

Crypto Tracker

Antonio José Álvarez Aguilera

Máster universitario de Desarrollo de aplicaciones para dispositivos móviles
Trabajo final de máster DADM

Carles Garrigues Olivella
Pau Dominkovics Coll

Diciembre de 2021



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-

SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Crypto Tracker</i>
Nombre del autor:	<i>Antonio José Álvarez Aguilera</i>
Nombre del consultor/a:	<i>Pau Dominkovics Coll</i>
Nombre del PRA:	<i>Carles Garrigues Olivella</i>
Fecha de entrega (mm/aaaa):	12/2021
Titulación:	Máster universitario de Desarrollo de aplicaciones para dispositivos móviles
Área del Trabajo Final:	Trabajo final de máster DADM
Idioma del trabajo:	<i>Español</i>
Palabras clave	<i>Criptomoneda, Android, noticias</i>
<p>Resumen del Trabajo (máximo 250 palabras): <i>Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.</i></p>	
<p>La presente memoria describe el proceso de conceptualización, diseño e implementación de <i>Crypto Tracker</i>, una aplicación nativa para dispositivos que funcionen bajo el sistema operativo Android, creada a partir del IDE Android Studio con el lenguaje de programación Kotlin.</p> <p>El objetivo de <i>Crypto Tracker</i> es permitir llevar el control de las criptomonedas que le interesen al usuario a través de la evolución y alertas de precios, noticias y tasas de cambio.</p> <p>La idea del proyecto surge de observar el <i>boom</i> que vive ahora mismo todo lo relacionado con las criptomonedas, que ha llevado a un aumento considerable del número de aplicaciones relacionadas con este mundo en la <i>store</i> de Android. La mayoría de esas aplicaciones se basan en mostrar información global sobre las diferentes criptomonedas que se encuentran en el mercado, siendo complementada según las preferencias y elecciones del usuario. Por ello, se ha desarrollado una aplicación que, a partir de las criptomonedas que el usuario decida seguir, se organice el resto de la información proporcionada por la aplicación, sin necesidad de mostrar aquella que sea superflua al usuario. Las elecciones del usuario son la piedra angular del funcionamiento de la aplicación.</p> <p>Debido a las desviaciones sufridas en la planificación, el producto entregado no cuenta con todas las funcionalidades planteadas.</p>	

Al finalizar este proyecto, se ha conseguido crear una aplicación con buen rendimiento, que consume mucha información en tiempo real a partir de APIs externas, y con un diseño minimalista y agradable a la vista.

Abstract (in English, 250 words or less):

This report describes the process of conceptualization, design and implementation of *Crypto Tracker*, a native application for devices running under the Android operating system, created from the Android Studio IDE with the Kotlin programming language.

The goal of *Crypto Tracker* is to allow to keep track of cryptocurrencies that interest the user through price evolution and alerts, news and exchange rates.

The idea of the project comes from observing the boom that everything related to cryptocurrencies is experiencing right now, which has led to a considerable increase in the number of applications related to this world in the *Android store*. Most of these applications are based on displaying global information about the different cryptocurrencies that are on the market, being complemented according to the user's preferences and choices. Therefore, an application has been developed that, based on the cryptocurrencies that the user chooses to follow, organizes the rest of the information provided by the application, without the need to show that which is superfluous to the user. The user's choices are the cornerstone of the application's operation.

Due to the deviations suffered in the planning, the delivered product does not have all the proposed functionalities.

At the end of this project, we have managed to create an application with good performance, which consumes a lot of information in real time from external APIs, and with a minimalist and pleasing design.

Índice

1. Introducción	11
1.1 Contexto y justificación del Trabajo.....	11
1.2 Objetivos del Trabajo	15
1.2.1 Objetivos funcionales.....	15
1.2.2. Objetivos no funcionales.....	16
1.3 Enfoque y método seguido.....	16
1.4 Planificación del Trabajo	18
1.4.1. Recursos de hardware y software	18
1.4.2. Planificación de tareas.....	20
1.4.3. Planificación temporal de tareas.....	21
1.4.4. Riesgos en la planificación	22
1.5 Breve resumen de productos obtenidos.....	23
1.6 Breve descripción de los otros capítulos de la memoria	23
2. Diseño	24
2.1. Encuestas.....	24
2.1.1. Pregunta 1	24
2.1.2. Pregunta 2	25
2.1.3. Pregunta 3	25
2.1.4. Pregunta 4	26
2.1.5. Pregunta 5	26
2.1.6. Pregunta 6	27
2.1.7. Pregunta 7	27
2.1.8. Pregunta 8	28
2.1.9. Pregunta 9	28
2.1.10. Pregunta 10	29
2.1.11. Conclusiones	29
2.2. Usuarios y contexto de uso	30
2.2.1. <i>Personas</i> y escenarios de uso.....	30
2.3. Diseño conceptual.....	33
2.3.1. Árbol de navegación	33
2.3.2. Flujo de interacción.....	35
2.4. Prototipado	36
2.4.1. Sketches	36
2.4.2. Prototipado de alta fidelidad	44
2.4.3. Evaluación	49
2.5. Diseño técnico.....	52
2.5.1. Definición de los casos de uso	52
2.5.2. Diseño de la arquitectura	71
3. Implementación	74
3.1. Desarrollo	74
3.1.1. Librerías utilizadas	77
3.2. Pruebas	78
3.3. Observaciones tras implementación	84
4. Conclusiones	85
5. Glosario	87

6. Bibliografía	87
7. Anexos	89
7.1. Manual de usuario.....	89

Lista de figuras

Figura 1. Mercado global de criptomonedas a lo largo del tiempo [2].....	11
Figura 2. Evolución del mercado global de Android frente a iOS [6].....	17
Figura 3. Cuota de mercado de las versiones de Android [6]	18
Figura 4. Visión genérica de la planificación temporal de tareas	21
Figura 5. Planificación temporal de las tareas de la PEC 1 - Plan de trabajo...21	
Figura 6. Planificación temporal de las tareas de la PEC 2 - Diseño.....	21
Figura 7. Planificación temporal de las tareas de la PEC 3 - Implementación..22	
Figura 8. Planificación temporal de las tareas de la PEC 4 - Entrega final.....	22
Figura 9. Encuesta - Pregunta 1	24
Figura 10. Encuesta - Pregunta 2	25
Figura 11. Encuesta - Pregunta 3	25
Figura 12. Encuesta - Pregunta 4	26
Figura 13. Encuesta - Pregunta 5	26
Figura 14. Encuesta - Pregunta 6	27
Figura 15. Encuesta - Pregunta 7	27
Figura 16. Encuesta - Pregunta 8	28
Figura 17. Encuesta - Pregunta 9	28
Figura 18. Encuesta - Pregunta 10	29
Figura 19. Imagen de User - Persona 1	30
Figura 20. Imagen de User - Persona 2	32
Figura 21. Árbol de navegación	34
Figura 22. Flujo interacción	35
Figura 23. Pantalla Registro.....	36
Figura 24. Pantalla Login	36
Figura 25. Pantalla Home.....	37
Figura 26. Pantalla detalle criptomoneda.....	38
Figura 27. Pantalla noticias	39
Figura 28. Pantalla noticias guardadas	40
Figura 29. Pantalla detalle de noticia	40
Figura 30. Pantalla de conversión de monedas	41
Figura 31. Pantalla alertas	42
Figura 32. Pantalla detalle alerta.....	42
Figura 33. Pantalla cuenta del usuario.....	43
Figura 34. Pantalla preferencias	44
Figura 35. Pantalla registro de usuario	45
Figura 36. Pantalla login.....	45
Figura 37. Vista menú	46
Figura 38. Vista añadir criptomoneda	46
Figura 39. Pantalla detalle criptomoneda.....	46
Figura 40. Pantalla noticias	46
Figura 41. Pantalla noticias guardadas	47
Figura 42. Pantalla detalle noticia	47
Figura 43. Pantalla conversor	47
Figura 44. Pantalla alertas	47

Figura 45. Pantalla detalle alerta.....	48
Figura 46. Pantalla usuario	48
Figura 47. Pantalla preferencias	48
Figura 48. Primera implementación tras evaluación de usuarios.....	49
Figura 49. Segunda implementación tras evaluación de usuarios.....	50
Figura 50. Segunda implementación tras evaluación de usuarios (2)	50
Figura 51. Tercera implementación tras evaluación de usuarios.....	51
Figura 52. Patrón arquitectónico MVVM, establecido por Google	71
Figura 53. Arquitectura limpia, definida por Robert C. Martin	73
Figura 54. Diagrama entidad – relación de la base de datos.....	73
Figura 55. Clase Either.....	75
Figura 56. Métodos que se encuentra en la clase abstracta BaseService	76
Figura 57. Imágenes manual de usuario.....	89
Figura 58. Imágenes manual de usuario (2)	90
Figura 59. Imágenes manual de usuario (3)	90

Lista de tablas

Tabla 1. Recursos desglosados de hardware y software.....	19
Tabla 2. Definición de tareas.....	20
Tabla 3. Análisis de riesgos	22
Tabla 4. Ficha User - Persona 1	31
Tabla 5. Ficha User - Persona 2	33
Tabla 6. CU – 01: Registro de usuario	52
Tabla 7. CU – 02: Login de usuario.....	53
Tabla 8. CU – 03: Eliminar cuenta de usuario.....	53
Tabla 9. CU – 04: Logout de usuario	54
Tabla 10. CU – 05: Acceso a la pantalla de home	55
Tabla 11. CU – 06: Añadir criptomoneda	55
Tabla 12. CU – 07: Eliminar criptomoneda	56
Tabla 13. CU – 08: Actualizar criptomoneda.....	56
Tabla 14. CU – 09: Añadir criptomoneda al portfolio	57
Tabla 15. CU – 10: Modificar criptomoneda del portfolio	57
Tabla 16. CU – 11: Eliminar criptomoneda del portfolio.....	58
Tabla 17. CU – 12: Acceder a la pantalla de detalle de una criptomoneda	58
Tabla 18. CU – 13: Actualizar la información en la pantalla de detalle de una criptomoneda.....	59
Tabla 19. CU – 14: Modificar el intervalo de tiempo en la gráfica mostrada en la pantalla de detalle de una criptomoneda	60
Tabla 20. CU – 15: Acceder a los enlaces de interés de una criptomoneda	60
Tabla 21. CU – 16: Acceder a la pantalla de noticia	61
Tabla 22. CU – 17: Filtrar noticias.....	62
Tabla 23. CU – 18: Acceder al detalle de una noticia	62
Tabla 24. CU – 19: Acceder a noticias guardadas.....	63
Tabla 25. CU - 20: Guardar noticia	63
Tabla 26. CU – 21: Borrar una noticia guardada.....	64
Tabla 27. CU – 22: Convertir un valor.....	65
Tabla 28. CU – 23: Añadir información adicional a la conversión de un valor ..	65
Tabla 29. CU – 24: Acceder a la pantalla de alertas.....	66
Tabla 30. CU – 25: Añadir una alerta.....	67
Tabla 31. CU – 26: Ver una alerta.....	67
Tabla 32. CU – 27: Editar una alerta.....	68
Tabla 33. CU – 28: Eliminar una alerta	68
Tabla 34. CU – 29: Notificar al usuario del cambio de precio de una criptomoneda	69
Tabla 35. CU – 30: Acceder a la pantalla de cuenta del usuario	69
Tabla 36. CU – 31: Modificar la información del usuario.....	70
Tabla 37. CU – 32: Acceder a la pantalla de preferencias.....	71
Tabla 38. CU – 33: Editar preferencias	71
Tabla 39. Pruebas en pantalla de splash.....	78
Tabla 40. Pruebas en pantalla de registro	79
Tabla 41. Pruebas en pantalla de login.....	79
Tabla 42. Pruebas en pantalla de home	80
Tabla 43. Pruebas en pantalla de detalle de una criptomoneda	81
Tabla 44. Pruebas en pantalla de noticias	81

Tabla 45. Pruebas en pantalla de una noticia	82
Tabla 46. Pruebas en pantalla de noticias guardadas	82
Tabla 47. Pruebas en pantalla de conversión	83
Tabla 48. Pruebas en pantalla de usuario.....	84
Tabla 49. Pruebas en pantalla de preferencias del usuario	84

1. Introducción

1.1 Contexto y justificación del Trabajo

Según un estudio realizado en 2018 por “IAB Spain”, hay un 48% de usuarios que utilizan diariamente aplicaciones relacionadas con las finanzas [1]. Dentro de estas aplicaciones, encontramos aquellas relacionadas con el mundo de las criptomonedas.

Según el informe “Measuring Global Crypto Users” de Julio de 2021 realizado por www.crypto.com [2], el número global de usuarios que realizan operaciones con criptomonedas ha alcanzado el número de 221 millones en junio de 2021, necesitando solamente 4 meses para duplicar la población mundial de 100 millones a 200 millones.

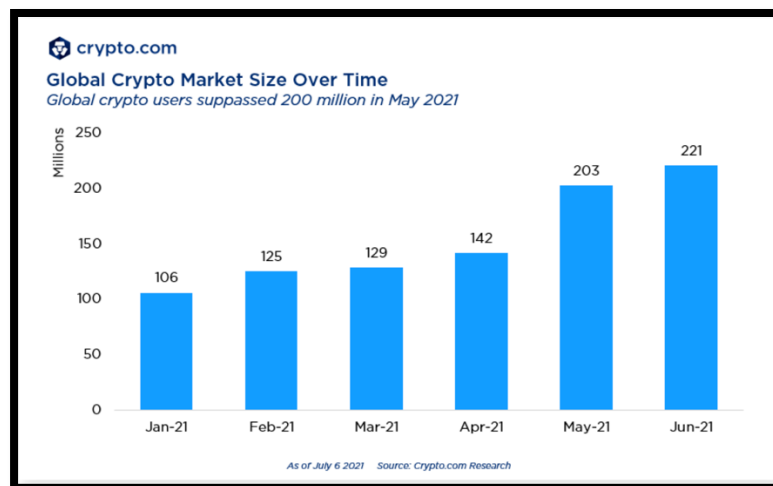


Figura 1. Mercado global de criptomonedas a lo largo del tiempo [2]

Este aumento considerable de usuarios que utilizan criptomonedas se debe a que son una oportunidad de inversión global disponible para cualquier persona en cualquier lugar del mundo [3]. Sin embargo, aunque son instrumentos altamente rentables, son volátiles y con cierto factor de riesgo. Este auge de usuarios se debe, sobre todo, a que grandes instituciones financieras estén aceptando el uso de criptomonedas en sus plataformas, como Mastercard o Visa. Además, El Salvador se ha convertido el primer país del mundo en permitir el uso de una criptomoneda como moneda de curso legal [2].

Pero, ¿Qué son las criptomonedas y qué las hacen tan especiales? Las criptomonedas son un medio digital de intercambio, que cumplen el papel de una moneda fiduciaria. Sin embargo, su valor se basa en algo digital, ya que utiliza métodos criptográficos para asegurar transacciones financieras, controlar la creación de nuevas unidades y verificar las transferencias de activos [4]. Estos métodos criptográficos se basan en la tecnología blockchain. Esta tecnología

está compuesta por una red de ordenadores independientes del control centralizado y que mantiene un registro de todas las transacciones que han ocurrido y que ocurrirán en un futuro [3].

Con el fin de que un usuario, a partir de su smartphone, pueda obtener información relacionada con el mundo de las criptomonedas, han aparecido numerosas aplicaciones en las *stores* de los diferentes sistemas operativos, convirtiéndose en un nicho de mercado bastante amplio.

De entre todas las aplicaciones con características similares a la aplicación que se pretende desarrollar, se ha tomado un subconjunto de ellas para su análisis teniendo como base el número de descargas y su puntuación:

- **CoinGecko – BitCoin & Cryptoconcurrency Price Tracker** [5] es una aplicación que ofrece un análisis fundamental del mercado y permite llevar a cabo un seguimiento del precio, el volumen y la capitalización del mercado. Sus principales funcionalidades son:
 - Datos de mercado en tiempo real para más de 8.000 criptomonedas.
 - Creación de un portfolio de criptomonedas para el seguimiento de las ganancias y pérdidas.
 - Creación de alertas para recibir notificaciones cuando el precio de una criptomoneda varíe.
 - Capacidad de añadir un widget asociado a la aplicación.
 - Calculadora para la conversión de criptomonedas a monedas fiduciarias.
 - Información sobre plataformas de intercambio entre criptomonedas y monedas fiduciarias.
 - Gamificación de acciones por parte del usuario para conseguir descuentos, libros, NFTs, suscripciones Premium.

Sus puntos débiles son:

- Experiencia de usuario no muy agradable, muestra demasiada información en cada una de las pantallas de la aplicación.
 - No permite filtrar noticias.
 - En ciertos momentos, la aplicación funciona de forma lenta debido a la cantidad de datos que muestra.
 - Permite solamente registro mediante correo, no permite hacer login con las principales redes sociales.
- **The Crypto App – Widgets, Alertas, Noticias** [6] es una aplicación que permite hacer un seguimiento de los tipos de cambio entre las diferentes criptomonedas, recibir noticias y alertas en tiempo real sobre cualquier criptomoneda y calcular cambios a otras monedas o viceversa. Entre las diversas funcionalidades que contempla, destacan las siguientes:
 - Búsqueda en una lista de más de 1.000 criptomonedas.

- Posibilidad de configurar una lista prioritaria con las criptomonedas favoritas del usuario.
- Visualización detallada con gráficos de las criptomonedas actuales.
- Posibilidad de configurar alertas de precio.
- Posibilidad de agregar widgets.
- Información sobre las últimas novedades sobre criptomonedas.
- Interfaz agradable para el usuario.

Sus puntos débiles son:

- Versión de pago que contiene un gran número de las funcionalidades anteriormente mencionadas.
- Anuncios en la versión gratuita.

- **Crypto News: Noticias, Precios de Bitcoin, Alertas** [7] proporciona, a los entusiastas de las criptomonedas, un agregador de noticias y artículos para monitorear el mercado de criptomonedas, además de un administrador de portfolio. La aplicación está bien optimizada, ocupa poco espacio en disco y ahorra en la carga de batería. Sus principales funcionalidades son:

- Recopila noticias de diferentes fuentes.
- Permite crear un feed de noticias para recibir información de temas y fuentes concretas.
- Permite enviar notificaciones push como recordatorio para noticias.
- Las noticias proporcionadas están disponibles en diferentes idiomas.
- Datos de mercado en tiempo real para las diferentes criptomonedas disponibles.
- Posibilidad de configurar el tipo de gráfica para ver la evolución de las criptomonedas.
- Permite crear notificaciones para cambios en el precio o su capitalización.
- Permite gestionar un portfolio de criptomonedas.
- Agregar seguridad extra para poder acceder al portfolio (huella digital, contraseña o FaceID).

Sus puntos débiles son:

- Interfaz con demasiada información, hace torpe la navegación entre las diferentes pantallas de la aplicación.
- En la versión gratuita muestra anuncios.
- Versión de pago que contiene algunas de las funcionalidades anteriormente mencionadas. Además, en esta versión, permite enviar notificaciones de forma ilimitada y permite el modo offline en la lectura de noticias.

- **CoinMarketCap – Rastreador de precios criptográficos** [8] se define como el sitio web de seguimiento de precios de criptoactivos más referenciado del mundo en el espacio de criptomonedas, facilitando a los usuarios minoristas información imparcial, de alta calidad y precisa que permita al usuario sacar sus propias conclusiones. Las funcionalidades más destacadas son:
 - Capitalización en el mercado de las diferentes criptomonedas.
 - Permite establecer alertas de precios.
 - Proporciona gráficos interactivos para el usuario.
 - Configurar un portfolio para el seguimiento del precio de las criptomonedas.
 - Convertidor de divisas entre criptomonedas y monedas fiduciarias.
 - Herramienta para comparar criptomonedas.
 - Obtención de las tasas de las diferentes criptomonedas de manera precisa y en tiempo real.
 - Obtención de noticias de acuerdo a los intereses del usuario.
 - Seguridad al acceder a la aplicación mediante Face ID, Touch ID o código de acceso.

Sus puntos débiles son:

- El soporte de atención al cliente no es bueno.
- Las alertas de precio no están sincronizadas.
- Errores aleatorios a la hora de utilizar la aplicación.
- Interfaz de usuario sobrecargada, el usuario se pierde entre tanta información.
- Contiene anuncios.
- No permite hacer login con las principales redes sociales.

Tras realizar un análisis de las aplicaciones similares elegidas, vemos que todas tienen un punto en común, ofrecen información global que viene complementada con información personal según las preferencias y elecciones del usuario. Esto puede proporcionar una información más completa sobre el mundo de las criptomonedas, sin embargo, tal cantidad de información puede hacer que el usuario no se sienta cómodo, haciendo que la experiencia de usuario sea pobre.

La aplicación que se pretende desarrollar se enfoca en el caso contrario, es decir, a partir de las elecciones del usuario se organiza toda la información relacionada con el mundo de las criptomonedas que se va a mostrar en la aplicación, sin necesidad de mostrar aquella que sea superflua al usuario. Con este enfoque lo que se busca es que el usuario vea y use lo que quiere sin necesidad de obtener información extra. Todo ello se complementará con un diseño minimalista y claro, haciendo que la experiencia de usuario sea lo más agradable posible.

1.2 Objetivos del Trabajo

El objetivo principal de la aplicación es la de ofrecer información relevante sobre el mundo de las criptomonedas que, según las necesidades del usuario, se considere oportuna. Para ello, se utilizará todos los conocimientos que se han adquirido a lo largo de este máster.

De forma personal, se pretende ampliar los conocimientos y habilidades como desarrollador en las tecnologías que se van a utilizar.

1.2.1 Objetivos funcionales

- **Registro de usuario:** El usuario deberá registrarse para poder hacer uso de la aplicación, ya sea creando una cuenta nueva, solicitando el nombre, correo y contraseña; o bien, utilizando la cuenta de sus redes sociales.
- **Login de usuario:** El usuario se identificará proporcionando el correo y la contraseña introducidos en el registro; o bien, utilizando la cuenta de sus redes sociales.
- **Logout de usuario:** El usuario podrá cerrar sesión.
- **Eliminar cuenta de usuario:** El usuario podrá ser capaz de eliminar la cuenta creada.
- **Gestión de preferencias de la aplicación:** El usuario podrá configurar los parámetros esenciales para ver la aplicación como más le guste. Establecer el tiempo entre actualizaciones para obtener la evolución de las criptomonedas seleccionadas, configuración del umbral de tiempo en el que se quiere ver la evolución del precio de una criptomoneda, tipo de moneda fiduciaria para el funcionamiento de la calculadora de conversión...
- **Gestión de criptomonedas:** El usuario podrá añadir y eliminar criptomonedas, con el fin de crear un portfolio.
- **Gestión de la evolución de las criptomonedas:** El usuario podrá ver, de manera gráfica, la evolución del precio de mercado de las criptomonedas seleccionadas, así como las características generales de la misma.
- **Gestión del detalle de las criptomonedas:** El usuario podrá ver, de forma detallada, todas las características de la criptomoneda.
- **Calculadora de conversión:** El usuario podrá convertir una criptomoneda a una moneda fiduciaria, previamente seleccionada.

- **Gestión de alertas:** El usuario podrá establecer alertas para cuando una criptomoneda varíe su precio en un rango específico.
- **Gestión de noticias:** El usuario podrá revisar noticias relacionadas con las criptomonedas que ha añadido a su portfolio.
- **Filtrar noticias:** Las noticias podrán ser filtradas por criptomoneda y fecha de publicación.
- **Guardado de noticias:** El usuario podrá guardar noticias para que puedan ser leídas más tarde.

1.2.2. Objetivos no funcionales

- **Desarrollo:** La aplicación será desarrollada mediante el lenguaje de programación Kotlin, implementando principios que hagan que la aplicación sea escalable, con el fin de poder realizar mejoras en un futuro. El desarrollo será de código abierto.
- **Entorno:** La aplicación estará disponible para dispositivos móviles que hagan uso del sistema operativo Android.
- **Usabilidad:** La experiencia de usuario debe ser agradable, intuitiva y minimalista. El diseño de la interfaz permitirá una curva de aprendizaje baja.
- **Recurso de datos:** Obtención de los datos necesarios a partir de APIs públicas relacionadas con el mundo de las criptomonedas.
- **Almacenamiento:** La información será guardada entre la memoria del teléfono y el almacenamiento en la nube proporcionado por firebase.
- **Seguridad:** El sistema debe garantizar la privacidad de los datos personales introducidos por el usuario.
- **Fiabilidad:** Implementación de testing para garantizar que el software sea fiable, garantizando la calidad del producto con el fin de fidelizar al usuario y reducir los costes de mantenimiento.

1.3 Enfoque y método seguido

Realizar el estudio de aplicaciones similares en la stores disponibles ha servido para esclarecer algunos puntos, mejorar algunas ideas que se tenían o agregar otras funcionalidades que no estaban previstas a la aplicación. Además, también ha servido para identificar 2 tipos de aplicaciones relacionadas con las

criptomonedas. Aquellas que proporcionan información del mundo relacionado con las criptomonedas, y aquellas que proporcionan todo lo anterior, además de incluir una wallet. Una wallet es una herramienta con la que los usuarios almacenan y gestionan sus criptomonedas, permitiendo enviar y recibir criptomonedas a través de blockchain [9].

La estrategia adoptada será la de desarrollar un producto nuevo que contenga ciertas funcionalidades que utilizan otras grandes aplicaciones relacionadas con el mundo de las criptodivisas, pero utilizando como piedra angular la selección de las criptomonedas por parte del usuario para el funcionamiento de todas las funcionalidades de la aplicación. Todo ello acompañado de un diseño minimalista para que la experiencia de usuario sea lo más grata posible.

La aplicación será desarrollada para el sistema operativo Android. Esta elección se basa en un hecho que es muy relevante a la hora de elegir un sistema operativo u otro: Android es el responsable del 71% de todo el uso de smartphones en el primer trimestre 2021, siendo la plataforma dominante en la mayoría de países del mundo [10].

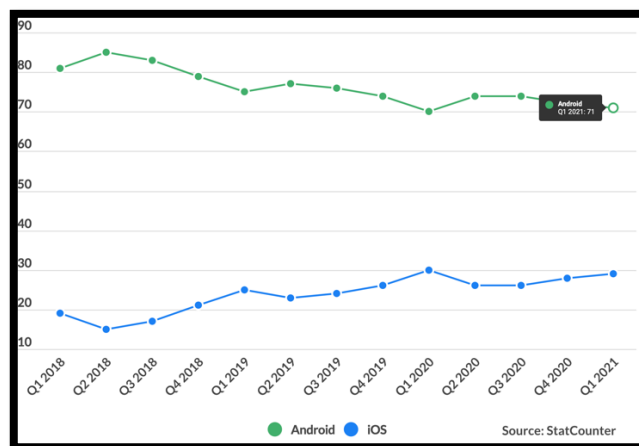


Figura 2. Evolución del mercado global de Android frente a iOS [6]

Para el tipo de aplicación a desarrollar se ha optado por un desarrollo nativo. Aunque el coste puede ser mayor si en un futuro pensamos en publicar la aplicación en el sistema operativo iOS, al tener que desarrollar la misma aplicación para dicho sistema operativo; desarrollar una aplicación nativa permite ofrecer una gran experiencia de usuario, un mejor rendimiento y aprovechar las características que nos proporciona el dispositivo [11].

El lenguaje elegido para desarrollar es Kotlin. La razón principal de uso es la novedad del propio lenguaje y que, recientemente, ha recibido el apoyo de Google. Además, el código en Kotlin es más seguro y más conciso que el código en Java. No sólo es útil par el desarrollo de nuevas aplicaciones, sino que sus archivos pueden coexistir con los archivos de Java, existiendo interoperabilidad entre lenguajes, lo que significa que este lenguaje también tiene cabida en aplicaciones ya existentes en Java [12].

La versión mínima de la API de Android elegida será la versión 8.0-8.1, más conocida como Oreo. Al elegir esta versión, se ocupa un 78,3% del mercado Android [10]. Gracias a esto, no tendremos que realizar demasiadas tareas de mantenimiento, aprovechando todos los beneficios que nos aporta esta versión.

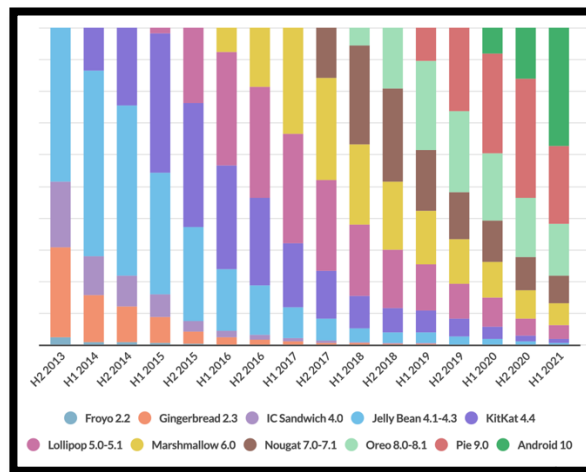


Figura 3. Cuota de mercado de las versiones de Android [6]

Una vez seleccionado la plataforma y el lenguaje de programación, es hora de elegir el método de desarrollo. Para la realización de este trabajo final de máster, debemos de tener los requisitos bien definidos, ya que se debe de cumplir tiempos de entrega. Por ello, el desarrollo se va a dividir en fases totalmente secuenciales. El modelo de desarrollo escogido va a ser el “modelo waterfall”. Este modelo le da mucho énfasis a la planificación, a los tiempos y a las fechas límite, encajando perfectamente en como se pretende desarrollar esta aplicación.

Con el enfoque explicado, se pretende conseguir todos los objetivos que se propongan a lo largo de todo el semestre y, si surgiera algún imprevisto, poder reestablecer los requisitos con el fin de entregar una aplicación lo más acorde posible a lo definido en un principio.

1.4 Planificación del Trabajo

1.4.1. Recursos de hardware y software

Los recursos que se van a utilizar para el desarrollo de esta aplicación se enumeran a continuación:

Hardware	
Dispositivo de desarrollo	<p>MacBook Pro 13":</p> <ul style="list-style-type: none"> ▪ Procesador: Intel i5 de doble núcleo a 2,3 GHz. ▪ Capacidad: 256 GB de SSD. ▪ Memoria: 8 GB de RAM. ▪ Gráfica: Intel Iris Plus Graphics. ▪ Versión: macOS BigSur 11.5.2
Software	
Emulador	Nexus 6P API 30.
Planificación	GranttProject: Herramienta open source que permite la elaboración de diagramas Gantt.
Prototipado	Adobe XD: Herramienta de edición de gráficos que funciona para crear interfaces de páginas web y de aplicaciones.
Diagramas	Draw.io: Aplicación web (www.draw.io) que permite crear y editar diagramas de cualquier tipo.
Documentación	Microsoft Word: Programa de procesamiento de texto.
Control de versiones	Git: Sistema de control de versiones distribuido de código abierto que permite descargar software, realizar cambios y generar versiones.
Interfaz para el control de versiones	Github Desktop: Herramienta que permite interactuar con Github . Github es una interfaz que permite almacenar y administrar el código en la nube utilizando git .
Entorno de desarrollo integrado (IDE)	Android Studio: Software que cuenta con herramientas y servicios para que los desarrolladores puedan crear aplicaciones Android.
Almacenamiento en la nube	Firebase: Plataforma en la nube para el desarrollo de aplicaciones web y móvil.

Tabla 1. Recursos desglosados de hardware y software

1.4.2. Planificación de tareas

A continuación, se definen las tareas a realizar:

Tareas
Plan de trabajo
Definición de la idea
Contexto y justificación del trabajo
Estudio de mercado
Definición de objetivos funcionales y no funcionales
Enfoque y método seguido
Planificación del trabajo
Diseño de la aplicación
Usuarios y contexto de uso
Diseño conceptual
Prototipado
Evaluación
Definición de casos de uso
Diseño de arquitectura
Implementación de las funcionalidades
Registro de usuario
Login de usuario
Logout de usuario
Eliminar cuenta de usuario
Gestión de preferencias de la aplicación
Gestión de criptomonedas
Gestión de la evolución de las criptomonedas
Gestión del detalle de las criptomonedas
Calculadora de conversión
Gestión de alertas
Gestión de noticias
Filtro de noticias
Guardado de noticias
Testing
Entrega final
Revisión y corrección de la memoria
Manual de usuario
Elaboración de la presentación

Tabla 2. Definición de tareas

1.4.3. Planificación temporal de tareas

Para los recursos temporales se ha tenido en cuenta que es necesario complementar el desarrollo de este proyecto con un trabajo a jornada completa. Además, habrá ciertos días que, por causas ajenas o personales, no se pueda realizar el tiempo estimado para los días planeados (Por ejemplo: viajes, bodas, festivos de navidad...). Con todo esto, la disponibilidad en número de horas, es la siguiente:

- Días laborables (de lunes a jueves): **2,5 horas** cada día.
- Días laborables (viernes): **4 horas**.
- Sábados, domingos y días festivos: **5 horas** cada día.

Definida la disponibilidad en número de horas, la cantidad total estimada para la realización del proyecto es de **338 horas**.

A continuación, se mostrará la distribución de las tareas a lo largo de una línea de tiempo utilizando un diagrama de Gantt:

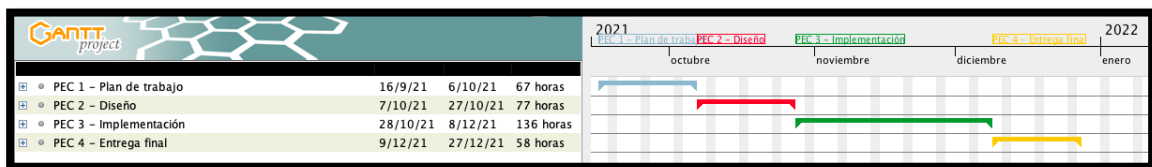


Figura 4. Visión genérica de la planificación temporal de tareas

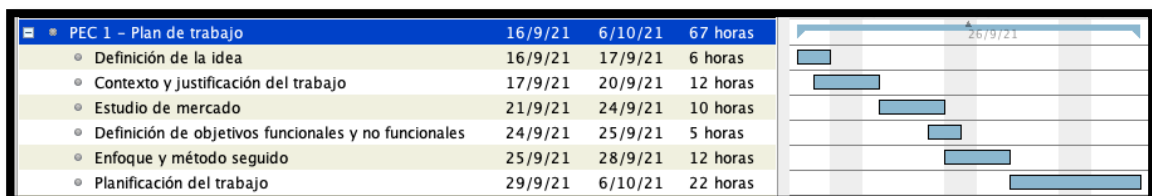


Figura 5. Planificación temporal de las tareas de la PEC 1 - Plan de trabajo

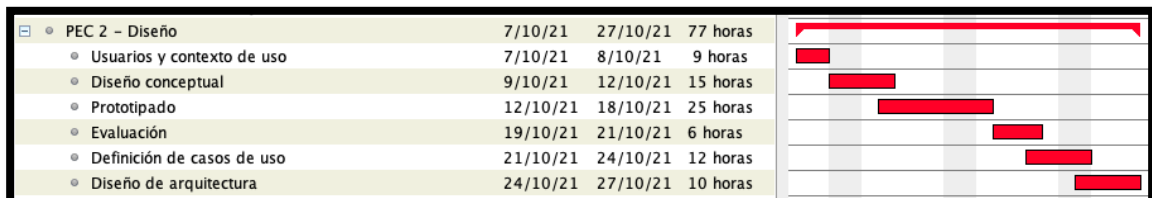


Figura 6. Planificación temporal de las tareas de la PEC 2 - Diseño

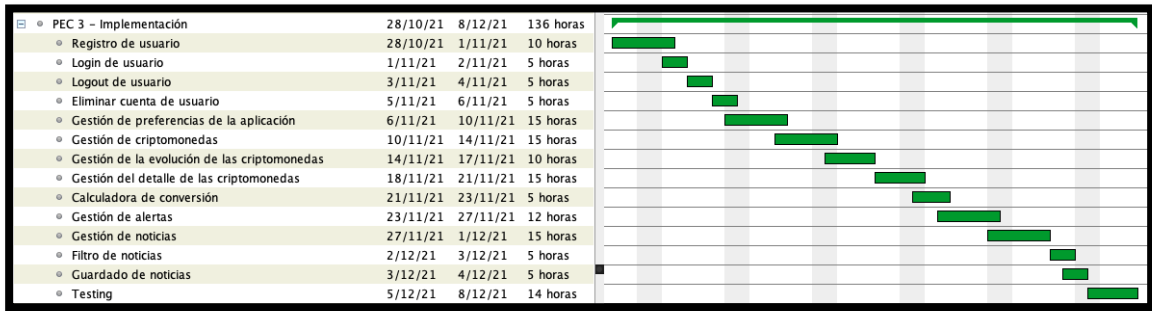


Figura 7. Planificación temporal de las tareas de la PEC 3 - Implementación

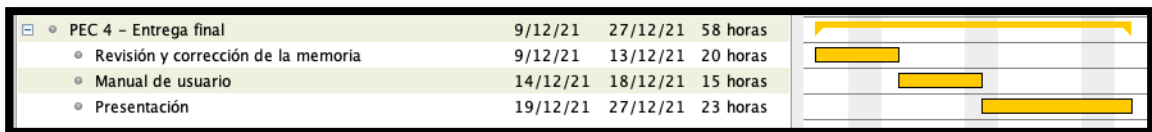


Figura 8. Planificación temporal de las tareas de la PEC 4 - Entrega final

1.4.4. Riesgos en la planificación

Como se ha comentado en el apartado anterior, habrá ciertos días que, por causas ajenas o personales, no se pueda realizar el tiempo de la tarea asignada. Por ello, vamos a incluir aquellos riesgos que puedan darse a lo largo de la elaboración de este proyecto:

Riesgo	Probabilidad	Impacto
Imposibilidad de realizar el tiempo estimado para una tarea en cierto día por causas ajenas o personales.	Alta	Alto
Desconocimiento en el uso de librerías que utilizaremos a lo largo del desarrollo.	Media	Alto
No haber planificado adecuadamente los tiempos de las tareas establecidas.	Baja	Alto
Cambio de trabajo en octubre de 2021, teniendo que aprender nuevas tecnologías.	Alta	Alto

Tabla 3. Análisis de riesgos

1.5 Breve resumen de productos obtenidos

Al finalizar el proyecto, se obtendrá una serie de elementos:

- Documento con la memoria del proyecto.
- APK nativa para dispositivos que utilicen el sistema operativo Android con una versión 8.0 – 8.1 o superior.
- Manual de usuario en el cual se explica como hacer un uso correcto de la aplicación.
- Código fuente de la aplicación.
- Vídeo que contenga la presentación del proyecto.

1.6 Breve descripción de los otros capítulos de la memoria

En los siguientes capítulos se documentará los demás aspectos relevantes del desarrollo de la aplicación, como son el diseño y arquitectura, la implementación a nivel de código y la conclusión final.

El capítulo dos estará centrado en el diseño. Se realizará una investigación para conocer el *target* del público que usará la aplicación, el prototipado a bajo nivel y alto nivel, y los casos de uso. Para terminar este capítulo, se tratará el diseño de la arquitectura explicando que se va a utilizar y por qué.

El capítulo tres contemplará la implementación, a nivel de código, de las funcionalidades explicadas anteriormente junto con el diseño definido. Se recogen las decisiones tomadas a nivel de desarrollo, así como las librerías utilizadas para conseguir el funcionamiento deseado. También se explicará las pruebas realizadas.

El capítulo cuatro expondrá las conclusiones sacadas a lo largo del desarrollo de la aplicación, informando de la consecución de los objetivos establecidos en la planificación y de las lecciones aprendidas durante el desarrollo.

En el capítulo cinco se definirá todo el glosario de términos.

En el seis cinco se abordan todas las fuentes utilizadas a lo largo de toda la memoria. Se citarán tanto las que se han obtenido de la web, como de libros o artículos.

En el siete se incluirá un manual de usuario. Este manual de usuario se explicará el correcto uso de la aplicación, con el fin de facilitar el aprendizaje al usuario que haga uso de ella.

2. Diseño

Cuando se quiere desarrollar una aplicación, se necesita saber quienes podrían ser los potenciales usuarios que hagan uso de la misma, así como si dicha aplicación cumple con las necesidades del usuario. Por ello, se hará uso de la filosofía/metodología del diseño centrado en el usuario (DCU), que consiste en involucrar al usuario en todas las fases del desarrollo con el fin de crear un producto que satisfaga todos sus objetivos, motivaciones y necesidades de manera útil y usable [13].

2.1. Encuestas

Como método de indagación, se ha optado por el uso de encuestas. Las encuestas son el método más conocido para obtener información acerca de los usuarios en base a un conjunto definido de preguntas [14].

Para la encuesta se ha elaborado un total de diez preguntas, y se ha encuestado a un total de 40 personas. A continuación, se muestran los datos obtenidos:

2.1.1. Pregunta 1

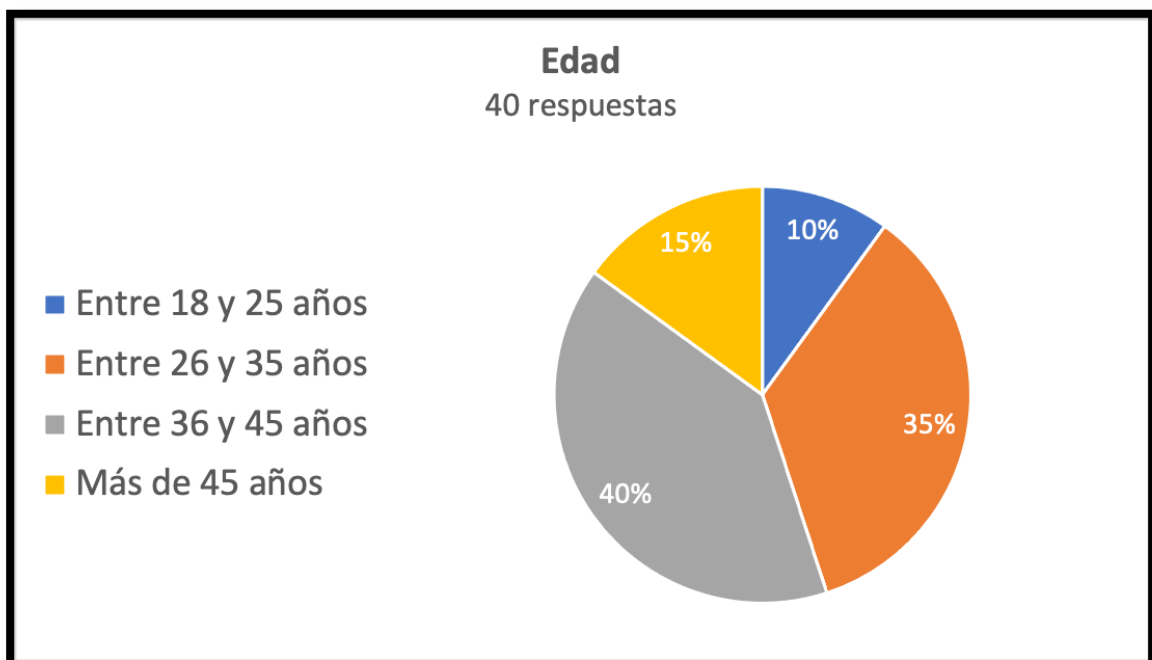


Figura 9. Encuesta - Pregunta 1

2.1.2. Pregunta 2

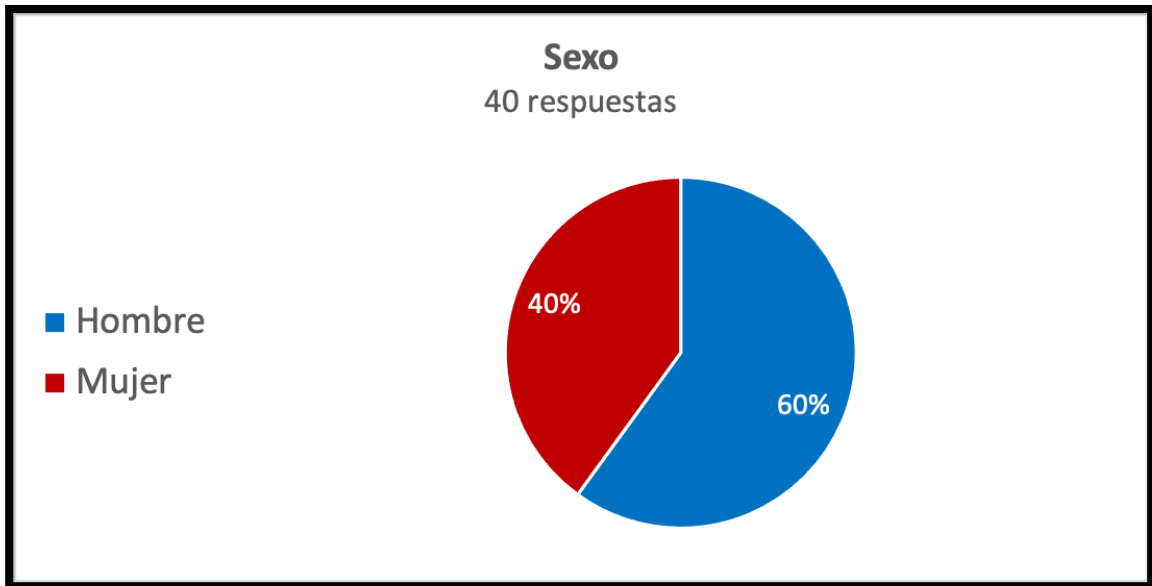


Figura 10. Encuesta - Pregunta 2

2.1.3. Pregunta 3

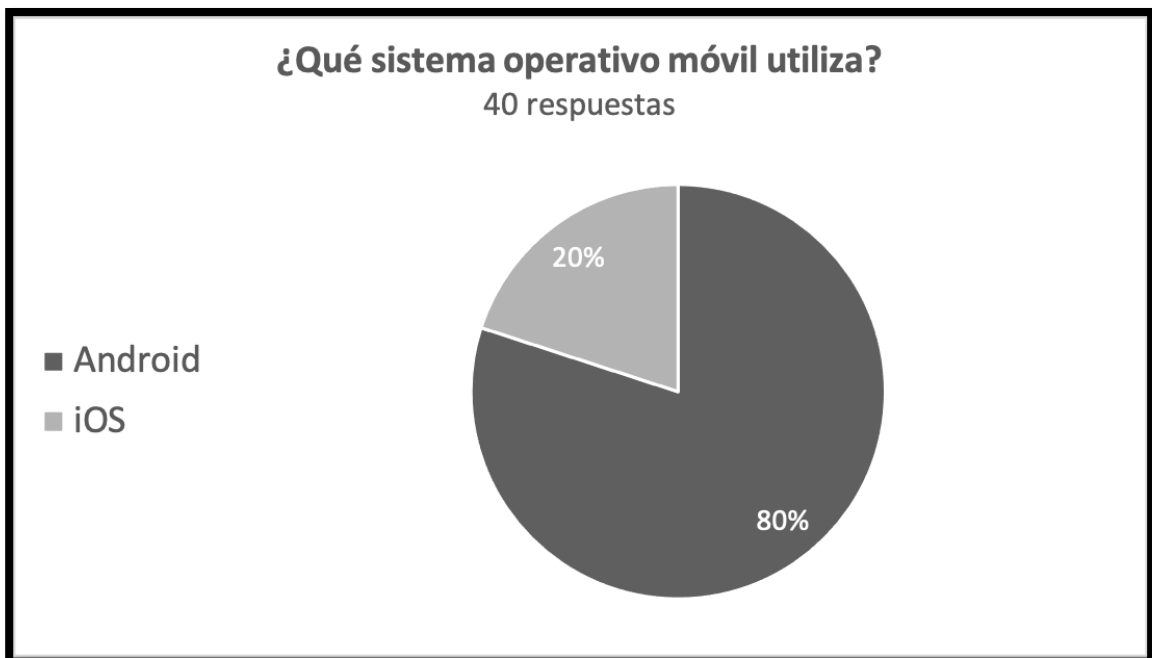


Figura 11. Encuesta - Pregunta 3

2.1.4. Pregunta 4

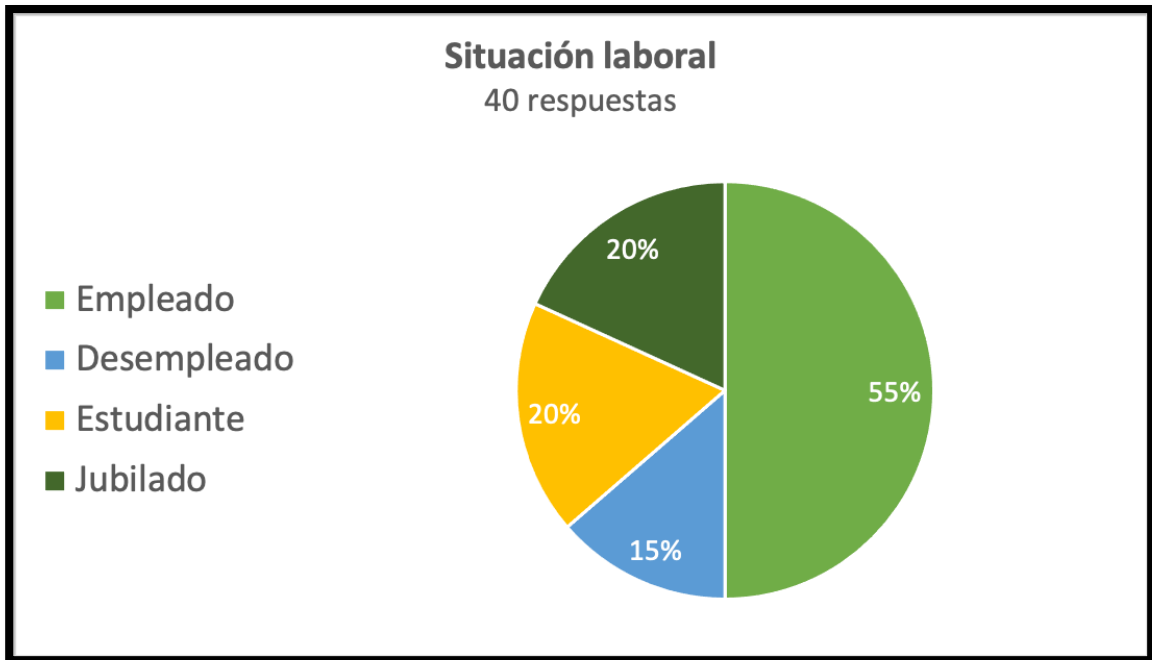


Figura 12. Encuesta - Pregunta 4

2.1.5. Pregunta 5

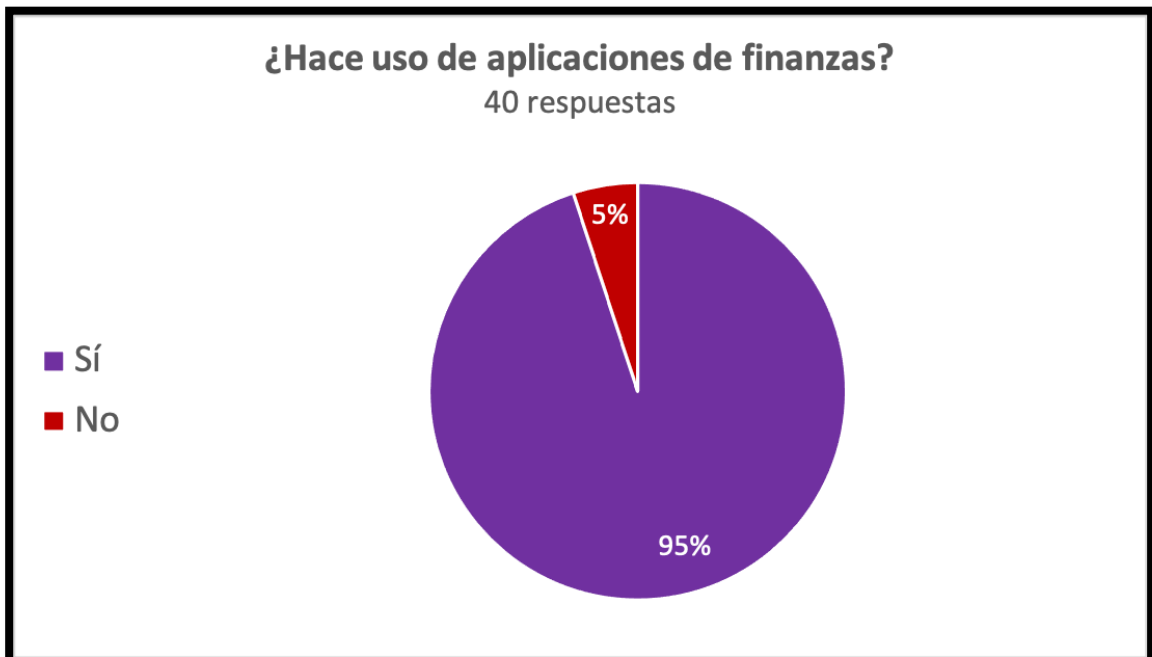


Figura 13. Encuesta - Pregunta 5

2.1.6. Pregunta 6

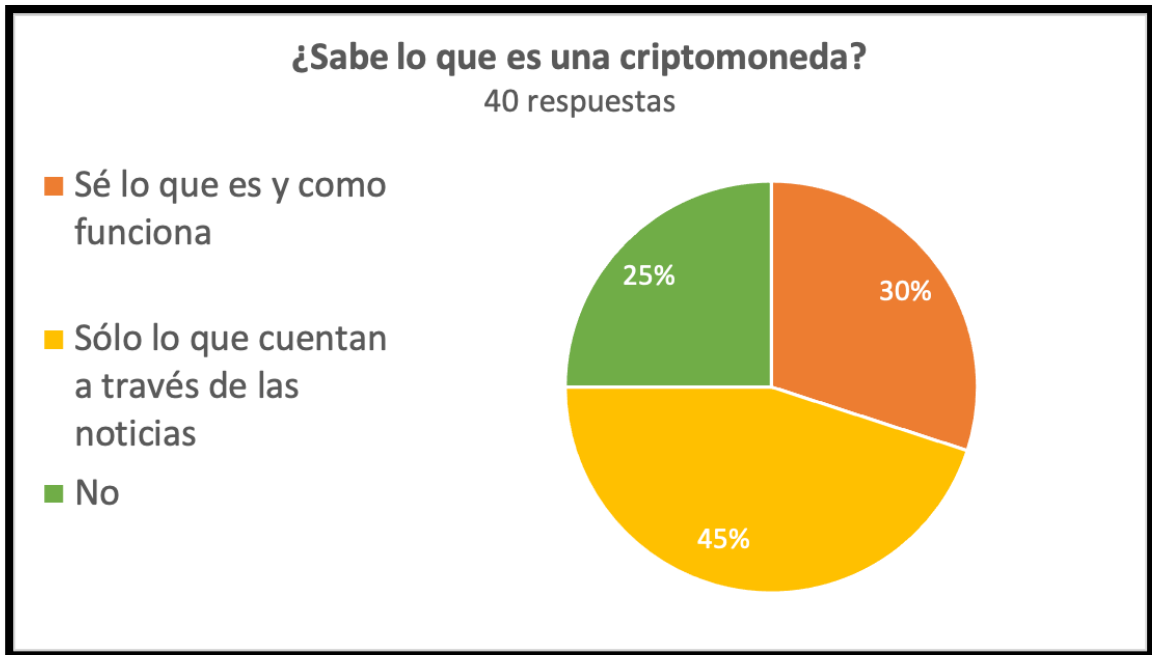


Figura 14. Encuesta - Pregunta 6

2.1.7. Pregunta 7

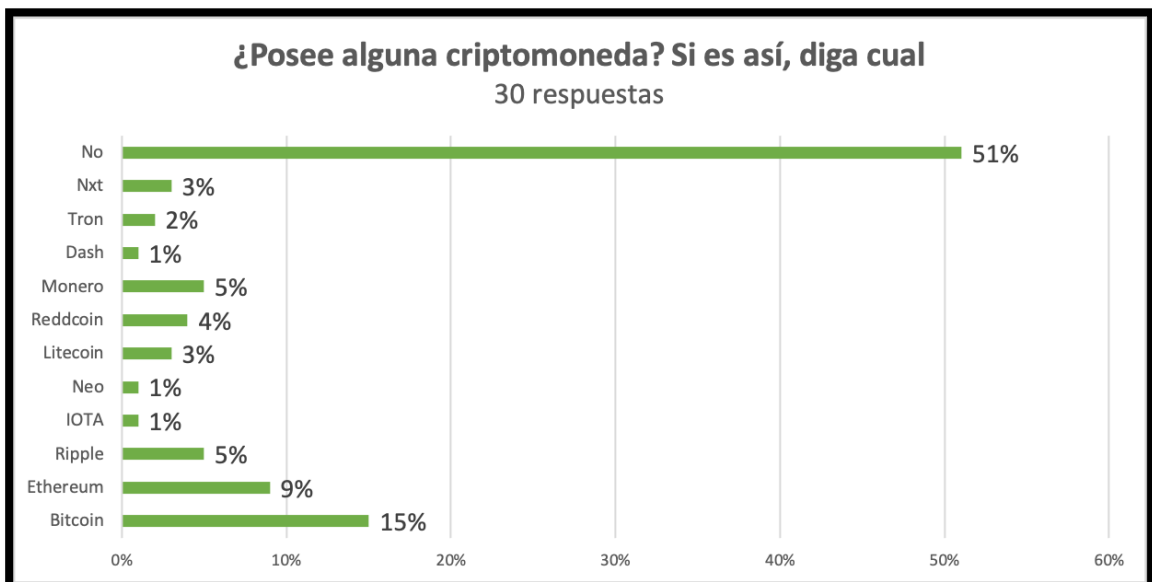


Figura 15. Encuesta - Pregunta 7

2.1.8. Pregunta 8

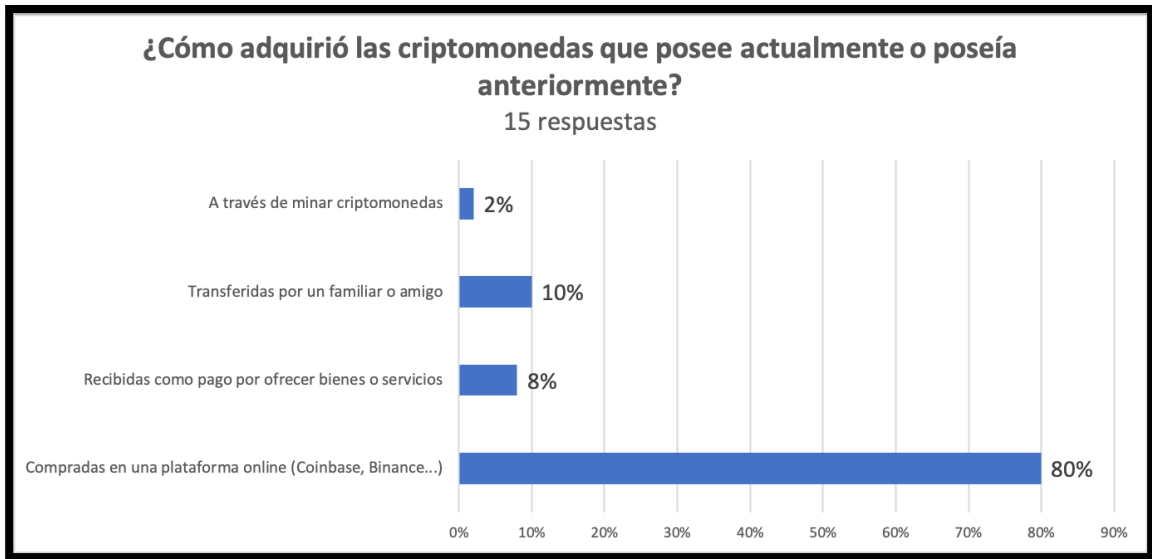


Figura 16. Encuesta - Pregunta 8

2.1.9. Pregunta 9

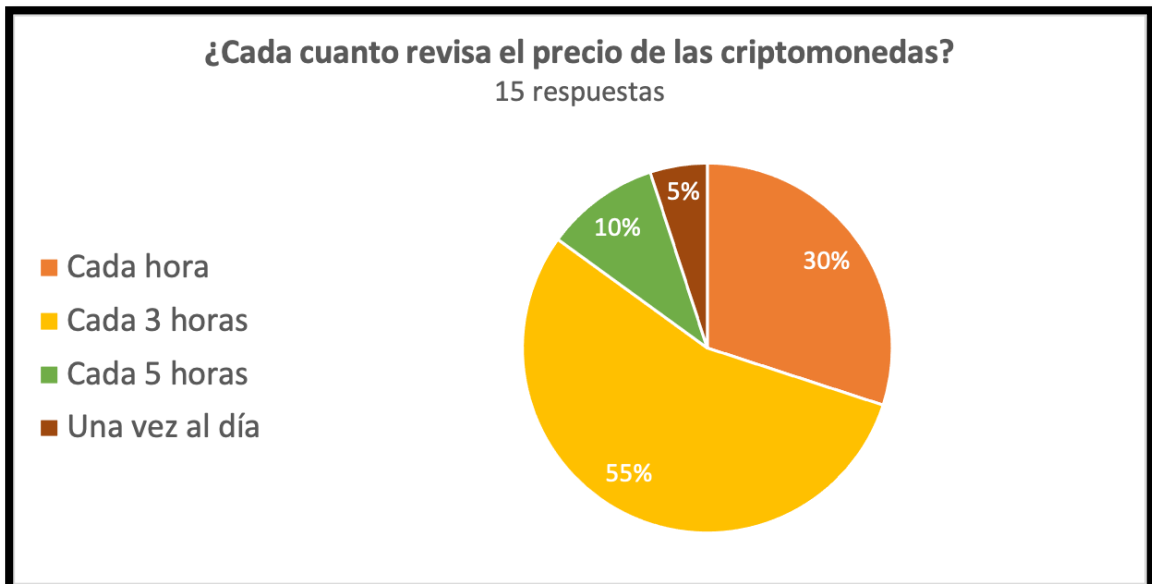


Figura 17. Encuesta - Pregunta 9

2.1.10. Pregunta 10

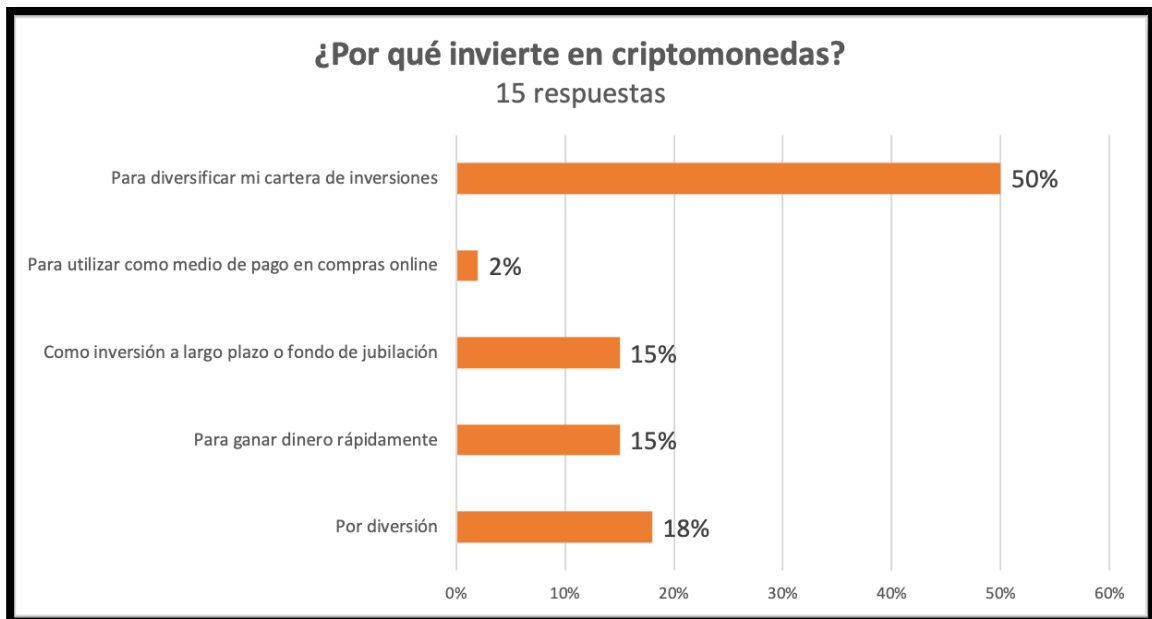


Figura 18. Encuesta - Pregunta 10

2.1.11. Conclusiones

Las conclusiones más relevantes que se pueden sacar tras el análisis de las respuestas obtenidas son las siguientes:

- La opción de desarrollar la aplicación para el sistema operativo Android ha sido acertada, ya que el 80% de las personas encuestadas utilizan un dispositivo con este sistema operativo.
- Solamente un 55% de los encuestados tenían conocimientos sobre criptomonedas. Sin embargo, solo un 30% sabían qué son y cómo funcionan. Junto al 51% de personas que no poseen ninguna criptomoneda de ese 55% que poseían conocimientos, entendemos que el nicho de mercado de la aplicación a desarrollar es bastante pequeño.
- El 50% de las personas que invierten criptomonedas lo hacen para diversificar su cartera de inversiones. Por lo tanto, se entiende que el 75% de los encuestados sean empleados o jubilados.
- El 80% de las personas que poseían criptomonedas las adquirieron a través de una aplicación. Un 85% de las personas revisan el precio de las criptomonedas cada 1 – 3 horas. Esto quiere decir que, pese a que el nicho de mercado es bastante pequeño, las personas que poseen criptomonedas suelen hacer bastante uso de las aplicaciones que gestionan o aportan información sobre las mismas.


2.2. Usuarios y contexto de uso

Tras los resultados y conclusiones obtenidos en la encuesta, se ha podido establecer dos perfiles de usuarios focales:

- Usuario que es empleado, perteneciente a un grupo de edad de entre 26 a 45 años. Dicho empleo le proporciona la capacidad monetaria para poder realizar inversiones en diferentes bienes que le permiten obtener ingresos extras que pueden ser empleados en diferentes objetivos.
- Usuario que es empleado / jubilado, perteneciente a un grupo de edad mayor de 45 años. Este tipo de usuario ya posee un patrimonio suficiente que le permite invertir en bienes para ganar dinero de forma más o menos rápida y poder tener ahorros suficientes para / durante la jubilación.

El principal contexto de uso de la aplicación, es el cuál en el que el usuario se encuentra revisando sus inversiones y necesita obtener información directa y simplificada acerca de alguna criptomoneda en la que haya invertido, revisando el precio de la misma para poder realizar cualquier acción en base a la información revisada. Dicha información puede ser consultada en cualquier lugar y en cualquier momento.

2.2.1. Personas y escenarios de uso

Persona	
	
<i>Figura 19. Imagen de User - Persona 1</i>	
Datos personales	
Nombre:	Judith Gómez
Edad:	38 años
Sexo:	Mujer

Ocupación:	<i>Product Owner</i>
Estado civil:	Casada y con dos hijos
Residencia:	Madrid

Biografía

Judith vive con su pareja y sus hijos en las afueras de Madrid y trabaja como *Product Owner* en una empresa mediana dedicada a la consultoría IT.

Tiene un horario de 8:00 a 15:00, ya que está realizando el horario de jornada reducida debido a que su último hijo nació hace dos años.

En su tiempo libre, le gusta dedicar tiempo a su familia, siendo su prioridad absoluta. También disfruta de sus amigos de toda la vida, ya sea viajando o yendo a cenar con ellos. Le apasiona el deporte, todos los días realiza una hora de ejercicio físico antes de ir a trabajar.

Antes de nacer su segundo hijo, decidió invertir en criptomonedas gracias a la recomendación de un compañero de trabajo. Tras realizar una búsqueda, a fondo, de la información que hay sobre este mundo, decidió invertir en las dos grandes criptomonedas conocidas, Bitcoin y Ethereum. Judith piensa en esta inversión como un fondo de ayuda para sus hijos, para que puedan tener un futuro mejor. Suele revisar la evolución de los precios una vez al día, así como la búsqueda de noticias relevantes.

Judith posee un móvil Android de alta gama utilizado, en gran medida, para una productividad mayor en su día a día.

Escenario

Es sábado por la mañana, toda su familia ya ha desayunado. Su marido se encuentra leyendo las noticias del día y sus hijos se encuentran jugando en el jardín, ya que hace buen tiempo.

Judith se encuentra sentada en su sillón revisando noticias relacionadas con sus criptomonedas, pero le resulta tedioso tener que estar buscando las noticias en diferentes páginas webs, así que decide buscar una aplicación que le permita agilizar todo este proceso.

Judith se encuentra en la *play store*, y da con la aplicación **Crypto Tracker**. Se la descarga. Una vez familiarizada con ella, le permite ser mucho más productiva en la tarea de buscar información sobre las criptomonedas en las que ha hecho una inversión, así como la evolución de su precio. Tras una semana de uso, Judith decide dar una puntuación máxima a la aplicación en la *play store*.

Tabla 4. Ficha User - Persona 1

Persona



Figura 20. Imagen de User - Persona 2

Nombre:	Paolo Ramírez
Edad:	50 años
Sexo:	Hombre
Ocupación:	Inversor
Estado civil:	Soltero
Residencia:	Palma de Mallorca

Biografía

Paolo vive solo, en un apartamento a pie de playa en Palma de Mallorca, junto a sus dos perros. Lleva siendo inversor toda su vida, aunque ahora se dedica más a mantener su patrimonio y solo invierte en aquellas cosas que pueden dar un gran beneficio.

No tiene un horario establecido, dependiendo del día y de la información que haya revisado sobre sus inversiones, dedicará más o menos tiempo. Por lo tanto, revisará la información necesaria a lo largo del día, pero en cortos periodos de tiempo.

En su tiempo libre, le gusta pasear con sus perros, y es un gran aficionado al vino. Para mantenerse en forma, suele realizar alguna actividad física a primera hora de la mañana, permitiéndole estar activo todo el día.

Debido al auge de las criptomonedas, Paolo ha visto que es una buena oportunidad invertir en ellas. Se ha hecho una cartera de diferentes criptomonedas, estudiando bien cuales tienen un valor seguro que le permitan obtener beneficios a medio – largo plazo; y cuales tienen un valor más oscilante, para poder obtener beneficios a corto plazo.

Paolo ha tenido móviles de alta gama con diferentes sistemas operativos, aunque ahora utiliza uno con un sistema operativo Android. Necesita que su móvil vaya fluido y sea rápido, porque el tiempo en el mundo de la inversión puede determinar ganar más o menos dinero.

Escenario

Es martes por la mañana, Paolo ya ha desayunado y se encuentra en su despacho leyendo periódicos relacionados con el mundo de las finanzas de diferentes países.

Posteriormente, decide abrir la aplicación en la cual tiene almacenada la información de las criptomonedas en las que tiene algo invertido, para ver su evolución. Lleva ya algún tiempo revisando este tipo de aplicaciones en la *play store*, y no hay ninguna que le resulte agradable e intuitiva, ya que muestran demasiada información, o la que muestra es algo confusa.

Por casualidad, revisando la *play store*, da con la aplicación **Crypto Tracker**, así que decide descargársela y darle una oportunidad. Tras estar trasteando con ella durante un rato, ve que es la aplicación que buscaba, ya que permite obtener la evolución de los precios de sus inversiones de forma clara e intuitiva, así como ver noticias relacionadas con sus inversiones. Además, puede establecer alarmas para saber cuando el precio de una criptomoneda ha variado en un rango establecido, cosa que agradece. Tras este tiempo de prueba, Paolo da la máxima puntuación a la aplicación en la *play store*, y ya la considera una de las aplicaciones que más va a usar a lo largo del día.

Tabla 5. Ficha User - Persona 2

2.3. Diseño conceptual

Para conceptualizar una aplicación, es necesario establecer cual será la estructura de la aplicación a través de un árbol de navegación; así como su flujo de interacción.

La elaboración de un árbol de navegación determina las pantallas necesarias de la aplicación que se pretende desarrollar, estableciendo la relación entre ellas. El flujo de interacción muestra la vía en la cual los usuarios pueden interactuar con la aplicación para obtener información o realizar una o varias tareas.

2.3.1. Árbol de navegación

Para la estructura de navegación, se ha decidido implementar la estructura jerárquica de dos niveles. Dicha estructura permite organizar la información en forma de árbol, y ayuda al usuario a encontrar rápidamente lo que busca y orientarse de manera fácil en la navegación.

Para la navegación entre las secciones principales, se ha decidido utilizar un menú hamburguesa. Este menú facilita la navegación y deja más espacio para el contenido.

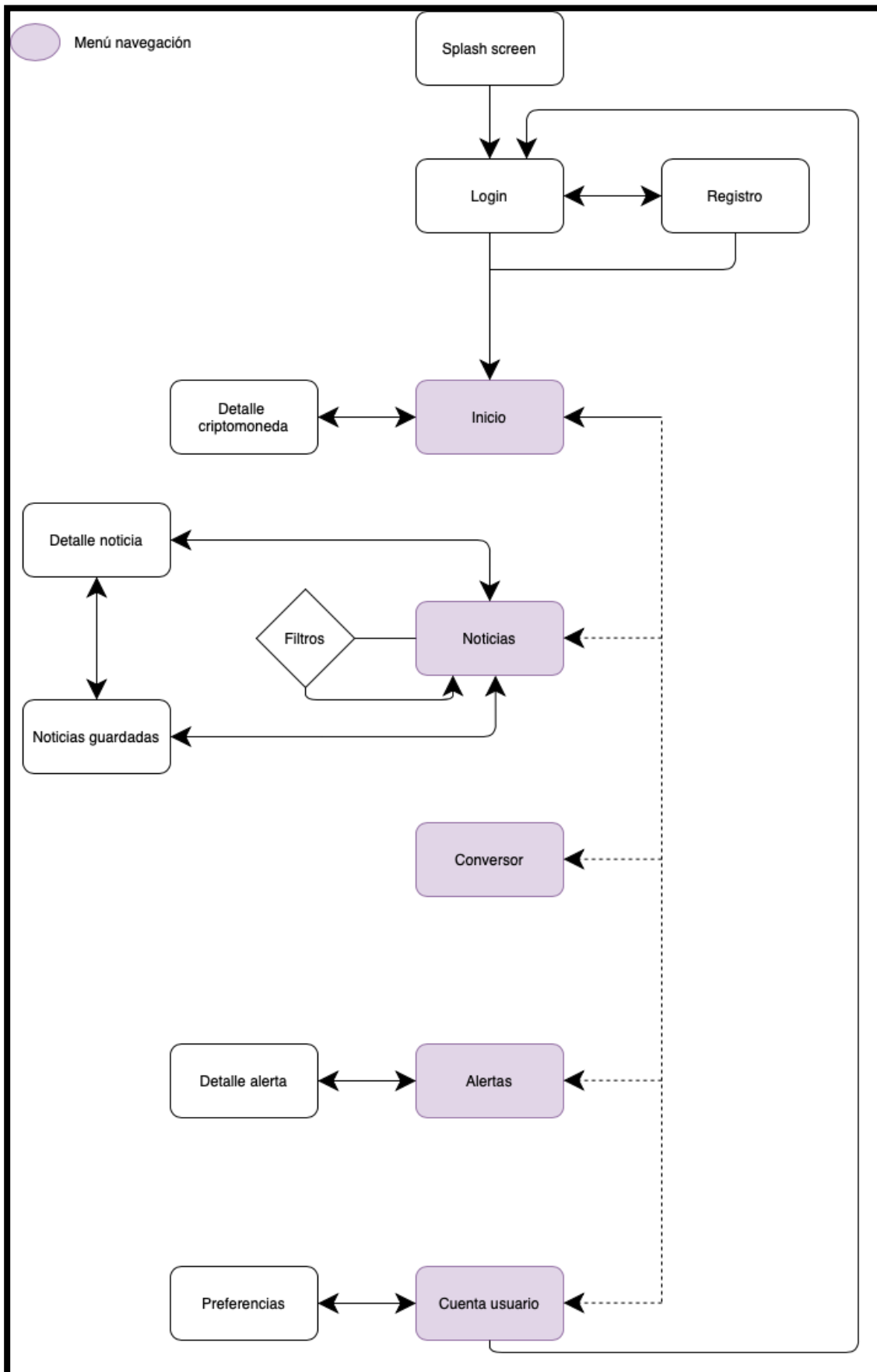


Figura 21. Árbol de navegación

2.3.2. Flujo de interacción

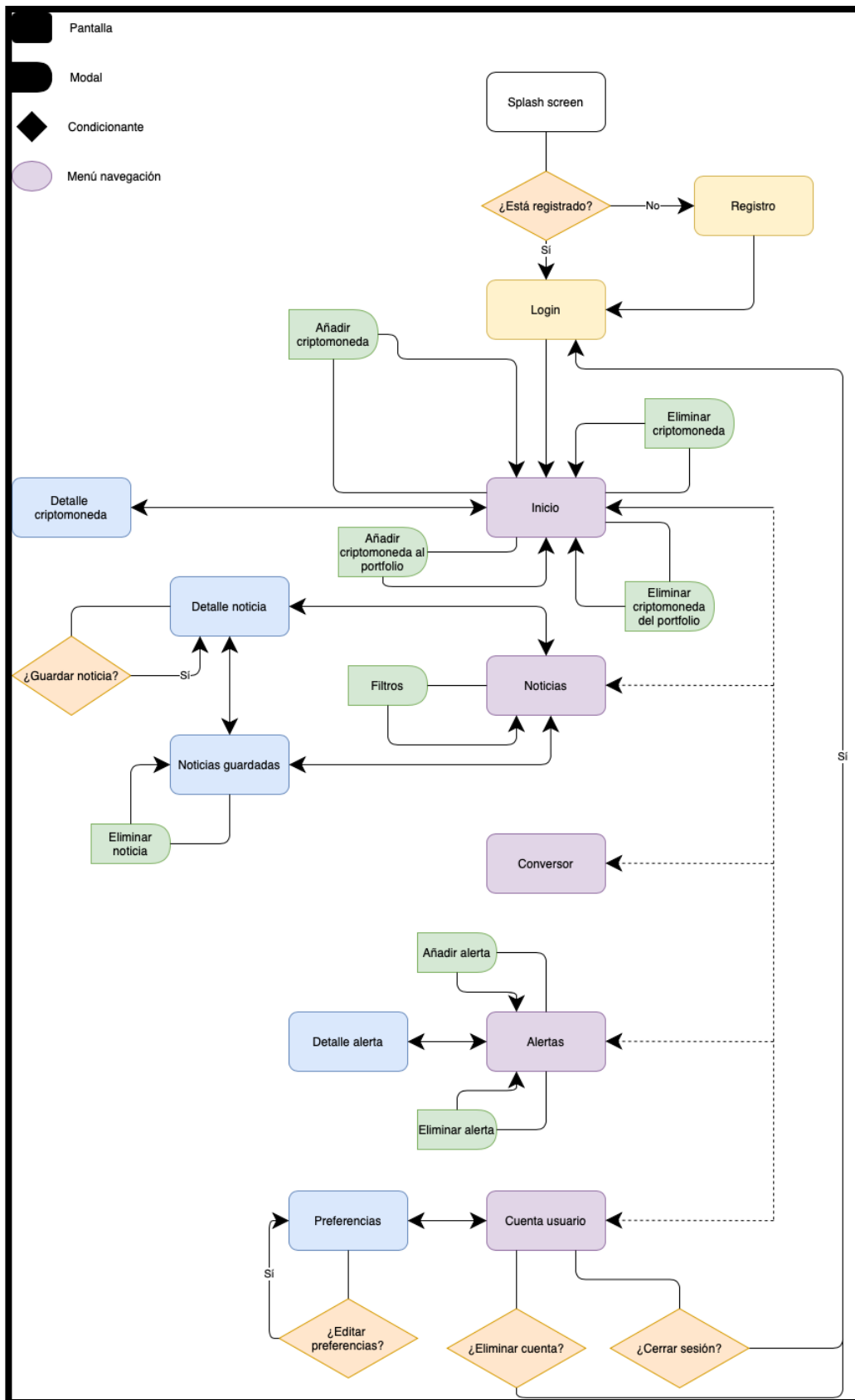


Figura 22. Flujo interacción

2.4. Prototipado

2.4.1. Sketches

El primer paso que se realiza para empezar a prototipar una aplicación, es la de realizar los bocetos de las pantallas a mano. Teniendo en cuenta el flujo de interacción mostrado en la [sección 2.3.2](#), se define la distribución de los elementos gráficos en cada una de las pantallas de la aplicación:

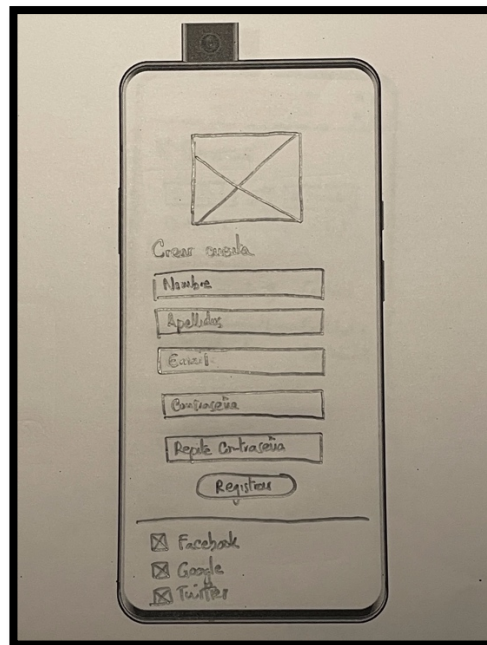


Figura 23. Pantalla Registro

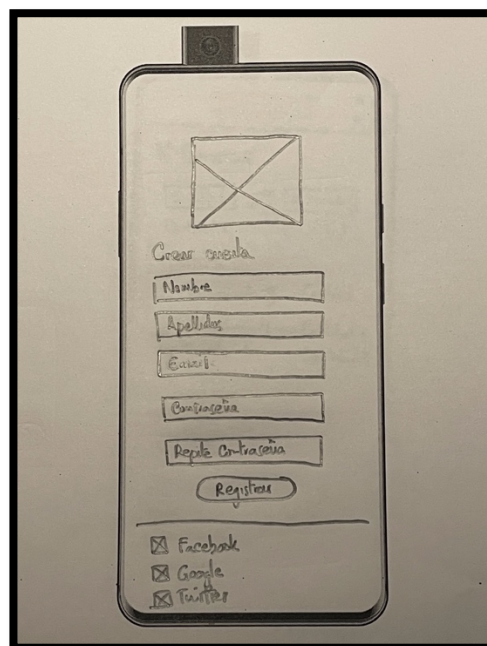


Figura 24. Pantalla Login

Tras la pantalla de *Splash*, las pantallas que se mostrarán son la de registro, o la de *login*. Esta última se mostrará si el usuario ya está registrado. En ambas pantallas, el usuario podrá identificarse a través de su email, sin embargo, se da la posibilidad de poder ingresar en la aplicación a través de sus redes sociales.

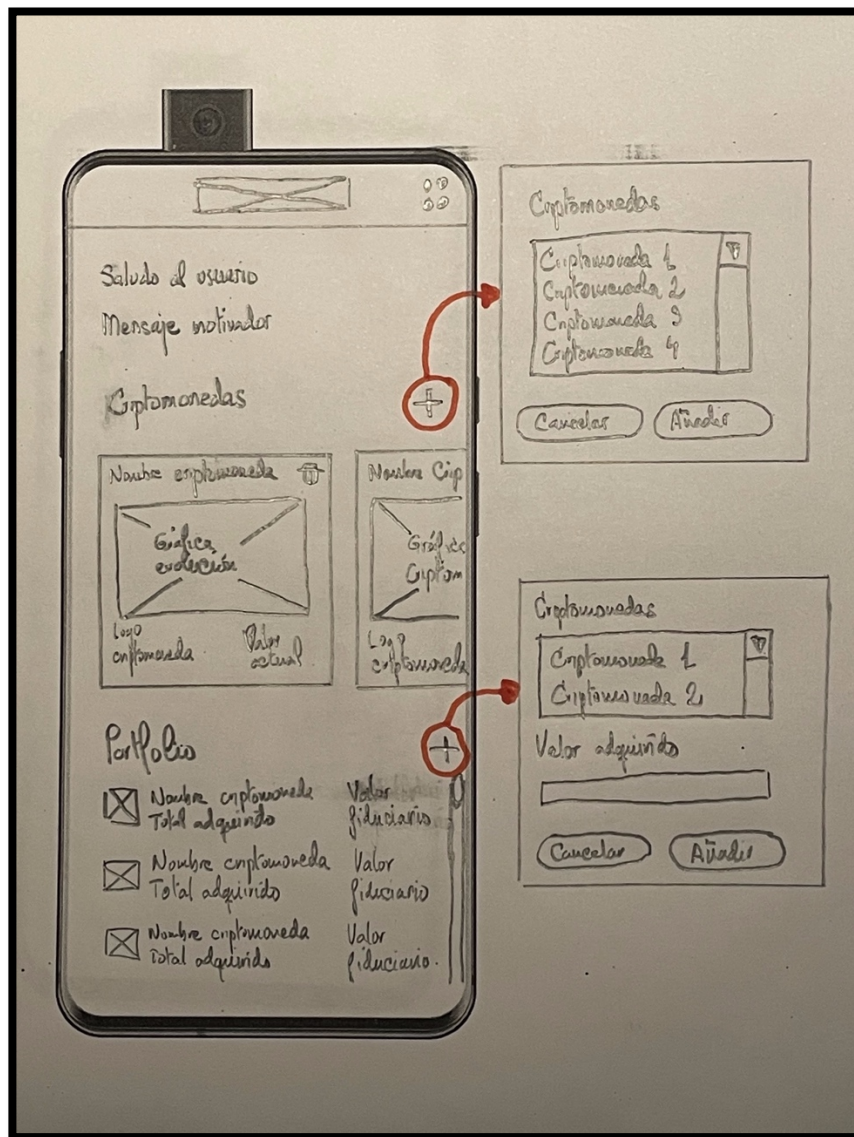


Figura 25. Pantalla Home

Cuando el usuario ingrese en la aplicación, verá la pantalla anterior pantalla. Esta pantalla estará compuesta por:

- Un saludo y mensaje motivador al usuario.
- Una sección de evolución de las criptomonedas que haya agregado para seguir, en modo tarjeta. Esta tarjeta se mostrará una gráfica, el valor actual y la posibilidad de poder eliminar. Esta sección mostrará las tarjetas en forma de vista carrusel.

- La última sección estará formada por aquellas criptomonedas que haya agregado el usuario, en las cuales, se ha hecho una inversión.

En las dos secciones principales, habrá un botón de añadir. Para que la aplicación sea intuitiva y haga que la experiencia de usuario sea agradable, al pulsar el botón de añadir aparecerá un *modal* que permitirá agregar una criptomoneda.

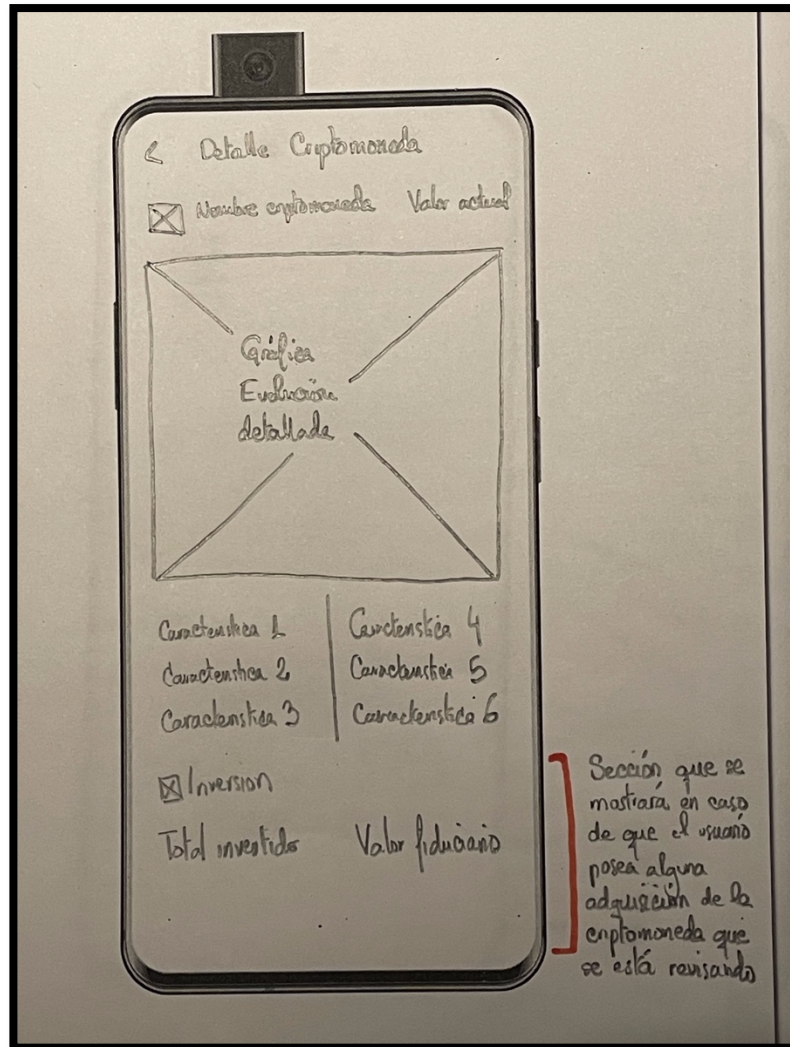


Figura 26. Pantalla detalle criptomoneda

Esta pantalla se mostrará cuando el usuario pulse sobre una tarjeta de la sección de evolución de criptomonedas. Esta pantalla mostrará una gráfica más detallada sobre la evolución de dicha criptomoneda, además de ver las características que tenga en el mercado en ese momento.

Si el usuario tiene esa criptomoneda en la sección de portfolio, es decir, ha realizado alguna inversión en ella, aparecerá una sección en esta pantalla llamada inversión, donde se vea la información del total invertido y el valor fiduciario. Si el usuario no tiene ninguna inversión en la criptomoneda que se está consultando, esta sección no se mostrará, es decir, es una sección dinámica.

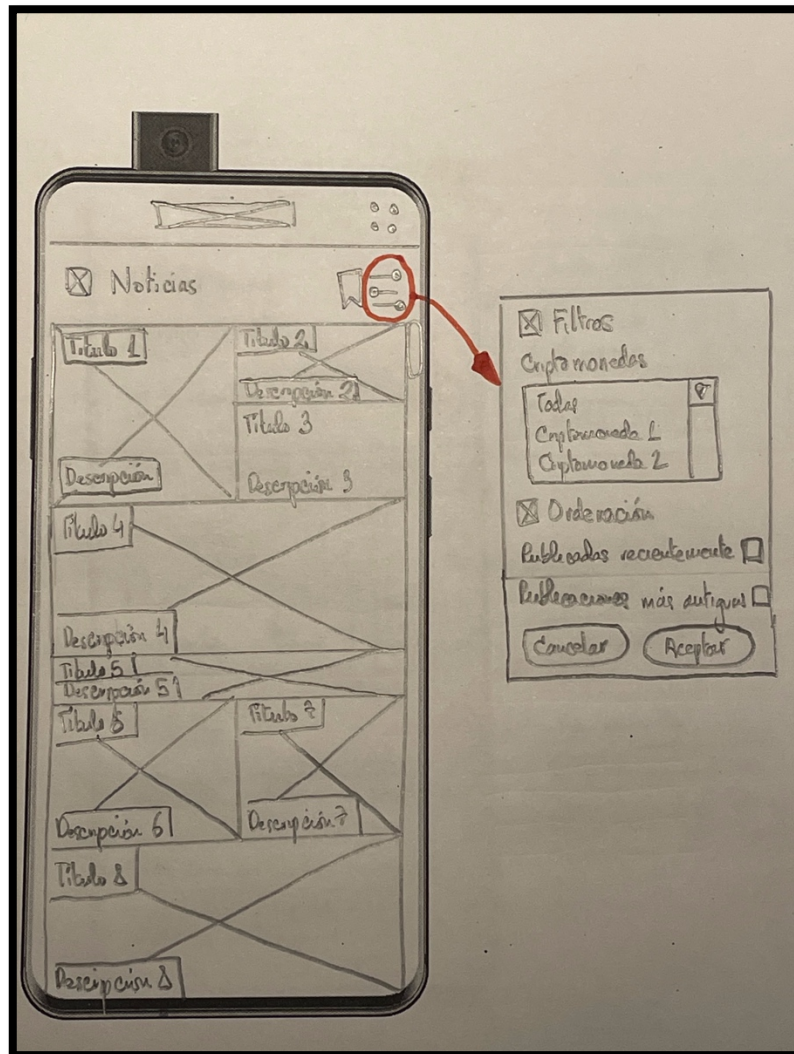


Figura 27. Pantalla noticias

Esta pantalla mostrará las noticias relacionadas con las criptomonedas que haya seleccionado el usuario, en el caso de que el usuario no tenga seleccionada ninguna, se mostrarán todas las noticias más relevantes con el mundo de las criptomonedas. Cada noticia de la lista de noticias será mostrada en tamaños diferentes.

Dentro de esta pantalla se tendrá dos botones. Un botón que permitirá acceder a la sección de noticias guardadas, y un botón que permitirá filtrar las noticias. Cuando el usuario pulse en el botón de filtros, aparecerá un modal. El modal permitirá filtrar por las criptomonedas que tenga el usuario, seleccionándolas de modo individual. Sin embargo, también se proporcionará la opción de mostrar noticias de todas las criptomonedas. Además, en este modal también se podrá ordenar las noticias en base a su fecha de publicación, mostrando primero las noticias que tengan una fecha de publicación más actual o más antigua.

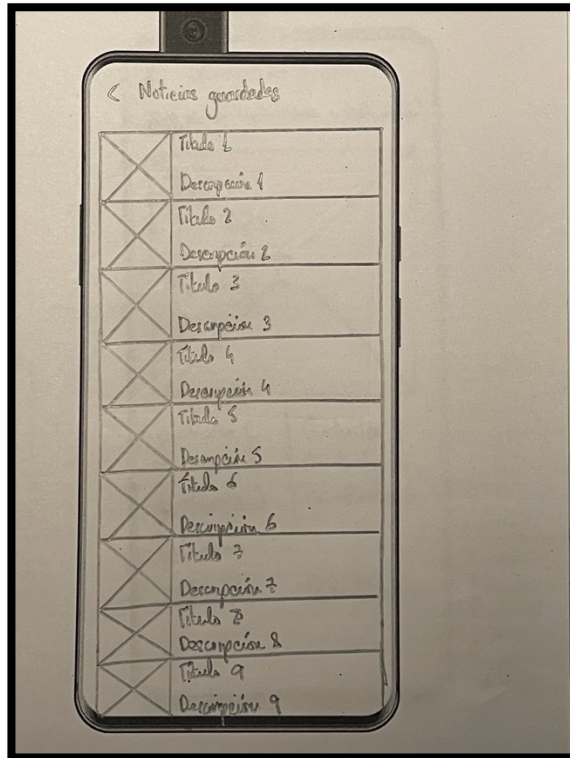


Figura 28. Pantalla noticias guardadas

Esta pantalla mostrará aquellas noticias que el usuario haya decidido guardar, ya sea porque le resultan interesantes o porque quiere guardarlas para leer más tarde. Las noticias se mostrarán en una vista de lista estándar.

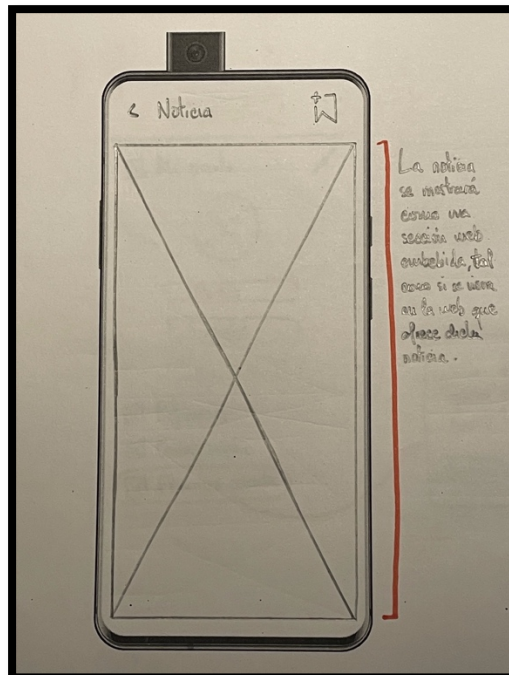


Figura 29. Pantalla detalle de noticia

La pantalla de detalle de una noticia tendrá un botón que permitirá guardar la noticia que se está revisando, para poder consultarla más tarde. La noticia se mostrará embebida, es decir, se utilizará un objeto de la librería proporcionada por el *framework* de Android que permitirá ver la noticia como si se estuviese viéndola en la web de origen.

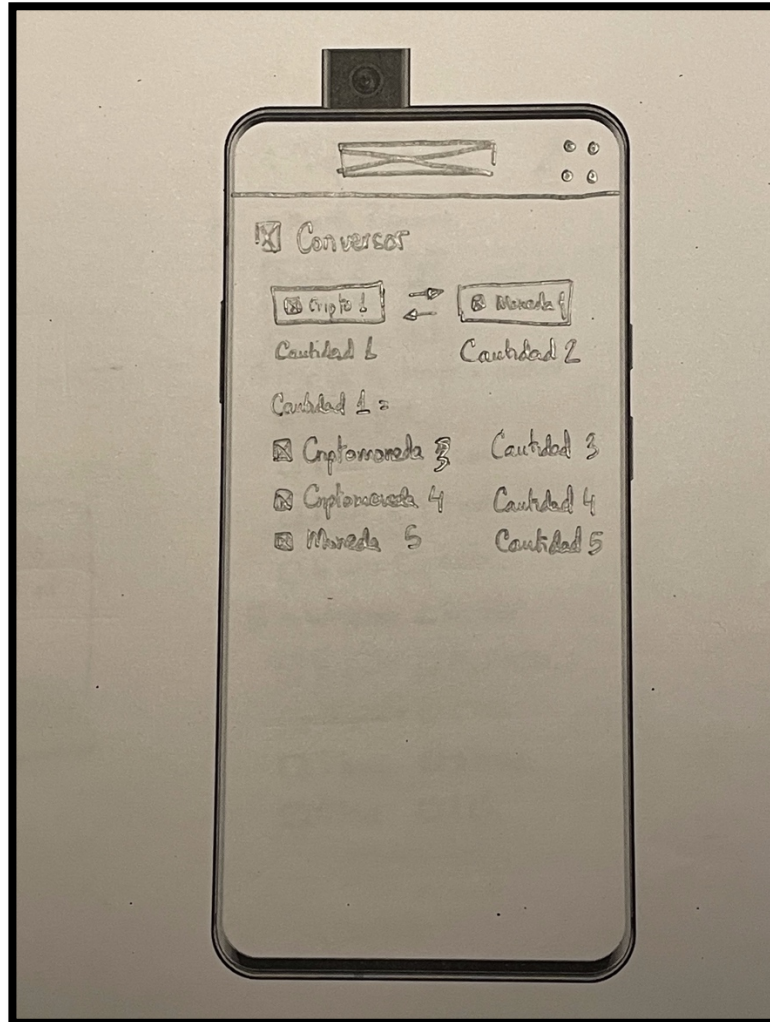


Figura 30. Pantalla de conversión de monedas

Esta pantalla mostrará la conversión entre una criptomoneda y una moneda fiduciaria, o a la inversa. La criptomoneda que se seleccionará por defecto es aleatoria, siempre basándose en las criptomonedas que haya añadido el usuario. La moneda fiduciaria que se seleccionará por defecto será la que el usuario haya seleccionado en el panel de preferencias. Podrá seleccionarse cualquier criptomoneda y moneda fiduciaria para hacer la conversión.

Como información adicional, podrá verse la conversión en las otras criptomonedas que el usuario haya agregado en la pantalla *home* de la aplicación.

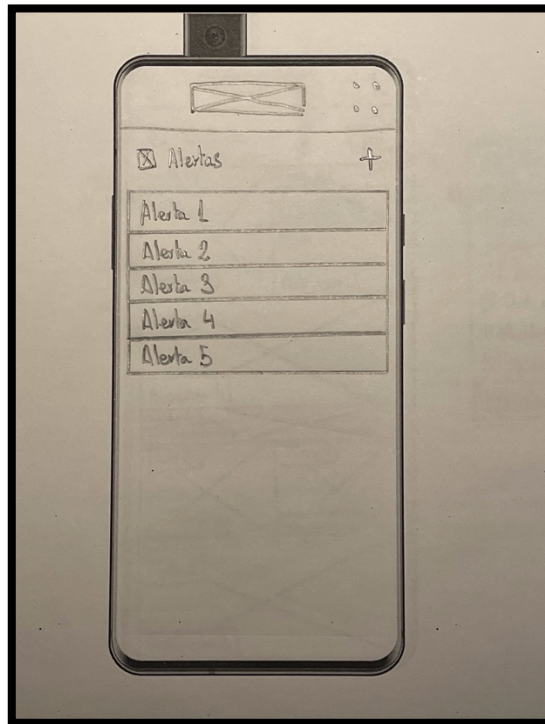


Figura 31. Pantalla alertas

La pantalla de alertas es una pantalla sencilla. En ella se encuentran, en formato lista, todas las alertas que se han creado. Además, existe un botón para la creación de alertas.

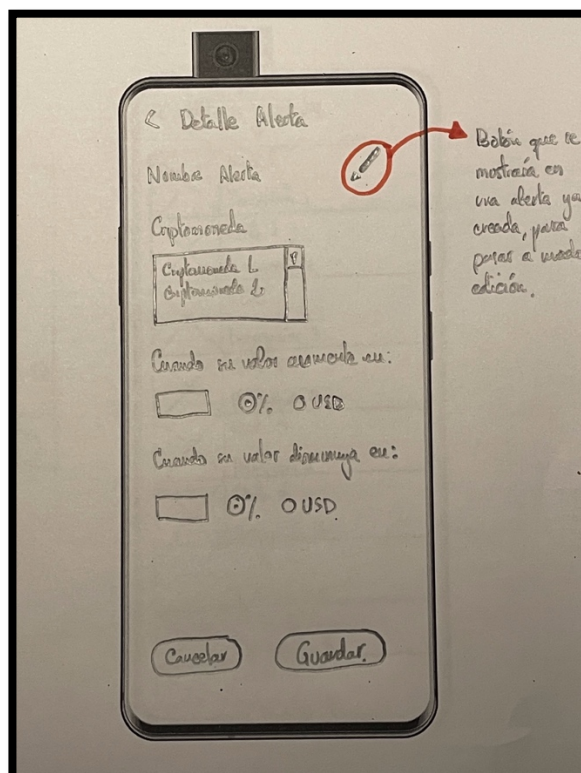


Figura 32. Pantalla detalle alerta

La pantalla del detalle de una alerta podrá revisarse de dos formas: una vista en forma de consulta, donde se mostrará la alerta sin poder modificar ninguna de sus características; y una vista en modo modificación. A ésta última podrá accederse cuando se cree una nueva, o cuando se pulse sobre el botón de modificación que vendrá implementado en la propia pantalla.

En ella, se podrá seleccionar sobre que criptomoneda se creará la alerta, seleccionando el rango superior o inferior del porcentaje o cantidad del valor asociado a la criptomoneda.

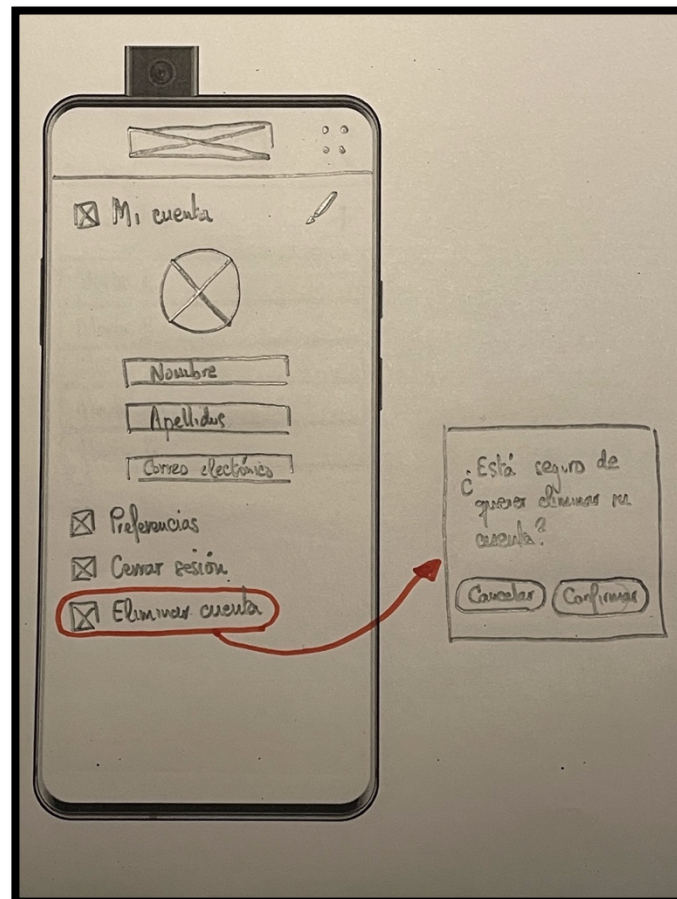


Figura 33. Pantalla cuenta del usuario

La pantalla de la cuenta de usuario mostrará la información que se ha proporcionado al registrarse. Esta información puede ser modificada cuando se pulse el botón de editar. Además, cuenta con tres botones:

- Preferencias: Abrió la pantalla de preferencias.
- Cerrar sesión: El usuario podrá cerrar sesión al pulsar este botón, volviendo a la pantalla de *login*.
- Eliminar cuenta: Cuando pulse este botón, el usuario verá un modal de confirmación de borrado de cuenta. Si el usuario confirma, se borrarán todos los datos almacenados del mismo, y se volverá a la pantalla de *login*.

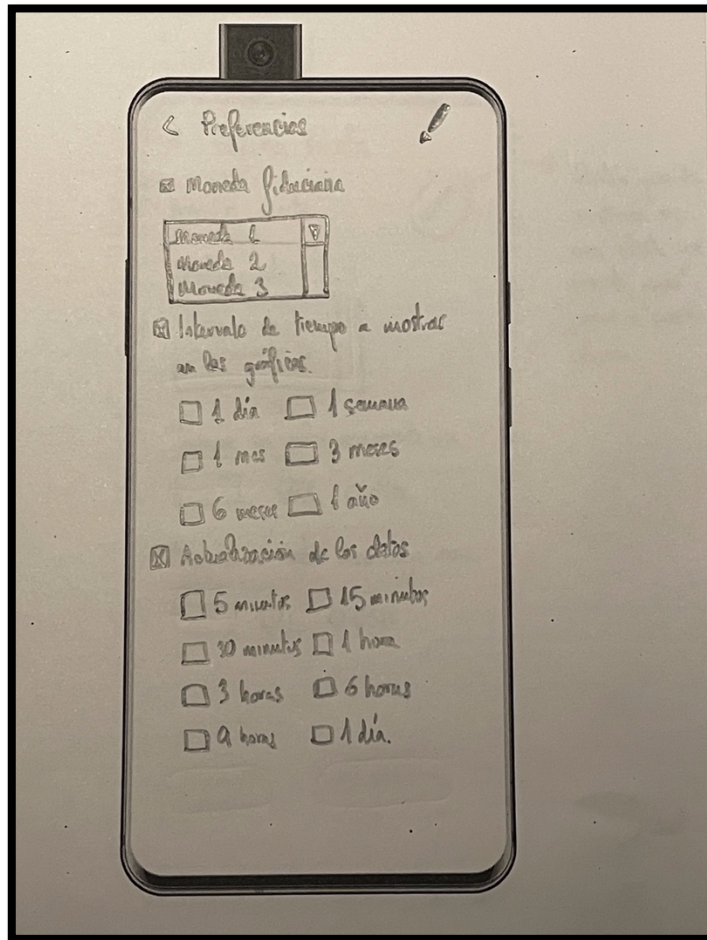


Figura 34. Pantalla preferencias

La pantalla preferencias tendrá dos opciones de visualización: una en modo consulta y otra en modo edición, tal como se ha realizado en las pantallas de “cuenta del usuario” y “detalle alerta”. En ella se podrá personalizar sobre la moneda fiduciaria que se mostrará, por defecto, a lo largo de la aplicación, el intervalo de tiempo a mostrar en los gráficos de las criptomonedas y el tiempo requerido para actualizar la información de las criptomonedas.

2.4.2. Prototipado de alta fidelidad

Un prototipado de alta fidelidad es una representación de alto detalle sobre la aplicación que se está desarrollando, similar al producto que se espera obtener, incluyendo la representación de la interfaz gráfica, el sistema de navegación y las dificultades del proyecto.

Para la elaboración del prototipado de alta fidelidad se ha decidido utilizar el programa **Adobe XD**.

A medida que se ha ido realizando el prototipado de alta fidelidad, ciertos diseños que se han definido en los [sketches](#) han sido modificados para que la interfaz de

usuario fuera más simple, accesible y minimalista con el fin de que la experiencia de usuario sea lo más agradable posible.

A continuación, se muestran algunos prototipos de alta fidelidad:

The registration screen features a dark blue background. At the top, it says "Bienvenido a CRYPTO TRACKER" in white and red. Below this is a light blue box with the text "¡Únete a nosotros!". Three social login buttons are stacked vertically: "Continuar con Google" (red), "Continuar con Facebook" (blue), and "Continuar con Twitter" (light blue). A horizontal separator line with a small circle in the middle follows. Below the separator are five input fields with labels: "Nombre", "Apellidos", "Email", "Contraseña", and "Repetir contraseña". At the bottom of the form is a light blue "Crear cuenta" button and a link that says "¿Ya eres miembro? [Inicia sesión](#)".

Figura 35. Pantalla registro de usuario

The login screen features a dark blue background. At the top, it says "Bienvenido a CRYPTO TRACKER" in white and red. Below this is a light blue box with the text "¡Hola de nuevo!". Three social login buttons are stacked vertically: "Continuar con Google" (red), "Continuar con Facebook" (blue), and "Continuar con Twitter" (light blue). A horizontal separator line with a small circle in the middle follows. Below the separator are three input fields with labels: "Email", "Contraseña", and "Iniciar sesión". At the bottom of the form is a light blue "Iniciar sesión" button and a link that says "¿Todavía no eres miembro? [Crear cuenta](#)".

Figura 36. Pantalla login



Figura 37. Vista menú

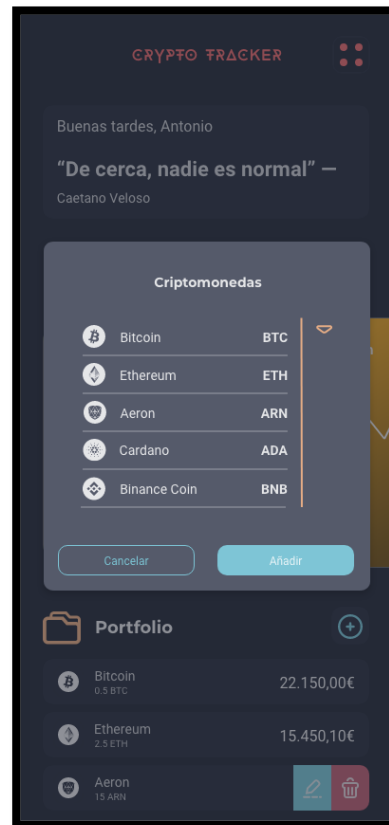


Figura 38. Vista añadir criptomoneda

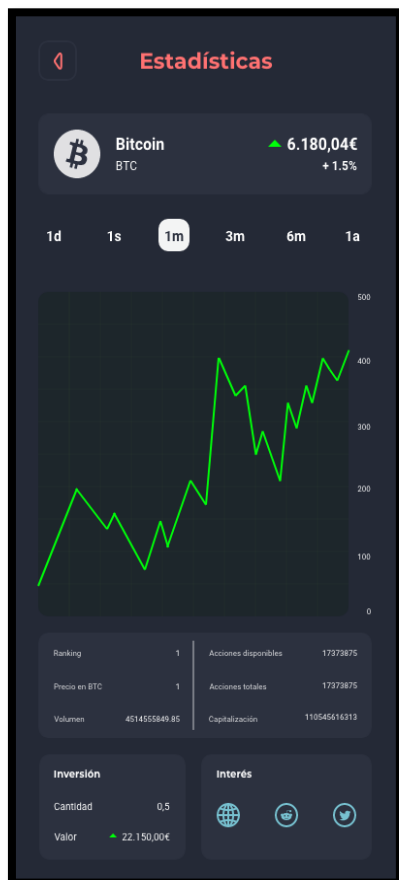


Figura 39. Pantalla detalle criptomoneda



Figura 40. Pantalla noticias

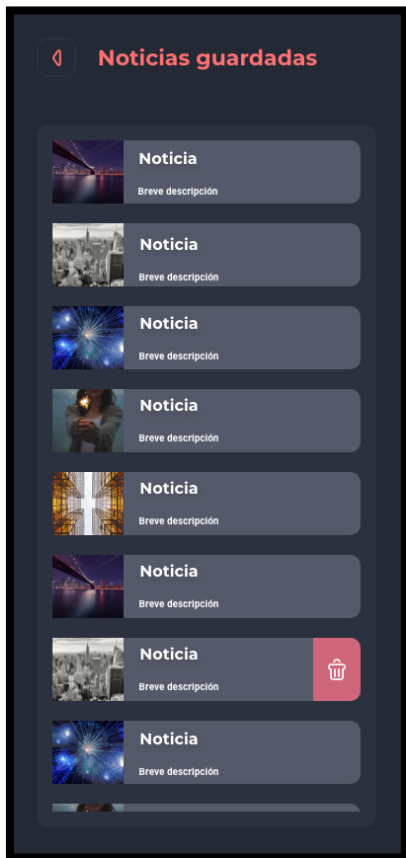


Figura 41. Pantalla noticias guardadas



Figura 42. Pantalla detalle noticia

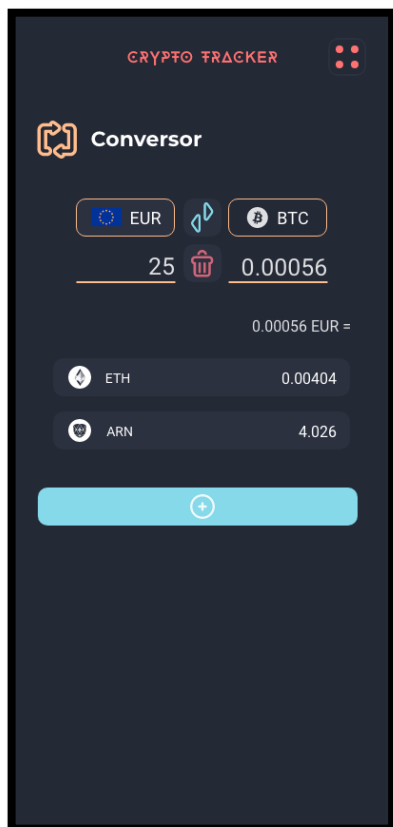


Figura 43. Pantalla convertor

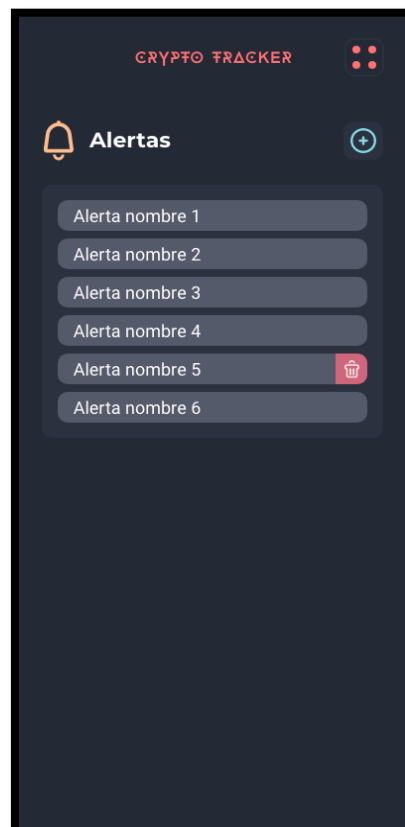


Figura 44. Pantalla alertas

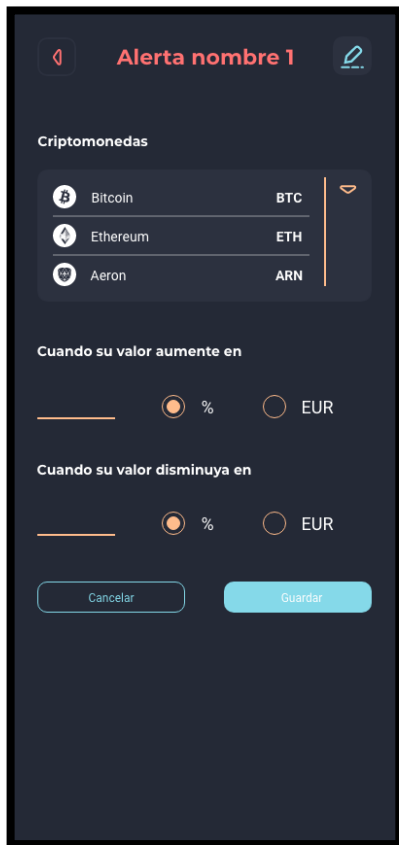


Figura 45. Pantalla detalle alerta

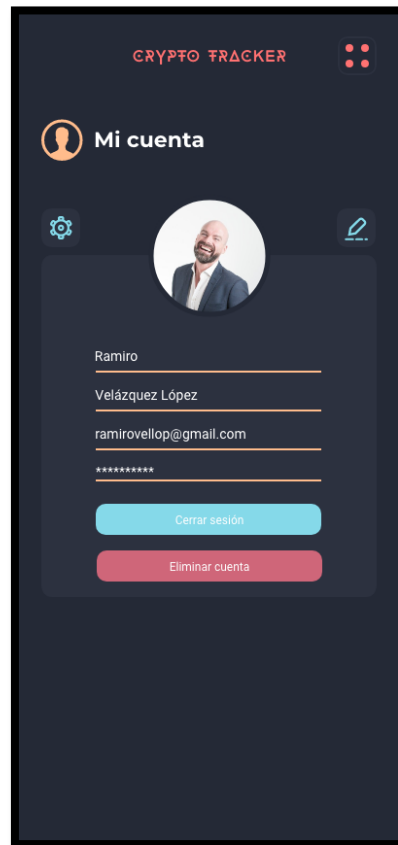


Figura 46. Pantalla usuario

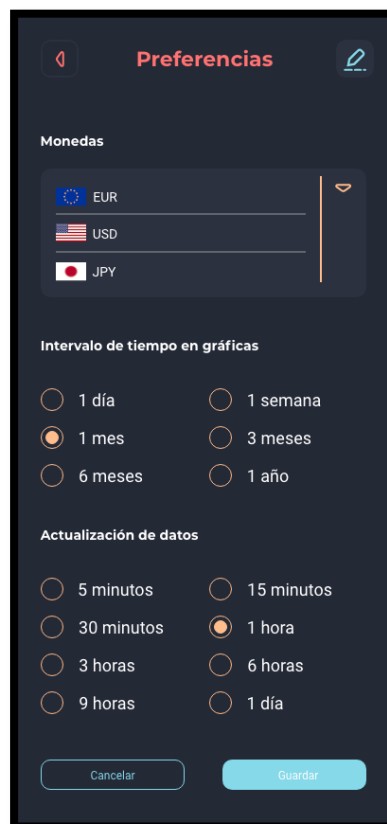


Figura 47. Pantalla preferencias

2.4.3. Evaluación

Una vez finalizado el prototipado en alta fidelidad, este fue evaluado por una serie de usuarios con el fin de mejorar la experiencia de usuario entre las pantallas diseñadas.

Estos usuarios detectaron ciertos factores, que se han tenido en cuenta, para mejorar la experiencia de usuario.

Una de las primeras recomendaciones que nos hicieron la mayoría de usuarios que han realizado las pruebas, es la poder acceder más cómodamente a la *pantalla de noticias* y a la *pantalla home*, es decir, que para acceder a ambas pantallas no haga falta acceder al menú de navegación.

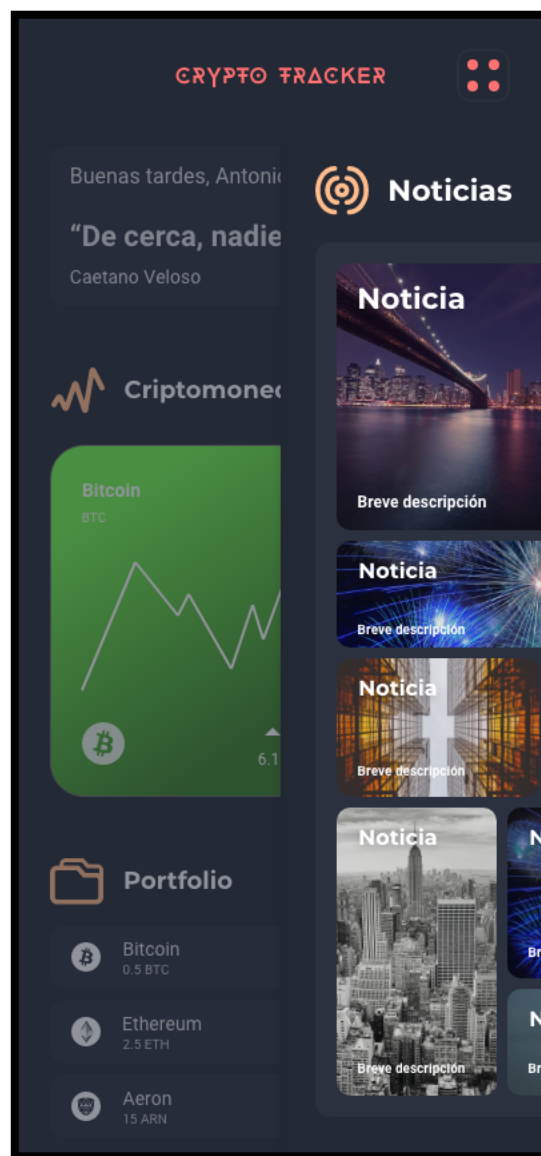


Figura 48. Primera implementación tras evaluación de usuarios

Para ello, se ha decidido implementar un *swipe* de derecha a izquierda (acceder a la pantalla de noticias) y de izquierda a derecha (acceder a la pantalla principal) entre estas dos pantallas, para que se puede acceder a ellas de la forma más dinámica posible, sin quitar la opción de acceder a cada una de ellas desde el menú de navegación.

Otro punto importante que han detectado algunos de los usuarios es la de poder actualizar los datos de las criptomonedas cuando ellos quieran. La opción de establecer un periodo de tiempo de actualización de datos mediante la pantalla de preferencias les ha parecido bien, sobre todo, para la configuración de alertas. Pero también quieren recibir los datos actualizados cuando ellos demanden.



Figura 49. Segunda implementación tras evaluación de usuarios

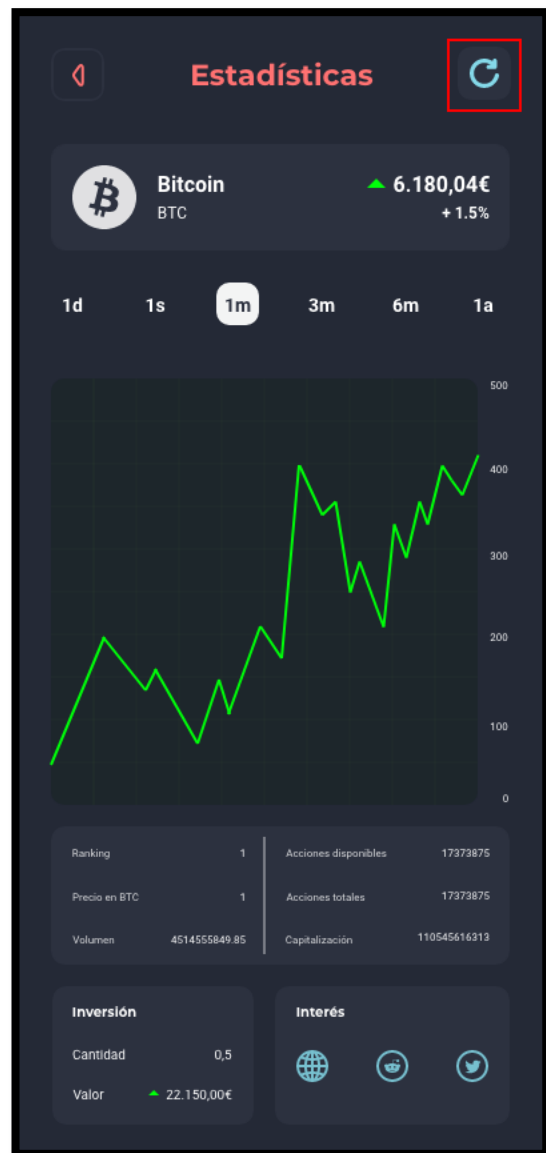


Figura 50. Segunda implementación tras evaluación de usuarios (2)

Se ha implementado un botón en la *pantalla de home* y otro botón en la *pantalla de detalle de una criptomoneda*.

En la *pantalla de home* el botón se encuentra en la vista de las gráficas de cada criptomoneda, y permite actualizar los datos de dicha criptomoneda en tiempo real. Cuando el botón se pulse, se informará al usuario que se están actualizando los datos y posteriormente aparecerán en pantalla refrescados. Esto tendrá un efecto secundario ya que, si tenemos esa criptomoneda en la sección de portfolio, los datos que se muestran también se verán afectados y se actualizarán con el valor correspondiente en ese momento.

En la *pantalla de detalle de una criptomoneda* el botón se ha implementado en la barra de navegación. Tendrá el mismo comportamiento que en la *pantalla de home*, pero la sección que también se verá actualizada, como efecto colateral, es la de inversión.

Un usuario detectó una funcionalidad que, tras estudiarla, sería bastante útil implementarla. El usuario comentó que le parecía un poco tedioso el tener que ir desde la *pantalla de detalle de una criptomoneda* a la *pantalla de alertas* y crear una alerta allí.

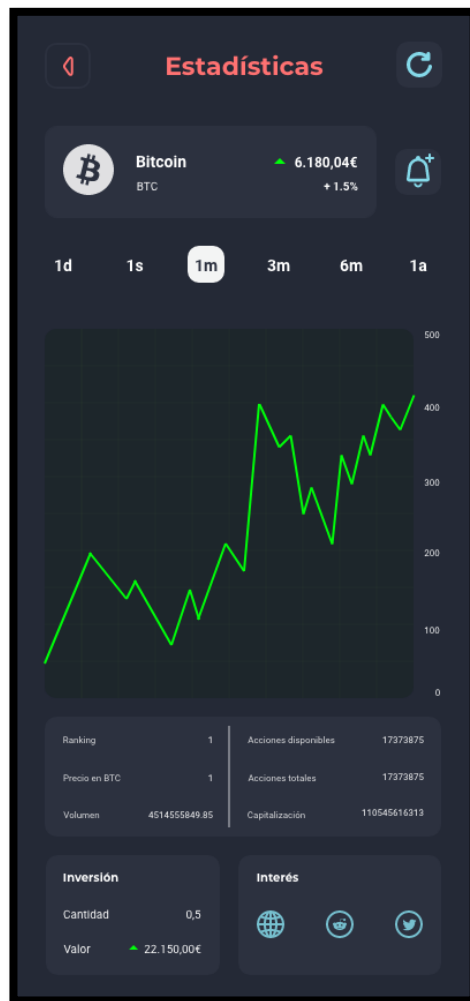


Figura 51. Tercera implementación tras evaluación de usuarios

Se ha añadido un botón para añadir alertas desde la *pantalla de detalle de una criptomoneda*. Cuando se pulse en este botón, se abrirá la *pantalla de detalle de una alerta*, y se podrá configurar la alerta que quiera el usuario. Una vez creada, se volverá a la *pantalla de detalle de una criptomoneda*.

Como se ha podido observar, algunos diseños han sido modificados desde su planteamiento en los [sketches](#) hasta su [prototipado de alta fidelidad](#). Esto puede seguir ocurriendo a medida que vayamos avanzando durante la implementación a nivel de código.

2.5. Diseño técnico

2.5.1. Definición de los casos de uso

CU – 01: Registro de usuario	
Descripción	Registro de un usuario en la aplicación.
Prioridad	Media
Actores	Usuario no registrado.
Prerrequisitos	El usuario debe poseer al menos: <ul style="list-style-type: none"> ▪ Un correo electrónico válido. ▪ Una cuenta de Google. ▪ Una cuenta de Facebook. ▪ Una cuenta de Twitter.
Flujo	<ul style="list-style-type: none"> ▪ Usuario abre la aplicación. ▪ Si es la primera vez que entra, se le abre la <i>pantalla de registro</i>. ▪ Al usuario se le da como método de registro principal el uso de sus redes sociales. Si no, rellena el formulario de registro. ▪ La aplicación comprueba que el correo electrónico no exista en la base de datos. ▪ La aplicación dirige al usuario a la <i>pantalla de home</i>.
Requisitos a posteriori	El usuario se ha registrado satisfactoriamente y ha iniciado sesión.
Comentarios	Si el usuario introduce una contraseña que no concuerda con los requisitos mínimos, la aplicación muestra un <i>modal</i> informando del error y la forma correcta de crear una contraseña.

Tabla 6. CU – 01: Registro de usuario

CU – 02: Login de usuario	
Descripción	Inicio de sesión de un usuario en la aplicación.
Prioridad	Media
Actores	Usuario registrado no identificado.

Prerrequisitos	El usuario debe estar registrado en la aplicación (CU – 01).
Flujo	<ul style="list-style-type: none"> ▪ Usuario abre la aplicación. ▪ Si las credenciales de <i>login</i> han caducado o nunca ha iniciado sesión, la aplicación muestra la <i>pantalla de login</i> y el usuario debe iniciar sesión mediante la autenticación a través de sus redes sociales, o rellenar sus credenciales de acceso. ▪ Si el usuario ha rellenado sus credenciales, la aplicación comprueba que son correctas. ▪ La aplicación dirige al usuario a la <i>pantalla de home</i>.
Requisitos a posteriori	El usuario ha iniciado sesión correctamente.
Comentarios	Si el usuario no ha introducido sus credenciales correctamente, la aplicación mostrará un <i>modal</i> informando del error.

Tabla 7. CU – 02: Login de usuario

CU – 03: Eliminar cuenta de usuario	
Descripción	Dar de baja una cuenta de usuario en la aplicación.
Prioridad	Baja
Actores	Usuario identificado.
Prerrequisitos	El usuario debe estar registrado en la aplicación (CU – 01) e identificado (CU – 02).
Flujo	<ul style="list-style-type: none"> ▪ El usuario se encuentra en la <i>pantalla cuenta del usuario</i>. ▪ El usuario pulsa en el botón de <i>eliminar cuenta</i>. ▪ La aplicación muestra un <i>modal</i> de confirmación. ▪ El usuario confirma la eliminación de la cuenta. ▪ La aplicación elimina al usuario de cualquier sistema de almacenamiento que se esté usando. ▪ La aplicación redirige al usuario a la <i>pantalla de login</i>.
Requisitos a posteriori	El usuario ya no está identificado y no se encuentra ningún dato en el sistema de almacenamiento relacionado con él.
Comentarios	

Tabla 8. CU – 03: Eliminar cuenta de usuario

CU – 04: Logout de usuario	
Descripción	Cierre de sesión por parte de un usuario en la aplicación.
Prioridad	Baja
Actores	Usuario identificado.
Prerrequisitos	El usuario debe estar registrado en la aplicación (CU – 01) e identificado (CU – 02).
Flujo	<ul style="list-style-type: none"> ▪ El usuario se encuentra en la <i>pantalla cuenta del usuario</i>.

	<ul style="list-style-type: none"> ▪ El usuario pulsa en el botón de <i>cerrar sesión</i>. ▪ La aplicación cierra la sesión del usuario identificado. ▪ La aplicación redirige al usuario a la <i>pantalla de login</i>.
Requisitos a posteriori	El usuario ya no está identificado en la aplicación.
Comentarios	

Tabla 9. CU – 04: Logout de usuario

CU – 05: Acceso a la pantalla de home	
Descripción	Acceder a la <i>pantalla de home</i> en la que se muestra un mensaje de bienvenida, las criptomonedas que le interesan al usuario y un portfolio con lo que ha invertido el usuario.
Prioridad	Alta
Actores	Usuario registrado.
Prerrequisitos	El usuario debe estar registrado en la aplicación (CU – 01).
Flujo	<ul style="list-style-type: none"> • El usuario abre la aplicación. • El usuario se identifica en la <i>pantalla de login</i> (CU – 02). • La aplicación redirige al usuario a la <i>pantalla de home</i>. <p>Flujo alternativo 1:</p> <ul style="list-style-type: none"> • El usuario abre la aplicación. • Si el usuario ha entrado previamente, se habrá almacenado sus credenciales, por lo tanto, no deberá identificarse en la <i>pantalla de login</i>. • La aplicación redirige al usuario a la <i>pantalla de home</i>. <p>Flujo alternativo 2:</p> <ul style="list-style-type: none"> • El usuario se encuentra en cualquier pantalla de la aplicación donde se encuentre el botón del menú de navegación. • El usuario pulsa el botón de <i>menú de navegación</i>. • El usuario pulsa el botón de <i>home</i>. • La aplicación redirige al usuario a la <i>pantalla de home</i>. <p>Flujo alternativo 3:</p> <ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de noticias</i> (CU – 13). • El usuario realiza un <i>swipe</i> de izquierda a derecha. • La aplicación muestra la <i>pantalla de home</i>.
Requisitos a posteriori	El usuario se encuentra en la <i>pantalla de home</i> y puede ver las diferentes secciones que conforman esta pantalla.

Comentarios	Si el usuario es la primera vez que entra en la aplicación, las secciones de <i>criptomonedas</i> y <i>portfolio</i> se encontrarán vacías, mostrando un mensaje que le invite a añadir información.
--------------------	--

Tabla 10. CU – 05: Acceso a la pantalla de home

CU – 06: Añadir criptomoneda	
Descripción	Permite al usuario añadir una criptomoneda a seguir.
Prioridad	Alta
Actores	Usuario identificado.
Prerrequisitos	El usuario debe estar registrado en la aplicación (CU – 01) e identificado (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de home</i>. • El usuario pulsa en el botón de <i>añadir una criptomoneda</i> de la sección de <i>criptomonedas</i>. • La aplicación muestra un <i>modal</i> con una lista multiseleccionable de criptomonedas. • El usuario selecciona una o varias criptomonedas. • El usuario pulsa el botón de <i>añadir</i>. • La aplicación almacena las criptomonedas añadidas. • El modal se cierra. • La aplicación recarga la sección de <i>criptomonedas</i> de la <i>pantalla de home</i>.
Requisitos a posteriori	La aplicación cuenta con una o varias criptomonedas nuevas.
Comentarios	

Tabla 11. CU – 06: Añadir criptomoneda

CU – 07: Eliminar criptomoneda	
Descripción	Permite al usuario eliminar una criptomoneda que estaba siguiendo.
Prioridad	Baja
Actores	Usuario identificado.
Prerrequisitos	El usuario debe estar: <ul style="list-style-type: none"> • Registrado en la aplicación (CU – 01). • Identificado en la aplicación (CU – 02). • Haber añadido previamente la criptomoneda que desea eliminar (CU – 06).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de home</i>. • El usuario pulsa en el icono de <i>eliminar</i> en una de las vistas en modo tarjeta de una criptomoneda de la sección de <i>criptomonedas</i>. • La aplicación elimina la información almacenada sobre esa moneda. • La aplicación refresca la sección <i>criptomonedas</i> de la <i>pantalla de home</i>.

Requisitos a posteriori	La aplicación cuenta con una criptomoneda menos.
Comentarios	

Tabla 12. CU – 07: Eliminar criptomoneda

CU – 08: Actualizar criptomoneda	
Descripción	Permite al usuario refrescar información sobre una criptomoneda.
Prioridad	Baja
Actores	Usuario identificado.
Prerrequisitos	El usuario debe estar: <ul style="list-style-type: none"> • Registrado en la aplicación (CU – 01). • Identificado en la aplicación (CU – 02). • Haber añadido previamente la criptomoneda que desea eliminar (CU – 06).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de home</i>. • El usuario pulsa en el icono de <i>actualizar</i> en una de las vistas en modo tarjeta de una criptomoneda de la sección de <i>criptomonedas</i>. • La aplicación refresca la tarjeta con la información actualizada de la criptomoneda.
Requisitos a posteriori	La información de la criptomoneda se ha actualizado.
Comentarios	Si el usuario posee la criptomoneda que ha actualizado en la sección de <i>portfolio</i> de la <i>pantalla de home</i> , la información que se encuentra en esta sección también se verá actualizada.

Tabla 13. CU – 08: Actualizar criptomoneda

CU – 09: Añadir criptomoneda al portfolio	
Descripción	Permite al usuario añadir una criptomoneda en la que haya realizado alguna inversión.
Prioridad	Alta
Actores	Usuario identificado.
Prerrequisitos	El usuario debe estar: <ul style="list-style-type: none"> • Registrado en la aplicación (CU – 01). • Identificado en la aplicación (CU – 02). • Haber añadido previamente una criptomoneda (CU – 06).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de home</i>. • El usuario pulsa en el botón de <i>añadir</i> una criptomoneda de la sección <i>portfolio</i>. • La aplicación muestra un modal con una lista seleccionable de aquellas criptomonedas que haya añadido el usuario en la sección <i>criptomonedas</i> y un <i>EditText</i> para añadir la cantidad invertida en dicha criptomoneda. • El usuario selecciona una criptomoneda e introduce el valor invertido.

	<ul style="list-style-type: none"> • El usuario pulsa el botón de <i>añadir</i>. • La aplicación almacena la información añadida. • El modal se cierra. • La aplicación recarga la sección de <i>portfolio</i>.
Requisitos a posteriori	La aplicación cuenta con una criptomoneda en la que se ha realizado una inversión.
Comentarios	

Tabla 14. CU – 09: Añadir criptomoneda al portfolio

CU – 10: Modificar criptomoneda del portfolio	
Descripción	Permite al usuario modificar la inversión realizada en una criptomoneda.
Prioridad	Baja
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02). • Haber añadido previamente una criptomoneda a la sección <i>criptomonedas</i> de la <i>pantalla de home</i> (CU – 06). • Haber añadido previamente una criptomoneda a la sección <i>portfolio</i> de la <i>pantalla de home</i> (CU – 09).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de home</i>. • El usuario realiza un <i>swipe</i> de derecha a izquierda en una de las criptomonedas añadidas en la sección de <i>portfolio</i>. • El usuario pulsa en el botón de <i>modificar</i>. • La aplicación muestra un <i>modal</i> con la criptomoneda seleccionada y un <i>EditText</i> para modificar el valor introducido anteriormente. • El usuario modifica el valor. • El usuario pulsa el botón de <i>guardar</i>. • La aplicación almacena la información modificada. • El modal se cierra. • La aplicación recarga la sección de <i>portfolio</i>.
Requisitos a posteriori	La aplicación cuenta con la información actualizada de la inversión realizada en una criptomoneda.
Comentarios	

Tabla 15. CU – 10: Modificar criptomoneda del portfolio

CU – 11: Eliminar criptomoneda del portfolio	
Descripción	Permite al usuario borrar la inversión realizada en una criptomoneda.
Prioridad	Baja
Actores	Usuario identificado.

Prerrequisitos	<p>El usuario debe:</p> <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02). • Haber añadido previamente una criptomoneda a la sección <i>criptomonedas</i> de la <i>pantalla de home</i> (CU – 06). • Haber añadido previamente una criptomoneda a la sección <i>portfolio</i> de la <i>pantalla de home</i> (CU – 09).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de home</i>. • El usuario realiza un <i>swipe</i> de derecha a izquierda en una de las criptomonedas añadidas en la sección de <i>portfolio</i>. • El usuario pulsa en el botón de <i>eliminar</i>. • La aplicación elimina la información relacionada sobre la criptomoneda en la sección de <i>portfolio</i>. • La aplicación recarga la sección de <i>portfolio</i>.
Requisitos a posteriori	La aplicación ya no cuenta con la inversión realizada en una criptomoneda.
Comentarios	

Tabla 16. CU – 11: Eliminar criptomoneda del portfolio

CU – 12: Acceder a la pantalla de detalle de una criptomoneda	
Descripción	Permite al usuario revisar la información detallada de una criptomoneda.
Prioridad	Alta
Actores	Usuario identificado.
Prerrequisitos	<p>El usuario debe:</p> <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02). • Haber añadido previamente una criptomoneda a la sección <i>criptomonedas</i> de la <i>pantalla de home</i> (CU – 06).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de home</i>. • El usuario pulsa en una de las vistas en modo tarjeta de una criptomoneda de la sección de <i>criptomonedas</i>. • La aplicación redirige al usuario a la <i>pantalla de detalle de una criptomoneda</i>.
Requisitos a posteriori	Se muestra información detallada de aquella criptomoneda que haya seleccionado el usuario.
Comentarios	La sección <i>inversión</i> se mostrará si el usuario ha añadido una criptomoneda en la sección <i>portfolio</i> de la <i>pantalla de home</i> (CU – 09).

Tabla 17. CU – 12: Acceder a la pantalla de detalle de una criptomoneda

CU – 13: Actualizar la información en la pantalla de detalle de una criptomoneda	
Descripción	Permite al usuario actualizar información detallada sobre una criptomoneda.
Prioridad	Baja
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02). • Haber añadido previamente una criptomoneda a la sección <i>criptomonedas</i> de la <i>pantalla de home</i> (CU – 06).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de detalle de una criptomoneda</i>. • El usuario pulsa el botón de <i>actualizar</i>. • La aplicación actualiza la información de la criptomoneda. • La aplicación muestra la información detallada actualizada.
Requisitos a posteriori	La información detallada de la criptomoneda se ha actualizado.
Comentarios	

Tabla 18. CU – 13: Actualizar la información en la pantalla de detalle de una criptomoneda

CU – 14: Modificar el intervalo de tiempo en la gráfica mostrada en la pantalla de detalle de una criptomoneda	
Descripción	Permite al usuario modificar el intervalo de tiempo de la evolución del precio que se ve en la gráfica de la <i>pantalla de detalle de una criptomoneda</i> .
Prioridad	Media
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02). • Haber añadido previamente una criptomoneda a la sección <i>criptomonedas</i> de la <i>pantalla de home</i> (CU – 06).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de detalle de una criptomoneda</i>. • El usuario selecciona uno de los botones disponibles para seleccionar el intervalo de tiempo. • La aplicación recalcula la gráfica en base al botón que haya seleccionado el usuario. • La aplicación muestra la gráfica recalculada.
Requisitos a posteriori	La gráfica que muestra la evolución del precio de una criptomoneda se verá actualizada según las decisiones del usuario.

Comentarios	<p>El intervalo de tiempo disponible es:</p> <ul style="list-style-type: none"> • 1d (1 día) • 1s (1 semana) • 1m (1 mes) • 3m (3 meses) • 6m (6 meses) • 1ª (1 año)
--------------------	--

Tabla 19. CU – 14: Modificar el intervalo de tiempo en la gráfica mostrada en la pantalla de detalle de una criptomoneda

CU – 15: Acceder a los enlaces de interés de una criptomoneda	
Descripción	Permite al usuario acceder de forma externa a los enlaces de interés relacionados con una criptomoneda.
Prioridad	Baja
Actores	Usuario identificado.
Prerrequisitos	<p>El usuario debe:</p> <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02). • Haber añadido previamente una criptomoneda a la sección <i>criptomonedas</i> de la <i>pantalla de home</i> (CU – 06).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de detalle de una criptomoneda</i>. • El usuario selecciona uno de los enlaces de interés de una criptomoneda. • La aplicación redirige al usuario al enlace que haya seleccionado.
Requisitos a posteriori	El usuario verá la información de forma externa. Si el usuario posee la aplicación instalada del enlace de interés que haya seleccionado, la aplicación se abrirá; si no, se abrirá como un enlace web en el navegador que tenga el usuario preconfigurado.
Comentarios	Los enlaces de interés disponibles son: web, reddit y twitter de la criptomoneda.

Tabla 20. CU – 15: Acceder a los enlaces de interés de una criptomoneda

CU – 16: Acceder a la pantalla de noticias	
Descripción	Permite al usuario revisar las noticias relacionadas con las criptomonedas que ha añadido.
Prioridad	Alta
Actores	Usuario identificado.
Prerrequisitos	<p>El usuario debe:</p> <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en cualquier pantalla de la aplicación donde se encuentre el botón del menú de navegación. • El usuario pulsa el botón del <i>menú de navegación</i>.

	<ul style="list-style-type: none"> • El usuario pulsa el botón de <i>noticias</i>. • La aplicación redirige al usuario a la <i>pantalla de noticias</i>. • La aplicación muestra las noticias en forma de mosaico. <p>Flujo alternativo 1:</p> <ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de home</i>. • El usuario realiza un <i>swipe</i> de derecha a izquierda. • La aplicación muestra la <i>pantalla de noticias</i>.
Requisitos a posteriori	Se muestra la <i>pantalla de noticias</i> relacionadas con el mundo de las criptomonedas. Todas las noticias que se encuentran en la lista pueden pulsarse para acceder al detalle de la noticia.
Comentarios	Si el usuario no ha añadido ninguna criptomoneda a la sección de <i>criptomonedas</i> de la <i>pantalla de home</i> , se mostrarán noticias relacionadas con todas las criptomonedas que se encuentra en el mercado. En caso contrario, solamente se mostrarán aquellas noticias relacionadas con las criptomonedas que haya añadido el usuario.

Tabla 21. CU – 16: Acceder a la pantalla de noticia

CU – 17: Filtrar noticias	
Descripción	Permite al usuario filtrar noticias en la pantalla de noticias.
Prioridad	Media
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de noticias</i>. • El usuario pulsa el botón de <i>filtros</i>. • La aplicación abre un <i>modal</i> mostrando una lista de las criptomonedas que el usuario ha añadido en la sección <i>criptomonedas</i> de la <i>pantalla de home</i> (CU – 06), y dos opciones por las que ordenar. • El usuario hace su selección en el <i>modal</i>. • La aplicación obtiene las noticias aplicando los filtros que ha establecido el usuario. • La aplicación muestra las noticias filtradas.
Requisitos a posteriori	Se muestra la <i>pantalla de noticias</i> relacionadas con el mundo de las criptomonedas filtradas. Todas las noticias que se encuentran en la lista pueden pulsarse para acceder al detalle de la noticia.

Comentarios	Si el usuario no ha añadido ninguna criptomoneda en la sección <i>criptomonedas</i> de la <i>pantalla de home</i> (CU - 06), la lista de criptomonedas que se muestra en el <i>modal</i> estará bloqueada, y estará seleccionada el valor <i>Todas</i> . Los valores de ordenación estarán relacionados con la fecha de publicación de la noticia.
--------------------	--

Tabla 22. CU – 17: Filtrar noticias

CU – 18: Acceder al detalle de una noticia	
Descripción	Permite al usuario revisar el detalle de una noticia.
Prioridad	Alta
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de noticias</i>. • El usuario selecciona una de las noticias que se muestra por pantalla. • La aplicación redirige al usuario a la <i>pantalla de detalle de una noticia</i>. • La noticia es mostrada de forma embebida. <p>Flujo alternativo 1:</p> <ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de noticias guardadas</i>. • El usuario selecciona una de las noticias que se muestra por pantalla. • La aplicación redirige al usuario a la <i>pantalla de detalle de una noticia</i>. • La noticia es mostrada de forma embebida.
Requisitos a posteriori	Se muestra al usuario la noticia de forma embebida, de forma que consulta dicha noticia como si estuviera en la ruta de origen de la misma.
Comentarios	

Tabla 23. CU – 18: Acceder al detalle de una noticia

CU – 19: Acceder a noticias guardadas	
Descripción	Permite al usuario revisar las noticias de las criptomonedas que ha guardado.
Prioridad	Alta
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de noticias</i>.

	<ul style="list-style-type: none"> • El usuario pulsa en el botón de <i>guardados</i>. • La aplicación redirige al usuario a la <i>pantalla de noticias guardadas</i>. • La aplicación muestra las noticias guardadas en modo lista.
Requisitos a posteriori	Se muestra al usuario aquellas noticias relacionadas con el mundo de las criptomonedas que haya decidido guardar para volver a leer o para leer más tarde. Todas las noticias que se encuentran en la lista pueden pulsarse para acceder al detalle de la noticia.
Comentarios	Si el usuario no tiene ninguna noticia guardada, se mostrará una imagen informando de que no se hay noticias guardadas.

Tabla 24. CU – 19: Acceder a noticias guardadas

CU – 20: Guardar una noticia	
Descripción	Permite al usuario guardar una noticia.
Prioridad	Media
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de detalle de una noticia</i>. • El usuario pulsa el botón de <i>guardar una noticia</i>. • La aplicación almacena la noticia. • La aplicación informa al usuario de que la noticia ha sido guardada.
Requisitos a posteriori	El usuario podrá consultar todas las noticias que haya guardado durante su uso en la aplicación.
Comentarios	

Tabla 25. CU - 20: Guardar noticia

CU – 21: Borrar una noticia guardada	
Descripción	Permite al usuario borrar una noticia guardada.
Prioridad	Baja
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02). • El usuario debe haber guardado una noticia (CU – 20).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de noticias guardadas</i>. • El usuario hace un <i>swipe</i> de derecha a izquierda en uno de los elementos de la lista.

	<ul style="list-style-type: none"> • La aplicación muestra un botón de <i>borrar</i>. • El usuario pulsa el botón de <i>borrar</i>. • La aplicación borra la información almacenada relacionada con la noticia guardada. • La aplicación recarga la lista de noticias guardadas.
Requisitos a posteriori	La aplicación ya no cuenta con la noticia guardada.
Comentarios	

Tabla 26. CU – 21: Borrar una noticia guardada

CU – 22: Convertir un valor	
Descripción	Permite al usuario realizar una conversión entre una criptomoneda y una moneda fiduciaria, o a la inversa.
Prioridad	Alta
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en cualquier pantalla de la aplicación donde se encuentre el botón del <i>menú de navegación</i>. • El usuario pulsa el botón del <i>menú de navegación</i>. • El usuario pulsa el botón de <i>convertidor</i>. • La aplicación redirige a la <i>pantalla de conversión</i>. • La aplicación seleccionará la criptomoneda de forma aleatoria, basándose en las criptomonedas que el usuario haya añadido en la sección de <i>criptomonedas</i> en la <i>pantalla de home</i> (CU – 06). • La aplicación selecciona, por defecto, la moneda fiduciaria que el usuario haya seleccionado en el panel de preferencias. • El usuario podrá seleccionar cualquier criptomoneda y moneda fiduciaria para hacer la conversión. • El usuario introduce la cantidad que quiere convertir. • El usuario podrá intercambiar la posición entre la criptomoneda y la moneda fiduciaria para ver la conversión. • El usuario podrá borrar la cantidad introducida a convertir. • La aplicación, como información adicional, muestra la conversión en las otras criptomonedas que el usuario haya agregado en la sección de <i>criptomonedas</i> en la <i>pantalla de home</i>.
Requisitos a posteriori	El usuario podrá revisar la conversión entre una criptomoneda y una moneda fiduciaria, además

	de las otras criptomonedas que el usuario haya agregado en la sección de <i>criptomonedas</i> en la <i>pantalla de home</i> .
Comentarios	

Tabla 27. CU – 22: Convertir un valor

CU – 23: Añadir información adicional a la conversión de un valor	
Descripción	Permite al usuario añadir una criptomoneda o una moneda fiduciaria extra en la conversión de un valor.
Prioridad	Baja
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de conversión</i>. • El usuario pulsa en el botón de <i>añadir</i>. • La aplicación muestra un <i>modal</i> con una lista multiseleccionable de criptomonedas y una lista multiseleccionable de monedas fiduciarias. • El usuario selecciona aquellas criptomonedas y monedas fiduciarias que le interesan. • El usuario pulsa el botón de <i>añadir</i>. • La aplicación añadirá la selección del usuario en la conversión adicional de la conversión entre una criptomoneda y una moneda fiduciaria.
Requisitos a posteriori	El usuario podrá revisar la conversión en otras monedas adicionales que haya añadido de manera manual.
Comentarios	Cuando el usuario no se encuentre en la <i>pantalla de conversión</i> la información adicional añadida por el usuario será borrada.

Tabla 28. CU – 23: Añadir información adicional a la conversión de un valor

CU – 24: Acceder a la pantalla de alertas	
Descripción	Permite al usuario revisar las alertas relacionadas con la evolución del precio de una criptomoneda.
Prioridad	Alta
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en cualquier pantalla de la aplicación donde se encuentre el botón del <i>menú de navegación</i>.

	<ul style="list-style-type: none"> • El usuario pulsa el botón del <i>menú de navegación</i>. • El usuario pulsa el botón de <i>alertas</i>. • La aplicación redirige al usuario a la <i>pantalla de alertas</i>. • La aplicación muestra las alertas creadas por el usuario en forma de lista.
Requisitos a posteriori	El usuario podrá revisar las alertas que ha creado. Todas las alertas que se encuentran en la lista pueden pulsarse para acceder al detalle de la alerta.
Comentarios	Si el usuario no ha creado ninguna alerta, la <i>pantalla de alertas</i> no mostrará nada.

Tabla 29. CU – 24: Acceder a la pantalla de alertas

CU – 25: Añadir una alerta	
Descripción	Permite al usuario añadir una alerta sobre la evolución del precio de una criptomoneda.
Prioridad	Alta
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de alertas</i>. • El usuario pulsa el botón de <i>añadir</i>. • La aplicación redirige al usuario a la <i>pantalla de detalle de una alerta</i>. • La aplicación muestra una lista con las criptomonedas que el usuario haya añadido en la sección de <i>criptomonedas</i> en la <i>pantalla de home</i> (CU – 06), un <i>EditText</i> para que el usuario lo rellene con el valor de aumento del precio / porcentaje del precio de la criptomoneda y un <i>EditText</i> para que el usuario lo rellene con el valor de disminución del precio / porcentaje del precio de la criptomoneda. • El usuario selecciona y rellena la información requerida. • El usuario pulsa el botón de <i>guardar</i>. • La aplicación almacena la información. • La aplicación redirige al usuario a la <i>pantalla de alertas</i>. <p>Flujo alternativo 1:</p> <ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de detalle de una criptomoneda</i>. • El usuario pulsa el botón de <i>añadir alarma</i>. • La aplicación redirige al usuario a la <i>pantalla de detalle de una alerta</i>. • La aplicación muestra una lista con las criptomonedas que el usuario haya añadido en la sección de <i>criptomonedas</i> en la <i>pantalla de home</i> (CU – 06), un

	<p><i>EditText</i> para que el usuario lo rellene con el valor de aumento del precio / porcentaje del precio de la criptomoneda y un <i>EditText</i> para que el usuario lo rellene con el valor de disminución del precio / porcentaje del precio de la criptomoneda.</p> <ul style="list-style-type: none"> • El usuario selecciona y rellena la información requerida. • El usuario pulsa el botón de <i>guardar</i>. • La aplicación almacena la información. • La aplicación redirige al usuario a la <i>pantalla de detalle de una criptomoneda</i>.
Requisitos a posteriori	La aplicación permite al usuario crear una alerta para detectar la evolución del precio de una criptomoneda en base a unos valores determinados por el mismo.
Comentarios	

Tabla 30. CU – 25: Añadir una alerta

CU – 26: Ver una alerta	
Descripción	Permite al usuario ver una alerta.
Prioridad	Media
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02). • Haber creado una alerta (CU – 25).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de alertas</i>. • El usuario pulsa en una de las alertas que se muestra en la lista de alertas. • La aplicación redirige al usuario a la <i>pantalla de detalle de una alerta</i>. • El usuario ve la <i>pantalla de detalle de una alerta</i> en modo vista.
Requisitos a posteriori	El usuario podrá revisar, en detalle, una alerta que haya creado el mismo.
Comentarios	

Tabla 31. CU – 26: Ver una alerta

CU – 27: Editar una alerta	
Descripción	Permite al usuario modificar una alerta ya creada.
Prioridad	Baja
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02). • Haber creado una alerta (CU – 25).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de detalle de una alerta</i>.

	<ul style="list-style-type: none"> • El usuario pulsa el botón de <i>editar</i>. • La aplicación permite al usuario modificar los valores que haya establecido anteriormente al crear la alerta. • El usuario modifica la información. • El usuario pulsa el botón de <i>guardar</i>. • La aplicación almacena la información. • La aplicación redirige al usuario a la <i>pantalla de alertas</i>.
Requisitos a posteriori	El usuario podrá modificar una alerta ya creada y consultar, posteriormente, la alerta actualizada.
Comentarios	

Tabla 32. CU – 27: Editar una alerta

CU – 28: Eliminar una alerta	
Descripción	Permite al usuario eliminar una alerta.
Prioridad	Baja
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02). • Haber creado una alerta (CU – 25).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de alertas</i>. • El usuario realiza un <i>swipe</i> de derecha a izquierda sobre un elemento de la lista de alertas. • La aplicación muestra un botón de <i>eliminar</i>. • El usuario pulsa el botón de <i>eliminar</i>. • La aplicación borra la información almacenada relacionada con la alerta que se desea borrar. • La aplicación refresca la lista de alertas.
Requisitos a posteriori	La aplicación no contará con la alerta que el usuario haya creado previamente.
Comentarios	

Tabla 33. CU – 28: Eliminar una alerta

CU – 29: Notificar al usuario del cambio de precio de una criptomoneda	
Descripción	Permite al usuario recibir una notificación relacionada con el cambio de precio de una criptomoneda.
Prioridad	Media
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02). • Haber creado una alerta (CU – 25).

Flujo	<ul style="list-style-type: none"> • El usuario se encuentra identificado, pero sin tener la aplicación abierta. • La aplicación tendrá un proceso en segundo plano que irá preguntando cada X tiempo si el precio de una criptomoneda ha cambiado en base a la alerta que haya creado el usuario previamente. • El proceso detecta que ha cambiado el precio de la criptomoneda. • La aplicación envía una notificación al usuario informando de que el precio de una criptomoneda ha variado en base a la alerta que haya creado.
Requisitos a posteriori	El usuario podrá recibir notificaciones relacionadas con las alertas en los precios de las criptomonedas que el haya definido.
Comentarios	

Tabla 34. CU – 29: Notificar al usuario del cambio de precio de una criptomoneda

CU – 30: Acceder a la pantalla de cuenta del usuario	
Descripción	Permite al usuario revisar su cuenta de usuario.
Prioridad	Alta
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en cualquier pantalla de la aplicación donde se encuentre el botón del <i>menú de navegación</i>. • El usuario pulsa el botón del <i>menú de navegación</i>. • El usuario pulsa el botón de <i>usuario</i>. • La aplicación redirige al usuario a la <i>pantalla de cuenta del usuario</i>. • La aplicación muestra la información relacionada con el usuario identificado en modo vista.
Requisitos a posteriori	El usuario podrá revisar su información personal.
Comentarios	

Tabla 35. CU – 30: Acceder a la pantalla de cuenta del usuario

CU – 31: Modificar la información del usuario	
Descripción	Permite al usuario modificar su información personal.
Prioridad	Media
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).

Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de cuenta del usuario</i>. • El usuario pulsa el botón de <i>editar</i>. • La aplicación muestra la información relacionada con el usuario identificado, y permite a este mismo poder editarla. • El usuario modifica su información personal. • El usuario pulsa en el botón de <i>guardar</i>. • La aplicación almacena la información actualizada. • La aplicación muestra la información actualizada en modo vista.
Requisitos a posteriori	El usuario podrá modificar su información personal y, consultar posteriormente, la información actualizada.
Comentarios	

Tabla 36. CU – 31: Modificar la información del usuario

CU – 32: Acceder a la pantalla de preferencias	
Descripción	Permite al usuario ver la configuración de la aplicación.
Prioridad	Alta
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de cuenta del usuario</i>. • El usuario pulsa el botón de <i>preferencias</i>. • La aplicación redirige al usuario a la <i>pantalla de preferencias</i>. • La aplicación muestra al usuario los datos que pueden ser configurados en modo vista. <p>Flujo alternativo 1:</p> <ul style="list-style-type: none"> • El usuario se encuentra en cualquier pantalla de la aplicación donde se encuentre el botón del menú de navegación. • El usuario pulsa el botón del <i>menú de navegación</i>. • El usuario pulsa el botón de <i>preferencias</i>. • La aplicación redirige al usuario a la <i>pantalla de preferencias</i>. • La aplicación muestra al usuario los datos que pueden ser configurados en modo vista.
Requisitos a posteriori	El usuario podrá revisar la configuración que tiene establecida que determina los parámetros de ciertos servicios.

Comentarios	La configuración inicial tendrá unos valores por defecto cuando el usuario se registre y se identifique por primera vez en la aplicación.
--------------------	---

Tabla 37. CU – 32: Acceder a la pantalla de preferencias

CU – 33: Editar preferencias	
Descripción	Permite al usuario editar las preferencias de la aplicación.
Prioridad	Media
Actores	Usuario identificado.
Prerrequisitos	El usuario debe: <ul style="list-style-type: none"> • Estar registrado en la aplicación (CU – 01). • Estar identificado en la aplicación (CU – 02).
Flujo	<ul style="list-style-type: none"> • El usuario se encuentra en la <i>pantalla de preferencias</i>. • El usuario pulsa el botón de <i>editar</i>. • La aplicación permite al usuario modificar datos. • El usuario edita aquellos datos que vea necesarios. • El usuario pulsa el botón de <i>guardar</i>. • La aplicación almacena los datos modificados. • La aplicación muestra la <i>pantalla de preferencias</i> en modo vista.
Requisitos a posteriori	El usuario podrá consultar las preferencias modificadas.
Comentarios	

Tabla 38. CU – 33: Editar preferencias

2.5.2. Diseño de la arquitectura

Desde el principio del planteamiento de la aplicación, se ha optado por el patrón arquitectónico MVVM (Model – View – ViewModel). Además de que es un patrón arquitectónico que recomienda Google [15], encaja muy bien con el tipo de aplicación que se quiere realizar.

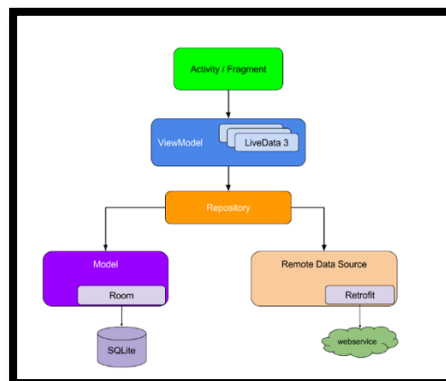


Figura 52. Patrón arquitectónico MVVM, establecido por Google

Un patrón arquitectónico MVVM tiene como finalidad la de separar la lógica de negocio de la interfaz de usuario, facilitando las pruebas, mantenimiento y la estabilidad de los proyectos en los que se emplee. Está compuesto por 3 componentes:

- **View:** Se encarga de mostrar la interfaz. Su principal función es la de observar componentes **ViewModel** para obtener los datos necesarios que necesita y actualizar la interfaz según sea conveniente. También informa al componente **ViewModel** sobre las acciones del usuario. El componente **View** puede tener dependencias con uno o más componentes **ViewModel**, pero este último nunca puede tener información sobre el componente **View**.
- **Model:** Está compuesto por todas las clases relacionadas con el acceso de datos, reglas de negocio y obtención de la información. Además, es el encargado de exponer los datos necesarios al componente **ViewModel** de forma que puedan ser consumidos fácilmente. También recibe cualquier evento del componente **ViewModel** que necesite para realizar cualquier operación necesaria sobre los datos. Es importante seguir el principio *single responsibility* para la creación de un modelo por cada objeto lógico de la fuente de datos.
- **ViewModel:** Componente que retorna la información proveniente del componente **Model** hacia el componente **View**, aplicando las operaciones necesarias. Está diseñado para manejar y almacenar la información, sin que esta se vea afectada por los cambios de configuración o del ciclo de vida de un elemento perteneciente al componente **View**. Para ello, se mantiene en memoria hasta que el objeto del ciclo de vida al que pertenece haya terminado completamente. Además, este componente expone eventos que el componente **View** puede observar y reaccionar.

Los beneficios que aporta este patrón arquitectónico sobre los demás existentes son:

- Permite localizar y cambiar partes del código más fácilmente, debido a la separación de responsabilidades.
- En caso de que el código que se haya implementado de la manera correcta, todas las dependencias internas y externas se mantendrán alejadas del código que contiene la lógica de negocio, pudiendo realizar pruebas de manera más fácil.
- La capa de lógica y la de presentación están débilmente acopladas.
- Capacidad de testear los componentes de manera independiente.

Pero no todo son beneficios. Cualquier tipo de patrón arquitectónico tiene sus desventajas:

- Demasiado complejo para aplicaciones que tienen una interfaz simple.
- Se necesita crear observables por cada componente de la interfaz.
- El código puede ser excesivo.

Se ha decidido también, implementar otra arquitectura que va a servir de complemento a la anteriormente mencionada, y es la arquitectura limpia.

Esta arquitectura es una filosofía de diseño de software presentada por *Robert C. Martin* en el libro *Clean Architecture*, y principalmente se refiere a la organización del código en componentes o módulos separados y como estos elementos se relacionan entre sí. Una de las principales razones por las que se ha decidido implementar es porque permite al código tener pocas dependencias y sea fácil de entender y mantener, reduciendo costes y maximizando la productividad del desarrollador.

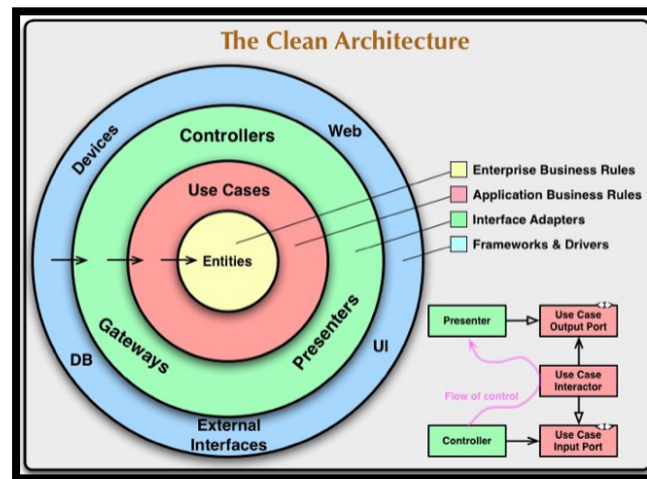


Figura 53. Arquitectura limpia, definida por Robert C. Martin

Para el diseño de la base de datos, se ha tenido en cuenta que la mayoría de datos que se va a mostrar en la aplicación van a ser proporcionados por APIs externas. El resto de datos que vamos a mantener que sean relevantes en el uso de la aplicación están expuestos en la siguiente imagen, que representa un diagrama entidad relación:

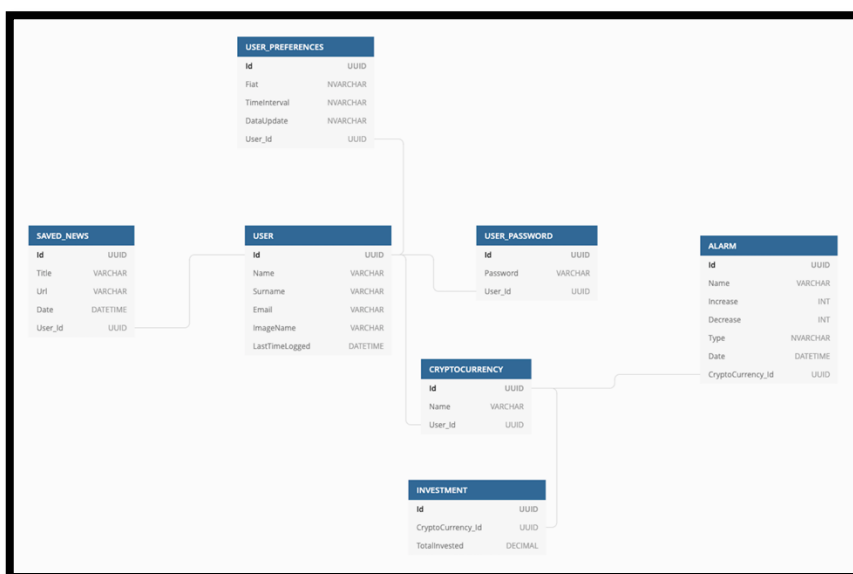


Figura 54. Diagrama entidad – relación de la base de datos

- **USER_PREFERENCES:** Entidad que representa las preferencias que defina el usuario.
- **USER:** Representa la información referente al usuario. Esta tabla se rellenará cuando el usuario decida registrarse o identificarse mediante el formulario correspondiente.
- **USER_PASSWORD:** Representa la contraseña de acceso con la que se registrará el usuario. Esta tabla se rellenará cuando el usuario se registre mediante el formulario proporcionado.
- **SAVED_NEWS:** Aquí se almacenará la información relacionada con las noticias que haya guardado el usuario.
- **CRYPTOCURRENCY:** Representa aquellas criptomonedas que el usuario ha decidido añadir a la aplicación para su seguimiento. Se almacenará la información básica para poder realizar la petición a la API externa y obtener sus datos completos.
- **INVESTMENT:** Guarda la información relacionada con la inversión realizada por el usuario en una o más criptomonedas.
- **ALARM:** Representa las alarmas que ha creado el usuario sobre la evolución del precio de una o más criptomonedas.

Los datos proporcionados por las APIs externas serán serializadas a entidades dentro de la aplicación. Sin embargo, para evitar estar constantemente haciendo llamadas a las APIs, se va a implementar un mecanismo de caché para que la experiencia de usuario sea lo más satisfactoria posible.

Una de las decisiones que ha impactado sobre como se almacenar cierta información del usuario, es la de incorporar el uso de las credenciales de redes sociales en el registro/*login* del usuario. Para la implementación de esta funcionalidad, se ha decidido utilizar la librería *firebase*, ya que proporciona dicha funcionalidad de forma fácil y eficiente.

Por lo tanto, está aplicación implementará dos formas de almacenar los datos: una, mediante el almacenamiento de datos de forma relacional a través de la librería *Room* (Base de datos local); y otra mediante el almacenamiento de datos de forma no relacional, a través de la librería de *firebase* (Base de datos remota).

3. Implementación

En esta sección se exponen los aspectos más relevantes de la implementación de la aplicación, que se ha llevado a cabo con el IDE Android Studio, en su versión 4.1.2; y con el lenguaje de programación Kotlin.

3.1. Desarrollo

La principal problemática que se ha encontrado a lo largo del desarrollo de la aplicación ha sido la de realizar peticiones, tanto a APIs externas como a

la base de datos local; y el manejo de su resultado. Por ello, se ha implementado una clase denominada *Either*.

Either es un contenedor de datos usado en programación funcional (FP) que representa un valor de dos posibles tipos, denominados *Left* y *Right*. Las convenciones de la programación funcional determinan que el valor *Left* se utilice para errores y el valor *Right* para valores correctos.

```
sealed class Either<out L, out R> {
    // Represent the left side
    data class Left<out L>(val failure: L): Either<L, Nothing>()
    // Represent the right side
    data class Right<out R>(val success: R): Either<Nothing, R>()

    // Return true if this is a Right, false otherwise
    val isRight get() = this is Right<R>
    // Return true if this is a Left, false otherwise
    val isLeft get() = this is Left<L>

    // Creates a Left Type
    fun <L> left(failure: L) = Either.Left(failure)
    // Create a Right Type
    fun <R> right(success: R) = Either.Right(success)
    // Applies fnL if this is a Left or fnR if this is a Right
    fun fold(fnL: (L) -> Any, fnR: (R) -> Any): Any =
        when(this) {
            is Left -> fnL(failure)
            is Right -> fnR(success)
        }
}

// Returns the value from this `Right` or the given argument if this is a `Left`.
fun <L, R> Either<L, R>.getOrElse(value: R): R =
    when (this) {
        is Either.Left -> value
        is Either.Right -> success
    }
}
```

Figura 55. Clase *Either*

El uso de *Either* es excelente para el manejo de errores, ya que hace que el código sea más declarativo, legible y robusto [16].

Para todas las peticiones que se hacen a APIs externas, se ha decidido crear una clase abstracta que tiene definida dos funciones: Una función que permite realizar la petición y obtener los datos tal cual los devuelve la API; y otra función que permite realizar la petición y convertir los datos obtenidos según las necesidades del servicio.

```

/**
 * Call external GET request and convert the response
 */
suspend inline fun <reified T,R> requestGET(
    httpClient: HttpClient,
    endpoint: String,
    params: Map<String, Any?>,
    converter: (T) -> R
): Either<Fail, R> {
    return try {
        val response = httpClient.get<T>(endpoint) { this: HttpRequestBuilder()
            params?.let { it: Map<String, Any?>
                it.forEach { (k, v) -> parameter(k, v)}
            }
        }
        return response?.let { Right(converter(response)) } ?: Left(Fail.NotFoundFail)
    } catch(exception: Throwable){
        Log.e(tag: "BaseService", exception.stackTraceToString())
        Left(Fail.ServerFail)
    }
}

/**
 * Call external GET request
 */
suspend inline fun <reified T> requestGETRaw(
    httpClient: HttpClient,
    endpoint: String,
    params: Map<String, Any?>
): Either<Fail, T> {
    return try {
        val response = httpClient.get<T>(endpoint) { this: HttpRequestBuilder()
            params?.let { it: Map<String, Any?>
                it.forEach { (k, v) -> parameter(k, v)}
            }
        }
        return response?.let { Right(response) } ?: Left(Fail.NotFoundFail)
    } catch(exception: Throwable){
        Log.e(tag: "BaseService", exception.stackTraceToString())
        Left(Fail.ServerFail)
    }
}

```

Figura 56. Métodos que se encuentra en la clase abstracta BaseService

Todas las llamadas a APIs externas hacen uso de una URL, unos parámetros, una función que convierte datos (en caso de que sea necesario) y un objeto *httpClient* que permite realizar la llamada. Configurando todos estos objetos como parámetros, se puede encapsular todas las llamadas en una clase abstracta evitando repetir código y haciéndolo más robusto.

Otras de las problemáticas que se han encontrado ha sido la de crear una *toolbar* diferente para las pantallas que forman la aplicación. Para solucionarlo, se ha creado una *Activity* base que contiene dos secciones: una sección donde se implementará el contenido de la pantalla, y otra sección donde se implementará el *toolbar* específico.


```

open class BaseActivity : AppCompatActivity() {
    lateinit var toolbar: Toolbar

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_base)
    }

    /**
     * Set the specific toolbar for each activity
     */
    fun initSetToolbar(toolbarLayout: Int, toolbarId: Int) {
        // Clear and set toolbar view
        val toolbarContent = findViewById<ViewStub>(R.id.toolbar_content)
        toolbarContent.layoutResource = (toolbarLayout)
        toolbarContent.inflate()
        toolbar = findViewById(toolbarId)
        setSupportActionBar(toolbar)
    }

    /**
     * Set the specific layout for each activity
     */
    fun initSetLayout(activityLayout: Int) {
        val inflater = getSystemService(Context.LAYOUT_INFLATER_SERVICE) as LayoutInflater
        val view = inflater.inflate(activityLayout, root = null)
        main_content_below.addView(view)
    }

    /**
     * Show/Hide specific view
     */
    fun showView(view: View, show: Boolean) {
        view.visibility = if(show) View.VISIBLE else View.GONE
    }
}

```

La clase que hace referencia a la *Activity* base está compuesta de dos funciones principales:

- Una función que permite establecer cada *toolbar* específico a la pantalla correspondiente.
- Una función que permite establecer el contenido de la pantalla.

Toda *Activity* que necesite de un *toolbar* específico, debe heredar esta clase y establecer la información pertinente en cada función explicada anteriormente. Estas funciones deben ser llamadas justo en la función **onCreate**.

3.1.1. Librerías utilizadas

Las librerías que han sido utilizadas para conseguir los objetivos en el desarrollo de la aplicación son:

- **Ktor**: Framework que permite realizar peticiones asíncronas a través de la red. Gracias a su cliente **httpClient**, se han podido establecer las llamadas a APIs externas.
- **Kotlinx – Serialization / Gson**: Permite serializar los datos obtenidos en las peticiones a APIs externas a modelos de datos definidos en la aplicación.
- **Coroutines**: Permite escribir código asíncrono de una manera más natural, es decir, es un estilo de programación secuencial, que es más comprensible y legible al ojo humano.

- **Lyfecycle:** Permite definir los objetos que tienen un ciclo de vida en Android junto con su estado e información de eventos. Es una librería necesaria para poder utilizar *ViewModels*.
- **MPAndroidChart:** Librería que permite establecer gráficas en Android.
- **Material Design:** Permite hacer uso de componentes específicos en Android. Se ha hecho un uso específico para la creación de *toolbars*.
- **Glide:** Permite cargar imágenes. Su uso es intuitivo y permite mostrar la imagen de diferentes formas.
- **Swipe Reveal Layout:** Librería que permite añadir el gesto *swipe* en un *layout* para mostrar otro *layout*. Esta librería se ha utilizado en los ítems que se encuentran en ciertos *recycler views*, con el fin de poder editar o borrar.
- **Room:** Librería de base de datos que simplifica la tarea de trabajar con base de datos en Android.

3.2. Pruebas

Para la ejecución de las pruebas se ha utilizado el emulador que proporciona Android Studio, simulando un dispositivo **Pixel 4 XL** con Android 11.

Pantalla de Splash	
Prueba	Resultado
Si el usuario tiene una sesión almacenada en la aplicación, se navega directamente a la <i>pantalla de home</i> .	Se aplica.
Si el usuario no tiene una sesión almacenada, se navega directamente a la <i>pantalla de login</i> .	Se aplica.
Si el usuario tiene una sesión almacenada, se obtiene el usuario y sus preferencias y se establecen como variables globales.	Se aplica.
Si el usuario tiene una sesión almacenada, se actualiza la fecha de <i>login</i> .	Se aplica.
Si se produce algún error en la <i>pantalla de Splash</i> se informa al usuario con un mensaje por pantalla.	Se aplica.

Tabla 39. Pruebas en pantalla de splash

Pantalla de Registro	
Prueba	Resultado
Ningún campo del registro debe estar vacío.	Se aplica. Se le muestra al usuario un mensaje por pantalla
El formato del email debe ser válido.	Se aplica. Se le muestra al usuario un mensaje por pantalla.
El campo contraseña y repetir contraseña deben ser iguales.	Se aplica. Se le muestra al usuario un mensaje por pantalla.
El email introducido no debe existir en la base de datos.	Se aplica. Se le muestra al usuario un mensaje por pantalla.

Si todos los campos son válidos, se guardan los datos del usuario y se navega a la <i>pantalla de login</i> .	Se aplica.
Al momento de guardar el usuario, se guardan unas preferencias por defecto.	Se aplica.
Se accede a la <i>pantalla de login</i> a través del botón iniciar sesión .	Se aplica.
Registro a través de Google .	No implementado.
Registro a través de Facebook .	No implementado.
Registro a través de Twitter .	No implementado.

Tabla 40. Pruebas en pantalla de registro

Pantalla de Login	
Prueba	Resultado
Ningún campo del <i>login</i> debe estar vacío.	Se aplica. Se le muestra al usuario un mensaje por pantalla
El email introducido debe existir en la base de datos.	Se aplica. Se le muestra al usuario un mensaje por pantalla.
La contraseña introducida debe estar almacenada en la base de datos y relacionada con el email introducido.	Se aplica. Se le muestra al usuario un mensaje por pantalla.
Si todos los campos son válidos, se actualiza la fecha de <i>login</i> , se almacena la sesión y se navega a la <i>pantalla de home</i> .	Se aplica.
Se accede a la <i>pantalla de registro</i> a través del botón crear cuenta .	Se aplica.
<i>Login</i> a través de Google .	No implementado.
<i>Login</i> a través de Facebook .	No implementado.
<i>Login</i> a través de Twitter .	No implementado.

Tabla 41. Pruebas en pantalla de login

Pantalla de Home	
Prueba	Resultado
Aparece un mensaje de bienvenida junto a una frase motivadora y su autor.	Se aplica.
Si la sección de criptomonedas está vacía, se muestra un icono de que la sección está vacía.	Se aplica.
Al pulsar en el botón de añadir de la sección criptomonedas aparece un modal con una lista de criptomonedas a añadir.	Se aplica.
Si no se selecciona ningún elemento del modal de criptomonedas no permite añadir.	Sí. Se le muestra al usuario un mensaje por pantalla.
Al añadir una o varias criptomonedas, las criptomonedas se guardan en la base de datos, la sección de criptomonedas se recarga y muestra una lista con la información de cada una de las criptomonedas añadidas.	Se aplica.
Al pulsar en el botón de actualizar en uno de los ítems de la lista de criptomonedas, dicha información se actualiza.	Se aplica.

Al pulsar el botón de borrar en uno de los ítems de la lista de criptomonedas, dicha criptomoneda se borra de la base de datos y de la lista, y si tiene alguna inversión en la sección del portfolio, también se borra.	Se aplica.
Si se pulsa en uno de los ítems de la lista de criptomonedas, se navega a la <i>pantalla de detalle</i> del ítem seleccionado.	Sí.
Si no hay ninguna criptomoneda en la sección de criptomonedas, no se permite añadir ninguna inversión en la sección de portfolio.	Sí.
Al pulsar el botón de añadir de la sección de portfolio aparece un modal con la lista de criptomonedas añadidas en la sección de criptomonedas y un campo para añadir la cantidad invertida.	Sí.
Si no se selecciona ninguna criptomoneda o no se introduce una cantidad invertida en el modal de inversiones, no permite añadir.	Sí. Se le muestra al usuario un mensaje por pantalla.
Al añadir una inversión, la sección de portfolio se recarga y muestra una lista con la información de la inversión de cada una de las criptomonedas.	Sí.
Al hacer <i>swipe</i> en cada uno de ítems de la sección de portfolio, aparece un botón de actualizar y borrar.	Sí.
Si se pulsa en el botón de actualizar en uno de los ítems de la sección de portfolio, aparece un modal que permite actualizar la cantidad de inversión realizada sobre una criptomoneda.	Sí.
Si no se introduce una cantidad invertida en el modal de actualización de inversiones, no se permite actualizar.	Sí.
Al actualizar la inversión, la lista de la sección de portfolio se refresca.	Sí.
Si se pulsa en el botón de eliminar en uno de los ítems de la sección del portfolio, la inversión seleccionada se elimina y se refresca la sección.	Sí.
Si se produce algún error en alguna de las secciones, se muestra un icono de error y un mensaje informando al usuario.	Sí.
Si se pulsa en el icono del menú del <i>toolbar</i> de la pantalla, permite navegar a las pantallas de <i>noticias</i> , <i>conversor</i> , <i>usuario</i> y <i>preferencias del usuario</i> .	Sí.

Tabla 42. Pruebas en pantalla de home

Pantalla de detalle de una criptomoneda	
Prueba	Resultado
Se muestra información detallada sobre una criptomoneda.	Sí.
Si existe una inversión sobre la criptomoneda, la sección de inversión se rellena; en caso contrario se muestra un icono de que la sección está vacía.	Sí.
Cuando se pulsa en uno de los botones para modificar el intervalo de tiempo en el que se	Sí.

muestra los datos de la gráfica, la gráfica se refresca.	
Al pulsar en uno de los botones mostrados en la sección de enlaces, abre un enlace web.	Sí.
Si el enlace de uno de los botones mostrados en la sección de enlaces no está disponible, el botón se deshabilita.	Sí.
Al pulsar el botón de actualizar, toda la información de la pantalla se refresca.	Sí.
Al pulsar el botón de atrás, se navega a la <i>pantalla de home</i> .	Sí.
Si se produce algún error en alguna de las secciones, se muestra un icono de error y un mensaje informando al usuario.	Sí.

Tabla 43. Pruebas en pantalla de detalle de una criptomoneda

Pantalla de Noticias	
Prueba	Resultado
Se muestra una lista de noticias.	Sí.
Cuando se selecciona un ítem de la lista de noticias, se navega hacia la <i>pantalla de detalle de una noticia</i> .	Sí.
Cuando se pulsa el botón de noticias guardadas, se navega hacia la <i>pantalla de noticias guardadas</i> .	Sí.
Cuando se pulsa en el botón de filtros, aparece un modal con la lista de criptomonedas añadidas por el usuario en la <i>pantalla de home</i> , y una sección de botones para poder ordenar las noticias.	Sí.
Si no se selecciona ninguna criptomoneda o un campo por el que ordenar en el modal de filtros, no se filtra.	Sí. Se le muestra al usuario un mensaje por pantalla.
Limpia filtros aplicados.	No implementado.
Al aplicar filtros, se muestra una lista de noticias filtradas; en caso contrario, se muestra un icono de que la lista está vacía.	Sí.
Si se produce algún error en alguna de las secciones, se muestra un icono de error y un mensaje informando al usuario.	Sí.
Si se pulsa en el icono del menú del <i>toolbar</i> de la pantalla, permite navegar a las pantallas de <i>noticias</i> , <i>conversor</i> , <i>usuario</i> y <i>preferencias del usuario</i> .	Sí.

Tabla 44. Pruebas en pantalla de noticias

Pantalla de detalle de una noticia	
Prueba	Resultado
Se muestra la noticia en un contenedor web.	Sí.
Al pulsar el botón de guardar noticia, si la noticia ya existe en la base de datos, no se guarda.	Sí. Se le muestra al usuario un mensaje por pantalla.
Al pulsar el botón de guardar noticia, si la noticia no existe en la base de datos, se guarda.	Sí. Se le muestra al usuario un mensaje por pantalla.

Al pulsar el botón de atrás, se navega a la <i>pantalla de noticias</i> .	Sí.
---	-----

Tabla 45. Pruebas en pantalla de una noticia

Pantalla de noticias guardadas	
Prueba	Resultado
Se muestra una lista con las noticias guardadas por el usuario; en caso contrario, se muestra un icono de que la lista está vacía.	Sí.
Al pulsar en un ítem de la lista, se navega hacia la <i>pantalla de detalle de una noticia</i> .	Sí.
Al hacer <i>swipe</i> en cada uno de ítems de las noticias guardadas, aparece un botón de borrar.	Sí.
Si se pulsa en el botón de eliminar en uno de los ítems de la lista, la noticia seleccionada se elimina y se refresca la lista.	Sí.
Al pulsar el botón de atrás, se navega a la <i>pantalla de noticias</i> .	Sí.
Si se produce algún error en alguna de las secciones, se muestra un icono de error y un mensaje informando al usuario.	Sí.

Tabla 46. Pruebas en pantalla de noticias guardadas

Pantalla de conversión	
Prueba	Resultado
Cuando se entra por primera vez, la moneda a convertir es la moneda que tenga el usuario configurado en la <i>pantalla de preferencias</i> .	Sí.
Cuando entra por primera vez, la criptomoneda a convertir es la primera, en orden alfabético, que está almacenada en la base de datos.	Sí.
Cuando entra por primera vez el campo de cantidad está vacío.	Sí.
Cuando se entra por primera vez, la sección de criptomonedas almacenadas y sección de cantidad no están visibles.	Sí.
Cuando se introduce una cantidad, al cabo de un segundo se cierra el teclado.	Sí.
Cuando se introduce una cantidad, aparece la conversión en la criptomoneda seleccionada, la sección de cantidad y la sección de criptomonedas almacenadas.	Sí.
Cuando aparece la sección de criptomonedas, si la criptomoneda seleccionada es una de las criptomonedas que se encuentran en la sección, no se muestra en dicha sección.	Sí.
Cuando se pulsa en la moneda, aparece un modal con una lista de monedas.	Sí.
Si no se selecciona ningún elemento de la lista de monedas, no permite añadir.	Sí. Se le muestra al usuario un mensaje por pantalla.
Cuando se añade una moneda, se recalcula toda la conversión para dicha moneda.	Sí.

Cuando se pulsa en la criptomoneda, aparece un modal con una lista de criptomonedas.	Sí.
Si no se selecciona ningún elemento de la lista de criptomonedas, no permite añadir	Sí. Se le muestra al usuario un mensaje por pantalla.
Cuando se añade una criptomoneda, se recalcula la conversión para dicha criptomoneda y, si la criptomoneda previa era una de las criptomonedas almacenadas, se añade a la sección de criptomonedas.	Sí.
Cuando se pulsa el botón de eliminar, se muestra la pantalla como si entrada por primera vez.	Sí.
Si se produce algún error en alguna de las secciones, se muestra un icono de error y un mensaje informando al usuario.	Sí.
Si se pulsa en el icono del menú del <i>toolbar</i> de la pantalla, permite navegar a las pantallas de <i>noticias</i> , <i>conversor</i> , <i>usuario</i> y <i>preferencias del usuario</i> .	Sí.

Tabla 47. Pruebas en pantalla de conversión

Pantalla de usuario	
Prueba	Resultado
Al entrar en la pantalla podemos ver la información del usuario en modo vista.	Sí.
Al pulsar en el botón de preferencias, se navega a la <i>pantalla de preferencias del usuario</i> .	Sí.
Al pulsar el botón de editar, se muestran los campos en modo edición.	Sí.
Al pulsar el botón de cerrar sesión, se borra la sesión almacenada del usuario y se navega hasta la <i>pantalla de login</i> .	Sí.
Al pulsar el botón de eliminar cuenta, aparece un modal de confirmación.	Sí.
Al pulsar el botón de confirmar en el modal de confirmación, se borra toda la información de la base de datos y de la sesión relacionada con el usuario y se navega a la <i>pantalla de login</i> .	Sí.
Cuando aparece la sección de criptomonedas, si la criptomoneda seleccionada es una de las criptomonedas que se encuentran en la sección, no se muestra en dicha sección.	Sí.
Cuando se pulsa en la moneda, aparece un modal con una lista de monedas.	Sí.
Si no se selecciona ningún elemento de la lista de monedas, no permite añadir.	Sí. Se le muestra al usuario un mensaje por pantalla.
Cuando se añade una moneda, se recalcula toda la conversión para dicha moneda.	Sí.
Cuando se pulsa en la criptomoneda, aparece un modal con una lista de criptomonedas.	Sí.
Si no se selecciona ningún elemento de la lista de criptomonedas, no permite añadir	Sí. Se le muestra al usuario un mensaje por pantalla.
Cuando se añade una criptomoneda, se recalcula la conversión para dicha criptomoneda y, si la criptomoneda previa era una de las criptomonedas	Sí.

almacenadas, se añade a la sección de criptomonedas.	
Cuando se pulsa el botón de eliminar, se muestra la pantalla como si entrada por primera vez.	Sí.
Ningún campo de la edición del usuario debe estar vacío.	Sí. Se le muestra al usuario un mensaje por pantalla
El formato del email debe ser válido.	Sí. Se le muestra al usuario un mensaje por pantalla.
El campo contraseña y repetir contraseña deben ser iguales.	Sí. Se le muestra al usuario un mensaje por pantalla.
El email introducido no debe existir en la base de datos.	Sí. Se le muestra al usuario un mensaje por pantalla.
Si todos los campos son válidos, se guardan los datos del usuario se muestra la pantalla en modo vista.	Sí.
Si se pulsa el botón de cancelar en el modo edición, se vuelve al modo vista.	Sí
Si se pulsa en el icono del menú del <i>toolbar</i> de la pantalla, permite navegar a las pantallas de <i>noticias</i> , <i>conversor</i> , <i>usuario</i> y <i>preferencias del usuario</i> .	Sí.

Tabla 48. Pruebas en pantalla de usuario

Pantalla de preferencias del usuario	
Prueba	Resultado
Se muestra una sección de lista de monedas, una sección de intervalo de tiempo en gráficas y una sección de actualización de datos en modo vista.	Sí.
Al pulsar el botón de editar, aparecen los botones de cancelar y guardar.	Sí.
Al pulsar el botón de guardar se guarda las nuevas preferencias del usuario, y se vuelve al modo vista.	Sí.
Al guardar una nueva moneda y un nuevo intervalo de tiempo en gráficas, esto repercute en la <i>pantalla de home</i> , en la <i>pantalla de detalle de una criptomoneda</i> y en la <i>pantalla conversión</i> .	Sí.
Al pulsar el botón de atrás, se navega a la pantalla anterior.	Sí.

Tabla 49. Pruebas en pantalla de preferencias del usuario

3.3. Observaciones tras implementación

Durante la fase de desarrollo de la aplicación, han surgido una serie de imprevistos que han provocado la toma de decisiones significativas.

El tiempo de desarrollo se ha visto reducido debido a una serie de acontecimientos. La más significativa ha sido cambiar de empleo al principio de empezar la PEC3. Las entrevistas realizadas y la aclimatación al nuevo puesto de trabajo, han restado un tiempo considerable que hacía difícil cumplir con las previsiones expuestas en la planificación.

Otro punto importante ha sido la obtención de datos a partir de APIs Externas, concretamente la API de noticias. Ciertas funcionalidades solo estaban disponibles en una versión de pago. Por lo tanto, se ha tenido que reencauzar el desarrollo para poder aplicar los datos que se obtienen a partir de las funcionalidades que estaban en la versión gratuita. Además, la librería de deserialización ha tenido problemas con la API externa que obtiene la frase del día, teniendo que personalizar el servicio establecido y deserializar la información a partir de otra librería.

El poco conocimiento en el uso de *Fragments* así como la elaboración de pantallas con bastante información y detalle en el diseño, ha supuesto un exceso de tiempo a la hora de atender a todos los pequeños detalles que necesitan ser implementados.

Con todo lo anteriormente mencionado, y revisando que funcionalidades eran más importantes de implementar con el tiempo que se manejaba, se ha decidió no implementar lo siguiente:

- CU – 24: Acceder a la pantalla de alertas.
- CU – 25: Añadir una alerta.
- CU – 26: Ver una alerta.
- CU – 29: Notificar al usuario del cambio de precio de una criptomoneda.
- CU – 27: Editar una alerta.
- CU – 28: Eliminar una alerta.
- Registro / login a través de cuentas de otras aplicaciones.
- Limpiar la aplicación de filtros en la *pantalla de noticias*.

4. Conclusiones

El desarrollo de este proyecto ha dejado como lección más importante la necesidad de planificar adecuadamente los objetivos. La complejidad de las tareas a realizar, tanto en el diseño como en la implementación de código, hacen que la estimación detallada durante la planificación temporal de las tareas no haya sido la real. También ha influido, en gran medida, el cambio de trabajo al inicio de la implementación de la aplicación, ya que la realización de entrevistas y la aclimatación al nuevo puesto de trabajo ha reducido el tiempo de desarrollo.

Otra lección que se ha aprendido, como consecuencia de lo anteriormente expuesto, es la de establecer cierto margen en la planificación temporal, con el fin de evitar o reducir los problemas que surjan durante el tiempo de implementación.

La realización de este proyecto ha permitido profundizar en diferentes herramientas: IDE de Android Studio, Kotlin, GitHub¹ y AdobeXD. La mayoría de

¹ Fuentes del proyecto: <https://github.com/antonalag/crypto-tracker.git>

ellas estarán presentes en futuros proyectos, tanto profesionales como personales, con el fin de asentar los conocimientos obtenidos como en la realización exitosa de los objetivos que se planteen.

Aunque se haya utilizado la arquitectura MVVM durante la realización de ciertas asignaturas de este máster, su uso durante la implementación de este proyecto ha permitido profundizar en diferentes aspectos como el manejo de la asincronía y el ciclo de vida de los componentes visuales, facilitando el desarrollo y mejorando la estructura de las clases a lo largo del código.

Uno de los objetivos más complejos que se plantearon desde el inicio fue el poder gestionar, de manera personal, la consecución de este proyecto. El tener que compaginar un trabajo a jornada completa y vida personal, con el desarrollo del trabajo de fin de máster ha supuesto un reto, día a día, en la gestión del tiempo y en la autodisciplina. Esto ha llevado a una conclusión clara: para realizar un proyecto completo de forma solitaria, se requiere de una gran cantidad de tiempo. Por lo tanto, siempre hay que pensar de forma realista la planificación de cualquier proyecto que se tenga entre manos, con el fin de evitar abarcar demasiados objetivos que no puedan ser alcanzados en un tiempo establecido, provocando malestar y baja autoestima en el desarrollador.

A pesar de que no se haya conseguido el 100% de los objetivos, se puede estar orgulloso de lo realizado a lo largo de la implementación. Se ha conseguido crear una aplicación con buen rendimiento, que consume mucha información en tiempo real a partir de APIs externas, con un diseño minimalista y agradable a la vista, que permite obtener una experiencia de usuario satisfactoria.

Como líneas de trabajo futuro, se establecen aquellos casos de uso que no han podido realizarse durante la fase de implementación, que están determinados en el [punto 3.3](#) de esta memoria. Además, se pueden incluir ciertas funcionalidades que pueden otorgar un valor añadido a la aplicación:

- Capacidad de importar inversiones sobre ciertas criptomonedas a través de diferentes archivos, como pueden ser ficheros JSON o CSV.
- Otorgar al usuario la capacidad de poder personalizar el color de la gráfica que se muestra en la *pantalla de home* y en la *pantalla de detalle de una criptomoneda*.
- Otorgar al usuario la capacidad de poder determinar el tipo de gráfica que se quiere mostrar en la *pantalla de home* y en la *pantalla de detalle de una criptomoneda*. Por ejemplo, de tipo barras, puntos o líneas discontinuas.
- No solo notificar al usuario mediante notificaciones *push*, sino también mediante correos.
- Recomendar al usuario criptomonedas tomando como base las que ya ha añadido.

5. Glosario

- **Minar criptomonedas:** Proceso de invertir capacidad computacional para procesar transacciones, garantizar la seguridad de la red, y conseguir que todos los participantes estén sincronizados.
- **Product Owner:** Profesional encargado de maximizar el valor de una solución generada en una empresa a partir del uso de las metodologías ágiles.
- **Play store:** Tienda de aplicaciones móviles para dispositivos con sistema operativo Android.
- **Pantalla splash:** Primera pantalla visible para el usuario cuando inicia la aplicación.
- **Pantalla modal:** Pantalla de tamaño reducido donde el usuario solo puede actuar dentro de la misma, estando los demás elementos de la aplicación inaccesibles.
- **Vista carrusel:** Mostrar elementos de una lista. Se podrá pasar de un elemento a otro cuando el usuario haga un gesto de deslizamiento.
- **Adobe XD:** Software para crear y compartir interfaces para aplicaciones webs y móviles, con el foco puesto en la experiencia de usuario.
- **Swipe:** Acción de deslizar el dedo sobre la pantalla del teléfono móvil.
- **EditText:** Elemento de la interfaz de usuario de dispositivos Android para introducir y modificar texto.
- **Principio single responsibility:** Incluido como uno de los principios S.O.L.I.D establecidos por Robert C.Martin, y establece que cada clase debe tener una única responsabilidad encapsulada dentro de la misma.
- **Toolbar:** Barra de acciones que aglutina varios elementos, permitiendo al usuario tener una experiencia unificada.
- **API:** Abreviatura de *Application Programming Interfaces*, y se trata del conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones.
- **JSON:** Acrónimo de *Javascript Object Notation*, es un formato de texto sencillo para el intercambio de datos.
- **CSV:** Abreviatura de *Comma Separated Values* y es cualquier archivo de texto en el cual los caracteres están separados por comas, haciendo una especie de tabla en filas. Columnas.

6. Bibliografía

- [1] *Un 48% de la población utiliza apps de finanzas según el estudio de IAB Spain.* [En línea]. [Fecha de consulta: 17/09/2021]. URL: <https://lapublicidad.net/un-48-de-la-poblacion-utiliza-apps-de-finanzas-segun-el-estudio-de-iab-spain/>

- [2] *Measuring global crypto users*. [En línea]. [Fecha de consulta: 18/09/2021]. URL: https://crypto.com/images/202107_DataReport_OnChain_Market_Sizing.pdf
- [3] *¿Es rentable invertir en criptomonedas?* [En línea]. [Fecha de consulta: 19/09/2021]. URL: <https://www.raisin.es/inversion/rentable-invertir-criptomonedas/>
- [4] *Bitcoin, blockchain y criptomonedas, explicado de forma sencilla (y en vídeo)*. [En línea]. [Fecha de consulta: 20/09/2021]. URL: <https://www.xataka.com/criptomonedas/bitcoin-blockchain-y-criptomonedas-explicado-de-forma-sencilla-y-en-video>
- [5] *CoinGecko – Bitcoin & Cryptoconcurrency Price Tracker*. [En línea]. [Fecha de consulta: 21/09/2021]. URL: <https://play.google.com/store/apps/details?id=com.coingecko.coingeckoapp>
- [6] *The Crypto App – Widgets, Alertas, Noticias*. [En línea]. [Fecha de consulta: 22/09/2021]. URL: <https://play.google.com/store/apps/details?id=com.crypter.cryptocyrrency>
- [7] *Crypto News: Noticias, Precios de Bitcoin, Alertas*. [En línea]. [Fecha de consulta: 23/09/2021]. URL: <https://play.google.com/store/apps/details?id=com.cryptonews>
- [8] *CoinMarketCap – rastreador de precios criptográficos*. [En línea]. [Fecha de consulta: 24/09/2021]. URL: <https://play.google.com/store/apps/details?id=com.coinmarketcap.android>
- [9] *Criptomonedas y Wallet: qué son y cuáles son las diferencias*. [En línea]. [Fecha de consulta: 25/09/2021]. URL: <https://www.xataka.com/basics/criptomonedas-wallet-que-cuales-diferencias>
- [10] *Android Statistics (2021)*. [En línea]. [Fecha de consulta: 26/09/2021]. URL: <https://www.businessofapps.com/data/android-statistics/>
- [11] *5 key benefits of Native Mobile App Development*. [En línea]. [Fecha de consulta: 27/09/2021]. URL: <https://clearbridgemobile.com/benefits-of-native-mobile-app-development/>
- [12] *Why Choose Kotlin over Java?* [En línea]. [Fecha de consulta: 28/09/2021]. URL: <https://medium.com/@ChromelInfotech/why-choose-kotlin-over-java-3d9889bdce12>
- [13] Flamarich Zampalo, Jordi (No consta fecha). *Conceptualización* [Material de la asignatura (p. 14). Universitat Oberta de Catalunya.
- [14] *Diseño centrado en el usuario: aprende cómo crear la mejor experiencia*. [En línea]. [Fecha de consulta: 07/10/2021]. URL: <https://blog.hubspot.es/marketing/diseño-centrado-usuario>
- [15] *Guía de arquitecturas de apps*. [En línea]. [Fecha de consulta: 26/10/2021]. URL: <https://developer.android.com/jetpack/guide?hl=es-419>
- [16] *Either en Kotlin*. [En línea]. [Fecha de consulta: 06/12/2021]. URL: <https://www.adictosaltrabajo.com/2019/07/04/either-en-kotlin/>

7. Anexos

7.1. Manual de usuario

En esta sección se explica el funcionamiento de la aplicación con la ayuda de capturas de pantalla.

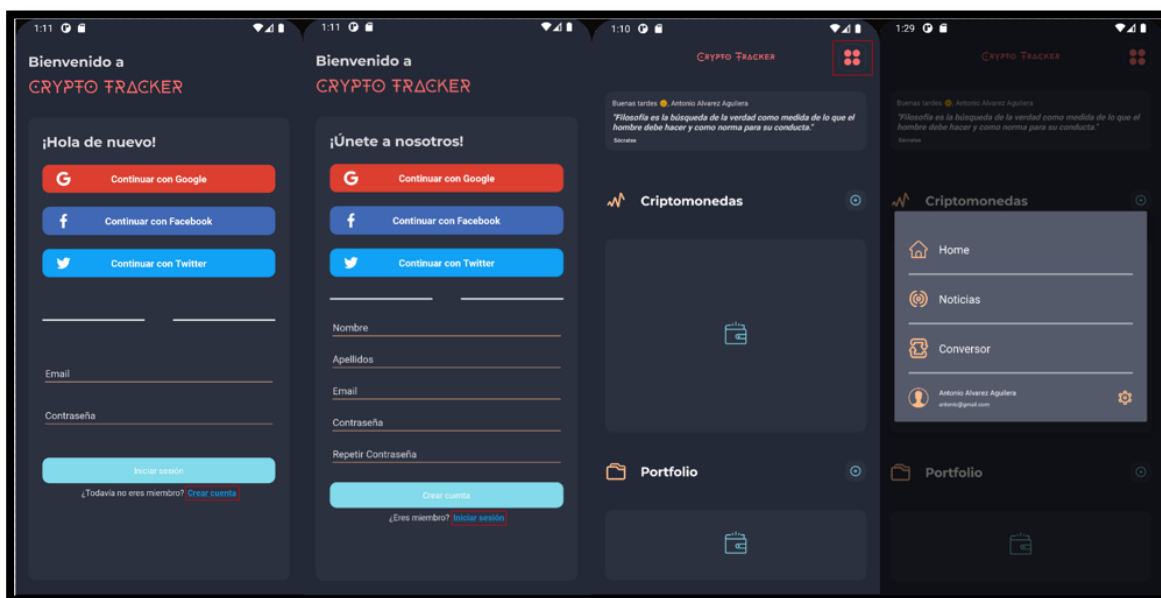


Figura 57. Imágenes manual de usuario

Cuando un usuario abre la aplicación por primera vez, verá la *pantalla de login*. Deberá acceder a la *pantalla de registro* mediante el botón de crear cuenta. Una vez en la *pantalla de registro*, deberá insertar los datos necesarios y crear una cuenta. Automáticamente será redirigido a la *pantalla de login*, donde podrá insertar los datos correspondientes, y acceder a la *pantalla de home*.

Una vez en la *pantalla home*, podrá añadir las criptomonedas que desea seguir a través del botón añadir de la sección de criptomonedas. También podrá añadir criptomonedas en las que tenga alguna inversión a través del botón de añadir de la sección de portfolio.

Todas las pantallas principales tienen acceso al menú de la aplicación, donde se podrá acceder a cada una de ellas.

Cuando se haya añadido alguna criptomoneda en la sección de criptomonedas, se mostrará una lista de elementos que, al pulsar en ellos, se podrá acceder a la *pantalla de detalle de una criptomoneda*. Además, cada uno de ellos podrá eliminarse o actualizarse a través de sus respectivos botones.

En cuanto a las inversiones realizadas, al deslizar en cada uno de los elementos mostrados, se podrá modificar o borrar la inversión realizada.

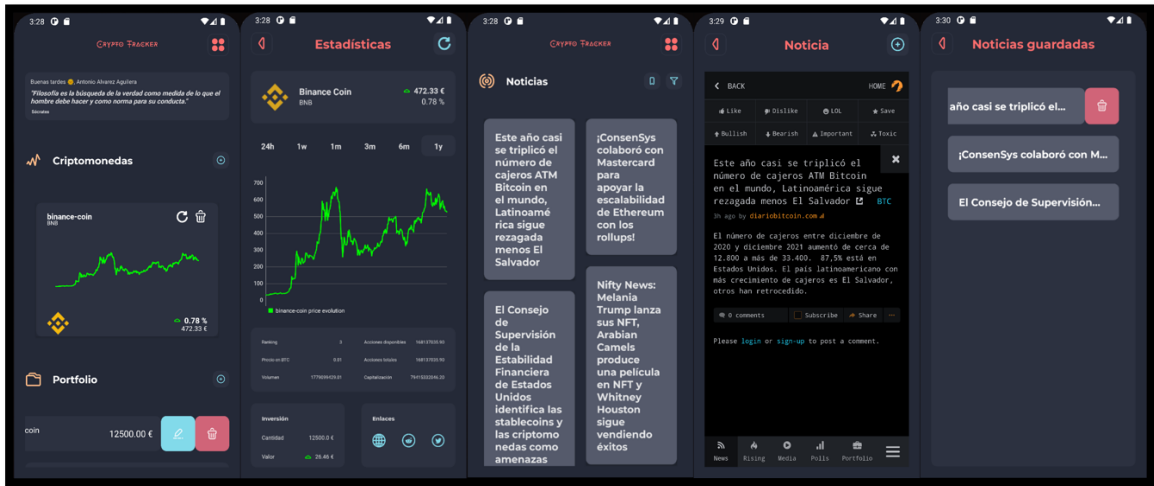


Figura 58. Imágenes manual de usuario (2)

Cuando se acceda a la *pantalla de noticias*, se mostrará una lista de noticias y dos botones: uno para filtrar la lista de noticias y otro para acceder a la *pantalla de noticias guardadas*. Se podrá acceder a la *pantalla de detalle de una noticia* cuando el usuario pulse en alguna noticia de la lista y, cuando el usuario acceda, podrá guardar dicha noticia para poder consultarla más adelante. Toda noticia guardada puede ser borrada.

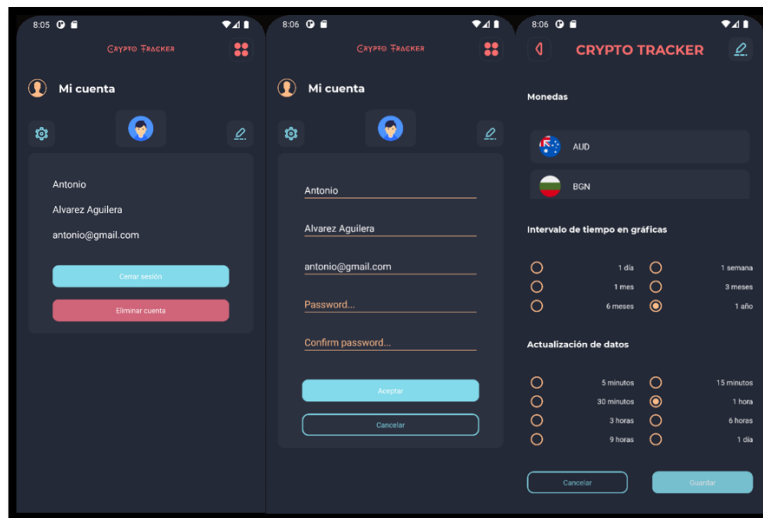


Figura 59. Imágenes manual de usuario (3)

Cuando se acceda a la *pantalla de usuario*, se podrá modificar toda la información introducida en la *pantalla de registro*. Desde esta pantalla, se puede acceder a la *pantalla de preferencias de usuario*, donde se podrá modificar ciertos parámetros que influyen en el resto de la aplicación.