Universitat Oberta de Catalunya

uoc.edu

Diseño e implementación de la aplicación móvil Regala

Montse Blanco de Frutos
Máster Universitario en Ingeniería Informática
Trabajo Final de Máster - Desarrollo de aplicaciones sobre dispositivos móvil

Jordi Ceballos Villach
Jordi Almirall López
Robert Clarisó Viladrosa

Enero 2020

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

FICHA DE TRABAJO FINAL

Título del trabajo:	*Diseño e implementación de la aplicación móvil Regala*
Nombre del autor:	*Montse Blanco de Frutos*
Nombre del consultor/a:	*Jordi Ceballos Villach* *Jordi Almirall López*
Nombre del PRA:	*Robert Clarisó Viladrosa*
Fecha de entrega (mm/aaaa):	*01/2020*
Titulación:	*Máster Universitario en Ingeniería Informática*
Área del Trabajo Final:	*Desarrollo de aplicaciones sobre dispositivos móvil*
Idioma del trabajo:	*Español*
Palabras clave	*App, Android, Firebase*

Resumen del Trabajo (máximo 250 palabras):

Hay una realidad que viven bastantes personas durante todo el año y en especial cuando llegan fechas claves como cumpleaños o Navidad. Normalmente somos conscientes de cuando tenemos que hacer un regalo pero en muchas ocasiones acabamos comprando cualquier cosa a última hora. ¿Pero y si además nos tenemos que organizar con otras personas? La situación se complica aún más. De ahí surge la idea de una aplicación para evitar que la gestión y la compra de los regalos no se acabe convirtiendo en una agonía. El propósito de este trabajo es diseñar y desarrollar una aplicación móvil Android llamada Regala. Esta permite planificar los regalos para cualquier tipo de evento, dirigidos a una persona en concreto, en una fecha específica, pudiendo organizar de forma individual como en grupo.

La metodología elegida entre las distintas opciones es la del ciclo de vida en cascada aplicando a su vez el Diseño Centrado en el Usuario para identificar y dar solución a las necesidades y expectativas de los usuarios.

La aplicación ha sido desarrollada de forma nativa con Android Studio y el SDK de Android, utilizando Firebase como servidor backend para la gestión de los usuarios, la base de datos y el almacenamiento de imágenes. También se ha usado el framework Glide para la carga de imágenes.

Finalmente, como producto final, se ha obtenido una aplicación móvil que cumple con los principales objetivos propuestos al comienzo del proyecto.

Abstract (in English, 250 words or less):

There is a reality that many people live throughout the year and especially when key dates such as birthdays or Christmas arrive. Normally we are aware of when we have to make a gift but on may occasions we end up buying anything at the last minute. What if we also have to organize with other people? The situation is further complicated. That's where the idea come of an application to prevent the management and purchase of gifts does not end up becoming an agony. The purpose of this project is to design and develop and Android mobile application called Regala. This allows you to plan gifts for any type of event, addressed to a specific person, on a specific date, being able to organize individually or in groups.

The methodology chosen amog the different options is the Waterfall model, also applying the User Centered Design to identify and solve the needs and expectations of the users.

The application has been developed natively with Android Studio and the Android SDK, using Firebase as a backend server for user management, database and image storage. Also, the Glide framework has been used for load images.

Finally, as a final product, a mobile application has been obtained that fulfill the main goals proposed at the beginning of this project.

Índice

1. Plan de trabajo **1**
1.1. Contexto y justificación del Trabajo....1
1.1.1. Estudio de mercado....2
1.2. Objetivos del Trabajo....4
1.3. Enfoque y método seguido....6
1.4. Planificación del Trabajo....7
1.4.1. Recursos necesarios....7
1.4.2. Definición de tareas....8
1.4.3. Planificación temporal....10
1.4.4. Riesgos del proyecto....12
1.5. Breve sumario de productos obtenidos....13
1.6. Breve descripción de los otros capítulos de la memoria....13
2. Diseño y Arquitectura **14**
2.1. Diseño Centrado en el Usuario....14
2.1.1. Usuarios y contexto de uso....14
2.1.1.1 Benchmarking....15
2.1.1.2 Entrevista en profundidad....15
2.1.1.3 Encuestas....17
2.1.1.4 Perfiles de usuario....22
2.1.2. Diseño conceptual....26
2.1.2.1 Escenarios de uso....26
2.1.2.2 Flujos de interacción....33
2.1.3. Prototipado....34
2.1.3.1 Sketches....34
2.1.3.2 Prototipos horizontales de alta fidelidad....35
2.1.4. Evaluación....39
2.1.4.1 Perfil socio-demográfico de los usuarios....39
2.1.4.2 Preguntas sobre el usuario (pre-test)....40
2.1.4.3 Tareas a realizar....40
2.1.4.4 Preguntas referentes a las tareas (post-test)....41
2.1.4.5 Test con un usuario, análisis y mejoras....41
2.2. Diseño Técnico....43
2.2.1. Definición de algunos casos de uso....43
2.2.2. Diseño de la arquitectura....52
2.2.2.1 Diseño de la base de datos....52
2.2.2.2 Diseño de las entidades y clases....53
2.2.2.3 Arquitectura del sistema....53
3. Implementación **55**
3.1. Herramientas utilizadas....55
3.1.1. Android....55
3.1.2. Android Studio....56
3.1.3. Java y XML....56
3.1.4. Firebase....56
3.1.5. Glide....57
3.2. Detalles del desarrollo y el diseño....57
3.2.1. Estructura del proyecto....57
3.2.2. Diseño de las pantallas....59

3.2.3. Cambios en el diseño y funcionalidades....60
3.2.4. Menú lateral con DrawerLayout y Navigation Component....61
3.2.5. Listas con RecyclerView y Adapter....62
3.2.6. Paso de parámetros entre Activities y Fragments....63
3.2.7. Gestión de los usuarios con Firebase Authentication....63
3.2.8. Estructura de los datos en Cloud Firestore....64
3.2.9. Almacenaje y recuperación de datos con Cloud Firestore....66
3.2.10. Gestión de permisos....68
3.2.11. Obtención de imágenes....68
3.2.12. Almacenamiento de imágenes en Cloud Storage....69
3.2.13. Descarga de imágenes con Glide y FirebaseUI....70
3.3. Pruebas....70
3.3.1. Información sobre las pruebas....70
3.3.2. Pruebas realizadas....71
3.4. Funcionalidades de la aplicación....73
3.4.1. Gestión de usuarios....73
3.4.2. Gestión de las planificaciones....74
3.4.3. Gestión de la lista de deseos....77
3.4.4. Gestión de los contactos....78
4. Conclusiones 79
4.1. Líneas de trabajo futuras....80
5. Glosario 81
6. Bibliografía 82
7. Anexos 84

1. Plan de trabajo

1.1. Contexto y justificación del Trabajo

Hoy en día es difícil encontrar a alguna persona entre los 18 y los 65 años que no tenga un Smartphone y no utilice aplicaciones móviles en su día a día. Si algo brinda la tecnología, es que hace la vida de uno mejor. En general la sociedad se ha acostumbrado a usar cualquier aparato o producto tecnológico para desempeñar casi cualquier cosa. Existen todo tipo de aplicaciones que permiten a las personas resolver un problema, cubrir sus necesidades, simplificar sus tareas y/o satisfacer sus deseos. Dentro de este contexto es necesario plantear una herramienta para solucionar problemas bastante específicos y facilitar la vida del usuario final.

Hay una realidad que viven muchas personas durante todo el año y en especial cuando llegan fechas claves como cumpleaños y Navidad. Normalmente las personas son conscientes que se acerca ese momento, lo más probable es que quieran comprar los regalos pero la verdad es que van pasando los días y en el último momento se acaba corriendo a la tienda o recurriendo a cualquier cosa de internet. Al final parece que se trata más de cumplir con el deber en vez de una muestra de afecto o agradecimiento. Hay que reconocer que el tema de los regalos no es tan sencillo, tener el obsequio perfecto en la fecha indicada se puede llegar a convertir en una misión imposible aunque en la mayor parte de casos se trata más de un problema de mala planificación.

Por tanto, en este Trabajo Final de Máster se busca diseñar e implementar una herramienta que ayude a evitar que la compra de los regalos se acabe convirtiendo en un auténtico calvario. En concreto, una aplicación para organizar los regalos de la forma más cómoda posible y controlar todos los detalles. La idea gira en torno a la acción de añadir una nueva planificación de regalos para un tipo de evento u ocasión, dirigido a una persona, para una fecha en concreto, un presupuesto máximo y los obsequios con su respectivo estado ya sea una idea inicial, este pendiente de comprar, etc. Posteriormente, en esa planificación se podrán añadir o borrar los regalos entre otras acciones. De este modo los usuarios tendrán la posibilidad de registrar ideas a medida que van surgiendo, anotar todas aquellas necesidades de compra o deseos para alguien o los ya comprados. Y una vez obsequiados entonces marcarlos como finalizados. Todo esto permite saber cuántos regalos se han realizado, cuánto han costado, cuándo se han regalado y a quién.

De esta forma, siguiendo unos pasos, resultará mucho más sencillo planificar los regalos, se podrán tomar mejores decisiones y hacer buenas compras partiendo de un presupuesto prefijado. Y todo esto desde el propio dispositivo móvil para organizarse en cualquier momento y disponer de los datos en todo momento.

¿Y si se desea coordinar la compra de regalos con otras personas? La cosa se complica y más si no se organiza de alguna manera. La típica situación es acabar en un grupo de whatsapp con cientos de mensajes sin leer, perdiendo el hilo de la conversación y con ideas dispersas. Y una vez comprado el presente empiezan las discusiones sobre quien ha pagado que y quien debe a quien. Pero los desarrollos tecnológicos reducen las distancias entre la gente y facilitan el ponerse de acuerdo entre todos. Utilizando la misma aplicación planteada, si se quiere coordinar un regalo con otras personas, se podrán ir agregando participantes para organizar las compras en común. Allí el grupo verá las ideas o aportará de nuevas, intercambiarán opiniones, controlarán el presupuesto y el gasto individual, etc.

Este enfoque colaborativo pretende a la vez tener un estilo similar a una red social para que los usuarios puedan interactuar entre ellos, intercambiar ideas y sincronizar datos de forma segura entre dispositivos. Como toda red social se encontrará compuesta por perfiles de usuarios con una representación de cada uno, conexiones sociales entre ellos y la posibilidad de mostrar una lista de deseos con los productos que le hayan despertado interés para compartirlos con las otras personas a las que les puedan interesar.

1.1.1. Estudio de mercado

Una vez contextualizada la idea del proyecto, el siguiente paso es realizar un breve estudio de las aplicaciones con características similares y que estén disponibles en la tienda de Android, Google Play. Hay una gran cantidad de aplicaciones relacionadas con la gestión, organización y listado de regalos así que se ha reducido a las que tienen mejor valoración (3,9 como mínimo) o características similares a la propuesta.

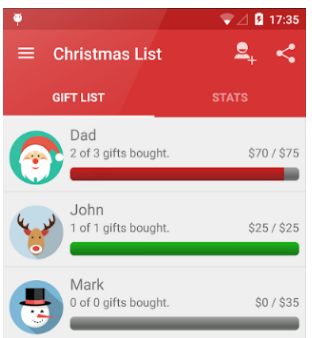


Figura 1: Detalle de la app Christmas Gift List
Fuente: engApps (2018)

- Christmas Gift List de engApps

Para realizar un seguimiento de las ideas de regalo de Navidad, establecer un presupuesto de regalos para cada persona y marcar el regalo como comprado y/o envuelto. Además, cada año nuevo la lista se archiva automáticamente y se guarda en la app.

Destacar en especial el detalle del gasto respecto al presupuesto, la utilización de los colores es simple y práctico. Y permite hacer una copia de seguridad generando un archivo en la nube usando Google Drive.

Algunos inconvenientes es que para añadir regalos debe existir previamente la persona en la lista. Creo que se podrían ahorrar pasos creando la persona y los regalos a la vez o poder agregar un regalo sin la necesidad de entrar dentro de la persona. Está muy escondida la opción de borrar a una persona del listado. Y tiene publicidad aunque puede eliminarse comprando la versión Premium.

- Regalos Presupuesto de Navidad de Axl Softs

Permite anotar todas las ideas de regalos y el presupuesto poco a poco conforme se realizan las compras de Navidad. En este caso el presupuesto se controla a nivel global detallando el gasto en comprados, en previstos y el saldo disponible.

Destacable el poder seguir de un solo vistazo los regalos para cada persona, el avance en las compras navideñas y el presupuesto. Además, cuenta con un botón para añadir regalo, no hace falta entrar dentro de la lista de regalos de la persona en concreto. Es realmente útil poder gestionarlo todo desde la pantalla principal y disponer de búsqueda.

En cambio, se hecha en falta no poder adjuntar una fotografía a los regalos ni un nivel de detalle más profundo como enlaces, anotaciones, etc. Tampoco permite borrar ni personas ni regalos. Tiene publicidad aunque puede eliminarse comprando la versión Premium.

Cada año se archiva la lista en curso y las antiguas pueden ser consultadas. Parece que no ofrece la posibilidad de hacer un backup en la nube.

Figura 2: Detalle de la app Regalos Presupuesto de Navidad
Fuente: Axl Softs (2016)

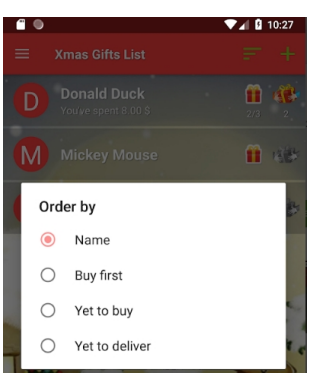


Figura 3: Detalle de la app Xmas Gifts List
Fuente: Montalto, P. (2018)

- Xmas Gifts List de Paolo Montalto

Recordatorio de todos los regalos en proceso de comprar en Navidad mediante un listado de personas junto al importe gastado y la opción de marcar el regalo como comprado y/o envuelto. Incluye un control del gasto y los regalos pendientes de comprar.

Interesante poder ordenar las personas por nombre, fecha límite (día en que debe comprar el regalo), aún no comprado, etc. aunque no permite lo mismo para los regalos. Y es práctico poder editar/eliminar personas y regalos deslizando o presionando prolongadamente un elemento.

No dispone de búsqueda ni se pueden añadir fotografías. Tiene publicidad y sólo está disponible en inglés.

Existe una versión Premium que elimina la publicidad y hace un backup de la app aunque no especifica donde se almacenan los datos.

- Christmas Gift List de Medialoha

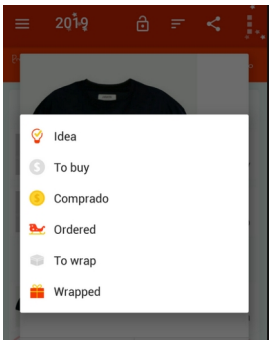


Figura 4: Detalle de la app Christmas Gift List

Fuente: Medialoha (2018)

Aplicación para ayudar a organizar las listas de Navidad y su presupuesto para pasar menos tiempo comprando regalos. Permite establecer presupuestos, planificar los regalos, realizar un seguimiento del progreso, compartir, ver las estadísticas, etc.

Es interesante los diversos filtros que ofrece dependiendo de la página en la que se encuentra. Además, en los regalos se puede indicar un estado, si es una idea, si se tiene que comprar, si ya está comprado, etc.. Para luego en una lista To-Do poder filtrar por el estado en el que se encuentran.

El problema es que está sobrecargada de opciones y pestañas, es poco intuitiva y es extraño porque algunos textos están en inglés y otros en español (aunque parecen traducidos automáticamente).

Existe una versión de pago sin publicidad con la opción de hacer un backup de los datos o restaurarlos. Dispone de estadísticas por persona, por regalos y persona y por grupo.

- Gift Tracker – Birthday and Christmas Gift Lists! De Leewrigg

Seguimiento de las ideas para regalos de Navidad, cumpleaños y otros eventos en un sólo lugar. Crear nuevos eventos y rastrear cualquier idea que se tenga en mente, así como los regalos que ya se compraron y envolvieron.

Sin duda la gracia de esta app es poder crear eventos personalizados durante todo el año. También es muy útil ver los días que faltan para que venza un evento y disponer de una lista completa de todos los regalos que aún están pendientes de comprar y de los eventos pasados. Se pueden agregar más personas a las que comprar regalos en un evento. Y resulta práctico cambiar el estado del regalo sin la necesidad de apretar en modificar.

Aunque en algunos aspectos es bastante completa en otros es todo lo contrario. No tiene ninguna opción para hacer una copia de seguridad de los datos. Tampoco hay ninguna posibilidad de filtrar ni buscar, no se pueden añadir fotografías y sólo está disponible en inglés. Y encuentro que hay excesivas pantallas, podrían estar los regalos más a la vista aunque pertenezcan a una persona y evento en concreto.

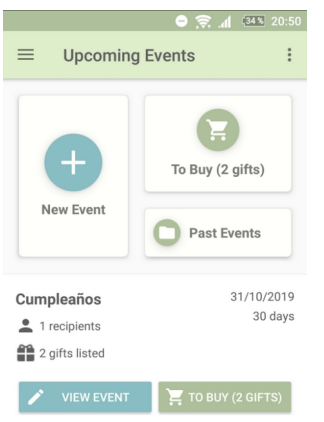


Figura 5: Detalle de la app Gift Tracker

Fuente: Leewrigg (2019)

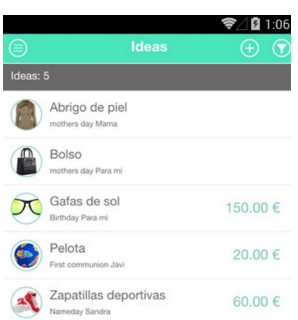


Figura 6: Detalle de la app lastpresent

Fuente: lastpresent (2016)

- lastpresent: regalos y compras de lastpresent

Es como una agenda de datos que ayuda a organizar los regalos y compras de cualquier tipo de evento, crear listas, anotar ideas y compartirlas. Permite saber cuantos regalos o compras se han realizado, cuanto han costado, cuando se han regalado y a quién.

La app se centra en tres grandes grupos: las ideas para regalar o para que regalen a uno mismo, los regalos entregados pudiendo gestionar por año, celebraciones y contactos y los regalos recibidos también por año, celebraciones y contactos.

Haciendo alguna prueba no se han cargado las fotografías ni desde álbum ni fotografiando al momento. Sólo se pueden hacer copias de seguridad en la versión de pago. Otro gran inconveniente es que se pierde un poco el enfoque del menú cuando se entran a las diferentes pantallas, llega un momento que se pierde la trazabilidad. Y los regalos no disponen de estados (como idea, comprados, envueltos, etc.).

Una vez finalizado este breve estudio de mercado se puede observar que la gran mayoría de las aplicaciones están destinadas a planificar regalos sólo para Navidad sin contemplar otro tipo de eventos, algunas son poco intuitivas y otras ni siquiera están planteadas como un planificador en sí. Y probablemente muchas no cuenten con un diseño pensado en el usuario y su usabilidad. En general, todas tienen como elemento central un listado para los regalos aunque cada una los presenta de una forma distinta, con un estilo diferente y con un punto de vista en concreto.

También se llega a la conclusión que hay muchas aplicaciones para gestionar regalos de forma individual, es decir, en el que el único participante del obsequio es uno mismo. En cambio, ninguna contempla la posibilidad de planificar los

regalos en grupo, que cubra la necesidad de la colaboración de diversas personas en un obsequio común pudiendo aportar ideas, control de gastos o comunicación entre los participantes. Tampoco dan la posibilidad de registrarse en la aplicación para tener siempre disponibles los datos u ofrecer un perfil propio en el que, por ejemplo, mostrar una lista de deseos para dar ideas a los contactos agregados. Al estar registrados los usuarios en la app también será mucho más sencillo añadirlos en los regalos colaborativos o planificar un regalo para ese usuario. Y, por último, ninguna de las aplicaciones cuenta con notificación de alertas para avisar de la aproximación de la fecha límite para hacer los regalos en un evento.

La conclusión es que ninguna de las aplicaciones analizadas cubre todas las funcionalidades que se plantean desarrollar en este proyecto.

1.2. Objetivos del Trabajo

El objetivo principal de este Trabajo Final de Máster es la construcción de una aplicación móvil que permita planificar y gestionar de una forma clara, rápida y sencilla los regalos para cualquier tipo de evento tanto en modo individual como en grupo. Los regalos grupales permitirán a un conjunto de usuarios planificar los regalos comunes para una persona en concreto pudiendo aportar ideas, controlar el aporte económico de cada participante, comunicación entre los participantes, etc. La app ha de poder crear perfiles de nuevos usuarios así como iniciar sesión a usuarios que ya estén registrados. Y proporcionar un sistema muy visual para controlar de un solo vistazo todos los regalos en curso por evento, personas y fechas, con la posibilidad de cambiar el estado de los mismos y un detalle del gasto respecto al presupuesto. Generar notificaciones cuando se aproximen fechas límite para la compra de los regalos. Y disponer de un historial de todos los regalos obsequiados en el pasado, en que evento, a que personas, las fechas y precios.

A continuación, se indican a alto nivel los requerimientos funcionales de la aplicación, exactamente aquellos que son funcionalidades claves requeridas por el sistema:

- **Gestión de los usuarios**
  - Registro de nuevos usuarios
  - Inicio de sesión mediante correo electrónico y una contraseña
  - Restablecer la contraseña en caso de olvido
  - Cerrar la sesión
  - Visualizar / Editar el perfil de usuario
  - Dar de baja el usuario

- **Gestión de los regalos en planificaciones**
  - Planificar regalos de forma individual o en grupo
  - Listar las planificaciones de regalos en curso o las finalizadas
  - Búsqueda de planificaciones en el listado
  - Filtrar las planificaciones del listado, por ejemplo por evento, persona, estado del regalo, etc.
  - Eliminar regalo

- **Planificación regalo individual**
  - Añadir regalos / Mostrar los regalos añadidos
  - Añadir fotografía en cada regalo / Mostrar fotografía de cada regalo si tiene
  - Activar / Desactivar notificación de fecha límite próxima
  - Control presupuesto
  - Marcar / Desmarcar regalo como realizado (u otros estados)

- **Planificación regalo en grupo**
  - Añadir participantes / Mostrar los participantes añadidos
  - Añadir regalos / Mostrar los regalos ya añadidos
  - Añadir fotografía en cada regalo / Mostrar fotografía de cada regalo si tiene
  - Activar / Desactivar notificación de fecha límite próxima
  - Marcar / Desmarcar regalo como realizado (u otros estados)
  - Controlar aporte económico por participante, quien ha pagado que, quien debe dinero, etc.
  - Permitir la comunicación entre los participantes del regalo en grupo mediante el uso de un chat (funcionalidad opcional)

- **Gestión de los contactos**
  - Visualizar los contactos agregados del usuario
  - Agregar contacto al usuario
  - Borrar contacto del usuario

- **Gestión de las notificaciones**
  - Ver las notificaciones del usuario
  - Gestionar que tipo de notificaciones no recibir

Por último, se exponen los requerimientos no funcionales de la aplicación incluyendo las plataformas y/o dispositivos a los que se dirige:

- La plataforma a la que se dirigirá la aplicación será Android con una versión mínima de Android Lollipop (5.1.1).
- La aplicación se podrá utilizar en dispositivos móviles como Smartphones y se plantea la posibilidad de adaptarla también para Tablets.
- La aplicación se implementará en el entorno de desarrollo Android Studio.
- Para desarrollar de forma nativa para Android se utilizará el lenguaje de programación java.
- La aplicación tendrá un diseño limpio e intuitivo centrado en el usuario. Y deberá ser sencilla de usar y con un tiempo de aprendizaje rápido.
- Los datos de la aplicación deberán almacenarse en un sistema, para ello se hará uso de un backend en el que se almacenarán los objetos necesarios.
- Los usuarios deberán estar conectados a internet en todo momento para la recuperación y modificación de los datos en el backend. El backend también se encargará de autentificar a los usuarios.
- El sistema tendrá que ofrecer las medidas de seguridad necesarias para evitar el acceso no autorizado a los datos del backend.
- El sistema deberá ser robusto y fiable evitando errores críticos que produzcan el cierre de la aplicación de forma inesperada.
- La aplicación sólo estará disponible en español.
- Las notificaciones que podrá recibir el usuario o los permisos que necesitará la aplicación para poder funcionar se habrán de especificar claramente y en un lenguaje común entendible por todos.
- La aplicación deberá proporcionar mensajes de error que sean informativos para que el usuario entienda que está ocurriendo.

1.3. Enfoque y método seguido

Para determinar la estrategia a seguir para llevar a cabo el trabajo se ha hecho un pequeño estudio de mercado en el apartado de Contexto y justificación del trabajo. En él se han analizado varias aplicaciones con características similares, todas disponibles en la tienda de Android, Google Play. Con el estudio se ha concluido que ninguna cubre exactamente todos los requisitos deseados en la aplicación y tampoco cuentan con la novedad de poder planificar regalos en grupo para facilitar la elección y posterior compra de esos regalos realizados en común.

Por tanto, una posible estrategia a seguir sería trabajar en base a una de esas apps adaptándola a los requerimientos necesarios y añadiendo nuevas funcionalidades. El problema es que estas aplicaciones son privadas, no hay acceso a su código fuente y llegar a negociar con las empresas a las que pertenecen alargarían considerablemente la planificación temporal de este proyecto. Además, otro punto importante es que se desconoce si estas aplicaciones se han diseñado centrándose en las necesidades de los usuarios que van a hacer uso del mismo.

Teniendo en cuenta todo lo expuesto, finalmente se toma la decisión de crear una aplicación nueva. Aunque la creación de una app desde cero supone una inversión inicial superior también tiene sus ventajas como enfocar la app desde un principio a los requerimientos escogidos, añadirle novedades, prestar atención a los comportamientos de los usuarios y diseñar siguiendo un proceso dirigido por información sobre las personas que van a hacer uso del producto.

Por otra parte, como se ha comentado en los requerimientos no funcionales de la aplicación, se ha decidido desarrollar la app de forma nativa utilizando la plataforma Android. Es cierto que la construcción de una aplicación híbrida suena atractiva ya que permitiría abarcar los principales sistemas operativos para móviles pero emular ciertas características o componentes de las apps nativas podría repercutir en el margen de tiempo disponible. Y de todas formas, no se podrían acceder a todas las funcionalidades del dispositivo móvil y puede ser difícil conseguir efectos sofisticados en la interfaz de usuario (Prieto, Ramírez, Morillo y Domingo, 2011). Así que la elección de Android se debe principalmente a su mayor cuota de mercado y los bajos costes económicos del desarrollo en comparación con otras plataformas. Asimismo, la implementación de una aplicación nativa ofrece la ventaja de poder acceder a todas las características de maquinaria del dispositivo y proporciona un mejor rendimiento.

A continuación, se ha contemplado la utilización de la aplicación en dispositivos móviles como Smartphones y seguramente también para Tablets. Como uno de los requerimientos de la aplicación es proporcionar la opción de añadir fotografías en la planificación de un regalo entonces es necesario disponer de acceso al dispositivo. Ese detalle ha condicionado la elección del tipo de dispositivos. También se ha decidido que la aplicación almacene los datos del usuario en la nube, es decir, no se guardarán dichos datos en la propia memoria del dispositivos. El motivo es por facilitar el acceso a los datos guardados desde distintos dispositivos y en caso de borrarse la aplicación o renovar el móvil sólo con volverla a instalar e iniciar sesión se vuelven a recuperar los datos.

En cuanto a la metodología de desarrollo de software, se ha decidido seguir el modelo en cascada que sugiere un enfoque sistemático y secuencial para el desarrollo del software. Se siguen las fases de forma sucesiva y la etapa siguiente empieza cuando termina la etapa anterior.

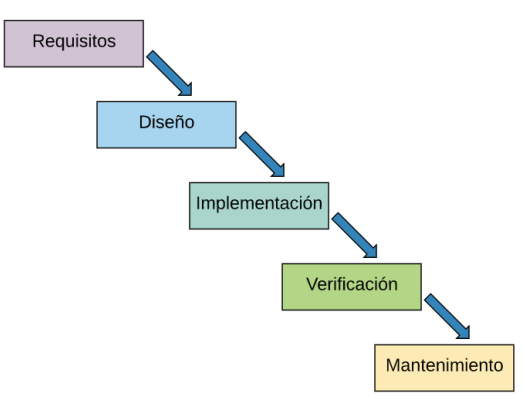


Figura 7: Las etapas del modelo en cascada
Fuente: Domínguez, P. (2017)

Se ha escogido esta estrategia de desarrollo porque es sencilla y el proyecto resulta más fácil de entender al estructurarlo de este modo, las fases quedan bien definidas. De hecho como se verá más tarde, las diversas pruebas de evaluación continua (PECs) en las que se compone el semestre del TFM coinciden con las fases de este modelo y su consecución lineal por lo que se hace natural enfocarse en este modelo. También es ideal porque los requisitos de este proyecto están claros desde la fase inicial y, en principio, no van a cambiar a lo largo del proceso de desarrollo.

A la vez también se aplicará la metodología del Diseño Centrado en el Usuario en todas las fases del desarrollo, es decir, en el análisis, el diseño, el desarrollo y la evaluación del sistema. Es un proceso cíclico, por tanto, se aplicará a todas las fases del modelo en cascada de forma iterativa para evaluar la usabilidad del diseño, orientar el diseño y desarrollo del producto e ir mejorando incrementalmente. La idea es situar al usuario en el centro del diseño de la aplicación para asegurar la consecución de un producto que cubra las necesidades de todos sus usuarios potenciales, adaptarla a sus expectativas y crear interfaces que faciliten el logro de sus objetivos. El DCU se tratará con más detalle en capítulos posteriores de este documento.

1.4. Planificación del Trabajo

En este apartado se define el plan de trabajo que se seguirá durante el desarrollo del Trabajo Final de Máster para poder cumplir con los objetivos identificados anteriormente y tener un mayor control del proceso que permita asegurar el éxito del trabajo. La planificación se compone de los siguientes pasos: descripción de los recursos necesarios para realizar el proyecto, la división del trabajo en tareas, la identificación de los hitos, la estimación de tiempos para cada tarea, la planificación temporal de cada tarea utilizando un diagrama de Gantt y, finalmente, la identificación de los riesgos que pueden provocar el retraso o fracaso absoluto del proyecto.

1.4.1. Recursos necesarios

El equipo de trabajo de este TFM está constituido por un número reducido de personas, en concreto tres personas. La alumna y autora de este proyecto, Montse Blanco de Frutos. Es la responsable del desarrollo y de la calidad del resultado final asumiendo los roles de jefa de proyecto, analista, diseñadora, programadora, responsable de ejecutar las pruebas, etc. Y los consultores Jordi Ceballos Villach que hace el seguimiento de todo el trabajo asesorando y valorando el progreso y Jordi Almirall López que, además de guiar también durante el proceso y proporcionar feedback del trabajo realizado, principalmente da soporte en los aspectos relacionados con el diseño de interacción y usabilidad.

En cuanto a los recursos hardware disponibles son los siguientes:

- Un ordenador portátil Asus con un Intel Core i5, 6GB de memoria RAM y sistema operativo instalado Windows 10 Home. Equipo para el diseño, desarrollo y documentación de todo el TFM.
- Un Smartphone Sony Xperia Z2 con Android Marshmallow (6.0.1). Para la realización de las pruebas de ejecución de la app.
- Un Smartphone Huawei P20 Pro con EMUI 9.1. También para la realización de las pruebas de ejecución de la app.

Y los recursos software a emplear son:

- LibreOffice 6.3.2.2. Suite ofimática libre para la redacción de la documentación.
- GanttProject 2.8.10. Programa de código abierto para la planificación y gestión del proyecto utilizando diagramas de Gantt.
- Java SE Development Kit 7. Software que proporciona herramientas de desarrollo para programas Java.
- Android Studio 3.5.2. Entorno de desarrollo oficial para implementar aplicaciones móviles nativas del sistema Android.
- Firebase. Plataforma integral de desarrollo para dispositivos móviles que ofrece un conjunto de herramientas como la autenticación de usuarios, almacenamiento en bases de datos y sincronización de datos en la nube.

1.4.2. Definición de tareas

En toda gestión de un proyecto se hace necesario descomponer el mismo en diferentes fases de trabajo para facilitar la planificación. En este TFM se identifican las fases en las que se divide el proyecto en función de la primera entrega de la propuesta, las diversas pruebas de evaluación continua (PECs) en las que se compone el semestre y la defensa virtual del trabajo. En cada una de estas fases se establece un hito que debe alcanzarse para dar por finalizada cada etapa. Estos hitos vienen marcados por las propias fechas de entrega de las diferentes PECs a entregar durante el desarrollo del TFM.

La siguiente tabla muestra las cinco fases con las que cuenta el plan de trabajo, delimitadas en el tiempo por una fecha de inicio y una fecha fin (fecha del hito):

Fase	Fecha inicio	Fecha fin
Definición del proyecto	18/09/2019	19/09/2019
Plan de trabajo	20/09/2019	02/10/2019
Diseño y Arquitectura	03/10/2019	30/10/2019
Implementación	31/10/2019	11/12/2019
Entrega Final	12/12/2019	08/01/2020
Defensa Virtual	20/01/2020	24/01/2020

Las fases del proyecto a su vez se dividen en tareas como los objetivos del proyecto que deberán ser cubiertos para poder finalizar el TFM con éxito. Son principalmente identificables por las actividades que deben ser desarrolladas para producir los entregables de las diferentes PECs. A continuación, se detallan las tareas a realizar en cada fase junto a una breve descripción de cada etapa y resaltando los hitos con una flecha.

Fase 0 – Definición del proyecto

- Definir la idea del proyecto
- Redactar la propuesta del TFM
- ➔ Entrega de la propuesta del TFM

En la fase inicial se concreta el trabajo que se quiere realizar y se redacta la propuesta del TFM informando las especificaciones indicadas por el consultor. La fase finaliza con la entrega de la propuesta del TFM.

Fase 1 – Plan de trabajo

- Contextualizar y justificar el trabajo
- Listar los objetivos
- Indicar el enfoque y metodología a seguir
- Planificar el trabajo
- Enumerar los productos obtenidos
- Describir el resto de capítulos de la memoria
- Búsqueda de bibliografía
- Desarrollar app tipo “Hola mundo” para la entrega
- ➔ Entrega PEC 1

En la primera fase, centrada en el plan de trabajo, se debe contextualizar y justificar el trabajo, listar los objetivos del mismo junto a los requerimientos funcionales y no funcionales de la aplicación a desarrollar. También hay que indicar el enfoque y metodología que se va a seguir durante el trabajo y detallar la planificación temporal. Se aprovechará también

para hacer una búsqueda de bibliografía para las próximas fases y se desarrollará una app muy sencilla. La documentación de la memoria viene implícita en cada una de las tareas. La fase finaliza con la entrega de la primera PEC.

Fase 2 – Diseño y Arquitectura

- ➢ Diseño Centrado en el Usuario
  - Análisis usuarios y contexto de uso
  - Diseño conceptual
  - Prototipado
  - Evaluación
- ➢ Diseño Técnico
  - Definir los casos de uso
  - Diseñar la arquitectura
- Terminar la documentación de la entrega
- Formación pre-implementación
- ➔ Entrega PEC 2

En la segunda fase uno de los principales objetivos es aplicar la metodología del diseño centrado en el usuario para diseñar y desarrollar la aplicación enfocándose en las necesidades de los usuarios objetivo. Se llevan a cabo sus cuatro fases de forma iterativa entre ellas: Usuarios y contexto de uso, Diseño conceptual, Prototipado y Evaluación. Una vez cerrada la fase del DCU hará falta realizar el diseño técnico de la aplicación trabajando dos aspectos. Por un lado la definición de los casos de uso para establecer las funcionalidades de la aplicación. Y por otro lado, definir la arquitectura del sistema, identificando las entidades de la base de datos, las clases y objetos que se utilizarán para gestionar los diferentes procesos y la estructura API de comunicación entre el cliente y el servidor.

No se especifica una tarea explícita para la documentación ya que la idea es dedicar una parte del día a plasmar la faena en la memoria. Si se ha añadido una tarea para acabar la documentación de la entrega. Añadir también que durante la finalización de esta fase se contarán con unos días de vacaciones que se aprovecharán para formarse de cara a la implementación para así disponer de más tiempo para el desarrollo. La fase finaliza con la entrega de la segunda PEC.

Fase 3 – Implementación

- ➢ Desarrollo
  - Preparar el entorno de la app
  - Preparar el entorno del backend
  - Desarrollo de la app
- ➢ Pruebas
  - Definir las pruebas
  - Realizar las pruebas
  - Corregir los errores surgidos
- Terminar la documentación de la entrega
- ➔ Entrega PEC 3

La tercera fase consiste principalmente en la construcción de la aplicación móvil. Al principio se dedicará un tiempo a preparar el entorno necesario para el desarrollo de la app configurando todas las librerías necesarias, instalando el software a utilizar, etc. A continuación, también se prepará el entorno del backend para la comunicación entre cliente y servidor. A continuación, se pasará al desarrollo de la aplicación móvil. Una vez finalizada la construcción, habrá que definir las pruebas y ejecutar las mismas. Si se detectan errores deberán ser corregidos el máximo posible.

Como se ha comentado en anteriores fases, la idea es dedicar una parte del día a plasmar en la memoria el proceso de implementación. Si se ha añadido una tarea para acabar la documentación de la entrega. La fase finaliza con la entrega de la tercera PEC.

Fase 4 – Entrega Final

- Terminar la app
- Terminar la memoria del TFM
- Redactar el manual de usuario
- Elaborar la presentación
- Grabar vídeo de la presentación
- Preparar la entrega
- ➔ Entrega Final

En la cuarta fase se deberán finalizar la app (si ha quedado algo pendiente) y la redacción de la memoria del TFM. Después habrá que elaborar una presentación en la que se expongan los puntos más destacados del trabajo y que sintetiza los objetivos del mismo, el proceso de diseño y desarrollo y los resultados obtenidos. También se deberá grabar el vídeo de la presentación incluyendo una pequeña demo que muestre el funcionamiento de la aplicación desarrollada. La fase finaliza con la entrega de los productos obtenidos que se describirán más adelante.

Fase 5 – Defensa Virtual

- Responder preguntas
- ➔ Fin Defensa Virtual

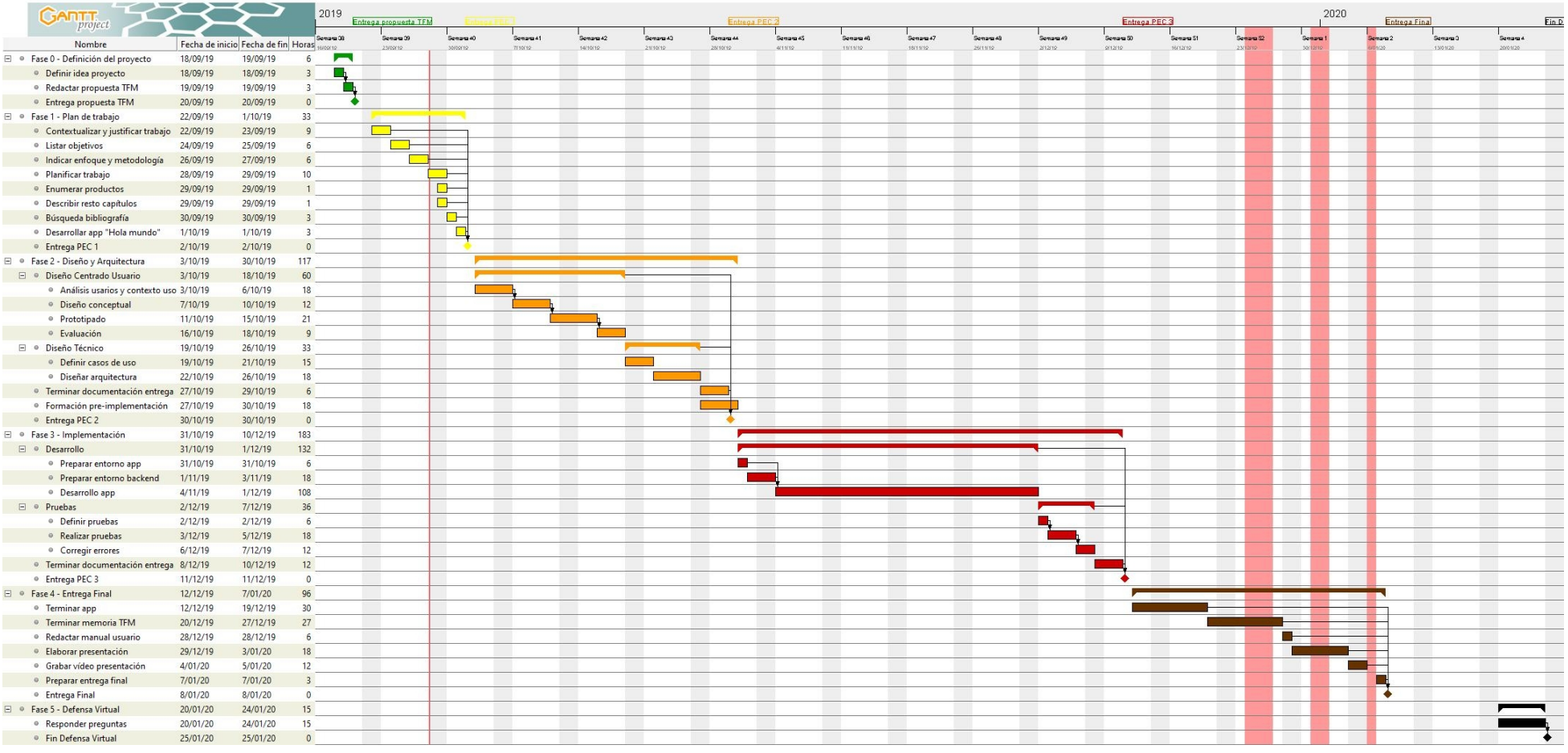
La última etapa consiste en la defensa virtual. Durante unos días los miembros del tribunal plantearán preguntas sobre el trabajo en el foro del aula y los productos que forman parte de la entrega final. La tarea consistirá en responder las preguntas en un máximo de 24 horas. La fase finaliza cuando acaba la defensa y también se daría por finalizado el TFM.

1.4.3. Planificación temporal

Una vez identificadas las tareas a ejecutar en cada fase del proyecto se pasa a la asignación de tiempos. Cada tarea está delimitada en el tiempo por una fecha de inicio y una fecha fin. En todas ellas se especifica el número de horas de dedicación excepto los hitos que son tareas especiales con duración nula y marcan el final de cada fase. Además, sólo se contará con un único recurso para llevarlas a cabo, la alumna y autora del TFM.

Para estimar su duración se ha tenido en cuenta la disponibilidad horaria disponiendo de una dedicación de 3 horas en días laborables (lunes a viernes) y de 6 horas los fines de semana y algunos festivos. Los días 28, 29, 30 y 31 de octubre y los días 2, 3, 4 y 5 de diciembre aunque son laborales se dedican 6 horas ya que son días de vacaciones y se aprovechan para evitar posibles retrasos. También comentar que los días 24, 25, 26 y 31 de diciembre y los días 1 y 6 de enero no se realizará ninguna tarea de la planificación. En total se estima un total de 450 horas de esfuerzo al proyecto contando la defensa virtual.

A continuación se puede visualizar la planificación temporal del proyecto en el siguiente diagrama de Gantt donde se exponen las tareas a ejecutar en cada fase, con las fechas de inicio y fin y la estimación de horas tanto para las fases en su totalidad como para cada tarea:



Nombre	Fecha de inicio	Fecha de fin	Horas
Fase 0 - Definición del proyecto	18/09/19	19/09/19	6
Definir idea proyecto	18/09/19	18/09/19	3
Redactar propuesta TFM	19/09/19	19/09/19	3
Entrega propuesta TFM	20/09/19	20/09/19	0
Fase 1 - Plan de trabajo	22/09/19	1/10/19	33
Contextualizar y justificar trabajo	22/09/19	23/09/19	9
Listar objetivos	24/09/19	25/09/19	6
Indicar enfoque y metodología	26/09/19	27/09/19	6
Planificar trabajo	28/09/19	29/09/19	10
Enumerar productos	29/09/19	29/09/19	1
Describir resto capítulos	29/09/19	29/09/19	1
Búsqueda bibliografía	30/09/19	30/09/19	3
Desarrollar app "Hola mundo"	1/10/19	1/10/19	3
Entrega PEC 1	2/10/19	2/10/19	0
Fase 2 - Diseño y Arquitectura	3/10/19	30/10/19	117
Diseño Centrado Usuario	3/10/19	18/10/19	60
Análisis usarios y contexto uso	3/10/19	6/10/19	18
Diseño conceptual	7/10/19	10/10/19	12
Prototipado	11/10/19	15/10/19	21
Evaluación	16/10/19	18/10/19	9
Diseño Técnico	19/10/19	26/10/19	33
Definir casos de uso	19/10/19	21/10/19	15
Diseñar arquitectura	22/10/19	26/10/19	18
Terminar documentación entrega	27/10/19	29/10/19	6
Formación pre-implementación	27/10/19	30/10/19	18
Entrega PEC 2	30/10/19	30/10/19	0
Fase 3 - Implementación	31/10/19	10/12/19	183
Desarrollo	31/10/19	1/12/19	132
Preparar entorno app	31/10/19	31/10/19	6
Preparar entorno backend	1/11/19	3/11/19	18
Desarrollo app	4/11/19	1/12/19	108
Pruebas	2/12/19	7/12/19	36
Definir pruebas	2/12/19	2/12/19	6
Realizar pruebas	3/12/19	5/12/19	18
Corregir errores	6/12/19	7/12/19	12
Terminar documentación entrega	8/12/19	10/12/19	12
Entrega PEC 3	11/12/19	11/12/19	0
Fase 4 - Entrega Final	12/12/19	7/01/20	96
Terminar app	12/12/19	19/12/19	30
Terminar memoria TFM	20/12/19	27/12/19	27
Redactar manual usuario	28/12/19	28/12/19	6
Elaborar presentación	29/12/19	3/01/20	18
Grabar vídeo presentación	4/01/20	5/01/20	12
Preparar entrega final	7/01/20	7/01/20	3
Entrega Final	8/01/20	8/01/20	0
Fase 5 - Defensa Virtual	20/01/20	24/01/20	15
Responder preguntas	20/01/20	24/01/20	15
Fin Defensa Virtual	25/01/20	25/01/20	0

Figura 8: Diagrama de Gantt del proyecto

1.4.4. Riesgos del proyecto

Un apartado importante en la gestión de un proyecto es anticipar los riesgos que podrían afectar a la planificación, el cumplimiento de los objetivos y/o a la calidad del software a desarrollar. Es preciso anticiparse a ellos, comprender bien el impacto de estos y considerar los pasos para evitarlos. En el caso que al final acaben sucediendo, se deberán crear planes de contingencia para aplicar acciones de recuperación. A continuación, se identifican los riesgos del proyecto y se realiza el plan de contingencia proponiendo acciones preventivas, mitigadoras y recuperadoras.

Riesgo	Descripción	Probabilidad	Impacto	Plan	
				Categoría	Medidas
Aumento carga de trabajo	Posibilidad que surjan días de sobrecarga de trabajo que alarguen la jornada laboral	Alta	Alto	Respaldo	Dejar margen de reacción en la planificación horaria del proyecto reservando horas. En los períodos de menor carga aprovechar para adelantar todas las tareas posibles.
				Emergencia	Si entre semana aumenta algún día la jornada laboral procurar que la situación se alargue lo mínimo posible.
				Recuperación	Utilizar las horas reservadas fuera del horario laboral para no causar desviaciones en la planificación.
Planificación incorrecta	Planificación temporal del proyecto demasiado optimista y ajustada que ponga en riesgo la entrega a tiempo	Media	Alto	Respaldo	En la planificación inicial determinar con prudencia los plazos de ejecución concediendo cierto margen para aquellas eventualidades que se puedan generar durante la ejecución.
				Emergencia	Comprobar periódicamente que se van cumpliendo los diferentes hitos del proyecto. En caso de detectar una desviación realizar pequeños ajustes en la planificación.
				Recuperación	En caso de incumplimiento total de un gran número de tareas entonces averiguar las causas, aumentar las horas de dedicación al proyecto y actualizar la planificación.
Conocimientos insuficientes	Desviaciones en la planificación del desarrollo al tener que aumentar el tiempo de aprendizaje	Media	Medio	Respaldo	Previamente a la fase de implementación, guardar una franja temporal para documentarse sobre las herramientas y tecnologías que se van a utilizar.
				Emergencia	A medida que se vaya avanzando en el desarrollo, a la mínima duda que surja consultar manuales y/o foros Online.
				Recuperación	En el caso de llegar a un punto de estancamiento, pasar a desarrollar otra funcionalidad de la aplicación y retomar la estancada en otro momento.
Problemas de salud	Sufrir de alguna enfermedad durante el proyecto, no descansar suficientes horas, etc.	Media	Bajo	Respaldo	Procurar llevar una vida sana, descansar las suficientes horas y desconectar al menos un par de veces al día.
				Emergencia	Escuchar al cuerpo para prevenir situaciones de malestar, descansar en cuanto haya señales de agotamiento o acudir al médico si los síntomas van a peor.
				Recuperación	Una vez recuperados del problema de salud retomar las tareas que se quedaron pendientes .
Interrupciones externas	Constantes interrupciones tanto si se trata de llamadas, correos electrónicos o visitas inesperadas.	Baja	Bajo	Respaldo	Reservar una franja de tiempo preestablecida en la cual evitar interrupciones mediante acciones como apagar el móvil, no consultar el correo electrónico, nada de ocio, etc.
				Emergencia	Ignorar todos aquellos temas que no sean urgentes. Si no hay alternativa entonces perder el menor tiempo posible.
				Recuperación	Identificar las condiciones que facilitan las interrupciones para poder evitarlas. Si las interrupciones se alargan en exceso, utilizar las horas reservadas fuera del horario laboral.

1.5. Breve sumario de productos obtenidos

Al acabar el Trabajo Final de Máster se entregarán los siguientes productos para su evaluación:

- Una **aplicación móvil** nativa para Android que refleja, de forma práctica, el sistema desarrollado en el marco de este proyecto. Consta de un fichero ejecutable .apk de la aplicación y el código fuente.
- La **memoria** del Trabajo Final de Máster en la que se documentan el planteamiento, los procesos de diseño y desarrollo y los resultados obtenidos del trabajo realizado y las decisiones tomadas.
- Un **manual de usuario** que que describe los pasos a seguir para instalar la aplicación en un dispositivo Android y se detallan las funcionalidades disponibles de la aplicación para su correcta utilización.
- La **presentación** del proyecto que expone los puntos más destacados del mismo.
- El **vídeo de presentación** del proyecto en el que se explica una síntesis del trabajo desarrollado incluyendo una pequeña demo del funcionamiento de la aplicación móvil creada.

1.6. Breve descripción de los otros capítulos de la memoria

En este apartado se presenta y explica brevemente en que consisten cada uno de los siguientes capítulos en los que se divide la memoria del Trabajo Final de Máster. La estructura del documento está basada en las diferentes fases que conforman el proyecto: análisis y plan de trabajo, diseño y arquitectura, implementación y entrega final.

El capítulo de **Diseño y Arquitectura** está formado por dos apartados. Por un lado, se detalla la aplicación de la metodología de **Diseño Centrado en el Usuario** para la aplicación definida en el Plan de trabajo. Está formada por las cuatro fases del DCU: Usuarios y contexto de uso, Diseño conceptual, Prototipado y Evaluación. Y por otra parte, el **Diseño Técnico** engloba la definición de los casos de uso de la aplicación que facilitan la descripción de como se usará el sistema y el diseño de la arquitectura identificando las entidades de la base de datos, las clases y objetos que se utilizarán para gestionar los diferentes procesos y la estructura API de comunicación entre el cliente y el servidor.

A continuación, en el capítulo de **Implementación** se explican los aspectos más importantes relacionados con el desarrollo de la aplicación a partir de los diseños realizados en la fase anterior y los objetivos definidos en el Plan de trabajo. Se exponen las tecnologías y herramientas utilizadas para la obtención del producto final. Muy importante también el apartado donde se explican las diversas **pruebas** ejecutadas de la aplicación para detectar posibles errores y garantizar su correcto funcionamiento.

Una vez finalizada la implementación y las pruebas, en el capítulo de **Conclusiones** se reflexiona sobre el cumplimiento de los objetivos claves planteados al inicio, las decisiones tomadas, las lecciones aprendidas y los problemas encontrados a lo largo del proceso. Además, se plantean líneas de **trabajo futuro**.

Y, por último, se incluyen un capítulo con el **glosario** en el que se definen los términos y acrónimos más relevantes usados en el presente documento. Otro capítulo, la **bibliografía**, para listar los recursos en formato físico o en formato web consultados. Y asimismo, al final se citan los **anexos** que añaden información adicional y permiten conocer más a fondo algunos aspectos específicos que debido a su extensión se tratan fuera de esta memoria.

2. Diseño y Arquitectura

2.1. Diseño Centrado en el Usuario

El diseño centrado en el usuario (DCU) es un enfoque de diseño que sitúa al usuario en el centro de todo el proceso de desarrollo del sistema y según el cual cualquier actividad de diseño debe tener en cuenta a las personas que van a hacer uso del producto y también los contextos de uso. A su vez, es una metodología que involucra al usuario en todas las fases del trabajo, desde el análisis hasta la evaluación del sistema, incluso a veces en el desarrollo. Su naturaleza iterativa implica que los procesos de desarrollo de la solución no pueden tratarse de forma lineal sino que a lo largo del ciclo de vida del proyecto es necesario un trabajo constante de revisiones, evaluaciones, adecuación y experimentación para la mejora y orientación del diseño y así hasta satisfacer las necesidades de los usuarios y del producto en sí. El DCU está formado por un conjunto de etapas que también se llevan a cabo de forma iterativa hasta conseguir los objetivos deseados, estas fases son: Usuarios y contexto de uso, Diseño conceptual, Prototipado y Evaluación.

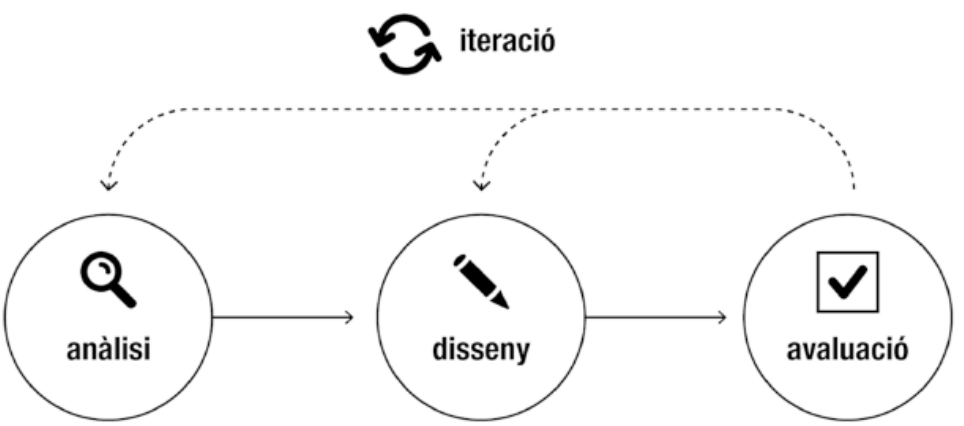


Figura 9: Flujo de trabajo del DCU
Fuente: Garreta Domingo, M.; Mor Pera, E.

En los siguientes apartados se expone la aplicación de las diferentes fases del DCU en el producto definido en el Plan de trabajo.

2.1.1. Usuarios y contexto de uso

La primera fase del DCU tiene como objetivo conocer más a fondo las características de los usuarios, sus necesidades, objetivos y actitudes, además de sus contextos de uso. La finalidad es poder detectar las funcionalidades que deberá tener la aplicación para satisfacer a los usuarios, hacer un diseño centrado en el usuario y para desarrollar una aplicación más usable. Para conseguir toda esta información se aplicarán diversos métodos de indagación que involucran a los usuarios en las diferentes actividades que se llevan a cabo. Estos métodos están clasificados en función de la información que permiten recoger: métodos cualitativos (observación, entrevistas, etc.) y métodos cuantitativos (encuestas y cuestionarios).

En este proyecto se ha decidido aplicar los métodos de indagación Benchmarking, entrevistas en profundidad y encuestas. En cambio se han descartado otros métodos como la observación e investigación contextual y las dinámicas de grupo ya que demandan una cantidad de tiempo considerable del cual no se dispone y no se ha podido reunir al suficiente número de personas para llevarlos a cabo.

En las siguientes secciones se explica el proceso de planteamiento, el desarrollo, los resultados y las conclusiones obtenidas de aplicar los métodos de indagación escogidos para finalmente identificar los diferentes perfiles de usuario y las funcionalidades que deberá tener la aplicación.

2.1.1.1 Benchmarking

El análisis competitivo o análisis comparativo, también conocido como benchmarking, es una herramienta que consiste en analizar y evaluar otros productos que son similares y/o compiten con el sistema interactivo que se está diseñando. Se comparan entre ellos y con el propio producto que se pretende desarrollar. Con el análisis de estas aplicaciones se pueden entender las tendencias del mercado, aprender de los errores, descubrir las funcionalidades básicas o comunes, estudiar las interfaces, etc.

Inconscientemente este método se aplicó en el estudio de mercado realizado en el **Contexto y justificación del trabajo**. Se realizó un breve estudio de algunas aplicaciones con características similares y que están disponibles en la tienda de Android, Google Play.

Algunas de las conclusiones desprendidas del análisis son:

- La gran mayoría de las aplicaciones están destinadas a planificar regalos únicamente para Navidad.
- Ninguna de las aplicaciones analizadas cubre todas las funcionalidades que se plantean desarrollar en este proyecto. En especial ninguna cubre las planificaciones colaborativas.
- Tampoco dan la posibilidad de registrarse en la aplicación para tener siempre disponibles los datos u ofrecer un perfil propio.

Y se destacaron algunas funcionalidades descubiertas:

- El poder seguir de un solo vistazo los obsequios para cada persona y el poder añadirlos sin la necesidad de entrar dentro de la lista de regalos de la persona en concreto.
- Poder crear eventos personalizados durante todo el año.
- Se extrajo en especial el detalle del gasto respecto al presupuesto asignado a una persona. La utilización de los colores verde y rojo para indicar si se ha sobrepasado o no el presupuesto es simple y práctico.
- Poder ordenar por nombre, fecha límite, aún no comprado, etc.
- La disponibilidad de indicar un estado para el regalo: idea, comprado, envuelto, etc.
- Ver los días que faltan para que venza un evento.

2.1.1.2 Entrevista en profundidad

El segundo método de indagación aplicado han sido las entrevistas en profundidad que permiten obtener información de tipo cualitativo para entender bien las necesidades, preferencias y experiencias de los usuarios con el producto objetivo de este trabajo. Se ha escogido este procedimiento ya que ayuda a profundizar en los posibles perfiles de usuario y aporta datos valiosos a tener en cuenta en las siguientes fases.

Las dos entrevistas han sido informales pero sin perder la intención de obtener la máxima información posible sobre cada usuario respecto a la planificación de regalos, recoger requisitos y comprobar que las funcionalidades planteadas para la aplicación son aptas y suficientes para su correcto funcionamiento. Las entrevistas se han hecho mediante una conversación en un lugar tranquilo con dos personas cercanas a las que pueden considerarse usuarios potenciales de la aplicación. El guion utilizado no es excesivamente estructurado ni largo sino que se han planteado diferentes preguntas sobre el usuario, su uso de la tecnología y como planea la compra de los regalos para poder comprender mejor las necesidades de estas personas. Este es el guion no estructurado seguido en las entrevistas:

- Edad y profesión
- Uso del móvil al día y donde se usa. Preguntar sobre los dispositivos móviles que se usan y el sistema operativo del dispositivo móvil habitual.
- ¿Haces regalos? ¿En qué ocasiones? ¿Los planificas a tiempo?

- Preguntar sobre el proceso que sigue, desde la primera idea hasta el día de regalarlo. Si hace lista de ideas, se marca un presupuesto, etc.
- ¿Usas aplicaciones para estas planificaciones? ¿O te ayudas de alguna otra aplicación que no tiene nada que ver con la temática?
- Preguntar sobre regalar en grupo y si ha hecho de organizador, participante o ambos. Como los lleva a cabo y que problemas ha tenido.
- ¿Cómo crees que debería ser una aplicación para planificar regalos? Exponer las funcionalidades planteadas en el plan de trabajo para valorar cuan útiles le son.

A continuación, para cada entrevista se hace un resumen de la respuestas y aportaciones:

Primera entrevista

Chico de 27 años que trabaja en un taller de motos. Utiliza el móvil unas 2-3 horas al día tanto en casa como fuera de ella excepto en el trabajo. Sólo dispone de un dispositivo móvil, un Huawei P20 Pro con sistema operativo Android. Considera que tiene un nivel usuario avanzado en tecnología y generalmente se adapta bien a las novedades.

Cada año suele hacer regalos para las mismas ocasiones como navidad, el tió de nadal, cumpleaños, aniversario de pareja y Sant Jordi. Y dependiendo si se da la circunstancia o no también nacimientos y bodas.

En general suele planificarlos demasiado tarde y piensa que no los gestiona bien. Su forma de proceder es intentar pensar en varias ideas y marcarse un presupuesto para cada persona. En otras ocasiones comenta que ya tiene una idea incluso bastante tiempo antes que llegue la fecha en la que lo entregará.

No usa ninguna app para planificar las compras de los obsequios aunque a veces las realiza en tiendas online. Comenta que entonces va añadiendo los productos en la cesta o en favoritos y a medida que va descartando los elimina. Esa opción les es de mucha ayuda ya que cuando los compra en tiendas físicas a veces se le olvidan las ideas previas.

En cuanto a si regala en grupo ha respondido que no mucho y siempre como participante. Confiesa que no suele colaborar en exceso, como mucho aportando algunas ideas, valorando y opinando sobre las diferentes opciones planteadas y acompañando a los demás a las compras.

Considera que una aplicación para planificar regalos debería dejar añadir diferentes obsequios para una misma persona, precio, cargar las opciones pensadas junto a una foto y el sitio donde se ha visto. También poder guardar el enlace al producto si se ha visto en una tienda online o poder marcar o indicar de algún modo cuales son las opciones favoritas. También dice que estaría bien, a parte de crear, poder borrar los obsequios añadidos si ya no interesan y ordenarlos por precio. Y piensa que sería una buena opción que se guardaran los que ya se han entregado para evitar repetir alguno por error en un futuro.

Segunda entrevista

Chica de 33 años que trabaja en una pequeña editorial. Utiliza el móvil unas 4 horas al día tanto para ocio como para el trabajo. Dispone de dos dispositivos móviles, un smartphone Xiaomi Mi A2 y una tablet Samsung Galaxy Note 10.1, ambos con sistema operativo Android. Considera que tiene bastante experiencia con los dispositivos móviles y no le cuesta aprender a usar nuevas aplicaciones.

Le encanta regalar cosas y vivencias a sus familiares y amigos para todo tipo de ocasiones como navidad, reyes magos, cumpleaños, San Valentín, el día del padre, el día de la madre, el aniversario con su pareja, nacimientos, bodas, etc.

Planifica con bastante antelación y normalmente siempre siguiendo la misma rutina. Primero añade en el calendario del móvil las fechas en las que entregará los regalos para tenerlas siempre en mente. Luego marca un presupuesto para cada persona y empieza a hacer listas de ideas. A veces incluso pregunta a los demás si necesitan alguna cosa o les hace ilusión una cosa en concreto. Cuando tiene claro que comprar entonces se lo deja anotado y va a hacer las compras.

Comenta que no utiliza ninguna aplicación que sea expresamente para planificar regalos pero se ayuda de Evernote. Explica que allí va añadiendo notas para cada evento y persona. En la nota apunta las ideas que se le ocurren y/o productos que encuentra por internet. Una vez que los ha comprado, en la misma nota escribe los obsequios entregados y el precio de cada uno para así tenerlo siempre todo registrado.

En cuanto a si regala en grupo ha respondido que sí, tanto como planificadora como participante. Explica que como planificadora suele gestionar más la aportación de las ideas como eliminando del listado aquellas que no gustan tanto, controlar el presupuesto, el dinero que ha aportado cada uno, quien ha pagado o no, etc. Y como participante toma un rol más pasivo/colaborativo aportando algunas ideas, dando su opinión, valorando sobre las ideas aportadas por todos y poco más.

Para acabar expresa que la aplicación objetivo de este proyecto le sería de mucha utilidad y valora muy positivamente funcionalidades como que queden todas las planificaciones registradas en su propia cuenta, la lista de deseos de otros usuarios, la página inicial con las planificaciones en curso y disponer del historial de planificaciones pasadas.

Las conclusiones a las que se han llegado tras la realización de las entrevistas son:

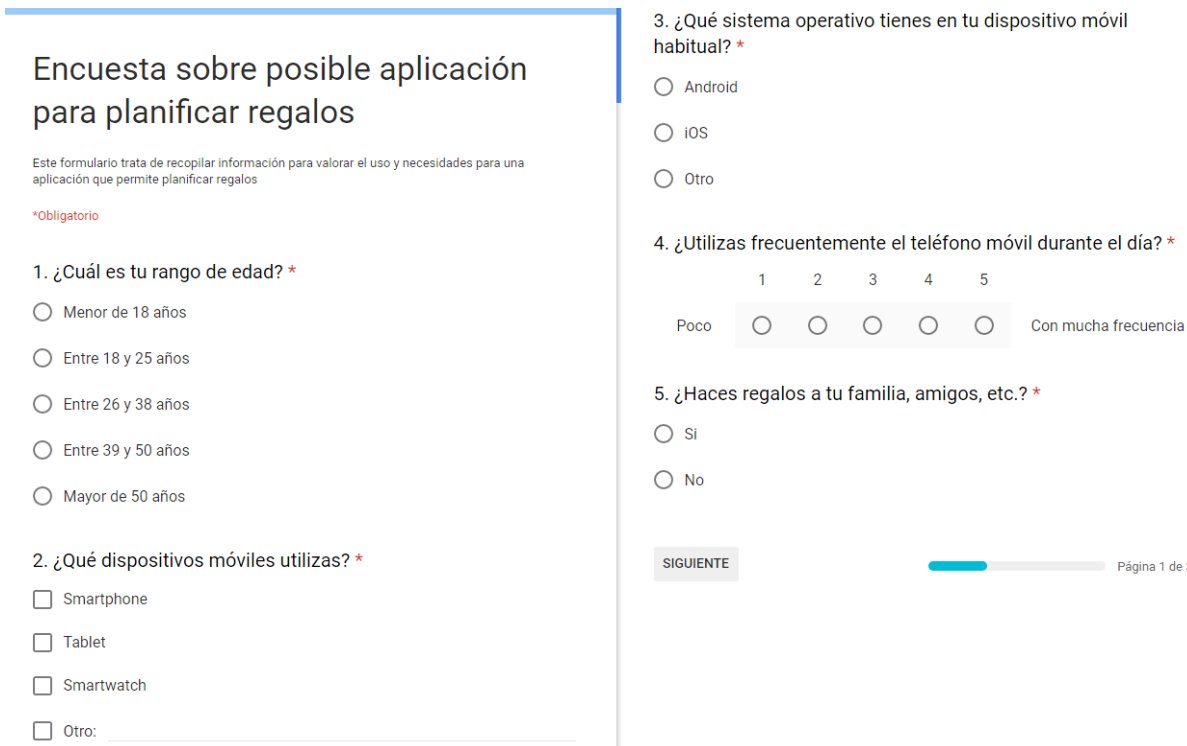
- Los dos entrevistados utilizan bastante el móvil así que la aplicación puede diseñarse para el usuario medio, sin ser excesivamente simple y sencilla.
- Queda reflejado que la elección de realizar la aplicación para sistema operativo Android fue una buena elección.
- La aplicación puede ser de utilidad tanto para personas que son muy organizadas como las que no lo son tanto o muy poco, cada uno la podría adaptar a sus necesidades.
- Se observa que en las planificaciones colaborativas hay como dos roles distintos, el planificador y los participantes. La persona que organiza, a parte de colaborar, parece que adquiere más la iniciativa en la toma de decisiones, el control del presupuesto, etc. y los participantes son más de colaborar, aportar ideas, valorar las diferentes opciones y opinar.
- A ambos les ha gustado la idea de este proyecto y piensan que la aplicación les sería de utilidad. Están bastante de acuerdo con las funcionalidades planteadas desde el principio.
- Han aportado nuevas funcionalidades como el valorar los regalos en las planificaciones en grupo y el marcar como favorito en las planificaciones individuales que indicarían cuales son las ideas preferidas.

2.1.1.3 Encuestas

El último método de indagación aplicado han sido las encuestas que son una herramienta cuantitativa que se realiza mediante preguntas estructuradas a un conjunto de usuarios. Como la aplicación a desarrollar en este proyecto no tiene un público objetivo muy concreto, cualquiera puede planificar regalos, resulta bastante útil aplicar una técnica cuantitativa para obtener la máxima variedad de opiniones posibles. Además, permite obtener datos de forma remota y la información que se obtiene tiene significado estadístico. A partir de las respuestas, se hará un posterior tratamiento y estudio y eso ayudará a terminar de definir las características de la aplicación.

El formulario se ha creado mediante la tecnología Google Forms y consta de 13 preguntas repartidas en tres secciones. Para definir las preguntas se han tenido en cuenta los resultados obtenidos en el benchmarking y también en las entrevistas. Al final de la primera y la segunda sección hay una pregunta que en función de la respuesta se sigue a la siguiente sección o se da por finalizada la encuesta. El cuestionario está disponible en la siguiente url: https://forms.gle/VrQ4ijfje1Ci3noM6

Las preguntas de la primera sección están relacionadas con el usuario y su relación con la tecnología móvil. El propósito es obtener el rango de edad mayoritario y los dispositivos móviles y sistemas operativos móvil más habituales. La última pregunta sirve para descartar a las personas que no hacen regalos, estas finalizarán la encuesta en este punto, el resto pasarán a la segunda sección.



Encuesta sobre posible aplicación para planificar regalos

Este formulario trata de recopilar información para valorar el uso y necesidades para una aplicación que permite planificar regalos

*Obligatorio

1. ¿Cuál es tu rango de edad? *
- Menor de 18 años
- Entre 18 y 25 años
- Entre 26 y 38 años
- Entre 39 y 50 años
- Mayor de 50 años

2. ¿Qué dispositivos móviles utilizas? *
- Smartphone
- Tablet
- Smartwatch
- Otro:

3. ¿Qué sistema operativo tienes en tu dispositivo móvil habitual? *
- Android
- iOS
- Otro

4. ¿Utilizas frecuentemente el teléfono móvil durante el día? *

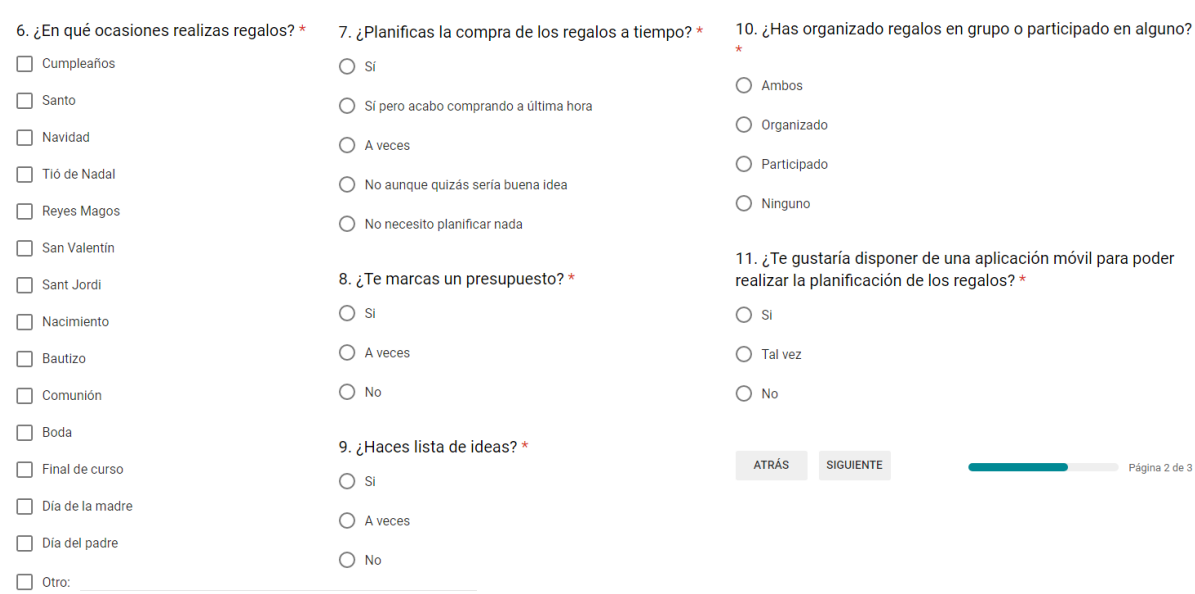
	1	2	3	4	5	
Poco	○	○	○	○	○	Con mucha frecuencia

5. ¿Haces regalos a tu familia, amigos, etc.? *
- Si
- No

SIGUIENTE Página 1 de 3

Figura 10: Primera sección de la encuesta

La segunda sección va encarada a obtener información sobre los eventos en los cuales suelen realizar obsequios los usuarios, entre otras preguntas relacionadas con como los planean. Por último, se les pregunta si les gustaría disponer de una aplicación móvil para la planificación de los regalos. Si están interesados se pasa a la siguiente y última sección sino se da por finalizada la encuesta.



6. ¿En qué ocasiones realizas regalos? *
- Cumpleaños
- Santo
- Navidad
- Tió de Nadal
- Reyes Magos
- San Valentín
- Sant Jordi
- Nacimiento
- Bautizo
- Comunión
- Boda
- Final de curso
- Día de la madre
- Día del padre
- Otro:

7. ¿Planificas la compra de los regalos a tiempo? *
- Sí
- Sí pero acabo comprando a última hora
- A veces
- No aunque quizás sería buena idea
- No necesito planificar nada

8. ¿Te marcas un presupuesto? *
- Si
- A veces
- No

9. ¿Haces lista de ideas? *
- Si
- A veces
- No

10. ¿Has organizado regalos en grupo o participado en alguno? *
- Ambos
- Organizado
- Participado
- Ninguno

11. ¿Te gustaría disponer de una aplicación móvil para poder realizar la planificación de los regalos? *
- Si
- Tal vez
- No

ATRÁS SIGUIENTE Página 2 de 3

Figura 11: Segunda sección de la encuesta

En la última sección se pretende obtener información más especifica sobre una posible aplicación para planificar los regalos. Para ello se pide valorar de menos a más importante un seguido de funcionalidades. No se da la opción de añadir otras funcionalidades que esperarían o desearían tener ya que las expuestas son las principales o las que se dudaba más que opinión podría tener la gente. Y por último se pide marcar los estados para indicar el grado de avance

en la planificación de cada obsequio. Se podrían haber añadido más preguntas específicas sobre las funcionalidades pero entonces la encuesta hubiera quedado demasiado larga y eso puede provocar que se abandone a la mitad.

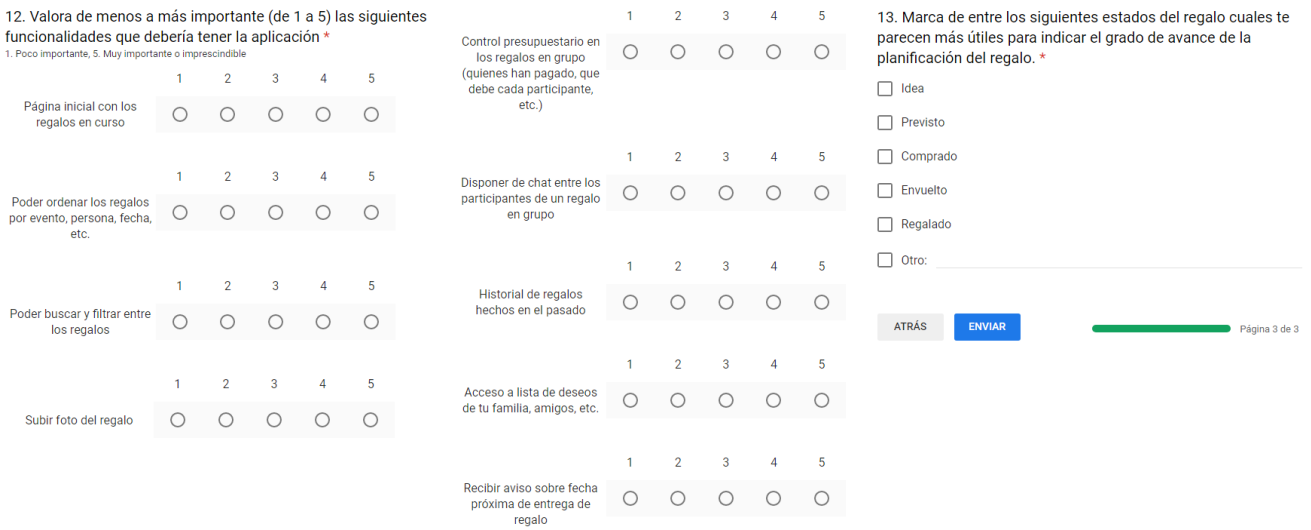


Figura 12: Tercera y última sección de la encuesta

La encuesta se envío por correo electrónico y también se compartió directamente el enlace a algunos amigos y conocidos, a los que se vio dispuestos a colaborar. Finalmente 31 personas respondieron a la encuesta. No es una muestra muy representativa de usuarios pero se intentará extraer conclusiones con los resultados obtenidos.

En los resultados se puede observar que han respondido personas de distintos rangos de edad aunque sobre todo individuos entre los 26 y los 38 años aunque también un porcentaje parecido entre los 39 y 50 años.

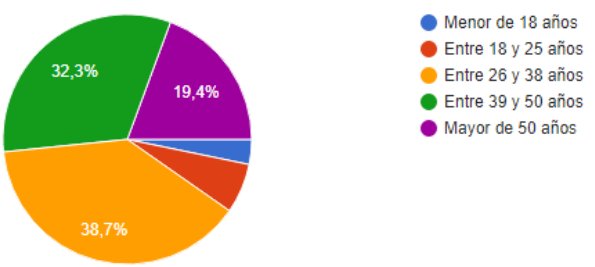


Figura 13: Edad de los encuestados

En cuanto a las preguntas sobre la relación de los usuarios con los dispositivos móviles, como era de esperar el 100% de los usuarios utiliza un smartphone. Sin olvidar que casi la mitad de los usuarios, un 48,4%, también usan la tablet. Sólo un 19,4% dispone de smartwatch. Tampoco hay ninguna sorpresa respecto al sistema operativo más utilizado al ver el 87,1% de Android respecto al 12,9% de iOs. Y en general, los encuestados utilizan con bastante frecuencia el teléfono móvil durante el día así que se puede deducir que estarán familiarizados con el uso de aplicaciones móviles y tendrán cierta facilidad de aprendizaje ante una nueva aplicación.

En la encuesta también se refleja que casi todos los participantes hacen regalos si bien ha habido tres personas que han respondido que no. Las personas que han respondido afirmativamente han pasado a la siguiente sección más enfocada al planteamiento de la aplicación que se pretende desarrollar. En la primera pregunta de esta sección han indicado las ocasiones en las que suelen realizar obsequios. No hay sorpresa en cuanto al alto porcentaje de los Cumpleaños y Navidad. En este punto interesa saber que otras ocasiones indican, como Reyes Magos, bodas y día de la madre que han sido las otras ocasiones con mayor porcentaje de selección. Se considerarán también las opciones añadidas como otras opciones, la celebración de los años en un relación o en un matrimonio.

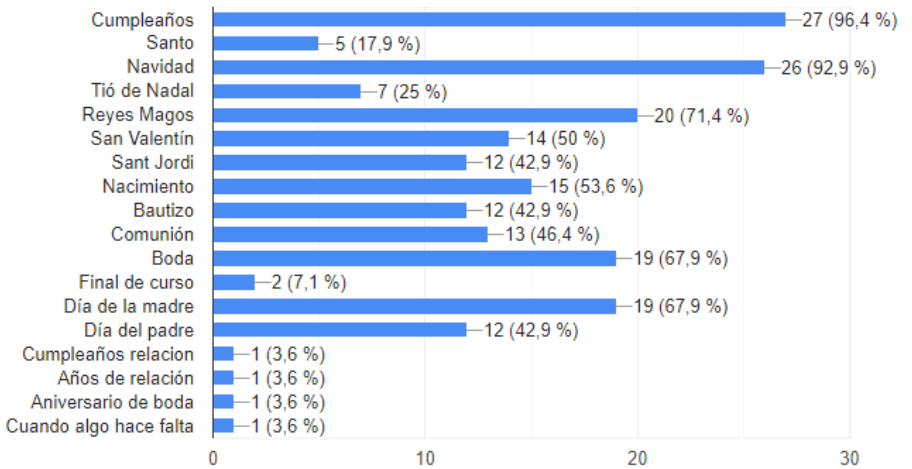


Figura 14: Ocasiones en las que se realizan regalos

Un poco más de la mitad de los encuestados planifica las compras de los regalos con anterioridad sin embargo un 21,4% acaba comprando a última hora, se presupone que debido a una ineficiente gestión. Un 25% a veces las planifica y el resto no lo hace si bien un pequeño porcentaje estaría dispuesto a cambiar. En cuanto a si se marcan un presupuesto o no, puede verse una clara división de opiniones. La mayoría hacen presupuesto en algunas ocasiones, seguido de los que sí lo hacen en todas las ocasiones y por último los que no lo hacen nunca. Sumando los que han respondido a veces y los que sí, hay un 75% de personas que necesitarían disponer en la aplicación de una funcionalidad para gestionar el presupuesto aunque no siempre todos hicieran uso de ella.

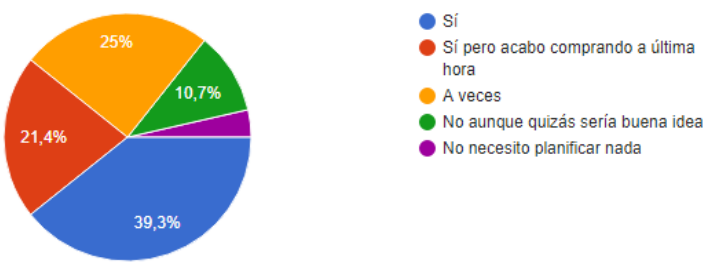


Figura 15: ¿Planificas la compra de los regalos a tiempo?

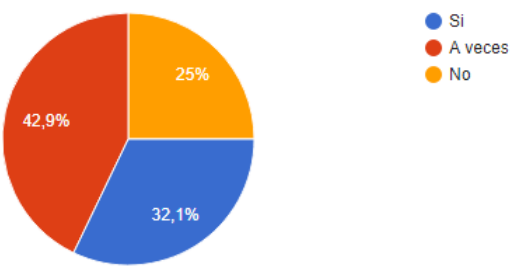


Figura 16: ¿Te marcas un presupuesto?

También hay una clara división de opiniones en cuanto al uso de listas de ideas, de hecho más o menos hay el mismo porcentaje de encuestados que las hacen, que las elaboran a veces y que nunca. Aunque la suma de los porcentajes que han respondido que si y a veces no es tan alto como en el caso del presupuesto así que esta funcionalidad en principio deberá estar pero no es tan prioritaria. Y muy importante las respuestas sobre la organización o participación en regalos colaborativos. Puede verse que más de la mitad de los encuestados tanto han organizado en grupo como participado en alguno por lo que los usuarios deberían poder tener los dos roles simultáneamente en la aplicación.

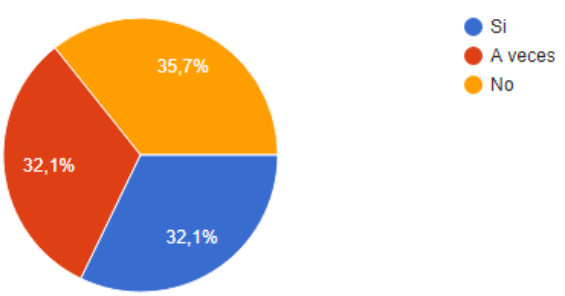


Figura 17: ¿Haces lista de ideas?

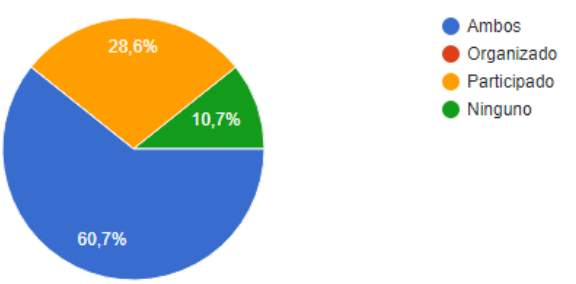


Figura 18: ¿Has organizado regalos en grupo o participado en alguno?

La idea que se desarrolla en este trabajo no ha tenido la aceptación esperada, el 46,4% de los encuestados responden que tal vez les gustaría disponer de una aplicación móvil para poder realizar la planificación de los regalos. Un 35,7% responden que sí estarían interesados en ella y un 17,9% que no. Es un resultado un tanto decepcionante aunque se puede traducir en que simplemente deberá ser mucho mayor el esfuerzo en implementar una buena aplicación para que sea aceptada por todos y ese porcentaje de personas que es más dubitativa quieran seguro utilizar la aplicación.

En cuanto a la valoración de las funcionalidades indicadas se han obtenido los siguientes resultados:

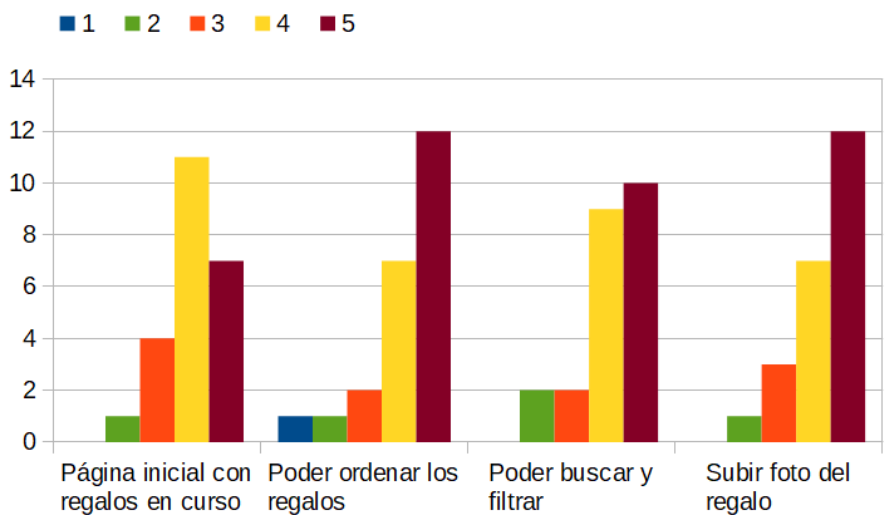


Figura 19: Valoración de las funcionalidades I

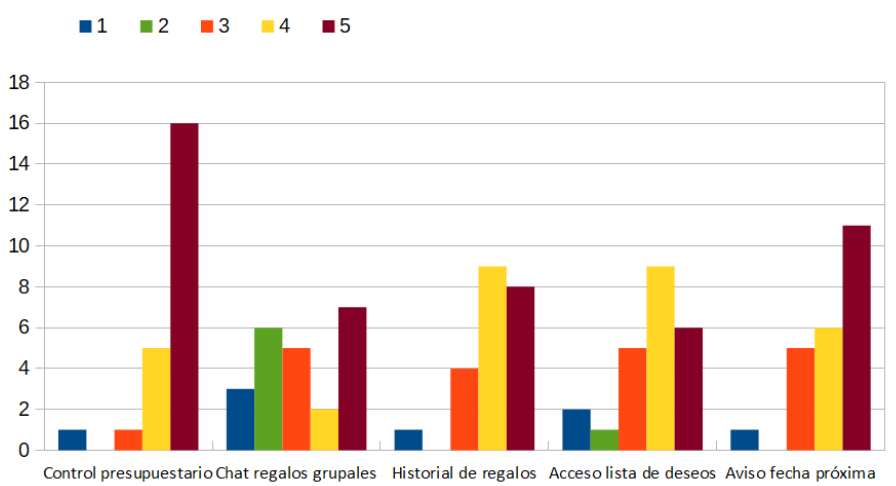


Figura 20: Valoración de las funcionalidades II

Las funcionalidades que la mayoría de usuarios consideran imprescindibles para una aplicación sobre planificación de regalos son el control presupuestario en los regalos en grupo y la disponibilidad de subir una foto de un regalo. A continuación, se consideran necesarias las funcionalidades para poder ordenar los regalos por evento, persona, fecha, etc. y el buscar y filtrar entre los regalos. Y, por supuesto, les sigue muy de cerca el recibir un aviso sobre fecha próxima de entrega de regalo. Algo menos indispensables son el disponer de un historial de regalos hechos en el pasado y, la más sorprendente, la página inicial con los regalos en curso. La verdad es que esta última funcionalidad se esperaba que fuera más indispensable para los usuarios. Y las funcionalidades con menor valoración son el acceso a la lista de deseos de familiares, amigos, etc. y disponer de un chat entre los participantes de un regalo en grupo.

Por último, se pedía marcar de entre un listado de diferentes estados para indicar el grado de avance de la planificación de un regalo, cuales resultan más útiles y/o añadir algún estado no contemplado. Los estados más valorados han sido "Comprado" e "Idea", seguido de "Regalado" y "Previsto". El menos valorado ha sido "Envuelto". Una persona ha añadido el estado "Pendiente comprar" que sería parecido al estado "Previsto".

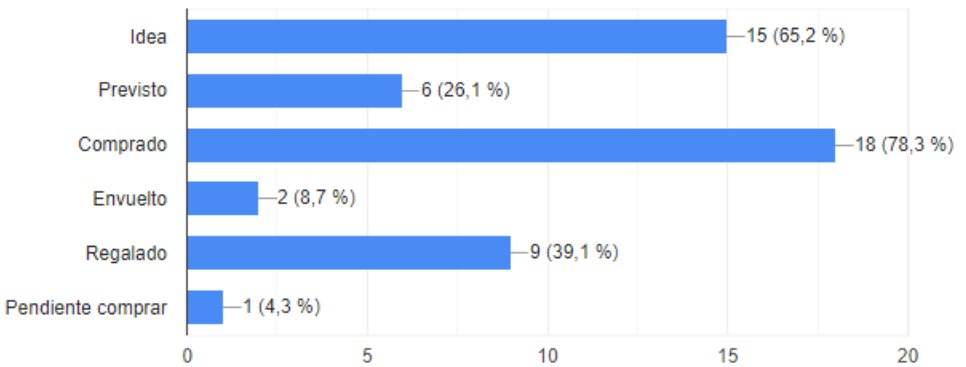


Figura 21: Valoración de los estados de los regalos

Las conclusiones a las que se han llegado con el análisis de los resultados de las encuestas son:

- El público objetivo de la aplicación serán usuarios con una edad comprendida entre los 26 y los 50 años.
- Queda reflejado que la elección de realizar la aplicación para sistema operativo Android fue una buena elección.
- La aplicación deberá poderse utilizar tanto en smartphones como tablets ya que son los dispositivos móviles más habituales entre los usuarios como se ha visto.
- Los usuarios utilizan bastante el móvil así que la aplicación puede diseñarse para el usuario medio, sin ser excesivamente simple y sencilla.
- Se ha visto que hay encuestados que aunque planifican los regalos a tiempo acaban comprando a última hora. Otros parece que a veces son más de improvisar y el resto no lo hace si bien un pequeño porcentaje esta dispuesto a cambiar. De aquí se puede extraer que los usuarios necesitan una buena aplicación y sencilla de usar y así también convencer a los dubitativos que les ayudará utilizarla.
- Se necesitará disponer en la aplicación de una funcionalidad para indicar el presupuesto de los regalos aunque no siempre todos hicieran uso de ella.
- Los usuarios pueden tener tanto el rol planificador de regalos individual, el de en grupo y el rol de participante en la aplicación, los tres roles simultáneamente en la aplicación aunque en planificaciones distintas.
- La idea que se desarrolla en este trabajo no ha tenido la aceptación esperada pero se trabajará duro en implementar una aplicación que sea útil para todos, que sea aceptada por ese porcentaje de personas que es más dubitativa y así quieran seguro utilizar la aplicación.
- En general las funcionalidades presentadas en el plan de trabajo complacen las principales necesidades de los usuarios aunque el disponer de chat entre los participantes de un regalo en grupo ha sido la peor valorada. Debido a su posible complejidad y que no ha gustado se decide no implementarla en la aplicación.

Las respuestas de cada uno de los participantes pueden consultarse en los **Anexos**.

2.1.1.4 Perfiles de usuario

Una vez aplicados los métodos de indagación se han analizado los resultados, extraído conclusiones y recopilado requisitos para definir los perfiles de usuario, cada uno con sus características y necesidades que se relacionan con las funcionalidades de la aplicación. Recordar que se determinó que un usuario puede ser planificador individual, planificador en grupo y participante simultáneamente si se requiere. A continuación, se presentan los cuatro perfiles de usuario identificados:

Usuario Planificador Individual Habitual	
Características del perfil	Los usuarios de este perfil en general tienen una edad comprendida entre los 26 y los 50 años, aunque puede haber personas de otras edades que planifiquen regalos habitualmente. Utilizan bastante los dispositivos móviles durante el día así que están familiarizados con el uso de aplicaciones móviles. Son usuarios que en su mayoría hacen bastantes regalos y por lo común planean a tiempo. Así que el objetivo fundamental de este perfil de usuario es poder planificar y gestionar con antelación y de forma ordenada y organizada los regalos para diferentes ocasiones y personas. Son usuarios que normalmente se anticipan y tienen en mente las fechas en las que quieren hacer los regalos. Antes de ir a comprarlos hacen listas de ideas aportando fotos, enlaces o anotaciones del sitio donde se pueden comprar. Tienen en cuenta los gustos, personalidad y necesidades de los demás. Y se marcan un presupuesto, normalmente por persona. Además, requerirán herramientas que les ayuden a buscar entre toda la información o a filtrarla para encontrar más fácilmente una planificación, una persona, un regalo o cualquier otro concepto de forma rápida y precisa. En definitiva, son personas que lo tienen todo pensado y normalmente son muy metódicas. Así que necesitarán realizar un uso extenso de las funcionalidades de la aplicación.
Contexto de uso	Las personas que entran dentro de este perfil de usuario utilizarán la aplicación cuando se encuentren en casa y también cuando estén fuera de ella en situaciones que requieran realizar cambios en la planificación como cuando se están buscando ideas, mirando tiendas, comprando los regalos, etc. Aunque deberán estar conectados a Internet. Normalmente la usarán en cualquier sitio y en cualquier momento que les haga falta ya que son usuarios que irán aportando muchos cambios como añadir más ideas, eliminar las que van descartando, hacer anotaciones, ir cambiando el estado en el grado de avance de los regalos, etc.
Tareas	- Registrarse en la aplicación. - Iniciar sesión en la aplicación. - Listar las planificaciones en curso. - Listar las planificaciones realizadas en el pasado. - Búsqueda de planificaciones en el listado. - Filtrar las planificaciones del listado. - Ordenar las planificaciones por evento, persona, fecha, etc. - Crear/Visualizar/Editar/Eliminar una planificación de regalos de forma individual. - Indicar presupuesto. - Activar/Desactivar notificación de fecha límite próxima. - Añadir/Editar/Borrar regalo en una planificación. - Visualizar/Subir fotografía de un regalo. - Indicar precio y/o enlace a un regalo. - Hacer anotaciones sobre un regalo. - Cambiar estado del grado de avance de un regalo. - Marcar regalos como favoritos. - Dar por finalizada una planificación. - Agregar/Eliminar usuario de contactos. - Ver las notificaciones del usuario. - Ver la lista de deseos de un usuario agregado a contactos. - Visualizar/Editar/Borrar la propia lista de deseos.
Características descubiertas	A partir de los métodos de indagación se ha descubierto que en este perfil es de mucha utilidad que todas las planificaciones queden registradas en su propia cuenta, la página inicial con los regalos en curso y el disponer del historial de los regalos realizados en el pasado. También les resulta útil disponer de funcionalidades para ordenar las planificaciones por evento, persona, fecha, etc. y el buscar y filtrar. Además el recibir un aviso sobre la fecha próxima de entrega de un regalo.

Usuario Planificador Individual Esporádico	
Características del perfil	Los usuarios de este perfil suelen tener una edad comprendida entre los 39 y los 50 años o más, aunque puede haber personas de otras edades que planeen regalos de forma esporádica. Utilizan ocasionalmente los dispositivos móviles durante el día así que puede ser que no estén tan familiarizados con el uso de aplicaciones móviles. Son usuarios que no hacen muchos regalos durante el año y por lo común no planifican con antelación así que normalmente acaban comprando a última hora. El objetivo fundamental de este perfil de usuario es disponer de las funcionalidades más básicas para llevar la planificación de los regalos. Tienen suficiente con añadir regalos por evento y persona, indicar presupuesto si lo desean, el precio, marcar los regalos favoritos, ver los regalos que tienen en curso y poco más. A diferencia del anterior perfil no planean excesivamente la compra de los regalos, no suelen ser muy estructurados ni seguir siempre los mismos métodos de organización y más bien irán utilizando más o menos funcionalidades de la aplicación en función de las circunstancias.
Contexto de uso	Las personas que entran dentro de este perfil de usuario no utilizarán la aplicación de forma tan habitual como el anterior perfil por lo que seguramente se utilizará más dentro de casa o en sitios muy concretos como por ejemplo cuando estén pensando ideas o comprando los regalos. Deberán estar conectados a Internet.
Tareas	- Registrarse en la aplicación. - Iniciar sesión en la aplicación. - Listar las planificaciones en curso. - Crear/Visualizar/Editar/Eliminar planificación de regalos de forma individual. - Indicar presupuesto. - Añadir/Editar/Borrar regalo en una planificación. - Visualizar/Subir fotografía de un regalo. - Indicar precio y/o enlace a un regalo. - Marcar los regalos favoritos. - Dar por finalizada una planificación.
Características descubiertas	Gracias a los métodos de indagación se ha descubierto que existe un perfil planificador individual más esporádico y que no necesita tantas funcionalidades en la aplicación. De todas formas las tendrá disponibles todas por si en algún momento las quisiera utilizar. También se ha descubierto que, aunque no son tan organizados como el anterior perfil, si es especialmente importante indicar un presupuesto en la planificación. Además, quieren poder subir fotos de los regalos y marcar los favoritos de entre los añadidos para facilitarles el posterior descarte de aquellos que al final no se vayan a comprar y que eran sólo ideas.

Usuario Planificador Grupo	
Características del perfil	Los usuarios de este perfil en general tienen una edad comprendida entre los 26 y los 38 años, aunque puede haber personas de otras edades que planeen regalos colaborativos con otras personas. Utilizan bastante los dispositivos móviles durante el día así que están familiarizados con el uso de aplicaciones móviles. Son usuarios que planifican regalos colaborativos y, por tanto, adquieren más la iniciativa en la toma de decisiones, el control del presupuesto, se guían también de las opiniones de los participantes, etc. Es parecido al usuario planificador individual habitual pero aplicado en el organizador de un grupo.
Contexto de uso	Las personas que entran dentro de este perfil de usuario utilizarán la aplicación cuando se encuentren en casa y también cuando estén fuera de ella aunque deberán estar conectados a Internet. Normalmente la usarán en cualquier sitio y en cualquier momento que les haga falta ya que al ser

	usuarios planificadores en un grupo deben mantener una organización, estar al tanto de las modificaciones y aportaciones que van haciendo los participantes, borrar todo aquello que se va descartando, etc. Seguramente también la utilizarán en momentos muy concretos como por ejemplo cuando estén pensando ideas o comprando los regalos.
Tareas	- Registrarse en la aplicación. - Iniciar sesión en la aplicación. - Listar las planificaciones en curso. - Listar las planificaciones realizadas en el pasado. - Búsqueda de planificaciones en el listado. - Filtrar las planificaciones del listado. - Ordenar las planificaciones por evento, persona, fecha, etc. - Crear/Visualizar/Editar/Eliminar planificación de regalos de forma colaborativa. - Indicar presupuesto. - Activar/Desactivar notificación de fecha límite próxima. - Agregar/Eliminar participante en una planificación en grupo. - Añadir/Editar/Borrar regalo en una planificación en grupo. - Visualizar/Subir fotografía de un regalo. - Indicar precio y/o enlace a un regalo. - Hacer anotaciones sobre un regalo. - Controlar el presupuesto como que debe cada participante, quien ha pagado, etc. - Cambiar estado del grado de avance de un regalo. - Valorar los regalos. - Dar por finalizada una planificación. - Agregar/Eliminar usuario de contactos. - Ver las notificaciones de usuario. - Ver la lista de deseos de un usuario agregado a contactos.
Características descubiertas	Tras analizar los resultados de los métodos de indagación se ha observado que en los regalos colaborativos se diferencian dos roles distintos muy marcados, el planificador y los participantes, y se han definido más específicamente las funcionalidades que diferencian ambos roles. Los planificadores tendrán funcionalidades que los participantes no dispondrán como borrar regalos en el grupo, añadir o eliminar un participante, activar/desactivar la notificación de fecha límite próxima, el control del presupuesto, cambiar el estado del grado de avance de un regalo y dar por finalizada una planificación en grupo.

Usuario Participante	
Características del perfil	Los usuarios de este perfil suelen tener una edad comprendida entre los 26 y los 50 años, aunque puede haber personas de otras edades que participen en regalos en grupo con otras personas. Utilizan bastante los dispositivos móviles durante el día así que están familiarizados con el uso de aplicaciones móviles. Son usuarios que participan en regalos grupales y, por tanto, toman un rol más pasivo/colaborativo aportando algunas ideas, dando su opinión, valorando las ideas aportadas por todos y poco más.
Contexto de uso	Las personas que entran dentro de este perfil de usuario seguramente no utilizarán la aplicación de forma muy habitual ya que toman un rol más bien pasivo/colaborativo. Seguramente se utilizará más dentro de casa, en momentos o en sitios muy concretos como por ejemplo cuando estén decidiendo los regalos. Deberán estar conectados a Internet.
Tareas	- Registrarse en la aplicación. - Iniciar sesión en la aplicación. - Listar las planificaciones en curso. - Añadir/Editar regalo en una planificación. - Visualizar/Subir fotografía de un regalo. - Indicar precio y/o enlace a un regalo. - Hacer una anotación en un regalo. - Valorar los regalos. - Agregar/Eliminar usuario de contactos. - Ver las notificaciones de usuario. - Ver la lista de deseos de un usuario agregado a contactos.
Características descubiertas	Como se ha comentado en el anterior perfil, a partir de los resultados de los métodos de indagación se definió más específicamente las diferencias entre los roles planificador y los participantes en las planificaciones en grupo. Los participantes son más de colaborar, aportar ideas, opinar, etc. Además, en el anterior perfil se han explicado las funcionalidades que los participantes no tienen disponible a diferencia del planificador. Asimismo, se descubrió una funcionalidad que no se había planteado desde el principio como el valorar los regalos en las planificaciones colaborativas que indicarán cuales son las ideas favoritas del planificador y los participantes de entre todas las opciones añadidas.

2.1.2. Diseño conceptual

En la segunda fase del DCU se utiliza toda la información recopilada en la primera fase para elaborar los escenarios de uso y el flujo de interacción de la aplicación. Estas técnicas ayudan a tomar decisiones de diseño que hagan cumplir los requisitos determinados de los usuarios.

2.1.2.1 Escenarios de uso

A continuación, se presentan los escenarios de uso que describen a los usuarios con unos objetivos concretos utilizando la aplicación en situaciones que podrían ser completamente reales. La construcción de estos escenarios ayuda a identificar objetivos y necesidades de los usuarios y funcionalidades para el diseño.

Escenario de uso 1 – Crear una planificación de regalos individual simplificada	
Perfil de usuario	Usuario Planificador Individual Esporádico y Usuario Planificador Individual Habitual.
Contexto en el que se encuentra	Una chica está repartiendo los turnos de trabajo de la semana, al mirar el calendario se da cuenta que pronto es el día de la madre y se le había olvidado. Así que decide que esa misma tarde, mientras va en metro hacia el centro en el que asiste a clases nocturnas de formación profesional, buscará ideas desde su tablet y guardará los enlaces en la aplicación. No tiene mucho tiempo libre así que detallará lo mínimo para en otro momento descartar rápido las que menos le gusten para después realizar la compra.
Objetivos que tiene	Planificar de forma rápida los regalos que necesita hacer para el día de la madre.

Tareas que lleva a cabo	Crear una planificación individual informando los detalles obligatorios y añadir en esta los regalos indicando la mínima información correspondiente.
Necesidades de información	Tener claro por lo menos para que evento y persona se hace la planificación individual y saber en todo caso un regalo que se desea añadir y una breve información complementaria como el nombre y un enlace a la web donde ver información y/o para comprar.
Funcionalidades necesarias	- Crear una planificación individual. - Añadir un regalo en la planificación.
Desarrollo de las tareas	1. [Registrarse]. 2. Iniciar sesión. 3. Seleccionar la opción de crear planificación individual. 4. Rellenar la información mínima de la planificación. 5. Seleccionar la opción de añadir regalo. 6. Rellenar la información mínima del regalo. 7. Crear la planificación individual.

Escenario de uso 2 – Crear una planificación individual detallada	
Perfil de usuario	Usuario Planificador Individual Habitual.
Contexto en el que se encuentra	Un hombre está revisando unos documentos excel repletos de datos bancarios, se nota cansado así que aparta un momento la vista de la pantalla de su portátil y ve el retrato de su marido. Piensa en él, en su matrimonio y que dentro de un mes celebrarán cinco años de casados. Quiere hacerle un regalo muy especial y como es bastante previsor anota en la agenda de su teléfono móvil que por la noche, en su momento de relax, empiece la planificación del regalo en la aplicación y entrar ideas en ella.
Objetivos que tiene	Gestionar con bastante tiempo de antelación la planificación de los regalos que necesita hacer para conseguir darle a su marido el mejor aniversario de bodas hasta ahora.
Tareas que lleva a cabo	Crear una planificación individual informando los detalles específicos y añadir en esta los regalos indicando el máximo detalle posible de cada uno.
Necesidades de información	Tener claro para que evento, persona y fecha se hace la planificación individual, el presupuesto disponible para la compra y saber al menos un regalo que se desea añadir y la información complementaria que requiere informar como el nombre, el precio, un enlace a la web donde ver información y/o para comprar, anotaciones, etc. Además, disponer de la fotografía que se quiere subir.
Funcionalidades necesarias	- Crear una planificación individual. - Informar el presupuesto disponible. - Activar la notificación para avisar cuando se aproxime la fecha indicada en la planificación. - Añadir un regalo en la planificación. - Añadir una fotografía del regalo. - Cambiar el estado de grado de avance del regalo.
Desarrollo de las tareas	1. [Registrarse]. 2. Iniciar sesión. 3. Seleccionar la opción de crear planificación individual. 4. Rellenar toda la información de la planificación. 5. Indicar un presupuesto para la planificación. 6. Elegir la opción para activar la notificación de fecha límite próxima. 7. Seleccionar la opción de añadir regalo. 8. Rellenar toda la información del regalo. 9. Seleccionar la opción para subir una fotografía del regalo. 10. Escribir unas anotaciones. 11. Elegir la opción “idea” como estado del regalo. 12. Crear la planificación individual.

Escenario de uso 3 – Crear una planificación de regalos en grupo	
Perfil de usuario	Usuario Planificador Grupo.
Contexto en el que se encuentra	Una mujer que trabaja de profesora de informática en una academia ha sido invitada por una amiga suya para celebrar su cumpleaños dentro de dos semanas. En la fiesta serán pocas personas así que le propone al resto de invitados de hacer los regalos conjuntamente ya que se trata de un aniversario especial, la amiga cumple los cuarenta años, y así le hacen una sorpresa. Como ella es quien ha sugerido la idea, y quien domina más de tecnología, entonces se encargará de gestionarla en la aplicación desde su smartphone aunque no tiene ninguna idea por ahora.
Objetivos que tiene	Gestionar una nueva planificación colaborativa de regalos que necesita hacer conjuntamente con otras personas para el cumpleaños de una amiga en común.
Tareas que lleva a cabo	Crear una planificación en grupo informando los detalles específicos y agregar a los otros convidados como participantes (si alguno tiene usuario en la planificación entonces previamente agregarlo en contactos).
Necesidades de información	Tener claro para que evento, persona y fecha se hace la planificación colaborativa, el presupuesto disponible para la compra y conocer que otras personas van a participar.
Funcionalidades necesarias	- Añadir a otro usuario en contactos (sólo si no se tenían agregados antes). - Crear una planificación en grupo. - Informar el presupuesto disponible. - Activar la notificación para avisar cuando se aproxime la fecha indicada en la planificación. - Agregar participantes en la planificación.
Desarrollo de las tareas	1. [Registrarse]. 2. Iniciar sesión. 3. [Agregar contacto] 4. Seleccionar la opción de crear planificación en grupo. 5. Rellenar toda la información de la planificación. 6. Indicar un presupuesto para la planificación. 7. Activar la notificación de fecha límite próxima. 8. Seleccionar la opción para agregar a un contacto como participante en la planificación. 9. Crear la planificación en grupo.

Escenario de uso 4 – Participar en una planificación de regalos en curso	
Perfil de usuario	Usuario Participante. También adaptable a cualquiera de los otros perfiles de usuario porque todos pueden añadir regalos en una planificación en curso (no se repiten ya que las diferencias en los escenario de uso son mínimas).
Contexto en el que se encuentra	Un hombre está hablando por teléfono con una amiga de la universidad y le comenta que está invitado para celebrar su cumpleaños dentro de dos semanas. Una de las convidadas a la fiesta les ha sugerido a todos los invitados de hacer los regalos conjuntamente ya que se trata de un aniversario especial, la amiga cumple los cuarenta años, y así le hacen una sorpresa. Él propone al resto que quien gestione la planificación sea la que les ha sugerido de hacer el regalo conjuntamente ya que como es profesora de informática seguro que sabrá utilizar mucho mejor la aplicación y todos están de acuerdo. Él no tiene mucha experiencia con los dispositivos móviles, dispone de un smartphone muy sencillo y sólo con acceso a internet desde wifi así que en la aplicación colaborará cuando esté en casa o en algún espacio que disponga de wifi gratuita.
Objetivos que tiene	Participar en una nueva planificación grupalde regalos en la que es invitado por la planificadora del grupo para hacer los regalos conjuntamente con otras personas para el cumpleaños de una amiga en común.
Tareas que lleva a cabo	Listar las planificaciones que tiene en curso y editar aquella en la que está como participante. Añadir en esta los regalos indicando la mínima información correspondiente. Previamente, si la planificadora y él no se tienen agregados a contactos entonces antes la planificadora le ha tenido que añadir en contactos por lo que deberá aceptar la solicitud de unirse a contactos.

Necesidades de información	Ser consciente de en que planificación en grupo va a colaborar como participante y saber al menos un regalo que se desea añadir. De este conocer la información complementaria que requiere informar como el nombre, el precio, un enlace a la web donde ver información y/o para comprar, anotaciones, etc. Además, disponer de la fotografía que se quiere subir.
Funcionalidades necesarias	- Aceptar la solicitud de un usuario para agregar a contactos (sólo si no se tenían agregados antes). - Listar planificaciones en curso. - Editar una planificación en grupo. - Añadir un regalo en la planificación. - Añadir una fotografía del regalo.
Desarrollo de las tareas	1. Iniciar sesión. 2. [Aceptar solicitud de un usuario para agregar a contactos] 3. Seleccionar la opción para listar las planificaciones en curso. 4. Seleccionar la opción de editar una planificación en grupo. 5. Seleccionar la opción de añadir regalo. 6. Rellenar la información del regalo. 7. Seleccionar la opción para subir una fotografía del regalo. 8. Escribir unas anotaciones. 9. Guardar la planificación en grupo.

Escenario de uso 5 – Agregar a otro usuario como contacto	
Perfil de usuario	Usuario Planificador Individual Habitual. También el Usuario Planificador Grupo y el Usuario Participante en un contexto adaptado a cada uno.
Contexto en el que se encuentra	Una chica que cursa el último año de instituto utiliza frecuentemente la aplicación para gestionar los regalos para su familia y amigos. Va guardando todo lo que le gusta en su lista de deseos y se adelanta a los próximos eventos creando planificaciones con bastante tiempo de antelación para ir añadiendo ideas. Durante el día aprovecha cualquier momento libre para utilizar el smartphone y en casa lo cambia por la tablet ya que está más cómoda. El caso es que un día entre clase y clase está modificando su lista de deseos y un amigo le comenta que se ha instalado la planificación. Emocionada le pide su nombre de usuario para agregarlo y así tenerlo localizado para ver su lista de deseos y hacer regalos en conjunto.
Objetivos que tiene	Añadir en sus contactos de la aplicación a un usuario existente en el sistema.
Tareas que lleva a cabo	Ir a los contactos de la aplicación, buscar por nombre de usuario y agregarlo.
Necesidades de información	Debe conocer el nombre de usuario al que desea añadir a su lista de contactos de la aplicación y este debe existir en el sistema.
Funcionalidades necesarias	- Entrar a los contactos del usuario. - Añadir a otro usuario en contactos.
Desarrollo de las tareas	1. [Registrarse]. 2. Iniciar sesión. 3. Elegir la opción para acceder a los contactos. 4. Seleccionar la opción para añadir un usuario en contactos. 5. Indicar el nombre del usuario. 6. Agregar usuario a contactos.

Escenario de uso 6 – Buscar regalos hechos a una persona	
Perfil de usuario	Usuario Planificador Grupo. También el Usuario Planificador Individual Habitual en un contexto adaptado a este perfil.
Contexto en el que se encuentra	Un chico está gestionando en la aplicación los regalos de Navidad para su padre, conjuntamente a sus hermanos, y teme que al no tener muchas ideas entonces cometer el error de repetir algún regalo que ya le hayan hecho en el pasado así que revisará el historial y aprovechará también para buscar por el nombre de algún producto específico.
Objetivos que tiene	Encontrar que regalos le hizo en el pasado a una persona en concreto.
Tareas que lleva a cabo	Acceder al historial de planificaciones finalizadas, filtrar y ordenar el listado del historial y buscar un regalo.
Necesidades de información	Saber que filtros se quieren aplicar, por cuales conceptos se quiere ordenar y con cual palabra buscar.
Funcionalidades necesarias	- Acceder a historial de planificaciones finalizadas. - Filtrar los resultados. - Ordenar los resultados. - Buscar un regalo por una palabra en concreto.
Desarrollo de las tareas	1. Iniciar sesión. 2. Seleccionar la opción para listar las planificaciones realizadas. 3. Filtrar las planificaciones por el nombre de una persona. 4. Ordenar las planificaciones por fecha descendente. 5. Buscar un regalo indicando un concepto en la búsqueda. 6. Seleccionar la opción para visualizar una planificación.

Escenario de uso 7 – Marcar un regalo como favorito	
Perfil de usuario	Usuario Planificador Individual Esporádico y Usuario Planificador Individual Habitual.
Contexto en el que se encuentra	Una chica que a penas tiene tiempo libre entre el trabajo y estudiar empezó hace unos días la planificación para escoger un regalo para el día de la madre. Entonces añadió en la aplicación todas las ideas que se le ocurrieron y/o encontró navegando por internet mientras viajaba hacia el centro en el que asiste a clases nocturnas de formación profesional. Hoy quiere aprovechar otro de esos momentos para repasar desde su tablet todas las propuestas guardadas y destacar la que más le guste para comprarlo en cuanto pueda.
Objetivos que tiene	Resaltar como favorito uno de los regalos de los que hay añadidos en una planificación.
Tareas que lleva a cabo	Listar las planificaciones que tiene en curso, editar aquella en la que quiere hacer una modificación e indicar un regalo como favorito.
Necesidades de información	Saber la planificación individual en la que va a realizar una modificación y el regalo que va a marcar como favorito.
Funcionalidades necesarias	- Listar planificaciones en curso. - Editar una planificación individual. - Indicar un regalo como favorito.
Desarrollo de las tareas	1. Iniciar sesión. 2. Seleccionar la opción para listar las planificaciones en curso. 3. Seleccionar la opción de editar una planificación individual. 4. Marcar la opción de favorito en un regalo. 5. Guardar la planificación.

Escenario de uso 8 – Valorar un regalo en una planificación de regalos en grupo	
Perfil de usuario	Usuario Participante. También el Usuario Planificador Grupo en un contexto adaptado a su perfil.
Contexto en el que se encuentra	Hace una semana una mujer fue invitada al cuarenta cumpleaños de una amiga suya y una de las convidadas propuso de realizar los regalos conjuntamente. La planificadora de los regalos en la aplicación la agregó a contactos y la añadió como participante. La mujer utiliza poco el smartphone y en estos días a penas ha contribuido en la planificación pero al final se anima a al menos entrar a valorar las ideas que han sugerido el resto.
Objetivos que tiene	Aportar en el grupo su valoración de los regalos/ideas añadidos por todos los participantes y el planificador.
Tareas que lleva a cabo	Listar las planificaciones que tiene en curso, editar aquella en la que quiere hacer una modificación y valorar un regalo.
Necesidades de información	Saber la planificación colaborativa en la que va a realizar una modificación, el regalo que va a puntuar y la valoración que quiere dar al regalo.
Funcionalidades necesarias	- Listar planificaciones en curso. - Editar una planificación en grupo. - Valorar un regalo.
Desarrollo de las tareas	1. Iniciar sesión. 2. Seleccionar la opción para listar las planificaciones en curso. 3. Seleccionar la opción de editar una planificación en grupo. 4. Valorar un regalo de la planificación. 5. Guardar la planificación en grupo.

Escenario de uso 9 – Ver la lista de deseos de un usuario	
Perfil de usuario	Usuario Planificador Individual Habitual. También el Usuario Planificador Grupo y el Usuario Participante en un contexto adaptado a cada uno.
Contexto en el que se encuentra	Una chica que cursa el último año de instituto utiliza frecuentemente la aplicación para gestionar los regalos para su pareja, familia y amigos. Durante el día aprovecha cualquier momento libre para utilizar el smartphone. Está aburrida en el sofá mirando vídeos con la tablet y de repente recuerda que por la mañana ha agregado a un amigo suyo a la aplicación así que le entra la curiosidad de ir a ver que ha añadido en la lista de deseos.
Objetivos que tiene	Ver la lista de deseos de un usuario
Tareas que lleva a cabo	Acceder a los contactos agregados, visualizar el perfil de uno de los contactos y entrar a la lista de los regalos que desea ese contacto.
Necesidades de información	Saber a que usuario agregado en contactos se quiere ir a visualizar la lista de deseos.
Funcionalidades necesarias	- Entrar a los contactos del usuario. - Ver el perfil de un contacto. - Acceder a la lista de deseos del contacto.
Desarrollo de las tareas	1. Iniciar sesión. 2. Elegir la opción para acceder a los contactos. 3. Visualizar el perfil de un contacto. 4. Acceder a la lista de deseos del contacto.

Escenario de uso 10 – Poner al día una planificación de regalos en curso	
Perfil de usuario	Usuario Planificador Grupo. También el Usuario Planificador Individual Habitual en un contexto adaptado a este perfil.
Contexto en el que se encuentra	Un chico lleva unas semanas gestionando en la aplicación los regalos de Navidad para su padre, conjuntamente a sus hermanos, y durante este tiempo todos han ido aportando ideas y valorando cada una. Algunas de las propuestas a penas han gustado y otras tienen una valoración bastante alta así que borrará las ideas con menos puntuación y cambiará el estado de los regalos previstos como definitivos.
Objetivos que tiene	Poner al día la planificación colaborativa eliminando las ideas que menos interés han suscitado y cambiando el estado de los regalos mejor valorados y que tienen pendiente decidir si serán los definitivos.
Tareas que lleva a cabo	Listar las planificaciones que tiene en curso, editar aquella en la que quiere hacer una modificación, borrar un regalo y actualizar el estado de un regalo.
Necesidades de información	Saber la planificación colaborativa en la que va a realizar una modificación, conocer el regalo que va a quitar y también el que va a cambiar el grado de avance específico del regalo en el momento.
Funcionalidades necesarias	- Listar planificaciones en curso. - Editar una planificación en grupo. - Eliminar un regalo. - Modificar estado de un regalo.
Desarrollo de las tareas	1. Iniciar sesión. 2. Seleccionar la opción para listar las planificaciones en curso. 3. Seleccionar la opción de editar una planificación en grupo. 4. Seleccionar la opción de eliminar un regalo. 5. Cambiar el estado de un regalo. 6. Guardar la planificación.

2.1.2.2 Flujos de interacción

Los flujos de interacción representan visualmente las decisiones que tomarán y como interactuarán los usuarios con la aplicación para conseguir sus objetivos. Estos diagramas permiten la obtención de una visión global y pueden ayudar a detectar necesidades funcionales o de información de los usuarios en cualquiera de los puntos.

En la siguiente imagen se aprecian los flujos de interacción que muestran de forma gráfica la estructura general de la aplicación:

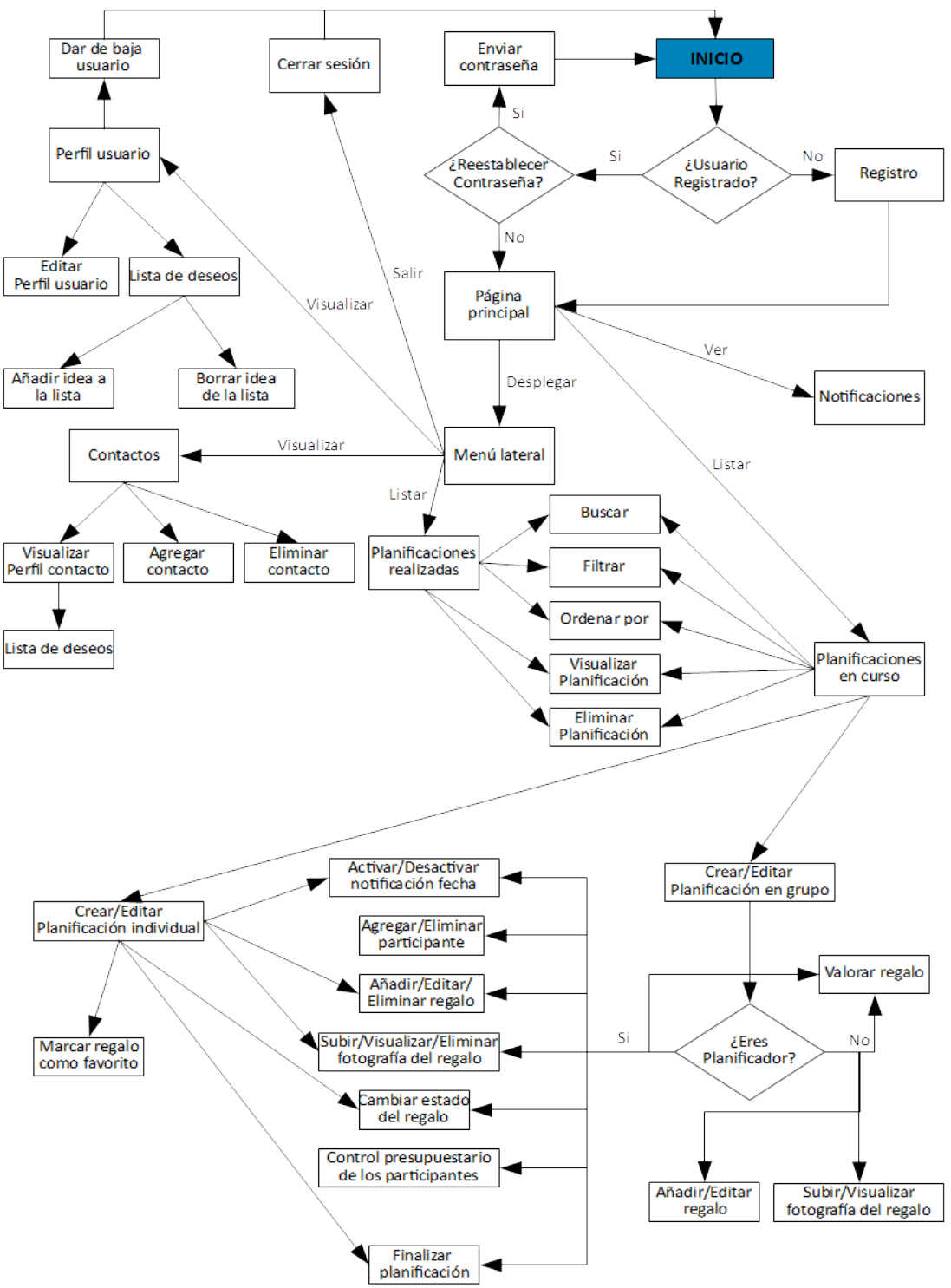


Figura 22: Flujos de interacción de la aplicación

2.1.3. Prototipado

La tercera fase del DCU consiste en diseñar los prototipos de la aplicación a partir de toda la información recopilada en los anteriores apartados, es decir, con los perfiles de usuario y los escenarios que simbolizan los resultados obtenidos en el análisis de usuarios y la definición de sus requisitos.

El prototipado es una de las fases claves en el proceso de conceptualización, diseño y desarrollo de una app ya que permite representar un modelo cercano al resultado final, comunicar decisiones de diseño y evaluarlo antes de desarrollar la app definitiva. Además, es posible modificarlo iterativamente de forma sencilla y económica incorporando mejoras fruto de la ejecución de pruebas que no se podrían realizar hasta que el proyecto estuviera totalmente acabado.

En este apartado se han llevado a cabo los sketches que son los primeros argumentos de diseño hechos a mano y, a continuación, los prototipos horizontales de alta fidelidad que son más cercanos al producto final y en soporte digital.

2.1.3.1 Sketches

Los sketches son prototipos de baja fidelidad que representan de forma esquemática la futura app. Se esbozan en papel sin detallar mucho las funciones que realizará cada sección. Proporcionan una primera idea de como será la interfaz y la disposición de sus elementos y así poder detectar posibles errores que son fácilmente corregibles en este punto. A continuación, se muestran los sketches escaneados.

Figura 23: Sketches I

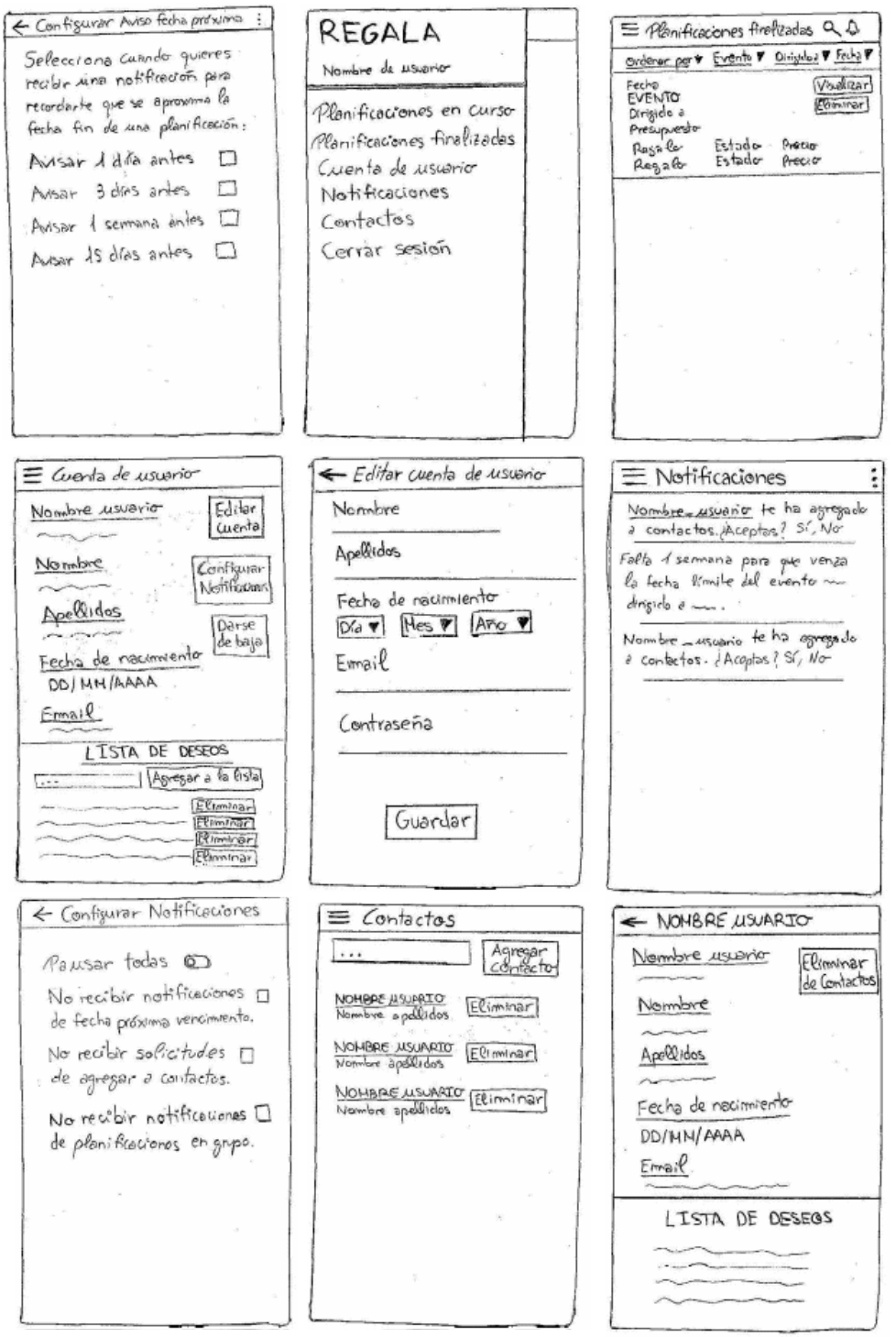


Figura 24: Sketches II

2.1.3.2 Prototipos horizontales de alta fidelidad

Los prototipos de alta fidelidad son una muestra digital de la app lo más cercana posible al sistema que se diseña y se desarrolla. Está basado en la construcción de un modelo que abarca todas o casi todas la funcionalidades, espacios, elementos y menús del producto sin que estén implementadas ni funcionen realmente. Proporcionan una forma más precisa de evaluar los aspectos funcionales y de usabilidad.

Seguidamente se exponen los prototipos horizontales de alta fidelidad de la app que han sido elaborados con el software **Justinmind Prototyper**. Estos son los prototipos finales actualizados con las mejoras obtenidas tras la realización del test de usuario en la siguiente fase de evaluación.

Las siguientes pantallas corresponden a la de inicio que permite acceder a la app si el usuario ya está registrado, restablecer la contraseña si el usuario la ha olvidado o si no está registrado entonces crear una cuenta nueva. También se puede observar la pantalla de registro para crear una cuenta nueva en la app y la pantalla para resetear la contraseña.

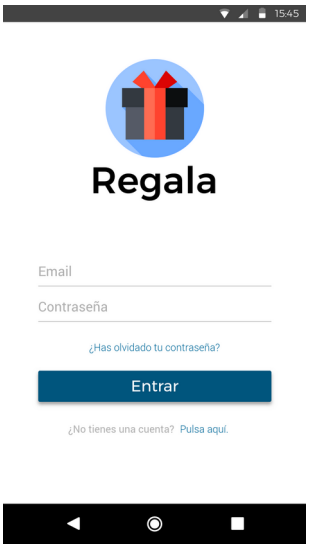


Figura 25: Pantalla del inicio de sesión

Figura 26: Pantalla de registro

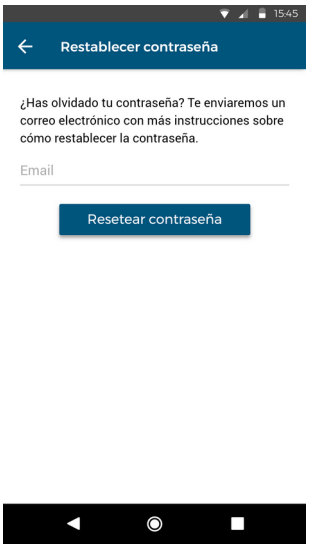


Figura 27: Pantalla para restablecer contraseña

A continuación, la primera pantalla es de las planificaciones que es como el control central de las planificaciones donde se pueden crear, editar, finalizar o eliminar. Respecto al sketch se ha decidido simplificar los botones de editar y eliminar por unos iconos más que conocidos por cualquier usuario habitual de Android. Las otras dos imágenes son la pantalla para crear una planificación individual donde se informan los datos mediante un formulario. El precio de los regalos (el total) se calcula automáticamente al añadir el precio de cada regalo. La pantalla para editar la planificación individual sería exactamente igual a la de crear pero con los datos previamente informados.

Figura 29: Pantalla de las Planificaciones

Figura 28: Pantalla para crear planificación individual I

Figura 30: Pantalla para crear planificación individual II

Las siguientes capturas de imagen son respectivamente la creación de una planificación en grupo y la configuración del recordatorio de fecha próxima de vencimiento de una planificación. Como se puede ver la creación de la planificación grupal es bastante similar a la individual excepto que aquí existe la opción de añadir participantes. El recordatorio de vencimiento se abre cuando se marca en "Recordatorio fecha próxima" para indicar al sistema cuantos días antes debe enviar una notificación.

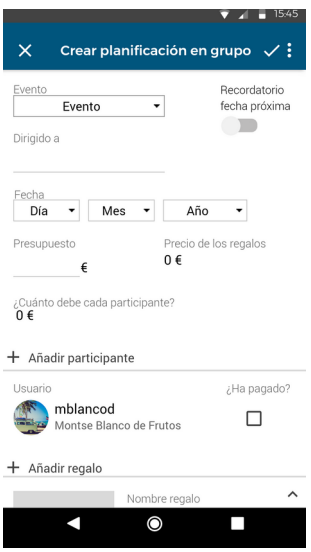


Figura 31: Pantalla para crear planificación en grupo I

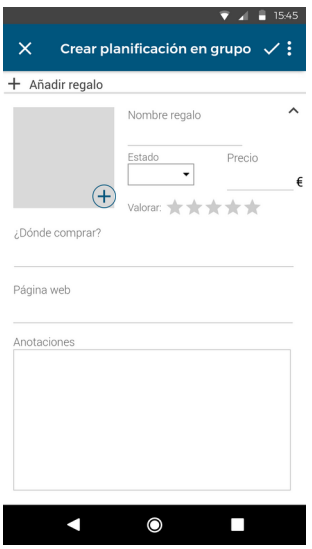


Figura 32: Pantalla para crear planificación en grupo II

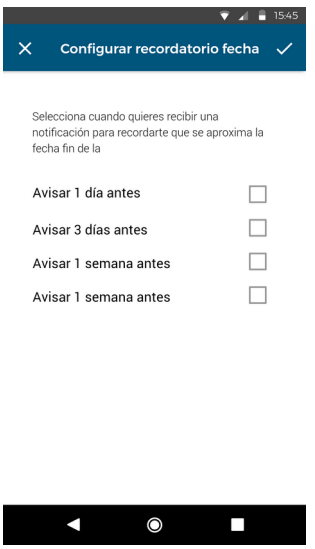


Figura 33: Pantalla para configurar recordatorio fecha

Las pantallas donde se visualiza una planificación individual o en grupo son también prácticamente idénticas. Si se pulsa el botón de arriba de los tres puntos entonces se abren las opciones de Editar, Eliminar y Finalizar la planificación.

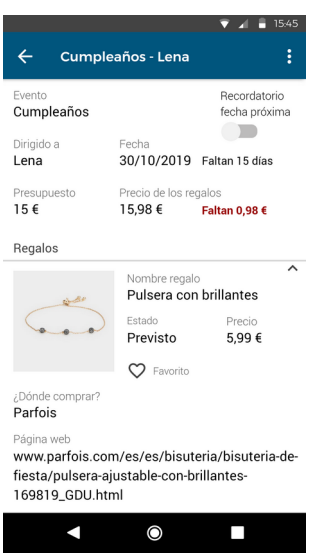


Figura 35: Pantalla para visualizar planificación individual I

Figura 36: Pantalla para visualizar planificación individual I

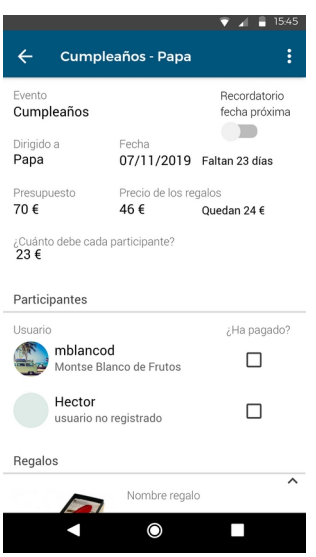


Figura 37: Pantalla para visualizar planificación en grupo I

Figura 34: Pantalla para visualizar planificación en grupo II

El menú lateral está disponible en toda la aplicación desde el inicio de sesión y ofrece un acceso rápido a todas las secciones de la aplicación así como la posibilidad de cerrar la sesión activa del usuario. La pantalla con el histórico de planificaciones es prácticamente idéntica a la pantalla de las planificaciones en curso, la diferencia es que aquí se muestran aquellas que ya están finalizadas. Estas se pueden visualizar y eliminar pero no editar. En la pantalla de la cuenta de usuario se muestra la información del perfil del usuario que ha accedido a la app. Aquí puede editar el nombre de usuario, el email, la contraseña, etc. pulsando en el lápiz. También se puede recomendar la app y darse de baja.

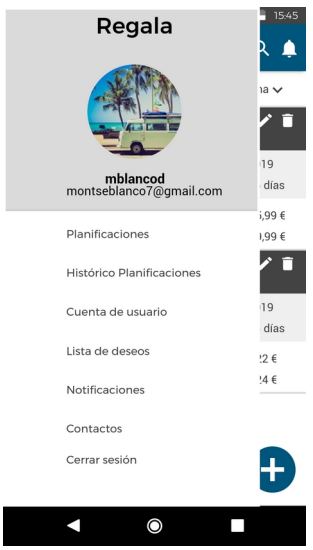


Figura 38: Pantalla del menú

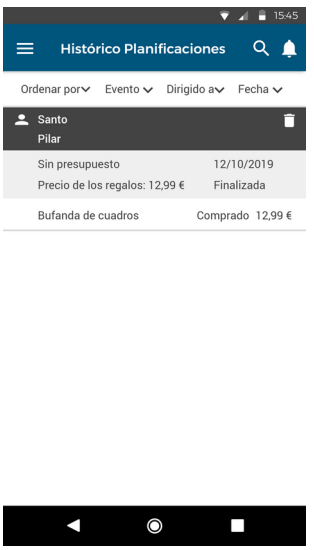


Figura 39: Pantalla del Histórico de Planificaciones

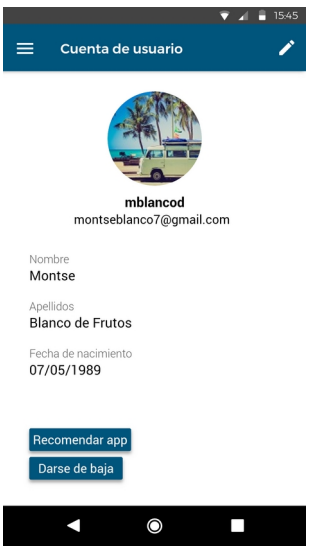


Figura 40: Pantalla de la cuenta de usuario

Figura 41: Pantalla para editar la cuenta de usuario

En la pantalla de la lista de deseos el usuario que ha iniciado sesión en la planificación puede añadir deseos en su propia lista que luego podrán visualizar los contactos que tiene agregados. También puede eliminar deseos de esa lista o buscar.

En la pantalla de notificaciones el usuario puede ver si alguien le ha enviado una solicitud para agregarlo a contactos, los avisos de fecha próxima de vencimiento de una planificación o si ha habido cambios en una planificación en grupo. Desde esta pantalla presionando en el botón de arriba de los tres puntos aparece la opción de acceder a configurar las notificaciones que es la última pantalla que se muestra a continuación. Ahí el usuario puede decidir si pausa todo el tipo de notificaciones que existen en el sistema o alguna en concreto.

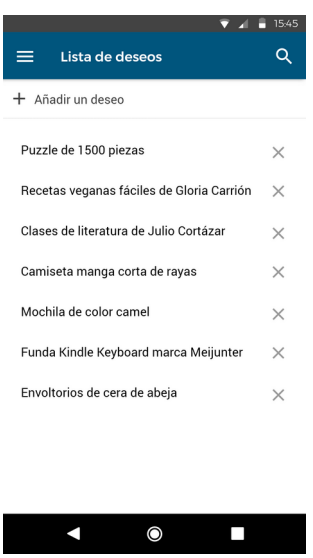


Figura 42: Pantalla de la lista de deseos propia

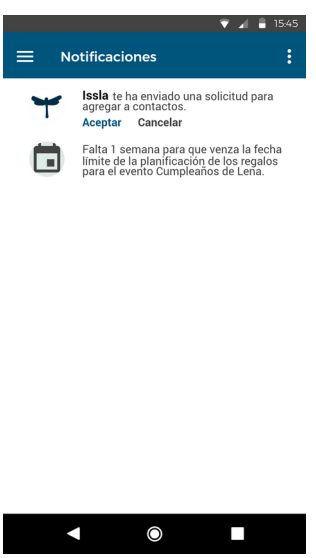


Figura 43: Pantalla de las Notificaciones

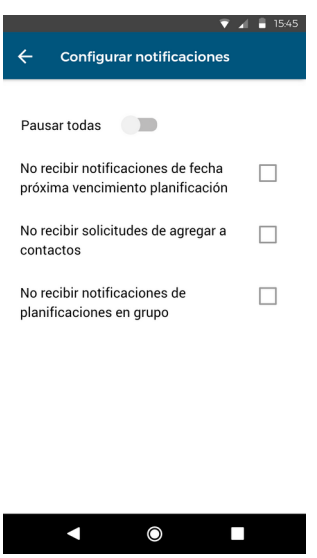


Figura 44: Pantalla para configurar las notificaciones

Los últimos prototipos corresponden a la pantalla de los contactos donde el usuario visualiza todos los usuarios que tiene agregados. Aquí puede utilizar el buscador para encontrar un contacto en concreto o agregar a un usuario. Al pulsar sobre un contacto agregado entonces se abre el perfil de usuario donde por un lado se visualiza su información y por otro la lista de deseos (si es que tiene) con un buscador disponible.

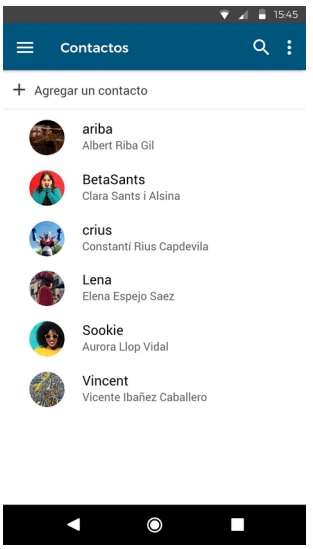


Figura 45: Pantalla de los contactos

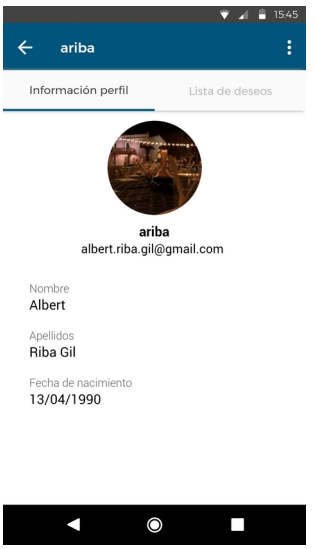


Figura 46: Pantalla de la información de perfil de un contacto

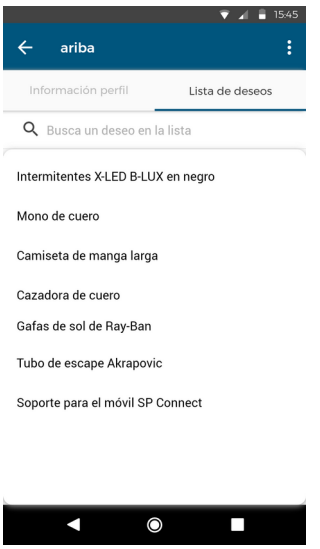


Figura 47: Pantalla de la lista de deseos de un contacto

En general se ha decido optar por un diseño claro y sencillo inspirado en el minimalismo y el principio de "menos es más". Todas las pantallas son de fondo blanco para dar esa sensación de pulcro y limpio que además facilita la visualización de los datos. También se han utilizado dos colores más, un tono azul en concreto para la barra superior de la app y algunos botones y varios tonos de gris tanto para diferenciar secciones, textos, iconos, etc. En cuanto a los textos fuente sólo se han utilizado dos, la Montserrat especialmente para títulos y la Roboto sobre todo para datos, ambas en diferentes tamaños de letra. Además, se ha procurado cargar el mínimo posible las pantallas y utilizar iconos típicos en Android que cualquier usuario habitual conocerá para así facilitar el uso de la app.

2.1.4. Evaluación

La cuarta y última fase del DCU es la planificación y ejecución de la evaluación de los prototipos diseñados en la anterior fase para obtener información sobre ellos y de esta forma mejorarlos y/o validarlos. Cabe recordar que el proceso de DCU es iterativo y, por tanto, es necesario ir evaluando los diseños y corrigiendo los errores de forma iterativa hasta alcanzar el mejor de los productos posibles.

El método de evaluación utilizado en este proyecto es la técnica del test con usuarios que permite hacer pruebas y comprobaciones de los prototipos con los usuarios. Se comprueba si las interfaces y los procesos están bien diseñados y si los usuarios encuentran problemas y dificultades para llevar a cabo sus objetivos además de evaluar su grado de satisfacción. Así que en los siguientes apartados se definirá la forma en que se evalúan los prototipos utilizando esta técnica.

2.1.4.1 Perfil socio-demográfico de los usuarios

Para llevar a cabo los test de usuario es importante contar con una muestra representativa de usuarios potenciales del producto para evitar obtener información sesgada. Por tanto, es necesario identificar las características que deben cumplir los usuarios como la edad, el género y su destreza tecnológica para reclutar a usuarios de un tipo de perfil concreto y así obtener la información que deseamos.

En las anteriores fases se ha visto que la aplicación a desarrollar en este proyecto no tiene un público objetivo muy marcado. Se ha determinado que los usuarios potenciales de la aplicación tienen una edad comprendida entre los 26 y 50 años aunque puede utilizarla personas de cualquier edad. Pueden ser de cualquier género, lugar y condición. Aunque como mínimo sí deben saber español ya que la app sólo estará disponible en ese idioma, es necesario que suelan hacer regalos y que tengan unos mínimos conocimientos de uso de dispositivos móviles para que sean capaces de desenvolverse con cierta soltura. Por tanto, se descartarán aquellas personas que no cumplan estos mínimos a partir de la realización del siguiente cuestionario inicial:

Cuestionario inicial	
Rango de edad:	[ ] Menor de 18 años [ ] 18 – 25 años [ ] 26 – 38 años [ ] 39 – 50 años [ ] Mayor de 50 años
Idiomas conocidos:	
Nivel de experiencia con los dispositivos móviles:	[ ] No uso [ ] Principiante [ ] Medio [ ] Alto [ ] Experto
Haces regalos:	[ ] Sí [ ] No

2.1.4.2 Preguntas sobre el usuario (pre-test)

Antes de la realización del test, los usuarios deben cumplimentar un cuestionario para obtener información sobre ellos como datos socio-demográficos, uso de dispositivos móviles y la planificación de regalos para así conocer mejor a cada usuario y tener en cuenta sus necesidades y preferencias en el proceso de recopilación y análisis de los resultados:

Cuestionario Pre-test
¿Cuál es tu nombre?
¿Qué edad tienes?
¿A qué te dedicas?
¿Qué dispositivos móviles utilizas?
¿Utilizas frecuentemente el teléfono móvil durante el día? ¿En qué lugares?
¿Qué aplicaciones móviles sueles utilizar?
¿Haces regalos? ¿En qué ocasiones?
¿Los planificas a tiempo? ¿Cómo te planificas desde la primera idea hasta el día que los das?
¿Te marcas un presupuesto? ¿Y haces una primera recopilación de ideas?
¿Pides lista de deseos a las personas a las que quieres hacer un regalo?
¿Haces regalos en grupo? ¿Has organizado alguno, participado y/o ambos?
¿Como os organizáis en los regalos colaborativos?

2.1.4.3 Tareas a realizar

Después de responder al cuestionario pre-test, se explica a los usuarios la app que se va a analizar y se indica las tareas que deben ejecutar sobre los prototipos de la aplicación para probar el diseño de las funcionalidades principales de la aplicación. Durante la prueba se irá anotando en otra tabla si el usuario ha sido capaz de llevar a cabo la tarea o no y comentarios ya sean tanto del usuario como del moderador del test.

Tareas a realizar
1. Registrate en la aplicación.
2. Crea una planificación individual para el día de la madre y añade varios regalos con el estado “idea”.
3. Cambia el email con el que te has registrado en la aplicación.
4. Te ha llegado una notificación para aceptar la solicitud que te ha enviado la usuaria Issla para agregarte a contactos. Aceptala.
5. Crea una planificación en grupo para el cumpleaños de una amiga, con dos participantes: una es la usuaria Issla y el otro es un amigo que no está registrado en la aplicación. Añade un regalo adjuntando una fotografía.
6. Pausa todas las notificaciones del sistema.

7. Visualiza el perfil de usuario del contacto que has aceptado la solicitud de agregar a contactos.
8. Busca en las planificaciones en curso la planificación en grupo que has creado antes suponiendo que tienes bastantes planificaciones en marcha y no se encuentra a simple vista.
9. Una vez encontrada esa planificación en grupo valora uno de los regalos.
10. Finaliza la planificación de regalos que creaste para el día de la madre.
11. Elimina al usuario crius ya que sólo lo agregaste para hacer una planificación de regalos en grupo.
12. Visualiza la planificación del día de la madre que has finalizado anteriormente.

2.1.4.4 Preguntas referentes a las tareas (post-test)

Después de la sesión de test de las tareas por parte del usuario se le deja cierto tiempo para que exprese su opinión sobre la app. Para ello se elabora un cuestionario que ayudará a guiar las preguntas sobre la experiencia y conocer la percepción que ha tenido con la app y el grado de satisfacción del mismo.

Cuestionario post-test						
En una escala del 0 al 5, siendo 0 horrible y 5 excelente.	**0**	**1**	**2**	**3**	**4**	**5**
¿Qué te ha parecido la estética y el diseño?						
¿Crees que ha sido fácil utilizar la app?						
¿Qué te han parecido los contenidos y la información?						
¿Te ha parecido de utilidad la app?						
¿Qué es lo que más te ha gustado de la app?						
¿Qué es lo que menos te ha gustado de la app?						
¿Qué tareas te han parecido más complicadas? ¿Por qué?						
¿Algo te ha parecido confuso?						
¿Has sabido en todo momento en que pantalla estabas?						
¿Cambiarías alguna cosa de la app?						
¿Añadirías o eliminarías alguna funcionalidad?						

2.1.4.5 Test con un usuario, análisis y mejoras

Debido al ajustado tiempo de planificación del proyecto solamente se realiza el test de usuario a una única persona a la que se la he hecho previamente el cuestionario inicial para ver si realmente cumplía las mínimas características del perfil esperado. Antes de empezar con el test se intenta crear un ambiente cercano y confortable para que el usuario no se ponga nervioso al sentirse observado ante la persona que actúa como facilitadora/evaluadora, en este caso, la autora de este TFM. Así que primero se interactúa un poco con él explicándole el objetivo de la prueba, que es evaluar la calidad de uso de la app que se desea desarrollar. Se le deja bien claro que no se le está evaluando a él y que si comete algún error durante el test no pasa nada, no es culpa suya sino del diseño.

Posteriormente, se le explica en qué consiste el test de usuario y las partes que tiene (el pre-test, las tareas y el post-test). También se le insiste en qué es muy importante que vaya diciendo que piensa mientras ejecuta las tareas, si algo le está llamando la atención, qué le motiva a ejecutar cada acción, si hay algo que no comprende, etc. Es especialmente significativo escuchar sus opiniones y todo lo que se le pase por la cabeza. Además, debe quedar muy claro que la función de la facilitadora es la de observar, en ningún momento puede ayudar al usuario.

Después, se le entrega la carta de autorización para que la firme y así dar su consentimiento para poder almacenar y tratar los datos que se recogen durante el test. Seguidamente se le realizan las preguntas del cuestionario pre-test para conocer lo máximo posible al usuario. Una vez finalizado entonces se le explica el producto que se va a analizar, las tareas que deberá llevar a cabo y se muestran los prototipos indicándole que no puede interactuar con ellos como si fuera una aplicación real así que en este caso si que deberemos actuar para simular las acciones interactivas. Si mientras realiza las tareas se percibe que el usuario no sabe finalizar una tarea, se le comentará que no hay problema y se pasará a la siguiente y así hasta finalizarlas todas.

Finalmente, una vez el usuario ha acabado las tareas entonces puede pasar a responder el cuestionario post-test para evaluar la opinión que tiene de la app y para que explique cuáles han sido sus percepciones y sensaciones durante la prueba. Es importante aprovechar el momento para obtener todos los comentarios u observaciones posibles.

Todos los cuestionarios cumplimentados por el usuario y los comentarios extraídos de la realización de las tareas por parte del usuario se encuentran en los **Anexos**.

A continuación se exponen las conclusiones obtenidas y las mejoras determinadas para corregir y afinar los prototipos horizontales de alta fidelidad.

- Al usuario le ha gustado bastante la aplicación puntuando como excelente la facilidad de uso y los contenidos y la información proporcionados. De hecho ha comentado que es bastante intuitiva. En cuanto a la estética y diseño le ha gustado bastante aunque hay cierto margen de mejora (un 4 sobre 5). En cuanto al nivel de utilidad también lo ha puntuado bastante alto, un 4 sobre 5. Y destaca en concreto el poder gestionar regalos en grupo.
- Encuentra sobrecargada la pantalla de planificaciones con los botones tan grandes.
- Echa en falta el poder cargar una imagen cuando crea la cuenta de usuario y cambiarla al editar la cuenta, además de modificar el nombre de usuario.
- En los contactos comenta que estaría bien disponer de un buscador en los contactos para encontrar a usuarios más fácilmente. También falta la opción de recomendar la app o invitar a personas a utilizarla.
- En todo momento sabía en que pantalla se encontraba aunque ha reaccionado confuso cuando no estaba en la pantalla de planificaciones y en el menú no sabía exactamente donde ir para crear una planificación nueva. Al final ha ido a “Planificaciones en curso” pero ha recomendado modificar los títulos de las Planificaciones.
- No ha tenido problema para utilizar el buscador y los filtros en las planificaciones pero no sabía diferenciar si una planificación es individual o en grupo ya que aparece exactamente la misma información.
- Otros cambios y nuevas funcionalidades que sugieren son poner la lista de deseos propia a parte ya que la cuenta de usuario queda demasiado sobrecargada. También en visualizar planificación sería muy cómodo poder expandir o esconder los detalles de cada regalo en vez de tener que ir uno por uno.

Y se han determinado las siguientes mejoras para corregir y perfeccionar los prototipos horizontales de alta fidelidad:

- Añadir una imagen de usuario cuando se registra un usuario y, por tanto, los perfiles de usuario contarán con foto que se podrá cambiar. También se podrá modificar el nombre de usuario.
- En general, en la app intentar utilizar menos botones grandes que sobrecargan la pantalla por iconos propios de Android que cualquier usuario habitual conoce. Como el lápiz para modificar, la papelera para borrar, la flecha para tirar atrás, el visto para crear/guardar, la X para cancelar, los tres puntos para más opciones, etc.
- En la pantalla de Planificaciones y el Histórico añadir iconos para diferenciar si una planificación es individual o grupal, por ejemplo, utilizar un icono de un sólo usuario o uno de dos usuarios.
- Cambiar los títulos tanto en el menú como en las pantallas de “Planificaciones en curso” a “Planificaciones” y de “Planificaciones finalizadas” a “Histórico Planificaciones”.
- Dentro de una planificación en concreto añadir en más opciones que se puedan expandir o esconder los detalles de cada regalo en vez de tener que ir uno por uno.
- Sacar la lista de deseos de la cuenta de usuario y hacer una pantalla única para esto a la cual se pueda acceder desde el menú lateral. También se pondrá a parte la lista de deseos cuando se visualice el perfil de un contacto.
- Añadir un buscador en la lista de contactos para encontrar a usuarios más fácilmente.

2.2. Diseño Técnico

Una vez finalizada la fase de Diseño Centrado en el Usuario se ha obtenido diversa información como los escenarios de uso, los flujos de interacción y los prototipos que son valiosos para elaborar el Diseño Técnico. Este apartado engloba la definición de los casos de uso de la aplicación que facilitan la descripción de como se usará el sistema y las funcionalidades de las que dispondrá. También consta del diseño de la arquitectura identificando las entidades de la base de datos, las clases y objetos que se utilizarán para gestionar los diferentes procesos y la estructura API de comunicación entre el cliente y el servidor.

2.2.1. Definición de algunos casos de uso

Los casos de uso son descripciones de una actividad que realiza el usuario interactuando con la aplicación. Como se ha comentado, los casos de uso sirven para establecer las funcionalidades de la app a partir del estudio hecho en las etapas anteriores y son esenciales para el buen desempeño del producto final.

Los diferentes casos de uso del sistema se pueden agrupar en cinco paquetes diferentes:

- **Gestión de usuarios:** Registro de usuario, Cargar fotografía de perfil, Inicio de sesión, Restablecer contraseña, Visualizar la cuenta de usuario, Editar el perfil de usuario, Cambiar la contraseña, Eliminar la cuenta y Cerrar la sesión de usuario.
- **Gestión de las planificaciones:** Acceder a Planificaciones, Acceder a Histórico de Planificaciones, Refrescar pantalla planificaciones/histórico, Buscar, ordenar y/o filtrar las planificaciones, Crear una planificación, Visualizar una planificación, Editar una planificación, Finalizar una planificación, Eliminar una planificación, Añadir un regalo, Cambiar el estado de un regalo, Cargar fotografía de un regalo, Marcar un regalo como favorito, Valorar un regalo, Eliminar un regalo, Editar grupo, Visualizar grupo, Añadir participantes, Marcar que un participante ha pagado, Quitar un participante y Activar o desactivar el recordatorio de fecha próxima.
- **Gestión de la lista de deseos propia**: Visualizar la lista de deseos, Añadir un deseo, Borrar un deseo y Buscar un deseo.
- **Gestión de los contactos**: Visualizar los contactos, Añadir contactos, Buscar un usuario, Eliminar un contacto, Buscar un contacto, Visualizar el perfil de usuario de un contacto, Visualizar la lista de deseos de un contacto y Recomendar la app.
- **Gestión de las notificaciones**: Visualizar las notificaciones y Configurar las notificaciones.

A continuación se definen los principales casos de uso relacionados con la gestión de usuarios y la gestión de las planificaciones. El resto de casos de uso relacionados con la gestión de usuarios, las planificaciones, la lista de deseos propia del usuario, las notificaciones y los contactos se pueden ver en un documento con todos los casos de uso de la aplicación en los **Anexos**. Debido a su extensión se ha visto conveniente mostrar en esta memoria sólo una parte y la totalidad en otro documento.

Identificador	CU-001
Nombre	Registro de usuario.
Prioridad	Alta
Descripción	Dar de alta un nuevo usuario en el sistema.
Actores	Usuario y Sistema
Pre-Condiciones	* El usuario debe tener instalada la aplicación en su dispositivo móvil y debe tener acceso a internet. * No debe existir previamente una cuenta en el sistema con el mismo nombre de usuario y/o email.
Flujo	1. Abrir la aplicación. 2. Pulsar en la opción para ir a crear una nueva cuenta. 3. Rellenar el formulario de registro y/o adjuntar una fotografía para el usuario. 4. Pulsar en “Crear cuenta”. 5. El sistema valida la información introducida para comprobar que ese usuario no esté registrado ya en la app.

	6. El sistema crea el nuevo usuario con los datos introducidos, guarda los datos del usuario en el backend, almacena la fotografía en el storage del backend y redirige al usuario a la página principal que es la de las Planificaciones aunque estará vacía.
Excepciones	Se muestra un mensaje de error en la pantalla si se dan las siguientes excepciones: 5.1. Si el usuario no ha informado todos los campos obligatorios. 5.2. Si ya existe una cuenta de usuario con el mismo nombre de usuario y/o email. 5.3. Si el usuario tiene menos de 13 años entonces se informa de la edad mínima necesaria para registrarse en la aplicación. 5.4. El sistema también valida que el email informado sea válido y las contraseñas informadas coincidan. 5.5. Se produce un error al almacenar los datos del usuario en el backend.
Post-Condiciones	El sistema crea una nueva cuenta de usuario.
Notas	Al registrarse en la aplicación el sistema inicia la sesión automáticamente con el nuevo usuario.

Identificador	**CU-002**
Nombre	Cargar fotografía de perfil.
Prioridad	Media
Descripción	Cargar una fotografía en el perfil de usuario.
Actores	Usuario y Sistema
Pre-Condiciones	* Mientras se está realizando el caso de uso CU-001 – Registro de usuario. * O mientras se está realizando el caso de uso CU-006 – Editar el perfil de usuario.
Flujo	1. Pulsar en la fotografía de perfil o En el texto “Añadir una foto de perfil”. 2. Escoger si se hace la foto con la cámara o se selecciona una imagen de la galería de fotos. 3. Hacer la fotografía o escoger una de la galería. 4. El sistema carga la fotografía hecha o la imagen seleccionada en la vista de la fotografía de perfil.
Excepciones	3.1. Se muestra un mensaje de error en la pantalla si se da un error al recuperar la fotografía hecha o seleccionada.
Post-Condiciones	El sistema muestra la fotografía de perfil.
Notas	La fotografía de perfil se almacena en el storage del backend cuando se crea la cuenta.

Identificador	**CU-003**
Nombre	Inicio de sesión.
Prioridad	Alta
Descripción	Iniciar sesión para acceder a la aplicación por parte de un usuario previamente registrado.
Actores	Usuario y Sistema
Pre-Condiciones	* Realizado el caso de uso CU-001 – Registro de usuario. * Recordar el email y la contraseña.
Flujo	1. Abrir la aplicación. 2. Introducir el email y la contraseña de la cuenta. 3. Pulsar en “Entrar”. 4. El sistema valida las credenciales indicadas. 5. El sistema obtiene del backend toda la información del perfil de usuario. 6. El sistema redirige al usuario a la página principal que es la de las Planificaciones.
Excepciones	4.1. Si el usuario no ha informado todos los campos obligatorios entonces el sistema muestra un mensaje de error informativo en la pantalla. 4.2. Si el sistema no valida con éxito las credenciales indicadas entonces muestra un mensaje de error en la pantalla informando de ello.
Post-Condiciones	El usuario consigue iniciar sesión en la app.

Notas	N/A.

Identificador	**CU-004**
Nombre	Restablecer contraseña.
Prioridad	Media
Descripción	Recuperar la contraseña de la cuenta de usuario para poder acceder a la aplicación.
Actores	Usuario y Sistema
Pre-Condiciones	* Realizado el caso de uso CU-001 – Registro de usuario. * El usuario no ha iniciado sesión porque no recuerda la contraseña.
Flujo	1. Abrir la aplicación. 2. Pulsar en la opción de en caso de haber olvidado la contraseña. 3. Introducir el email de la cuenta de usuario. 4. Pulsar en "Resetear contraseña". 5. El sistema envía un correo electrónico al email indicado con las instrucciones para restablecer la contraseña.
Excepciones	5.1. Si el sistema no valida que el email indicado pertenece a una cuenta de usuario existente entonces muestra un mensaje de error en la pantalla para informar al usuario.
Post-Condiciones	* El sistema redirige al usuario a la pantalla de inicio de sesión. * El usuario recibe un correo electrónico en el email indicado con las instrucciones para reiniciar la contraseña.
Notas	El correo electrónico enviado tendrá un tiempo concreto de validez como medida de seguridad.

Identificador	**CU-006**
Nombre	Editar el perfil de usuario.
Prioridad	Baja
Descripción	Editar los datos del perfil de usuario (nombre de usuario, nombre y apellidos, fecha de nacimiento y email).
Actores	Usuario y Sistema
Pre-Condiciones	Realizado el caso de uso CU-005 – Visualizar la cuenta de usuario.
Flujo	1. Pulsar sobre la opción para editar el perfil de usuario. 2. Modificar los datos del perfil de usuario. También se puede modificar la fotografía de usuario. 3. Pulsar en el botón del visto para guardar los cambios. 4. El sistema valida los datos introducidos. 5. El sistema almacena los datos del perfil de usuario en el backend. 6. El sistema redirige de nuevo a la visualización de la cuenta de usuario.
Excepciones	4.1. Si el usuario no ha informado todos los campos obligatorios entonces el sistema muestra un mensaje de error informativo en la pantalla. 4.2. Si se modifican el nombre de usuario y/o el email y ya existe una cuenta de usuario con esos mismos datos, el sistema avisa de ello mediante un mensaje de error en la pantalla. 4.3. Si al modificar la fecha de nacimiento se detecta que el usuario tiene menos de 13 años entonces el sistema muestra un mensaje de error informando de la edad mínima necesaria para utilizar la aplicación.
Post-Condiciones	El usuario visualiza de nuevo la cuenta de usuario con los datos actualizados si corresponde y tiene acceso a la gestión de su cuenta.
Notas	Dentro de la edición de la cuenta de usuario está la opción de cancelar por si al final no se desea modificar nada.

Identificador	CU-007
Nombre	Cambiar la contraseña.
Prioridad	Media
Descripción	Modificar la contraseña con la que se accede a la aplicación.
Actores	Usuario y Sistema
Pre-Condiciones	* Realizado el caso de uso CU-005 – Visualizar la cuenta de usuario. * El usuario debe recordar su contraseña actual de acceso.
Flujo	1. Pulsar sobre la opción para cambiar la contraseña. 2. Informar la contraseña actual, la contraseña nueva y repetir la contraseña nueva. 3. Pulsar en el botón del visto para guardar los cambios. 4. El sistema valida los datos introducidos. 5. El sistema redirige de nuevo a la visualización de la cuenta de usuario.
Excepciones	4.1. Si la contraseña actual informada no coincide con la contraseña de acceso del usuario entonces el sistema muestra un mensaje de error informativo en la pantalla. 4.2. Si la nueva contraseña no es diferente a la actual, el sistema avisa de ello mediante un mensaje de error en la pantalla. 4.3. El sistema también valida que la nueva contraseña y la repetida informadas coincidan y tengan como mínimo 6 caracteres sino avisa de ello mediante un mensaje de error en la pantalla.
Post-Condiciones	Si procede la contraseña se habrá cambiado pero no se podrá comprobar hasta salir de la sesión.
Notas	Dentro de la edición de la contraseña está la opción de cancelar por si al final no se desea modificar nada.

Identificador	CU-010
Nombre	Acceder a Planificaciones.
Prioridad	Alta
Descripción	Acceder a la pantalla de Planificaciones que es como el control central de la app.
Actores	Usuario y Sistema
Pre-Condiciones	Realizado el caso de uso CU-003 – Inicio de sesión.
Flujo	1. Abrir el menú lateral. 2. Pulsar sobre la opción de Planificaciones. 3. El sistema obtiene del backend toda la información asociada a las planificaciones en curso del usuario tanto como planificador como participante. 4. El sistema abre la pantalla de Planificaciones donde se muestra un resumen de todas las planificaciones en curso obtenidas previamente.
Excepciones	N/A.
Post-Condiciones	En la pantalla se visualiza un listado de las planificaciones en curso o vacía en el caso que el usuario no tenga o no participe en ninguna.
Notas	En Planificaciones el usuario puede crear nuevas planificaciones y buscar, ordenar, filtrar, visualizar, editar, finalizar y eliminar aquellas que están en curso (aún no se han finalizado). Como participante el usuario sólo puede visualizar y editar.

Identificador	CU-011
Nombre	Acceder a Histórico de Planificaciones
Prioridad	Media
Descripción	Acceder a la pantalla de Histórico de Planificaciones donde se muestran las planificaciones finalizadas.
Actores	Usuario y Sistema

Pre-Condiciones	Realizado el caso de uso CU-003 – Inicio de sesión.
Flujo	1. Abrir el menú lateral. 2. Pulsar sobre la opción de Histórico Planificaciones. 3. El sistema obtiene del backend toda la información asociada a las planificaciones finalizadas del usuario tanto como planificador como participante. 4. El sistema abre la pantalla Histórico de Planificaciones donde se muestra un resumen de todas las planificaciones finalizadas obtenidas previamente.
Excepciones	N/A.
Post-Condiciones	En la pantalla se visualiza un listado de las planificaciones finalizadas o vacía en el caso que el usuario no tenga ninguna finalizada o no haya participado en ninguna.
Notas	En el Histórico de Planificaciones se puede buscar, ordenar, filtrar, visualizar y eliminar planificaciones finalizadas. Como participante el usuario sólo puede visualizar.

Identificador	**CU-014**
Nombre	Crear una planificación.
Prioridad	Alta
Descripción	Crear una nueva planificación, individual o en grupo, en el sistema introduciendo la información requerida.
Actores	Usuario y Sistema
Pre-Condiciones	* Realizado el caso de uso CU-003 – Inicio de sesión * O el caso de uso CU-010 – Acceder a Planificaciones. * Saber al menos para que evento es y a quién va dirigida la planificación.
Flujo	1. Pulsar sobre el botón de añadir. 2. Presionar la opción de crear una nueva planificación (Individual o en grupo). 3. Introducir al menos los datos obligatorios para crear la nueva planificación. 4. Pulsar en el botón del visto para crear. 5. El sistema comprueba la información introducida por el usuario. 6. Se crea la nueva planificación en el sistema, también sus regalos (si tiene) y los participantes (si es en grupo) y se almacenan los datos en el backend. 7. El sistema redirige al usuario a la página de Planificaciones.
Excepciones	Se muestra un mensaje de error en la pantalla si se dan las siguientes excepciones: 5.1. El usuario no ha informado todos los campos obligatorios (evento, persona a la que va dirigida…). 5.2. Se produce un error al almacenar la nueva planificación en el backend.
Post-Condiciones	* El sistema crea la nueva planificación introducida por el usuario. * El usuario es redirigido a la pantalla de Planificaciones donde aparece la nueva planificación creada.
Notas	Se pueden crear planificaciones individuales o en grupo. Tanto en la individual como en la colaborativa es obligatorio indicar el evento y a quien van dirigidos los regalos. La fecha es opcional aunque es recomendable especificarla. Al indicar una fecha en la ficha de la planificación se muestran los días que quedan hasta la fecha indicada. También se pueden añadir regalos o editarlos una vez se han añadido en la sección de regalos de la planificación, se les puede cambiar los datos, actualizar el estado, añadir una fotografía del regalo o cambiarla.

Identificador	CU-015
Nombre	Visualizar una planificación.
Prioridad	Media
Descripción	Visualizar toda la información al detalle de una planificación, individual o en grupo, tanto si está en curso como finalizada.
Actores	Usuario y Sistema
Pre-Condiciones	* Realizado el caso de uso CU-002 – Inicio de sesión. * O el caso de uso CU-010 – Acceder a Planificaciones si se trata de una planificación en curso. * O el caso de uso CU-011 – Acceder a Histórico de Planificaciones si se trata de una planificación finalizada. * Conocer al menos el evento, persona y/o fecha de la planificación que se desea visualizar. * La planificación a visualizar debe existir en el sistema.
Flujo	1. Localizar en el listado de planificaciones aquella que se desea visualizar ya sea utilizando una búsqueda, ordenando, filtrando o mirando todo el listado. 2. Acceder a visualizar la planificación pulsando sobre el nombre del evento o la persona. También se puede pulsando en más opciones y después en Visualizar. 3. El sistema obtiene del backend los participantes en el caso de las planificaciones en grupo. Los datos de la planificación y sus regalos ya se han recuperado previamente. 4. Se muestra en pantalla la planificación y sus regalos (si tiene).
Excepciones	N/A.
Post-Condiciones	Se mostrará en pantalla toda la información asociada de una planificación a la que el usuario ha accedido a visualizar.
Notas	El sistema muestra cuantos días faltan para llegar a la fecha indicada en la planificación. El precio total de los regalos se calcula automáticamente y si se ha especificado un presupuesto el sistema también calcula automáticamente cuanto dinero queda del presupuesto respecto al total de los regalos. Desde el modo visualización, pulsando en más opciones, se abre un menú para acceder a editar, finalizar o eliminar la planificación. En el caso de las planificaciones colaborativas hay un botón para acceder a visualizar los datos del grupo y los participantes sólo pueden editar la planificación, nunca finalizar ni eliminar.

Identificador	CU-016
Nombre	Editar una planificación.
Prioridad	Alta
Descripción	Modificar una planificación, individual o en grupo, sólo si está en curso.
Actores	Usuario y Sistema
Pre-Condiciones	* Realizado el caso de uso CU-003 – Inicio de sesión. * O el caso de uso CU-010 – Acceder a Planificaciones. * O el caso de uso CU-015 – Visualizar una planificación. * Conocer al menos el evento, persona y/o fecha de la planificación que se desea modificar. * La planificación a editar debe existir en el sistema y no puede estar finalizada.
Flujo	1.1. Localizar en el listado de planificaciones aquella que se desea modificar ya sea utilizando una búsqueda, ordenando, filtrando o mirando todo el listado. 1.2. O estar visualizando la planificación en concreto. 2.1. Pulsar en el botón de más opciones de la planificación y se abre un menú donde está la opción de editar. 2.2. También se puede acceder a modificar la planificación cuando se está visualizando. Pulsar en el botón de más opciones y se abre un menú donde está la opción de editar. 3. El sistema obtiene del backend los participantes en el caso de las planificaciones en grupo. Los datos de la planificación y sus regalos ya se han recuperado previamente y se muestran en pantalla. 4. El usuario realiza las modificaciones pertinentes como actualizar algún dato, añadir regalos, etc. 5. Pulsar en el icono del visto para guardar los cambios. 6. Se actualizan los datos de la planificación en el sistema, también sus regalos (si tiene) y los participantes (si es en grupo) y se almacenan los datos en el backend. 7. El usuario es redirigido a la pantalla de Planificaciones.

Excepciones	5.1. Se produce un error al almacenar las modificaciones de la planificación en el backend por lo que se muestra un mensaje de error en pantalla.
Post-Condiciones	Se actualiza la planificación con la información especificada por el usuario.
Notas	El usuario puede cancelar la edición de la planificación pulsando el icono de la X. Al indicar una fecha en la ficha de la planificación se muestran los días que quedan hasta la fecha indicada. También se pueden añadir regalos o editarlos una vez se han añadido en la sección de regalos de la planificación, se les puede cambiar los datos, actualizar el estado, añadir una fotografía del regalo o cambiarla. En el caso de las planificaciones colaborativas hay un botón para acceder a editar los datos del grupo aunque sólo lo tiene disponible el planificador. Los participantes sólo pueden visualizar el grupo y, además, tienen bloqueados todos los campos respectivos a la planificación, pueden añadir regalos pero no tienen permitido actualizar el estado del regalo. Desde el modo edición, pulsando en más opciones, se abre un menú para acceder a finalizar o eliminar la planificación. Los participantes no tienen disponible el botón de más opciones.

Identificador	**CU-017**
Nombre	Finalizar una planificación.
Prioridad	Media
Descripción	Finalizar una planificación en curso del sistema, es indiferente si es individual o en grupo.
Actores	Usuario y Sistema
Pre-Condiciones	* Realizado el caso de uso CU-003 – Inicio de sesión. * O el caso de uso CU-010 – Acceder a Planificaciones. * O el caso de uso CU-015 – Visualizar una planificación. * O el caso de uso CU-016 – Editar una planificación. * Conocer al menos el evento, persona y/o fecha de la planificación que se desea finalizar. * La planificación a finalizar debe existir en el sistema y no debe estar finalizada previamente.
Flujo	1.1. Localizar en el listado de planificaciones aquella que se desea finalizar ya sea utilizando una búsqueda, ordenando, filtrando o mirando todo el listado. 1.2. O estar visualizando la planificación en concreto. 1.3. O estar editando la planificación en concreto. 2.1. Pulsar en el botón de más opciones de la planificación y se abre un menú donde está la opción de finalizar. 2.2.- 2.3. También se puede finalizar la planificación cuando se está visualizando o editando. Pulsar en el botón de más opciones y se abre un menú donde está la opción de finalizar. 3. El sistema finaliza la planificación y la marca como finalizada en los datos del backend. 4. El usuario es redirigido a la pantalla de Planificaciones y se muestra un mensaje informando que la planificación se ha finalizado.
Excepciones	3.1. Se produce un error al marcar como finalizada la planificación en el backend por lo que se muestra un mensaje de error en pantalla informando de ello.
Post-Condiciones	La planificación está finalizada y, por lo tanto, ya no se pueden realizar modificaciones en ella y ya no aparece en el listado de planificaciones de la pantalla. Para visualizar hay que acceder al Histórico de Planificaciones.
Notas	En el caso de las planificaciones en grupo esta acción sólo la puede llevar a cabo el planificador.

Identificador	**CU-018**
Nombre	Eliminar una planificación.
Prioridad	Baja
Descripción	Eliminar una planificación del sistema, individual o en grupo, tanto si está en curso como finalizada.
Actores	Usuario y Sistema
Pre-Condiciones	* Realizado el caso de uso CU-003 – Inicio de sesión. * O el caso de uso CU-010 – Acceder a Planificaciones si se trata de una planificación en curso. * O el caso de uso CU-011 – Acceder a Histórico de Planificaciones si se trata de una planificación finalizada.

	* O el caso de uso CU-015 – Visualizar una planificación. * O el caso de uso CU-016 – Editar una planificación. * Conocer al menos el evento, persona y/o fecha de la planificación que se desea eliminar. * La planificación a eliminar debe existir en el sistema.
Flujo	1.1. Localizar en el listado de planificaciones aquella que se desea finalizar ya sea utilizando una búsqueda, ordenando, filtrando o mirando todo el listado. 1.2. O estar visualizando la planificación en concreto. 1.3. O estar editando la planificación en concreto. 2.1. Pulsar en el botón de más opciones de la planificación y se abre un menú donde está la opción de eliminar. 2.2.- 2.3. También se puede eliminar la planificación cuando se está visualizando o editando. Pulsar en el botón de más opciones y se abre un menú donde está la opción de eliminar. 3. El sistema elimina la planificación juntamente con todos los datos asociados (regalos, fotos, participantes…) y en el backend. 4. El usuario es redirigido a la pantalla de Planificaciones o Histórico de Planificaciones y se muestra un mensaje informando que la planificación se ha eliminado.
Excepciones	3.1. Se produce un error al eliminar la planificación en el backend por lo que se muestra un mensaje de error en pantalla informando de ello.
Post-Condiciones	La planificación es eliminada del sistema y ya no aparece en el listado de planificaciones de la pantalla.
Notas	En el caso de las planificaciones en grupo esta acción sólo la puede llevar a cabo el planificador.

Identificador	**CU-019**
Nombre	Añadir un regalo.
Prioridad	Alta
Descripción	Añadir un regalo en una planificación, individual o en grupo, que no esté finalizada.
Actores	Usuario y Sistema
Pre-Condiciones	* Mientras se está realizando el caso de uso CU-014 – Crear una planificación. * O mientras se está realizando el caso de uso CU-016 – Editar una planificación. * Saber al menos el nombre a especificar como regalo.
Flujo	1. Pulsar en la opción de añadir regalo. 2. El sistema abre una nueva pantalla con un formulario en el que rellenar los campos de la ficha de un regalo. 3. Una vez rellenada la ficha del regalo se pulsa en el botón del visto. 4. El usuario es redirigido a la pantalla de la planificación que se está creando o editando.
Excepciones	3.1. Si el usuario no ha informado el nombre del regalo se muestra un mensaje de error informando de ello.
Post-Condiciones	El regalo añadido aparece en la sección de regalos de la planificación que se está creando o editando.
Notas	El usuario puede cancelar la inserción de un regalo pulsando el icono de la X. Cuando se añade un regalo, si se ha informado precio entonces se actualiza el total del precio de los regalos en la planificación y si está informado el presupuesto de la planificación entonces también se actualiza cuanto dinero queda del presupuesto respecto al total del precio de los regalos. Los regalos e almacenan en el backend cuando se crea o edita definitivamente la planificación.

Identificador	CU-025
Nombre	Editar grupo
Prioridad	Alta
Descripción	Acceso a la edición del grupo formado por el planificador y los participantes de una planificación en grupo que no esté finalizada.
Actores	Usuario y Sistema
Pre-Condiciones	* Mientras se está realizando el caso de uso CU-014 – Crear una planificación. * O mientras se está realizando el caso de uso CU-016 – Editar una planificación.
Flujo	1. Pulsar en la opción de editar grupo. 2. El sistema abre una nueva pantalla en la que se muestra un listado con los participantes de la planificación (incluido el planificador) y el dinero que debe cada participante. 3. El usuario realiza las acciones pertinentes.
Excepciones	N/A.
Post-Condiciones	En cuanto el usuario ha acabado puede pulsar el botón de tirar atrás y es redirigido a la pantalla de la planificación que se está creando o editando.
Notas	Sólo el planificador de la planificación colaborativa puede editar el grupo, los participantes sólo pueden visualizar. Se puede volver en cualquier momento a la planificación pulsando el botón de tirar atrás. El importe que debe cada participante se calcula dividiendo el total del precio de los regalos por el número de participantes (planificador incluido). Desde aquí puede añadir nuevos participantes al grupo o eliminarlos y marcar si han pagado o no. Y todos los cambios realizados en el grupo se harán efectivos en el backend cuando se cree o guarden los cambios de la planificación definitivamente.

Identificador	CU-027
Nombre	Añadir participantes.
Prioridad	Alta
Descripción	Añadir participantes en el grupo de una planificación en grupo que no esté finalizada.
Actores	Usuario y Sistema
Pre-Condiciones	* Mientras se está realizando el caso de uso CU-025 – Editar grupo. * Saber el usuario de los contactos que se desea añadir al grupo.
Flujo	1. Pulsar en la opción de añadir participante. 2. El sistema recupera del backend la lista de contactos del usuario y la muestra en pantalla (si tiene). 3. El usuario pulsa el botón Añadir que hay a la derecha del contacto que desea añadir. 4. El sistema muestra un mensaje informando que se ha añadido el contacto al grupo.
Excepciones	N/A.
Post-Condiciones	El contacto desaparece del listado ya que se ha incluido en el grupo. Al volver a la pantalla del grupo el contacto añadido aparece en la sección de participantes de la planificación.
Notas	El planificador puede añadir como participante a cualquiera de sus contactos. Y al añadir un participante se actualiza automáticamente el importe que debe cada participante ya que se vuelve a dividir el precio total de los regalos por el nuevo número total de participantes. En cuanto el usuario ha acabado puede pulsar el botón de tirar atrás y es redirigido a la pantalla de Editar grupo.

Como se ha comentado anteriormente, el resto de casos de uso relacionados con la gestión de usuarios, las planificaciones, la lista de deseos propia del usuario, las notificaciones y los contactos se pueden ver en un documento con todos los casos de uso de la aplicación en los **Anexos**.

2.2.2. Diseño de la arquitectura

En el siguiente apartado se define la arquitectura del sistema, identificando las entidades que se representarán en la base de datos, las clases y objetos que se utilizarán para gestionar los diferentes procesos y la estructura del API que servirá para realizar las peticiones efectuadas al servidor desde un cliente.

2.2.2.1 Diseño de la base de datos

En un principio la aplicación que se va a desarrollar en este proyecto utilizará la base de datos que proporciona Firebase en el producto RealTime Database. Esta es de tipo NoSQL y almacena los datos en formato JSON. De todas formas el primer diagrama de la base de datos para el desarrollo de la aplicación se ha decidido elaborarlo con MySQL para tener una idea más visual y de uso más habitual. En la fase de implementación se transformará este diseño de datos para adaptarlo al tipo de bases de datos de Firebase y al contexto del proyecto.

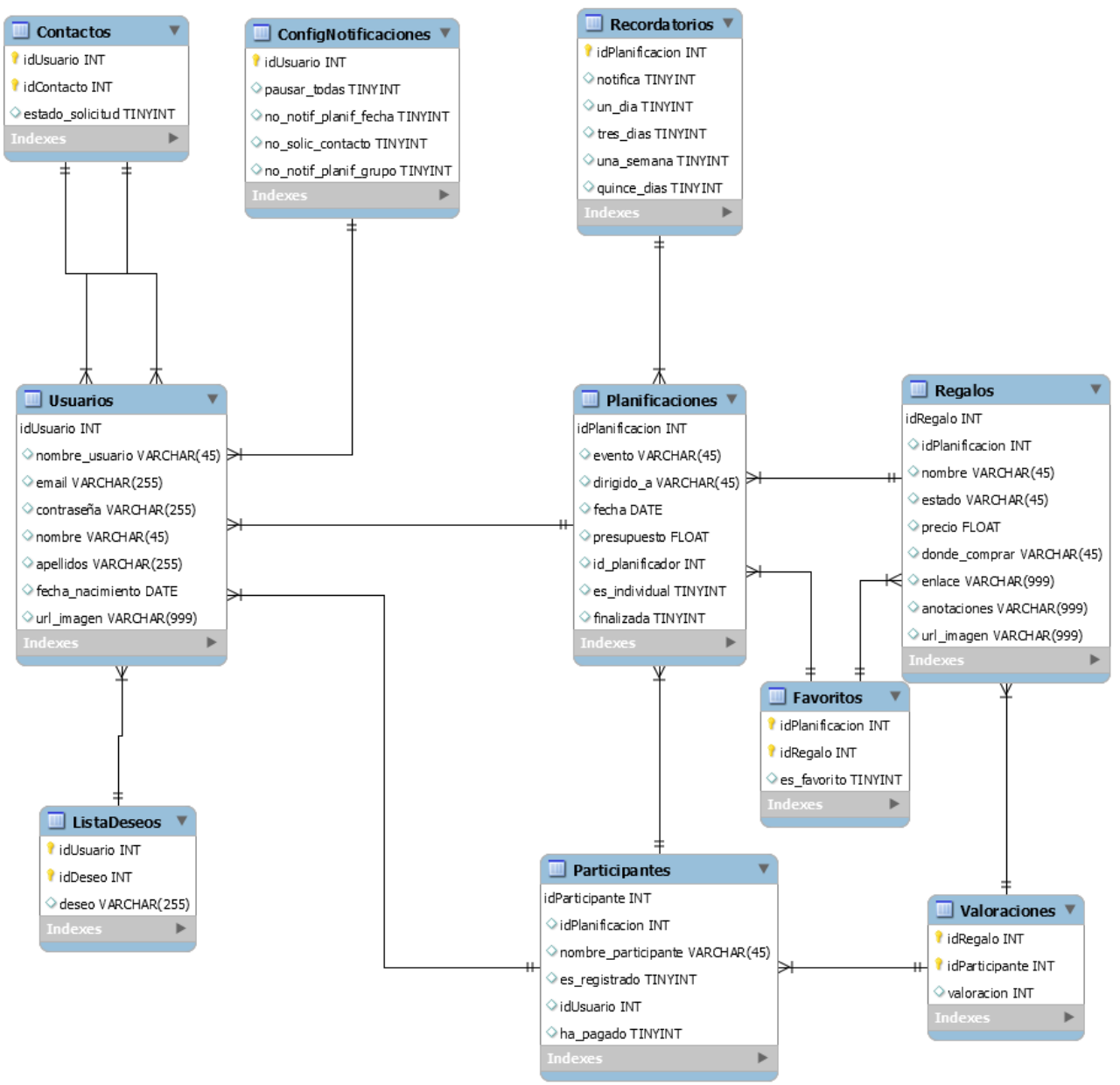


Figura 48: Diagrama de la base de datos

2.2.2.2 Diseño de las entidades y clases

A continuación, se expone el diagrama UML de las entidades, las clases y sus relaciones. Hay que tener en cuenta que es una simple aproximación inicial en la que sólo se han indicado los atributos.

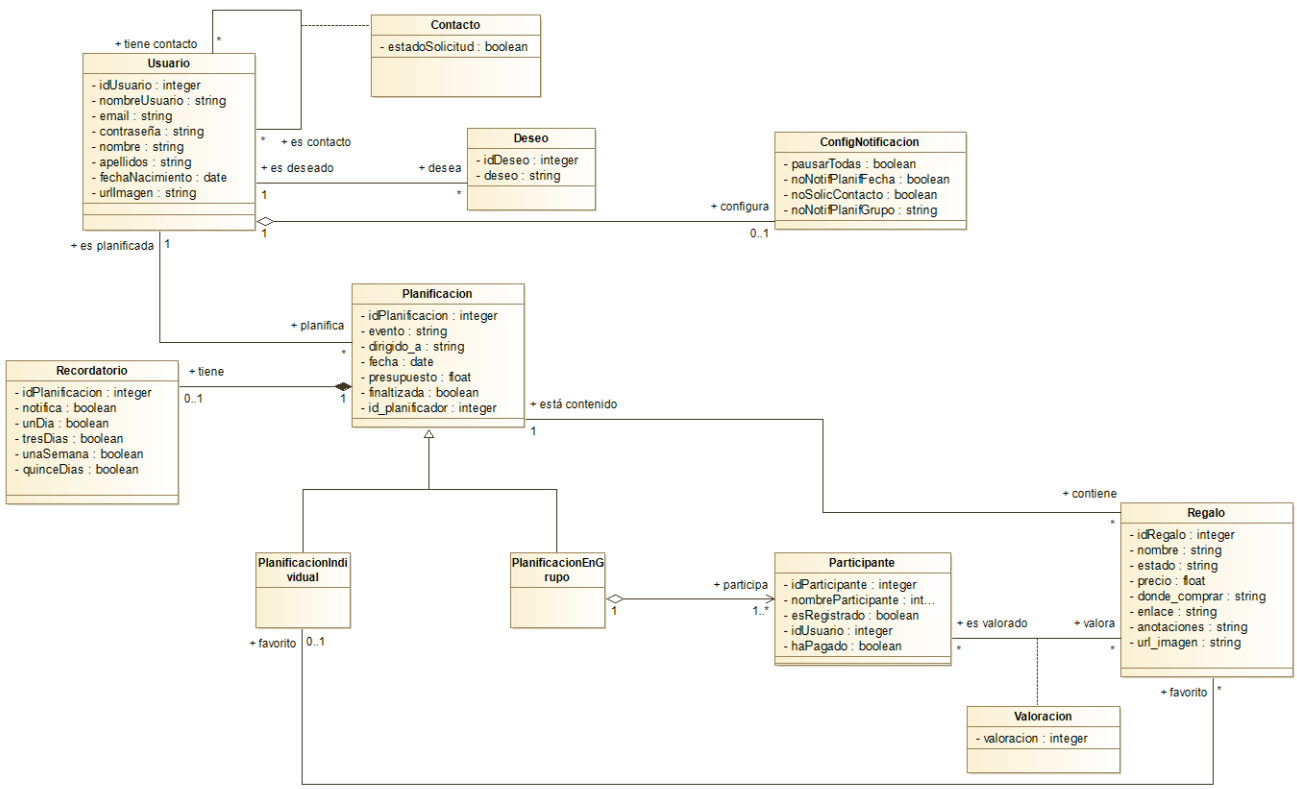


Figura 49: Diagrama de las clases

2.2.2.3 Arquitectura del sistema

La arquitectura que será utilizada por el sistema del proyecto es la de Cliente-Servidor, que es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados Servidores, y los demandantes, llamados Clientes. Estos últimos realizan peticiones y los servidores son quien le da respuesta. Esta tecnología permite distribuir información de forma ágil y eficaz.

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores ayudando así al procesado. Aunque la mayor ventaja es organizativa debido a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y agiliza el diseño del sistema.

Figura 50: Arquitectura del sistema

En el anterior diagrama puede verse que el sistema se divide en dos componentes que se explican brevemente a continuación:

- La **aplicación Android** nativa y que ejecutará peticiones al servidor backend para obtener los datos necesarios para mostrar en las pantallas y para llevar a cabo las diversas funcionalidades a los usuarios. Por tanto, la aplicación actuará como Cliente desde el dispositivo móvil y se encargará de toda la interacción con el usuario a través de la interfaz gráfica y la recopilación de la información necesaria. Se conecta al backend mediante el API y realiza tanto las solicitudes al servidor como la espera y recibimiento de las respuestas por parte del backend.
- El **servidor backend de Firebase** se encargará de responder a las peticiones realizadas por las aplicaciones para la autenticación de los usuarios, para proporcionar al sistema datos almacenados en la base de datos y/o sincronizarlos en tiempo real, para almacenar o recuperar las imágenes tanto de los regalos como de los perfiles de usuario, etc. Por tanto, Firebase ejerce el papel de servidor esperando que lleguen las solicitudes de los clientes, desempeña entonces un papel pasivo en la comunicación. Tras la recepción de una solicitud, la procesa y luego envía la respuesta al cliente. En general, acepta las conexiones de un gran número de clientes.

En cuanto a la arquitectura de la aplicación, se ha decidido utilizar el patrón de diseño de software Modelo Vista Controlador, también conocido como MVC, que separa los datos de la aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos.

- **Modelo:** Es la capa de datos que contiene una representación de los datos que maneja el sistema, administra su lógica de negocio y sus mecanismos de persistencia. Se compondrá de las clases que permiten el acceso a la base de datos remota alojada en Firebase, para realizar inserciones, modificaciones y eliminaciones de datos.
- **Vista:** Es la capa de la interfaz de usuario que compone la información que se envía al cliente y los mecanismos de interacción con éste. La vista obtiene los datos del Modelo, pudiendo haber múltiples vistas del modelo. Para el desarrollo de la vista se utilizará XML.
- **Controlador:** Es la capa lógica que actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno. Para el desarrollo del controlador se utilizará Java.

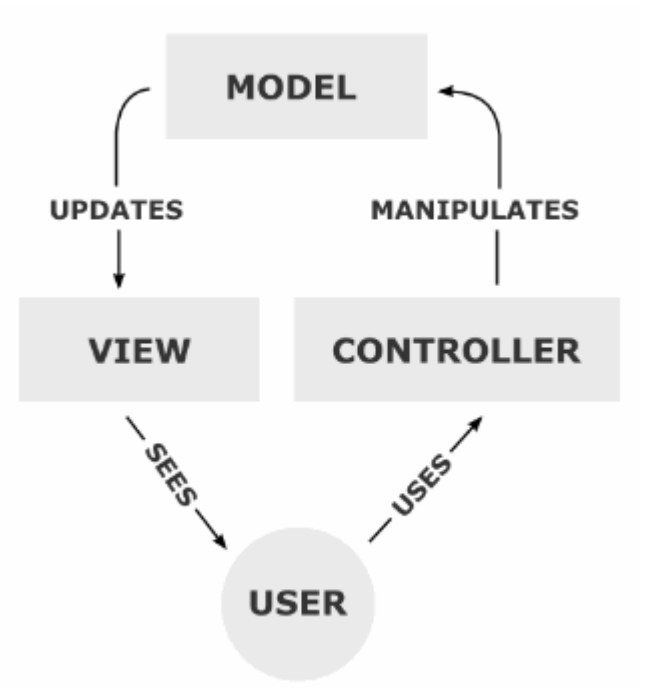


Figura 51: Patrón de arquitectura MVC

3. Implementación

En este capítulo de Implementación se explican los aspectos más importantes relacionados con el desarrollo de la aplicación, llamada Regala, a partir de los diseños realizados en la fase anterior y los objetivos definidos en el Plan de trabajo. Primero se exponen las tecnologías y herramientas utilizadas para la obtención del producto final. En segundo lugar, se comentan los detalles más destacables del desarrollo y el diseño junto a las decisiones tomadas a la hora de implementar las funciones para cumplir los objetivos marcados. A continuación, se exponen las diversas pruebas ejecutadas de la aplicación para detectar posibles errores y garantizar su correcto funcionamiento. Por último, se añade un apartado donde se muestran capturas de la ejecución de la aplicación en las que se observan las principales funcionalidades definidas en los casos de uso del proyecto.

3.1. Herramientas utilizadas

En esta fase para llevar a cabo el desarrollo del sistema se han utilizado diferentes herramientas de software que han sido de especial utilidad para poder cumplir los objetivos. A continuación se profundiza brevemente en algunas de ellas.

3.1.1. Android

Android es un sistema operativo diseñado por Google principalmente para teléfonos móviles o smartphones basado en el sistema operativo Linux. Android está basado en Open Source, esto significa que el código del sistema y el software en general que lo compone está disponible para ser consultado y modificado libremente por cualquiera que lo desee. Así que puede personalizarse ampliamente pudiendo cada fabricante adaptarlo como mejor le parezca. Es fácilmente transportable, hay dispositivos Android de todo tipo así que la oferta es realmente amplia y se puede encontrar lo que se necesita relativamente fácil. También ofrece un gran número de aplicaciones, la gran mayoría se encuentran disponibles en Google Play a través de la aplicación llamada Play Store.

Como se comentó en los requerimientos no funcionales de la primera fase del proyecto, se ha decidido desarrollar la aplicación de forma nativa utilizando la plataforma/sistema operativo Android. La decisión sobre la elección de esta plataforma para el desarrollo de la app se debe esencialmente a su mayor cuota de mercado y en la evolución de uso de dicho sistema operativo a nivel mundial. Asimismo, la implementación de una aplicación nativa ofrece la ventaja de poder acceder a todas las características de maquinaria del dispositivo y proporciona un mejor rendimiento.

La versión mínima de la app es Lollipop (5.1.1) cuyo nivel de API es la 22. Una API se podría definir como un conjunto de herramientas que pone a disposición el desarrollador o fabricante de una plataforma para facilitar la codificación de nuevas aplicaciones y servicios sobre su producto.

Como se puede ver en la Figura 1 esta versión cubre un 80,2% de uso. La elección se debe a que proporciona un equilibrio entre un nivel de la API que se ajusta a las necesidades del desarrollo y que aporta el mayor número de facilidades posible, y una cuota de usuarios amplia que aporta el mayor público objetivo.

ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION
4.0 Ice Cream Sandwich	15	
4.1 Jelly Bean	16	99,6%
4.2 Jelly Bean	17	98,1%
4.3 Jelly Bean	18	95,9%
4.4 KitKat	19	95,3%
5.0 Lollipop	21	85,0%
5.1 Lollipop	22	80,2%
6.0 Marshmallow	23	62,6%
7.0 Nougat	24	37,1%
7.1 Nougat	25	14,2%
8.0 Oreo	26	6,0%
8.1 Oreo	27	1,1%

Figura 52: Niveles de Android

3.1.2. Android Studio

La aplicación se implementa en el entorno de desarrollo Android Studio ya que el mismo sistema Android lo ofrece por defecto como kit de herramientas apropiado e indicado para el desarrollo de aplicaciones Android. Este es el IDE (Integrated Development Environment) oficial de Android basado en IntelliJ IDEA, un entorno o ambiente de desarrollo para programas, que posee potentes herramientas de edición de código. El mismo destaca los errores de forma inmediata, para así dar una solución más rápida de estos. Como herramientas integradas para la construcción de programas en Android, contiene una interfaz de usuario que es diseñada previamente, con varios modelos de pantalla, donde en ella los elementos existentes pueden ser desplazados. Adicionalmente se abarca depuradores para emuladores y la posibilidad de trabajo con Logcat. También posee distintos componentes que ayudan a la tarea de desarrollo y diseño de aplicaciones como un sistema de construcción basado en Gradle, la construcción de variantes y múltiples archivo APK, como también plantillas de código como ayuda. Dispone de un completo editor de diseño con soporte para la edición de arrastrar y soltar el tema elementos. Facilidad de uso y compatibilidad de versiones y consumo de recursos cada vez menor con Gradle. Y un potente simulador de los diferentes dispositivos de la familia Android disponibles en el mercado.

3.1.3. Java y XML

IntelliJ IDEA soporta variados lenguajes que se basan en JVM: Java, Clojure, Groovy, Koltin y Scala. Más un soporte para Maven y Gradle. Para desarrollar de forma nativa para Android se decidió emplear el lenguaje de programación Java que es uno de los más demandados. Una de las ventajas de utilizar Java para desarrollar aplicaciones Android es que Google, con Android Studio, proporciona muchas de las herramientas que serán de ayuda. En un principio se había decidido trabajar con la versión 8 que es la más reciente e incluye nuevas e interesantes características y mejoras. Aunque surgieron problemas al intentar aplicar las expresiones lambda y la API Stream y el propio Android Studio avisó que las versiones mínimas de la aplicación no lo soportan así que finalmente se ha trabajado con la versión 7.

En el proyecto también se ha desarrollado con XML que es un lenguaje de marcas basado en las etiquetas. También se encuentra de forma nativa en Android Studio. Con él se crean y organizan los diferentes elementos que componen las pantallas de la aplicación.

3.1.4. Firebase

Firebase es una plataforma de Google que ofrece diferentes aplicaciones para el desarrollo de aplicaciones web y aplicaciones móviles de una forma sencilla y rápida. Los servicios más destacados que ofrece Firebase son bases de datos, autenticación, almacenamiento, hosting, informe de fallos, analytics, etc.

Estuvo claro desde casi el comienzo del proyecto el uso de Firebase como servidor backend. Este se encarga de responder a las peticiones realizadas por la aplicación para la autenticación de los usuarios y para proporcionar al sistema datos almacenados en la base de datos y/o sincronizarlos en tiempo real. Con Firebase no se hace necesario desarrollar una API paralela para cubrir esas funcionalidades así que usando los servicios que ofrece se ahorra tiempo en la implementación de la aplicación.

Por tanto, de los diferentes productos integrados ofrecidos por Firebase se han utilizado los siguientes en la implementación de la aplicación:

- **Firebase Authentication**

  Es un servicio que simplifica el inicio de sesión y la gestión de la misma. Permite identificar a los usuarios, facilita la creación de un sistema de autenticación seguro y mejora la experiencia de incorporación y acceso para los usuarios finales. Firebase Authentication proporciona servicios de backend, SDK fáciles de usar y bibliotecas de IU (Interfaz de Usuario) ya elaboradas para autenticar a los usuarios en la app. Además, proporciona muchas funcionalidades extras como la recuperación de cuentas, actualización de los datos de autenticación, etc.

- **Cloud Firestore**

  Es un servicio derivado de Google Cloud Platform, adaptado a la plataforma de Firebase. Es una base de datos NoSQL flexible, escalabe y en la nube a fin de almacenar y sincronizar datos para la programación en el lado del

cliente y del servidor. Los datos se almacenan en la base de datos en formato JSON y se sincronizan en tiempo real con cada cliente conectado. Se organiza en forma de documentos agrupados en colecciones y en ellos se pueden incluir tanto campos de diversos tipos (cadenas de texto, números, booleanos, marcas de tiempo, etc.) como otras subcolecciones.

- **Cloud Storage**

  Cloud Storage es un servicio de almacenamiento de objetos potente, simple y rentable. Proporciona cargas y descargas seguras y robustas de archivos para aplicaciones Firebase, sin importar la calidad de la red mediante los SDK de Firebase para Cloud Storage. Se puede utilizar para almacenar imágenes, audio, video o cualquier otro contenido generado por el usuario.

3.1.5. Glide

Glide es un rápido y eficiente framework open source de Android para la carga de imágenes y gestión de recursos multimedia que integra la decodificación de archivos multimedia, manejo de memoria y cache del dispositivo y la agrupación de recursos en una interfaz simple y fácil de usar. Está incorporado en el módulo Storage de FirebaseUI indirectamente para descargar las fotografías almacenadas en la nube en Firebase Cloud Storage.

Glide admite la obtención, decodificación y visualización de imágenes, vídeo y GIF animados. Su enfoque principal es hacer que el desplazamiento de cualquier tipo de lista de imágenes sea lo más suave y rápido posible, pero también es efectivo para casi cualquier caso en el que se necesite buscar, cambiar el tamaño y mostrar una imagen remota.

3.2. Detalles del desarrollo y el diseño

3.2.1. Estructura del proyecto

Como se ha comentado en el anterior apartado, la aplicación ha sido desarrollada de forma nativa bajo el IDE Android Studio siendo compatibles con dispositivos que tengan una versión 5.1 – Lollipop con la API 22 o superior. En la Figura 1 se puede ver que esta versión cubre un 80,2% de uso. La elección se debe a que proporciona un equilibrio entre un nivel de la API que se ajusta a las necesidades del desarrollo y que aporta el mayor número de facilidades posible, y una cuota de usuarios amplia que aporta el mayor público objetivo. Se han descartado las versiones anteriores por temor a los posibles riesgos y problemas que puede conllevar desarrollar para versiones antiguas por tema de incompatibilidades.

La aplicación ha sido desarrollada con el lenguaje de programación Java y, el código, se encuentra en el paquete mblancodef.tfm.app.regala. En una imagen de la siguiente página se puede ver la representación predeterminada de Android Studio para mostrar los archivos de un proyecto de Android. Con esta vista organizada en módulos se puede acceder rápidamente a los archivos fuente clave del proyecto.

El módulo de app contiene las siguientes carpetas:

- **manifests:** Contiene el archivo AndroidManifest.xml que incluye información esencial y necesaria sobre el sistema Android y el proyecto, ya que se declaran las distintas actividades, los requisitos del sistema y los permisos que necesita la aplicación.
- **java:** Contiene los archivos de código fuente Java. El paquete mblancodef.tfm.app.regala alberga todas las clases de las actividades y de los fragment y a su vez tres paquetes más. Un paquete adapters que contiene las clases que extienden de Adapters para la utilización de listados mediante RecyclerViews. El segundo paquete, helper, que tiene dos clases auxiliares, RegalaUtils y PhotoUtils, la primera para diferentes métodos necesarios en la gran mayoría de actividades y fragments, la segunda para tratar las funcionalidades relacionadas con la fotografía de perfil de usuario, los contactos, los participantes en un grupo y la de cada regalo. Y por último, el paquete modelos que cuenta con las diferentes clases que representan el modelo de datos.

- **res:** Contiene todos los recursos sin código, como diseños XML para las pantallas, menú y toolbar, strings, dimensiones, colores, imágenes, iconos, etc.

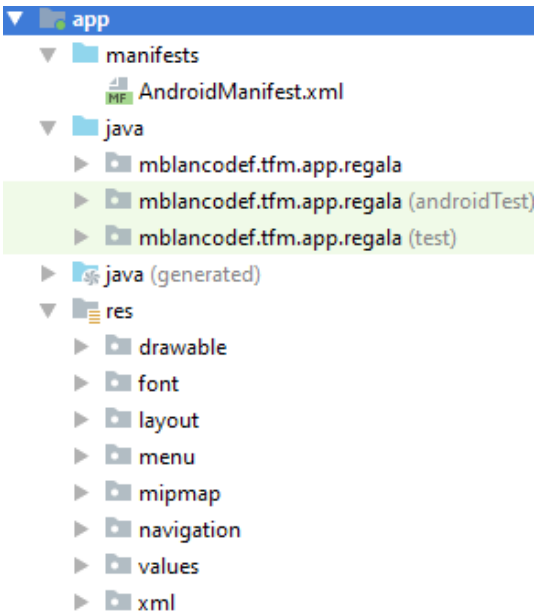


Figura 53: Estructura del proyecto

A continuación se muestra una imagen con el detalle de la carpeta Java del módulo app del proyecto en Android Studio que contiene los archivos de código fuente Java:

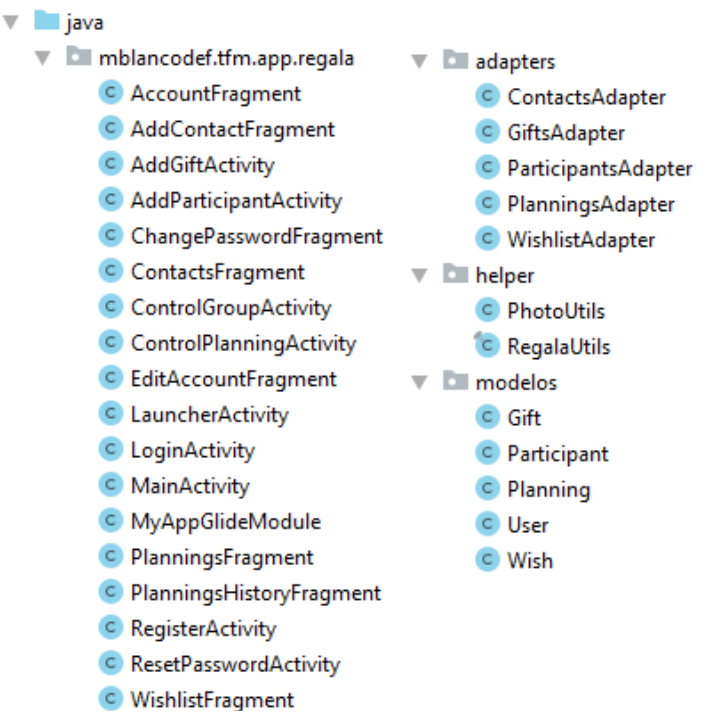


Figura 54: Detalle del paquete mblancodef.tfm.app.regala

3.2.2. Diseño de las pantallas

Los diseños de las pantallas están realizados en los layouts que son los elementos que representan el diseño de la interfaz de usuario de las distintas actividades y fragments que componen el proyecto. Ellos se encargan de actuar como contenedores de views para establecer un orden visual, que facilite la comunicación del usuario con la interfaz de la aplicación. Todos los elementos del diseño se crean usando una jerarquía de objetos View y ViewGroup. Una View suele mostrar un elemento que el usuario puede ver y con el que puede interactuar. En cambio, ViewGroup es un contenedor invisible que define la estructura de diseño de View y otros objetos ViewGroup.

Los diseños se han desarrollado declarando elementos de la IU con un vocabulario XML simple y también utilizando la función Layout Editor de Android Studio para crear el diseño XML mediante una interfaz de arrastrar y soltar y pudiendo cambiar los atributos desde un menú. De esta forma, la presentación gráfica de las diferentes pantallas de la aplicación queda separada del código que controla su comportamiento.

Por otro lado, los dispositivos Android vienen de todas formas y tamaños, por lo que el diseño de la app debe ser flexible. En lugar de definir un diseño con dimensiones rígidas para un tamaño de pantalla y una relación de aspecto determinados, se ha intentado que el diseño se ajuste a las dimensiones de la pantalla con el objetivo de maximizar el número de dispositivos compatibles con la aplicación desarrollada. Según la documentación de Android Developer, en general no se recomienda especificar el ancho ni la altura de un diseño con unidades absolutas como píxeles. En cambio, el uso de medidas relativas como unidades de píxeles independientes de densidad (dp), píxeles independientes de la escala(sp) en el caso de texto, wrap_content o match_parent es un mejor enfoque, ya que ayuda a garantizar que la aplicación se muestre correctamente en dispositivos con pantallas de diferentes tamaños. Así que generalmente se ha usado una de estas constantes para establecer el ancho o la altura:

- **wrap_content:** indica a la vista que modifique su tamaño conforme a los requisitos de este contenido.
- **match_parent:** indica a la vista que se agrande tanto como lo permita su grupo de vistas principal.

La mayor parte de la interfaz gráfica de la aplicación está diseñada con layouts relativos que permiten especificar la ubicación de los objetos secundarios en función de ellos mismos (por ejemplo, el objeto secundario A a la izquierda del objeto secundario B) o en función del elemento primario (por ejemplo, alineado con la parte superior del elemento primario). Principalmente se han empleado ConstraintLayout, CoordinatorLayout, RelativeLayout, ScrollView, NestedScrollView, CardView y FrameLayout, dependiendo de las situaciones específicas.

La mejor manera de crear un diseño receptivo para diferentes tamaños es usar ConstraintLayout ya que permite especificar la posición y el tamaño de cada view en función de las relaciones espaciales con otros views del diseño y con el parent del mismo. De esta manera, todas las vistas pueden moverse y estirarse a la vez, a medida que cambia el tamaño de la pantalla. Es similar a RelativeLayout, pero es más flexible y fácil de usar, así que se ha utilizado en pantallas con diseño más complejo y con más vistas.

Sobre los otros layouts comentar que CoordinatorLayout ha permitido separar fácilmente el toolbar del resto del contenido de la pantalla o en el caso del fragment de planificaciones, mantener el floating action button en un extremo. RelativeLayout se ha usado en layouts sencillos, con pocos elementos, en los que todas las views se establecen según las relaciones entre las views de hermanos y el diseño de los padres. Con ScrollView se ha conseguido que otros layouts, como ConstraintLayout, sean scrollables, es decir, que se pueda deslizar verticalmente la pantalla para seguir viendo el contenido. Con NestedScrollView también se consigue ese efecto de scroll pero se ha utilizado en las pantallas de crear, visualizar o editar una planificación para que haya a la vez scroll en el ConstraintLayout que contiene la información de la planificación y el RecyclerView que contiene el listado de regalos; con ScrollView no funcionaba del todo correcto. CardView se ha usado para diseñar los elementos visuales en forma de tarjetas de información para el listado de planificaciones. Y FrameLayout sólo se utiliza en la actividad de inicio ya que solamente se muestra el logo.

También se ha intentado seguir la recomendación de mantener la jerarquía de diseño lo más sencilla posible anidando el mínimo posible de vistas ya que se procesa más rápido si tiene menos diseños anidados (una jerarquía de vistas ancha es mejor una una de vista profunda).

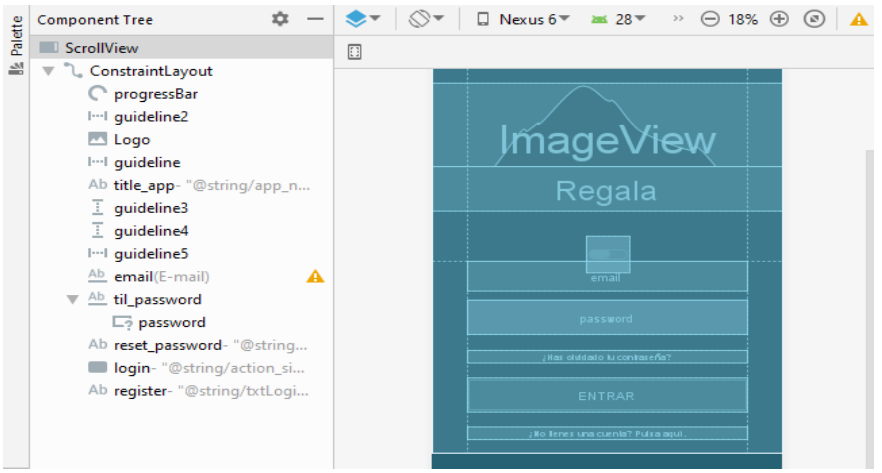


Figura 55: Interfaz gráfica de LoginActivity diseñada con ConstraintLayout

Otro detalle a resaltar en el diseño de las pantallas es la reutilización de diseños mediante la etiqueta <include/> que permite volver a usar diseños completos de manera eficiente. En este proyecto en concreto, el contenido para crear o editar un regalo es un componente que se maneja tanto en la actividad para añadir un regalo tanto cuando se esta creando como editando una planificación y también el mismo diseño se utiliza en el RecyclerView que lista todos los regalos añadidos en la planificación y que es editable. Es un componente grande y con bastantes views así que la posibilidad de reutilizarlos en diferentes layouts ha sido un punto clave.

Y por último, también destacar el uso de ViewStub para cambiar los layouts mediante programación en función de unos criterios específicos. Esto se ha aplicado en el RecyclerView de regalos para utilizar el layout de regalos editable o el de visualización según convenga. También se ha aplicado en el RecyclerView de contactos para visualizar el layout de contactos del usuario o el layout de búsqueda de usuarios para añadir a contactos. Se ha realizado de esta forma para aprovechar el mismo RecyclerView y Adapter en el caso de regalos y por otro lado el de los contactos.

En la actividad para la creación, visualización y edición de planificaciones también se ha utilizado ViewStub para cargar el layout que corresponda según la acción a realizar en las planificaciones y así controlar toda la parte de programación desde la misma actividad ya que tienen bastantes métodos en común.

3.2.3. Cambios en el diseño y funcionalidades

Han habido cambios en el diseño y algunas funcionalidades respecto a los prototipos de alta fidelidad y los primeros casos de uso definidos en el diseño técnico. Estas modificaciones se han realizado en base a comentarios realizados por uno de los consultores tras la entrega de la PAC2 y consideraciones que han surgido durante el desarrollo para facilitar la implementación e intentar mejorar la usabilidad de la aplicación. Estas correcciones se detallan a continuación:

- En el registro de un nuevo usuario se ha añadido la repetición de la contraseña.
- En la pantalla de registro y en la edición del perfil de usuario se ha cambiado el selector de la fecha. También en las planificaciones. Es más práctico seleccionar la fecha en un calendario.
- Tanto para la foto de perfil del usuario, la de los contactos y regalos aparece una imagen por defecto.
- En las planificaciones en curso y en el histórico se han quitado los botones de editar y eliminar y se muestran cuando se pulsa en el botón de tres puntos para más opciones. Además, se han quitado los regalos de la ficha de cada planificación para que la pantalla no se vea tan sobrecargada de información.
- En el detalle de una planificación al final no hay botón para mostrar/ocultar parte de la información del regalo. Y se ha agregado un botón para añadir un regalo que abre una nueva pantalla donde cumplimentar los datos del nuevo regalo.

- En el caso de las planificaciones en grupo, el listado de participantes se ha extraído a una nueva pantalla a la cual se accede desde un botón de la planificación. Así el detalle de la planificación no queda tan sobrecargado de datos.
- En la cuenta de usuario se ha quitado la opción de recomendar app y se ha añadido en el menú lateral.
- El cambio de contraseña se ha extraído fuera de la edición del perfil de usuario y se realiza desde una opción a parte en la cuenta de usuario pulsando en un botón que abre una nueva pantalla para ello.
- Finalmente los contactos se eliminan de uno en uno ya que así se muestra un cuadro de diálogo al usuario para que confirme si realmente quiere quitarlo de contactos o cancelar la acción.

3.2.4. Menú lateral con DrawerLayout y Navigation Component

Cuando el usuario se registra en la aplicación o se autentifica en la pantalla de Login es redirigido a la pantalla principal en la que se muestran las planificaciones en curso. En esta consta el panel lateral de navegación que es un panel de IU que muestra el menú de navegación principal de la app. El panel lateral aparece cuando el usuario toca el icono del menú lateral en la barra superior de la app o cuando el usuario desliza un dedo desde el borde izquierdo de la pantalla.

Para desarrollar el panel lateral se ha utilizado un DrawerLayout junto el componente Navigation de Android Jetpack que permite implementar la navegación de las diferentes opciones del menú y de ahí a otros fragments. Entonces el diseño de la interfaz está compuesto por varios layouts combinados de la siguiente forma:

- **activity_main.xml:** Es el layout de la actividad MainActivity que controla y gestiona enteramente el Navigation Drawer. Dentro del DrawerLayout se agrega un diseño para el contenido principal de la IU (app_bar_main) y un NavigationView que tiene el contenido del panel lateral de navegación, por un lado el layout nav_header_main como cabecera y por otro el menú activity_main_drawer.
- **nav_header_main.xml:** Contiene el diseño de la cabecera del menú formado por el título de la app, la foto de perfil del usuario, su nombre de usuario y el email. Este header no es más que un layout que el componente NavigationView inserta en la parte superior del menú.
- **activity_main_drawer.xml:** Las diferentes opciones que forman el menú.
- **app_bar_main.xml:** Se compone de un CoordinatorLayout que contiene el AppBarLayout, es decir, la barra superior (toolbar) de la pantalla e incluye el layout content_main para cargar el contenido de cada pantalla del menú.
- **content_main.xml:** Contenido de la actividad principal MainActivity que está configurado para trabajar con fragments. Está formado por un NavHostFragment que muestra el Fragment correspondiente a la opción seleccionada en el menú.

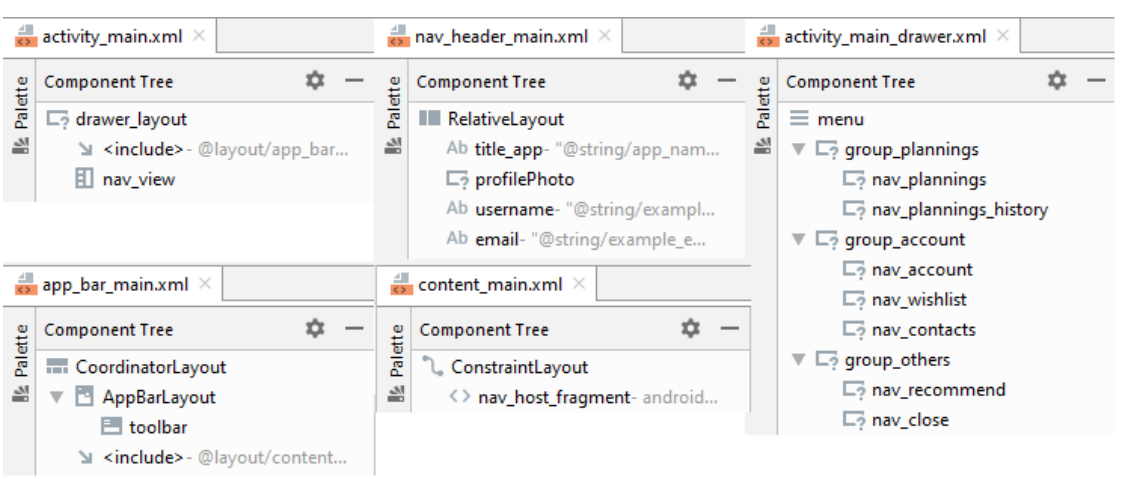


Figura 56: Diseño UI del Navigation Drawer para el menú

Por otro lado, el componente Navigation consta de tres partes claves que se describen a continuación:

- **Gráfico de navegación** (mobile_navigation.xml): Es un recurso XML que contiene toda la información relacionada con la navegación con las áreas de contenido individuales dentro de la app, llamadas destinos, así como las posibles rutas que un usuario puede tomar a través de la app.
- **NavHost:** Es un contenedor vacío que muestra los destinos del gráfico de navegación. Se ha utilizado una implementación NavHost predeterminada, NavHostFragment, que muestra los destinos de los fragmentos.
- **NavController:** Es un objeto que administra la navegación de la app dentro de un NavHost. NavController coordina el intercambio de contenido de destino en el objeto NavHost a medida que los usuarios se mueven a través de la app. NavController es el responsable de reemplazar el contenido de NavHost con el nuevo destino.

Mientras se navega por la app, cada vez que se pulsa en una opción del menú se está comunicando a NavController que se quiere navegar por una ruta específica del gráfico de navegación. Luego NavController muestra el destino apropiado en NavHost.

En la siguiente imagen se muestra una representación visual del gráfico de navegación de la aplicación que contiene varios destinos sin conectar ya que cada uno representa una opción del menú, es decir, un Fragment determinado. Hay tres destinos conectados, dos desde la cuenta de usuario y uno desde contactos. Cada destino se representa con una miniatura de vista previa, mientras que las acciones de conexión se representan a través de fechas que muestran cómo los usuarios pueden navegar de un destino al otro. También se han utilizado los Argument Default Values en la pestaña de atributos de los destinos para pasar datos del usuario y la lista de contactos.

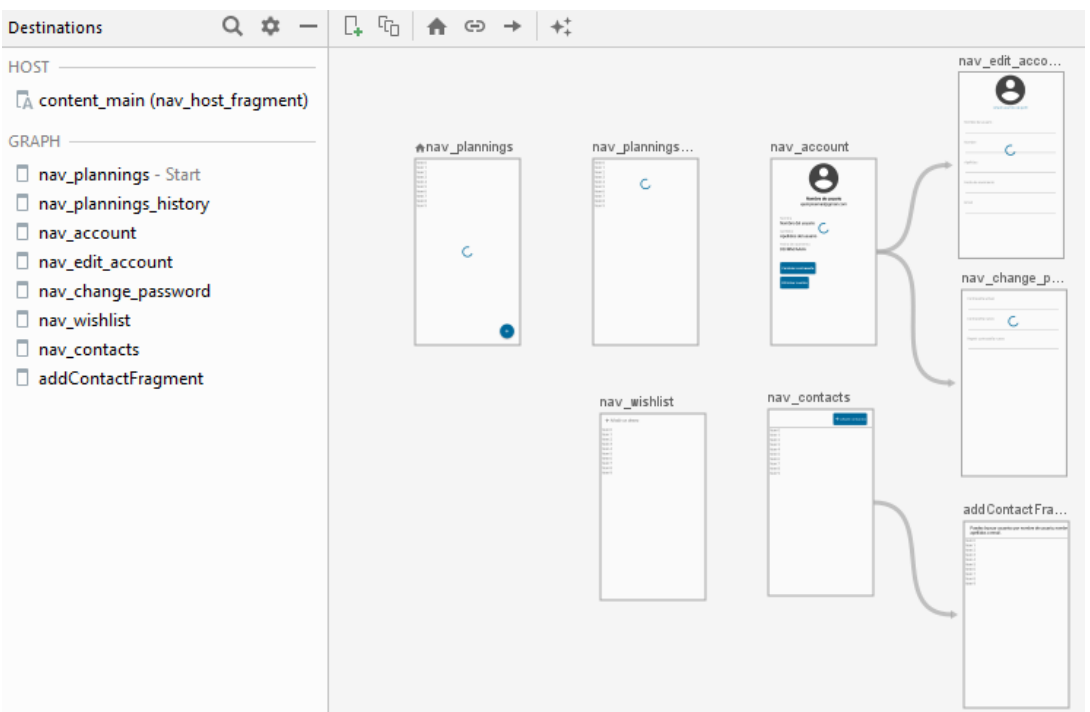


Figura 57: Gráfico de navegación del menú de la app

3.2.5. Listas con RecyclerView y Adapter

En la aplicación se muestran diferentes listados, unos con diseños simples y otros más complejos, para mostrar las planificaciones en curso, las finalizadas, los regalos de una planificación, los participantes en una planificación en grupo, los contactos, usuarios del sistema para agregar a contactos y la lista de deseos. Todos los listados se representan en el layout mediante el uso de la vista RecyclerView que visualiza una lista que permite mostrar información en una gran colección de vistas de una forma más eficiente y potente. Cada elemento de la lista es un layout.

Para personalizar los elementos a mostrar en el RecyclerView se ha usado un Adapter (adaptador). Un adaptador es un mecanismo estándar en Android que permite crear una serie de vistas que han de ser mostradas dentro de un contenedor. Con RecyclerView se utiliza la clase RecyclerView.Adapter. En el constructor se inicializa el conjunto de datos a mostrar y otras variables globales a la clase, entre ellas un listener OnItemClickListener para detectar cuando se pulsa en un botón, un icono de eliminar, un texto clickable, etc.

Después, en cada Adapter, se ha creado la clase ViewHolder que contiene las vistas que se quieren modificar de un elemento como textos, botones, imágenes, etc. Con esta clase se evita tener que crear las vistas de cada elemento desde cero. El método onCreateViewHolder() devuelve una vista de un elemento sin personalizar. Se utiliza el método inflate() para crear una vista a partir del layout XML. Y el método onBindViewHolder() personaliza un elemento del tipo ViewHolder según la posición. A partir del ViewHolder previamente personalizado ya es el sistema quien se encarga de crear la vista definitiva que será la insertada en el RecyclerView. Para actualizar los datos en el adaptador se llama el método notifyDataSetChanged() que notifica a los observadores adjuntos que los datos subyacentes han cambiado y que en el RecyclerView se deben reflejar el conjunto de datos actualizado. También se han utilizado otros métodos como notifyItemRangeRemoved o notifyItemRangeInserted para notificar sólo el cambio en un conjunto de posiciones.

3.2.6. Paso de parámetros entre Activities y Fragments

En Android si necesitamos pasar datos de tipos básicos o primitivos entre nuestras Activities basta con establecerlos en un objeto Intent con el método putExtra(). En el caso de pasarlos entre Fragments también se puede crear un objeto Bundle indicándole los argumentos con el método setArguments(). Aunque en algunas situaciones de este proyecto se hace necesario ampliar esta funcionalidad para poder enviar un objeto y/o lista de objetos de las clases User, Planning y Gift de una actividad a otra o entre fragmentos.

Por tanto, para conseguir esta funcionalidad hay que implementar en esas clases la interface Parcelable. Para ello hay que implementar una serie de métodos como writeToParcel() que permite escribir las instancias de dicha clase desde un Parcel. Un constructor que recibe un argumento de la clase Parcel y que se llamará desde la actividad receptora y describeContents() que devuelve el código hash del objeto. Y también Parcelable.Creator con los métodos createFromParcel() para reconstruir el objeto propio en base a un objeto de la clase Parcel y newArray para "desempaquetar" el Parcel.

Una vez implementada la interface Parcelable ya se pueden enviar objetos de User, Planning y Gift entre actividades, fragmentos o ambos mediante los métodos putExtra() o putParcelableArrayListExtra() de la clase Intent y recibirlos mediante getParcelableExtra() o getParcelableArrayListExtra() de la misma clase.

3.2.7. Gestión de los usuarios con Firebase Authentication

Con el uso de Firebase Authentication en la aplicación esta proporciona las siguientes funcionalidades: registro de nuevos usuarios, autenticación con la cuenta creada, acceso automático a la app, restablecer la contraseña, modificar los datos de acceso, cierre de sesión y eliminación de la cuenta. Se ha decantando por el método clásico de inicio de sesión mediante correo electrónico y contraseña cuyos datos se almacenan en Firebase cuando se registra el usuario.

En la primera actividad de la app, LauncherActivity, se comprueba si el usuario se encuentra logueado mediante la inicialización de un listener, FirebaseAuth.AuthStateListener, y añadiéndolo con addAuthStateListener() para evitar que los usuarios tengan que autenticarse cada vez que usen la app. El observador onAuthStateChanged detecta si ha habido cambios en el estado de autenticación.

En LoginActivity se validan las credenciales introducidas por el usuario mediante el método signInWithEmailAndPassword() de la case FirebaseAuth y se define un listener para comprobar si el login tuvo éxito o no en función de la Task<AuthResult> devuelta al llamar el método isSuccessful() una vez completada la escucha.

Y la creación de un nuevo usuario se realiza en la actividad RegisterActivity mediante el método createUserWithEmailAndPassword() de la clase FirebaseAuth. A continuación, se actualiza el perfil del usuario usando el método updateProfile() de FirebaseUser pasándole un UserProfileChangeRequest con el nombre de usuario.
En todos los cambios se define un listener para comprobar el resultado una vez completada la escucha y en caso de error tratarlos y mostrar un mensaje al usuario.

Para el restablecimiento de la contraseña se llama a la actividad ResetPasswordActivity donde mediante el método sendPasswordResetEmail() de la clase FirebaseAuth() se solicita el envío de un correo electrónico de restablecimiento de la contraseña. Si el envío del email ha sido correcto entonces se avisa al usuario y se le redirige a la pantalla de Login.

Una vez se tiene acceso a la app, si el usuario desea cerrar su sesión debe pulsar la opción correspondiente en el menú lateral que está gestionado por MainActivity que llama al método signOut() de la clase FirebaseAuth. Para comprobar si se ha cerrado la sesión se instancia FirebaseAuth con getInstance() y se intentan obtener los datos de su cuenta con el método getCurrentUser(). En caso de obtener un resultado nulo entonces se redirige a la pantalla de Login.

La modificación de los datos del perfil del usuario, entre ellos el correo electrónico, se realiza en el fragment EditAccountFragment llamando al método updateEmail() de la clase FirebaseUser. Si se ha cambiado el nombre de usuario entonces se actualiza usando el método updateProfile() de FirebaseUser pasándole un UserProfileChangeRequest con el nuevo nombre de usuario. El usuario es redirigido al fragment AccountFragment. La contraseña se cambia desde en el fragment ChangePasswordFragment, llamando al método updatePassword() de la clase FirebaseUser. Se utilizan las SharedPreferences para tener disponible en todo momento la contraseña actual del usuario y así poder realizar la validación de la contraseña actual informada por el usuario.

Y por último, la eliminación de la cuenta se realiza en el propio fragment AccountFragment donde se lanza un cuadro de diálogo donde se informa al usuario que perderá todos los datos de la aplicación y confirme que realmente quiere eliminar la cuenta. Para la eliminación se ejecuta el método delete() de la clase FirebaseUser y paralelamente se borran todos los datos creados por el usuario en Firestore.

Muy importante también comentar que por motivos de seguridad, algunas acciones, como borrar una cuenta, cambiar una dirección de correo electrónico y cambiar una contraseña, no pueden realizarse si el usuario no accedió a su cuenta recientemente. Si transcurrió demasiado tiempo desde que el usuario accedió a su cuenta por última vez, cualquier intento de realizar estas acciones fallará. Para evitarlo se vuelve a autenticar al usuario llamando al método reauthenticate() al cual se le pasa un objeto AuthCredential con las credenciales de acceso nuevas del usuario que se han obtenido con EmailAuthProvider.getCredential() pasándole el email y la contraseña guardados en las SharedPreferences.

3.2.8. Estructura de los datos en Cloud Firestore

En un principio se iba a utilizar Realtime Database Firebase para la base de datos pero finalmente se optó por Cloud Firestore principalmente por las carencias de la primera en cuanto a las consultas compuestas, mencionado en su apartado de desventajas en la documentación de Firebase. Otros motivos fueron que la estructura visible en la consola es más sencilla y fácil de escalar con el manejo de subcolecciones. Se puede combinar filtros con ordenamiento y al consultar solo trae el documento o los documentos sin traer subcolecciones, esto se pide a demanda. Además, soporta operaciones batch las cuales se pueden realizar operaciones de insertar, actualizar o eliminar al mismo tiempo.

La base de datos que proporciona Firestore es de tipo NoSQL y almacena los datos en formato JSON así que se tuvo que transformar el diseño de datos planteado en el apartado de arquitectura para adaptarlo a este tipo de bases de datos. Se modeló en función de cómo la aplicación va a leer los datos y también teniendo en cuenta la forma de escribirlos para que el proceso sea lo más simple posible consiguiendo un mejor rendimiento y optimización.

Finalmente, la estructura definida se divide en dos grandes colecciones, Users y Plannings y estas a su vez por documentos y subcolecciones como se detalla a continuación:

- **Colección Users:**

  Alberga un documento por cada uno de los usuarios registrados en la aplicación. Cada uno de estos documentos se identifica mediante un ID unívoco. El documento contiene toda la información del usuario como el nombre de usuario, el nombre, los apellidos, la fecha de nacimiento y el email. Como los datos del perfil de usuario son accesibles desde otros usuarios añadidos a Contactos se decidió no guardar la contraseña. Además, puede tener (si existe) dos subcolecciones, Whislist para la lista de deseos, y Contacts para los contactos del usuario. De esta forma se puede obtener dichas subcolecciones para un determinado usuario.

- **Subcolección Whislist:**

  Contiene un documento para cada uno de los deseos insertado por el usuario en su lista de deseos. Cada uno de estos documentos se identifica mediante un ID unívoco y contiene el texto del deseo y la fecha de creación.

- **Subcolección Contacts:**

  Alberga un documento para cada uno de los usuarios agregados a contactos por un usuario de la aplicación. Cada uno de estos documentos se identifica mediante el ID unívoco del usuario agregado para así después poder buscar su información mucho más rápido al disponer de su ID.

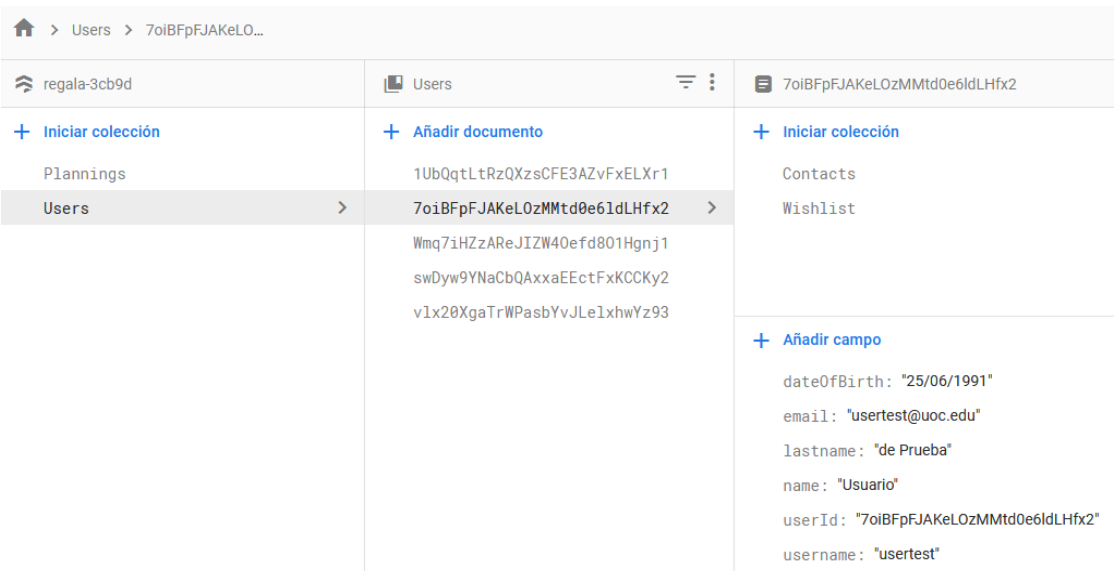


Figura 58: Detalle del documento de un usuario en Cloud Firestore

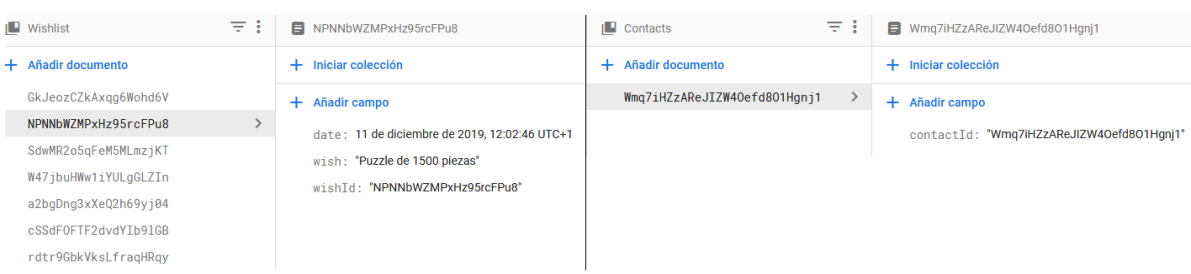


Figura 59: Detalle del documento de un deseo y de un contacto de un usuario en Cloud Firestore

- **Colección Plannings:**

  Contiene un documento para cada una de las planificaciones creadas en la aplicación. Cada uno de estos documentos se identifica mediante un ID unívoco para diferenciar las planificaciones. El documento contiene el ID unívoco del usuario que ha creado la planificación (el planificador) y toda la información de la planificación como el evento, el nombre de a quien va dirigido, la fecha del evento, si es individual o no (en grupo), el presupuesto, el precio total de los regalos y si la planificación está finalizada o no. Además, puede tener (si existe) dos subcolecciones, Gifts para los regalos de la planificación y Participants para en el caso de una planificación en grupo poder almacenar todos los participantes de la misma.

  - **Subcolección Gifts:**

    Alberga un documento para cada uno de los regalos añadidos en la planificación. Cada uno de estos documentos se identifica mediante un ID unívoco y contiene el ID de la planificación para luego poder

relacionarlos mucho más rápido. También el nombre del regalo, el estado, el precio, si es favorito o no (sólo en las individuales), donde comprarlo, el enlace web y anotaciones.

- **Subcolección Participants:**

  Contiene un documento para cada uno de los participantes que forman parte de una planificación en grupo. Cada uno de estos documentos se identifica mediante el ID unívoco del usuario agregado al grupo para así después poder buscar su información mucho más rápido al disponer de su ID. También se almacena el ID unívoco de la planificación, si el participante es el planificador o no y si ha pagado su parte de los regalos.

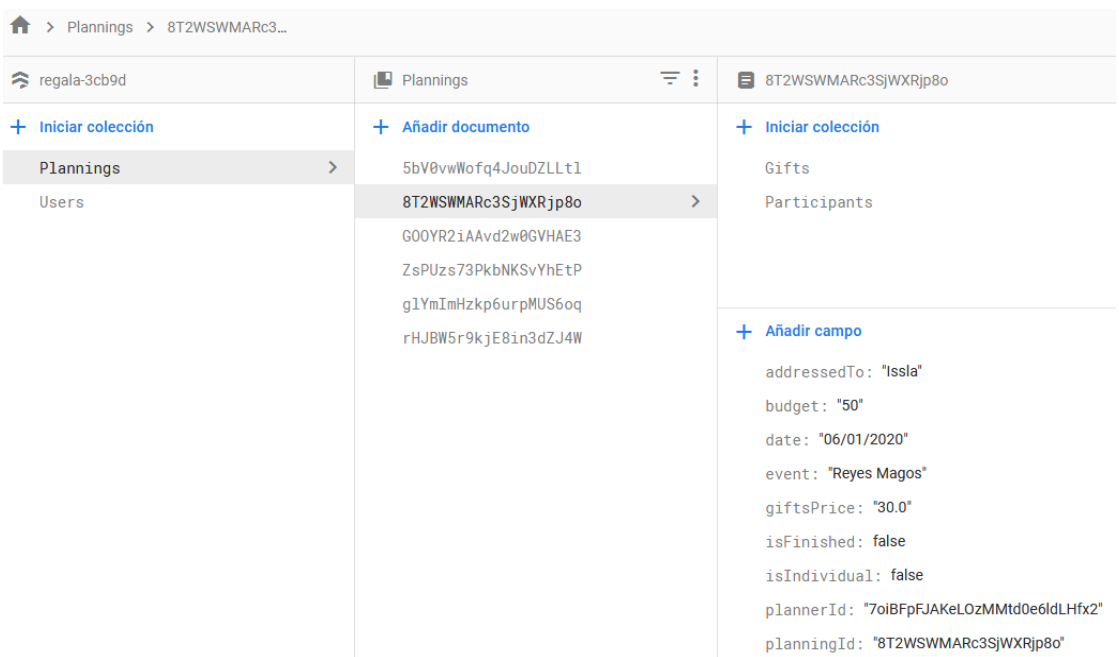


Figura 60: Detalle del documento de una planificación colaborativa en Cloud Firestore

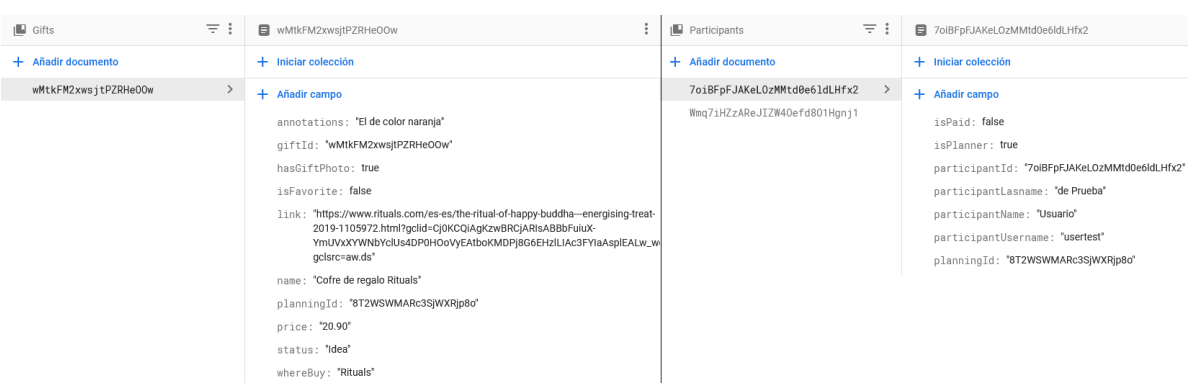


Figura 61: Detalle del documento de un regalo y de un participante de una planificación en grupo en Cloud Firestore

3.2.9. Almacenaje y recuperación de datos con Cloud Firestore

En primer lugar, para poder trabajar con la base de datos de Firestore es necesario escribir las reglas de seguridad que proporcionan control de acceso y validación de datos. Los usuarios los puede recuperar cualquiera para así poder realizar la validación del nombre de usuario cuando aún no se está registrado. Y sólo el propio usuario puede crear, actualizar y borrar sus datos. En cambio, cualquier acción contra los contactos y la lista de deseos sólo la puede hacer el usuario al que pertenecen esos datos.

```
rules_version = '2';
service cloud.firestore {
  match /databases/{database}/documents {
    // Rules for Users
    match /Users/{userId} {
      allow read: if true
      allow create, update, delete: if request.auth.uid == userId;
    }
```

Figura 62: Reglas de acceso para los usuarios

Las planificaciones sólo las puede crear y borrar el usuario que está identificado como planificador. Y las puede recuperar y actualizar cualquier usuario (en teoría sólo el planificador y los participantes pero no se ha logrado definir).

```
// Rules for Plannings
match /Plannings/{planningId} {
  allow create: if request.resource.data.plannerId == request.auth.uid;
  allow delete: if resource.data.plannerId == request.auth.uid;
  allow read, update: if request.auth.uid != null;
}
```

Figura 63: Reglas de acceso para las planificaciones

También se ha creado una función para identificar si el usuario que realiza algunas acciones en concreto contra la base de datos es el planificador de la planificación o no para cuando se elimina un regalo de una planificación y cuando se actualiza o elimina un participante de una planificación en grupo. El resto de acciones para los regalos y los participantes simplemente se comprueba que el usuario que lo realiza está al menos registrado.

```
// True if the user is the planning planner - Existing Data
function isPlanningPlanner() {
  return get(/databases/$(database)/documents/Plannings/$(resource.data.planningId))
  .data.plannerId == request.auth.uid;
}

// Rules for Gifts of Planning
match /Plannings/{planningId}/Gifts/{giftId} {
  allow delete: if isPlanningPlanner();
  allow create, read, update: if request.auth.uid != null;
}
```

Figura 64: Ejemplo de uso de una función en las reglas de acceso para los regalos

Y para poder consultar la subcolección Participants para cualquier documento de planificación se ha utilizado una sintaxis de comodines recurrentes, {path=**}, y a continuación la subcolección Participans. De esta forma se indica que la regla aplica a dicha subcolección para cualquier colección y documento padre.

Por otra parte, también se han creado índices compuestos para poder recuperar los usuarios filtrando por el ID de usuario y ordenando el nombre de usuario de forma ascendente cuando se recupera la información de los contactos de un usuario en concreto. Para filtrar si una planificación está finalizada o no y a la vez ordenar por fecha descendente. Y para poder recuperar los datos de participantes filtrando si es planificador o no y el ID de participante (el ID de usuario).

Hecho esto, lo siguiente que se necesita para poder trabajar con la base de datos es obtener una instancia de FirebaseFirestore mediante FirebaseFirestore.getInstance().

Firestore permite escribir una variedad de tipos de datos dentro de un documento, incluidos objetos y matrices anidadas, strings, booleanos, números, fechas y nulos. En la app desarrollada se ha usado principalmente objetos Java con clases personalizadas para la escritura, el propio Firestore convierte internamente los objetos en tipos de datos compatibles. Y para crear o reemplazar un solo documento, se ha usado el método set() de la clase FirebaseFirestore. Si el documento no existe, se creará. Si el documento no existe, su contenido se reemplaza por los datos proporcionados o en algunos casos por los datos especificados que se combinan en el documento existente. En algunos casos de creación se ha especificado el ID para el documento que se va a crear pero a veces no hay un ID significativo para el documento y se deja que Firestore lo genere automáticamente.

Para actualizar algunos campos de un documento sin reemplazarlo por completo se ha utilizado el método update(). Además, para indicar la colección se utiliza el método collection() indicando el nombre de la colección y si se especifica el ID del documento entonces se utiliza el método document().
Para borrar un documento simplemente se emplea el método delete() aunque Firestore no borra automáticamente los documentos dentro de sus subcolecciones, se ha tenido que hacer de forma manual. Para borrar por completo una colección o subcolección en Firestore se han tenido que recuperar todos los documentos de la colección o subcolección (si no se habían recuperado ya previamente) y se borran de uno en uno.

En cuanto a las operaciones de lectura, para recuperar datos en una determinada colección, subcolección o un documento específico se utiliza el método get() de la clase FirebaseFirestore. Como en la escritura, para indicar la colección se utiliza el método collection() indicando el nombre de la colección y si se especifica el ID del documento entonces se utiliza el método document(). Si sólo se indica la colección entonces se recuperan todos los documentos contenidos en ella. Además, se define el listener OnCompleteListener con el método addOnCompleteListener() que cuando ha completado la lectura entonces llama al método onComplete de OnCompleteListener. En dicho método se comprueba si Task<QuerySnapshot>, en el caso de las colecciones, o Task<DocumentSnapshot>, en el caso de los documentos, ha tenido éxito con el método isSuccesfull(). El resultado obtenido se convierte a un objeto personalizado de Java, es decir, las clases creadas para el modelo de datos.
También añadir que cuando se recuperan datos de Firestore se puede filtrar por un campo específico mediante el método whereEqualTo(), filtrar para varios valores en un campo con whereIn() y ordenar con orderBy().

Por último, también se ha aprovechado una de las funcionalidades que proporciona Firestore, la escritura en lotes. En casos que no se necesita leer documentos en el conjunto de operaciones, se han ejecutado varias operaciones de escritura como un lote único que incluye cualquier combinación de operaciones set(), update() o delete(). El lote de escrituras se completa de forma atómica y puede escribir en varios documentos. Se ha empleado por ejemplo para almacenar varios regalos y participantes (en caso de grupo) a la vez cuando se crea o edita una planificación y para borrar los datos de los regalos, los participantes (en caso de grupo) y la planificación a la vez. Para ello se instancia la clase WriteBatch y se van agregando las operaciones al lote de escritura, por ejemplo set() para actualizar o delete() para borrar pasando el documento en un objeto DocumentReference. Por último, para confirmar todas las escrituras del lote como una sola unidad atómica se ejecuta el método commit() de WriteBatch.

3.2.10. Gestión de permisos

La aplicación necesita permisos para tener acceso a Internet, a la cámara (CAMERA) y de lectura (READ_EXTERNAL_STORAGE) y escritura (WRITE_EXTERNAL_STORAGE) en los espacios de almacenamiento compartido para la obtención y almacenamiento de imágenes. Estos permisos se gestionan en los ajustes del dispositivo, en el apartado de aplicaciones. En caso que el usuario no haya dado permiso para el acceso de alguno de estos elementos, se presenta un diálogo de requerimiento de permisos en la pantalla correspondiente. Si selecciona la opción de no habilitar el permiso se muestra un mensaje de aviso informando que se ha denegado el permiso y no deja realizar la acción pertinente.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="mblancodef.tfm.app.regala">

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.CAMERA" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
```

Figura 65: Permisos de la aplicación Regala

3.2.11. Obtención de imágenes

En la aplicación Regala se da la opción de añadir o cambiar una fotografía de perfil cuando se registra un nuevo usuario o en la edición del perfil de usuario. También se puede añadir una fotografía de un regalo cuando se está añadiendo uno en una planificación o editando.

En todos los casos se permite añadir la fotografía tomándola directamente con la cámara o seleccionándola de la galería de imágenes del dispositivo móvil. El usuario puede escoger entre ambas alternativas en el AlertDialog que se muestra y donde se le pregunta desde dónde desea obtener la imagen. A continuación se detalla el procedimiento de cada elección:

- **Haz una foto**
  Primero se comprueba si la aplicación tiene concedidos los permisos de CAMERA y WRITE_EXTERNAL_STORAGE para acceder al dispositivo de la cámara y para escribir en el almacenamiento externo respectivamente. Se usa el método ContextCompart.checkSelfPermission() para la comprobación y en caso de no estar concedidos se solicitan con requestPermissions(). Una vez concedidos, se crea con el método createTempFile() un fichero con una ruta pública donde se almacenará la imagen. Con un Intent se solicita la cámara del dispositivo mediante MediaStore.*EXTRA_OUTPUT* para realizar la fotografía pasando el fichero creado en forma de Uri de contenido generado con el componente FileProvider. La imagen capturada en forma de Uri se obtiene en onActivityResult() y es tratada para adaptarla al tamaño de la vista donde se va a mostrar. Finalmente,la imagen se guarda en la galería mediante un Intent con ACTION_MEDIA_SCANNER_SCAN_FILE, al que se le pasa el Uri obtenido a partir del fichero donde está almacenada la fotografía.

- **Seleccionar de la Galería**
  El primer paso consiste en comprobar si la aplicación tiene concedidos los permisos de READ_EXTERNAL_STORAGE para que la aplicación pueda acceder al almacenamiento externo del dispositivo. Una vez concedidos, se crea un Intent que permite al usuario seleccionar la fuente desde la que quiere obtener la imagen. La imagen capturada en forma de Uri se obtiene en onActivityResult() y es tratada para adaptarla al tamaño de la vista donde se va a mostrar.

El tratamiento de la imagen capturada en forma de Uri de contenido es el mismo. Primero se reduce la imagen al tamaño de la vista donde se va a mostrar. Para esto, se transforma el Uri en un inputStream, calcula el factor de escala en función del tamaño real de la imagen y la escala convirtiéndola en un bitmap mediante el método decodeStream() de BitmapFactory. También se verifica si la orientación es la correcta y si es necesario rotar la imagen los grados necesarios para corregirla. Y finalmente, el bitmap obtenido se muestra en el contenido de la vista correspondiente.

3.2.12. Almacenamiento de imágenes en Cloud Storage

Una vez obtenida la fotografía hecha con la cámara por el usuario o seleccionada de la galería de imágenes para el perfil de usuario o para un regalo se procede a subirla a Cloud Storage de Firebase. De esta forma las imágenes estarán almacenadas en la nube y disponibles en cualquier momento para descargarlas y mostrarlas en la aplicación.

Para poder trabajar con Cloud Storage en primer lugar es necesario escribir las reglas de seguridad para determinar quién tiene acceso de lectura y escritura a los archivos almacenados. Se ha decidido por seguridad que sólo los usuarios registrados pueden leer o escribir en cualquier ruta de Cloud Storage.

```
rules_version = '2';
service firebase.storage {
  match /b/{bucket}/o {
    match /{allPaths=**} {
      allow read, write: if request.auth != null;
    }
  }
}
```

Figura 66: Reglas de acceso en Firebase Storage

Una vez hecho esto, para subir las fotografías o imágenes al almacenamiento en la nube en primer lugar se instancia la clase FirebaseStorage con el método getInstance() y se obtiene una referencia al Storage (StorageReference) con el método getReference(). Después se define la ruta exacta donde se almacenará la fotografía indicando el nombre de la carpeta y el nombre del archivo, ambos con el método child(). En el caso de la foto de perfil de un usuario la carpeta siempre será “ProfilePhotos” y el nombre del archivo corresponde al ID unívoco del usuario y la extensión jpg. Y para la

foto de cada regalo la carpeta siempre será “GiftPhotos” y el nombre del archivo corresponde al ID unívoco del regalo y la extensión jpg. Con el StorageReference exacto de la ruta se llama al método putFile() pasando la imagen en data Uri.

Se define el listener OnSuccessListener que llama a la función callback onSuccess() si la imagen se ha cargado en Cloud Storage. Y otro listener, OnFailureListener, que llama a la función callback onFailure si la imagen no se ha subido y, por tanto, ha fallado el almacenamiento.

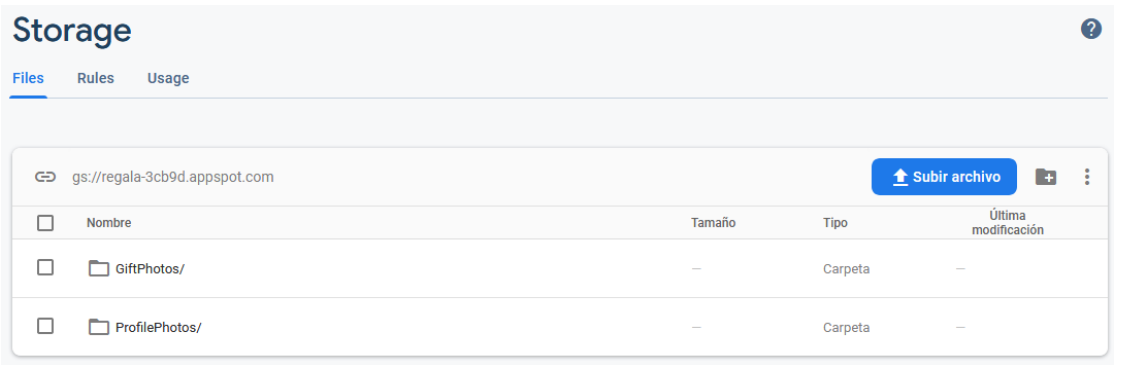


Figura 67: Consola de administración de Firebase Cloud Storage

3.2.13. Descarga de imágenes con Glide y FirebaseUI

Se ha utilizado la librería de Glide combinada con FirebaseUI para descargar, almacenar en caché y mostrar imágenes en una view (vista) con rapidez y facilidad desde Cloud Storage. Se utiliza en varios puntos de la aplicación para mostrar la foto de perfil tanto del usuario como de los contactos o usuarios a agregar a contactos y para mostrar la foto de cada regalo de una planificación.

Para obtener una imagen almacenada previamente en Cloud Storage, en primer lugar se instancia la clase FirebaseStorage con el método getInstance() y se obtiene una referencia al Storage (StorageReference) con el método getReference(). Después se recupera la ruta donde está almacenada la fotografía indicando el nombre de la carpeta y el nombre del archivo, ambos con el método child(). Con el StorageReference exacto de la imagen a mostrar se llama al método getDownloadUrl() para obtener la url de descarga. Se define el listener OnSuccessListener para detectar cuando se ha descargado la imagen en un objeto Uri. Finalmente, se llama a Glide especificando en with() el contexto de la actividad o fragment donde se muestra, en load() el Uri descargado convertido a String y en into() la view donde se quiere mostrar la imagen.

3.3. Pruebas

En este apartado se detalla el plan de pruebas seguido para comprobar la funcionalidad de la aplicación y así garantizar el correcto funcionamiento del sistema. Se han realizado pruebas unitarias sobre la aplicación a lo largo de toda la fase de desarrollo para ir solucionando errores y verificando tras realizar correcciones cuando se han dado errores. Al finalizar la implementación se ejecutaron pruebas de integración para probar todos los componentes de la aplicación en conjunto. También comentar que todas las pruebas han sido manuales.

3.3.1. Información sobre las pruebas

Las pruebas se han llevado a cabo con un smartphone Sony Xperia Z2 con Android Marshmallow (6.0.1). También con un smartphone Huawei P20 Pro con EMUI 9.1. y sobre el simulador que proporciona Android Studio para distintos dispositivos y niveles de S.O.

En el siguiente subapartado todas las pruebas documentadas están realizadas con el usuario usertest del cual pongo los datos de acceso.

Nombre de usuario: usertest
Email: usertest@uoc.edu
Contraseña: pass123

Tener en cuenta que el email no es real entonces no sirve para hacer la prueba del restablecimiento de la contraseña. Para esta prueba se ha utilizado un email personal.

3.3.2. Pruebas realizadas

A continuación muestro el listado de pruebas que se ha seguido:

Pantallas	Pruebas
Launcher (Inicial)	Muestra el logo de la aplicación mientras verifica si el usuario está conectado o no
	Determinar si el usuario está conectado o no con Firebase Authentication
Login	Acceso al restablecimiento de la contraseña o al registro de un nuevo usuario
	En caso de acceder a la app hace validaciones como email y contraseña requeridos, verificar si existe un usuario con ese email, verificar si la contraseña es correcta, etc.
	Validación de las credenciales
Registro	Dar la opción de hacer una foto o seleccionar de la galería para la foto de perfil de usuario.
	Validaciones como todos los campos informados, verificar si el nombre de usuario ya está en uso, al igual que el email, edad mínima de 13 años, etc.
	Creación del nuevo usuario en Firebase y almacenaje de los datos del usuario en Firestore.
Restablecer contraseña	Verificar si se ha informado el email y si hay algún usuario registrado con él.
	Enviar el email para reestablecer la contraseña.
Main / Planificaciones	Muestra las planificaciones que tiene en curso el usuario.
	Creación de planificación individual o en grupo.
	Desde cada planificación se puede acceder a visualizar, editar, finalizar y eliminar.
	Menú lateral con las distintas secciones de la app. Cerrar la sesión y redirigir a la pantalla de login.
Crear una planificación individual / grupo	Validar que como mínimo se ha informado el evento y a quien van dirigidos los regalos de la planificación.
	Acceso a añadir regalo o editarlo directamente en la misma pantalla.
	Añadir la planificación y los regalos en Firestore con el usuario como planificador. En caso de planificación en grupo añadir los participantes en Firestore, indicando cual es el planificador.
	Mostrar la nueva planificación en el listado de planificaciones en curso.
Añadir/Editar regalo	Validar que como mínimo se ha informado el nombre del usuario.
	Dar la opción de hacer una foto o seleccionar de la galería para la foto de un regalo.
Editar grupo	Visualizar el listado de participantes del grupo, quien ha pagado o no y el importe que debe cada participante.
	Probar a añadir contactos como participantes y eliminar participantes. Verificar que cuando cambia el número de participantes se actualiza el importe que debe cada participante.
Visualizar grupo	Visualizar el listado de participantes del grupo, quien ha pagado o no y el importe que debe cada participante.

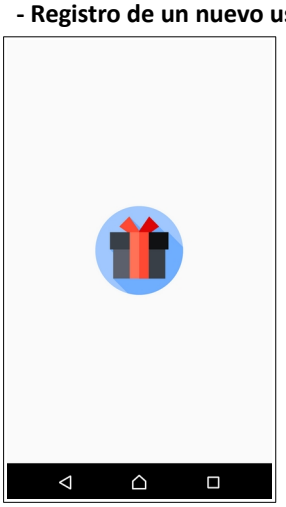
Visualizar una planificación individual / grupo	Mostrar los datos de la planificación y sus regalos. En el caso de grupo acceder a visualizar el grupo y se muestra el listado de participantes (planificador incluido).
Editar una planificación individual / grupo	Mostrar los datos de la planificación y sus regalos.
	Acceso a añadir nuevos regalos.
	Actualizar los días que quedan hasta la fecha, el precio de los regalos y el presupuesto que queda.
	Actualizar la planificación y sus regalos en Firestore. En el caso de grupo también los participantes.
Finalizar una planificación individual / grupo	Finalizar una planificación desde el listado y desde el detalle de la misma.
	Actualizar a finalizada la planificación en Firestore.
	La planificación finalizada no se muestra en el listado de planificaciones en curso.
Eliminar una planificación individual / grupo	Eliminar una planificación desde el listado y desde el detalle de la misma.
	Eliminar los regalos de la planificación y esta misma en Firestore. Los participantes en caso de grupo.
	La planificación eliminada no se muestra en el listado de planificaciones en curso.
Cuenta de usuario	Muestra la información del usuario.
	Acceso a editar el perfil de usuario, a cambiar la contraseña y eliminar la cuenta.
Editar perfil de usuario	Muestra la información del usuario.
	Dar la opción de hacer una foto o seleccionar de la galería para la foto de perfil de usuario.
	Validaciones como todos los campos informados, verificar si el nombre de usuario ya está en uso, al igual que el email, edad mínima de 13 años, etc.
	Actualizar el email de acceso en Firebase y los datos del usuario en Firestore.
Cambiar contraseña	Verificar que la contraseña actual indicada es igual a la contraseña de acceso del usuario, que la contraseña nueva y la de confirmación son iguales, etc.
	Actualizar la contraseña de acceso en Firebase.
Eliminar cuenta	Mostrar un popup para verificar si realmente se desea borrar el usuario y sus datos.
	Cancelar la eliminación de la cuenta.
	Eliminar la cuenta del usuario en Firebase y todos los datos creados en Firestore.
Lista de deseos	Mostrar la lista de deseos del usuario y buscar en la lista de deseos.
	Añadir un deseo en la lista y en la lista de deseos del usuario en Firestore.
	Eliminar un deseo de la lista y también de la lista de deseos del usuario en Firestore.
Contactos	Mostrar la lista de contactos del usuario y buscar en la lista de contactos.
	Eliminar un contacto de la lista y también de contactos del usuario en Firestore.
Añadir contactos	Buscar en la lista de usuarios que no están añadidos a contactos.
	Añadir un usuario a contactos y desaparece de la lista de usuarios que no están añadidos a contactos.
	Añadir al usuario en contactos del usuario en Firestore.

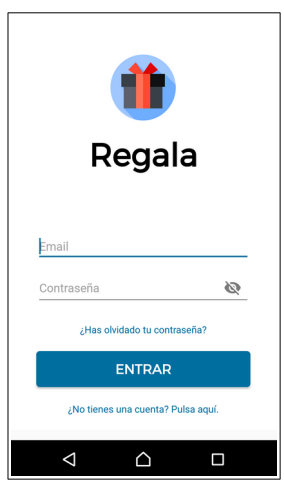
3.4. Funcionalidades de la aplicación

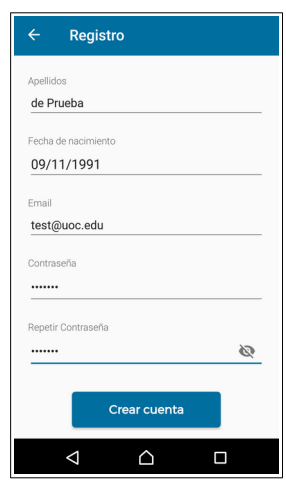
A continuación, se muestran capturas de la ejecución de la aplicación en las que se observan las principales funcionalidades definidas en los casos de uso del proyecto.

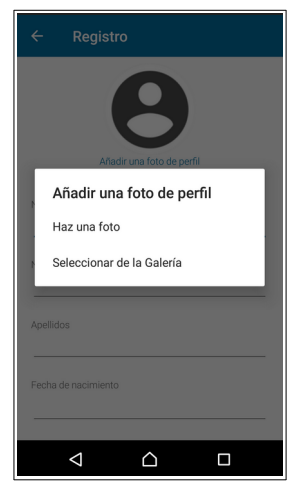
3.4.1. Gestión de usuarios

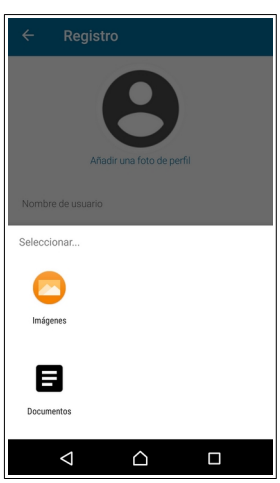
- Registro de un nuevo usuario:

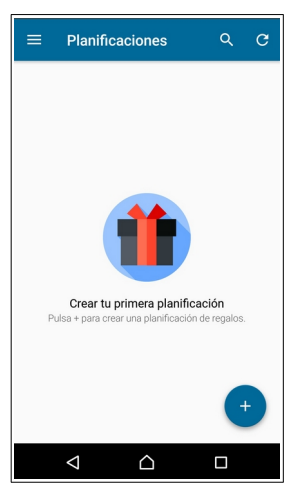




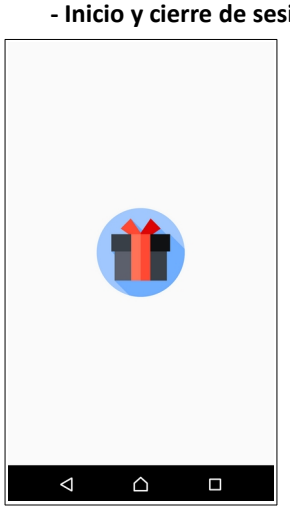


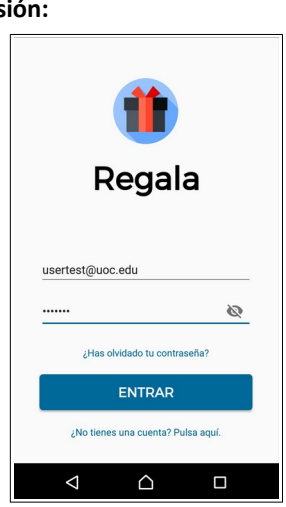


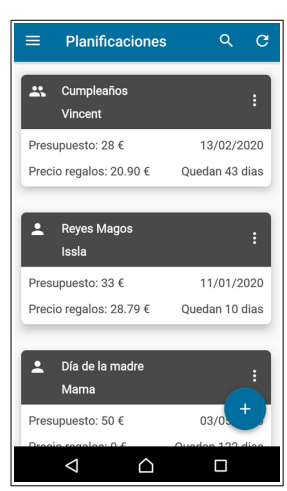


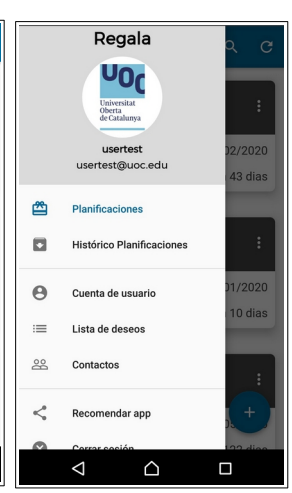


- Inicio y cierre de sesión:

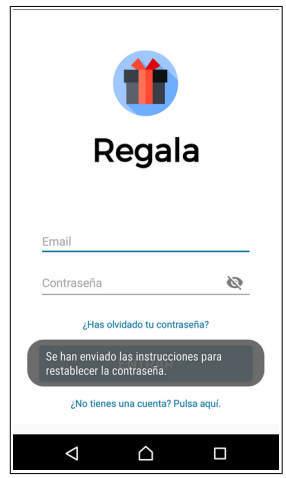




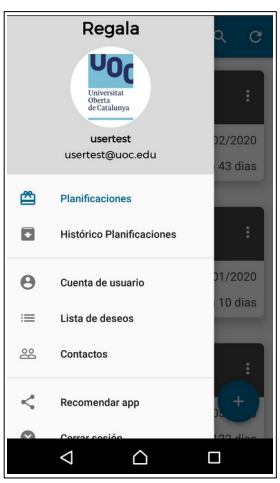


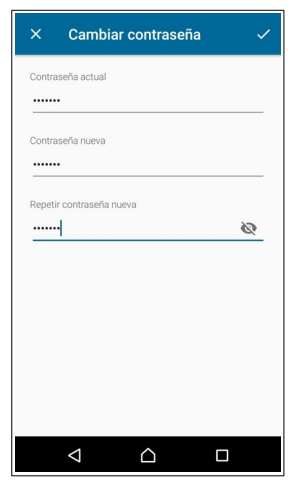


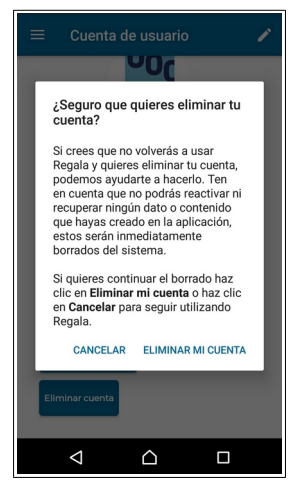
- Restablecer la contraseña



- Visualizar y editar la cuenta de usuario, cambiar la contraseña y eliminar la cuenta:

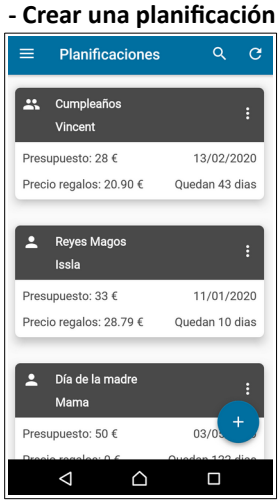


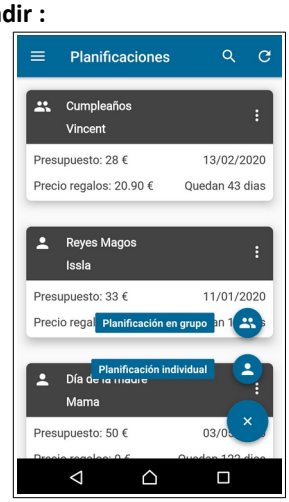




3.4.2. Gestión de las planificaciones

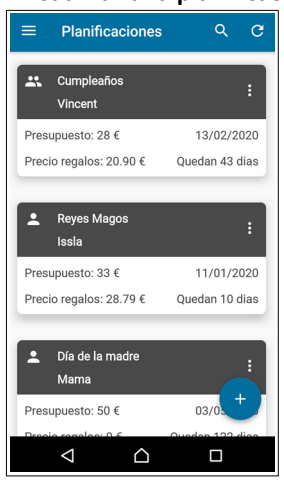
- Crear una planificación y añadir :

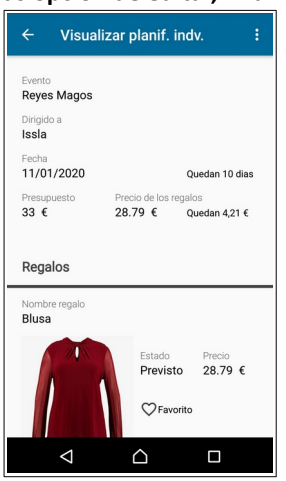


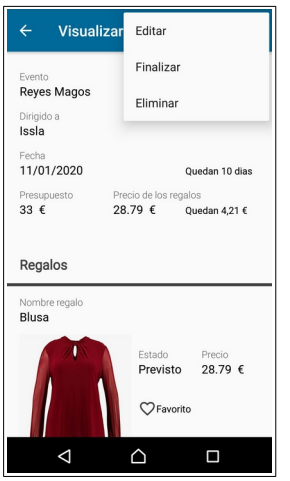


La planificación en grupo es exactamente igual que la individual excepto que existe un botón para acceder a la edición del grupo y los regalos se valoran, no existe la opción de marcar favorito.

- Visualizar una planificación (más opción de editar, finalizar o eliminar):

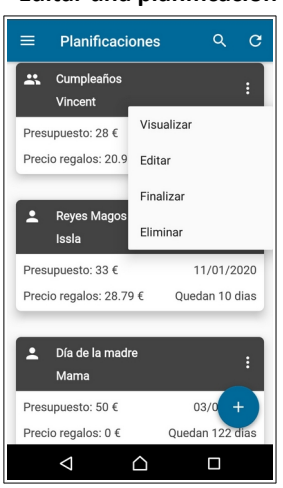


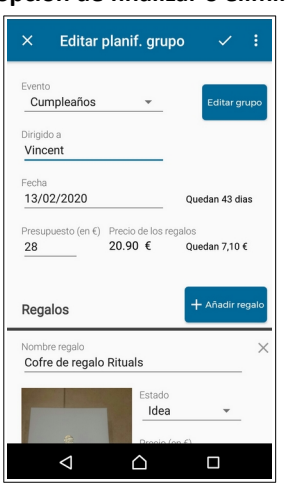


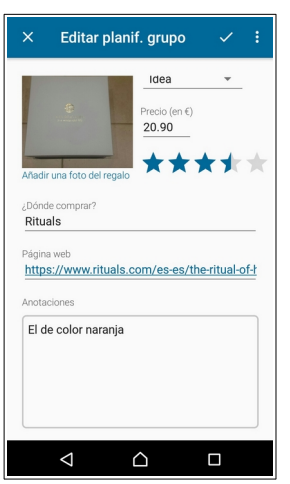


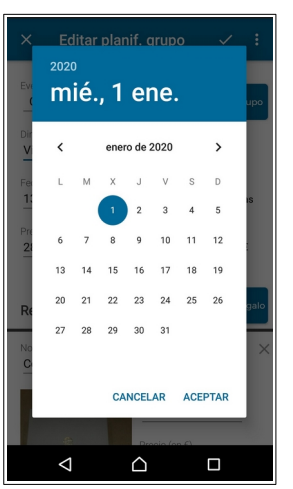
Pulsando el botón de más opciones se abre un menú para escoger entre editar, finalizar o eliminar la planificación. En caso de una planificación en grupo sólo el planificador tiene esas opciones, los participantes sólo pueden editar. Cuando se finaliza o elimina una planificación se redirige automáticamente a la pantalla de planificaciones.

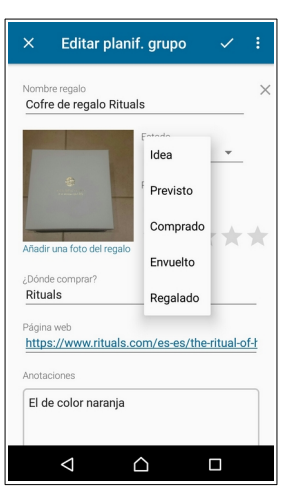
- Editar una planificación (más opción de finalizar o eliminar):

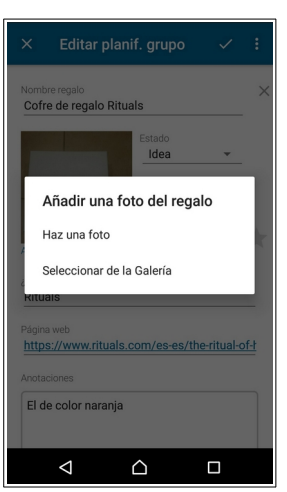












Pulsando el botón de más opciones se abre un menú para escoger entre finalizar o eliminar. Para eliminar un regalo sólo es necesario pulsar la X que hay en la parte superior derecha de cada regalo. En caso de una planificación en grupo sólo

el planificador puede finalizarla o eliminarla. Además, los participantes no pueden modificar ningún campo de la planificación aunque si pueden añadir nuevos regalos y editarlos excepto cambiar el estado del regalo.

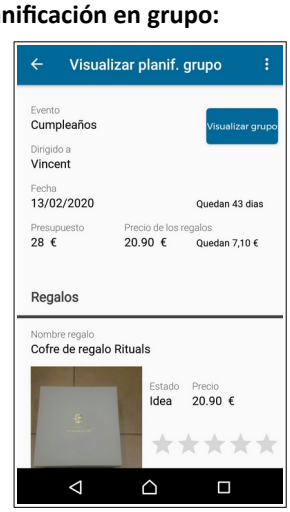
- Finalizar una planificación:

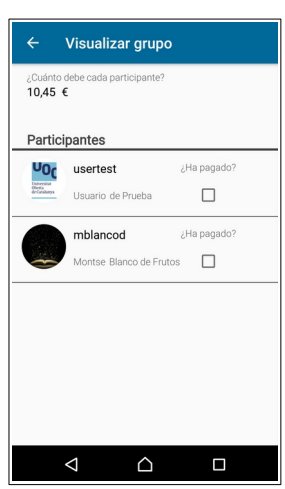
También se puede finalizar cuando se está visualizando o editando una.

- Eliminar una planificación:

También se puede eliminar desde el listado de planificaciones (en más opciones) o editando esta. En el caso del histórico de planificaciones, desde más opciones en una planificación del listado o cuando se está visualizando una.

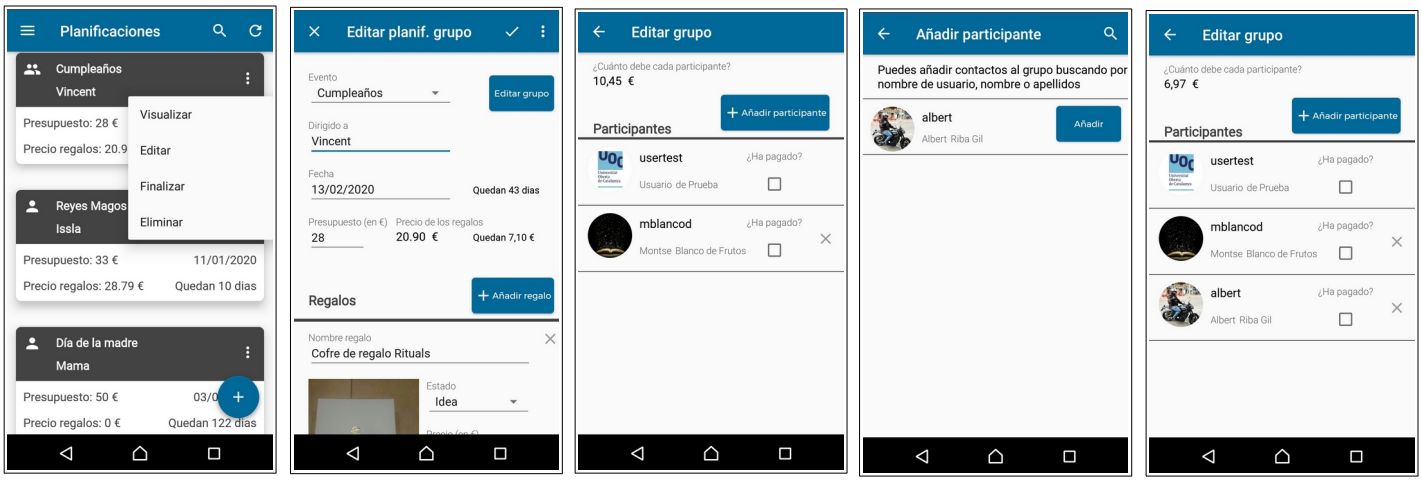
- Visualizar el grupo de una planificación en grupo:





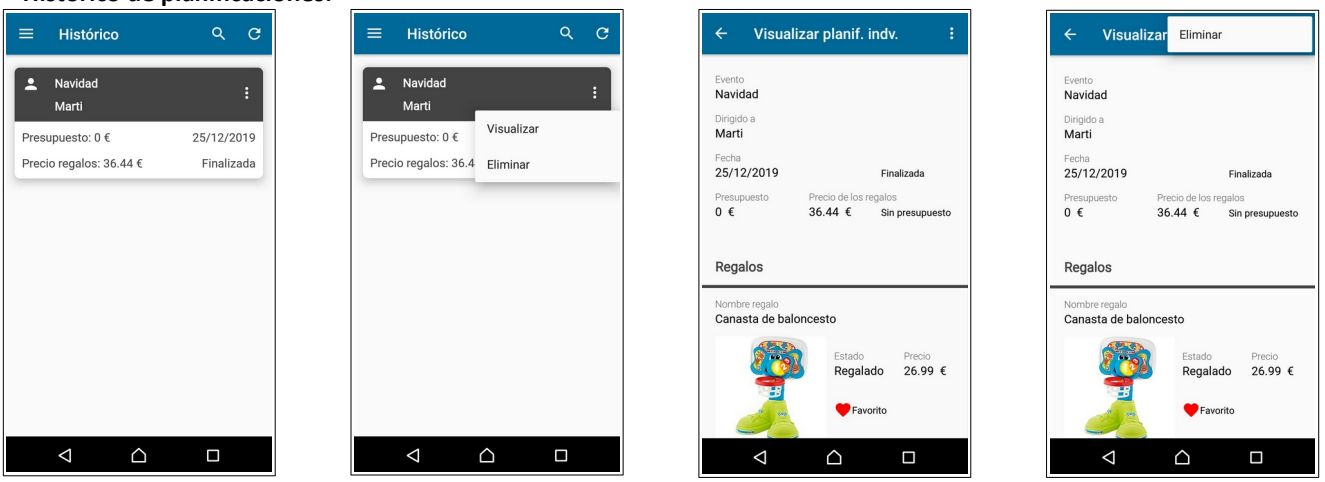
El primer usuario de la lista de participantes es el planificador.

- Editar el grupo de una planificación en grupo y añadir un participante:



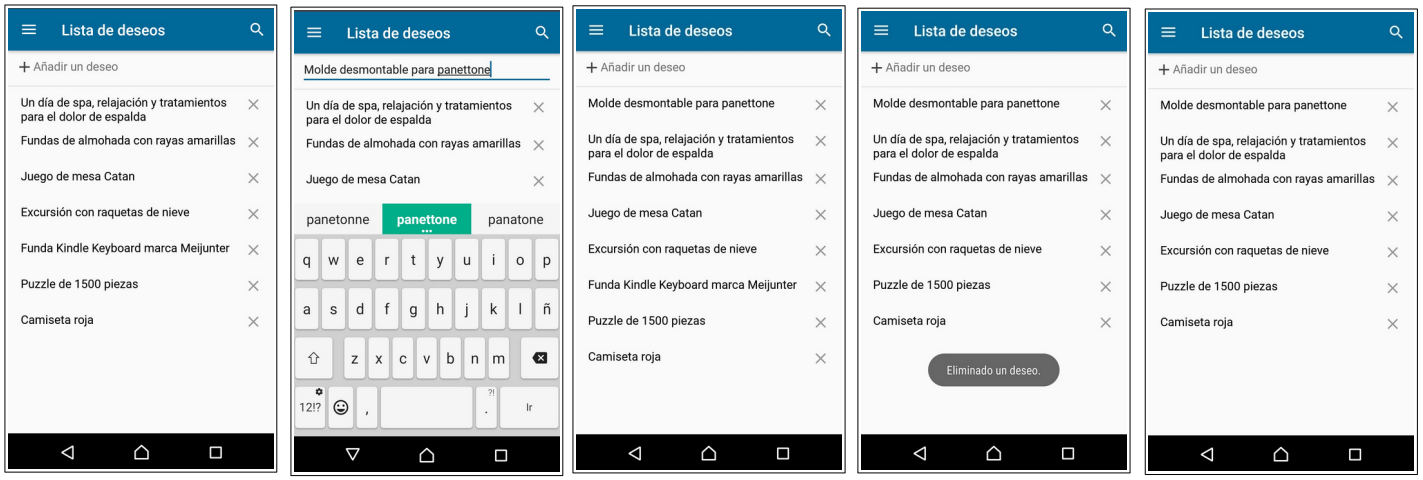
Sólo el planificador puede editar el grupo, añadir o eliminar participantes y marcar que han pagado su parte de los regalos. Para quitar a un participante del grupo sólo es necesario pulsar la X de la derecha de cada uno.

- Histórico de planificaciones:



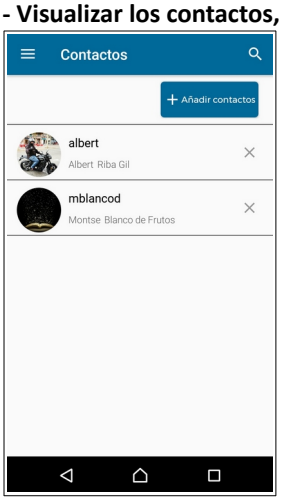
3.4.3. Gestión de la lista de deseos

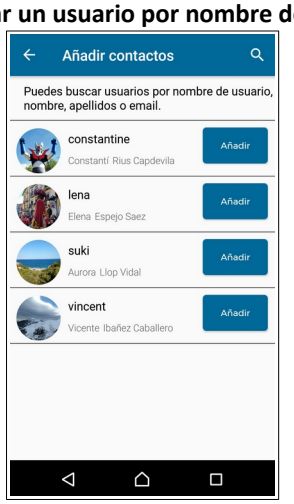
- Visualizar la lista de deseos y añadir y eliminar un deseo:

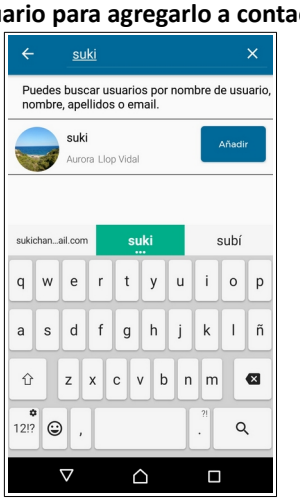


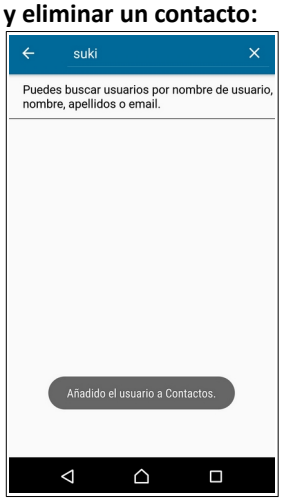
3.4.4. Gestión de los contactos

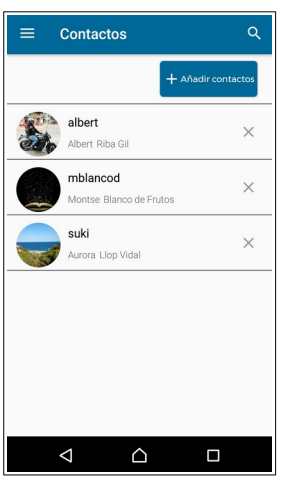
- Visualizar los contactos, buscar un usuario por nombre de usuario para agregarlo a contactos y eliminar un contacto:

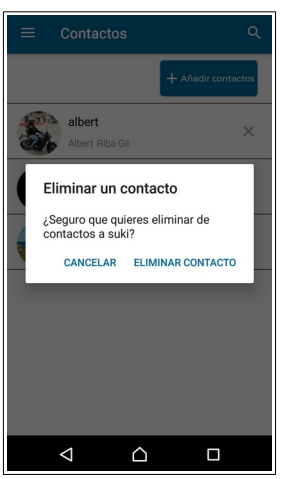


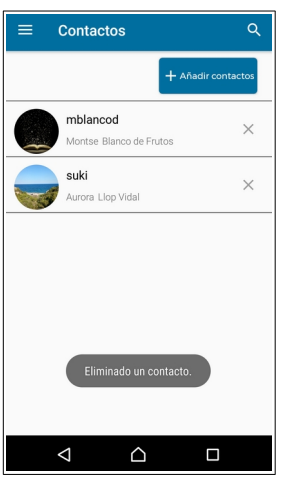












4. Conclusiones

Una vez finalizada la aplicación y efectuadas las pruebas del software es hora de parar a reflexionar, observar y comprobar el trabajo realizado y todos los nuevos conocimientos adquiridos durante este semestre.

La experiencia a nivel personal ha sido gratificante y a la vez intensa, dedicándole bastante tiempo y esfuerzo. El trabajo realizado ha ayudado a poner en práctica y aumentar los conocimientos obtenidos a lo largo de todo el máster, pudiendo así afianzar diversos conocimientos. Ha sido una oportunidad para aprender más acerca del desarrollo de aplicaciones Android, conocer nuevas tecnologías como Firebase y aprender a construir sistemas que combinen ambos elementos (frontend – app Android y backend – Firebase).

Respecto a los objetivos funcionales y requisitos planteados al principio del trabajo, se puede decir que en general se han logrado casi todos ellos y los requerimientos no funcionales se han cumplido todos. Las funciones que no se han podido implementar por falta de tiempo han sido las siguientes:

- Ordenar las planificaciones por evento, persona a quien va dirigida y fecha.
- Almacenaje y carga de las valoraciones de los regalos en las planificaciones en grupo.
- Visualizar el perfil de usuario y la lista de deseos de un contacto.
- Recomendar la aplicación.
- Todo el módulo de gestión de las notificaciones:
    - Activar/Desactivar la notificación de fecha límite próxima.
    - Visualizar las notificaciones y responder la solicitud de un contacto.
    - Configurar las notificaciones.

Diferentes factores han llevado a esta situación. Por un lado, los conocimientos básicos en desarrollo móvil de la autora de este proyecto repercutieron negativamente en el avance ya que condicionaron la agilidad y la determinación en la toma de decisiones. Las primeras semanas la implementación estuvo estancada y el avance fue mínimo. Y aunque se aumentaron las horas de dedicación para el desarrollo no se han logrado realizar todas las funcionalidades planteadas. Y otro problema ha sido el alcance del proyecto, se planteó una idea algo ambiciosa respecto al tiempo disponible para llevarla a cabo y los recursos de que se dispone.

En cuanto al seguimiento de la planificación, ha sido vital definir un plan de trabajo a seguir durante todo el desarrollo del TFM para poder cumplir con los objetivos y tener un mayor control del proceso para asegurar el éxito del mismo. También ha ayudado el ser constante con los objetivos a cumplir y no dejar la documentación para el último momento.

En todo el proyecto se han respetado las fechas de entrega establecidas desde un principio para las diferentes pruebas de evaluación continua a entregar durante el desarrollo del TFM. Las dos primeras fases, el plan de trabajo y el diseño y la arquitectura, se cumplieron sin mayor problema en los tiempos previstos y sin desviaciones.

Sin embargo, la planificación de la tercera fase del TFM resultó ser la menos precisa y la más complicada. En el diagrama de Gantt se indicaron unas fechas en concreto para cada una de las tareas que forman la etapa de implementación. Finalmente, se decidió realizar pequeñas tareas de cada una a lo largo de las semanas, según iba siendo necesario. Fue una mezcla de formación a medida que se iba avanzando el diseño y el desarrollo, configurando el entorno backend de Firebase cuando se hacía necesario por las acciones implementadas y las necesidades encontradas y, claro, desarrollando en sí la aplicación en Android Studio que tuvo una mayor carga de lo previsto. Al acabar la tercera fase quedaron sin implementar algunas funcionalidades planteadas en las anteriores fases aunque para la entrega final se consiguió reducir considerablemente las funciones no implementadas. Estas desviaciones se dieron por la poca experiencia en desarrollo móvil y una breve temporada con problemas de salud a mediados de noviembre que obligó a bajar un poco el ritmo durante unos días.

Para mitigar el efecto de esos riesgos e intentar finalizar la planificación con éxito, los días de vacaciones previstos en Diciembre, del 2 al 5, al final se alargaron hasta el día 11, el de la entrega. También se aumentaron las horas marcadas, pasando de unas 4 horas en los días laborables a más de 10 horas los días no laborales. Y en la última fase, para la entrega final, se siguieron dedicando más horas de las previstas para entregar sin problema el trabajo en la entrega final.

Respecto a la metodología, el modelo en cascada ha permitido seguir las diferentes fases de forma sucesiva con especial hincapié en las fases previas de análisis y diseño. Aunque se considera que se ha visto afectada al emplearla conjuntamente con la metodología DCU que al tener un componente altamente iterativo ha repercutido en la naturaleza lineal del modelo en cascada. Esto se debe a que tanto en la fase de diseño como en la de desarrollo se han tomado decisiones tanto de diseño como funcionales que han hecho variar y ampliar de alguna forma los requerimientos determinados en el análisis. Esto no respeta del todo la metodología escogida que da por cerradas las fases anteriores cuando se empieza una nueva.
De todas formas, la aplicación del DCU ha sido muy interesante aunque no se han podido realizar iteraciones posteriores. Ha ayudado a comprender la importancia de situar las necesidades y características del usuario en el centro del diseño de la aplicación para asegurar la consecución de un producto que cubra las necesidades de todos los usuarios potenciales y facilite el logro de sus objetivos.
Añadir además, que la realización de los prototipos de alta fidelidad y el diseño técnico, aunque tomaron bastante tiempo llevarlos a cabo, ayudó en la fluidez del desarrollo pudiendo centrar las energías en la implementación y diseñar las pantallas mucho más rápido al tener las ideas más claras.

4.1. Líneas de trabajo futuras

Durante la realización de la aplicación han ido surgiendo varias opciones de ampliación para el proyecto que podrían quedar como posibles cambios y futuras mejoras de la aplicación. Aunque primero se debería tratar de finalizar las funcionalidades que no se han podido implementar, mencionadas en este mismo capítulo, también sería interesante, con la versión actual implementada, aplicar de nuevo la metodología del Diseño Centrado en el Usuario. De hecho es un enfoque de diseño que involucra al usuario en todas las fases del trabajo, desde el análisis hasta la evaluación del sistema. Teniendo en cuenta todos estos aspectos, algunas de las ideas y funcionalidades que se podrían añadir para mejorar y enriquecer la aplicación son:

- Realizar una evaluación exhaustiva de la aplicación testeando con un número considerado de usuarios para la mejora y orientación del diseño y así hasta satisfacer las necesidades de los usuarios y del producto en sí.
- En las las planificaciones en grupo debería considerarse la implementación de una herramienta de bloqueo o actualización en tiempo real mientras se está editando para que no haya conflictos cuando más de una persona está aplicando cambios en la misma planificación.
- Dar la posibilidad de eliminar la fotografía de perfil y/o de los regalos. Además, sería de utilidad contar con una galería de fotos en cada regalo para, por ejemplo, disponer de imágenes tomadas desde diferentes ángulos de un producto. Y al pulsar sobre una imagen que esta se visualizara en un mayor tamaño.
- Incorporar nuevas funcionalidades para complementar el control presupuestario en las planificaciones en grupo como quienes han pagado los regalos en una planificación en grupo y cuanto le deben el resto de participantes.
- Recuperar la idea desestimada por falta de tiempo para implementar un chat en las planificaciones en grupo y así proporcionar una herramienta de comunicación entre el planificador y los participantes.
- Mejorar el diseño de la aplicación para ser más atractivo y moderno para el usuario final.
- Adaptar la aplicación para tablets. Aunque las pantallas diseñadas se adaptan a cualquier tamaño de pantalla, en el caso de dispositivos de mayor tamaño la interfaz no resulta muy estética y sería imprescindible crear nuevos diseño para aprovechar al máximo todo el espacio disponible en pantalla.
- Optimizar la aplicación para que consuma el menor número de recursos de RAM, CPU y batería posible ya que afectan el rendimiento del sistema.

5. Glosario

- **Android.** Sistema operativo desarrollado por Google para dispositivos móviles basado en el núcleo de Linux y otros software de código abierto.

- **Android Studio.** Entorno de desarrollo integrado oficial para la plataforma Android.

- API. Acrónimo en inglés de Application Programming Interface (en español Interfaz de Programación de aplicaciones). Conjunto de subrutinas, funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para se utilizada por otro software como una capa de abstracción.

- APK. Android Application Package (en español Paquete de Aplicación Android) es un paquete para el sistema operativo Android que se usa para distribuir e instalar componentes empaquetados para la plataforma Android.

- App. Abreviatura de la palabra en inglés application. Es una aplicación de software diseñada para ejecutarse en smartphones, tablets y otros dispositivos móviles.

- **Backend.** Parte de un sistema que procesa la entrada desde la parte que interactúa con los usuarios y devuelve generalmente una respuesta a esa otra parte que lo expone al usuario de una forma entendible para este.

- Backup. Hispanización de back up, una expresión inglesa que quiere decir "respaldo". En la informática significa copia de seguridad de la información almacenada en un dispositivo electrónico que permite la restauración de los datos.

- Diagrama de Gantt. Herramienta gráfica que muestra el tiempo de dedicación previsto para cada tarea a lo largo del proyecto.

- DCU. Acrónimo de Diseño Centrado en el Usuario que es una filosofía de diseño de productos y aplicaciones que sitúa al usuario en el centro de todo el proceso.

- EMUI. Sistema operativo móvil personalizado basado en Android que Huawei usa en la mayoría de los dispositivos de teléfonos inteligentes de marca Huawei y Honor.

- Firebase. Plataforma móvil creada por Google, cuya principal función es desarrollar y facilitar la creación de apps de elevada calidad de una forma rápida.

- Google Drive. Servicio de alojamiento de archivos que fue introducido por la empresa Google. Permite guardar los archivos de forma segura y abrirlos o editarlos desde cualquier dispositivo.

- **Google Play.** Plataforma de distribución digital de aplicaciones móviles para los dispositivos con sistema operativo Android, como una tienda en línea desarrollada y operada por Google.

- IDE. Integrated Drive Electronics (en español, entorno de desarrollo integrado) es una aplicación informática que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de software.

- **JSON**. Acrónimo de JavaScript Object Notation. Formato de texto sencillo para el intercambio de datos

- Nube. La computación en la nube es un paradigma que permite ofrecer servicios de computación a través de una red, que usualmente es Internet. Permite almacenar y acceder a datos y programas a través de Internet.

- Open Source. Modelo de desarrollo de software basado en la colaboración abierta. La idea del código abierto se centra en la premisa de que al compartir el código, el programa resultante tiende a ser de calidad superior al software propietario, es una visión técnica.

- Premium. Adjetivo en inglés que se utiliza para calificar a un servicio o un producto de características especiales. Lo habitual es que sea un privilegio destinado a aquellos consumidores que pagan un monto adicional.

- RAM. Abreviatura en inglés de Random Access Memory (en español memoria de acceso aleatorio) que se utiliza como memoria de trabajo de computadoras y otros dispositivos para el sistema operativo, los programas y software.

- SDK. Acrónimo en inglés de Software Development Kit (en español Kit de desarrollo de software). Generalmente es un conjunto de herramientas de desarrollo de software que permite a un desarrollador crear una aplicación informática para un sistema concreto.

- **Smartphone.** Término en inglés que se utiliza para denominar a un teléfono inteligente que es un equipo celular con funciones más avanzadas que las de un teléfono corriente.

- TFM. Abreviatura de la palabra Trabajo Final de Máster.

6. Bibliografía

Abizanda, V. (2019). *Test con usuarios: Qué es y pasos a llevar a cabo.* [En línea]. Disponible en: https://www.hiberus.com/crecemos-contigo/test-con-usuarios-que-es-pasos/ (Fecha de consulta: Octubre 2019).

Almirall López, J. (2009). *Disseny centrat en l'usuari per a dispositius mòbils.* [En línea]. Disponible en: http://cv.uoc.edu/webapps/xwiki/wiki/matm1202es/view/Main/WebHome (Fecha de consulta: Octubre 2019).

Android Developers (2019) *Documentación para desarrolladores de apps.* [En línea]. Disponible en: https://developer.android.com/docs (Fecha de consulta: Noviembre y Diciembre 2019)

Android Studio Faqs (2019) *Encontrarás tutoriales, dudas resueltas y todo lo que necesites para iniciarte en el fabuloso mundo de la programación de apps móviles en Android.* [En línea]. Disponible en: https://androidstudiofaqs.com/ (Fecha de consulta: Noviembre y Diciembre 2019)

Axl Softs (2016). *Aplicación Regalos Presupuesto de Navidad.* [En línea]. Disponible en: https://play.google.com/store/apps/details?id=com.axlsofts.christmas (Fecha de consulta: Septiembre 2019).

Calvo-Fernández Rodríguez, A.; Ortega Santamaría, S.; Valls Saez, A. *Métodos de evaluación con usuarios*. PID_00176614. Barcelona: Editorial UOC.

Clarisó Viladrosa, R.; Rodríguez Bermúdez, J.R. (2013). *Treball final de màster*. Barcelona: Editorial UOC.

DEV (2019) *Android – DEV Community.* [En línea]. Disponible en: https://dev.to/t/android (Fecha de consulta: Noviembre y Diciembre 2019)

Dodenhof, Henning (2014). *CircleImageView - A fast circular ImageView perfect for profile images.* [En línea]. Disponible en: https://github.com/hdodenhof/CircleImageView (Fecha de consulta: Noviembre y Diciembre 2019)

Domínguez, P. (2017). *En qué consiste el modelo en cascada.* [En línea]. Disponible en: https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada (Fecha de consulta: Septiembre 2019).

engApps (2018). *Aplicación Christmas Gift List.* [En línea]. Disponible en: https://play.google.com/store/apps/details?id=com.xmaslist (Fecha de consulta: Septiembre 2019).

Firebase (2019) *Documentación de Firebase.* [En línea]. Disponible en: https://developer.android.com/docs (Fecha de consulta: Noviembre y Diciembre 2019)

Garreta Domingo, M.; Mor Pera, E. *Disseny centrat en l'usuari*. PID_00176048. Barcelona: Editorial UOC.

Idler, S. (2001). *How user scenarios help to improve your UX*. [En línea]. Disponible en: https://usabilla.com/blog/how-user-scenarios-help-to-improve-your-ux/ (Fecha de consulta: Octubre 2019).

Judd, Sam (2019) *Glide - Fast and efficient image loading for Android* [En línea]. Disponible en: https://github.com/bumptech/glide (Fecha de consulta: Noviembre y Diciembre 2019)

lastpresent (2016). *Aplicación lastpresent: regalos y compras.* [En línea]. Disponible en: https://play.google.com/store/apps/details?id=com.imobils.lastpresent (Fecha de consulta: Septiembre 2019).

Leewrigg (2019). *Aplicación Gift Tracker – Birthday and Christmas Gift Lists!* [En línea]. Disponible en: https://play.google.com/store/apps/details?id=moloapps.gifttracker (Fecha de consulta: Septiembre 2019).

Manchón, E. (2003). *Herramientas para diseño web: los personajes (persona) y escenarios.* [En línea]. Disponible en: https://www.alzado.org/articulo.php?id_art=110 (Fecha de consulta: Octubre 2019).

Medialoha (2018). *Aplicación Christmas Gift List.* [En línea]. Disponible en: https://play.google.com/store/apps/details?id=com.medialoha.android.christmasgifts (Fecha de consulta: Septiembre 2019).

Medium (2019) Tagged in Android. [En línea]. Disponible en: https://medium.com/tag/android (Fecha de consulta: Noviembre y Diciembre 2019)

Montalto, P. (2018). *Aplicación Xmas Gifts List.* [En línea]. Disponible en: https://play.google.com/store/apps/details?id=it.xabaras.android.giftlist.xmas (Fecha de consulta: Septiembre 2019).

Prieto Blázquez, J.; Ramírez Vique, R.; Morillo Pozo, J.D.; Domingo Prieto, M. (2011). *Tecnologia i desenvolupament en dispositius mòbils*. Barcelona: Editorial UOC.

Sáenz Higueras, N.; Vidal Oltra, R. *Redacció de textos cientificotècnics*. P08/19018/00445. Barcelona: Editorial UOC.

Stackoverflow (2019) *Foro de ayuda Stackoverflow.* [En línea]. Disponible en: https://stackoverflow.com/ (Fecha de consulta: Noviembre y Diciembre 2019)

Wikipedia (2019). *Cliente-servidor*. [En línea]. Disponible en: https://es.wikipedia.org/wiki/Cliente-servidor (Fecha de consulta: Octubre 2019).

Youtube (2019) [En línea]. Disponible en: https://www.youtube.com/ (Fecha de consulta: Noviembre y Diciembre 2019)

7. Anexos

En los anexos se añade toda la información adicional que permite conocer más a fondo algunos aspectos específicos que debido a su extensión se tratan fuera de esta memoria.

- **Anexo 1 – Respuestas encuestas**. Este documento contiene las respuestas de cada uno de los participantes a la encuesta realizada en la fase de Usuarios y Contexto de uso del DCU.
- **Anexo 2 – Documentos del test de usuario.** En este documento se muestra el cuestionario inicial, el pre-test, los comentarios de las tareas, el post-test y la carta de consentimiento firmada del test de usuario realizado en la fase de Evaluación del DCU.
- **Anexo 3 – Casos de uso**. En este documento se definen todos los casos de uso del sistema incluyendo los mostrados en esta memoria y los que no se han incluido en esta debido a su extensión.
- **Anexo 4 – Manual de usuario**. Este documento contiene una pequeña guía de instalación y consideraciones a tener en cuenta de la aplicación y el manual de usuario.