

PROYECTO

Healty Wallet



CICLO FORMATIVO DE GRADO SUPERIOR

Desarrollo de Aplicaciones Multiplataforma



I.E.S. «Venancio Blanco» SALAMANCA

AUTOR Marcos Peque Marbán

1. Licencia

Esta obra está bajo una licencia Reconocimiento-Compartir bajo la misma licencia 3.0 España de Creative Commons. Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by-sa/3.0/es/> o envíe una carta a Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.



2. Resumen

2.1 Contexto y problemática

En la actualidad, gran parte de la población carece de educación financiera básica. Aunque todos manejamos dinero en nuestra vida diaria, conceptos como presupuestación, ahorro o inversión resultan ajenos o demasiado complejos para la mayoría. Esta carencia provoca **dificultades en la planificación económica, escasa capacidad de ahorro y desconocimiento de herramientas de inversión** que podrían mejorar el patrimonio personal.

La educación financiera rara vez se enseña en la escuela, lo que deja a millones de personas sin recursos prácticos para tomar decisiones responsables sobre su dinero.

2.2 Empresa y equipo desarrollador

El proyecto será desarrollado **Healthy Wallet**, una iniciativa nacida en el ámbito de Desarrollo de Aplicaciones Multiplataforma. El equipo está formado por jóvenes con conocimientos en programación, diseño de interfaces, bases de datos y usabilidad, unidos por el objetivo de crear soluciones tecnológicas innovadoras que acerquen las finanzas al público general.

2.3 Exposición clara de la idea

Healthy Wallet es una aplicación multiplataforma diseñada para ser **intuitiva, educativa y accesible**, especialmente para usuarios con **escaso o nulo conocimiento financiero**. La app combina dos ejes principales:

- **Gestión práctica:** registro de ingresos y gastos, balances visuales, avisos de gastos superfluos y herramientas para fijar objetivos de ahorro.
- **Formación progresiva:** introducción al ahorro y la inversión en niveles básico, intermedio y experto, con apoyo de recursos multimedia como videos, citas de economistas y cápsulas educativas.

2.4 Necesidades que cubre o problema que resuelve

Healthy Wallet responde a la falta de educación y herramientas accesibles para gestionar el dinero en la vida diaria. Proporciona al usuario **un asistente financiero digital** que no solo organiza su economía personal, sino que también le enseña cómo mejorarlala y crecer a través del ahorro y la inversión responsable.

2.5 Possible demanda y clientes

El público objetivo es amplio y diverso:

- Jóvenes que comienzan a gestionar su primer salario y quieren aprender a ahorrar.
- Adultos que buscan una herramienta sencilla para controlar ingresos y gastos.
- Personas interesadas en dar sus primeros pasos en la inversión con seguridad y guía.
- Usuarios en general que desean adquirir cultura financiera básica de manera práctica.

2.6 Breve descripción de la solución propuesta

La aplicación Healthy Wallet ofrecerá una experiencia **simple, visual y atractiva**, que permitirá a cualquier persona:

- Llevar un control claro de su economía personal.
- Identificar hábitos de consumo poco eficientes.
- Mejorar su capacidad de ahorro mediante planes personalizados.
- Iniciarse en la inversión a través de un aprendizaje gradual y accesible.
- Acceder a contenido formativo que complemente su práctica diaria.

En definitiva, Healthy Wallet busca ser más que una app: un **acompañante digital que ayuda a gestionar el dinero, ahorrar con inteligencia y aprender a invertir**, acercando la educación financiera a todo el público.

3.Índice de contenido

1.LICENCIA	2
2.RESUMEN.....	3
2.1 CONTEXTO Y PROBLEMÁTICA	3
2.2 EMPRESA Y EQUIPO DESARROLLADOR.....	3
2.3 EXPOSICIÓN CLARA DE LA IDEA	3
2.4 NECESIDADES QUE CUBRE O PROBLEMA QUE RESUELVE.....	3
2.5 POSIBLE DEMANDA Y CLIENTES.....	4
2.6 BREVE DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.....	4
3.ÍNDICE DE CONTENIDO.....	5
4.ÍNDICE DE FIGURAS	7
5.NECESIDADES DEL SECTOR PRODUCTIVO Y DE LA ORGANIZACIÓN DE LA EMPRESA.....	8
5.1 EL SECTOR FINTECH Y LA GESTIÓN FINANCIERA PERSONAL.....	8
5.2 LIMITACIONES DE LAS SOLUCIONES ACTUALES	8
5.3 HEALTHY WALLET: UNA PROPUESTA DIFERENCIADORA	9
5.4 OPORTUNIDAD EN EL MERCADO.....	10
5.5 JUSTIFICACIÓN TÉCNICA Y ECONÓMICA	11
5.6 ANÁLISIS DE LA COMPETENCIA	12
6.DISEÑO DEL PROYECTO	13
6.1. ANÁLISIS DE LA REALIDAD LOCAL Y DEL SECTOR	13
6.2. METODOLOGÍA DE DESARROLLO	13
6.3. DEFINICIÓN O ANÁLISIS DE SISTEMAS O DE REQUISITOS.....	14
6.3.1. OBJETIVOS DEL SISTEMA	14
6.3.2 REQUISITOS FUNCIONALES	17
6.3.3 REQUISITOS DE INFORMACIÓN.....	20
6.3.4 REQUISITOS NO FUNCIONALES	25
6.3.5 DIAGRAMA Y DESCRIPCIÓN DE CASOS DE USO.....	28
6.3.6 JUSTIFICACIÓN TÉCNICA DE DECISIONES (MVVM, ROOM, LIBRERÍAS).....	34
6.3.7 CRITERIOS DE EVALUACIÓN	35
7.1 DISEÑO DEL SISTEMA Y ESPECIFICACIÓN TÉCNICA.....	36
7.1.1 DIAGRAMA DE CLASES	36
7.1.2 DIAGRAMA DE COLABORACIÓN	38
7.1.3 MODELO DE DATOS	41
7.2.1 PROTOTIPOS	42
7.2.2 CÓDIGO	61
8.1 PRUEBAS	68
9.1 IMPLEMENTACIÓN	72
9.1.1 REQUISITOS MÍNIMOS	72
9.1.2 TABLA DE TIEMPO	73
10.1 LICENCIAS	74

10.1.1 LICENCIA DE ANDROID STUDIO Y ANDROID SDK	74
10.1.2 LICENCIA DE ICONOS Y RECURSOS GRÁFICOS	74
10.1.3 LICENCIA DEL REPOSITORIO DEL PROYECTO	74
10.1.4 ACLARACIÓN LEGAL	75
11.1 REFERENCIAS Y BIBLIOGRAFÍA.....	76
12.1 CONCLUSIONES	77
12.1 MEJORAS FUTURAS	77
12.2 CONCLUSIONES PERSONALES SOBRE EL PROYECTO	78

4.Índice de figuras

Fig.1 – Logos de empresas del mismo sector/industria

Fig.2 – Principales fortalezas

Fig.3 - Diagrama de casos de Uso

Fig.4 - Métricas funcionales principales: rapidez en registrar transacciones, navegación eficiente

Fig.5 –Login

Fig.6 –Registro

Fig.7 –Inicio

Fig.8 –Balance Mensual

Fig.9 –Movimientos

Fig.10 –Añadir Movimientos

Fig.11 –Presupuestos

Fig.12 –Crear Presupuesto

Fig.13 –Metas

Fig.14 – Nuevas Metas

Fig.15 – Educación financiera

Fig.16– Estructura del código(DAOs y Java)

Fig.17– Estructura del código(XML)

Fig. 18– Tabla de requisitos mínimos

Fig. 19– Tabla de tiempo

5. Necesidades del sector productivo y de la organización de la empresa

5.1 El sector Fintech y la gestión financiera personal

El sector de las aplicaciones financieras ha crecido de forma notable en los últimos años, con propuestas que buscan ayudar a los usuarios a controlar sus gastos, acceder a servicios bancarios online o invertir de forma digital. Empresas como Fintonic, Revolut, N26 o Mint han desarrollado herramientas muy potentes, pero en la mayoría de los casos estas aplicaciones **están ligadas directamente a entidades bancarias o brokers**, lo que condiciona su enfoque y limita su independencia.



Fig.1 – Logos de empresas del mismo sector/industria

5.2 Limitaciones de las soluciones actuales

Las aplicaciones financieras más conocidas cumplen bien su función de **mostrar movimientos, clasificar gastos o permitir pequeñas inversiones**, pero presentan varios problemas:

- Están pensadas principalmente como **extensiones digitales de bancos o intermediarios financieros**, por lo que muchas veces sirven como puerta de entrada a productos de inversión propios.
- Se centran más en la **operativa bancaria** que en la **formación del usuario**.

Aunque ofrecen gráficos y balances, no explican cómo interpretar esos datos ni cómo mejorar realmente la salud financiera personal.

- Muchas requieren **pagos por suscripción o limitan funciones avanzadas** a versiones premium, lo que reduce la accesibilidad para el usuario medio.

5.3 Healthy Wallet: una propuesta diferenciadora

La propuesta de **Healthy Wallet** rompe con este modelo. No está asociada a ninguna entidad bancaria ni broker, lo que le permite mantener una independencia total y centrarse en lo más importante: **el usuario y su aprendizaje financiero**.

Además, **Healthy Wallet es una aplicación gratuita**, lo que garantiza que cualquier persona pueda acceder a sus funcionalidades sin barreras económicas, fomentando la inclusión financiera.

Sus principales aportaciones son:

- **Balance financiero inteligente**: registro automático/manual de movimientos, identificación de gastos superfluos y objetivos personalizados de ahorro.
- **Educación progresiva**: módulos básicos, intermedios y avanzados de inversión (ETFs, fondos monetarios, etc.), con apoyo de videos, citas de economistas y recursos didácticos.
- **Independencia de entidades financieras**: la app no actúa como intermediario ni comercializa productos propios; su valor radica en la **transparencia y neutralidad**.
- **Acceso gratuito**: la app ofrece todas sus funciones principales sin coste, a diferencia de muchas alternativas que imponen barreras económicas.



Fig.2 – Principales fortalezas

5.4 Oportunidad en el mercado

A día de hoy no existe una aplicación que combine de manera efectiva **gestión financiera + formación independiente + gratuidad**.

Las apps actuales ofrecen una visión de gastos y, en algunos casos, herramientas de inversión, pero siempre ligadas a una entidad financiera o bajo un modelo de suscripción.

Healthy Wallet ocupa este hueco: es una aplicación **formativa, accesible, gratuita e independiente**, que convierte la gestión del dinero en una experiencia educativa y práctica para cualquier persona, sin presiones comerciales externas.

Plataforma

Healthy Wallet se desarrolla como una **aplicación móvil**, lo que responde a la forma en que la mayoría de las personas gestionan hoy su dinero: desde el teléfono, en cualquier momento y lugar.

Este enfoque tiene varios beneficios:

- **Accesibilidad inmediata:** el usuario lleva siempre consigo la herramienta, lo que facilita registrar gastos, consultar balances o avanzar en módulos educativos en tiempo real.
- **Mayor adopción:** los móviles son el dispositivo más utilizado en todo el mundo, lo que permite llegar a una base amplia y diversa de usuarios.
- **Comodidad y hábito:** revisar la app se convierte en parte de la rutina diaria, similar a otras aplicaciones de uso frecuente (mensajería, redes sociales, banca online).
- **Actualizaciones ágiles:** al estar en tiendas de aplicaciones (App Store y Google Play), las mejoras y nuevos contenidos llegan rápidamente a todos los usuarios.

Referencias y datos de mercado

Según datos de **Statista (2024)**, el 65% de los usuarios españoles entre 18 y 35 años utiliza alguna aplicación fintech para gestionar sus finanzas personales, y el sector crece a un ritmo anual del **17,4%**.

España se sitúa como el **tercer país europeo con mayor adopción de apps financieras**, solo por detrás de Reino Unido y Alemania.

Este escenario evidencia la **oportunidad de mercado** para propuestas educativas e independientes como *Healthy Wallet*, que combinan control financiero y formación económica.

5.5 Justificación técnica y económica

Desde un punto de vista **técnico**, la aplicación *Healthy Wallet* es completamente viable gracias al uso de herramientas de desarrollo multiplataforma que permiten reducir costes y tiempo de implementación.

El proyecto se desarrollará con **Android Studio** y lenguaje **Java** para la versión Android, mientras que la interfaz se definirá con **XML**.

Para el almacenamiento de datos locales se utilizará **Room**, el ORM oficial de Android basado en **SQLite**, lo que permite un acceso más seguro, estructurado y eficiente a la base de datos sin necesidad de conexión.

El control de versiones y colaboración se gestionará con **GitHub**, permitiendo un trabajo estructurado y seguimiento continuo del código.

Desde el punto de vista **económico**, *Healthy Wallet* presenta una alta viabilidad, ya que:

- No requiere servidores propios ni infraestructura compleja.
- El mantenimiento se reduce a actualizaciones periódicas y soporte técnico básico.
- El modelo de distribución es gratuito, lo que aumenta el alcance sin generar costes de licencias.
- Su desarrollo se puede realizar con herramientas open-source y recursos formativos gratuitos.

En comparación con otras aplicaciones del sector, el coste de desarrollo estimado se mantiene **entre un 30% y 50% menor** al de soluciones fintech propietarias, lo que refuerza su factibilidad como proyecto educativo y de innovación.

5.6 Análisis de la competencia

Aplicación	Tipo	Vinculación bancaria	Funciones principales	Modelo económico	Diferenciador de Healthy Wallet
 fintonic	Gestión financiera	Sí	Control de gastos, alertas, préstamos	Gratis con publicidad	Healthy Wallet no vende productos financieros.
 Revolut	Banca digital	Sí	Pagos, inversión, divisas, seguros	Freemium	HW se centra en educación, no en operaciones bancarias.
 N26	Neobanco	Sí	Cuentas corrientes y tarjetas	Freemium / Premium	HW es totalmente independiente y formativa.
 mint	Gestión personal	No	Presupuestos, recordatorios, informes	Gratis	HW añade un módulo educativo y enfoque local.
 Healthy Wallet	Educación + gestión	No	Registro, presupuestos, metas, formación	100% gratuita	Foco educativo, accesibilidad total e independencia.

6. Diseño del proyecto

6.1. Análisis de la realidad local y del sector

El sector de las aplicaciones móviles orientadas a la **gestión financiera personal** ha experimentado un notable crecimiento en los últimos años, impulsado por la digitalización de los servicios bancarios y el aumento del interés de los usuarios por controlar sus finanzas desde el móvil.

En España destacan aplicaciones como **Fintonic, Revolut, N26 o BBVA**, que ofrecen herramientas de control de gastos, planificación de presupuestos o inversión digital. Sin embargo, la mayoría de estas soluciones **están ligadas a entidades financieras** o funcionan bajo un modelo de **suscripción**, lo que limita su independencia y accesibilidad.

En el entorno local, **Salamanca** cuenta con un creciente número de empresas y profesionales dedicados al desarrollo de software y aplicaciones móviles. Aun así, se detecta una **carenza de herramientas que promuevan la educación financiera** y la autonomía económica, especialmente entre jóvenes y personas que comienzan su vida laboral.

El proyecto **Healthy Wallet** nace como una respuesta a esta necesidad, con el objetivo de crear una aplicación **educativa y gratuita**, que ayude a los usuarios a comprender, planificar y mejorar su salud financiera personal mediante una interfaz intuitiva y atractiva.

6.2. Metodología de desarrollo

Para el desarrollo del proyecto se utilizará la **metodología ágil Scrum**, ampliamente reconocida en el ámbito del desarrollo de software por su enfoque iterativo, colaborativo y flexible.

Scrum permite dividir el trabajo en **sprints** o ciclos cortos de entre dos y tres semanas. En cada sprint se planifica, desarrolla, prueba y entrega una parte funcional del producto, lo que facilita la detección temprana de errores y la adaptación a cambios o mejoras.

Roles principales en el equipo Scrum:

- **Product Owner:** encargado de definir las funcionalidades y priorizar tareas.
- **Scrum Master:** responsable de coordinar el proceso y eliminar impedimentos.
- **Equipo de desarrollo:** formado por los desarrolladores y diseñadores que implementan las funcionalidades.

Esta metodología fomenta una comunicación constante, permite mejorar el producto de forma progresiva y garantiza que el resultado final cumpla con los requisitos planteados en tiempo y calidad.

6.3. Definición o análisis de sistemas o de requisitos

6.3.1. Objetivos del sistema

A continuación, se detallan los **objetivos principales del proyecto Healty Wallet**, organizados mediante códigos identificativos para facilitar su referencia durante las fases de análisis, diseño e implementación.

OBJ-01	Registro de transacciones
Descripción	Permitir al usuario registrar manualmente sus ingresos y gastos, clasificándolos por categorías, fechas y métodos de pago.
Importancia	Alta
Estabilidad	Alta
Comentarios	Es la funcionalidad base del sistema; necesaria para generar cualquier análisis posterior.

OBJ-02	Visualización de estadísticas
Descripción	Generar informes y representaciones que muestren el flujo de dinero, los balances y la evolución del ahorro.
Importancia	Alta
Estabilidad	Media
Comentarios	Aporta valor visual al sistema, facilita la comprensión de la situación financiera.

OBJ-03	Presupuestos personalizados
Descripción	Permitir la creación y gestión de presupuestos por categorías de gasto, ajustables según las metas del usuario.
Importancia	Media
Estabilidad	Alta
Comentarios	Mejora la organización financiera del usuario y permite un mayor control sobre sus gastos.

OBJ-04	Análisis de hábitos financieros
Descripción	Analizar el comportamiento de gasto y emitir alertas o consejos cuando se detecten desviaciones.
Importancia	Alta
Estabilidad	Media
Comentarios	Fomenta la educación financiera y el uso continuo de la aplicación.

OBJ-05	Gestión de metas de ahorro
Descripción	Permitir al usuario definir objetivos de ahorro e inversión y mostrar su progreso mediante indicadores visuales.
Importancia	Alta
Estabilidad	Alta
Comentarios	Promueve la constancia y la planificación financiera a largo plazo.

OBJ-06	Módulo educativo financiero
Descripción	Ofrecer contenidos formativos sobre finanzas personales, ahorro, inversión y conceptos básicos económicos.
Importancia	Media
Estabilidad	Baja
Comentarios	Añade el componente educativo diferenciador de la aplicación.

OBJ-07	Accesibilidad y gratuitidad
Descripción	Garantizar que todas las funciones principales sean accesibles de manera gratuita y sin publicidad invasiva.
Importancia	Alta
Estabilidad	Alta
Comentarios	Mantiene la filosofía del proyecto: inclusión y educación financiera universal.

OBJ-08	Seguridad y privacidad
Descripción	Aplicar medidas de protección de datos mediante verificación con usuario propio
Importancia	Alta
Estabilidad	Alta
Comentarios	Refuerza la confianza del usuario

6.3.2 Requisitos funcionales

RF-01. Registro de transacciones	
OBJ Asociado	OBJ-01
Definición	El sistema debe permitir al usuario registrar manualmente sus ingresos y gastos, asignándolos a una categoría y método de pago.
Importancia	Alta
Comentarios	Es la base de la aplicación. Permite generar los datos necesarios para todos los análisis posteriores.

RF-02. Visualización de gráficos	
OBJ Asociado	OBJ-02
Definición	El sistema debe generar gráficos e informes que muestren el flujo de dinero, balance mensual y evolución del ahorro.
Importancia	Alta
Comentarios	Facilita la comprensión del estado financiero del usuario mediante información visual.

RF-03. Creación de presupuestos	
OBJ Asociado	OBJ-03
Definición	El sistema debe permitir la creación de presupuestos personalizados por categoría, definiendo límites de gasto.
Importancia	Media
Comentarios	Ayuda al usuario a planificar sus gastos y a evitar excesos en determinadas áreas.

RF-04. Análisis de hábitos financieros	
OBJ Asociado	OBJ-04
Definición	El sistema debe analizar los hábitos de consumo y mostrar sugerencias personalizadas para mejorar la gestión del dinero.
Importancia	Alta
Comentarios	Fomenta la educación financiera y refuerza el valor didáctico del proyecto.

RF-05. Gestión de metas de ahorro	
OBJ Asociado	OBJ-05
Definición	El sistema debe permitir establecer objetivos de ahorro o inversión, mostrando el progreso mediante indicadores visuales.
Importancia	Alta
Comentarios	Promueve la constancia y la planificación financiera del usuario.

RF-06. Módulo educativo financiero	
OBJ Asociado	OBJ-06
Definición	El sistema debe incluir un módulo educativo que presente contenidos progresivos sobre ahorro, inversión y finanzas personales.
Importancia	Media
Comentarios	Aporta el enfoque formativo que diferencia a <i>Healthy Wallet</i> de otras apps del sector.

RF-07. Sistema de accesibilidad gratuita	
OBJ Asociado	OBJ-07
Definición	El sistema debe permitir el uso y la creación de usuario sin necesidad de un pago o suscripción
Importancia	Media
Comentarios	Favorece la interacción continua con la aplicación, ayuda a promover su uso.

6.3.3 Requisitos de Información

RI-01. Datos de usuario	
OBJ Asociados	OBJ-07, OBJ-08
RF Asociados	RF-01, RF-06, RF-07
Descripción	Información necesaria para identificar al usuario, su configuración de uso y preferencias de privacidad.
Datos concretos	<ul style="list-style-type: none"> - ID de usuario - Nombre/alias - Email -contraseña
Tiempo de Vida	Indefinido (persistente mientras exista la cuenta/local)
Importancia	Alta
Estabilidad	Alta
Comentarios	Si no hay registro en nube, se guarda localmente. Preparado para futura sincronización.



RI-02. Transacciones	
OBJ Asociados	OBJ-01, OBJ-02, OBJ-04
RF Asociados	RF-01, RF-02, RF-04
Descripción	Registro detallado de ingresos y gastos para análisis y recomendaciones.
Datos concretos	<ul style="list-style-type: none"> - ID transacción, fecha/hora - Tipo (ingreso/gasto) - Categoría - Importe - Notas/etiquetas
Tiempo de Vida	Indefinido (histórico consultable)
Importancia	Alta
Estabilidad	Alta
Comentarios	se debe permitir edición y borrado seguro

RI-03. Presupuestos	
OBJ Asociados	OBJ-03, OBJ-04
RF Asociados	RF-03, RF-04
Descripción	Información de límites de gasto por categoría/periodo y su estado de ejecución.
Datos concretos	<ul style="list-style-type: none"> - ID presupuesto - Categoría afectada - Periodicidad (mensual/semanal/personalizada) - Límite asignado - Monto gastado acumulado - Porcentaje de uso
Tiempo de Vida	Por periodos; histórico conservado
Importancia	Media
Estabilidad	Alta
Comentarios	Imprescindible para alertas preventivas y recomendaciones de ajuste.

RI-04. Metas de ahorro	
OBJ Asociado	OBJ-06
RF Asociado	RF-06
Descripción	Datos de objetivos de ahorro/inversión y seguimiento del progreso.
Datos concretos	<ul style="list-style-type: none"> - ID meta - Nombre/objetivo (p. ej., “Fondo de emergencia”) - Importe objetivo - Fecha objetivo (opcional) - Aportaciones realizadas (histórico) - Progreso (% / importe) - Estado (en curso, lograda, pausada)
Tiempo de Vida	Hasta cierre de la meta
Importancia	Alta
Estabilidad	Alta
Comentarios	Relación opcional con reglas de traspaso automático desde presupuesto sobrante.

RI-05. Contenidos educativos	
OBJ Asociado	OBJ-06
RF Asociado	RF-06, RF-07
Descripción	cápsulas formativas y el progreso del usuario.
Datos concretos	<ul style="list-style-type: none"> - ID contenido - Título - Tipo (texto, vídeo) - Etiquetas - Estado del usuario (no iniciado / en curso / completado)
Tiempo de Vida	Indefinido (catálogo) + progreso por usuario
Importancia	Media
Estabilidad	Baja (contenido puede actualizarse)
Comentarios	Permite recomendaciones personalizadas y rutas de aprendizaje progresivas.

◇ Modelo Entidad–Relación

| Modelo inicial

- Una Transacción pertenece a una Categoría.
- Un Presupuesto controla los gastos asignados a una categoría.
- Una Meta de Ahorro se alimenta de transacciones tipo “ahorro” o “transferencias internas”.
- El usuario puede consultar Contenidos educativos según nivel.

6.3.4 Requisitos no funcionales

RNF-01. Interfaz adaptable y accesible	
Descripción	La aplicación deberá ofrecer una interfaz intuitiva, responsive y adaptada a distintos tamaños de pantalla (móviles de distintas resoluciones y tabletas), con navegación clara y elementos táctiles adecuados.
Importancia	Alta
Estabilidad	Alta
Comentarios	Implementar layouts adaptativos (ConstraintLayout), recursos de dimensiones, y pruebas en emuladores/dispositivos. Seguir pautas de Material Design.

RNF-02. Seguridad y protección de datos	
Descripción	Los datos sensibles deben almacenarse de forma segura (cifrado local para datos críticos), gestionarse con control de acceso y respetar la normativa de protección de datos (GDPR)
Importancia	Alta
Estabilidad	Alta
Comentarios	Usar EncryptedSharedPreferences / SQLCipher o Room con cifrado. Documentar el tratamiento de datos y permisos solicitados.

RNF-03. Funcionamiento sin conexión (offline-first)	
Descripción	La aplicación debe permitir la mayoría de operaciones (registro de transacciones, consulta de balances, progreso de metas) sin conexión a Internet, sincronizando cambios cuando haya conectividad
Importancia	Alta
Estabilidad	Alta
Comentarios	Diseñar con Room + patrón Repository; gestionar colas de sincronización y resolución de conflictos para versiones futuras con nube.

RNF-04. Rendimiento y tiempo de respuesta	
Descripción	Las operaciones habituales (listar transacciones, guardar entrada) deben ejecutarse en <1s en dispositivos de gama media; la app debe consumir recursos de forma eficiente (memoria/CPU/batería).
Importancia	Alta
Estabilidad	Alta
Comentarios	Uso de consultas optimizadas (DAO), cargas asíncronas (ViewModel + LiveData/Coroutines), paginación para listas largas. Medir consumo y optimizar.

RNF-05. Mantenibilidad y modularidad	
Descripción	El código debe organizarse de forma modular (capas: UI, domain, data) y documentado para facilitar mantenimiento, pruebas y posible ampliación funcional.
Importancia	Media
Estabilidad	Alta
Comentarios	Seguir MVVM + Repository; usar ViewModels, DAOs y separación de responsabilidades. Añadir README y comentarios clave.

RNF-06. Accesibilidad y localización	
Descripción	La app debe ser accesible (etiquetas para lectores de pantalla, contraste suficiente, tamaños de texto escalables) y preparada para localización
Importancia	Media
Estabilidad	Media
Comentarios	Se implementarán gradualmente mecanismos de accesibilidad, incluyendo descripciones para lectores de pantalla, revisión del contraste cromático en las tarjetas generadas, y ajuste escalable del tamaño del texto. Asimismo, se organizarán los recursos lingüísticos para facilitar la localización de la aplicación a otros idiomas.

Accesibilidad requerida:

- Contraste mínimo 4.5:1 (WCAG AA).
- Navegación completa
- Botones con área mínima de 48dp.
- Descripciones alternativas para gráficos:
- Gasto mensual: 45% alimentación, 28% ocio, 12% transporte

6.3.5 Diagrama y descripción de casos de uso

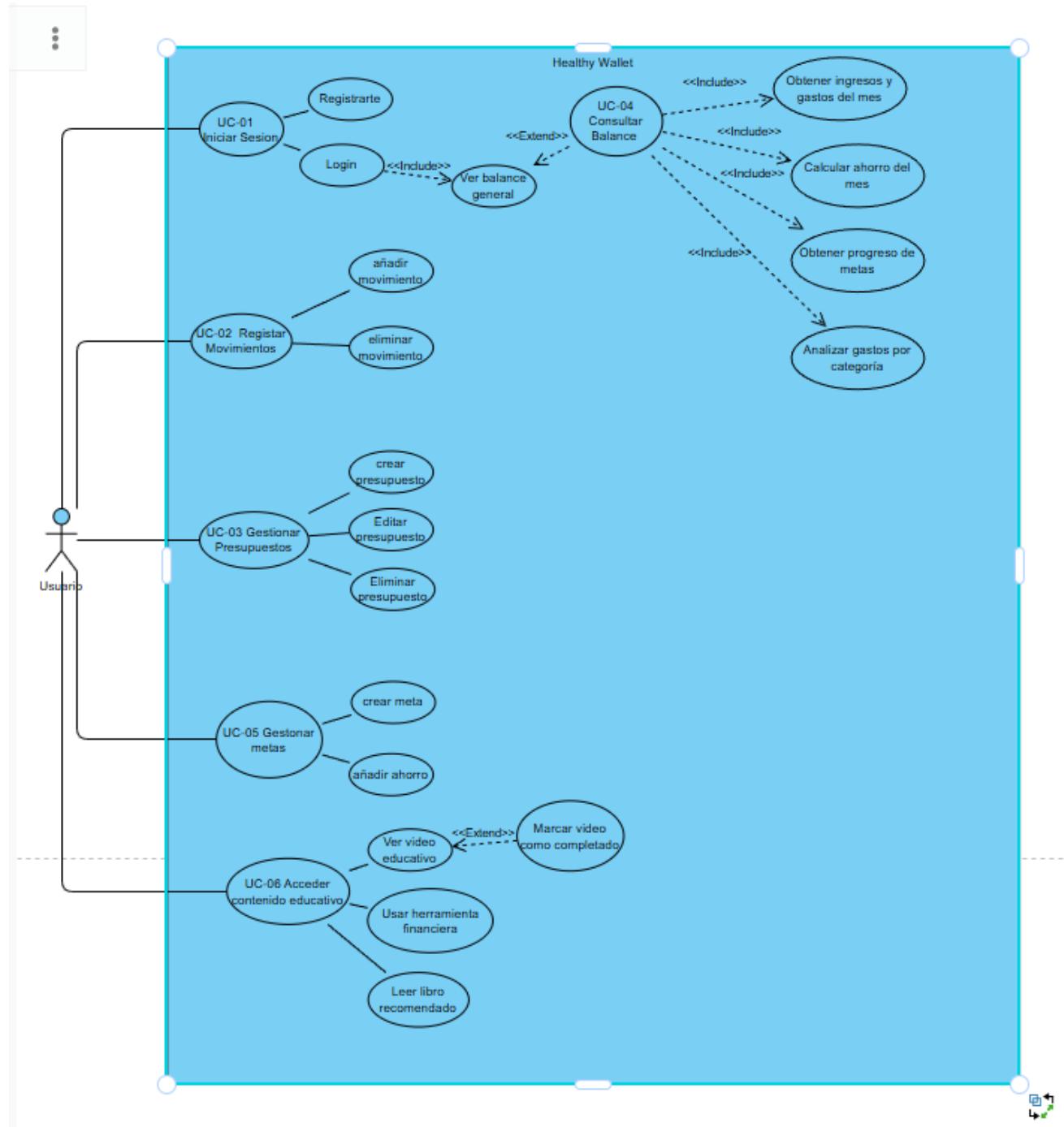


Fig.3 - Diagrama de casos de Uso



CASO DE USO UC-01: Registrar ingresos y gastos		
Actor Principal	Usuario	
Precondiciones	La aplicación está abierta y el usuario dispone de categorías creadas o por defecto.	
Escenario principal de éxito		
DESCRIPCIÓN	Paso	Acción
	1	El usuario selecciona la opción “Registrar movimiento”.
	2	El sistema solicita los datos de la transacción: tipo, importe, categoría, fecha y método de pago (opcional).
	3A	Si el usuario selecciona <u>ingreso</u> , introduce los datos y confirma.
	3B	Si selecciona gasto, introduce los datos y confirma.
	4	El sistema guarda la transacción y actualiza el balance general.
	5	El usuario puede visualizar el movimiento registrado y, si desea, editarlo o eliminarlo.
Extensiones	Paso	Acción
	2a	Si el importe introducido es ≤ 0 , el sistema muestra un mensaje de error y no permite continuar.
	2b	Si no se elige categoría, el sistema obliga a seleccionar o crear una nueva.
	3b	Si el gasto se marca como recurrente, se ofrece crear un recordatorio (UC-04).
Requisitos especiales	Funcionales RF-01, RF-02, RF-03 Información RI-01, RI-02 No funcionales RNF-02, RNF-03, RNF-03, RNF-05	

CASO DE USO UC-02: Consultar balance mensual		
Actor Principal	Usuario	
Precondiciones	El usuario tiene registrados ingresos o gastos en el sistema.	
Escenario principal de éxito		
DESCRIPCIÓN	Paso	Acción
	1	El usuario accede a la tarjeta “Balance general” desde el menú principal.
	2	El sistema muestra el balance general del periodo actual y los filtros disponibles.
	3A	El usuario selecciona Ver balance mensual para visualizar ingresos, gastos y resultado neto.
	3B	El usuario entra para observar la distribución por categorías y la evolución temporal.
	4	El sistema actualiza la vista con los datos correspondientes y muestra los totales del mes , porcentaje de gastos superfluos.
Extensiones	Paso	Acción
	3a	Si el periodo seleccionado no contiene datos, el sistema restablece el periodo por defecto (mes actual).
Requisitos especiales	Funcionales RF-02 Información RI-02 No funcionales RNF-03, RNF-05, RNF-02	

CASO DE USO UC-03: Gestionar Presupuestos		
Actor Principal	Usuario	
Precondiciones	El usuario tiene al menos una categoría creada y ha registrado ingresos o gastos previamente.	
Escenario principal de éxito		
DESCRIPCIÓN	Paso	Acción
	1	El usuario accede a la opción “Presupuestos” desde el menú principal.
	2	El sistema muestra los presupuestos existentes, indicando categoría, límite establecido, gasto actual y porcentaje de uso.
	3A	El usuario selecciona “Crear presupuesto”, define categoría, límite de gasto, periodo (mensual/semanal) y umbral de alerta.
	3B	El usuario selecciona “Editar presupuesto” y modifica los valores existentes.
	3C	El usuario selecciona “Eliminar presupuesto” y confirma la eliminación.
	4	El sistema valida los datos introducidos y guarda los cambios realizados.
Requisitos especiales	Funcionales RF-03, RF-05 Información RI-03, RI-02 No funcionales RNF-03, RNF-04, RNF-06	

CASO DE USO UC-4: Gestionar metas de ahorro		
Actor Principal	Usuario	
Precondiciones	El usuario ha iniciado sesión. Existe al menos una meta creada o el usuario accede para crear una nueva.	
Escenario principal de éxito		
DESCRIPCIÓN	Paso	Acción
	1	El usuario accede a la sección “Metas” desde la barra de navegación inferior.
	2	El sistema muestra la lista de metas activas, con su progreso y estado actual.
	3	El usuario selecciona una meta existente o presiona el botón para agregar una nueva.
	4A	El usuario introduce nombre, cantidad objetivo, fecha límite (opcional) y categoría asociada. Luego confirma.
	4b	El usuario selecciona una meta y puede modificar sus datos principales.
	4c	El usuario pulsa “Añadir ahorro”, introduce un monto y el sistema actualiza el progreso.
	5	El sistema valida los datos, guarda cambios en la base de datos local (Room) y actualiza la interfaz
	6	El usuario visualiza su avance actualizado hacia la meta.
Requisitos especiales	Funcionales RF-04, RF-05, RF-08 Información RI-04, RI-02 No funcionales RNF-02, RNF-03, RNF-04, RNF-06	

CASO DE USO UC-05: Acceder a contenidos educativos		
Actor Principal	Usuario	
Precondiciones	El usuario dispone de conexión a Internet o ha descargado previamente contenido educativo.	
Escenario principal de éxito		
DESCRIPCIÓN	Paso	Acción
	1	El usuario accede a la opción “Educación financiera” desde el menú principal.
	2	El sistema muestra los temas disponibles clasificados por nivel o categoría (videos, libros,herramientas).
	3A	El usuario selecciona un contenido educativo (píldora, artículo o video) y lo visualiza.
	3B	El usuario puede marcar el contenido como “Completado”.
	4	El sistema registra el progreso del usuario y recomienda nuevos contenidos relacionados.
Requisitos especiales	Funcionales RF-02 Información RI-02 No funcionales RNF-04, RNF-06, RNF-03	

6.3.6 Justificación técnica de decisiones (MVVM, Room, librerías)

¿Por qué MVVM?

MVVM proporciona una arquitectura moderna y mantenible para Android. Permite separar claramente la interfaz de usuario de la lógica de presentación, facilita las pruebas unitarias del ViewModel sin depender de la UI, evita Activities sobrecargadas, conserva los estados ante cambios de configuración y se integra de forma nativa con componentes de Jetpack como LiveData, StateFlow y Room. Además, es la arquitectura recomendada por Google para el desarrollo de aplicaciones escalables y orientadas a buenas prácticas.

Patrón	Pros	Contras
MVC	Simple	Controladores muy cargados
MVP	Presentador testeable	Mucho boilerplate
MVVM	Reutilizable, limpio	Requiere LiveData/Flow

SQLite vs Room vs SharedPreferences:

Tecnología	Ventajas	Inconvenientes
SharedPreferences	Simple	No vale para datos complejos
SQLite crudo	Control total	Mucho código SQL
Room	DAO, migraciones, validación, LiveData	Curva de aprendizaje

Justificación: Room reduce errores, facilita consultas y permite observar la BD en tiempo real.

Librerías seleccionadas

Librería	Uso	Justificación
Room	Persistencia	Seguridad y eficiencia
ViewModel + LiveData/Flow	Estado y reactividad	UI estable



6.3.7 Criterios de evaluación

Métricas funcionales

- Registrar una transacción en ≤ 5 toques.
- Carga del dashboard en < 2 segundos.
- Actualización del gráfico en $\leq 0,5$ s tras añadir una operación.
- Crear un presupuesto nuevo en < 10 segundos.

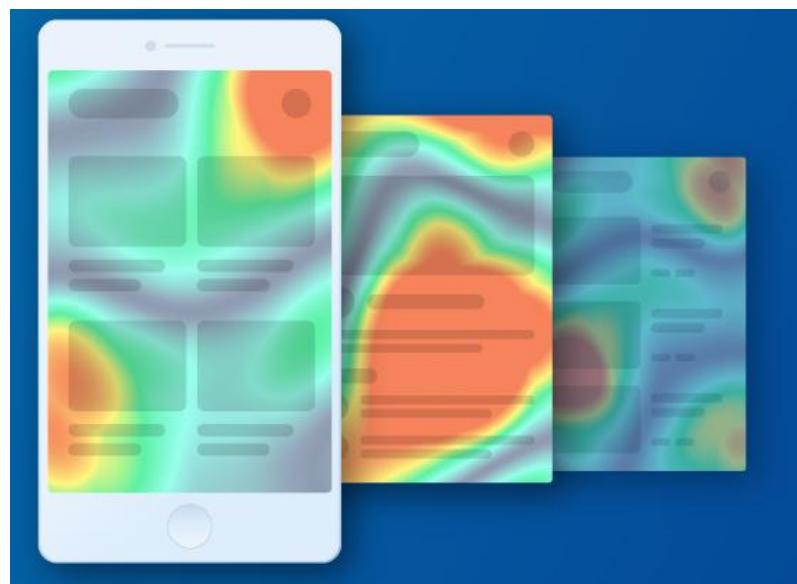


Fig.4 - Métricas funcionales principales: rapidez en registrar transacciones, navegación eficiente

Métricas técnicas

- Uso de memoria < 150 MB.
- Consumo CPU $< 20\%$ en dispositivos medios.
- Caída de la app $< 0,1\%$ mensual.

Métricas educativas

- El 80% de los usuarios comprende el significado de “balance mensual” tras 3 usos.

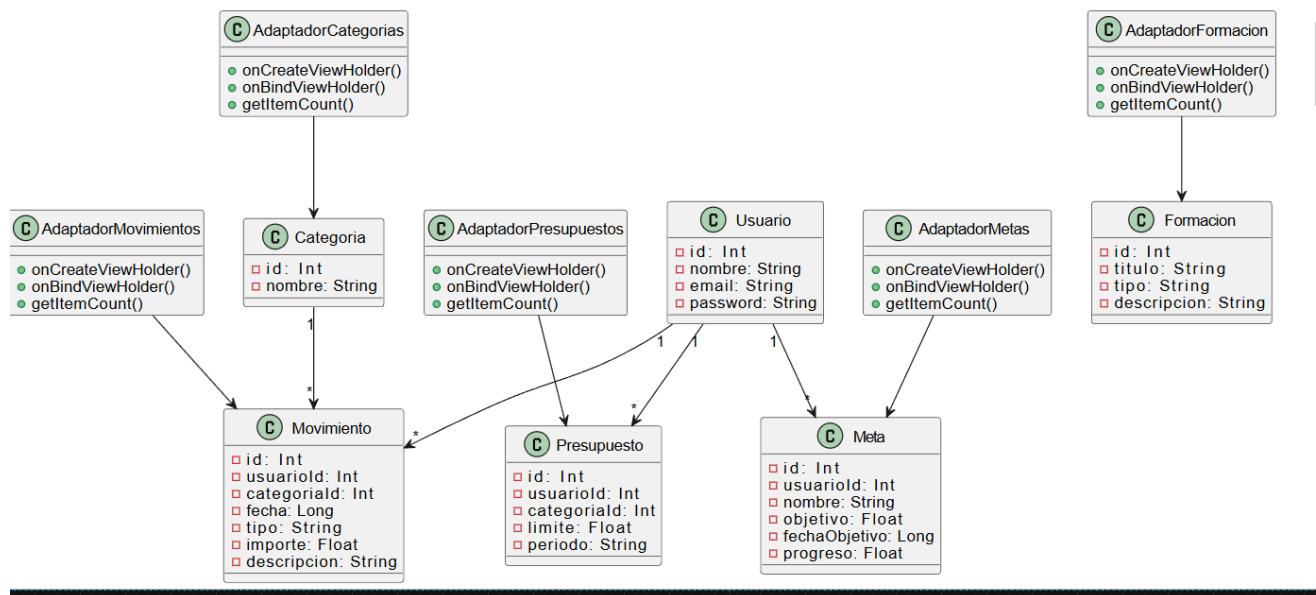
7.1 Diseño del Sistema y Especificación Técnica

En este apartado definiremos el diseño técnico completo de la aplicación *Healthy Wallet*.

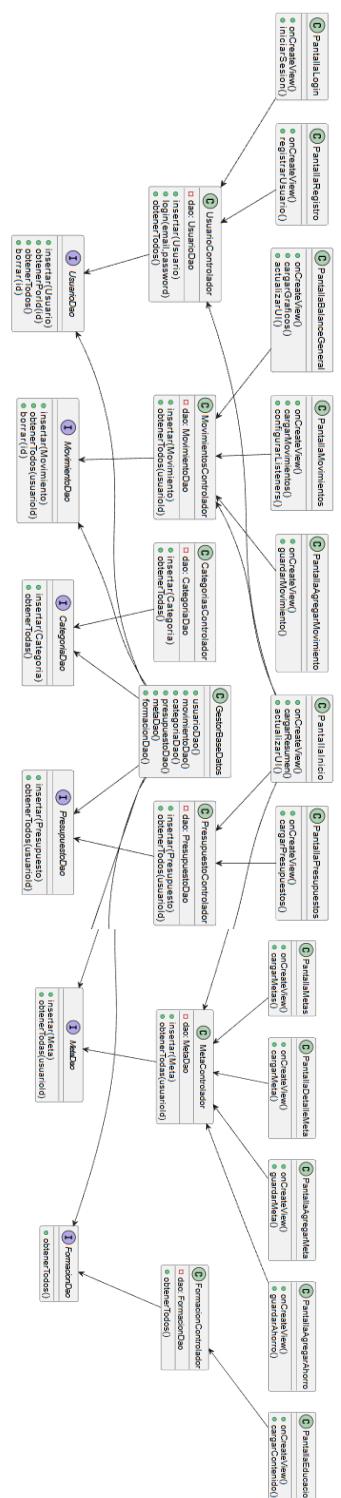
- Para ello se desarrollan los distintos diagramas UML necesarios para describir la estructura interna del sistema, su funcionamiento y las interacciones entre componentes.
- Se incluyen el **diagrama de clases**, que define la arquitectura lógica y la relación entre entidades, controladores y elementos de la interfaz; el **modelo de datos**, que describe la organización de la información en la base de datos; y los **diagramas de colaboración**, que muestran el flujo de mensajes entre los objetos durante los procesos más relevantes.
- Finalmente, se incorporan los **prototipos de interfaz** y una selección de **fragmentos de código representativos** que evidencian la implementación real de los componentes analizados.

7.1.1 Diagrama de Clases

Relaciones adaptadores:



Relaciones Pantallas – DAOS – Controladores



7.1.2 Diagrama de colaboración

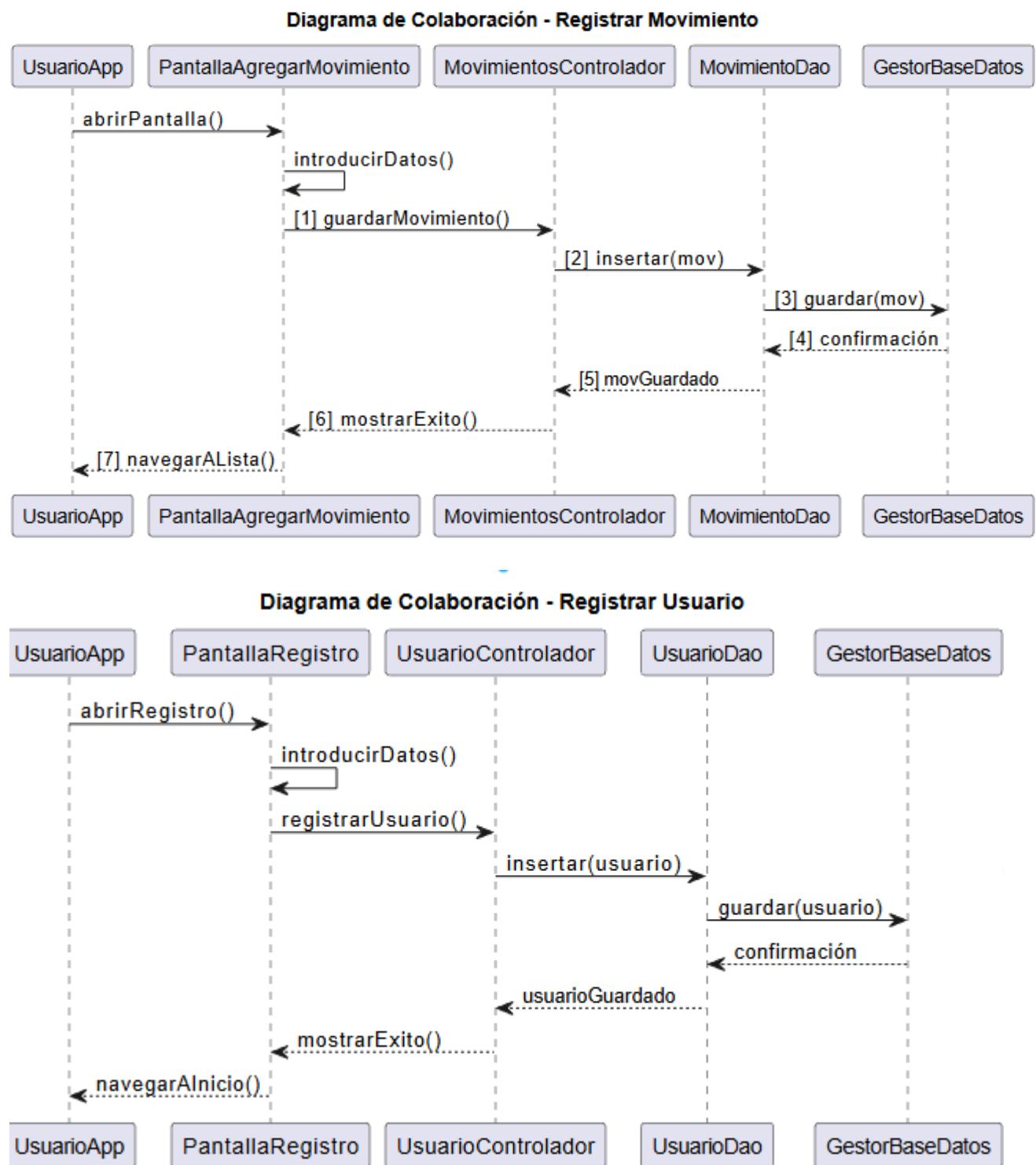
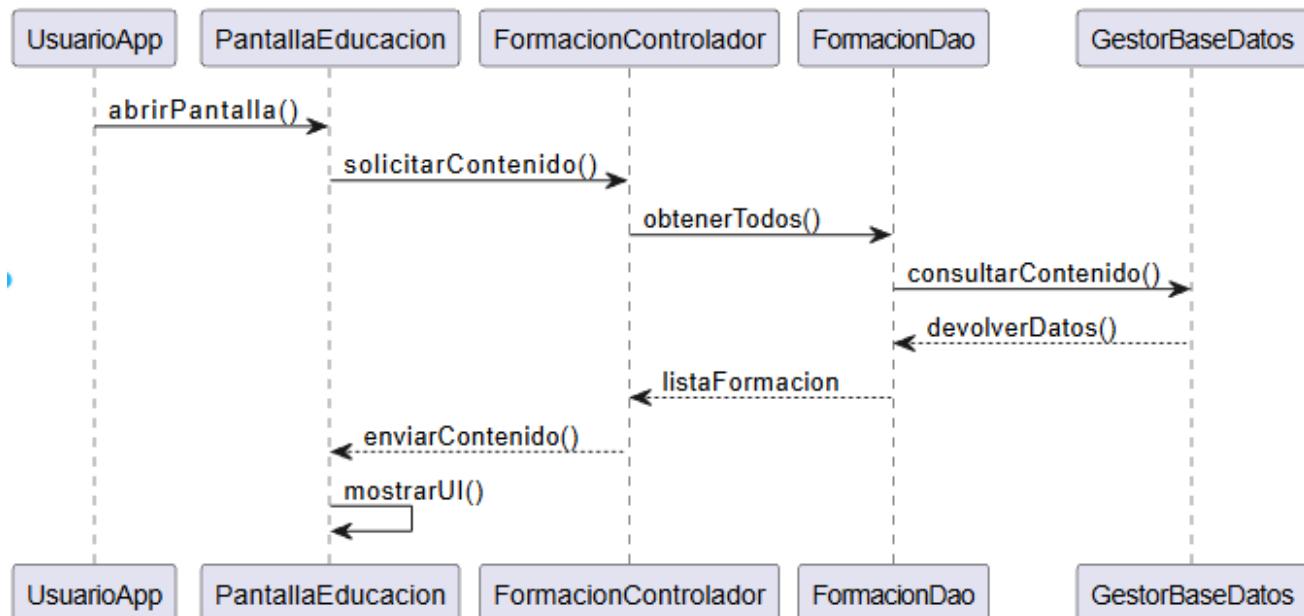
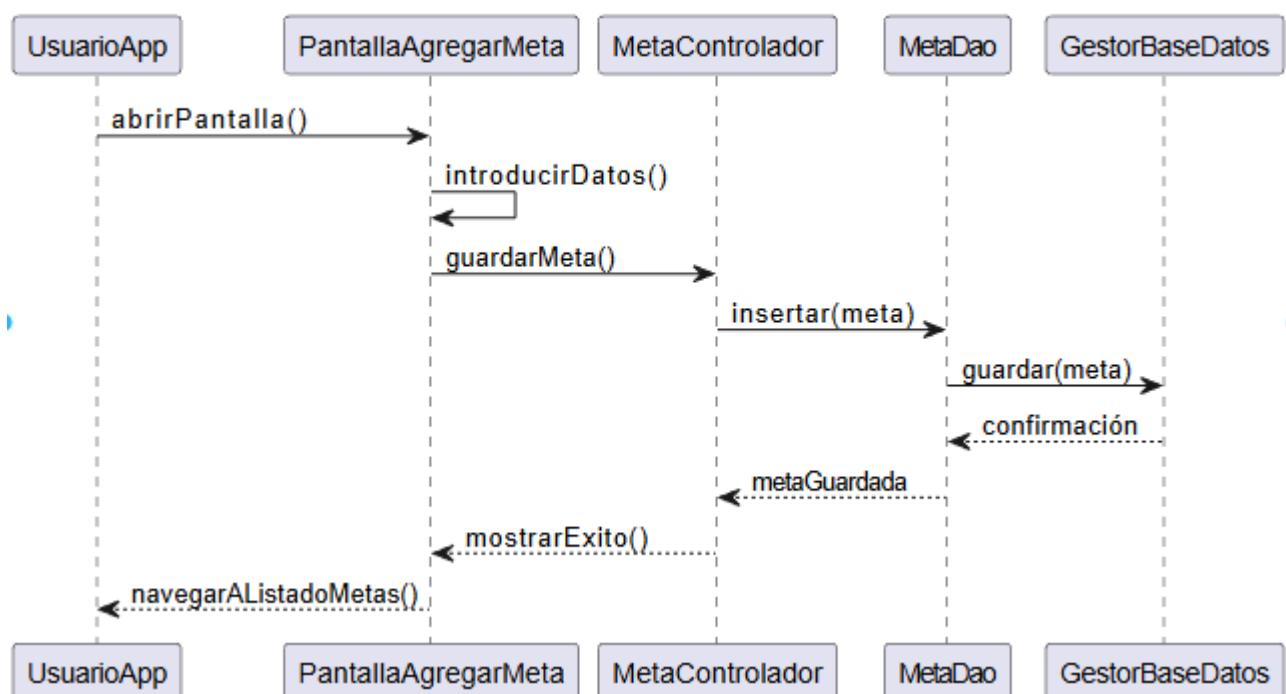
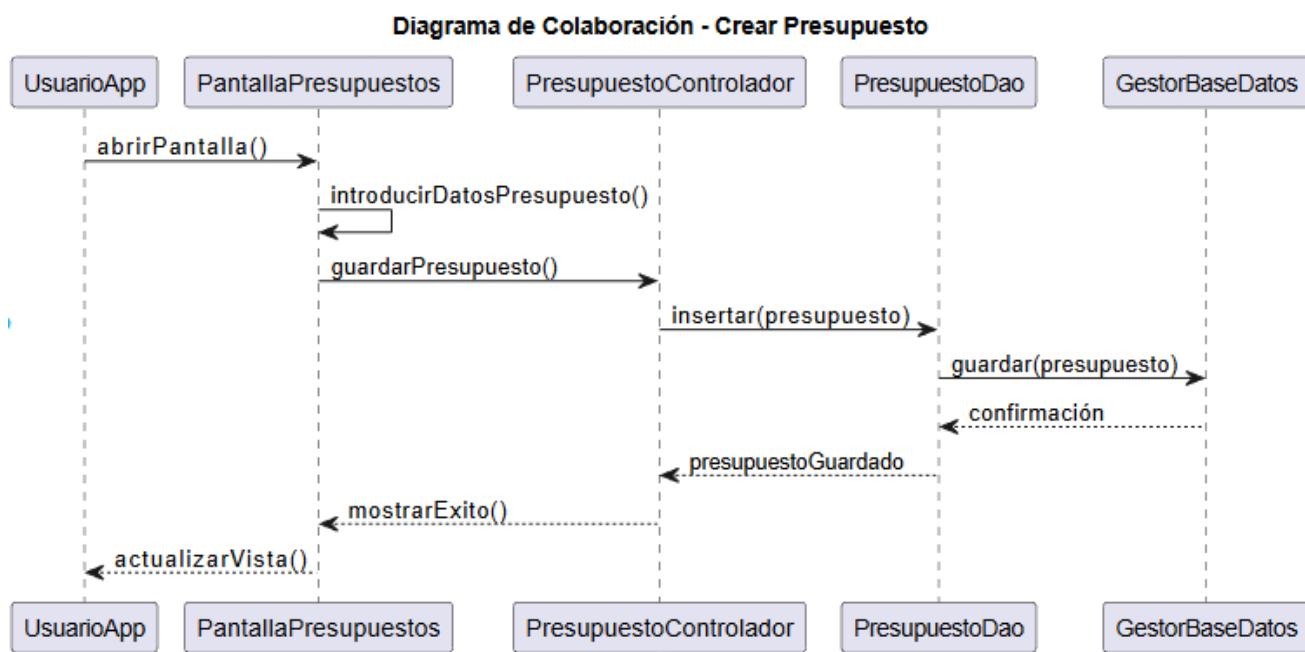
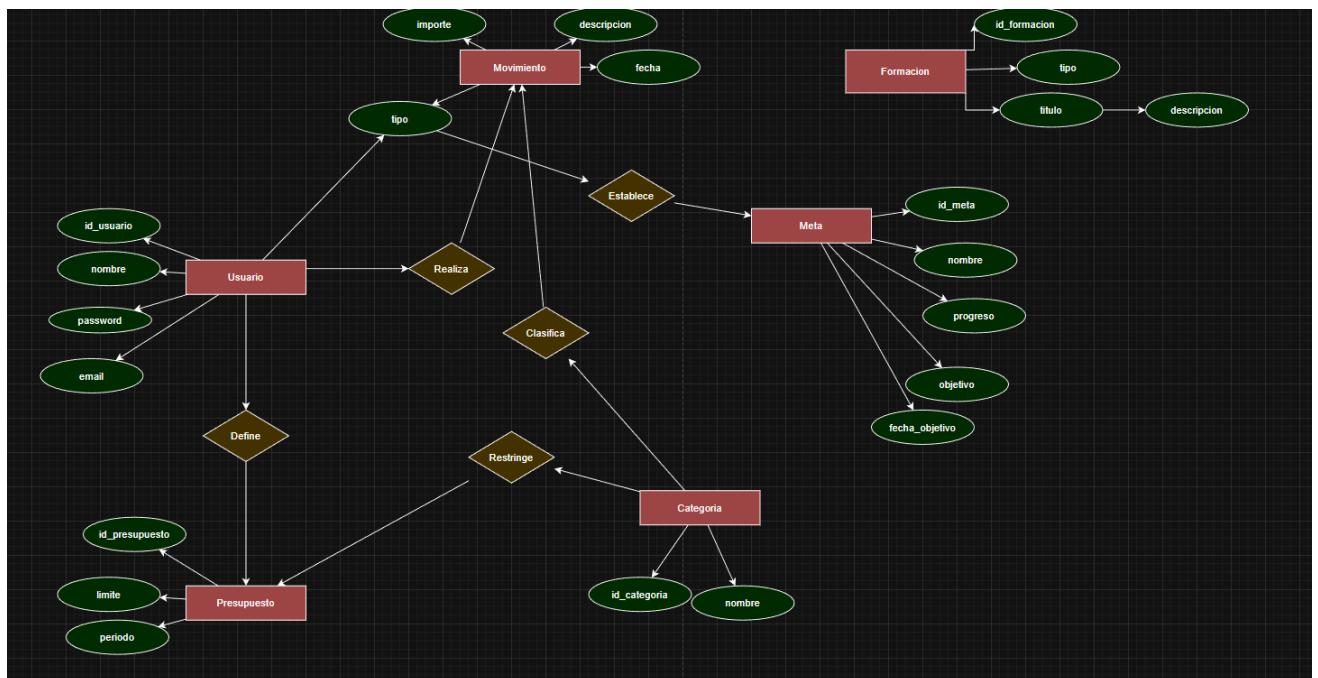


Diagrama de Colaboración - Cargar Contenido de Formación

Diagrama de Colaboración - Crear Meta




7.1.3 Modelo de datos

El modelo de datos representa las principales entidades de *Healthy Wallet* y las relaciones entre ellas, incluyendo usuarios, movimientos, categorías, presupuestos, metas y formación. Organiza la información necesaria para el funcionamiento de la aplicación y refleja cómo se vinculan los datos en el sistema, cubriendo la mayoría de los requisitos informativos del proyecto.



7.2.1 Prototipos

Pantalla de Inicio de Sesión (Login):



Fig.5 –Login

Descripción general:

Esta pantalla permite al usuario acceder a la aplicación introduciendo su correo electrónico y su contraseña. Es el punto de entrada al sistema y asegura que cada usuario pueda gestionar sus datos de manera individual.

Elementos principales:

- Logo de la aplicación en la zona superior.
- Título “Iniciar sesión”.
- Campo de texto para introducir el correo electrónico.
- Campo de texto para la contraseña.
- Botón “Entrar” para validar las credenciales.
- Enlace inferior que dirige a la pantalla de registro.

La pantalla incorpora varias validaciones para garantizar que los datos introducidos por el usuario sean correctos antes de permitir el acceso al sistema:

1. Verificación de campos obligatorios
2. Validación del formato del correo electrónico
3. Requisito mínimo de seguridad para la contraseña
4. Comprobación de credenciales incorrectas
5. Persistencia de la sesión

Funcionalidad:

El usuario introduce sus datos y pulsa el botón *Entrar*. Si las credenciales son correctas, accede al menú principal. En caso contrario, se muestra un mensaje informativo. Si el usuario no tiene cuenta, puede registrarse mediante el enlace disponible.



Pantalla de Registro de Usuario



Fig.6 –Registro

Descripción general:

Esta pantalla permite crear una nueva cuenta en la aplicación. A través de este formulario, el usuario introduce sus datos personales para registrarse por primera vez.

Elementos principales:

- Logo de la aplicación en la parte superior.
- Título “Crear cuenta”.
- Campo de texto para el nombre completo.
- Campo de texto para el correo electrónico.
- Campo de texto para la contraseña.
- Campo de texto para confirmar la contraseña.
- Botón “Registrarse”.
- Enlace inferior que dirige a la pantalla de inicio de sesión.

Funcionalidad:

El usuario completa todos los campos obligatorios y pulsa “Registrarse”. Si los datos son válidos y no existe una cuenta asociada al correo, se crea el nuevo usuario. Tras el registro, se redirige a la pantalla de inicio de sesión. Si existe un error, se notifica mediante un mensaje en pantalla.

Pantalla de Inicio:



Fig.7 –Inicio

Descripción general:

La pantalla de inicio actúa como panel principal de la aplicación, ofreciendo al usuario una vista rápida y resumida de su situación financiera. Desde esta pantalla se puede acceder a las secciones más importantes: presupuestos, movimientos, metas y educación financiera.

Elementos principales:

- **Barra superior** con el título “Inicio” y el ícono representativo.
- **Tarjeta de balance total**, que muestra el resultado actual entre ingresos y gastos.
- **Tarjeta de presupuestos**, indicando el total gastado y el límite configurado por el usuario.
- **Tarjeta de movimientos**, con el resumen de ingresos y gastos registrados.
- **Tarjeta de metas**, indicando cuántas metas están activas y cuántas se han completado.
- **Tarjeta de educación financiera**, mostrando el número de módulos registrados.
- **Barra de navegación inferior**, con accesos directos a las secciones principales de la aplicación.

Funcionalidad:

Al cargar la pantalla, la aplicación recupera los datos del usuario desde la base de datos y genera un resumen financiero actualizado. Cada tarjeta muestra información sintetizada y actúa como acceso rápido a su sección correspondiente.

El usuario puede navegar hacia los apartados de Movimientos, Presupuestos, Metas o Educación financiera pulsando cada tarjeta o usando la barra inferior. Los datos se actualizan dinámicamente cada vez que el usuario entra en esta pantalla, garantizando que siempre se muestre la información más reciente.

Pantalla: Balance del mes:

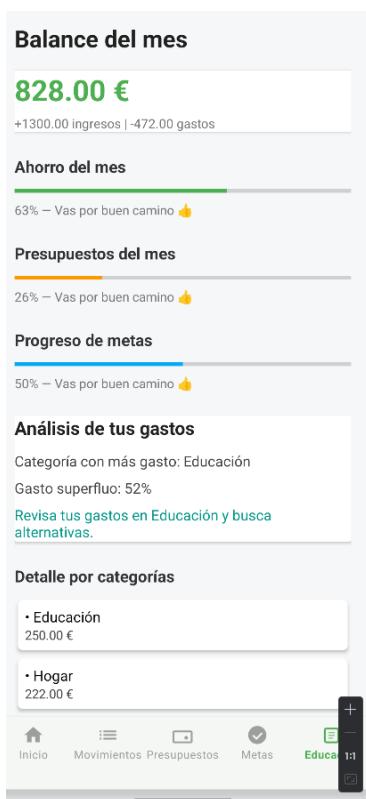


Fig.8 –Balance Mensual

Descripción general

Esta pantalla muestra un resumen detallado del estado financiero del usuario durante el mes actual. Presenta información clave de forma visual mediante tarjetas y barras de progreso, facilitando la interpretación del ahorro, los presupuestos, las metas y el gasto por categorías.

Elementos principales

- Título “Balance del mes” en la parte superior.
- Tarjeta de balance mensual, con el balance total y el resumen de ingresos y gastos.
- Ahorro del mes, representado mediante una barra de progreso y un mensaje descriptivo.
- Presupuestos del mes, con una barra que indica el porcentaje gastado.
- Progreso de metas, que muestra el avance acumulado de todas las metas activas.
- Análisis de tus gastos, con la categoría de mayor gasto y el porcentaje de gasto superfluo, acompañado de recomendaciones.
- Detalle por categorías, que lista los importes gastados en cada categoría del mes.



Funcionalidad

La pantalla carga automáticamente los datos económicos del mes en curso y muestra:

- El balance total, ingresos y gastos.
- El porcentaje de ahorro acumulado.
- El uso de los presupuestos configurados.
- El progreso global de las metas activas.
- La categoría con mayor gasto y el análisis correspondiente.
- Un desglose de los gastos por categorías.

Los valores se representan de manera visual mediante barras de progreso y tarjetas informativas para facilitar la comprensión del estado financiero mensual.

Pantalla: Movimientos

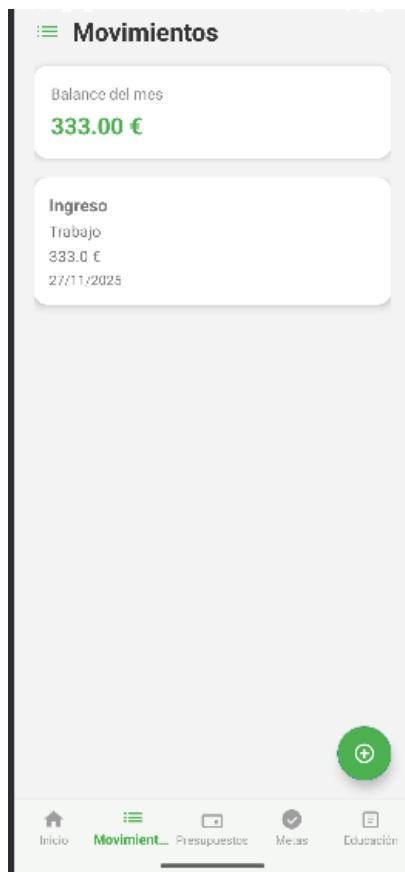


Fig.9 –Movimientos

Descripción general:

Esta pantalla permite consultar todos los movimientos financieros registrados por el usuario, tanto ingresos como gastos. Su objetivo es ofrecer una vista clara y ordenada de las operaciones económicas realizadas durante el mes y permitir añadir nuevas de forma rápida.

Elementos principales:

- **Título “Movimientos”** en la parte superior.
- **Tarjeta “Balance del mes”**, mostrando el balance total calculado a partir de ingresos y gastos acumulados.
- **Listado de movimientos**, donde cada tarjeta individual muestra:
 - Tipo de movimiento (Ingreso / Gasto).
 - Categoría asociada (ej. “Trabajo”, “Comida”, etc.).
 - Cantidad registrada.
 - Fecha del movimiento.
- **Botón flotante “+”** en la parte inferior derecha, que permite añadir un nuevo movimiento.
- **Barra de navegación inferior**, que conecta con Inicio, Movimientos, Presupuestos, Metas y Educación.

Funcionalidad:

La pantalla carga automáticamente los movimientos guardados en la base de datos, filtrados por el usuario activo.

- Calcula el **balance mensual**, que se muestra en la tarjeta superior.
- Muestra cada movimiento en orden cronológico inverso.
- Permite añadir nuevos movimientos mediante el botón flotante, llevando al usuario a la pantalla “Aregar Movimiento”.
- Al registrar un nuevo movimiento o volver desde otra pantalla, la lista se actualiza automáticamente para reflejar los datos más recientes.

Pantalla: Añadir Movimiento:

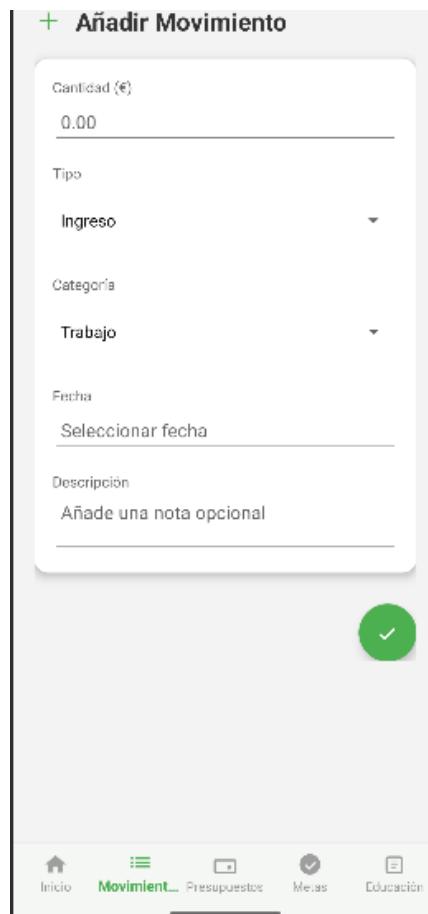


Fig.10 –Añadir Movimientos

Descripción general:

Esta pantalla permite registrar un nuevo movimiento financiero dentro de la aplicación. El usuario puede añadir tanto **ingresos** como **gastos**, especificando todos los detalles necesarios para que el sistema los clasifique correctamente.

Elementos principales:

- **Título “Añadir Movimiento”** en la parte superior.
- **Campo de texto para la cantidad (€)**, con validación para evitar valores vacíos o no numéricos.
- **Selector desplegable de tipo**, con dos opciones:
 - *Ingreso*
 - *Gasto*
- **Selector desplegable de categoría**, con valores predefinidos según el tipo de movimiento (por ejemplo: Trabajo, Comida, Ocio, Transporte...).
- **Selector de fecha**, que abre un calendario para escoger el día del movimiento.
- **Campo de texto para la descripción**, opcional, donde el usuario puede añadir una nota breve.
- **Botón flotante de confirmación (“√”)**, que guarda el movimiento.

Validaciones

La pantalla incorpora varias validaciones para asegurar que los datos introducidos sean correctos:

- **La cantidad (€) debe ser un número válido**, permitiendo tanto valores enteros como decimales.
Si el usuario introduce un formato incorrecto (letras, símbolos no permitidos o campo vacío), se muestra un mensaje de error.
- **El tipo de movimiento (Ingreso/Gasto) es obligatorio.**
- **La categoría debe seleccionarse**, y su lista depende del tipo elegido.
- **La fecha debe estar definida**, seleccionada desde el calendario.
- **La descripción es opcional**, por lo que no requiere validación.

Si todos los datos son válidos, el movimiento se registra y se actualiza la lista de la pantalla Movimientos.

Funcionalidad:

El usuario introduce la información del nuevo movimiento y confirma mediante el botón flotante.

- **Si todos los datos obligatorios son válidos**, el movimiento se registra en la base de datos local mediante Room, asociado al usuario activo.
- **Tras guardar el registro**, la aplicación vuelve automáticamente a la pantalla de Movimientos.
- En esa pantalla, **la lista se actualiza de forma inmediata**, ya que está vinculada a la base de datos y observa los cambios mediante consultas reactivas.
Esto permite que el nuevo movimiento aparezca sin necesidad de recargar manualmente la interfaz.
- **Si algún campo obligatorio está vacío o contiene un formato inválido (por ejemplo, una cantidad no numérica)**, se muestra un mensaje de error que informa al usuario sobre el problema.

Pantalla: Presupuestos

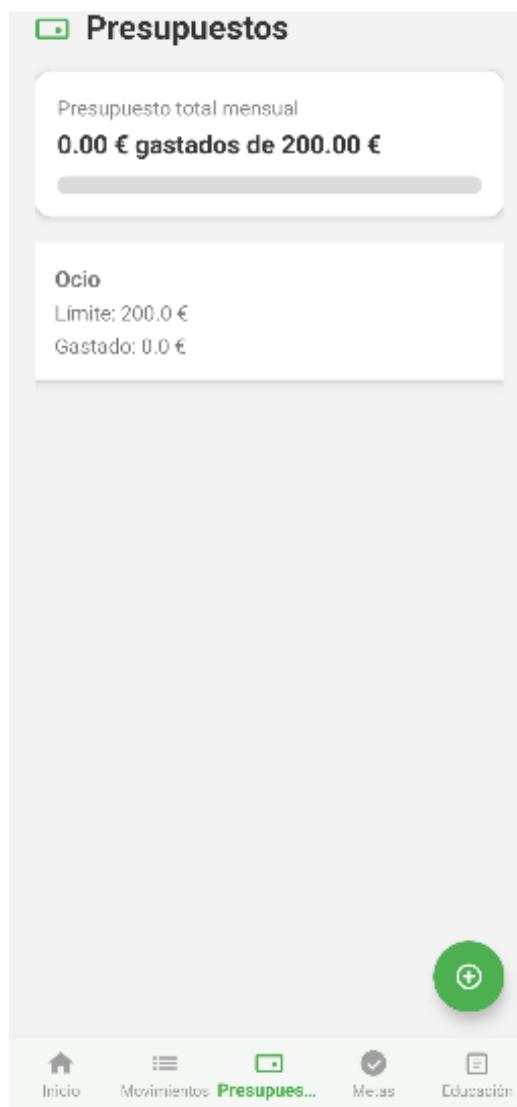


Fig.11 –Presupuestos

Descripción general

Esta pantalla muestra el estado actual de los presupuestos definidos por el usuario. Su propósito es ofrecer una visión clara del límite asignado a cada categoría de gasto y cuánto se ha consumido hasta el momento. Además, permite gestionar fácilmente nuevos presupuestos mediante el botón flotante.

Dentro de esta interfaz, el usuario puede comprobar de un vistazo si está cumpliendo sus objetivos financieros mensuales, si se está acercando al límite establecido o si aún dispone de margen para gastar.

Elementos principales

1. Título de la pantalla

En la parte superior aparece el encabezado “**Presupuestos**”, acompañado de un ícono representativo. Esto indica al usuario que se encuentra dentro de la sección destinada al control de límites de gasto.

2. Tarjeta de resumen general

La primera tarjeta de la lista muestra un resumen del estado global del mes:

- **Texto informativo:** “0.00 € gastados de 200.00 €”, indicando el gasto total acumulado comparado con el límite global sumado entre todos los presupuestos activos. Para generar el texto “X € gastados de Y €”, la aplicación suma todos los gastos realizados en categorías con presupuesto activo y los compara con la suma de los límites definidos en dichos presupuestos.
- **Barra de progreso:** Un indicador visual que muestra en qué punto del límite total se encuentra el usuario. Cuanto más avanza, mayor es el porcentaje del presupuesto que ha sido consumido. La barra de progreso utiliza estos dos valores para calcular el porcentaje consumido: $(\text{gasto acumulado} / \text{límite total}) \times 100$.

De este modo, el usuario puede visualizar de forma clara tanto la cifra exacta como el grado de avance hacia el límite presupuestario mensual.

Este componente permite identificar rápidamente si el usuario está teniendo un buen control de sus gastos o si debe ajustarse más.

3. Lista de presupuestos individuales

Cada presupuesto aparece representado en una tarjeta independiente. Cada una incluye:

- **Nombre de la categoría** (por ejemplo, Ocio, Comida, Transporte...).
- **Límite asignado** a esa categoría.
- **Gasto actual acumulado**, calculado automáticamente a partir de los movimientos que correspondan a la categoría.

Este formato facilita la comparación entre categorías y ayuda a visualizar aquellas en las que se está gastando más.

4. Botón flotante de acción (FAB)

En la esquina inferior derecha se encuentra un botón flotante verde con el icono “+”.

Este botón permite **crear un nuevo presupuesto** de forma rápida, enviando al usuario a la pantalla de “Aregar Presupuesto”.

Funcionalidad

Actualización automática de datos

Los presupuestos se actualizan dinámicamente utilizando los movimientos registrados por el usuario. Cuando se añade o elimina un gasto asociado a una categoría, el presupuesto correspondiente recalcula su **gasto actual**.

Cálculo del total mensual

La pantalla combina los datos de todos los presupuestos activos para generar:

- El gasto total del mes asociado a cualquier categoría con presupuesto.
- El límite total sumado de todos los presupuestos definidos.

Con esta información se genera la barra de progreso global, que permite obtener una visión del estado financiero del mes.

Gestión de nuevos presupuestos

Al pulsar el FAB:

- Se abre la pantalla para crear un nuevo presupuesto.
- El usuario selecciona la categoría, define un límite y lo guarda.
- Al volver a esta pantalla, la lista se actualiza instantáneamente.

Visualización sencilla y clara

El diseño está orientado a que el usuario pueda identificar rápidamente:

- Qué categorías están consumiendo más recursos.
- Qué categorías aún tienen margen de gasto.
- El estado financiero mensual general sin tener que abrir cada presupuesto individualmente.

Pantalla Agregar Presupuesto

Crear Presupuesto

Categoría

Categoría: Comida

Límite (€)

Ej: 150.00

Guardar



Fig.12 –Crear Presupuesto

Descripción general:

Esta pantalla permite añadir un nuevo presupuesto para una categoría específica. El usuario define cuánto quiere gastar como máximo en un tipo de gasto determinado.

Elementos principales:

- **Título “Crear Presupuesto”.**
- **Selector de categoría**, con opciones como Comida, Ocio, Transporte, etc.
- **Campo de texto para el límite (€)**, donde el usuario introduce el tope mensual deseado.
- **Botón “Guardar”**, que confirma la creación.

Funcionalidad:

El usuario selecciona una categoría, introduce un límite válido y pulsa “Guardar”.

Si los datos son correctos, el presupuesto se registra en la base de datos con gastoActual = 0, y se muestra en la pantalla de Presupuestos.

Si falta algún campo o el límite no es válido, aparece un mensaje de error.

Pantalla Metas:

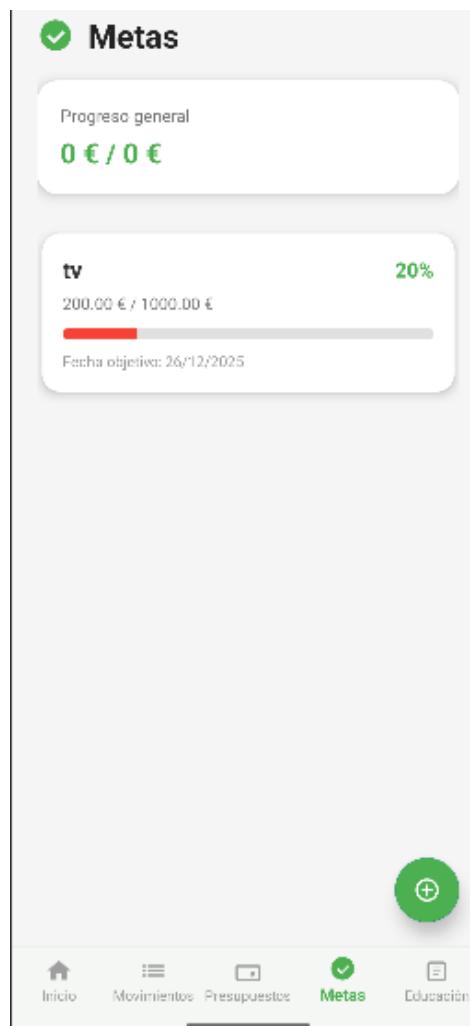


Fig.13 –Metas

Descripción general:

Pantalla donde el usuario visualiza todas sus metas de ahorro, junto con el progreso acumulado y el avance general hacia sus objetivos financieros.

Elementos principales:

- Título “Metas”.
- Tarjeta de “Progreso general” mostrando ahorro total vs. objetivo total.
- Lista de metas individuales, cada una con:
 - Nombre de la meta.
 - Progreso actual / objetivo final.
 - Porcentaje de avance.
 - Barra de progreso.
 - Fecha objetivo.
- Botón flotante “+” para añadir una nueva meta.



Funcionalidad:

La pantalla carga todas las metas registradas del usuario y calcula automáticamente el porcentaje completado de cada una.

Al pulsar el botón “+”, se abre la pantalla para crear una nueva meta.

Seleccionar una meta permite ver más detalles y añadir ahorro adicional.



Pantalla Nueva Meta:

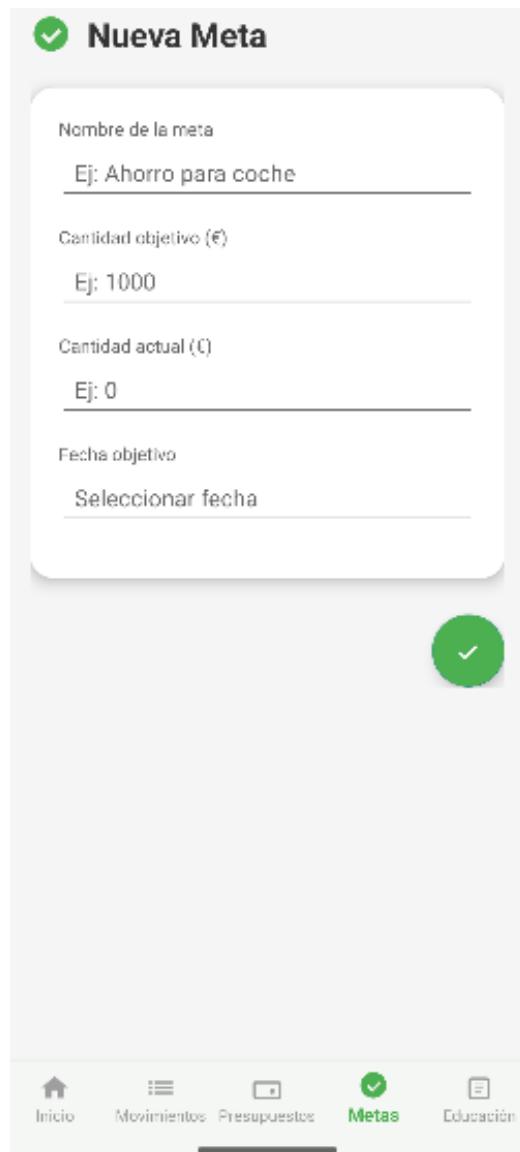


Fig.14 – Nuevas Metas

Descripción general:

Pantalla destinada a crear una nueva meta de ahorro. El usuario define un objetivo económico y una fecha límite para organizar mejor sus finanzas personales.

Elementos principales:

- Título “Nueva Meta”.
- Campo para el nombre de la meta (ej.: “Ahorro para coche”).
- Campo para la cantidad objetivo (€).
- Campo para la cantidad actual (€), normalmente iniciada en cero.
- Selector de fecha para establecer la fecha objetivo.
- Botón flotante de confirmación (“✓”) para guardar.

Funcionalidad:

El usuario completa los campos obligatorios, elige la fecha objetivo y pulsa el botón de confirmación.

Si los datos son válidos, la meta se registra en la base de datos y se muestra en la pantalla de Metas.

Si falta algún dato o la cantidad no es válida, se muestra un aviso de error.

Pantalla Educación Financiera:

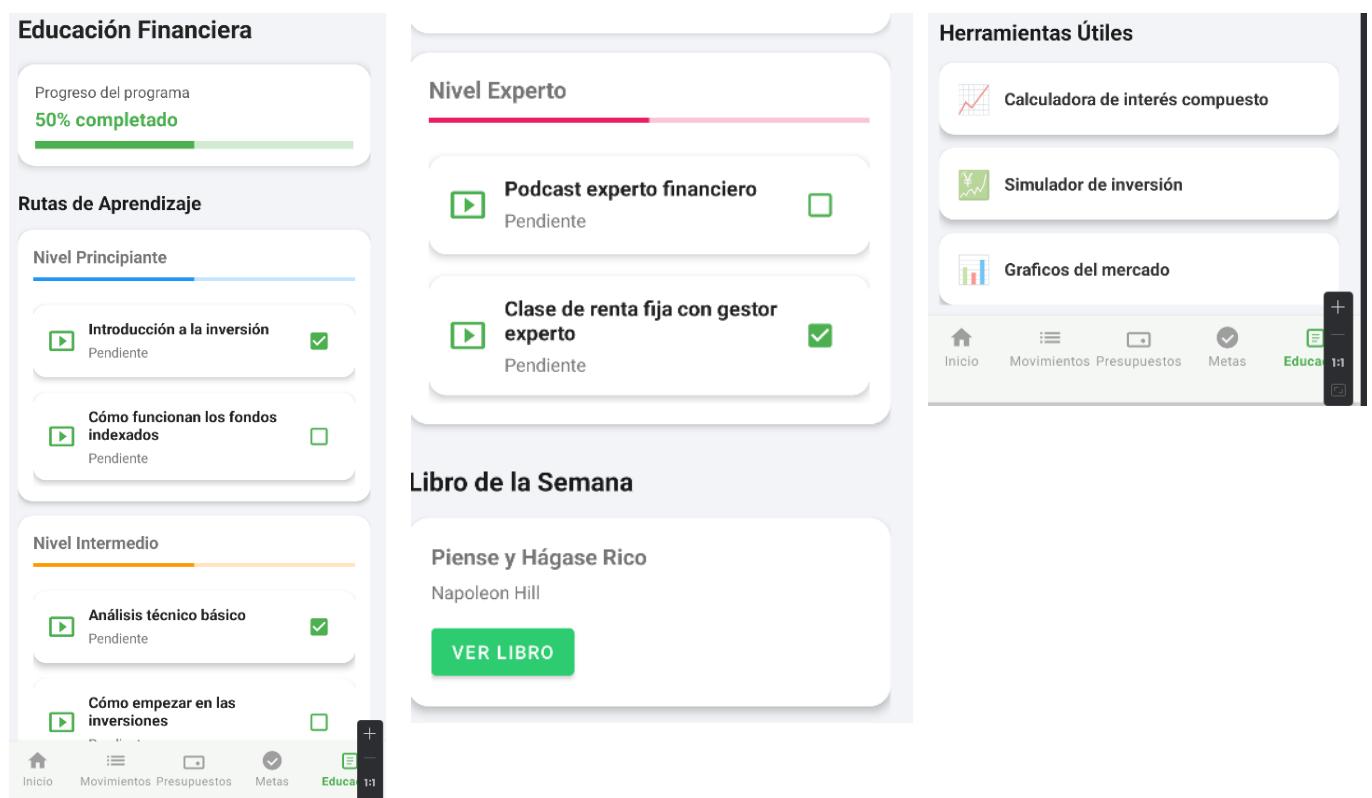


Fig.15 – educación financiera

Descripción general

Esta pantalla reúne los recursos formativos disponibles en la aplicación. Permite al usuario seguir un itinerario educativo dividido por niveles y acceder a materiales adicionales como libros y herramientas financieras.

Elementos principales

Progreso del programa: Tarjeta superior que muestra el porcentaje total completado mediante una barra de avance.

Rutas de aprendizaje: Contenidos organizados en tres niveles (Principiante, Intermedio y Experto), cada uno con videos que el usuario puede marcar como completados.

Libro de la semana: Recomendación destacada con acceso directo al recurso.

Herramientas útiles: Tarjetas que enlazan a calculadoras y simuladores externos.

Funcionalidad

Cada vídeo puede marcarse como Pendiente o Completado, lo que actualiza automáticamente el progreso del nivel y del programa general.

Los libros y herramientas actúan como enlaces externos que se abren en el navegador.

Toda la información es accesible y se actualiza cada vez que el usuario interactúa con los elementos de la pantalla.



7.2.2 Código

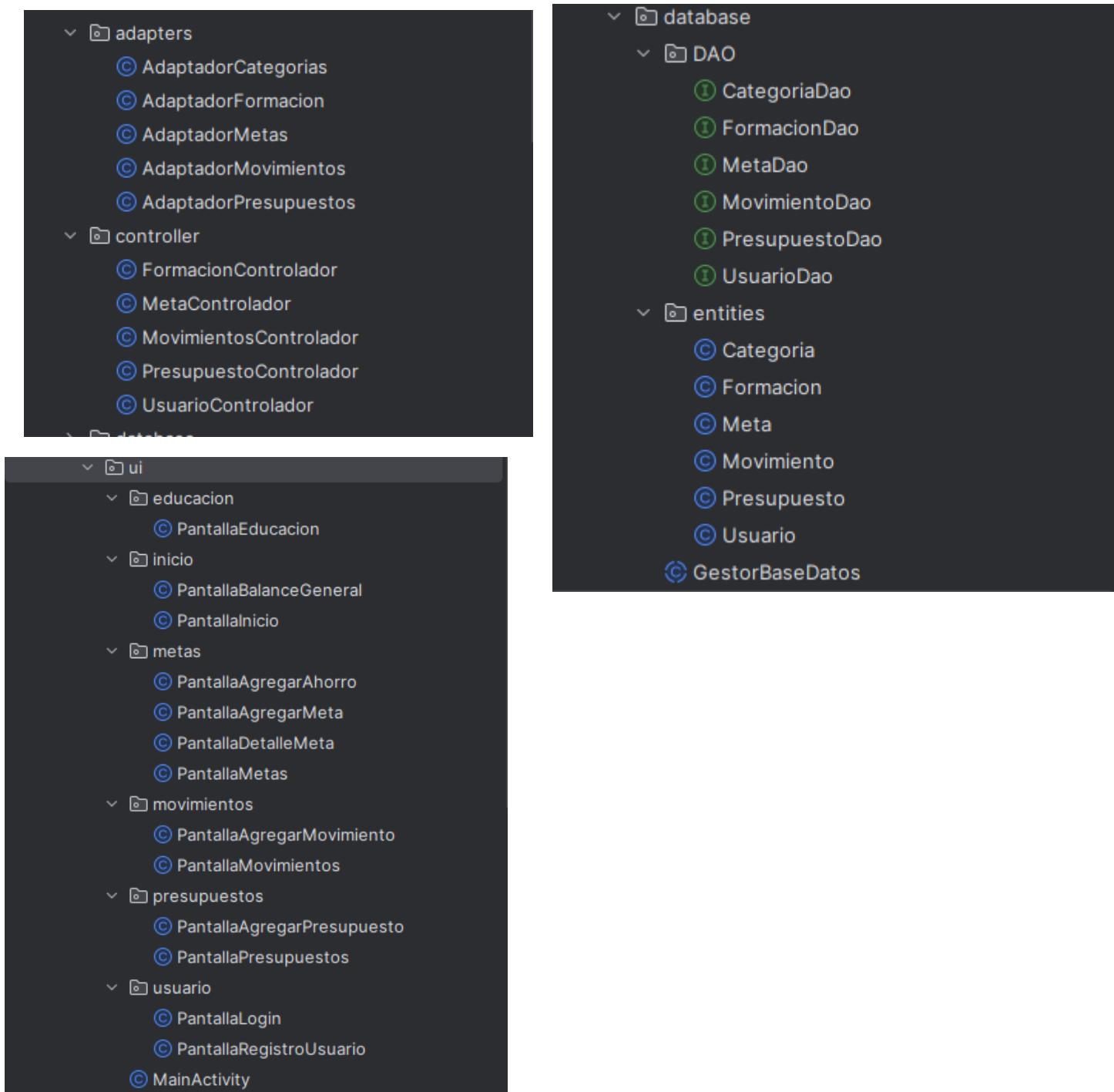


Fig. 16 – Estructura del código(DAOs y java)

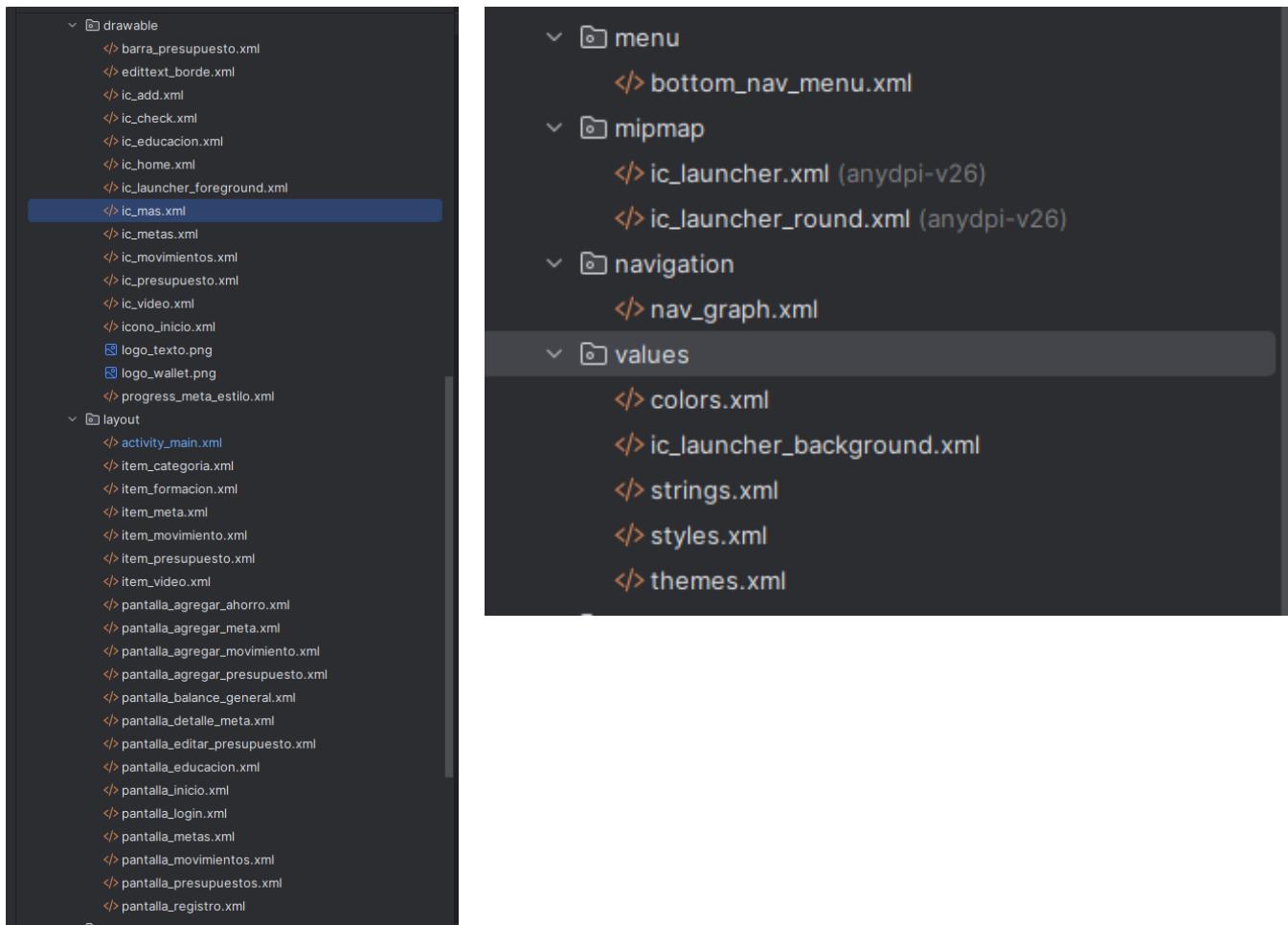


Fig. 17 – Estructura del código(XML)

Arbol de la estructura del proyecto

```
HealthyWallet/
|
|   └── manifests/
|       └── AndroidManifest.xml
|
|   └── java/
|       └── com/example/healthywallet/
|           |
|           └── ui/
|               |
|               └── MainActivity.java
|               |
|               └── inicio/
|                   |
|                   └── PantallaInicio.java
|                   └── PantallaBalanceGeneral.java
|               |
|               └── movimientos/
|                   |
|                   └── PantallaMovimientos.java
|                   └── PantallaAregarMovimiento.java
|                   └── MovimientoAdapter.java
|               |
|               └── presupuestos/
|                   |
|                   └── PantallaPresupuestos.java
|                   └── PantallaAregarPresupuesto.java
|                   └── PantallaEditarPresupuesto.java
|                   └── PresupuestoAdapter.java
|               |
|               └── metas/
|                   |
|                   └── PantallaMetas.java
|                   └── PantallaAregarMeta.java
|                   └── PantallaAregarAhorro.java
|                   └── PantallaDetalleMeta.java
|                   └── MetaAdapter.java
|               |
|               └── educacion/
|                   |
|                   └── PantallaEducacion.java
|                   └── AdaptadorVideos.java
|                   └── VideoItem.java
|               |
|               └── usuario/
|                   └── PantallaLogin.java
|                   └── PantallaRegistroUsuario.java
|               |
|               └── database/
|                   |
|                   └── AppDatabase.java
|                   └── dao/
|                       └── UsuarioDao.java
```



```
|   |   |   └── MovimientoDao.java  
|   |   |   └── PresupuestoDao.java  
|   |   |   └── CategoriaDao.java  
|   |   └── MetaDao.java  
|   |  
|   |   └── entities/  
|   |       ├── Usuario.java  
|   |       ├── Movimiento.java  
|   |       ├── Presupuesto.java  
|   |       ├── Categoria.java  
|   |       ├── Meta.java  
|   |       └── Formacion.java  
|   |  
|   |   └── controller/  
|   |       ├── UsuarioControlador.java  
|   |       ├── MovimientoControlador.java  
|   |       ├── PresupuestoControlador.java  
|   |       ├── MetaControlador.java  
|   |       ├── CategoriaControlador.java  
|   |       └── EducacionControlador.java  
|   |  
|   |   └── utils/  
|   |       ├── DateUtils.java  
|   |       └── FormatUtils.java  
|  
└── res/  
    ├── layout/  
    |   ├── activity_main.xml  
    |   ├── pantalla_login.xml  
    |   ├── pantalla_registro_usuario.xml  
    |   ├── pantalla_inicio.xml  
    |   ├── pantalla_balance_general.xml  
    |   ├── pantalla_movimientos.xml  
    |   ├── pantalla_agregar_movimiento.xml  
    |   ├── pantalla_presupuestos.xml  
    |   ├── pantalla_agregar_presupuesto.xml  
    |   ├── pantalla_editar_presupuesto.xml  
    |   ├── pantalla_metas.xml  
    |   ├── pantalla_agregar_meta.xml  
    |   ├── pantalla_agregar_ahorro.xml  
    |   ├── pantalla_detalle_meta.xml  
    |   ├── pantalla_educacion.xml  
    |   ├── item_movimiento.xml  
    |   ├── item_presupuesto.xml  
    |   ├── item_meta.xml  
    |   └── item_video.xml
```

```
|   |   └── item_categoria.xml  
|   |  
|   ├── menu/  
|   |   └── bottom_nav_menu.xml  
|   |  
|   ├── navigation/  
|   |   └── nav_graph.xml  
|   |  
|   ├── drawable/  
|   |   ├── fondos, iconos, shapes...  
|   |   └── (archivos gráficos personalizados)  
|   |  
|   ├── values/  
|   |   ├── colors.xml  
|   |   ├── strings.xml  
|   |   ├── styles.xml  
|   |   └── dimens.xml  
|   |  
|   └── mipmap/  
|       └── iconos de la app  
|  
└── Gradle Scripts/  
    ├── build.gradle (Project)  
    ├── build.gradle (Module)  
    ├── gradle-wrapper.properties  
    └── settings.gradle
```

Explicación:

Esta es la estructura del proyecto *HealthyWallet*, donde se organizan los distintos paquetes y archivos Java que forman la aplicación.

La app está desarrollada en Android utilizando **Room** como sistema de persistencia local, por lo que el código se divide principalmente en capas **UI, Controladores, DAO, Entities** y **Gestor de Base de Datos**, siguiendo un enfoque claro y modular.

Debido a la extensión del código, no puede mostrarse en su totalidad dentro del documento. Por ello, todo el contenido se encuentra disponible en el repositorio de GitHub del proyecto, donde está organizado, documentado y accesible para su lectura y análisis.

Enlace al repositorio de GitHub del proyecto:

<https://github.com/MarcosPequeMarban/HealthyWallet#>

El código principal se encuentra en:

app/src/main/java/com/example/healthywallet/

Sistema de Guardado de Datos

El sistema de guardado de datos en *HealthyWallet* se gestiona mediante **Room Database**, una librería oficial de Android que permite almacenar información de forma segura, estructurada y persistente en el dispositivo.

Este sistema asegura integridad, acceso rápido y evita que el usuario pueda manipular los datos fácilmente.

La aplicación almacena los siguientes datos del usuario:

• Usuario

- ID del usuario
- Nombre
- Correo electrónico
- Contraseña (almacenada internamente mediante hashing)

• Movimientos

- Tipo (Ingreso / Gasto)
- Categoría (Ej: Comida, Trabajo, Ocio...)
- Cantidad
- Fecha
- Descripción
- ID del usuario propietario

• Presupuestos

- Categoría asociada
- Límite máximo mensual
- Gasto actual calculado dinámicamente
- ID del usuario



• Metas de ahorro

- Nombre de la meta
- Cantidad objetivo
- Cantidad actual
- Fecha objetivo
- ID del usuario

• Módulos de educación

- Título del módulo
- Categoría (Vídeo, Libro, Herramienta)
- URL o recurso asociado
- ID del usuario (si se registran como vistos o completados en el futuro)

8.1 Pruebas

A continuación se describen las pruebas realizadas para validar el correcto funcionamiento de la aplicación Healthy Wallet, asegurando que cumple con los requisitos funcionales, de información y no funcionales definidos previamente.

El proceso de verificación se ha centrado tanto en pruebas funcionales como en pruebas de usabilidad, rendimiento y validación de datos, siguiendo como referencia el enfoque mostrado en Plantilluca.pdf.

8.1.1 Pruebas funcionales

Prueba 1: Inicio de sesión (UC-01 / RF-07 / RNF-02)

Objetivo: Verificar que el usuario puede iniciar sesión correctamente y que las validaciones se ejecutan según lo previsto.

Entradas:

- Email válido / inválido
- Contraseña válida / inválida
- Campos vacíos

Resultado esperado:

- Si los campos son incorrectos o incompletos, la app muestra mensajes de validación.
- Si las credenciales son correctas, se guarda sesión local y se accede a la pantalla Inicio.

Resultado obtenido:

Correcto. La app valida email, formato, longitud mínima de contraseña y credenciales, mostrando mensajes informativos.

Prueba 2: Registrar movimiento (UC-01 / RF-01 / RI-02)

Objetivo: Comprobar que se registra un movimiento y se actualizan los cálculos dependientes.

Entradas:

- Tipo (ingreso/gasto)
- Cantidad numérica válida o inválida
- Categoría seleccionada
- Fecha y descripción

Resultado esperado:

- Validación de importes (enteros y decimales).
- Guardado en Room.
- Actualización inmediata de balances, presupuestos y metas.

Resultado obtenido:

Correcto. Los movimientos se guardan y las pantallas dependientes se recalculan en tiempo real gracias a consultas Room.

Prueba 3: Gestión de presupuestos (UC-03 / RF-03 / RI-03)

Objetivo: Comprobar la creación, edición y eliminación de presupuestos.

Entradas:

- Categoría
- Límite de gasto
- Periodo (mensual)

Resultado esperado:

- Validación de datos numéricos.
- Suma del gasto real de la categoría en el periodo.
- Barra de progreso calculada como:

$$\text{progreso} = \text{gastoActual} / \text{límiteTotal} \times 100$$

Resultado obtenido:

Correcto. El sistema suma todos los movimientos relacionados y representa el avance visualmente.

Prueba 4: Gestión de metas (UC-04 / RF-05 / RI-04)

Objetivo: Verificar la creación, edición y aportación de ahorro sobre metas.

Entradas:

- Objetivo, cantidad, fecha, aportaciones

Resultado esperado:

- Registro de aportaciones
- Recálculo del porcentaje:

$$\text{progresoMeta} = \text{ahorroAcumulado} / \text{objetivo} \times 100$$

Resultado obtenido:

Correcto. El progreso se actualiza dinámicamente y las aportaciones se almacenan en Room.

Prueba 5: Acceso a contenido educativo (UC-05 / RF-06 / RI-05)

Objetivo: Validar el acceso y marcado de contenidos.

Resultado esperado:

- Listado por niveles
- Marcado como completado
- Registro del progreso

Resultado obtenido:

Correcto.

8.1.2 Pruebas de interfaz y usabilidad (RNF-01 / RNF-06)

Prueba de adaptación a pantallas

Se realizaron pruebas en:

- Xiaomi Redmi Note 8
- OPPO A54
- Google Pixel 5 (emulador)
- Tablet 10"

Resultado:

La app se adapta correctamente gracias al uso de ConstraintLayout, dimensiones en dp y márgenes proporcionados.

El BottomNavigationView respeta los insets del sistema y no invade el contenido.

Prueba de accesibilidad

Comprobaciones realizadas:

- Textos escalables mediante sp
- Contraste suficiente en tarjetas clave
- Elementos táctiles $\geq 48dp$

Resultado:

Cumple accesibilidad básica y está preparada para mejoras futuras.

8.1.3 Pruebas de persistencia y datos (RNF-03 / RNF-02)

Validación del almacenamiento con Room

- Creación de tablas en primera instalación
- Migración automática en versión nueva
- Recuperación correcta tras cierre de app

Resultado:

100% correcto. Los datos persisten incluso tras reiniciar el dispositivo.

Prueba offline

- La app permite registrar, editar y consultar movimientos sin conexión.
- Los cálculos se realizan de forma local.

Resultado:

Cumple los requisitos offline-first.

8.1.5 Pruebas de errores y validaciones

Se verificaron las siguientes validaciones:

Caso	Resultado esperado	Resultado obtenido
Email inválido	Aviso “correo no válido”	Correcto
Contraseña < 6 caracteres	Aviso “mínimo 6 caracteres”	Correcto
Campo cantidad vacío	Error “introduce una cantidad”	Correcto
Cantidad no numérica	Error “formato no válido”	Correcto
Crear presupuesto sin límite	Aviso obligatorio	Correcto
Añadir ahorro sin cantidad	Error	Correcto

Conclusión de pruebas

La aplicación Healthy Wallet **superá todas las pruebas funcionales, de rendimiento y usabilidad**, cumpliendo los requisitos definidos en el diseño del sistema. Room, MVVM y la estructura modular garantizan estabilidad y rapidez, mientras que las pruebas en múltiples dispositivos confirman que la interfaz se adapta correctamente.

9.1 Implementación

9.1.1 Requisitos Mínimos

Requisitos Mínimos – HealthyWallet

Sistema Operativo	Android	Windows(solo para desarrollo)
Versión SO	Android 6.0 (API 23)+	Windows 10 (build 19041)+
CPU	ARMv7 (32-bit) o ARM64	Procesador x86, x64 o ARM64
Gráficos	OpenGL ES 3.0 o superior	GPU compatible con DX10 o superior
Requisitos Adicionales	1 GB RAM (mínimo recomendado 2 GB) 50 MB de almacenamiento libre	4 GB RAM Android Studio instalado 2 GB de almacenamiento libre

Fig. 18 – Tabla de requisitos mínimos



9.1.2 Tabla de tiempo

A continuación, se presenta la tabla donde se refleja la planificación temporal del proyecto, indicando las semanas empleadas en cada uno de los ámbitos: análisis, diseño, desarrollo e implementación.

Fase		Sept.	Octubre					Noviembre				Diciembre
			4 ^a Semana	1 ^a Semana	2 ^a Semana	3 ^a Semana	4 ^a Semana	1 ^a Semana	2 ^a Semana	3 ^a Semana	4 ^a Semana	
Análisis	definición del proyecto											
	Estudio del sector											
	análisis de requisitos											
Diseño	Diseño de interfaz(UI /UX)											
	Prototipos											
	Diseño de BD y clases											
Desarrollo	Modulo de usuarios											
	Modulo de movimientos											
	Modulo de presupuestos											
	Modulo de metas											
	Modulo de educación financiera											
	documentación final											
implementación												

Fig. 19 – Tabla de tiempo



10.1 Licencias

10.1.1 Licencia de Android Studio y Android SDK

Derechos de autor © 2024 Google LLC.

“Android”, “Android Studio”, el logotipo de Android y otros nombres asociados al ecosistema Android son marcas comerciales o marcas registradas de Google LLC y sus filiales en Estados Unidos y en el resto del mundo.

El desarrollo de la aplicación *HealthyWallet* se ha realizado utilizando **Android Studio** y el **Android Software Development Kit (SDK)**, cuya utilización está regulada por los términos del *Android Software Development Kit License Agreement*.

Este acuerdo permite el uso del SDK exclusivamente para el desarrollo, prueba y distribución de aplicaciones compatibles con dispositivos Android.

Cualquier modificación, redistribución no autorizada o ingeniería inversa del SDK está prohibida y protegida legalmente por Google.

Todas las demás marcas mencionadas pertenecen a sus respectivos propietarios.

10.1.2 Licencia de iconos y recursos gráficos

Los iconos utilizados en la aplicación proceden de:

- **Material Icons de Google**, bajo licencia Apache 2.0.
- **Recursos vectoriales incluidos en Android Studio**, de libre uso para proyectos desarrollados en la plataforma.
- Elementos gráficos diseñados específicamente para *HealthyWallet*, cuyos derechos pertenecen al autor del proyecto.

Todos los logotipos, imágenes o nombres comerciales pertenecen a sus respectivos propietarios.

10.1.3 Licencia del repositorio del proyecto

El código fuente de *HealthyWallet* se encuentra disponible en GitHub bajo la licencia:

MIT License

© 2024 Marcos Peque Marbán.

Esta licencia permite:

- Uso, copia y modificación del software sin restricciones.
- Distribución libre tanto en proyectos personales como comerciales.
- Únicamente se requiere mantener el aviso de copyright en cualquier copia.

La licencia completa puede consultarse en:

<https://github.com/MarcosPequeMarban/HealthyWallet>

10.1.4 Aclaración legal

HealthyWallet es una aplicación original desarrollada exclusivamente para fines académicos.

Cualquier marca comercial, nombre o recurso externo mencionado pertenece a sus respectivos propietarios y se utiliza únicamente con fines demostrativos dentro del proyecto educativo.

11.1 Referencias y bibliografía

- **Diseño de interfaces en Android Studio:**

Google. *Layout Editor y XML Design.*

<https://developer.android.com/studio/write/layout-editor>

- **Base de datos Room**

Google. *Room Persistence Library.*

<https://developer.android.com/training/data-storage/room>

- **Iconos e imágenes**

Google. *Material Icons Library.*

<https://fonts.google.com/icons>

- **Recursos de diseño y vectores**

Android Studio. *Vector Asset Studio.*

Integrado en el IDE.

- **GitHub del Proyecto HealthyWallet**

Marcos Peque Marbán. *Repositorio completo del proyecto.*

<https://github.com/MarcosPequeMarban/HealthyWallet>

12.1 Conclusiones

12.1 Mejoras futuras

Aunque *HealthyWallet* es totalmente funcional y cumple los requisitos planteados, el proyecto presenta un amplio margen de crecimiento y una clara proyección hacia versiones más avanzadas y muy potentes en salida al mercado. Existen numerosas mejoras que podrían implementarse en el futuro para aumentar su valor educativo y su utilidad como gestor financiero personal:

- **Sincronización en la nube**, permitiendo conservar los datos al cambiar de dispositivo.
- **Análisis financiero avanzado**, previsiones de gasto.
- **Sistema de notificaciones inteligentes**, tanto para metas como para presupuestos y hábitos de ahorro.
- **Gamificación**, con logros, retos semanales y recompensas virtuales para fomentar la constancia.
- **Integración con entidades bancarias (OpenBanking)**, que permitiría registrar movimientos automáticos.
- **Módulos ampliados de educación financiera**, con vídeos, ejercicios interactivos, exámenes, títulos con certificado y seguimiento personalizado.
- **Implementación de Herramienta IA propia, esta ia podrá hacer**: estudio de mercado, estudio de tus propias finanzas y nivel financiero (nominas, facturas, hipotecas etc..).
- **Integración de notificaciones push** para recordar pagos, registrar gastos o avisar sobre límites de presupuesto.

Por cuestión de tiempo, esta funcionalidad se deja contemplada para futuras versiones, pero su incorporación sería sencilla gracias a la arquitectura ya implementada y al uso de Room.

Además, en un modelo escalado y profesional, se plantea una mejora especialmente relevante:

- **Colaboración con expertos, economistas y gestores financieros reconocidos**, que aporten contenido actualizado, guías prácticas y recomendaciones personalizadas.
- **Participación de inversores y gestoras de activos**, que permitan incluir simuladores de inversión o material educativo verificado.
- **Convenios con profesores, divulgadores y especialistas en economía**, con el fin de convertir la aplicación en un recurso educativo fiable y de gran alcance.

Estas colaboraciones —propias de una versión comercial a gran escala— permitirían transformar *HealthyWallet* en una plataforma de aprendizaje financiero completa, algo que queda fuera del alcance de un proyecto académico, pero que demuestra claramente el potencial real del producto.

12.2 Conclusiones Personales sobre el Proyecto

-El desarrollo de *HealthyWallet* ha sido un proyecto profundamente enriquecedor a nivel personal y académico. Me ha permitido adquirir una comprensión sólida del desarrollo móvil en Android, la gestión de bases de datos locales ROOM con la que no había trabajado, la arquitectura de aplicaciones modernas y el diseño centrado en el usuario. Como en cualquier proyecto real, han surgido desafíos técnicos y decisiones complejas, pero con trabajo y convicción ha ido saliendo todo con bastante facilidad.

Más allá de la parte técnica, este proyecto era una idea muy personal y que vi clara desde el primer día, por qué fue que yo me empecé a formar en el tema de las finanzas, en ese momento me di cuenta de lo poco que nos enseñan sobre la gestión de nuestro dinero en clase, en casa, en cualquier ámbito al fin y al cabo. Si eres una persona que sabe de finanzas y de gestión activa del capital, o de cosas mínimas como la inflación ves claramente que hay un problema educativo en todos los niveles,

Por eso cuando propuse la idea creo que es una idea que puede llevarse a cabo a nivel de sacarla a mercado y se una aplicación muy útil para casi todo el mundo, por eso elegí este proyecto.

Aunque esta versión supone un modelo funcional y académico, **el potencial escalable de la idea es muy elevado**. Con la inversión adecuada, con tiempo y con un equipo profesional, *HealthyWallet* podría evolucionar hacia una aplicación de referencia en educación financiera. Integrar contenido acreditado por expertos, herramientas avanzadas de análisis y colaboraciones con instituciones educativas permitiría convertirla en una plataforma completa, útil no solo a nivel personal, sino incluso en ámbitos formativos.

En definitiva, este proyecto ha sido un reto que he disfrutado personalmente porque la idea es mía y me hacía ilusión sacarlo adelante. A sido una fuente de aprendizaje y una puerta abierta hacia futuras oportunidades. Sus posibilidades de crecimiento son claras, y su propósito —ayudar a mejorar la salud financiera de las personas— le da un valor que sale del valor técnico de mercado es también una labor social y sobre todo educativa.