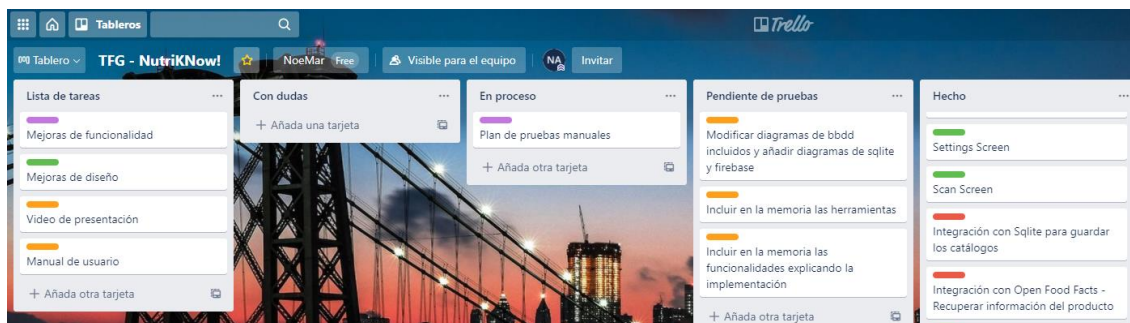


Figura 99: Captura de pantalla del tablero de Trello

En cuanto a la metodología elegida, ha sido adecuada porque se trata de un proyecto realizado por una única persona. En caso de haber sido más los involucrados en el proceso, la iteratividad que se ha seguido para llegar a los mejores resultados posibles tanto para el diseño como para el desarrollo y las pruebas podrían haber conllevado algún tipo de problema que se podría evitar eligiendo una metodología más ágil.

#### 5.4. Líneas de trabajo a futuro

La principal línea de trabajo a futuro que ha quedado sin explorar es, como ya se adelantó en la definición de los objetivos, la posibilidad de incluir nuevos productos escaneados no existentes en el banco de datos utilizado. Como se ha mencionado en varias ocasiones, se utiliza un conjunto de servicios open source que son compartidos por la comunidad tanto en el desarrollo como en la alimentación de los datos. En caso de poder abrir esa funcionalidad en esta aplicación, sería posible ir completando ese banco de datos que utiliza el servicio y que cada vez sea más útil para todas las personas que hacen uso de él, tanto de forma directa como por medio de diferentes aplicaciones.

Otra funcionalidad que ha quedado sin implementarse es el desarrollo de la lectura de etiquetas, haciendo uso de lectores de OCR para completar la información de los diferentes productos. También, pese a que en las pruebas realizadas con diferentes dispositivos no se ha encontrado el problema, sí puede suceder que la cámara del dispositivo utilizado para el escaneo no tenga la suficiente definición como para poder obtener una imagen de calidad en la que se pueda detectar el código de barras. Para ello, podría implementarse una ayuda que permita introducir de forma manual el código de barras en cuestión y realizar la obtención de la información como se hace con el escaneo directo.

En diversos puntos de este documento también se ha hablado de la posibilidad de incluir una pantalla de ayuda que guíe al usuario en las funcionalidades disponibles y, también, que vaya aportando diversos consejos nutricionales y de salud. Por último, podría ser interesante añadir la funcionalidad de compartir en redes sociales la información correspondiente a productos favoritos o al uso de la aplicación en general.

## 6. Glosario

- ✓ **Activity:** cada una de las pantallas con las que un usuario puede interactuar en una aplicación Android.
- ✓ **Android:** sistema operativo desarrollado por Google y basado en Linux utilizado en dispositivos móviles.
- ✓ **API:** conjunto de métodos proporcionados por una organización o aplicación para su uso por terceros a través de una dirección pública o de acceso restringido.
- ✓ **DCU:** enfoque de diseño cuyo proceso está dirigido por información sobre las personas que van a hacer uso del producto.
- ✓ **Endpoint:** interfaz de comunicación expuesta a terceros que da acceso a los métodos o funcionalidades que componen una determinada API.
- ✓ **Fragment:** comportamiento o parte de la interfaz que se engloba dentro de una activity concreta.
- ✓ **Framework:** conjunto de conceptos y librerías agrupados por su semántica para resolver un determinado tipo de problema de similar magnitud.
- ✓ **HTTP:** protocolo de comunicación que permite las transferencias de información en la World Wide Web.
- ✓ **IDE:** entorno de desarrollo integrado que presta todos los servicios y funcionalidades necesarios para la implementación de software.
- ✓ **JSON:** formato de texto utilizado especialmente en las comunicaciones REST de forma que facilita su comprensión y estructura los datos compartidos.
- ✓ **Listener:** objeto que escucha para detectar la ocurrencia de un determinado evento y hacer que el sistema reaccione de una forma particular.
- ✓ **MVC:** patrón de desarrollo de software que separa la diferente lógica de la aplicación en capas dependiendo del nivel de abstracción empleado.
- ✓ **Notificación push:** mensajes enviados vía Push Notification Service a una aplicación móvil o a un navegador.
- ✓ **Nova group:** clasificación que asigna un determinado grupo a los productos alimenticios basándose en cuánto han sido procesados.
- ✓ **Nutriscore:** etiquetado nutricional frontal para productos procesados envasados.
- ✓ **Open source:** proceso que se utiliza en los proyectos de la comunidad para desarrollar sistemas de software bajo una licencia que permite que cualquier persona pueda ver o modificar el código fuente.
- ✓ **Responsive:** técnica de diseño web que busca la correcta visualización de una misma página en distintos dispositivos
- ✓ **REST:** arquitectura utilizada en comunicaciones que proporciona diferentes métodos para compartir objetos en la web.
- ✓ **SDK:** conjunto de herramientas proporcionadas para realizar el desarrollo de software para una plataforma concreta.



## 7. Bibliografía

- ✓ Aecosan.msssi.gob.es. 2020. Aesan - Agencia Española De Seguridad Alimentaria Y Nutrición. Consultada el 21 de septiembre de 2020 en:  
[http://www.aecosan.msssi.gob.es/AECOSAN/web/para\\_el\\_consumidor/ampliacion/alergias.htm](http://www.aecosan.msssi.gob.es/AECOSAN/web/para_el_consumidor/ampliacion/alergias.htm)
- ✓ Android Asset Studio, 2020. Notification icon generator. Romannurik.github.io. Consultada el 20 de noviembre en:  
[https://romannurik.github.io/AndroidAssetStudio/icons-notification.html#source.space.trim=1&source.space.pad=0&name=ic\\_stat\\_example](https://romannurik.github.io/AndroidAssetStudio/icons-notification.html#source.space.trim=1&source.space.pad=0&name=ic_stat_example)
- ✓ Android Developers, 2020. Download Android Studio and SDK tools | Android Studio. Consultada el 20 de noviembre en:  
<https://developer.android.com/studio?hl=es>
- ✓ Atlantida.net. 2020. Factores Que Pueden Promover O Desencadenar Intolerancias Alimentarias | Atlàntida. Consultada el 21 de septiembre de 2020 en:  
<https://www.atlantida.net/factores-que-pueden-promover-o-desencadenar-intolerancias-alimentarias/?lang=es>
- ✓ Bitbucket, 2020. Bitbucket | The Git solution for professional teams. Consultada el 20 de noviembre en:  
<https://bitbucket.org/product/>
- ✓ Cascada, E. and Domínguez, P., 2020. En Qué Consiste El Modelo En Cascada. Consultada el 25 de septiembre de 2020 en:  
<https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada>
- ✓ Codebeautify, 2020. Free Online Tools For Developers - to Beautify, Validate, Minify, Analyse, Convert JSON, XML, JavaScript, CSS, HTML, Excel. Codebeautify.org. Consultada el 20 de noviembre en:  
<https://codebeautify.org/>
- ✓ dbdiagram.io - Database Relationship Diagrams Design Tool. 2020. Dbdiagram.io. (2020). Consultada el 20 de octubre de 2020 en:  
<https://dbdiagram.io/>
- ✓ DbVisualizer, 2020. The Universal Database Tool. Dbvis.com. Consultada el 20 de noviembre en:  
<https://www.dbvis.com/>
- ✓ EasyAppIcon, 2020. Create Mobile App Icon. Easyappicon.com. Consultada el 20 de noviembre en:  
<https://easyappicon.com/>
- ✓ E Etiquetadocuentamuchito.aecosan.msssi.gob.es. 2020. En El Etiquetado Alimentario Hay Algunas Menciones Que Son Obligatorias. Consultada el 24 de septiembre de 2020 en:  
<http://eetiquetadocuentamuchito.aecosan.msssi.gob.es/obligatoria.html>
- ✓ Espresso, 2020. Android Developers.. Consultada el 24 de noviembre en:  
<https://developer.android.com/training/testing/espresso>
- ✓ Excalidraw. 2020. Consultada el 11 de octubre de 2020 en:  
<https://excalidraw.com/>
- ✓ Firebase. 2020. Firebase. Consultada el 25 de septiembre de 2020 en:  
<https://firebase.google.com/>
- ✓ Flaticon. 2020. Flaticon. Consultada el 26 de septiembre de 2020 en:  
<https://www.flaticon.com/search>

- ✓ Font Awesome, 2020. Fontawesome.com. Consultada el 20 de noviembre en:  
<https://fontawesome.com/>
- ✓ Freepik | Recursos gráficos para todos. Freepik. 2020. Consultada el 11 de octubre de 2020 en:  
<https://www.freepik.es/home>
- ✓ Free prototyping tool for web & mobile apps - Justinmind. 2020. Consultada el 13 de octubre de 2020 en:  
<https://www.justinmind.com/>
- ✓ Garreta Domingo, Muriel y Mor Pera, Enric. 2020. UOC. Diseño centrado en el usuario. Consultada el 7 de octubre de 2020 en:  
[https://www.exabyteinformatica.com/uoc/Informatica/Interaccion\\_persona\\_ordenador/Interaccion\\_persona\\_ordenador\\_\(Modulo\\_3\).pdf](https://www.exabyteinformatica.com/uoc/Informatica/Interaccion_persona_ordenador/Interaccion_persona_ordenador_(Modulo_3).pdf)
- ✓ Generated Photos | Unique, worry-free model photos. 2020. Consultada el 11 de octubre de 2020 en:  
<https://generated.photos/>
- ✓ Git, 2020. Git-scm.com. Consultada el 20 de noviembre en:  
<https://git-scm.com/>
- ✓ Google Forms – create and analyse surveys, for free. 2020. Consultada el 8 de octubre de 2020 en:  
<https://docs.google.com/forms/u/1/?tgif=d>
- ✓ Group, I., 2020. El 86% De Los Smartphones Vendidos A Nivel Mundial Son Android. Consultada el 25 de septiembre de 2020 en:  
<https://www.itreseller.es/en-cifras/2020/05/el-86-de-los-smartphones-vendidos-a-nivel-mundial-son-android>
- ✓ JoanZapata/android-iconify, 2020. GitHub. Consultada el 20 de noviembre en:  
<https://github.com/JoanZapata/android-iconify>
- ✓ MagicDraw. 2020. Nomagic.com. Consultada el 20 de octubre de 2020 en:  
<https://www.nomagic.com/products/magicdraw>
- ✓ Material Design. 2020. Consultada el 13 de octubre de 2020 en:  
<https://material.io/design>
- ✓ Open Food Facts API Documentation, 2020. Consultada el 20 de noviembre en:  
<https://documenter.getpostman.com/view/8470508/SVtN3Wzy#5ee4f51d-be38-4efd-847b-181807c27498>
- ✓ Play.google.com. 2020. Consultada el 25 de septiembre de 2020 en:  
<https://play.google.com/store>
- ✓ Prevensystem.com. 2020. Principales Alérgenos Alimentarios. Consultada el 26 de septiembre de 2020 en:  
<https://www.prevensystem.com/internacional/987/noticia-principales-alergenos-alimentarios.html>
- ✓ Psicologiyamente.com. 2020. Las 10 Mejores Apps De Nutrición. Consultada el 24 de septiembre de 2020 en:  
<https://psicologiyamente.com/vida/mejores-apps-nutricion>
- ✓ Ramírez, I., 2020. Xatakandroid.com. Android 10 está en menos de 1 de cada 10 dispositivos, según los últimos datos de distribución de versiones. Consultada el 20 de noviembre en:  
<https://www.xatakandroid.com/sistema-operativo/android-10-esta-1-cada-10-dispositivos-ultimos-datos-distribucion-versiones>

- ✓ Retrofit, 2020. Square.github.io. Consultada el 20 de noviembre en:  
<https://square.github.io/retrofit/>
- ✓ Singlutenismo. 2020. Qué Es El Gluten Y Qué Es La Celiaquía - Singlutenismo. Consultada el 22 de septiembre de 2020 en:  
<https://www.singlutenismo.com/que-es-el-gluten>
- ✓ SoapUI, 2020. The World's Most Popular API Testing Tool | Soapui.org. Consultada el 20 de noviembre en:  
<https://www.soapui.org/>
- ✓ Sqlite.org, 2020. SQLite Home Page. Consultada el 20 de noviembre en:  
<https://www.sqlite.org/index.html>
- ✓ Trello. 2020. Trello.com. Consultada el 30 de noviembre en:  
<https://trello.com/>
- ✓ User Persona Template and Examples | Xtensio. 2020. Consultada el 10 de octubre de 2020 en:  
<https://xtensio.com/user-persona/>
- ✓ Venngage. 2020. Creador De Infografías Gratuito - Venngage. Consultada el 26 de septiembre de 2020 en:  
<https://es.venngage.com/>
- ✓ Wiki.openfoodfacts.org. 2020. Open Food Facts EN. Consultada el 25 de septiembre de 2020 en:  
[https://wiki.openfoodfacts.org/Main\\_Page](https://wiki.openfoodfacts.org/Main_Page)
- ✓ X-Wiki UOC. 2013. Métodos del diseño centrado en el usuario para dispositivos móviles. Consultada el 5 de octubre de 2020 en:  
<http://cv.uoc.edu/webapps/xwiki/wiki/matm1202es/view/Main/6.1+Indagaci%C3%B3+%5Ban%C3%A0lisi%5D>
- ✓ YML. 2020. Native VS Hybrid Mobile Apps - Here's How To Choose. Consultada el 25 de septiembre de 2020 en:  
<https://ymedialabs.com/hybrid-vs-native-mobile-apps-the-answer-is-clear>

## 8. Anexos

Este apartado se dedica a los apartados extensos y autocontenidos que terminan de dar sentido a esta memoria. Son documentos que se entregarán en documentos separados para enfatizar su sentido de documento adicional.

### 8.1. Manual de usuario

Este apartado engloba las instrucciones de uso y el contexto de la aplicación para que tanto los usuarios con altas habilidades tecnológicas como aquellos que no las dominan tanto dispongan de los detalles necesarios para poder conocer el propósito de la aplicación y ser capaces de utilizarla exitosamente sin dificultades.

### 8.2. Manual de instalación

Este apartado tiene una estructura muy simple y resume las instrucciones básicas para, a partir del proyecto Android, conseguir configurar e instalar la aplicación en un dispositivo móvil o emulador.