



This is the **published version** of the bachelor thesis:

Anglès Roca, Joel; Benítez Fernández, Yolanda, dir. Desarrollo de una Aplicación Nativa de Gestión de Ingresos y Gastos Mensuales. 2023. (Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/298943>

under the terms of the  license

IncomeManager: Desenvolupament d'una aplicació nativa de gestió d'ingressos i despeses mensuals

Joel Anglès Roca

28 de juny de 2024

Resum– Aquest projecte consisteix en el desenvolupament d'una aplicació per a dispositius mòbils iOS amb Swift i Core Data. L'objectiu és facilitar la gestió dels ingressos i les despeses mensuals dels usuaris. Inspirada en aplicacions com Monefy, destaca per la seva capacitat de poder fraccionar els ingressos en categories. La metodologia de desenvolupament utilitzada ha sigut el Waterfall Iteratiu, que combina planificació estructurada amb flexibilitat per ajustar-se a canvis durant el procés de desenvolupament.

Paraules clau– Aplicació, nativa, ingressos, despeses, distribució, Swift, Core Data, ViewModel, arquitectura, Waterfall Iteratiu, iOS, mensual.

Abstract– This project consists of developing a mobile application for iOS devices using Swift and Core Data. The goal is to facilitate users in managing their monthly incomes and expenses. Inspired by apps like Monefy, it stands out for its ability to categorize incomes. The development methodology used has been Iterative Waterfall, which combines structured planning with flexibility to accommodate changes during the development process.

Keywords– Application, native, income, expenses, distribution, Swift, Core Data, ViewModel, architecture, Waterfall Iterative, iOS, monthly.

1 INTRODUCCIÓ - CONTEXT DEL TREBALL

En el panorama actual, Espanya s'enfronta a múltiples reptes financers que afecten directament la vida quotidiana de les persones. El creixent cost del lloguer, l'augment de la inflació i l'increment dels preus dels serveis bàsics com la llum i l'aigua representen alguns dels desafiaments econòmics més destacats. Aquests factors de mica en mica han començat a generar una gran preocupació entre la població, ja que impacten directament en la capacitat adquisitiva i el benestar econòmic de les famílies. És per aquest motiu que en aquest context, la gestió financera personal s'està convertint en una prioritat crucial per cada vegada més persones. Mantenir un control eficaç dels ingressos i despeses és ara més essencial que mai, perquè permet als individus adaptar-se millor als canvis, minimitzar l'impacte

dels augments dels preus i assegurar un estil de vida sostenible.

Així, sorgeix la necessitat real d'una aplicació que ofereixi una solució completa i personalitzada per a poder dur a terme una bona gestió financera, ajudant els usuaris a afrontar amb èxit els desafiaments econòmics actuals i construir un futur més estable i segur. Per respondre a aquesta necessitat, proposo el desenvolupament d'una aplicació nativa utilitzant Swift, amb un enfocament específic en la gestió dels ingressos mensuals dels usuaris. L'objectiu principal és proporcionar una eina que permeti als usuaris registrar i gestionar els seus ingressos de manera eficient, ajudant-los a distribuir-los de manera equilibrada entre diferents categories.

- E-mail de contacte: joelanglesroca1@gmail.com
- Menció realitzada: Enginyeria del Software
- Treball tutoritzat per: Yolanda Benitez Fernandez (Departament de Ciències de la Computació)
- Curs 2023/24

2 ESTAT DE L'ART

A continuació, analitzo algunes de les aplicacions existents que ofereixen funcionalitats similars a la idea del projecte que he desenvolupat. Després d'una investigació inicial a l'App Store d'iOS, he identificat dues aplicacions d'aquest sector que destaquen per sobre de la resta: Monefy [2] i "Gestor de gastos, ingresos" [3].

Totes dues aplicacions segueixen una estructura similar en el seu funcionament. Permeten als usuaris seleccionar diverses categories tant per als ingressos com per a les despeses. No obstant això, cal destacar que Monefy limita la creació d'aquestes categories en la seva versió gratuïta. Tant Monefy com "Gestor de gastos, ingresos" ofereixen funcionalitats com la visualització de dades per dia, setmana, mes, any o interval de temps específic. A més, proporcionen opcions per a seleccionar el tipus de moneda, l'idioma i el tema de la pantalla (clar o fosc).

Personalment, faig servir l'aplicació Monefy en el meu dia a dia, ja que fins ara encara no havia desenvolupat la meua pròpia solució. Trobo que visualment és més intuïtiva i comprensible que l'alternativa mencionada, perquè ofereix una comprensió més clara del funcionament de l'aplicació i de les dades mostrades. No obstant això, "Gestor de gastos, ingresos" destaca per una característica interessant: la capacitat de generar gràfics dels ingressos, despeses, beneficis o pèrdues en el període de temps definit per l'usuari. Són dues aplicacions força aconseguides d'aquí les bones valoracions:

- Monefy: 2700 valoracions amb una mitjana de 4,6/5
- Gestor de gastos, ingresos: 1200 valoracions amb una mitjana de 4,9/5

Considero que aquests exemples han estat una font d'inspiració valuosa per al meu projecte. No només m'han permès obtenir idees funcionals, sinó també d'aspectes relacionats amb el disseny. Tot i això, la particularitat de la meua solució radica en la capacitat de fraccionar els ingressos, la qual cosa la diferencia de les opcions existents al mercat.

3 OBJECTIUS

En els punts anteriors exposo alguns dels objectius fonamentals del projecte. Ara, detallaré tant els objectius principals com els secundaris per a una millor comprensió i visualització dels resultats que es pretenen assolir.

3.1 Objectius principals

Els objectius principals formen el nucli de l'aplicació, enfocant-se en les parts essencials per a una gestió financera eficient i personalitzada. Els objectius principals definits per aquest projecte són els següents:

- Desenvolupar una aplicació nativa utilitzant Swift i Core Data.
- Proporcionar una solució completa i personalitzada per a una bona gestió financera, ajudant els usuaris a distribuir els seus ingressos de manera equilibrada i a registrar les diferents despeses.

- Permetre als usuaris fraccionar els ingressos en categories clares i específiques, concretament "Necessitats", "Oci", "Inversions" i "Estalvis".
- Les despeses es podran modificar i eliminar. Serà necessari indicar el cost de la despesa, el tipus (com ara alimentació, tecnologia, salut, etc.), la data en què es va realitzar i opcionalment un comentari per aclarir els detalls de la transacció.

3.2 Objectius secundaris

Els objectius secundaris se centren en millorar l'experiència d'usuari, assegurant que l'aplicació sigui accessible, còmoda i fiable. Els objectius secundaris definits per aquest projecte són els següents:

- Afegir suport per a diversos idiomes a l'aplicació per millorar la seva accessibilitat i usabilitat global.
- Incloure un mode de tema fosc per millorar la comoditat visual dels usuaris i reduir la fatiga ocular.
- Centrar-se en el desenvolupament de proves automàtiques al frontend, especialment en el ViewModel, per garantir la fiabilitat i estabilitat de l'aplicació i assegurar una experiència d'usuari òptima.

4 METODOLOGIA DE TREBALL

Per dur a terme aquest projecte, he emprat una metodologia de treball basada en el model Waterfall Iteratiu. Aquest enfocament combina elements del model Waterfall tradicional amb la flexibilitat de les metodologies iteratives, proporcionant diversos avantatges significatius en el desenvolupament d'aplicacions. Característiques claus que he aprofitat de la metodologia Waterfall Iteratiu:

1. Segmentació del procés: El projecte es divideix en fases ben definides, com ara recopilació de requisits, disseny, implementació i proves, cada una completada abans de passar a la següent.
2. Flexibilitat: Tot i seguir una estructura ordenada, el model Waterfall Iteratiu permet la revisió i modificació de requisits durant el procés de desenvolupament, adaptant-se a canvis necessaris.

Amb aquesta metodologia, he pogut adaptar el desenvolupament de l'aplicació a les necessitats emergents i canvis en els requisits del projecte, assegurant la flexibilitat i la qualitat del resultat final. Això m'ha permès una millor gestió del temps i dels recursos, així com una major eficàcia en l'entrega d'una solució òptima i satisfactòria.

5 PLANIFICACIÓ

Per comprendre l'evolució d'aquest projecte en termes de planificació, podem dividir-lo en dues etapes clau:

1. Durant la etapa inicial, em vaig dedicar a recollir els requisits dels usuaris i a dissenyar l'aplicació. Això va

incloure la creació del diagrama de casos d'ús, el model relacional de la base de dades i el disseny de la interfície d'usuari. En aquesta etapa, vaig poder adherir-me estrictament a la planificació sense cap complicació ni desviació notable.

Durant aquest període em vaig fer una primera planificació que mostro a continuació:



Fig. 1: Planificació provisional durant la primera etapa

2. En la segona etapa, que abasta des de la implementació fins l'acabament del projecte, vaig haver de fer alguns ajustos a la planificació inicial. Vaig allargar el temps dedicat a la implementació en 4 dies addicionals i vaig reduir la durada de les proves per centrar-me en altres aspectes prioritaris. A més, vaig incloure una nova tasca: la preparació de l'Informe Provisional, que la vaig considerar de vital importància per garantir una entrega final tranquil·la.
- Tot seguit es pot observar com va quedar la nova planificació en aquesta segona etapa:



Fig. 2: Replanificació durant la segona etapa

6 ANÀLISI I DISSENY

En aquest apartat presento els requisits de l'aplicació, dividits en dues categories: requisits funcionals i no funcionals. Aquests requisits defineixen les característiques i les condicions que ha de complir l'aplicació per satisfer les necessitats dels usuaris.

A més, també mostro el disseny inicial de l'aplicació el qual m'ha estat de gran ajuda, sobretot per concretar les primeres idees i tenir més clar l'estil abans de començar a programar.

6.1 Requisits funcionals

A continuació, presento una taula que detalla els requisits funcionals de l'aplicació. És important destacar que del RF-7 al RF-9, incloc els requisits que cobreixen els objectius secundaris, oferint característiques addicionals per millorar l'experiència de l'usuari i la flexibilitat de l'aplicació en diversos contextos d'ús.

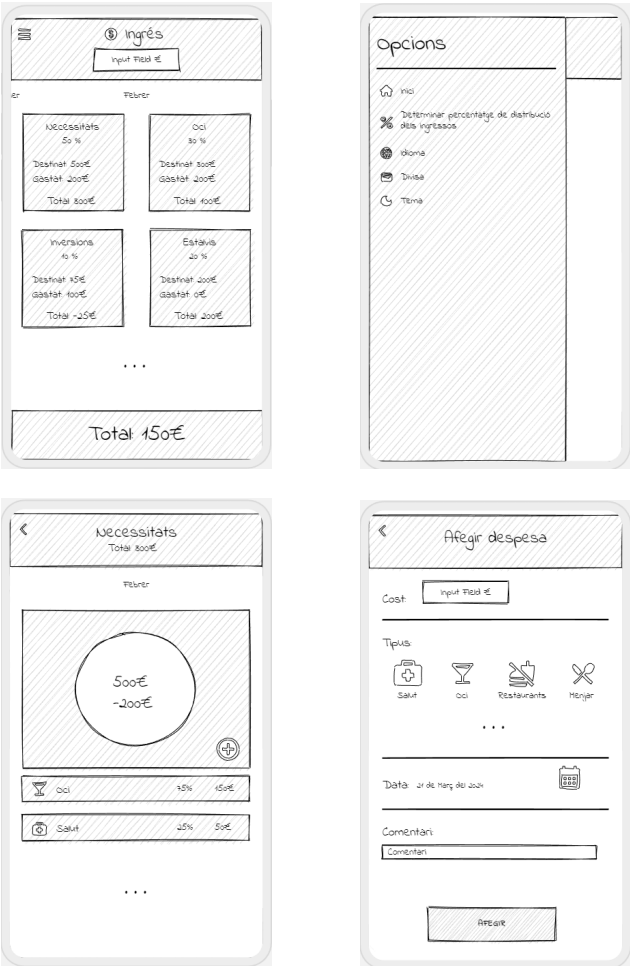
Id	Requisit	Prioritat
RF-1	L'aplicació ha de permetre als usuaris distribuir els ingressos entre les següents categories: necessitats, oci, inversions i estalvis.	5
RF-2	L'aplicació ha de permetre a l'usuari introduir les despeses associades a cada categoria. Cada despesa ha de tenir un cost, un tipus, una data i opcionalment un comentari.	5
RF-3	L'usuari ha de poder introduir l'ingrés mensual.	4
RF-4	L'usuari ha de poder determinar en quin tant per cent vol dividir els seus ingressos per cada categoria.	4
RF-5	L'usuari ha de poder navegar entre els diferents mesos per visualitzar les dades de cada període de temps de manera clara i senzilla.	3
RF-6	L'aplicació ha de permetre a l'usuari modificar o eliminar les despeses associades a cada categoria.	3
RF-7	L'aplicació ha de permetre l'execució de proves automàtiques al frontend amb un focus especial en el ViewModel, amb l'objectiu de garantir la fiabilitat i estabilitat de l'aplicació.	2
RF-8	L'aplicació ha de suportar múltiples idiomes (català, castellà i anglès). L'idioma s'adaptarà automàticament a l'idioma del sistema.	1
RF-9	L'aplicació ha de incloure un mode de tema fosc per millorar la comoditat visual dels usuaris i reduir la fatiga ocular.	1

6.2 Requisits no funcionals

Id	Requisit
RNF-1	L'aplicació ha de respondre de manera eficient i ràpida en la mesura de lo possible.
RNF-2	L'aplicació ha de ser fàcil d'utilitzar i intuïtiva per als usuaris finals, amb una interfície clara i senzilla.
RNF-3	L'aplicació ha de ser compatible amb els dispositius mòbils iOS més comuns.
RNF-4	L'aplicació ha de tenir un codi organitzat i estructurat de manera clara per facilitar-ne la comprensió i permetre'n futures modificacions. Aquest objectiu es complirà aplicant l'arquitectura Hexagonal i el patró Model-View-ViewModel.

6.3 Disseny

Tot seguit, mostro el disseny d'algunes pantalles de l'aplicació. He procurat dissenyar-la de manera que compleixi tant amb els requisits funcionals com amb els no funcionals. L'aplicació està concebuda amb la mínima quantitat de botons possible per facilitar-ne l'ús, alhora que mostra només la informació més rellevant per millorar la seva comprensibilitat.



7 DESENVOLUPAMENT

Tal com estava indicat en la planificació inicial, vaig començar el desenvolupament el dia 16 d'abril. En primer lloc, just després de crear el projecte, vaig començar a dissenyar la pantalla principal. Aquesta pantalla té diversos components visuals, així que vaig decidir crear aquests components per separat i després ajuntar-los. D'aquesta manera, vaig poder facilitar la comprensió del codi i la detecció d'errors. Aquesta ha estat la filosofia que he utilitzat al llarg del projecte en aquelles pantalles que eren una mica més complicades. A continuació, mostraré un exemple de com vaig construir aquesta pantalla principal pas a pas:

- 1. Identificació dels components visuals:** Vaig analitzar la pantalla principal per identificar els diferents components visuals necessaris, com ara botons, menús, i àrees de visualització de dades.
- 2. Creació dels components:** Vaig crear cada component per separat en fitxers individuals, assegurant-me que cadascun funcionés de manera autònoma.

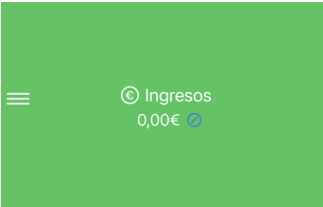


Fig. 5: Capçalera de la pantalla



Fig. 6: Component per seleccionar data



Fig. 7: Component d'una categoria

3. **Integració dels components:** Un cop creats, vaig integrar tots els components en la pantalla principal, comprovant que treballessin correctament junts.



Fig. 8: Pantalla principal

Una vegada vaig tenir acabada la pantalla principal a nivell visual, vaig procedir a implementar tota la lògica que podia ser integrada en aquell moment. Aquesta lògica va ser implementada al ViewModel, ja que és el responsable exclusivament de la lògica relacionada amb una única vista associada a ell. En aquest punt, m'agradaria destacar que totes les dades visuals les extreia dels mocks que vaig crear, un mock és una simulació d'una part del sistema amb l'objectiu de provar el comportament d'una part del codi. D'aquesta manera, vaig evitar la necessitat de crear la base de dades en etapes inicials, amb els possibles canvis posteriors, ja que algunes taules encara no les tenia del tot clares.

La següent vista que vaig crear va ser la del menú d'opcions. De primeres, semblava bastant senzilla, però la possibilitat d'indicar com l'usuari vol distribuir els seus ingressos entre les diferents categories em va complicar una mica la situació, ja que tècnicament ho vaig trobar difícil. Així doncs, vaig acabar tornant a crear una vista única i exclusiva per a aquest propòsit, facilitant així el procediment.

A continuació es pot observar el resultat final de la pantalla del menú:

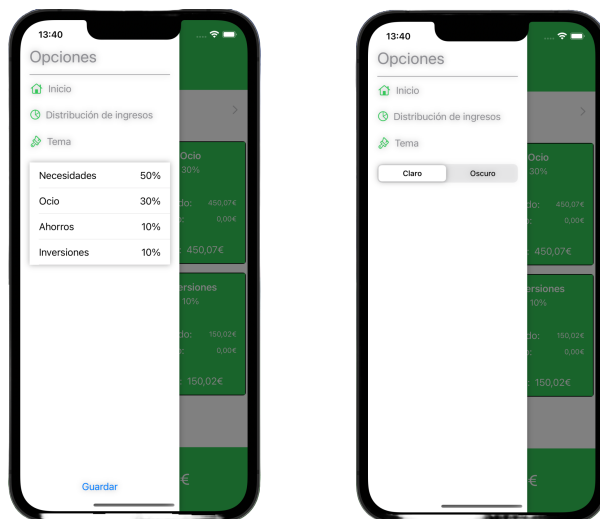


Fig. 9: Pantalla del menú d'opcions

En aquest punt del desenvolupament, ja tenia força avançada la part de poder distribuir els ingressos entre les diferents categories. Tot i això, aquesta funcionalitat no tenia gaire sentit si no podia afegir despeses. Per aquest motiu, vaig començar a construir la següent pantalla. La idea d'aquesta nova pantalla era que l'usuari pogués veure una mica més d'informació general sobre els diners gastats en cada tipus de despesa dins d'una categoria concreta, i que des de la mateixa pogués tant afegir una nova despesa com entrar al detall del tipus de despesa.

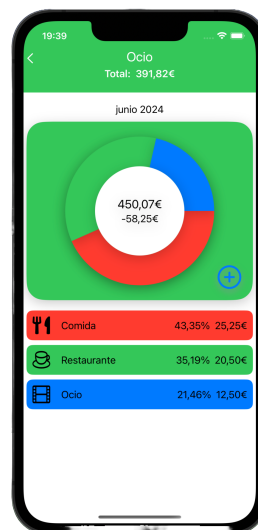


Fig. 10: Pantalla de detall de categoria

Com es pot observar per tal de fer-la més atractiva i entenedora, vaig decidir incloure un gràfic de suport visual que permetés interpretar els valors de forma ràpida i senzilla. Aquest gràfic no va ser fàcil d'implementar, així que també vaig separar la vista en un altre fitxer per poder enfrontar-me als problemes de manera més controlada i directa.

Tot i que a la Figura 10 es poden apreciar diferents tipus de despeses amb el valor dels diners destinats a cadascun d'ells, en aquest moment del desenvolupament encara no tenia implementada la lògica per afegir una despesa. Per

tant, tot i que no estava del tot obligat, vaig decidir que la següent pantalla que havia de fer era la d'afegir una nova despesa. Així, també podria verificar i testear les pantalles amb dades reals en lloc de dades simulades. Recordem que una despesa havia de tenir sí o sí un cost, un tipus, una data i, opcionalment, una descripció. Per tant, aquesta pantalla la vaig construir exactament amb aquests camps.

Tot seguit podeu observar el resultat:

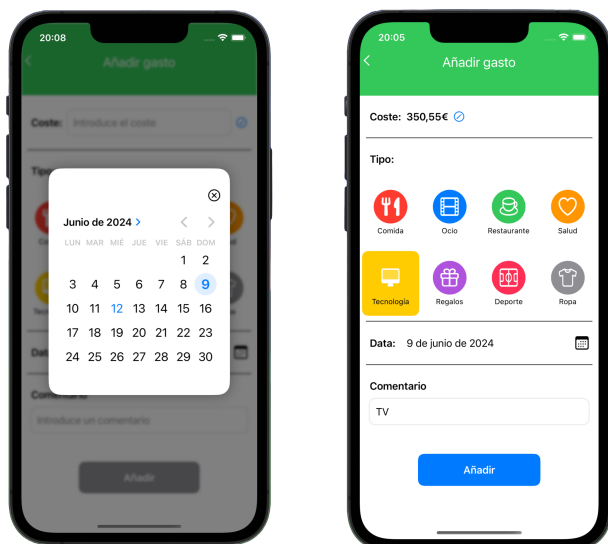


Fig. 11: Pantalla d'afegir despesa

Una part important de la lògica, que alhora em va resultar complicada, va ser la de permetre que l'usuari només pogués seleccionar un dia del mes en el qual es trobava en la pantalla principal. Aquesta pantalla principal tenia una opció per navegar de mes en mes, i volia assegurar-me que les despeses només es poguessin assignar a dies dins del mes seleccionat.

Finalment, per donar per acabada la construcció de les pantalles, vaig haver de crear aquella encarregada de mostrar totes les diferents despeses que un usuari realitza per cada tipus de despesa. Com quasi totes les pantalles al principi semblava fàcil, però no va ser fins que vaig acabar-la que em vaig adonar que durant la redacció inicial dels requisits no vaig pensar en la opció de poder eliminar i modificar una despesa. És per aquest motiu que en aquest projecte vaig optar per la metodologia iterativa, perquè em va permetre modificar o afegir requisits durant el desenvolupament. D'aquesta manera, vaig poder abordar els canvis que no havia considerat fins a l'etapa d'implementació.

En el resultat final de la pantalla hi ha dues opcions que pot fer l'usuari: mantenir clicada una despesa, de manera que aparegui l'opció d'eliminar-la, o bé clicar sobre la despesa, la qual cosa el portaria a la mateixa pantalla d'afegir despesa, però canviant el títol i el nom del botó, a més de precarregar les dades ja existents.

Tot seguit es pot observar el resultat:

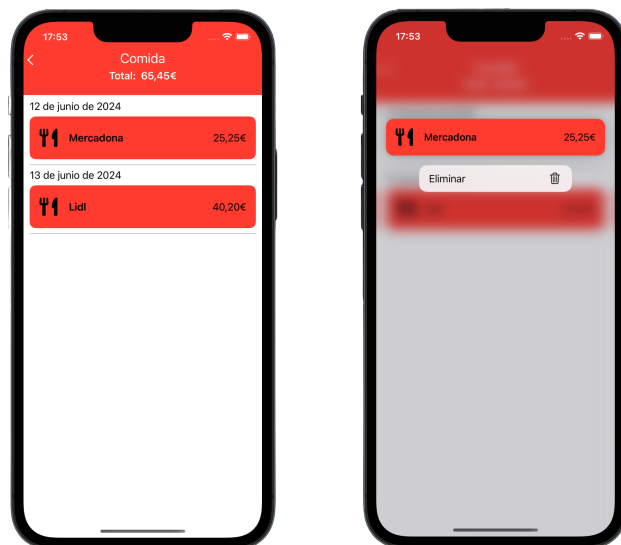


Fig. 12: Pantalla de detall de despesa

8 PROVES

Per garantir la qualitat i la robustesa de l'aplicació, he realitzat diverses proves que es poden agrupar en tres categories principals: proves d'usabilitat, proves pilot i proves unitàries.

8.1 Proves d'usabilitat

Les proves d'usabilitat les vaig dur a terme amb dos grups diferents de participants. El primer grup estava format per companys de carrera amb un alt nivell de coneixements informàtics. Aquest grup va proporcionar un feedback tècnic valuós sobre la interfície d'usuari, la navegació i l'experiència general de l'aplicació.

El segon grup estava format per amics sense tants coneixements d'informàtica. Aquestes proves eren essencials per avaluar la facilitat d'ús i la intuïció de l'aplicació des de la perspectiva d'usuaris menys tècnics. Les seves opinions van ajudar a identificar punts de millora en la usabilitat i l'accessibilitat.

8.2 Prova pilot

A més de les proves d'usabilitat, he fet una prova pilot en la qual he estat utilitzant l'aplicació durant un mes amb dades reals d'ingressos i despeses. Aquesta prova m'ha permès avaluar el funcionament de l'aplicació en un entorn real i detectar possibles errors o problemes en la gestió diària de les finances personals.

8.3 Proves unitàries

Finalment, també vaig dur a terme proves unitàries en els dos ViewModels més importants de l'aplicació. Em vaig centrar especialment en aquells que contenen camps per introduir quantitats de diners, ja que era crucial assegurar-se que l'aplicació pogués gestionar correctament valors atípics o erronis. Les proves unitàries van garantir que l'aplicació

no deixés de funcionar ni es comportés de manera inesperada quan els usuaris introduïen valors inusuals.

Aquestes proves exhaustives han sigut fonamentals per garantir que l'aplicació fos fiable, fàcil d'utilitzar i preparada per a un ús real.

9 LÍNIES OBERTES

Durant el desenvolupament de l'aplicació, és cert que hi ha hagut moments en què m'he plantejat afegir noves funcionalitats o millorar-les. No obstant això, havia de ser conscient que el temps era limitat i que, preferiblement, era millor no complicar-se més del compte. Algunes d'aquestes funcionalitats són les següents:

- Actualment, les categories per classificar els teus ingressos mensuals estan limitades. Per tal de millorar l'aplicació, una possible millora en el futur seria oferir l'opció de poder afegir categories personalitzades per l'usuari o afegir-ne més perquè l'usuari pugui seleccionar les que cregui necessàries.
- Ens trobem en la mateixa situació amb els tipus de despesa. Una altra millora seria augmentar la quantitat de tipus de despesa disponibles, com per exemple: casa, cotxe, emergències, etc.
- També vaig pensar en predefinir per defecte els tipus de despesa que solen estar associats a una categoria. Per exemple, en la categoria de "Necessitats", a l'afegir una despesa, que apareguessin els tipus de despesa que més s'hi associen (casa, menjar, cotxe, etc.).
- Tot i que el disseny finalment m'ha acabat convenent força, inicialment era una de les coses que menys m'agradaven. Per tant, també em plantejaria millorar alguns aspectes visuals.

10 CONCLUSIONS

Ara que he donat per acabat aquest treball de final de grau, em permeto reflexionar sobre l'esforç realitzat i les aportacions que he extret d'aquest procés. En primer lloc, a nivell personal, ha estat un repte significatiu, especialment considerant que també estava treballant a jornada parcial i fent dues assignatures més. Tot i això, considero que he gestionat aquesta situació amb èxit i que els resultats han sigut satisfactoris. M'orgulleixo de la meua capacitat per organitzar-me i gestionar l'estrès en setmanes carregades de treball. Destacaria també la satisfacció de poder desenvolupar una aplicació des de zero.

Tot i haver hagut de fer alguns ajustos a la planificació, és important destacar que no he experimentat problemes greus en el transcurs del projecte. En general, he pogut avançar amb èxit, mantenint-me en línia amb els objectius i les expectatives establertes inicialment.

Pel que fa a l'aspecte professional, aquest projecte m'ha fet evolucionar com a desenvolupador frontend i d'aplicacions mòbils, així com ampliar els meus coneixements tècnics en SwiftUI i CoreData. A més, no descarto que aquesta aplicació pugui ser una valuosa eina per mostrar en el meu perfil professional.

Estic molt satisfet d'haver cursat aquesta carrera i d'haver pogut estudiar allò que sempre havia desitjat. Tot i que ha estat un camí complicat, amb alts i baixos, considero que he afrontat els reptes amb determinació i els he superat de la millor manera possible. No puc deixar de sentir-me profundament agraït pel privilegi que m'ha brindat aquest grau, permetent-me accedir al món laboral fins i tot abans de la seva finalització. A partir d'ara, tinc l'oportunitat de centrar-me plenament en la meua carrera professional, la qual s'entreu com a molt atractiva i prometedora.

AGRAÏMENTS

Vull expressar el meu més sincer agraïment a totes les persones que han fet possible la realització d'aquest projecte.

En primer lloc, voldria agrair a la meua tutora, Yolanda Benitez, per la seva mentorització i orientació al llarg de tot el procés.

També vull donar les gràcies als meus companys de feina per la seva ajuda puntual en resoldre dubtes que han sorgit durant el desenvolupament de l'aplicació.

Un agraïment especial als meus amics i companys de carrera per dedicar una part del seu temps a ajudar-me en les proves de testing.

Finalment, voldria expressar la meua gratitud a la meua família pel seu suport al llarg de tot el projecte.

A tots vosaltres, moltes gràcies.

REFERÈNCIES

- [1] ChatGPT. (s/f). Consultes i millores gramaticals. <https://chat.openai.com/>
- [2] Reflectly ApS. (20 de febrer del 2024). Monefy: Control de gastos. <https://apps.apple.com/es/app/monefy-control-de-gastos/id1212024409>
- [3] Orange Dog. (20 de febrer del 2024). Gestor de gastos, ingresos. <https://apps.apple.com/es/app/gestor-de-gastos-ingresos/id1510997753>
- [4] Canva. (29 de febrer del 2024). Creació del calendari per a la planificació. <https://www.canva.com/>
- [5] PlantUML. (19 de març del 2024). Creació del diagrama de casos d'ús i del model relacional. <https://plantuml.com/es/>
- [6] Stack Overflow. (19 de març del 2024). Consulta sobre unitats de mesura per les vides/monedes. <https://stackoverflow.com/questions/2376853/core-data-decimal-type-for-currency>
- [7] Uizard. (s/f). Creació del disseny de la UI i UX de l'aplicació. <https://app.uizard.io/prototypes>
- [8] App Icon Generator. (20 de maig del 2024). Generar l'icona de l'app en diferents tamanys adaptats per a iPhone. <https://www.appicon.co/>

- [9] Developer. (20 de maig del 2024). Documentació sobre com configurar la icona de l'aplicació. <https://developer.apple.com/documentation/xcode/configuring-your-app-icon>
- [10] Flaticon. (20 de maig del 2024). Icona de l'app sense copyright i amb permisos per a projectes comercials i personals. https://www.flaticon.com/free-icon/money_3427292
- [11] Yet Another Swift Blog. (s/f). Documentació sobre patrons de comunicació entre vistes a SwiftUI. <https://www.vadimbulavin.com/passing-data-between-swiftui-views/>
- [12] Youtube. (s/f). Canal sobre coneixements de SwiftUI i exemples pràctics. <https://www.youtube.com/@SwiftlyPlace>

APÈNDIX

Disseny/prototip de l'aplicació

He realitzat el disseny de l'aplicació utilitzant una plataforma que ofereix la possibilitat de crear fins a cinc pantalles de forma gratuïta. Tot i no haver explorat altres opcions, aquesta quantitat de pantalles ha sigut suficient per representar les funcionalitats principals de l'aplicació. Com que es tracta d'un prototip executable, he decidit fer-lo públic perquè qualsevol persona que accedeixi al link pugui provar-lo: <https://app.uizard.io/p/ec175b6a>

Diagrama de casos d'ús

A continuació, es presenta la versió final del diagrama de casos d'ús, on l'únic actor involucrat és l'usuari de l'aplicació, ja que serà l'única entitat que realitzarà accions en el sistema:



Model relacional de base de dades

El model relacional de la base de dades proporciona una representació estructurada de com les dades estan organitzades i relacionades dins de l'aplicació, facilitant la comprensió i gestió eficient dels recursos de la informació. A continuació es mostra la versió definitiva del model relacional:

