

Trabajo de fin de grado
Grado en Ingeniería Informática

Planificación de gastos personales mediante acceso a notificaciones y datos bancarios

Planning personal expenses through banking information and notifications

Por
Juan Francisco Carrión Molina

Dirigido por
Gonzalo Rubén Méndez Pozo

Codirigido por
Eugenio Pablo Concepción Cuevas



UNIVERSIDAD
COMPLUTENSE
MADRID

Universidad Complutense de Madrid
Facultad de Informática
Departamento de Ingeniería del Software e Inteligencia Artificial

Madrid, septiembre de 2023

Planificación de gastos personales mediante acceso a notificaciones y datos bancarios

*Planning personal expenses through banking information
and notifications*

Trabajo de fin de Grado en Ingeniería Informática
Convocatoria de septiembre de 2023

Autor

Juan Francisco Carrión Molina

Director

Gonzalo Rubén Méndez Pozo

Codirector

Eugenio Pablo Concepción Cuevas

Universidad Complutense de Madrid
Facultad de Informática
Departamento de Ingeniería del Software e Inteligencia Artificial

Madrid, septiembre de 2023

Agradecimientos

A mis pilares. Mis hermanas, mi madre. Markel, Carmen, Manu, Pau, Nacho, Álvaro, Arturo, Mariam. Elvira. Por ser, estar y recordarme que podía.

A mis maestros de software y del día a día, Eugenio, Naucé, Rocío. A mi tutor, Gonzalo.

Resumen

Cada día nos enfrentamos a más y más pagos de múltiples tipos a través de diferentes canales. El auge de los pagos en línea, la aparición de cada vez más servicios de suscripción y la proliferación de productos financieros digitales hacen que día a día las personas tengan que hacer frente a una gran cantidad de gestiones en el ámbito bancario.

De manera paralela y acompañando al resto de la sociedad, el sector bancario y financiero ha vivido en los últimos años una transformación digital orientada a la gran disponibilidad de información y datos, lo cual ha obligado a las entidades bancarias a adaptar sus servicios para manejar y ofrecer dicha información de una manera más abierta.

El presente trabajo se centra, en primer lugar, en analizar tres tecnologías derivadas de dicha transformación, como son los agregadores bancarios, las notificaciones vía correo electrónico y la exportación de transacciones personales a hojas de cálculo. En segundo lugar, se estudian las nuevas necesidades de las personas con respecto a gastos personales.

Finalmente, se implementa una solución que cubra las nuevas necesidades de las personas, liberándolas así de las tediosas gestiones de la contabilidad y los pagos, para poder disfrutar plenamente de los productos y servicios que con ellos adquieren. La solución construida consiste en una aplicación de gestión de gastos personales dinámica e intuitiva, enriquecida con información proporcionada de manera automática por parte de las instituciones bancarias.

Palabras clave

Tecnología financiera, Fintech, Agregadores bancarios, Pagos, Gastos personales, Ingeniería del Software, Aplicaciones web, Arquitecturas de microservicios.

Abstract

Day to day we face more and more payments of multiple types, made through different channels. The rise of online payments, the emergence of subscription-based services, and the expansion of digital financial products, are making people have to deal with a large number of banking transactions on a daily basis.

At the same time, and hand in hand with the rest of society, the banking and financial sector has undergone a digital transformation in recent years focused on the availability of information and data. This has forced banking institutions to adapt their services to handle and expose this information in a more open way.

This project focuses, firstly, on analyzing three technologies derived from said transformation, such as banking aggregators, email notifications, and the export of personal transactions to spreadsheets. Secondly, an analysis is carried out of the new needs of people in relation to personal expenses.

Finally, the work developed leads to a solution that meets these new needs of people, thus freeing them from the tedious management tasks of accounting and expenses, so that they can fully enjoy the goods and services that they acquire in relation to said expenses. The resulting solution consists of a dynamic and intuitive personal expense management application, enriched with information automatically provided by banking institutions.

Keywords

Financial technology, Fintech, Banking aggregators, Payments, Personal expenses, Software Engineering, Web applications, Microservices Architecture.

Índice general

1. Introducción	1
1.1. Motivación	2
1.2. Objetivos	2
1.2.1. Objetivo principal	2
1.3. Metodología	3
1.4. Estructura de la memoria	3
1. Introduction	5
1.1. Motivation	6
1.2. Objectives	6
1.2.1. Main objective	6
1.3. Methodology	7
1.4. Structure of the report	7
2. Estado de la cuestión	9
2.1. Estado de la cuestión	10
2.1.1. El ecosistema de las <i>fintechs</i>	10
2.2. Regulación y sistemas de información bancaria	13
2.2.1. Sistemas de información bancaria, <i>open banking</i> y regulación europea	13
2.2.2. Notificaciones bancarias y de proveedores de servicios	15
2.2.3. Suministro de la información en lote de forma manual a través de ficheros de datos	16
3. Análisis funcional	18
3.1. Introducción	19
3.2. Análisis de requisitos	20
3.2.1. Hipótesis de personas	20
3.2.2. Selección de usuarios para las encuestas	20
3.2.3. Diseño de las encuestas	20
3.2.4. Realización de las encuestas	26
3.2.5. Resultados de las encuestas y extracción factual	26
3.3. Funcionalidad	33
3.3.1. Extracción de requisitos funcionales	33
3.3.2. Alcance funcional	35
3.3.3. Historias de usuario y casos de uso	36
4. Diseño del sistema	41
4.1. Nombre y marca del sistema	42

4.2.	Arquitectura del sistema	43
4.2.1.	Introducción	43
4.2.2.	Microservicios	43
4.2.3.	Definición de la arquitectura	44
4.2.4.	Organización por capas	47
4.2.5.	Microservicio Core	48
4.2.6.	Microservicio conector bancario	49
4.2.7.	Aplicación web	49
4.3.	Modelo de datos	50
4.3.1.	Gasto	50
4.3.2.	Gasto puntual	50
4.3.3.	Gasto recurrente	51
4.3.4.	Pago	51
4.3.5.	Previsión por periodo	52
4.4.	Asociación transacción-gasto y aplicación de cambios	53
4.4.1.	Asociación entre una transacción bancaria y un gasto registrado . . .	53
4.4.2.	Aplicación de cambios a un gasto a partir de una transacción	54
5.	Desarrollo e implementación del sistema	56
5.1.	Lenguajes, <i>frameworks</i> y protocolos de comunicación	57
5.1.1.	Backend: Kotlin y Spring Framework	57
5.1.2.	Comunicación síncrona en el backend: API REST a través de HTTP	60
5.1.3.	Comunicación asíncrona en el backend: mensajes a través de AMQP	60
5.1.4.	Frontend web: React	60
5.2.	Metodología de desarrollo, pruebas y gestión de documentación	61
5.2.1.	Control de versiones: Git y GitHub	61
5.2.2.	Desarrollo por tareas: sistema de tarjetas o <i>kanban</i>	61
5.2.3.	Pruebas o <i>testing</i>	61
5.3.	Desarrollo de los sistemas y aplicaciones	62
5.3.1.	Microservicio conector bancario	62
5.3.2.	Microservicio Core	68
5.3.3.	Aplicación web	68
5.4.	Resultados de la implementación	71
5.4.1.	Microservicio Core	71
5.4.2.	Microservicio conector bancario	71
5.4.3.	Aplicación web	71
6.	Conclusiones y trabajo futuro	76
6.1.	Planteamiento de las conclusiones sobre objetivos y resultados	77
6.2.	Conclusiones	77
6.3.	Trabajo futuro	81
6.	Conclusions and future work	82
6.1.	Approach to conclusions over objectives and results	83
6.2.	Conclusions	83
6.3.	Future work	87
	Bibliografía	88

A. Comparativa de agregadores bancarios de tipo proveedor AIS de open banking	90
B. Formularios de las encuestas del análisis de requisitos	92
C. Respuestas a las encuestas del análisis de requisitos	97
D. Hojas de cálculo proporcionadas por las personas encuestadas	106

Índice de figuras

4.1.	Logotipo del proyecto Balancer	42
4.2.	Diagrama de un ejemplo de arquitectura de microservicios	44
4.3.	Diagrama de modelo C4 del contexto del sistema	45
4.4.	Diagrama de modelo C4 de contenedores	46
4.5.	Diagrama de ejemplo de un servicio con organización por capas	48
4.6.	Diagrama de flujo del algoritmo de aplicación de cambios a un gasto en función de una transacción	55
5.1.	Diagrama de la estructura general con el gestor de proyectos Maven, a seguir en todos los microservicios Kotlin del proyecto	58
5.2.	Diagrama de secuencia que muestra la creación de un Item de Plaid y la obtención de su token de acceso	64
5.3.	Diagrama de secuencia que muestra la recepción de un <i>webhook</i> por parte de Plaid, el preprocesamiento de los datos y el envío al microservicio Core	65
5.4.	Vista de inicio. Esta vista tiene relación con la funcionalidad de consultar todos los gastos registrados (HU2), siendo un punto de entrada a la aplicación con un botón para navegar a la vista del listado.	72
5.5.	Vista de gastos con detalles de gastos puntuales. Tiene relación con la funcionalidad de consultar todos los gastos registrados (HU2). Además, la Figura 5.5a muestra un indicador de estado actualizado, relativo a la HU5.	73
5.6.	Vista de gastos con detalles de gastos recurrentes. Tiene relación con la funcionalidad de consultar todos los gastos registrados (HU2). Además, la Figura 5.6b muestra un indicador de estado actualizado, relativo a la HU5.	73
5.7.	Vista para añadir un gasto puntual. Tiene relación con la funcionalidad de añadir un gasto puntual (HU1).	74
5.8.	Vista para añadir un gasto recurrente. Tiene relación con la funcionalidad de añadir un gasto recurrente (HU2).	74
5.9.	Vistas de cuentas bancarias y para asociar una cuenta bancaria. Tienen relación con la funcionalidad de listar todas las cuentas bancarias asociadas (HU4) y la de asociar una nueva (HU3).	75
5.10.	Vista para asociar una cuenta mostrando Plaid Link y un resultado satisfactorio. Tiene relación con la funcionalidad de asociar cuenta bancaria (HU3).	75

Capítulo 1

Introducción

1.1. Motivación

La digitalización, el auge de los pagos en línea y la aparición de cada vez más tipos de servicios adquiribles, especialmente las suscripciones, están generando un aumento importante en la cantidad de gestiones financieras diarias, semanales y mensuales de las personas. De este aumento deriva una necesidad también creciente de prever los gastos personales de cada uno, con el fin de obtener una planificación financiera personal.

A un ritmo similar y a raíz también de la digitalización, numerosos servicios se han adaptado en los últimos años para proporcionar acceso de manera programática y a través de distintos tipos de interfaces a la gran cantidad de datos que generan, procesan o alojan.

Concretamente, los datos bancarios y las notificaciones de cargos y recibos a través de correo electrónico, expuestos actualmente a través de interfaces programáticas integrables, podrían ser útiles para cubrir la necesidad presentada en primer lugar en esta sección, conectando estas fuentes de información financiera con información sobre los usuarios y sus gastos a través de un sistema software que realice el procesamiento.

1.2. Objetivos

La intención de este trabajo es acercarse a una solución dirigida a cubrir a la necesidad mencionada en el apartado anterior, ofreciendo a las personas una previsión de sus gastos puntuales y recurrentes de manera sencilla y dinámica.

A partir de esta intención definida, se plantea un objetivo principal del trabajo y distintos objetivos específicos, que dan nombre a este proyecto.

1.2.1. Objetivo principal

Liberar a la persona de la burocracia, el tiempo y el esfuerzo dedicados a gestionar gastos personales puntuales y recurrentes para poder dedicar ese tiempo a disfrutar de los productos y servicios asociados a dichos gastos.

Objetivo específico C1

Desarrollar un sistema capaz de registrar los gastos del usuario, con sus características determinantes (parámetros temporales, categoría, importe, etc.), y ofrecer listados, informes, previsiones y gráficos en base a estos.

Objetivo específico C2

Desarrollar un sistema capaz de registrar gastos de tipo tanto puntual como recurrente, pudiendo al mismo tiempo mantener un estado para cada tipo (pendiente y pagado para el puntual; y activo y finalizado para el recurrente).

Objetivo específico C3

Desarrollar un sistema capaz de registrar los pagos asociados a cada gasto del usuario y actualizar el estado de dicho gasto en base a cambios en los pagos.

Objetivo específico C4

Desarrollar un sistema dotado de la capacidad automática para integrarse y recibir información bancaria a partir de una fuente de datos, procesarla y actualizar el estado de los pagos y los gastos en función de dicha información.

1.3. Metodología

Para el desarrollo del proyecto se ha establecido un plan de trabajo que comienza por analizar la situación actual relativa a los sistemas que de alguna manera cubren los mismos objetivos definidos en el apartado anterior, como son agregadores financieros personales en forma de aplicaciones para dispositivos móviles, en la mayoría de los casos. También se realiza un análisis del estado de los agregadores financieros que ofrecen sus servicios a terceros, por su relevancia como fuente de información bancaria para el sistema a desarrollar en este trabajo, y de otras dos opciones como fuentes de información

Una vez planteado el estado de la cuestión, se planifica un proceso de análisis y captura de requisitos para disponer de una base sobre la que definir el sistema y su funcionalidad. Este análisis se lleva a cabo con personas reales y a través de encuestas sobre sus hábitos financieros y gastos personales. A partir de este proceso se extrae una serie de requisitos funcionales que se formalizan después y se seleccionan y limitan, dando lugar al alcance del proyecto.

Tras definir la funcionalidad y el alcance, se procede a detallar el sistema que se va a construir, incluyendo su nombre, la arquitectura de software, el modelo de datos y la lógica para realizar el procesamiento de las transacciones en relación con los gastos.

La definición del sistema permite comenzar con la construcción técnica del mismo. Para ello, en primer lugar, se lleva a cabo una especificación de los lenguajes y tecnologías a utilizar, y se definen distintas tareas detalladas basadas en la funcionalidad que se quiere conseguir.

Siguiendo la arquitectura y las tareas definidas, y haciendo uso de los lenguajes y tecnologías especificados, se desarrolla un sistema software que se planteará como solución a las necesidades y los objetivos definidos más arriba.

1.4. Estructura de la memoria

La estructura de este documento sigue, como base, la metodología y el plan de trabajo definidos en el apartado anterior. Seis capítulos plasman el trabajo seguido a lo largo del desarrollo de este proyecto.

Se ha intentado seguir una progresión sobre el nivel de tecnicidad de la memoria, comenzando con capítulos más teóricos y analíticos, y llegando a detalles de implementación y tecnología más cercanos al propio código, que se adjunta a este documento a través de distintos repositorios en línea, indicados en cada capítulo correspondiente.

Comenzando por el segundo capítulo, se presenta un análisis del estado de la cuestión sobre agregadores bancarios existentes, tanto personales, que tienen similitud con el sistema objeto de este trabajo, como proveedores de información para terceros, que tienen relación con los objetivos de este proyecto de manera técnica. También se analizan

otras fuentes de información alternativas, como son las notificaciones a través de correo electrónico y el suministro manual de información bancaria a través de ficheros de hojas de cálculo.

Durante el tercer capítulo se especifica el proceso de análisis y captura de requisitos, comenzando por la definición de la metodología, pasando por la selección de personas y la realización de las encuestas, y finalizando con el procesamiento de los resultados de las encuestas y la extracción de funcionalidad.

El cuarto capítulo, en el que se cuenta con el sistema ya definido de manera teórica, cubre la especificación del mismo como sistema software. Dentro de esta parte de la memoria se recogen algunas decisiones técnicas de más alto nivel y relativas a una visión general del sistema, como son la arquitectura, el modelo de datos y los algoritmos de procesamiento entre transacciones y gastos, en un nivel de abstracción medio.

En el capítulo siguiente, el quinto, se procede a detallar algunas decisiones técnicas de más bajo nivel, como son el lenguaje concreto escogido, los *frameworks* y/o bibliotecas sobre los que construir los servicios de *frontend* y *backend* y una descripción relativamente concreta, sin llegar a ser el propio código, de la implementación de cada historia de usuario o caso de uso en cada parte del sistema. Este capítulo finaliza con un planteamiento de los resultados conseguidos y la funcionalidad que el sistema es capaz de cubrir.

Dicho planteamiento de resultados se utiliza en el siguiente y último capítulo, el sexto, para plantear una comparación con los objetivos fijados en el inicio del trabajo y generar, así, una serie de conclusiones. Para cerrar este capítulo y el trabajo, se ha hecho un último análisis en base al nivel de consecución de los objetivos, extrayendo distintas líneas para el trabajo futuro.

Al final de esta memoria, tras la bibliografía, se encuentran añadidos distintos apéndices como formularios o tablas, listados en un índice anteriormente y referenciados a lo largo de todo el texto.

Chapter 1

Introduction

1.1. Motivation

The global digitization process, the rise of online payments and the emergence of more and more types of acquirable services, especially the subscription ones, are generating a significant increase in the number of financial management tasks that people have to perform on a daily, weekly and monthly basis. This increase is also leading to a growing need to organize personal expenses, in order to obtain a personal financial plan.

At a similar rate and also as a result of the mentioned digitization process, a big amount of services have been adapted in recent years to provide programmatic access, through different types of interfaces, to the large amount of data that they generate, process or host.

Specifically, bank data and notifications of charges and receipts via email, currently exposed through integrable programming interfaces, could be useful to address the necessity presented at the beginning of this section, by connecting these financial information sources with information about users and their expenses through a software system that performs the processing.

1.2. Objectives

The intention of this work is to build a solution that approaches the need mentioned in the previous section, by providing people with a simple and dynamic forecast of their one-time and recurring expenses.

Based in this goal, a main objective or aim of the work is set along with different specific objectives. These give this project its name.

1.2.1. Main objective

To free the person from the bureaucracy, time and effort required to manage one-time and recurring personal expenses, so that they can dedicate that time to actually enjoying the goods and services associated with those expenses.

Specific objective C1

To develop a system capable of registering the user's expenses, with their specific properties (temporal parameters, category, amount, etc.), and to generate lists, reports, plans and graphs based on these.

Specific objective C2

To develop a system capable of registering expenses of both one-time and recurring types, while being able to maintain a status for each type (pending and paid for the one-off; and active and finished for the recurring).

Specific objective C3

To develop a system capable of registering the payments associated with each user expense and updating the status of that expense based on changes in its payments.

Specific objective C4

To develop a system with the ability to integrate and automatically receive bank information from a data source, process it and update the status of payments and expenses based on this information.

1.3. Methodology

In order to implement this project, a work plan has been designed beginning with the analysis of the current state of the type of systems that, in some way, cover the same objectives defined in the previous section, such as personal financial aggregators in the form of applications for mobile devices. Another analysis is also carried out of the state of financial aggregators that offer their services to third parties, due to their relevance as a source of banking information for the system to be developed in this work, and of two other options as information sources.

Once the state of the art has been presented, a process of analysis and capturing of requirements is performed in order to have a basis on which to define the system and its functionality. This analysis is carried out with the participation of real people through surveys about their financial habits and personal expenses. From this process, a series of functional requirements are extracted. These are formalized later and then they are selected and limited, which leads to the generation of the scope of the project.

After defining the functionality and scope, the system to be built needs to be completely specified, including its name, software architecture, data model and logic for processing transactions in relation to expenses.

The definition of the system allows us to start with its technical implementation. To do this, first, we carry out a specification of the languages and technologies to be used, and then different detailed tasks are defined based on the functionality intended to be achieved.

Following the defined architecture and tasks, and using the specified languages and technologies, a software system is developed which will be proposed as a solution to the needs and objectives defined above.

1.4. Structure of the report

The structure of this document follows the methodology and work plan defined in the previous section. Six chapters reflect the work followed throughout the development of this project.

An attempt has been made to follow a progression on the level of technicality of the report, starting with more theoretical and analytical chapters, and finally reaching implementation and technology details closer to the actual code, which is attached to this document through different online repositories indicated in each corresponding chapter.

Starting on chapter 2, we present an analysis of the current state of existing banking aggregators, both personal –which are similar to the system object of this work– and providers of information for third parties –which are related to the objectives of this

project in a technical way-. Other alternative information sources are also analyzed, such as email notifications and manual supply of bank information through spreadsheets.

Chapter 3 specifies the process of analysis and capture of requirements, beginning with the definition of the methodology to follow, then passing through the selection of real people and the realization of the surveys, and ending with the processing of the results of the surveys and the extraction of functionality.

Chapter 4, in which the system is already defined in a theoretical way, covers the specification of the same as a software system. This part of the report details some technical decisions related to a general overview of the system, such as the architecture, the data model and the processing algorithms between transactions and expenses.

Chapter 5 details some technical decisions of a lower level, such as the specific language chosen, the frameworks and libraries on which to build the frontend and backend services, and a relatively concrete description, without reaching the actual code, of the implementation of each user story or use case in each part of the system. This chapter ends with a statement of the results obtained and the functionality that the system is capable of covering.

That statement of results is used in the next and last chapter, the sixth one, to propose a comparison between it and the objectives set at the beginning of the work, and thus generate a set of conclusions. To close this chapter and the work, a final analysis has been made based on the level of achievement of the objectives, extracting different lines for future work.

At the end of this report, after the bibliography, various appendices such as forms or tables are added. These appendices are listed in an index in the beginning of the document, and referenced throughout the text.

Capítulo 2

Estado de la cuestión

2.1. Estado de la cuestión

La digitalización ha propugnado que el modelo de servicio del sector financiero evolucione radicalmente en las últimas dos décadas. Esta evolución se caracteriza por una mayor conectividad, una mayor automatización y un procesamiento más rápido de la información en la interacción con el cliente y en los procesos administrativos.

Recientemente, la digitalización ha pasado de centrarse en la mejora de la ejecución de tareas tradicionales a introducir modelos de negocio y oportunidades fundamentalmente centradas en facilitar a las empresas la prestación de servicios financieros agregados sobre los productos objeto de comercialización o *commodities* que crean los bancos. Las finanzas digitales se refieren a una amplia gama de proveedores de servicios financieros innovadores y empresas de tecnología financiera, conocidas como *fintech*, que ofrecen productos financieros de vanguardia, negocios, software y métodos novedosos de comunicación e interacción con el cliente. En vista de ello, la investigación sobre finanzas y sistemas de información ha comenzado a analizar estos cambios y cómo los avances digitales han afectado al sector financiero. Por ello, en esta sección se revisa el estado actual de la investigación en finanzas digitales.

2.1.1. El ecosistema de las *fintechs*

El término *fintech* (a veces *Fin-tech* o *FinTech*, del inglés *financial technology*) es un nuevo término que hace referencia a las modernas relaciones entre el usuario y la banca y, en particular, a las tecnologías relacionadas con Internet (por ejemplo, computación en la nube y dispositivos móviles) y las actividades empresariales en el sector de los servicios financieros (por ejemplo, préstamo de dinero y transacciones bancarias) (Suryono, Budi y Purwandari, 2020).

Debido a la nueva competencia de las empresas *fintech*, las finanzas digitales han comenzado a suponer un reto para los proveedores de servicios financieros asentados, como los bancos y las compañías de seguros. Sin embargo, también ofrece a los operadores tradicionales nuevas oportunidades para llegar a una clientela más joven y con mayores conocimientos tecnológicos.

En vista de dicha situación, los intermediarios financieros tradicionales están debatiendo continuamente cómo hacer frente a las *fintech* y si las adquisiciones, las estrategias cooperativas o competitivas, o la participación de esas empresas como proveedores de servicios se adaptan mejor a sus modelos de negocio. De 4 050 millones de dólares en 2013 a 12 210 millones en 2014, un aumento masivo de las inversiones mundiales en empresas *fintech* (Skan, Dickerson y Masood, 2015) refleja este patrón. Para seguir siendo competitivos y ofrecer a los clientes servicios novedosos y atractivos, los operadores tradicionales también aprovechan las nuevas oportunidades que brinda la tecnología y desarrollan conceptos novedosos en el ámbito de las finanzas digitales.

Antecedentes

La popularización de los agregadores financieros digitales de uso personal se remonta a la década de 2010, como respuesta a la creciente complejidad de las finanzas personales en la era digital. La proliferación de cuentas bancarias, tarjetas de crédito y otros productos financieros hizo que el seguimiento manual de las transacciones se volviera desafiante.

Además de Money Manager y la española Fintonic, destacan otros ejemplos como Mint en Estados Unidos, YNAB (You Need A Budget), que se enfoca en la planificación presupuestaria, y Personal Capital, que ofrece herramientas tanto para la gestión de presupuestos como para inversiones.

Aunque en esta parte se está realizando un análisis de los agregadores de uso personal por su similitud con el sistema objeto del proyecto, además de estos, existen otros de tipo servicio que ofrecen sus herramientas y la información que agregan a terceros, como otras aplicaciones o empresas. Más adelante, en la Sección 2.2, se realiza un análisis concreto sobre este tipo de agregadores, que son de mayor relevancia para el cumplimiento de los objetivos de este proyecto.

Características clave

Una serie de características comunes destacan en los agregadores financieros y las aplicaciones *fintech*:

- **Agregación de cuentas.** Estas aplicaciones permiten a los usuarios vincular cuentas bancarias, tarjetas de crédito, cuentas de inversión y otros productos financieros en un solo lugar. Ejemplos como Mint y Personal Capital incluso permiten añadir cuentas de jubilación y propiedades inmobiliarias.
- **Categorización automática.** Los agregadores utilizan algoritmos avanzados de aprendizaje automático para clasificar automáticamente las transacciones en categorías como alimentación, transporte, entretenimiento y más. YNAB se destaca por su enfoque en la categorización manual para fomentar la conciencia financiera. Por otro lado, Plaid destaca por haber colaborado con distintas compañías de Inteligencia Artificial para refinar sus algoritmos.
- **Análisis de tendencias.** Las herramientas de análisis permiten a los usuarios identificar tendencias en sus hábitos de gasto y ahorro. Personal Capital, por ejemplo, ofrece un análisis exhaustivo de inversiones y planificación de jubilación.
- **Alertas y recordatorios.** Estas aplicaciones proporcionan alertas y recordatorios para ayudar a los usuarios a realizar pagos a tiempo, evitar cargos por sobregiro y mantenerse al tanto de sus objetivos financieros. YNAB se destaca por su enfoque en la gestión activa de presupuestos y metas.
- **Seguridad y privacidad.** La seguridad y privacidad de los datos financieros son prioridades para todos los agregadores. Utilizan cifrado avanzado y autenticación de dos factores para proteger la información personal y financiera.

Evolución

Desde su inicio, debido a su naturaleza y a la necesidad que cubren, los agregadores financieros digitales han evolucionado de manera rápida y significativa. Han ampliado su alcance para incluir funciones de planificación financiera integral, como gestión de inversiones y planificación de jubilación. Además, han desarrollado aplicaciones móviles para brindar una experiencia más accesible y conveniente a los usuarios.

Dependiendo del territorio, los agregadores han podido avanzar de manera más o

menos rápida y normalizada según la disponibilidad de la información por parte de las entidades bancarias y financieras. En el caso de la Unión Europea, distintas regulaciones de distintos ámbitos facilitan una estandarización del acceso a este tipo de información, como se analiza más adelante en la Sección 2.2.

Impacto

Estas aplicaciones han tenido un impacto significativo en el área financiera de la vida de las personas. Han empoderado a los usuarios al proporcionarles un mayor control sobre sus finanzas y una comprensión más profunda de sus ingresos, gastos y hábitos asociados. También han promovido la educación financiera al hacer que la gestión financiera personal sea más accesible y comprensible para una audiencia más amplia.

En resumen, los agregadores financieros digitales representan una evolución importante en la gestión financiera personal en la era digital. Su capacidad para simplificar y mejorar la administración de las finanzas personales ha tenido un impacto positivo en la vida de los usuarios y seguirá desempeñando un papel crucial en la mejora de la salud financiera de las personas en el futuro.

2.2. Regulación y sistemas de información bancaria

Tras el planteamiento del estado de la cuestión sobre los agregadores bancarios de disponibilidad general o uso personal, es pertinente, para este proyecto, realizar un análisis acerca de otro ámbito: los agregadores bancarios proveedores de servicios de información bancaria.

Estos agregadores actúan como intermediarios entre las entidades bancarias y las aplicaciones de *fintech*, poniendo a disponibilidad de esta información sobre transacciones asociadas a un usuario. Estos agregadores son interesantes para este proyecto dada la necesidad de disponer de información bancaria asociable a gastos personales, como se detalla en las siguientes secciones.

En la actualidad, casi cualquier organización que genere o procese datos los exponga a otros entes a través de mecanismos programáticos, es decir, acoplables a sistemas informáticos. El caso de la información bancaria no es una excepción.

En la siguiente subsección se plantea una evaluación sobre las posibles fuentes de datos, la normativa vigente en esta materia y las tecnologías disponibles. Este análisis se centra en las interfaces de programación de aplicaciones (*API*, por sus siglas en inglés) y en el Espacio Económico Europeo (EEE) por el alcance limitado del proyecto y por la estandarización y la normativa específica desarrollada por la Unión Europea para este ámbito.

Posteriormente se plantean otras dos posibles fuentes de actualizaciones sobre transacciones bancarias como son las notificaciones de los bancos o de los proveedores de servicios a través de correo electrónico, o la aportación manual de las transacciones a través de un fichero de datos. La capacidad del sistema objeto de este proyecto para procesar distintas fuentes de información puede ser de gran utilidad para los usuarios que no puedan o no quieran ceder el acceso a su información bancaria o a sus mensajes de correo electrónico.

Aunque el análisis se centra en la utilización de la información bancaria para actualizar gastos previamente registrados por el usuario en el sistema, esta información podría usarse también para detectar automáticamente dichos gastos. Una posible funcionalidad de este tipo, en base a transacciones pasadas o correos electrónicos, podría ser de interés para el público objetivo de este proyecto. Por ejemplo, se podría añadir, para mejor experiencia de usuario, un asistente que realice una *entrevista* e identifique todos o la mayoría de los gastos previsibles o realizados.

2.2.1. Sistemas de información bancaria, *open banking* y regulación europea

La *Revised Payment Services Directive* (PSD2) (Parlamento Europeo y Consejo de la Unión Europea, 2015), transpuesta en España en el año 2018 (Congreso de los Diputados, 2018), es la directiva europea que regula los servicios de pago –transferencias, domiciliaciones, pagos con tarjetas– procesados dentro del Espacio Económico Europeo. Esta directiva obliga a los bancos a habilitar mecanismos para que terceros puedan intervenir de manera segura y transparente en los servicios y datos bancarios, en nombre de los clientes y siempre con su consentimiento.

La normativa europea y su transposición refuerzan, además, las medidas de seguridad para realizar las transacciones, incluyendo medidas de doble autenticación o autenticación reforzada (teléfono móvil, huella dactilar, etc.).

La participación de terceros en las operaciones entre entidades bancarias y clientes, es lo que se conoce como *open banking*.

Antes de proceder en este análisis, es importante mencionar los siguientes conceptos del entorno de la PSD2 y open banking (Congreso de los Diputados, 2018, artículo 4):

- Usuario de servicios de pago (*Payment Service User*, PSU): persona física o jurídica que utiliza un servicio de pago, ya sea como ordenante, beneficiario o ambos.
- Proveedor de servicios de pago gestor de cuenta (*Account Servicing Payment Service Provider*, ASPSP): entidad financiera en la que el usuario de servicios de pago tiene cuentas cuya información se puede utilizar o a partir de las cuales se pueden iniciar pagos.
- Servicios de información de cuentas (*Account Information Services*, AIS): servicios que permiten, con el consentimiento previo del usuario, obtener información de este y de sus cuentas.
- Proveedor de servicios de información de cuentas (*Account Information Services Provider*, AISP): proveedor de AIS. También se conocen como agregadores de cuentas.
- Servicios de iniciación del pago (*Payment Initiation Services*, PIS): servicios que permiten, con el consentimiento previo del usuario, iniciar pagos en nombre de este, sin necesidad de detalles de inicio de sesión o de una tarjeta bancaria.
- Proveedor de servicios de iniciación del pago (*Payment Initiation Services Provider*, PISP): proveedor de PIS. También se conocen como iniciadores de pagos.
- Autoridad nacional competente (*National Competent Authority*, NCA): organismo regulador y supervisor de entidades financieras que autoriza a estas y a terceros para operar en el marco de PSD2. En España es el Banco de España.
- Tercero proveedor (*Third-Party Provider*, TPP): prestador financiero AISP o PISP que actúa en nombre de un usuario de servicios de pago accediendo a su información bancaria o iniciando un pago. Necesita autorización por parte de la NCA para poder operar.
- Pagos recurrentes variables (*Variable Recurring Payments*, VRP): permiten a los PISP autorizar transferencias de pagos recurrentes que pueden diferir en cantidad cada mes.

Cualquier sistema o aplicación que desee integrarse y utilizar información bancaria de un usuario de servicios de pago, debe hacerlo a través de un TPP. Los propios bancos pueden ser TPP de ellos mismos o de otros bancos. Por ejemplo, entidades como BBVA y Openbank permiten, en su propia aplicación, añadir cuentas de otras entidades para operar con las mismas y visualizarlas.

Actualmente existen distintos TPP autorizados dentro del Espacio Económico Europeo. Un ejemplo en España, bastante conocido y autorizado como AISP y PISP, según

se puede comprobar en el registro de entidades del Banco de España, es Fintonic. Fintonic fue la primera entidad en conseguir las licencias para AIS y PIS en España, en octubre de 2019, y ofrece diferentes productos financieros como préstamos, seguros y gestión de movimientos.

Sin embargo, Fintonic actúa como plataforma final y no expone ninguna interfaz para que otros sistemas o aplicaciones puedan integrarse y hacer uso de los datos bancarios que recibe y procesa.

Aunque la PSD2 no obliga específicamente a que las entidades bancarias o proveedores utilicen interfaces de programación de aplicaciones (API) para cumplir con la misma, utilizarlas se ha convertido en una práctica común desde la implantación de la directiva.

Existen numerosos TPP que ponen distintas API a disposición de otros sistemas para hacer aún más amplia la oferta de servicios financieros, manteniendo siempre la seguridad y la confianza. Dado que estos terceros, intermediarios, suelen ser compatibles con numerosas entidades bancarias, en general y en esta memoria se les da el nombre de agregadores bancarios.

La existencia de los agregadores bancarios facilita en gran medida la integración con las entidades bancarias. Esto se debe a que la normativa no regula modelos de datos ni de operaciones, por lo que cada banco expone su API distinta a las del resto y, posteriormente, los agregadores unifican toda la información y los métodos y los ponen a disposición de los sistemas integradores a través de una API única.

Para descubrir agregadores bancarios, además de buscar en internet y utilizar los registros de las autoridades financieras, existen herramientas como openbankingtracker (openbankingtracker.com) que ofrecen información de interés a cerca de entidades del entorno de *open banking* y PSD2. openbankingtracker facilita, concretamente, una lista de proveedores autorizados como AISP y PISP y que además son agregadores de API de PSD2 y *open banking*.

Con el propósito de elegir un agregador bancario para este proyecto, se ha realizado un análisis de distintas opciones, cuyo detalle se encuentra en el Apéndice A.

La opción elegida es Plaid, un agregador financiero con sede en Estados Unidos que tiene autorización para operar en el EEE y que es compatible con la mayoría de instituciones bancarias. Es la mejor opción por su extensa documentación, la conveniencia para la realización de pruebas y la capacidad de notificar actualizaciones de manera asíncrona. Más adelante, en la Sección 5.3, se detalla de una manera técnica la integración con este agregador.

2.2.2. Notificaciones bancarias y de proveedores de servicios

Como alternativa a la entrada de información y actualizaciones desde un agregador bancario, las notificaciones que los usuarios reciben a través del correo electrónico por parte de los proveedores de servicios también pueden ofrecer datos útiles.

Algunos ejemplos de notificaciones útiles son los que se detallan a continuación.

- Mensaje de una entidad bancaria: “Te informamos que el recibo de *ABCDEF SL*, se cargará en tu cuenta *8739* el día *2023-05-17* por importe de *27.92 EUR*”

- Mensaje de una comercializadora de gas: “Adjunta te enviamos tu factura de gas de ABCDEF SL. También puedes consultarla en Mi Área Cliente. Importe: 27,92 €”
- Mensaje de proveedor de servicios de internet: “Hola ABC DEF, Ya está disponible tu factura de mayo de 2023. Importe 37,00 €”
- Mensaje de suscripción de streaming: “Tu suscripción de ABCDEF SL continúa y se te ha cobrado el importe correspondiente. Artículo: Suscripción Premium. 1,99 €”

En todos estos casos, se pueden utilizar los siguientes datos y metadatos para asociar e interpretar los mensajes como actualizaciones sobre gastos previstos registrados en el sistema:

- La estructura con formato fijo y conocido de cada mensaje, es decir, el mismo texto con los mismos campos cada vez que se recibe el mensaje.
- La fuente o el remitente de los mensajes.
- El asunto del mensaje.
- El nombre de la entidad dentro del mensaje.
- Algún concepto dentro del mensaje.

Y, gracias al análisis o *parsing* del asunto y el contenido del mensaje, se puede concretar dicha actualización extrayendo, entre otros:

- Tipo de actualización (cobro previsto, cobro producido, etc.).
- Importe.
- Fecha del posible cobro.

El principal inconveniente de los mensajes de correo electrónico como fuente de información es la diversidad de formatos y datos entre las diferentes entidades bancarias y proveedores de servicios. Sería necesario disponer de una amplia base de ejemplos y reglas para construir los algoritmos necesarios.

2.2.3. Suministro de la información en lote de forma manual a través de ficheros de datos

Como tercera opción para la obtención de información sobre gastos, y como alternativa más atractiva para la sensación de control sobre los datos sensibles a los que se da acceso, existiría la posibilidad de que el propio usuario proporcionara, de manera manual, dichos datos.

La mayoría de entidades bancarias ofrecen actualmente, a través de sus plataformas digitales (páginas web o aplicaciones móviles), la posibilidad de descargar ficheros de hoja de cálculo con listados de las últimas transacciones asociadas a las cuentas del usuario.

Estos ficheros son procesables de manera programática y podrían integrarse como fuente de información del sistema.

Sin embargo, existen dos principales inconvenientes del uso de estos ficheros como fuente de información. El principal inconveniente es la baja calidad y cantidad de información provista para cada gasto, compuesta normalmente por el nombre de la entidad y algunos datos de baja relevancia. No se incluye, a diferencia de los agregadores bancarios, una clasificación de gastos personales detallada. Tampoco es posible, como sí se podría a través de un agregador o de notificaciones, hacer un seguimiento de cada transacción o mensaje, pues no se provee un identificador único.

En segundo lugar, y de manera similar al problema de las notificaciones, utilizar estos ficheros como fuente de información añadiría un alto nivel de complejidad al sistema, por no estar normalizada la manera en que cada entidad bancaria exporta y facilita las hojas de cálculo, incluyendo los datos y los formatos utilizados.

Si se quisiera proceder con estos tipos de fuentes de información, sería necesaria, además, la intervención del usuario a la hora de asociar un movimiento o un mensaje con un gasto registrado. Es decir, en lugar de asociar un gasto directamente y aplicar cambios sobre este, el sistema tendría que notificar al usuario y confirmar las acciones a realizar o incluso el gasto sobre el que realizarlas.

Capítulo 3

Análisis funcional

3.1. Introducción

En este capítulo se documenta cómo se ha llevado a cabo la fase de análisis junto con la captación de requisitos para la definición del sistema.

La metodología seguida para este proceso es similar a la presentada como parte de la asignatura de Desarrollo de Sistemas Interactivos del Grado en Ingeniería Informática de esta Universidad. Esta metodología se basa en la colaboración con personas reales para conocer de primera mano los detalles de la necesidad detectada y planteada en la introducción de este trabajo (1) y formalizar, a partir de estos, una serie de requisitos que darán lugar a un alcance y a diferentes tareas técnicas.

Durante la siguiente sección, de análisis de requisitos, se plantean las diferentes hipótesis de personas creadas para este proyecto, se seleccionan diferentes usuarios para llevar a cabo una serie de encuestas y, tras completarlas, se muestran los resultados y una extracción factual procesada de estas.

Al final del capítulo se produce una transformación desde lo extraído de las encuestas hasta un listado de requisitos funcionales en un formato más granular y limitable a modo de alcance. Para terminar, tras definir el alcance, se detallan a nivel técnico las funcionalidades incluidas y se detallan las historias de usuario y casos de uso.

3.2. Análisis de requisitos

Para realizar un análisis de requisitos y comprender de una manera más precisa las necesidades del proyecto, se ha planteado una serie de encuestas a realizar a personas con un mínimo de interés en un sistema como el que se está diseñando.

3.2.1. Hipótesis de personas

Con el fin de abarcar a distintos tipos de personas en cuanto a su uso de sistemas, herramientas o métodos de planificación de gastos, se ha creado una hipótesis de personas, que serán de utilidad a la hora de diseñar las encuestas. Las personas tipo ideadas son las siguientes:

- Persona tipo A: persona que ya utiliza alguna aplicación de gestión de gastos
- Persona tipo B: persona que utiliza otro sistema de gestión de gastos, ya sea manual o digital
- Persona tipo C: persona que no utiliza ningún sistema de gestión de gastos, pero que tiene intención de empezar a utilizar alguno

3.2.2. Selección de usuarios para las encuestas

En base a la hipótesis de personas, se seleccionó un total de siete usuarios reales a los que realizar las encuestas. Estos usuarios se validaron mediante una prueba previa o *screening* con algunas preguntas breves sobre su uso de herramientas o técnicas de planificación de gastos.

Los usuarios encuestados, anonimizados por privacidad, son los siguientes:

1. Persona tipo A
 - a) N. Hervás
2. Persona tipo B
 - a) A. Leiva
 - b) J. Ferrer
 - c) H. Marco
 - d) P. Bonet
3. Persona tipo C
 - a) Y. Díaz
 - b) A. Zhou

3.2.3. Diseño de las encuestas

El diseño de las encuestas se realizó con orientación a conocer herramientas y sistemas de los cuales estuviera haciendo uso en ese momento. La intención es conocer, de

estos sistemas, su tipo, funcionalidades y aspectos de interés para este proyecto.

Parte 0: introducción previa

De manera previa e informal, se identifica a los usuarios a encuestar, se les presenta la encuesta, objetivos y formato, y se les solicita autorización para publicar los resultados de manera anonimizada.

Tanto esta parte previa como las partes 1 y 3 son comunes a todas las personas tipo de la hipótesis. La parte 2 varía en función de cada tipo.

Parte 1: introducción personal y filtro de idoneidad para la encuesta (*screening*)

Esta primera parte de la encuesta, comenzada ya de manera formal, tiene como objetivo validar la clasificación de la persona encuestada en la persona tipo ideada como hipótesis.

Las preguntas planteadas son las siguientes:

1. En general, actualmente, ¿a qué tipo de gastos tienes que hacer frente?
 - a) Aclaración: en este apartado me gustaría que me contaras en qué situación te encuentras y a qué gastos tienes que hacer frente en tu día a día, semana a semana o mes a mes. Debe ser a modo de resumen, aunque puedes explayarte todo lo que quieras. No es necesario que detalles aquí tus gastos, eso viene más adelante. Por ejemplo: *Actualmente vivo en un piso compartido con estudiantes, por lo que mensualmente cubro los gastos de mi casa. Además, tengo varias suscripciones de entretenimiento y hago ejercicio en un gimnasio.*
2. Mes a mes, ¿realizas gastos¹ relacionados con tu hogar², suscripciones³, financiación, seguros u otros cargos recurrentes?
 - ¹Ya sean domiciliados o manuales.
 - ²Hogar: alquiler, agua, luz, gas, telefonía, internet, etc.
 - ³Suscripciones: servicios de *streaming* de vídeo o música, servicios de comida a domicilio, etc.
3. ¿Utilizas alguna aplicación (para móviles u ordenadores) para la gestión de tus gastos personales?
4. Si no es así, ¿utilizas algún otro sistema (manual o digital) para el mismo fin?
5. Si no utilizas sistema alguno para gestionar tus gastos, ¿crees que podrías beneficiarte de utilizar alguno?

Partes 2.A, 2.B y 2C: preguntas para cada persona tipo sobre los sistemas que utilizan

En estas partes se pregunta a la persona encuestada sobre su planificación de gastos según las herramientas o técnicas que utiliza.

El propósito de estas partes de la encuesta es conocer qué funcionalidades utilizan y aprecian los usuarios de los sistemas que ya utilizan, y cuáles echan en falta, con la

finalidad de generar y cubrir casos de uso diversos en el apartado de diseño del proyecto.

Parte 2.A: preguntas para la persona tipo A sobre el sistema que utiliza

Las preguntas planteadas son las siguientes:

1. ¿Qué aplicación de gestión de gastos utilizas actualmente?
2. ¿Con qué frecuencia accedes a la aplicación?
3. ¿La aplicación te permite obtener una previsión de lo que vas a gastar en el mes siguiente? ¿Es un aspecto interesante para ti?
4. ¿La aplicación te permite manejar tus gastos recurrentes (luz, agua, suscripciones, etc.)? ¿Es un aspecto interesante para ti?
5. ¿La aplicación te calcula una cantidad a reservar para los gastos recurrentes del mes siguiente? ¿Es un aspecto interesante para ti?
6. ¿La aplicación está conectada con tu banco y puedes ver tus movimientos de manera dinámica? ¿Es un aspecto interesante para ti?
7. ¿La aplicación te indica cuando se ha producido un cobro? ¿Es un aspecto interesante para ti?
8. ¿Recibes notificaciones de la aplicación? ¿Son interesantes para ti?
9. ¿La aplicación te ofrece resúmenes de gasto? ¿Son interesantes para ti?
10. ¿La aplicación te permite establecer objetivos de ahorro? ¿Es un aspecto interesante para ti?
11. ¿La aplicación te permite categorizar los gastos o los categoriza automáticamente? ¿Es un aspecto interesante para ti?
12. De las funcionalidades de la aplicación mencionadas, ¿cuáles valoras o utilizas más?
13. ¿Qué otras funcionalidades no mencionadas ofrece la aplicación y utilizas habitualmente?
14. ¿Has probado otras aplicaciones? ¿Qué aspecto te hizo decantarte por la que usas actualmente?

Parte 2.B: preguntas para la persona tipo B sobre el sistema que utiliza

Las preguntas planteadas son las siguientes:

1. ¿Qué sistema de gestión de gastos utilizas actualmente? (Hoja de cálculo) ¿Lo has creado personalmente o has utilizado una plantilla?
2. ¿Con qué frecuencia utilizas o accedes a la hoja de cálculo?
3. ¿La hoja de cálculo te permite obtener una previsión de lo que vas a gastar en el mes siguiente? ¿Es un aspecto interesante para ti?
4. ¿La hoja de cálculo te permite manejar tus gastos recurrentes (luz, agua, suscripciones, etc.)? ¿Es un aspecto interesante para ti?

5. ¿La aplicación te calcula una cantidad a reservar para los gastos recurrentes del mes siguiente? ¿Es un aspecto interesante para ti?
6. ¿La hoja de cálculo te permite registrar cuando se ha producido un cobro y por cuánto ha sido? ¿Es un aspecto interesante para ti?
7. ¿La hoja de cálculo te ofrece resúmenes de gasto? ¿Son interesantes para ti?
8. ¿La hoja de cálculo te permite establecer objetivos de ahorro? ¿Es un aspecto interesante para ti?
9. ¿La hoja de cálculo te permite categorizar los gastos? ¿Es un aspecto interesante para ti?
10. De las funcionalidades de la hoja de cálculo mencionadas, ¿cuáles valoras o utilizas más?
11. ¿Qué otras funcionalidades no mencionadas ofrece la hoja de cálculo y utilizas habitualmente?
12. ¿Has probado otros sistemas como, por ejemplo, aplicaciones? ¿Qué aspecto te hizo decantarte por la hoja de cálculo?
13. ¿Te gustaría que el sistema que utilices pudiera conectarse a tu banco y añadir movimientos de manera dinámica?
14. ¿Te gustaría que el sistema que utilices pudiera enviarte notificaciones para estar al tanto de tus gastos? Si puedes, detalla cómo.

Parte 2.C: preguntas para la persona tipo C sobre el sistema que utiliza

Las preguntas planteadas son las siguientes:

1. ¿Utilizas la aplicación de tu banco para gestionar tus gastos recurrentes? ¿Qué sueles observar?
2. ¿Con qué frecuencia utilizas o accedes a la aplicación de tu banco?
3. ¿Llevas a cabo una previsión de lo que vas a gastar en el mes siguiente? ¿Te gustaría que una aplicación lo hiciera por ti?
4. ¿Reservas alguna cantidad para los gastos recurrentes del mes siguiente? ¿Cómo haces el cálculo?
5. ¿Te gustaría que una aplicación te avisara de una manera intuitiva sobre qué gastos recurrentes te cobran?
6. ¿Haces resúmenes, balances o análisis de tus gastos? ¿Cómo los llevas a cabo?
7. ¿Estableces objetivos de ahorro? ¿En qué te basas?
8. En tus gestiones sobre gastos recurrentes, ¿qué parte te parece más tediosa?
9. ¿Has probado aplicaciones u hojas de cálculo para llevar tus gastos personales? ¿Qué aspecto te hizo decantarte por no usarlas?
10. ¿Qué otros aspectos buscarías en una aplicación para gestionar tus gastos personales?

11. ¿Te gustaría que la aplicación pudiera conectarse a tu banco y añadir movimientos de manera dinámica?
12. ¿Te gustaría que la aplicación pudiera enviarte notificaciones para estar al tanto de tus gastos? Si puedes, detalla cómo.

Parte 3: evaluación de gastos

En la tercera parte de la encuesta, se solicita a la persona encuestada que detalle los gastos personales recurrentes a los que hace frente en periodos mensuales.

Esta parte se presenta a la persona encuestada en forma de tabla para responder con mayor facilidad a cada pregunta sobre cada gasto identificado. Además, se proporcionan ejemplos de gastos comunes, junto con las entidades que suelen proveer los servicios asociados.

Los resultados de esta parte permitirán conocer los diferentes casos de gastos a predecir, con diferentes tipos, duraciones y entidades asociadas. Además de en los casos de uso, esta información será de gran utilidad a la hora de definir el modelo de datos del sistema.

Las preguntas planteadas son las siguientes:

1. ¿A qué gastos recurrentes¹ tienes que hacer frente mes a mes? ¹Recurrentes: que se repitan en el tiempo de una forma más o menos predecible y homogénea. Ejemplos de gastos y entidades:
 - a) Hogar:
 - 1) Alquiler: particular, Alquiler Seguro, etc.
 - 2) Agua: Ista, Aguas de Murcia, etc.
 - 3) Luz (energía eléctrica): Endesa, Iberdrola, etc.
 - 4) Gas: Iberdrola, Curenergía, Repsol, etc.
 - 5) Telefonía: Movistar, Vodafone, etc.
 - 6) Internet: Movistar, Vodafone, etc.
 - 7) Comunidad.
 - 8) Impuesto sobre bienes inmuebles.
 - b) Suscripciones:
 - 1) Streaming de vídeo: Netflix, Disney+, Amazon Prime Video, HBO Max, YouTube Premium, Apple TV+, etc.
 - 2) Streaming de música: Spotify, Apple Music, Amazon Music, YouTube Music, etc.
 - 3) Compras y envíos: Amazon Prime (*anual), etc.
 - 4) Comida a domicilio: Uber Eats Pass, etc.
 - 5) Prensa y noticias: El País, etc.

- 6) Almacenamiento en la nube: Google Drive, Dropbox, Mega, etc.
 - 7) Transporte: Uber Pass, etc.
 - 8) Otras suscripciones: Twitter Blue, Google One, iCloud+, Apple One, hosting web, dominios web, revistas, plataformas de gaming y videojuegos, etc.
- c) Financiación:
- 1) Préstamos bancarios
 - 2) Tarjetas de crédito
 - 3) Compra financiada en Apple: Cetelem, etc.
 - 4) Compra financiada en IKEA (IKEA Visa): CaixaBank Payments & Consumer, etc.
- d) Seguros:
- 1) Seguro de móvil: Apple Care, Apple Care+, etc.
 - 2) Seguro de ordenador: Apple Care, Apple Care+, Rosellimac, El Corte Inglés, Fnac, etc.
 - 3) Seguro de salud
- e) Salud y bienestar:
- 1) Gimnasio: McFit, etc.
 - 2) Medicamentos
 - 3) Psicólogo
 - 4) Fisioterapeuta
- f) Otros gastos recurrentes:
- 1) Abono de transportes
 - 2) Compras personales recurrentes
2. Si no lo has indicado ya, ¿qué empresas o entidades te ofrecen dichos servicios?
 3. ¿Cuáles de esos gastos tienen un importe fijo y cuáles dependen del periodo o del consumo?
 4. ¿Cuáles de esos gastos tienes domiciliados en una cuenta bancaria o asociados a una tarjeta de crédito o débito?
 5. ¿Con qué frecuencia pagas cada gasto? (Mensual, bimensual, trimestral, etc.)
 6. ¿Qué día realizas cada pago o te pasan cada recibo (si lo sabes)?
 7. Si tienes algún gasto domiciliado, ¿tu banco te notifica cuando se produce el cobro? ¿Por qué medio lo hace (app, email...)?
 - a) Ejemplo: email de Banco Sabadell que indica ".^{el} recibo de Endesa se cargará en tu cuenta 3637 el día 2023-06-07 por importe de 20.00 EUR".
 8. ¿Qué entidades te notifican cuando se va a producir un cobro? ¿Por qué medio lo hacen (app, email...)?

- a) Ejemplo: email de Endesa con asunto "Ya puedes descargar tu factura.", que contiene el importe de la factura.
 - b) Ejemplo: email de Apple con asunto "Your invoice from Apple.", que contiene el importe de la factura.
9. En caso de tener varios gastos domiciliados, ¿dispones de una única cuenta bancaria o tienes varias para distintos propósitos?

Parte 4: agradecimiento final

Por último, se agradece a la persona encuestada su participación y se le plantea una última pregunta:

- ¿Te gustaría conocer el resultado final del proyecto?

3.2.4. Realización de las encuestas

Por el tipo de preguntas generadas, especialmente las de la parte 3 sobre los gastos detallados, se determinó que la mejor forma de realizar la encuesta sería a través de una hoja de cálculo de Excel.

Se generó un fichero Excel para cada persona tipo. Estos ficheros están disponibles, de manera agrupada, en el Apéndice B.

Los ficheros se facilitaron a las personas a encuestar a través de un enlace de Google Drive. Se les dio una semana de margen para rellenarlos. En general, cada persona tardó menos de una hora en completar la encuesta.

3.2.5. Resultados de las encuestas y extracción factual

Tras realizar las encuestas y recoger los ficheros Excel, se llevó a cabo una normalización y agrupación de los mismos en un fichero único tabulado, disponible en el Apéndice C.

A continuación se presenta un análisis de los resultados y una serie de hechos extraídos de los mismos, que serán de utilidad para la generación de requisitos funcionales y casos de uso.

Parte 1: introducción personal y filtro de idoneidad para la encuesta (*screener*)

Todas las personas encuestadas pasan el filtro previo o *screener* al indicar que realizan gastos recurrentes. Como se plantea en la hipótesis, algunas personas utilizan herramientas de planificación de gastos y otras no; y dentro de las que sí utilizan, hay personas que hacen uso de aplicaciones y otras que prefieren usar sistemas *manuales*.

Analizando la pregunta "En general, actualmente, ¿a qué tipo de gastos tienes que hacer frente?", podemos extraer la siguiente perspectiva general en forma de hechos:

- Todas las personas encuestadas viven en casas o pisos, solas o acompañadas, y esto les genera gastos recurrentes. Estos gastos se agrupan principalmente en alquiler y

suministros (agua, gas, electricidad, internet, etc.) y suelen ser fijos.

- Algunas personas hacen frente a gastos de compras regulares (alimentación, fungibles y demás), que suelen ser variables.
- Algunas personas hacen frente a gastos relacionados con suscripciones a servicios de entretenimiento (plataformas de *streaming* de vídeo y música).
- Algunas personas hacen frente a gastos de ocio, restauración, transportes, ropa, vehículo propio, temas laborales, mascotas, decoración, salud, inversiones y financiación.

Parte 2.A: preguntas para la persona tipo A sobre el sistema que utiliza

La persona encuestada para el tipo A, N. Hervás, responde a las preguntas de esta parte en base a su uso de la aplicación Money Manager para iPhone.

Hervás accede a la aplicación cada vez que tiene que introducir un gasto.

De las respuestas se extraen las siguientes funcionalidades de interés (entiéndase que los hechos comienzan con “el sistema permite”):

- Introducir gastos de forma manual.
- Asignar categorías y subcategorías a los gastos. El usuario entiende que esta es una de las mejores funcionalidades del sistema.
- Configurar presupuestos por categorías.
- Marcar determinados gastos como recurrentes.
- Contabilizar el dinero en efectivo.
- Obtener un resumen global de dinero asociando cuentas bancarias.
- Visualizar resúmenes de gasto de manera gráfica a través de un gráfico circular o *de tarta* y por categorías.
- Visualizar los gastos a través de una vista de calendario diario o mensual.
- Visualizar estadísticas de ingresos y gastos con gráficos por categorías para los periodos registrados.
- Importar y exportar el histórico de gastos.
- Adjuntar imágenes a los gastos.
- Añadir comentarios a los gastos.
- Utilizar una moneda secundaria para viajes al extranjero.

Por otro lado, se extrae que el sistema carece de las siguientes funcionalidades, de las cuales algunas también podrían ser de interés (entiéndase que los hechos comienzan con “el sistema no permite”):

- Obtener los gastos y sus detalles de manera dinámica mediante conexión a entidades bancarias.

- Recibir notificaciones sobre los gastos para no depender de la aplicación del banco.
- Establecer objetivos de ahorro.

Hervás menciona además la importancia de la privacidad y de que un posible sistema de notificaciones no llegue a ser invasivo para el usuario.

Por último, indica que probó la aplicación Fintonic pero la descartó por su baja flexibilidad en cuanto a ajustes y por no disponer de una vista clara del histórico de gastos.

Parte 2.B: preguntas para la persona tipo B sobre el sistema que utiliza

En esta parte, en la cual se entrevistó a cuatro personas, se pueden extraer los siguientes hechos iniciales:

- Todas las personas utilizan, como sistema alternativo a aplicaciones informáticas específicas de planificación de gastos, hojas de cálculo de Microsoft Excel u Hojas de cálculo de Google.
- Existe diversidad en cuanto al acceso a este sistema: desde diariamente hasta cada mes, pasando por varias veces a la semana.
- Para la mayoría de personas encuestadas es importante poder obtener algún tipo de media de gastos o de previsión mensual.
- Algunas de las personas encuestadas utilizan este sistema para prever qué cantidad tendrán que reservar para hacer frente a sus gastos del mes siguiente.
- Se valoran mayormente las funcionalidades relacionadas con obtener una vista general y conceptual de los gastos y con previsiones u objetivos de gasto y ahorro.

Por otro lado, se pueden extraer las siguientes funcionalidades que ofrecen las hojas de cálculo, de manera agrupada, a las personas encuestadas (entiéndase que los hechos comienzan con “el sistema permite”):

- Manejar gastos recurrentes (luz, agua, suscripciones, etc.).
- Observar el gasto mensual medio.
- Añadir comentarios para explicar gastos no previstos.
- Realizar estimaciones en base a los gastos realizados en meses anteriores.
- Obtener una cifra de gastos previstos para saber a qué cantidad se va a tener que hacer frente y, en su caso, *bloquearla* o asegurar su disponibilidad. Puede ser útil, por ejemplo, para separar una cantidad al cobrar una nómina.
- Que dicha cifra sea dinámica y se vaya actualizando conforme se añaden gastos y cobros. Puede ser útil, por ejemplo, si se tiene una cuenta bancaria específica para gastos recurrentes, de manera que siempre deberá haber en dicha cuenta, como mínimo, la cifra que indique el sistema.
- Indicar si un gasto es fijo o variable.

- Indicar cuándo empieza y cuándo acaba un gasto.
- Indicar cuándo se cobra o se paga un gasto.
- Indicar, dependiendo de si un gasto es fijo o variable, la cantidad que va a representar o una estimación.
- Registrar, una vez se conoce la cantidad de un gasto variable, el total que se pagó finalmente. Esta información puede ser útil para analizar el gasto total de un periodo y para liberar parte de una cantidad reservada, en su caso.
- Registrar ingresos de diversos tipos, como nóminas o dinero en efectivo.
- Asociar los gastos y sus cobros a un periodo mensual.
- Visualizar resúmenes de gastos generales.
- Visualizar un porcentaje que relaciona la cantidad de gasto mensual y el ahorro global del usuario, representando la capacidad de ahorro de la persona.
- Establecer objetivos de ahorro.
- Categorizar los gastos y visualizar la cantidad de gasto en cada categoría.
- Visualizar promedios y totales por categoría de gasto.
- Obtener una cifra total de gastos anuales.

Finalmente, es posible extraer una serie de funcionalidades de las cuales las hojas de cálculo carecen y que podrían ser interesantes (entiéndase que los hechos comienzan con “el sistema no permite”):

- Registrar datos concretos de los gastos (destinatario, fecha exacta, motivo, método de pago, etc.).
- Registrar los gastos y sus detalles de manera ágil, sencilla e intuitiva, desde una aplicación móvil.

Sobre otros sistemas probados, todos los usuarios indican que han probado alguna aplicación con anterioridad, pero no cubría sus necesidades en cuanto a privacidad, comparaciones mensuales, categorización o previsiones. También se menciona el excesivo número de notificaciones que envían aplicaciones de este tipo.

Otro aspecto que también mencionan varias de las personas encuestadas es el interés por disponer de una conexión con su entidad bancaria que permita obtener información sobre gastos de manera dinámica. A este respecto, se menciona también que, aunque sería interesante que los gastos se introdujeran de manera automática, tendría importancia la posibilidad de categorizar manualmente los gastos.

En cuanto a notificaciones, se mencionan las siguientes funcionalidades que parecerían interesantes a algunas de las personas encuestadas (entiéndase que los hechos comienzan con “el sistema debería permitir”):

- Recibir notificaciones si se excede una cantidad por categoría o mes.

- Recibir notificaciones cada vez que se añade un gasto de manera automática.
- Recibir notificaciones sobre previsiones de gasto de un periodo próximo a medida que se acerca el fin del periodo actual.
- Recibir notificaciones a través de distintos medios (notificaciones push, email, SMS, WhatsApp, Telegram, etc.).

Además, tres de las personas entrevistadas facilitaron los ficheros de las hojas de cálculo que utilizan. Estas hojas pueden encontrarse en el Apéndice D. Tras analizar estos documentos, se ha extraído la siguiente información, de manera adicional y complementaria a lo recogido anteriormente (entiéndase que los hechos comienzan con “el sistema permite”):

- Registrar o mantener registrados gastos recurrentes a los que se hacía frente en el pasado pero que ya han finalizado.
- Seleccionar, a nivel de gasto, si se quiere contabilizar para una previsión mensual.
- Indicar, para cada gasto, qué día o en qué momento del mes está previsto que se cobre.
- Indicar, para cada gasto, cuál es su periodo de cobro o repetición (mensual, bimensual, trimestral, ninguno de estos, etc.).
- Añadir y modificar gastos según sea necesario, no solo sus cobros puntuales.
- Visualizar claramente, para un periodo temporal, qué gastos han sido cobrados y cuáles siguen pendientes.

Parte 2.C: preguntas para la persona tipo C sobre el sistema que utiliza

En esta parte, en la que se pregunta a las personas encuestadas de forma más abstracta sobre un posible sistema de previsión de gastos que utilizarían, podemos extraer los siguientes hechos:

- Las dos personas encuestadas utilizan las aplicaciones de sus entidades bancarias varias veces por semana para comprobar sus gastos y los cobros que han recibido.
- En mayor o menor medida, ambas realizan una previsión de algunos gastos a los que van a hacer frente.
- Ninguna de las personas encuestadas reserva una cantidad para los gastos recurrentes del mes siguiente.
- A ambas personas les gustaría que un sistema automatizado o aplicación les informara sobre sus gastos recurrentes y sus cobros.
- Entre las gestiones más tediosas relacionadas con gastos recurrentes, se mencionan las suscripciones y los pagos de gastos variables.
- En cuanto a aspectos que valorarían en una aplicación, destacan alertas moderadas, usabilidad, simplicidad y visualización cómoda de datos y resúmenes.
- Ambas personas indican que les interesaría que la aplicación se conectara con su

entidad bancaria para obtener información de manera dinámica.

Parte 3: evaluación de gastos

En esta penúltima parte se pregunta a las personas encuestadas por detalles sobre sus gastos y, además, se les plantean algunas cuestiones relacionadas. Para este análisis, se recogen en primer lugar dichas cuestiones, y a continuación, los resultados relativos a los detalles.

En relación a las cuestiones, se puede extraer que se dan los siguientes tipos de situaciones, según cada cuestión:

- *“Si tienes algún gasto domiciliado, ¿tu banco te notifica cuando se produce el cobro? ¿Por qué medio lo hace (app, email...)?”*
 - Cobro de matrícula de universidad, a través de notificación push a la aplicación del banco.
 - Ingreso de la nómina en el banco, a través de notificación push a la aplicación del banco.
 - En general, cualquier cobro, a través de notificación push a la aplicación del banco. Suele contener la cantidad a cobrar y la entidad que gira el recibo.
- *“¿Qué entidades te notifican cuando se va a producir un cobro? ¿Por qué medio lo hacen (app, email...)?”*
 - Cobro de suscripción de Apple, a través de email.
 - Cobro de suscripción de Google, a través de email.
 - En general, cualquier gasto de suministros, a través de email.
- *“En caso de tener varios gastos domiciliados, ¿dispones de una única cuenta bancaria o tienes varias para distintos propósitos?”*
 - Una sola cuenta.
 - Varias cuentas bancarias con propósitos diferenciados.

En cuanto a los resultados detallados sobre gastos recurrentes, aunque estos serán de utilidad mayormente a la hora de definir el modelo de datos del sistema, se puede extraer la siguiente información:

- Gastos del hogar:
 - Alquiler, telefonía e internet suelen ser gastos fijos.
 - Agua, luz y gas, aunque pueden ser fijos, suelen ser gastos variables.
 - Alquiler, luz, gas, telefonía e internet se abonan, mayormente, en periodos mensuales.
 - El agua se abona en periodos de distinto tipo: mensual, bimensual o trimestral.
 - Los usuarios tienen bastante conocimiento sobre el día que les cobran estos

gastos.

- Suscripciones a servicios digitales:
 - Todas las personas encuestadas disponen de alguna suscripción a servicios de entretenimiento.
 - Todas las suscripciones, sea el producto o servicio que sea, son fijas.
 - La mayoría de suscripciones se abonan en periodos mensuales; solo unas pocas se abonan de manera anual.
 - Los usuarios tienen conocimiento medio o bajo sobre el día que les cobran estos gastos.
- Financiación y seguros:
 - Los gastos relacionados con financiación y seguros siempre son fijos y mensuales, y se conoce bien el día de cobro.
- Salud y bienestar:
 - Los gastos relacionados con salud y bienestar presentan numerosas diferencias en cuanto a tipo y frecuencia.
- Gastos profesionales:
 - Este tipo de gastos también presenta diversidad en cuanto a tipo y frecuencia.
- Otros gastos recurrentes:
 - Se menciona en varios casos el pago del abono de transportes, cuyo importe es fijo pero se abona según la necesidad.

Parte 4: agradecimiento final

Tras agradecer a las personas encuestadas su participación, a la pregunta “*¿Te gustaría conocer el resultado final del proyecto?*”, todas las personas responden de manera afirmativa.

3.3. Funcionalidad

En esta sección se pretende utilizar la información recogida y analizada en las encuestas para generar la potencial funcionalidad del sistema, acotar un alcance y definir distintas historias de usuario.

3.3.1. Extracción de requisitos funcionales

A partir de los resultados de las encuestas y de la visión inicial del proyecto, planteando un sistema que incluya la totalidad de las funcionalidades, se ha extraído una serie de requisitos agrupados por categorías:

1. Requisitos generales o comunes.
2. Requisitos de gastos.
3. Requisitos de cobros.
4. Requisitos de visualización y análisis.
5. Requisitos de notificaciones.

Para plantear los requisitos es necesario plantear un modelo de datos básico inicial:

- Gasto: consumo de un bien o servicio a cambio de un pago.
- Gasto recurrente: gasto que se repite a lo largo del tiempo de manera homogénea y previsible a cambio de distintos pagos.
- Pago: acción realizada para extinguir la deuda generada por un gasto o, si este es recurrente, la deuda generada por una repetición concreta del mismo.
- Notificación: aviso al usuario a través de un medio concreto para informarle sobre una novedad o un cambio de estado en su previsión de gastos.

Requisitos generales o comunes

- Importar y exportar los datos del sistema, incluido el histórico de gastos y pagos, a formatos amigables y legibles por humanos como Excel o CSV.
- Visualizar gastos y pagos realizados y previstos de manera sencilla a través de una vista de calendario diario, semanal o mensual.
- Utilizar diferentes monedas para, por ejemplo, viajes al extranjero.
- Obtener automáticamente detalles de gastos y pagos a partir de información bancaria y notificaciones.
- Establecer objetivos de ahorro.
- Asegurar la privacidad y la seguridad de los datos.
- Enfoque en la experiencia de usuario y la facilidad para registrar y visualizar los datos.
- Registrar ingresos de diversos tipos, como nóminas o dinero en efectivo.

- Mantener un histórico de gastos y pagos pasados y finalizados.

Requisitos de gastos

- Gestionar gastos de manera centralizada.
- Gestionar gastos recurrentes por periodos temporales (semanas, meses, etc.).
- Clasificar los gastos en diferentes categorías, por ejemplo: hogar, suscripciones, suministros, alimentación, limpieza, ocio, restauración, transportes, ropa, vehículos, temas laborales, mascotas, decoración, salud, inversiones y financiación.
- Introducir los gastos de forma manual.
- Configurar presupuestos de gastos por categorías.
- Contabilizar el dinero en efectivo.
- Visualizar resúmenes de gasto de manera gráfica a través de gráficos circulares o *de tarta* y por categorías.
- Adjuntar imágenes y comentarios a los gastos.
- Asociar nombres e iconos a los gastos para mejor visibilidad.
- Obtener previsiones de gastos basadas en periodos anteriores
- Registrar cuándo empieza y cuándo acaba un gasto, es decir, en qué periodo temporal se debe tener en cuenta.
- Registrar cuándo se espera que se produzcan los pagos de los gastos, sea puntuales o recurrentes, y cuál es el periodo de recurrencia (semanal, mensual, bimensual, trimestral, etc.).
- Registrar gastos con cantidades tanto fijas (por ejemplo, el servicio de internet) como variables (según el consumo, el periodo, etc., por ejemplo, la factura del gas o del agua).
- Indicar, a nivel de gasto, si se quiere contabilizar este para las previsiones mensuales o no.
- Indicar si los gastos están domiciliados o no.

Requisitos de pagos

- Asociar distintos pagos a un mismo gasto recurrente.
- Introducir los pagos de forma manual.
- Adjuntar imágenes y comentarios a los pagos.
- Obtener previsiones de pagos basadas en periodos anteriores.
- Detección automática de actualizaciones de detalles de pagos desde información bancaria o notificaciones.
- Solicitar aprobación manual para registrar pagos cuando el sistema los detecte automáticamente desde el banco, para tener constancia de que se han producido y evitar que se registren de manera incorrecta.

- Creación automática de pagos previstos a partir de la información de los gastos, su recurrencia, el periodo mensual, su fecha de pagos y la cantidad esperada.
- Registrar la cantidad final que va a suponer un pago y su fecha de realización, una vez se conoce, ya sea por aviso del banco o de la entidad, o porque ya se haya realizado.
- Contabilizar cada pago en su periodo mensual correspondiente, incluso si se retrasa unos días.

Requisitos de visualización y análisis

- Visualizar rápida y fácilmente lo gastado en un periodo mensual y lo que queda de gastos y cobros previstos, incluyendo los pagos pendientes del periodo o periodos anteriores, que pueden haberse retrasado.
- Disponer de una previsión de los gastos y cobros que se van a producir en el siguiente periodo o en los días y semanas próximos.
- Disponer de un análisis por periodos mensuales que esté disponible al día siguiente de acabar el mes.
- Visualizar resúmenes de gastos generales.
- Visualizar un porcentaje que relaciona la cantidad de gasto mensual y el ahorro global del usuario, representando la capacidad de ahorro de la persona.
- Visualizar la cantidad de gasto en cada categoría.
- Visualizar promedios y totales por categoría de gasto.
- Obtener una cifra total de gastos anuales.

Requisitos de notificaciones

- Recibir notificaciones sobre actualizaciones de estado y detalles de gastos y cobros.
- Recibir notificaciones sobre el cierre de periodos mensuales.
- Recibir notificaciones si se excede una cantidad por categoría o periodo mensual.
- Recibir notificaciones sobre previsiones de gasto de un periodo próximo a medida que se acerca el fin del periodo actual.
- Recibir notificaciones a través de distintos medios (notificaciones *push*, correo electrónico, SMS, WhatsApp, Telegram, etc.).

3.3.2. Alcance funcional

El alcance de este proyecto se centra, en base a lo definido en la introducción (1), en el objetivo de liberar al usuario del coste y tiempo de la gestión de sus gastos personales, específicamente a través de las siguientes dos capacidades:

- La capacidad para registrar gastos puntuales y recurrentes y consultarlos posteriormente.

- La capacidad para conocer el estado de dichos gastos y los cambios relacionados con los mismos de forma automática y dinámica.

Así, el alcance específico del proyecto se define a través de los siguientes puntos:

- Registro de gastos puntuales.
- Registro de gastos recurrentes.
- Listado de gastos.
- Consumo de información bancaria a partir de un agregador.

Considerando este proyecto como una prueba de concepto, en la que se centra el esfuerzo en las funcionalidades definidas por los objetivos, no se ha estimado oportuno introducir funcionalidades como las relacionadas con seguridad (inicio de sesión, permisos, etc.), con experiencia de usuario (uso de sistemas de notificaciones fiables, uso de comunicación del servidor al cliente web, etc.), con pruebas de software (aunque si se han implementado algunas puntuales en el código y de forma manual) y demás. Estas funcionalidades sí podrían tenerse en cuenta en posibles evoluciones del proyecto y del sistema, como se describe en el apartado de trabajo futuro (6.3).

En relación a la integración con fuentes de información bancaria, de las tres que se analizan en la Sección 2.2 se ha elegido limitar el alcance a la conexión y comunicación con el agregador bancario, por el interés académico del desarrollo y por servir como demostración para el procesamiento derivado de las otras dos alternativas no cubiertas.

La implementación de la funcionalidad arriba detallada se ha realizado, desde el punto de la interacción entre el usuario y el sistema, a través de una aplicación web.

3.3.3. Historias de usuario y casos de uso

A partir de los requisitos funcionales y del alcance definido, se generan historias de usuario para guiar la implementación final del sistema. Cada historia de usuario permitirá completar uno o varios requisitos y hará más sencillo validar, de manera granular y también general, la correcta implementación y funcionamiento de cada uno de estos.

En este proyecto, las historias de usuario se han definido de una manera más cercana al concepto de casos de uso en Agile, es decir, se define un resumen de tipo *como X quiero Y para Z*, pero además se añade un desglose sobre el camino o los pasos a seguir, los componentes que intervienen y los criterios de aceptación finales. La intención de este formato es conseguir de un vistazo una idea general de las partes a modificar y el resultado que se debe obtener.

Las historias se definen desde el punto de vista del aporte de valor para el usuario. Así, no se define una historia que arranque con “cuando el agregador bancario disponga de nueva información”, sino que se enfoca a la intención del usuario de estar al tanto de las actualizaciones de sus gastos.

HU1. Registrar un gasto

Como usuario de Balancer, quiero registrar un gasto para mantenerme al tanto de su estado y visualizarlo posteriormente.

Camino:

1. El usuario solicita la vista con el formulario para registrar un gasto.
2. La aplicación web muestra al usuario dicha vista.
3. El usuario rellena el formulario.
4. El usuario envía el formulario.
5. La aplicación web valida el formulario y lo envía al microservicio Core.
6. El microservicio Core confirma a la aplicación web el registro correcto.
7. La aplicación web confirma al usuario el registro del gasto.

Componentes:

1. Aplicación web
2. Microservicio Core

HU2. Consultar gastos previamente registrados

Como usuario de Balancer, quiero listar los gastos que había registrado previamente para conocer sus datos y su estado.

Camino:

1. El usuario solicita la vista de gastos.
2. La aplicación web solicita al microservicio Core la información de los gastos.
3. El microservicio Core proporciona la información de los gastos.
4. La aplicación web muestra al usuario la vista de gastos.

Componentes:

1. Aplicación web
2. Microservicio Core

HU3. Asociar una cuenta bancaria

Como usuario de **Balancer**, quiero asociar una de mis cuentas bancarias para tener la posibilidad de ser notificado sobre mis gastos posteriormente.

Camino:

1. El usuario solicita la vista con el formulario para asociar una cuenta.
2. La aplicación web muestra al usuario dicha vista con el *widget* del agregador bancario para asociar la cuenta.
3. El usuario completa los pasos del *widget*.
 - Los pasos del proceso del *widget* se omiten por no depender propiamente de este sistema y por ser específicos de cada agregador bancario.
 - Al completar el proceso del *widget*, este devolverá ciertos datos para continuar con la asociación de la cuenta. Esto se detalla más adelante, en la Sección 5.3.1.
4. La aplicación web envía los datos obtenidos del widget al conector bancario.
5. El conector bancario confirma a la aplicación web la correcta asociación de la cuenta.
6. La aplicación web confirma al usuario la correcta asociación de la cuenta.

Componentes:

1. Aplicación web
2. Microservicio conector bancario

HU4. Consultar las cuentas previamente asociadas

Como usuario de **Balancer**, quiero listar mis cuentas bancarias asociadas para conocer cuáles he asociado previamente y de cuales recibiré notificaciones.

Camino:

1. El usuario solicita la vista de cuentas asociadas.
2. La aplicación web solicita al conector bancario la información de los cuentas asociadas.
3. El conector bancario proporciona la información de los gastos.
4. La aplicación web muestra al usuario la vista de gastos.

Componentes:

1. Aplicación web
2. Microservicio conector bancario

HU5. Ser notificado cuando se paga un gasto previsto

Como usuario de **Balancer**, quiero que se me notifique cuando se pague un gasto previsto para ser consciente de dicho evento de manera inmediata.

Camino:

1. El agregador bancario avisa al conector bancario sobre la transacción a través de un *webhook*.
2. El conector bancario recibe el aviso a través del *webhook*.
3. El conector bancario solicita al agregador bancario los datos de la transacción.
4. El agregador bancario envía al conector bancario los datos de la transacción.
5. El conector bancario decodifica y preprocesa los datos de la transacción, los convierte a un formato legible para el microservicio Core, extrayendo diferentes parámetros, y se los envía a este.
 - a) Nota para la definición técnica posterior: el conector bancario deberá tener el motor (con lógica basada en reglas, inteligencia artificial, plantillas, etc.) capaz de detectar una transacción relevante y extraer información de la misma.
6. El microservicio Core recibe los datos y realiza las siguientes acciones:
 - a) Identifica el gasto previsto recurrente asociado a la transacción.
 - 1) Nota para la definición técnica posterior: esta identificación se debe realizar en base a ciertos parámetros como la entidad destinataria y el concepto. En última instancia también se podrían utilizar la fecha de valor y el importe consolidado, ya que si hay un pago previsto coincidente, se podría esperar que la transacción tuviera relación.
 - b) Identifica y asocia los parámetros obtenidos de los detalles de la transacción que deberían actualizarse en el gasto, por ejemplo, la fecha de pago o la cantidad final.
7. El microservicio Core actualiza el gasto y el pago y registra la nueva información.
8. El microservicio Core envía una notificación al usuario indicándole que se ha recibido un cambio de estado de un gasto.
9. El usuario recibe la notificación.

Componentes:

1. Microservicio conector bancario
2. Microservicio Core

3. Microservicio de notificaciones

Capítulo 4

Diseño del sistema

4.1. Nombre y marca del sistema

Para dar unicidad al proyecto de cara al público y disponer de una marca de referencia, tras analizar diferentes opciones en base a la temática en cuestión (finanzas personales, presupuestos, cuentas bancarias, etc.), se decide darle al proyecto el nombre de Balancer.

Este nombre se utiliza, por el momento, como referencia en las partes de esta memoria asociadas al diseño y la implementación del sistema, así como en los repositorios y documentos publicados que acompañan al proyecto.

Se ha definido además una marca muy básica con un icono de aplicación y unos colores, visibles en la aplicación web diseñada.



Figura 4.1: Logotipo del proyecto Balancer

4.2. Arquitectura del sistema

4.2.1. Introducción

El concepto de arquitectura en el ámbito de los sistemas de software se refiere al conjunto de decisiones técnicas que determinan la estructura y el comportamiento de dichos sistemas (University, 2021). Se trata de un elemento fundamental en el desarrollo de software, ya que proporciona una visión global de los sistemas, la cual funciona como base para construir la funcionalidad diseñada y asegurar su éxito.

A la hora de diseñar un sistema de software, es pertinente definir su arquitectura para poder manejar la complejidad del mismo, su evolución y el cumplimiento de los requisitos definidos, tanto funcionales como no funcionales.

Una arquitectura bien definida ayuda a reducir errores y mejorar aspectos como el rendimiento y la seguridad, además de incidir en la reducción de costes y mantenimiento y, a nivel de desarrollo, ayudar en la mejora de la comunicación y colaboración entre potenciales equipos de desarrolladores.

La definición y documentación de las arquitecturas de software facilita, además, la comunicación entre *stakeholders*, por la manera en que se representan las decisiones de diseño de alto nivel y la capacidad de reutilización de componentes entre diferentes proyectos.

La arquitectura de este proyecto, incluyendo estas características y observaciones, se define más adelante en esta sección.

4.2.2. Microservicios

Las arquitecturas de microservicios son un estilo de arquitectura basado en estructurar los sistemas como colecciones de servicios que cumplen una serie de características.

En primer lugar, cada microservicio debe ser desplegable de manera independiente. Esto no solo implica aspectos técnicos de compilación o publicación de ejecutables, sino que también significa que, cada microservicio, tras ser *testado* de manera aislada, está listo para su uso en entornos productivos. Esta característica permite también utilizar diferentes pilas o *stacks* tecnológicas y diferentes lenguajes en algunos microservicios, si así se requiriera.

En relación a las pruebas, los microservicios deben ser *testables* de manera unitaria y aislada, utilizando *test doubles* o simuladores de comportamiento del resto de servicios con los que se comunica. Esta característica, generalmente automatizable, posibilita la aceleración del desarrollo y la evolución del microservicio mientras que se mantiene la calidad.

Para que se pueda cumplir la propiedad anterior, es necesario que los microservicios estén débilmente acoplados. Esto quiere decir que, si bien pueden comunicarse entre sí, la especificación del comportamiento y de la vía de interacción con cada microservicio debería ser independiente a la del resto. En este aspecto es importante que dicha especificación esté bien definida y se conozca por parte del resto de microservicios y equipos. Este tema se trata más adelante en esta memoria.

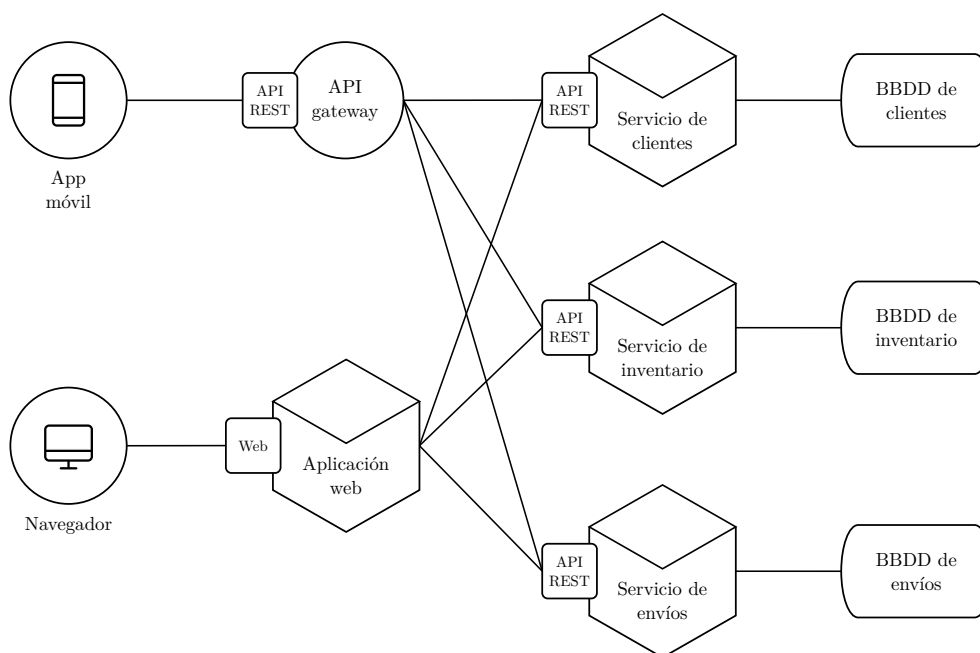


Figura 4.2: Diagrama de un ejemplo de arquitectura de microservicios

Otra propiedad importante de este tipo de arquitecturas es que cada microservicio debe tener una responsabilidad bien definida y relacionada con la lógica de negocio. En la Figura 4.2 se puede ver un ejemplo en el que tres microservicios en el *backend* se encargan de diferentes entidades del dominio del sistema: clientes, inventario o productos, y envíos. Cada microservicio es responsable de la persistencia en base de datos de estas entidades, y el resto de microservicios que necesiten acceder a estas, deberían hacerlo a través del microservicio responsable.

En este aspecto, haciendo un símil con el desarrollo de software a más bajo nivel, Toby Clemson (ThoughtWorks) define la arquitectura de microservicios como la consecuencia natural de aplicar el principio de responsabilidad única en el nivel estructural.

En el mismo ejemplo, el servicio de aplicación web actúa como *proxy* entre el navegador y los microservicios de *backend*. En el caso de la aplicación web, se define una API *gateway* para proteger y distribuir las peticiones según qué microservicio sea el responsable de atender cada una.

Al definir una arquitectura de este tipo se obtienen servicios sencillos con pequeños subdominios, lo que permite que sean más fáciles de entender y mantener.

En este tipo de arquitecturas, las operaciones definidas sobre el dominio pueden implicar a varios microservicios. Por ejemplo, al crear un pedido desde la aplicación web, se hará uso del servicio de clientes para verificar la identidad, se confirmará la disponibilidad de los productos a través del servicio de inventariado, y se generará un envío a través del servicio de envíos.

4.2.3. Definición de la arquitectura

Para definir la arquitectura del sistema es necesario tener, en primer lugar, una visión global del contexto en el que funcionará. El siguiente paso será, teniendo en cuenta

la funcionalidad definida, definir la composición interna y las distintas partes del sistema.

En estos casos es conveniente utilizar el modelo C4 (Brown, 2018), una técnica de notación gráfica de sistemas software con distintos niveles de abstracción que permite conocer de un vistazo y de manera clara la organización del sistema. El nombre del modelo proviene de los cuatro niveles de abstracción que define: contexto, contenedores, componentes y código.

Para la definición de la arquitectura de este proyecto conviene utilizar los dos primeros niveles de abstracción, pues ofrecen un nivel de abstracción adecuado para el propósito que se busca: mostrar los microservicios definidos, el usuario y los componentes externos.

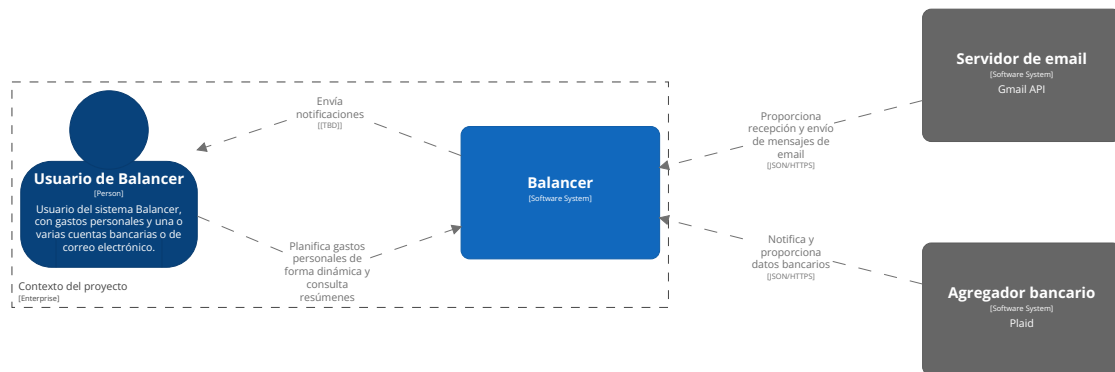


Figura 4.3: Diagrama de modelo C4 del contexto del sistema

El diagrama de contexto (Figura 4.3) permite obtener un primer nivel de perspectiva sobre el principal actor, la situación del sistema y los servicios externos. Una persona puede interactuar directamente con el sistema y este, a su vez, puede interactuar con un servidor de correo electrónico y un agregador bancario.

La persona, en azul oscuro y nombrada como usuario de Balancer, es el actor principal para la funcionalidad y el alcance del sistema. Esta persona dispone de una o varias cuentas bancarias y/o de correo electrónico e interactúa con el sistema planificando sus gastos personales de forma dinámica, recibiendo notificaciones sobre estos y consultando resúmenes y análisis.

El sistema Balancer, en el centro y con un color azul más claro, es el sistema software que se está desarrollando. Los otros dos sistemas con los que interactúa, externos, se representan en color gris. Aunque el alcance definido no cubra el uso del correo electrónico como fuente de información de gastos, se estima útil colocarlo en el diagrama para un posible desarrollo futuro.

Al bajar un nivel de abstracción, en el diagrama contenedores (Figura 4.4) se pueden observar las piezas que compondrán el sistema internamente y permitirán ofrecer e implementar la funcionalidad definida.

La división de los contenedores se lleva a cabo con la intención de crear una arquitectura de microservicios que realizan distintas tareas y tienen distintos requisitos de tecnologías y fuentes de datos, como se explica anteriormente.

En el centro del diagrama se encuentra el contenedor del microservicio Core, encargado de coordinar el resto de partes y proveer la funcionalidad del sistema al usuario a

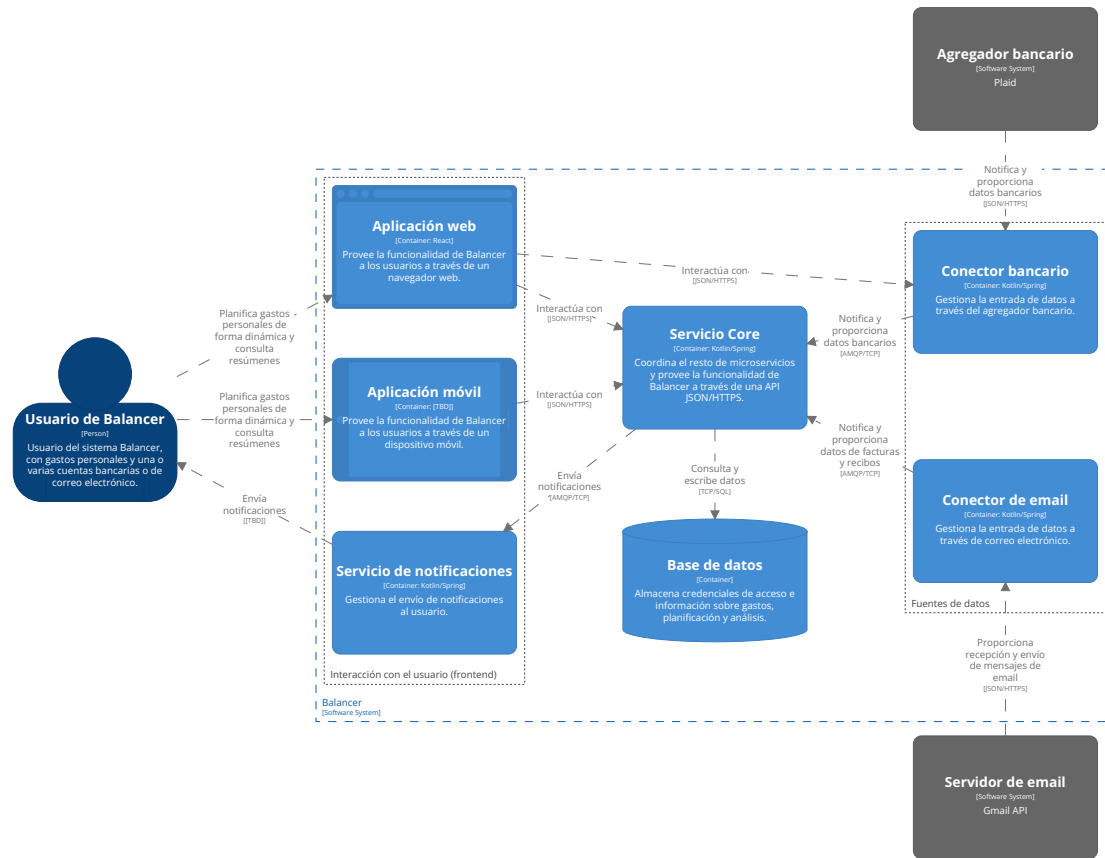


Figura 4.4: Diagrama de modelo C4 de contenedores

través de los servicios que interactúan con este. Este servicio dispone de una conexión a un sistema de persistencia, es decir, a una base de datos, en la cual se almacenan credenciales de acceso e información sobre gastos, planificación y análisis.

En el grupo de interacción con el usuario o *frontend* se definen tres contenedores:

- **Aplicación web:** permite al usuario hacer uso de la funcionalidad del sistema a través de un navegador web.
- **Aplicación móvil:** permite al usuario hacer uso de la funcionalidad del sistema a través de un dispositivo móvil.
- **Servicio de notificaciones:** gestiona el envío de notificaciones al usuario.

Por simplicidad, la aplicación web se comunica directamente tanto con el microservicio Core como con el conector bancario. En versiones futuras del proyecto, convendría que los componentes de *frontend* solo se comunicaran con un servicio que protegiera y filtrara los accesos, siguiendo un patrón de *API gateway* o *backend for frontend* (Richardson, 2023b).

Por otro lado, en el grupo de fuentes de datos, se definen dos contenedores que conectan con sendos servicios externos:

- Conector bancario: funciona como intermediario entre el agregador bancario y el servicio Core recibiendo eventos, obteniendo información de movimientos y transacciones y preprocesando los datos para que estos puedan ser de utilidad para sistema. Esta colaboración entre servicios permite que el usuario gestione sus gastos de una manera más dinámica sin necesidad de introducir toda la información de forma manual.
- Conector de email: de manera similar al conector bancario, se sitúa entre el servidor de email y el servicio Core para captar mensajes de correo electrónico que puedan aportar valor para el sistema, por ejemplo, cuando una entidad informe sobre un pago próximo o envíe una factura o un recibo.

Al igual que en el diagrama de contexto, en este diagrama se muestran algunas partes que no se incluyen en el alcance pero que podrían ser interesantes para evoluciones del sistema. Estos componentes son la aplicación móvil, el servicio de notificaciones y el conector de email.

Con respecto a la comunicación entre los distintos servicios, se ha estimado oportuno definirla de forma directa a nivel técnico en la Sección 5.1.

4.2.4. Organización por capas

A nivel de cada servicio de *backend*, para este proyecto se ha escogido aplicar un patrón de organización de software por capas. Se trata un patrón con bastante implantación cuyo principal objetivo es crear sistemas fáciles de mantener y evolucionar, además de robustos y flexibles, de manera similar al enfoque de la arquitectura de microservicios, pero a un nivel más bajo, es decir, en cada servicio.

Esta organización distingue una serie de capas con responsabilidad única en cada servicio, como pueden ser la capa de dominio, la capa de aplicación o la capa de infraestructura. Esta representación, que más adelante se llevará a la implementación, permite que las diferentes capas sean independientes y que el mantenimiento y la evolución de los microservicios sea más sencillo a lo largo del tiempo.

Para la representación visual de este tipo de organización se ha escogido un formato de diagrama similar al propuesto por Alistair Cockburn en su conocida como arquitectura como hexagonal, arquitectura de cebolla o patrón de puertos y adaptadores.

En la Figura 4.5 se muestra un diagrama que representa cómo la lógica de negocio, dividida en las capas de dominio y aplicación, se encuentra contenida en el núcleo o *core* del servicio. La capa de aplicación contendrá las operaciones y casos de uso aplicables sobre el dominio.

Alrededor del núcleo se disponen diferentes puertos de entrada y salida de la capa de infraestructura que, a través de adaptadores, reciben o envían información de servicios externos, clientes, bases de datos, etc.

Cuando un cliente desea utilizar el servicio, envía una petición que llega a la capa de aplicación tras atravesar un puerto y previamente pasar por un adaptador. Este adaptador permite que un mismo caso de uso pueda accionarse a partir de diferentes tecnologías, por ejemplo, una petición recibida por REST o un evento consumido por AMQP. De manera

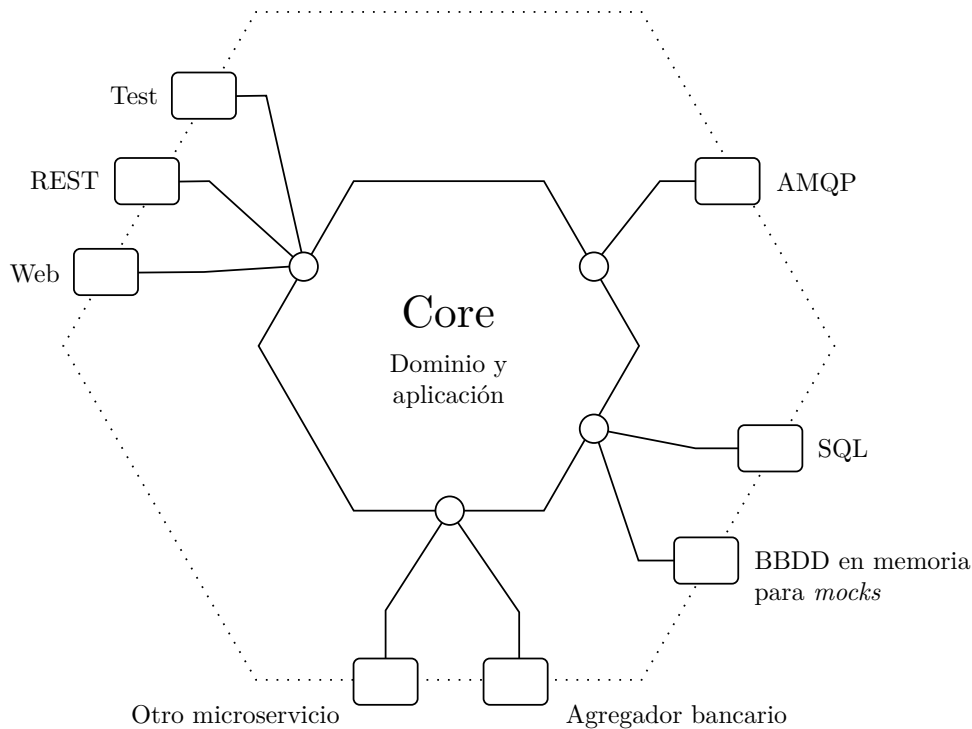


Figura 4.5: Diagrama de ejemplo de un servicio con organización por capas

similar, cuando la aplicación necesita enviar algo al exterior o solicitar información, hace uso de alguno de los puertos, que a su vez se conectarán con la infraestructura a través de distintos adaptadores.

La implementación debe permitir que se pueda interactuar con el núcleo de la aplicación, y que este pueda interactuar con su entorno y el exterior, sin necesidad de conocer la naturaleza concreta de los adaptadores *físicos* que se sitúan al otro lado de los puertos.

La principal ventaja de este patrón es la posibilidad de sustituir adaptadores y tecnologías cuando sea necesario sin tener que modificar las capas de aplicación o dominio. Por ejemplo, si un servicio externo cambia de REST a gRPC (llamada a procedimiento remoto), bastaría con ajustar el adaptador y el servicio seguiría funcionando de la misma manera.

A nivel de *testing*, y en relación con la característica de la arquitectura de microservicios sobre las pruebas de forma aislada, el patrón de puertos y adaptadores ofrece la posibilidad de establecer *test doubles* o simuladores de comportamiento de servicios externos como puertos alternativos. Así, durante la ejecución de un plan de pruebas, manual o automatizado, en lugar de realizar llamadas a la base de datos real, se podría establecer una conexión a una base de datos en memoria con datos de prueba (Figura 4.5), de nuevo, sin necesidad de modificar el núcleo de la aplicación.

4.2.5. Microservicio Core

Una vez definidos los parámetros y la arquitectura que se quiere conseguir, se detallan uno a uno los componentes implicados.

El microservicio Core se coloca en el centro de la arquitectura y aglutina toda la

lógica de gestión y procesamiento de gastos y usuarios. También realiza algunas labores de orquestación entre los distintos componentes.

Es el encargado de recibir las peticiones de las aplicaciones web y móvil para consultar y modificar gastos. Además, se ocupa de recibir actualizaciones sobre transacciones bancarias, preprocesadas por el conector bancario, y a partir de estas identificar gastos y aplicar cambios sobre estos.

4.2.6. Microservicio conector bancario

El microservicio conector bancario es una de las piezas fundamentales del proyecto, pues permite que la información sobre transacciones y movimientos bancarios se pueda obtener de manera programática y dinámica.

Este microservicio se comunica con el agregador bancario, Plaid, según se explica en la Subsección 2.2.1, y realiza una labor de transformación y preprocesamiento de la información, de manera que el microservicio Core recibe únicamente datos útiles y en el formato requerido.

También es competencia del conector bancario gestionar los flujos por los cuales un usuario de Balancer se identifica en su banco y autoriza al sistema a acceder a su información, todo a través del agregador bancario. Entrando en un mayor nivel de detalle, y aunque no se incluye la persistencia en el diagrama de contenedores por simplicidad, esta tarea incluye almacenar la información y las credenciales asociadas al usuario necesarias que el sistema pueda obtener los datos de transacciones bancarias con continuidad en el tiempo.

Además, el conector bancario realiza una función de desacoplamiento y asincronía con respecto al agregador bancario, ya que contiene lógica para recibir los eventos y obtener la información actualizada sin mantener al microservicio Core o al propio agregador bloqueados hasta que se procesen los datos.

4.2.7. Aplicación web

La aplicación web es el servicio definido para la interacción entre el usuario de Balancer y el sistema. Se trata de una interfaz visual, servida a través de un navegador web, que muestra al usuario la información sobre los gastos y le permite operar sobre estos.

4.3. Modelo de datos

4.3.1. Gasto

Un gasto es un elemento asociado a un usuario que representa una retirada de su crédito, realizada o a realizar en el futuro.

Tras analizar e iterar sobre los detalles recogidos en el Capítulo 3, se extrae que los gastos podrían ser de dos tipos principales: puntual y recurrente. La división se basa en los estados en que pueden encontrarse, los pagos asociados y la existencia de recurrencia en el segundo tipo.

Ambos tipos tienen las siguientes propiedades en común:

- **Identificador.**
- **Estado:** varía según el tipo.
- **Categoría:** un gasto puede categorizarse como alquiler, hogar, servicios, financiación, salud, etc.
- **Destinatario o entidad acreedora.**
- **Concepto.**
- **Comentarios.**
- **Cantidad o importe.**
- **Tipo de cantidad o importe:** puede ser de dos tipos:
 - **Fijo:** la cuota del gasto no puede variar y se conoce previamente.
 - **Aproximado al alza:** la cuota es variable, por lo que el importe especificado es una aproximación por lo alto, dentro de la cual se prevé que *quepa* la cantidad final consolidada.
- **Método de pago:** puede ser efectivo, tarjeta bancaria o domiciliación bancaria.
- **Pago o pagos asociados:** varía según el tipo.
- **Oculto en las previsiones:** si se incluye o no en las previsiones de cada periodo y se tiene en cuenta o no para los cálculos de estas.

4.3.2. Gasto puntual

Un gasto puntual es un subtipo de gasto que solo se realiza una vez, sea en el pasado o en el futuro. Serían gastos puntuales, por ejemplo, la compra de un billete de tren o la adquisición de un teléfono móvil.

Para este tipo de gastos, además de las propiedades comunes de un gasto, se especifican las siguientes:

- **Estado:** un gasto puntual puede encontrarse:
 - **Previsto o pendiente:** se conoce que el gasto se va a producir pero aún no se ha realizado.

- **Pagado o finalizado:** el crédito ya se ha retirado al usuario.
- **Pago asociado:** debe ser único. Se detalla más adelante, en la Subsección 4.3.4.

4.3.3. Gasto recurrente

A diferencia de un gasto puntual, un gasto recurrente representa la previsión de un usuario de retiradas periódicas de su crédito, asociadas a un mismo concepto. En este tipo de gastos entran, por ejemplo, las suscripciones y los suministros del hogar (electricidad, internet, telefonía, etc.).

Este tipo de gastos tiene asociadas, además de las propiedades comunes de un gasto, las siguientes:

- **Estado:** un gasto recurrente puede encontrarse:
 - **Activo o previsto:** el gasto se está pagando actualmente y se tiene en cuenta, es decir, se prevé que se produzcan pagos asociados a este gasto en el futuro, y potencialmente se habrán producido pagos asociados en el pasado.
 - **Completado o finalizado:** el período de previsión de este gasto ha finalizado y ya no se debe tener en cuenta.
- **Frecuencia:** definición del periodo de facturación o cobro del gasto, incluyendo el momento de cobro. Puede ser mensual un día específico, mensual el último día de cada mes, semanal, etc.
- **Fecha del primer pago:** fecha en la que se produjo el primer pago o a partir de la cual se producirá el gasto.
- **Fecha del pago final o último pago:** fecha en la que se realizará o realizó el pago final del gasto, es decir, a partir de la cual el gasto figurará como completado o finalizado.
- **Pagos asociados:** pueden existir múltiples, correspondientes a cada periodo de facturación. Se detallan más a continuación.

4.3.4. Pago

Un pago asociado a un gasto representa un cobro o retirada de crédito asociada a dicho gasto. En el caso de los gastos puntuales solo puede existir un pago asociado, mientras que los gastos recurrentes pueden tener varios asociados. Por ejemplo, en el caso de un gasto referente a alquiler, un pago concreto podría representar el pago del mes de enero.

Cada pago tiene asociadas las siguientes propiedades:

- **Identificador.**
- **Estado:** puede estar pendiente o consolidado. Tiene mayor significado para pagos asociados automáticamente a partir del agregador bancario.
- **Cantidad.**

- **Fecha:** fecha en la que se realizó o consolidó el pago.
- **Fecha de autorización:** fecha en la que el banco autorizó el pago, que suele representar la fecha en la que el usuario llevó a cabo la acción. Solo tiene significado para pagos asociados automáticamente a partir del agregador bancario.

Las propiedades de estado y fecha de autorización se definen a partir de la especificación del agregador bancario Plaid, como datos que pueden interesar al usuario directamente y a través de las previsiones por periodos.

4.3.5. Previsión por periodo

Una previsión por periodo representa una recopilación de todos los pagos asociados a gastos puntuales y recurrentes previstos dentro de un periodo temporal. En general, por consensos sociales en cuanto a periodos de facturación y pagos de nóminas, las previsiones cubren periodos mensuales, es decir, entre el primer y el último día de cada mes.

Cada previsión mensual tiene asociada la siguiente información:

- **Periodo concreto que cubren** (semana, mes, año, etc.).
- **Gastos y pagos previstos incluidos.**
- **Estado:** una previsión puede encontrarse:
 - **Prevista:** si la fecha de generación es anterior al periodo mensual y ningún cobro está en estado pendiente.
 - **En curso:**
 - si la fecha de generación está dentro del periodo mensual,
 - si es anterior y algún cobro está en estado efectuado, o
 - si es posterior y algún cobro está en estado pendiente.
 - **Finalizada:** si todos los cobros están en estado efectuado.
- **Fecha y hora de generación.**

4.4. Asociación transacción-gasto y aplicación de cambios

Una vez definida la gestión de gastos y el modelo de datos, se planifica el aprovechamiento de la información provista por el agregador bancario para aplicar cambios sobre dichos gastos.

Plaid proporciona actualizaciones sobre las transacciones bancarias a través de tres grupos: añadidas, modificadas y retiradas. Esto se debe a la forma de gestionar los pagos por parte de las entidades bancarias, que pueden registrar transacciones como autorizadas pero no realizadas, o realizar retenciones que finalmente se materializarán en pagos o no, siendo retiradas del histórico de transacciones.

El sistema debe ser capaz, por un lado, de identificar, dada una actualización de una transacción, el gasto al que esta debe ir asociada. Por otro lado, una vez identificado el gasto, es necesario disponer de la capacidad para aplicar cambios al gasto en función de la información que proporciona la actualización de la transacción. Estas dos funcionalidades se pueden considerar y abarcar de forma separada, facilitando su prueba unitaria y la sustitución de la implementación de cada una, según se necesite.

4.4.1. Asociación entre una transacción bancaria y un gasto registrado

Esta función debe recibir como entrada una transacción y como salida un gasto registrado previamente en el sistema.

En el caso de las transacciones retiradas, Plaid solo facilita el identificador de la transacción, por lo que será necesario asociar estos a cada gasto en el momento de registrar una transacción añadida o modificada.

Para las transacciones modificadas, aprovechando la información mencionada en el párrafo anterior, es trivial la asociación a un gasto si este ya tenía asociado el identificador de la transacción previamente. En caso contrario, se debe proceder como una transacción añadida.

Cuando se recibe una actualización relativa a una transacción añadida, el sistema debe procesar la información contenida en el evento (importe, fecha, nombre, vendedor, si está pendiente, categoría, etc.) (Plaid, 2023b) y la registrada para el gasto, según se especifica en Sección 4.3.

El enfoque escogido para este proyecto se basa en una lógica puntuaciones y reglas de comparación y cálculos entre campos. Dada una transacción y un identificador de usuario (añadido por el propio sistema), se aplican las reglas para cada gasto y se devuelve el de mayor puntuación, siempre y cuando alcance un mínimo.

En un primer acercamiento a la implementación de las reglas, se definen, de manera abstracta las siguientes:

1. Si la categoría de finanzas personales de la transacción coincide con la categoría del gasto, se añade un nivel de puntuación alto.

2. Se comparan las cantidades de la transacción y el gasto. Si la diferencia es pequeña, se añade un nivel de puntuación moderado. Si la diferencia es grande, se resta un nivel de puntuación moderado. En caso de que la diferencia esté dentro del rango entre pequeño y grande, no se modifica la puntuación. La justificación de esta regla se basa en la capacidad del usuario de prever la cantidad del gasto, que puede variar con la final.
3. Se comparan las palabras en común entre el concepto de la transacción y el del gasto, tras pasar una normalización de tildes y mayúsculas. Si el número es grande, se añade un nivel de puntuación moderado. Si el número es pequeño, se resta un nivel de puntuación moderado. En caso de que la diferencia esté dentro del rango entre pequeño y grande, no se modifica la puntuación. En este caso, la justificación tiene su base en la posibilidad de variación del lenguaje natural.

Cabe destacar que este enfoque se puede refinar, especialmente a través de iteraciones con datos de prueba y validación. También se pueden utilizar enfoques similares, como la utilización de la distancia de Levenshtein para comparación entre cadenas de texto, o radicalmente distintos, como modelos de inteligencia artificial y aprendizaje automático. Estos últimos enfoques y podrían ofrecer un mejor resultado dados los campos *libres* (en lenguaje natural) de los modelos de transacción y gasto.

Según la calidad de los resultados de cada enfoque, podría ser necesaria la confirmación por parte del usuario para asociar una transacción a un gasto. También es posible que no haya ningún gasto asociable a una transacción dada, en cuyo caso se ignoraría o, como posible funcionalidad, podría sugerirse el registro de un nuevo gasto.

4.4.2. Aplicación de cambios a un gasto a partir de una transacción

Una vez identificado el gasto asociado a una transacción, el sistema debe actuar sobre este en base a la información notificada.

Esta actuación se define también en base a las propiedades de ambos modelos. Por ejemplo, si se añade una transacción consolidada asociada a un gasto puntual que estaba en estado *previsto*, el gasto pasará a *pagado*.

El algoritmo de aplicación de cambios puede alcanzar una gran complejidad dada la cantidad de datos disponibles, las combinaciones posibles y la necesidad de mantener la coherencia entre la información facilitada por el usuario y la recibida por parte del agregador bancario.

En la Figura 4.6 se puede ver representado un posible algoritmo de aplicación de cambios que, tras recibir un evento de actualización de transacción, aplica diferentes operaciones condicionales y mutaciones sobre el gasto previamente registrado.

En esta parte del sistema es de gran importancia la normalización que realiza el agregador bancario sobre los datos provistos por las diferentes entidades, pues cada una puede tener una distinta interpretación sobre campos como la fecha del pago, la fecha de autorización o el estado de la transacción, y el agregador unifica campos y significados aportando consistencia y liberando al sistema de una gran carga de preprocesamiento.

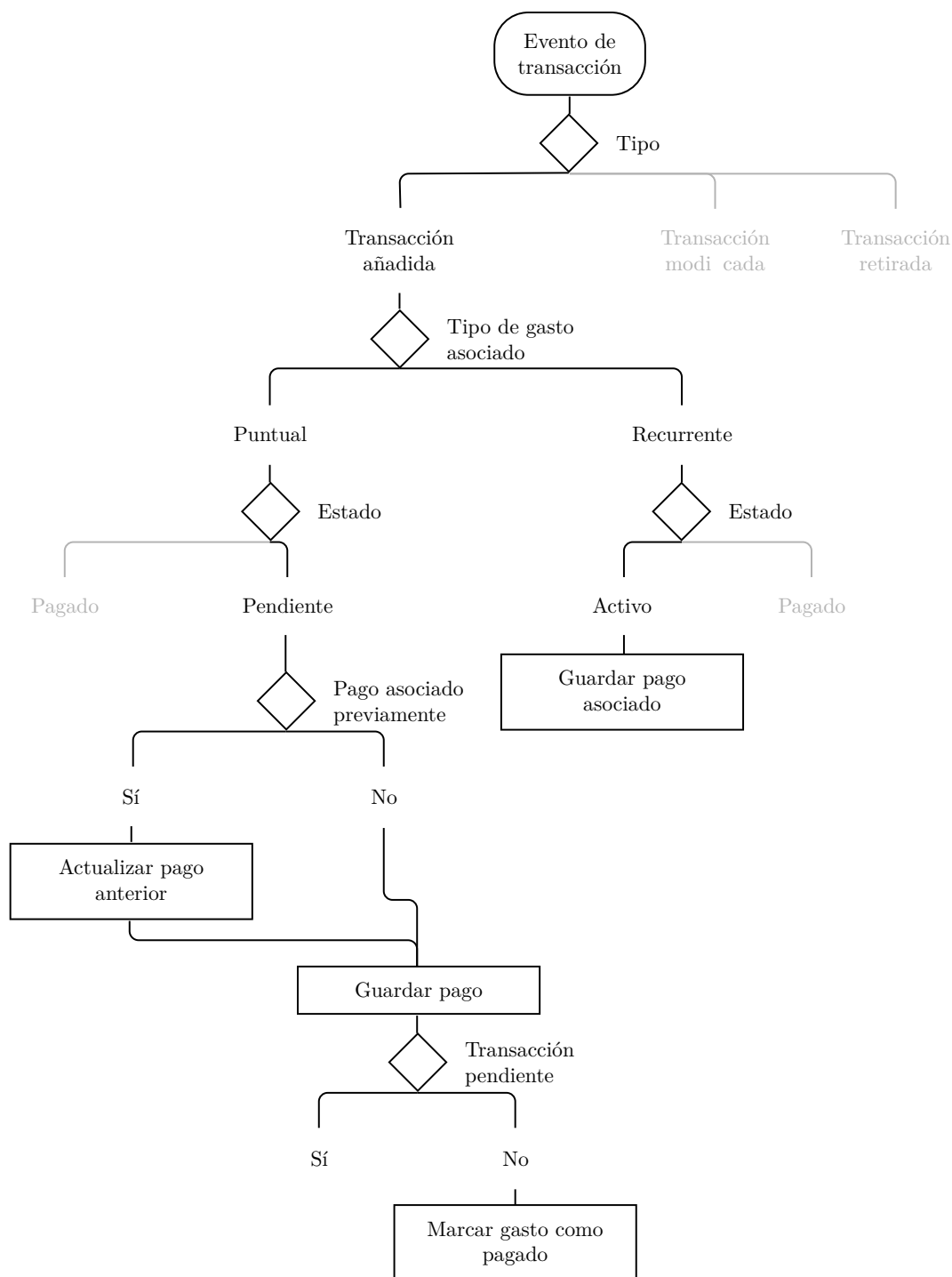


Figura 4.6: Diagrama de flujo del algoritmo de aplicación de cambios a un gasto en función de una transacción

Los rombos representan condiciones y diversificación de ramas del algoritmo, mientras que los rectángulos no redondeados representan operaciones a realizar.

Capítulo 5

Desarrollo e implementación del sistema

5.1. Lenguajes, *frameworks* y protocolos de comunicación

5.1.1. Backend: Kotlin y Spring Framework

Kotlin

Kotlin es un lenguaje de programación de propósito general que apareció en 2016 y se dio a conocer de manera general cuando Google lo estableció como lenguaje oficial para desarrollo Android en 2017 y, posteriormente, en 2019, lo definió como lenguaje preferido.

Durante estos años se ha ido expandiendo a otras plataformas y entornos y, en la actualidad, Kotlin se utiliza en gran medida en programación del lado del servidor (JetBrains, 2022). En 2023, un 9.7% de los desarrolladores profesionales utiliza Kotlin como lenguaje para sus proyectos (Stack Overflow, 2023).

El lenguaje ha podido avanzar tan rápidamente, entre otros motivos, gracias a su interoperabilidad con Java. Esto quiere decir que es posible utilizar bibliotecas y *frameworks* de Java en proyectos Kotlin, pudiendo también compartir y reutilizar código entre ambos lenguajes.

La sintaxis de Kotlin también es similar a la de Java, aunque es mucho más clara y concisa, lo que hace que el código sea más fácil de escribir y leer. Particularmente, tanto Java como Kotlin son lenguajes fuertemente tipados, pero este último dispone de una gestión de tipos que evita tener que especificarlos continuamente, reduciendo la verbosidad del código y aumentando la legibilidad.

Otro de los puntos clave de Kotlin, y que le da ventaja sobre Java, es lo que se denomina *null safety* o seguridad frente a nulidad. En Java, una variable o propiedad de cualquier tipo puede contener un valor nulo o `null` y esto no se puede saber hasta que el programa se ejecuta. En cambio, en Kotlin se especifica si ese tipo puede ser anulable o no, lo que permite realizar un desarrollo más robusto y resistente frente a nulos

Kotlin también integra, de manera nativa, uno de los últimos paradigmas del desarrollo software: la programación reactiva (Bonér et al., 2014). Este tipo de programación se basa en la respuesta a cambios en un entorno a través de eventos, construyendo *pipelines* de procesamiento asíncrono en entornos de código que parece secuencial.

El paradigma de la programación reactiva tiene aplicación en *frontend*, en respuesta a acciones del usuario o eventos del sistema, pero también en *backend*, donde permite crear aplicaciones con alto rendimiento y mayor escalabilidad que las convencionales. Por ejemplo, permite, para grandes flujos de elementos de datos, realizar una transmisión y un procesamiento progresivo según está disponible cada elemento, sin tener que esperar al conjunto completo.

Paralelamente, Kotlin ofrece, también de forma nativa, una gestión de hilos de ejecución más ligera y de mayor abstracción a través de lo que se conoce como corrutinas o *coroutines*. Las corrutinas son un concepto de programación asíncrona que permite ejecutar instrucciones de manera simultánea aunque en el código parece que se está haciendo de forma imperativa y secuencial.

Para este proyecto se ha escogido Kotlin como lenguaje para el *backend*, además

de por los motivos mencionados anteriormente, por su versatilidad, su gran comunidad y la amplia base de conocimientos disponible por todo Internet. También por la facilidad, mediante la combinación de la programación reactiva y las corrutinas, de crear un sistema rápido, no bloqueante y escalable sin implicar una larga curva de aprendizaje.

Organización por capas: proyecto Maven multimódulo

Maven es una de las herramientas más utilizadas para la gestión de proyectos Java. Un 73% de los desarrolladores Java declaran hacer uso de Maven en sus proyectos (JetBrains, 2022) por la facilidad que aporta una herramienta que automatice y conecte los procesos de gestión de dependencias, compilación y validación. También es compatible con Kotlin en la actualidad a través de un *plugin* (Kotlin Foundation, 2023).

Para este proyecto, los microservicios de *backend* se estructuran como proyectos Maven multimódulo, que permiten definir distintos módulos separados, permitiendo una mejor estructuración del código y un aislamiento mayor o menor, según se requiera, a través de dependencias internas.

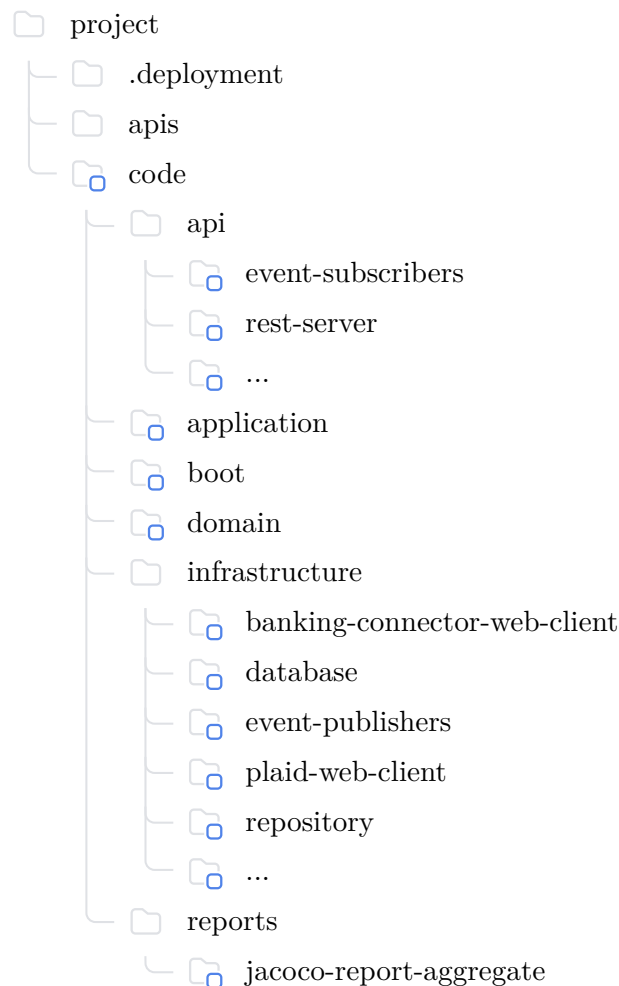


Figura 5.1: Diagrama de la estructura general con el gestor de proyectos Maven, a seguir en todos los microservicios Kotlin del proyecto

Con un cuadrado rojo, los directorios que a su vez representan módulos del proyecto Maven.

El uso de diferentes módulos permite, además, representar y aplicar las capas y puertos de la organización por capas a nivel de componente o microservicio software.

Para comunicar las distintas capas y puertos, se utilizan dependencias de Maven, interfaces y métodos de transformación de modelos, generalmente implementadas como funciones extensoras de Kotlin.

De manera complementaria al código, dentro de la carpeta `.deployment` se encuentran ficheros relativos al despliegue y a generación de entornos de pruebas mediante contenedores de Docker. En la carpeta de `apis` se encuentran los contratos o especificaciones para comunicarse con el microservicio.

El directorio `code` es la raíz del proyecto Maven. En el subdirectorio `api` se encuentran los módulos que implementan puertos o adaptadores primarios, es decir, de entrada (API REST o consumidores de eventos); y en el subdirectorio `infrastructure` se encuentran los módulos que implementan puertos o adaptadores secundarios, es decir, de salida (publicadores de eventos, persistencia o comunicación con otros servicios como el conector bancario o Plaid).

Para unificar el acceso a los datos persistidos tanto en base de datos como en otros microservicios, dentro de la capa de infraestructura se hace uso del patrón repositorio, con métodos para consultar y modificar datos, que se conectan con el resto de los módulos a través de interfaces definidas en la capa de dominio.

La capa de dominio, por su parte, se encuentra en el directorio `domain`, mientras que la capa de aplicación está contenida en el directorio `application`. El módulo `boot` contiene el arranque y las configuraciones de Spring Boot y los tests de integración automatizados que se lanzan al utilizar el *goal* de validación de Maven.

Por último, el módulo `jacoco-report-aggregate`, dentro del directorio `reports`, contiene un informe autogenerado sobre la cobertura de tests unitarios del código en todos los módulos.

Spring Framework y Spring Boot

Spring Framework es uno de los recursos más utilizados en la industria de desarrollo de software como base sólida para construir aplicaciones web escalables y robustas. Spring Boot, con estadísticas de uso de más del 67% en proyectos de Java (JetBrains, 2022), es una herramienta que acompaña a Spring Framework ofreciendo un arranque simple y rápido para los proyectos, incluyendo, por ejemplo, un servidor HTTP embebido.

El ecosistema de Spring es completamente compatible con Kotlin y con la programación reactiva, disponiendo de *frameworks* propios como Spring WebFlux que facilitan la implementación de flujos reactivos y no bloqueantes, desde la capa de REST hasta la persistencia en base de datos.

Para los microservicios de *backend* de este proyecto, Spring presenta una variedad de características y funcionalidades que aceleran el desarrollo de manera ordenada y fiable. Spring gestiona, entre otros, el arranque automático de servicios internos (cliente de base de datos, *binder* de AMQP, etc.) en base a configuraciones, el enrutamiento HTTP y la inyección de dependencias.

Además, dada la alta implantación del Spring en la comunidad de desarrollo y su definición como software de código abierto, este *framework* dispone de una enorme base de conocimientos y de preguntas respondidas por toda la red, lo que hace utilizarlo como base para desarrollar aplicaciones sea considerablemente más sencillo y rápido.

5.1.2. Comunicación síncrona en el backend: API REST a través de HTTP

Para comunicar los distintos microservicios de backend se hace uso de diferentes API REST, implementadas como controladores de Spring y definidas mediante contratos o especificaciones del estándar OpenAPI.

Esta tecnología, que es un estándar en la industria, permite comunicar dos partes de manera síncrona para consultar o modificar datos cuando se necesita una respuesta inmediata. Además, se trata de una tecnología sencilla y depurable gracias a las rutas legibles y a la serialización en formato JSON.

5.1.3. Comunicación asíncrona en el backend: mensajes a través de AMQP

Para cubrir la necesidad de una comunicación asíncrona e inicio de procedimientos cuya respuesta no se requiere, se hace uso del Advanced Message Queuing Protocol (AMQP), un protocolo estándar abierto orientado a mensajes y colas con enrutamiento, exactitud y seguridad.

Este protocolo permite que un microservicio deposite o publique mensajes que representan eventos en una cola, y que otro microservicio suscrito a dicha cola los consuma y los procese a medida que le sea posible. El principal caso en el que se hará uso de este tipo de comunicación es el envío de las actualizaciones de transacciones desde el agregador bancario al microservicio Core.

5.1.4. Frontend web: React

La tecnología escogida para implementar la aplicación web de Balancer es React, una biblioteca de JavaScript de código abierto para crear interfaces de usuario con una implantación del 55% en la comunidad de desarrollo de dicho lenguaje (JetBrains, 2022).

El desarrollo en React es eficiente, mantenible y extensible gracias a la aplicación del concepto de componentes. Los componentes son unidades de código reutilizables, que pueden crearse, actualizarse y probarse fácilmente de manera unitaria. Otra característica clave que hace destacar a React es su gestión del estado de los componentes para gestionar el comportamiento de la interfaz de usuario.

5.2. Metodología de desarrollo, pruebas y gestión de documentación

5.2.1. Control de versiones: Git y GitHub

Para almacenar el código y realizar un control de versiones ordenado a lo largo de las etapas del desarrollo, se han escogido el sistema Git y la herramienta GitHub.

Todo el código de los componentes desarrollados se encuentra disponible en la siguiente página de GitHub:

<https://github.com/balancer-project>

5.2.2. Desarrollo por tareas: sistema de tarjetas o *kanban*

En cuanto a la organización del proceso de desarrollo, se ha optado por la utilización de una metodología con base en *kanban*, a partir de la funcionalidad acotada en Subsección 3.3.3 y a través de tareas y subtareas. La elección de *kanban* sobre otras opciones como Scrum reside en la intención de aportar valor al proyecto de manera rápida, con un enfoque centrado más en el interés teórico del proyecto que en el del usuario final del sistema.

Se ha contado con el apoyo de la herramienta Trello como gestor de proyectos para planificar y definir técnicamente las tareas y subtareas, así como para realizar un seguimiento más completo del proceso realizado en cada momento.

5.2.3. Pruebas o *testing*

Durante el desarrollo del proyecto se han ido realizando pruebas manuales para validar el funcionamiento de los sistemas, generalmente a través de peticiones HTTP a través de Postman.

Así mismo, se han generado algunos tests de integración que validan el funcionamiento *end-to-end* de algunos microservicios, como el conector bancario, en el que distintas pruebas comprueban, de manera automática, que cuando se recibe una llamada al *webhook* del agregador bancario, se desencadenan los mecanismos y procesos pertinentes.

5.3. Desarrollo de los sistemas y aplicaciones

En esta sección se presentarán los detalles técnicos sobre la implementación de los diferentes servicios y aplicaciones del sistema.

Durante el proceso, la gestión del código se ha hecho a través de la capacidad de control de versiones de la plataforma GitHub, creando una organización para el proyecto en la siguiente dirección, donde se encuentran los repositorios de cada servicio:

<https://github.com/balancer-project/>

Para describir de manera más detallada esta implementación, conviene empezar por la fuente de datos bancarios y continuar en dirección al usuario, pasando por los distintos servicios definidos.

5.3.1. Microservicio conector bancario

Para implementar el microservicio conector bancario según lo definido en la Subsección 4.2.6, se crea un nuevo proyecto de Kotlin con Spring Framework. El código de este proyecto puede encontrarse en el siguiente repositorio de GitHub:

<https://github.com/balancer-project/balancer-banking-connector>

Integración con el agregador bancario

Tal y como se define en la Sección 2.2, el conector hace uso de los servicios del agregador bancario Plaid para obtener datos sobre transacciones bancarias. De los tres entornos que ofrece Plaid para la integración con sus servicios, para este proyecto se hace uso únicamente del entorno `sandbox`, ya que permite comprobar el funcionamiento y la correcta interacción de los componentes sin necesidad de manipular información financiera real, la cual tiene un alto nivel de sensibilidad y requeriría implementar mecanismos de seguridad adicionales, además de solicitar autorización al agregador.

Para integrar el conector bancario con Plaid se hace uso de una API REST a través de HTTPS, protegida por un mecanismo de autenticación básico a través de un `client_id` y un `secret`, que provee el agregador a través de su panel de control y que se envían en las cabeceras de las peticiones.

En la implementación del microservicio, se utiliza la especificación de OpenAPI de Plaid (Plaid, 2023e) para generar los modelos de las peticiones y los *endpoints*, y se inyecta un *bean* de cliente de la API de Plaid para realizar las llamadas de manera más fluida. Este cliente utiliza a su vez el cliente web reactivo incluido en el *framework* Spring WebFlux (Spring Framework, 2023).

Manejo de flujos de asociación de cuentas bancarias a través del agregador bancario

Con respecto a la funcionalidad que compete a este microservicio, de manera complementaria a la obtención y preprocesamiento de la información bancaria, el conector gestiona los flujos de asociación de cuentas bancarias por parte de los usuarios a través del agregador bancario.

Para este fin, Plaid, ofrece una herramienta llamada Link. Se trata de un *widget* visual que se carga en el lado del cliente y que permite al usuario identificarse en su entidad bancaria y autorizar la obtención de datos bancarios por parte del sistema a través de Plaid.

Durante el flujo, Link comunica al usuario qué información se va a ceder y con qué condiciones, permitiéndole seleccionar las cuentas que quiere compartir y dejando claras las partes implicadas: el usuario, el banco, Plaid y Balancer.

Una vez finalizado el flujo de asociación, se habrá creado lo que se denomina un Item de Plaid y se habrá obtenido token de acceso asociado a este. Un Item es un vínculo entre un usuario y un banco a través del cual se pueden obtener cuentas y transacciones. El token de acceso, compuesto por el tipo `access`, el entorno `sandbox` y un identificador en formato UUID, permitirá consultar los datos que arroja el Item, es decir, cuentas asociadas al usuario en el banco y las transacciones asociadas a estas.

De acuerdo con la documentación de Plaid (Plaid, 2023a), los tokens de acceso no caducan, pero es posible que necesiten actualizarse, por ejemplo cuando un usuario cambia su contraseña o, dentro del marco de la normativa PSD2, cuando se debe cumplir con la ventana de noventa días de consentimiento. El alcance de este proyecto no abarca la funcionalidad de renovación del consentimiento.

La Figura 5.2 muestra un diagrama de secuencia en el que se detalla el flujo mencionado anteriormente (Plaid, 2023f), que se inicia con una acción del usuario solicitando asociar una cuenta, pasa por la autorización en el banco a través de Link y finaliza con la confirmación de la asociación al usuario. Durante el proceso se guardan la información del Item y el token de acceso en la base de datos del conector bancario.

Para este flujo, se añaden dos rutas a la especificación de OpenAPI del microservicio. Una ruta corresponderá a crear un token para Link y otra será para arrancar el procesamiento del token público, obtener un token de acceso y asociarlo al usuario correspondiente. Ambas serán de tipo HTTP *POST*, pues modifican el estado del servidor y envían un cuerpo con datos.

Además, el servicio cuenta con una capa de persistencia para almacenar la asociación entre un usuario de Balancer y un Item de Plaid, junto con información complementaria como el token de acceso o la entidad financiera asociada, lo cual permite que no se creen Items duplicados para una misma entidad.

Comunicación desde el agregador bancario al conector

De nuevo en referencia a la funcionalidad que compete a este microservicio, en segundo lugar, aunque con mayor valor para el sistema, se encuentra la obtención de la información bancaria.

El conector bancario debe conocer, de manera continua, los cambios que se producen en el histórico de transacciones de las cuentas asociadas al usuario, incluyendo nuevos movimientos y modificaciones y eliminaciones de estos. Al conocer un cambio, debe preprocesarlo y, si es relevante, enviarlo al microservicio Core para que lo procese.

Para estar al tanto de los cambios en las transacciones, el microservicio podría tener configurada una tarea periódica o *job* que obtuviera listados de todas las cuentas vinculadas al sistema. Sin embargo, algunos agregadores bancarios, como es el caso de Plaid o

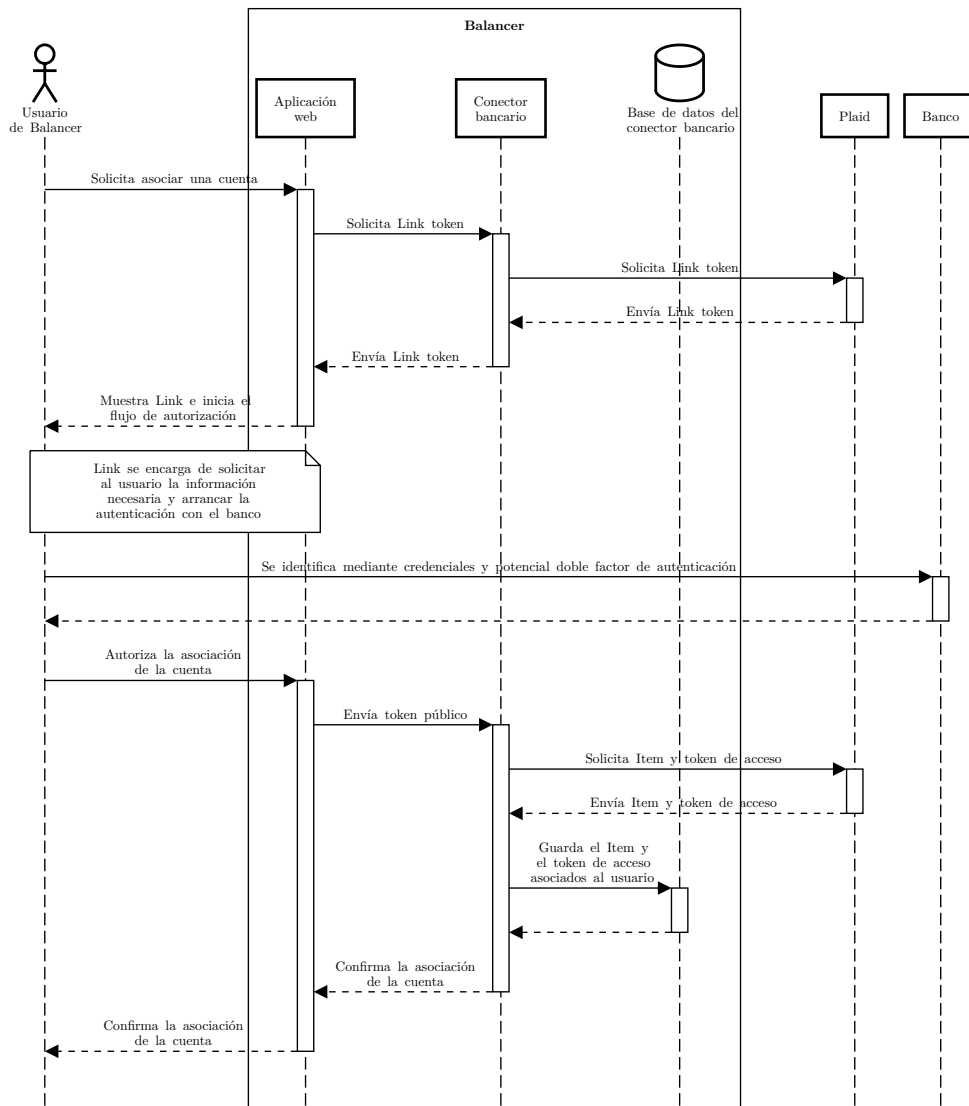


Figura 5.2: Diagrama de secuencia que muestra la creación de un Item de Plaid y la obtención de su token de acceso

Tink, proveen una manera de avisar directamente cuando hay nueva información disponible. Esta funcionalidad permitiría ahorrar sustancialmente en coste de procesamiento si la aplicación alcanzara un alto volumen de cuentas vinculadas.

En el caso de Plaid, el agregador permite configurar, al mismo tiempo que se crea un Item, un *webhook* para notificar actualizaciones sobre el mismo. Un *webhook* no es más que un *endpoint* de un servicio HTTP al que se puede llamar para comunicar que algún evento ha ocurrido. La diferencia entre un *endpoint* común y un *webhook* es que a los *webhooks* se les llama con una carga o *payload* muy reducida, mientras que a los primeros, por ejemplo, un *endpoint* REST, se les puede enviar cargas grandes como listados de entidades con diferentes campos.

Como se puede ver en la Figura 5.3, al recibir la llamada al *webhook*, se obtienen los nuevos datos, se preprocesan y se envían al microservicio Core.

Según la documentación de Plaid (Plaid, 2023d), los *webhooks* se deben implementar como *endpoints* de tipo HTTP *POST* que devuelvan una respuesta HTTP *OK* (código

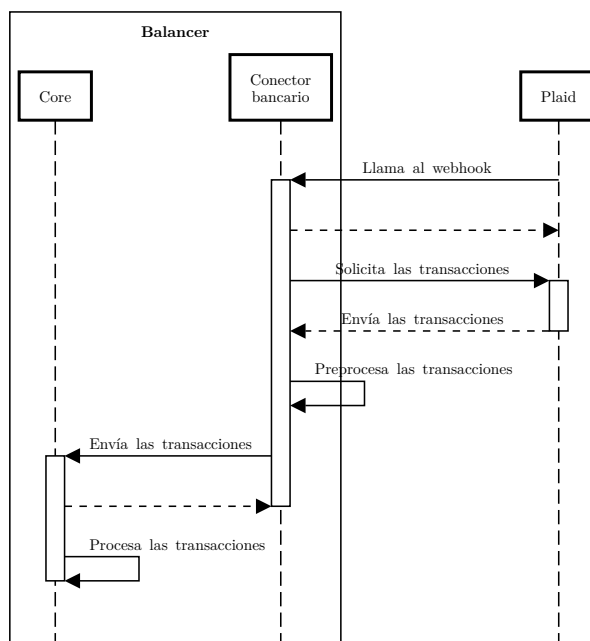


Figura 5.3: Diagrama de secuencia que muestra la recepción de un *webhook* por parte de Plaid, el preprocesamiento de los datos y el envío al microservicio Core

200). La manera adecuada de implementar estos *endpoints* sería crear un módulo adicional para recepción de *webhooks* y, por seguridad, separar su servicio HTTP del que se utiliza para los *endpoints* REST, lo cual podría conseguirse levantando un servidor HTTP adicional, configurando el existente para que escuchara en puertos diferenciados o instalando una *API gateway* o un *firewall*, es decir, un sistema centralizado que recibiera todas las peticiones y les permitiera o denegara el paso en función del origen, el método, la ruta concreta, etc.

En el contexto de este proyecto, por simplicidad, tanto los *endpoints* REST como los de *webhooks* utilizan el mismo servidor HTTP y la misma especificación de OpenAPI, por lo que se añade la ruta correspondiente al *webhook* dentro de dicha especificación.

Las llamadas a *webhooks* de Plaid (Plaid, 2023c) tienen un formato estándar con un cuerpo de datos o *payload* JSON que contiene un tipo, un código, el identificador del Item asociado y algunos metadatos más.

Procesamiento asíncrono de las actualizaciones de las transacciones bancarias

Todo el procesamiento dentro del *endpoint* del *webhook* se realiza de manera asíncrona. Es decir, cuando Plaid haga una llamada al *webhook*, en lugar de esperar hasta que se procesen las transacciones para entonces devolver una respuesta HTTP *OK*, el microservicio retornará de inmediato y lanzará un nuevo hilo de ejecución para obtener los datos y realizar el preprocesamiento. De esta manera se evita, por un lado, que Plaid se quede esperando que la información termine de procesarse y, por otro lado, que todos los hilos del conector queden bloqueados por diferentes llamadas a *webhooks* u otros *endpoints*.

Con este fin, se implementa un sistema interno de publicación y gestión de eventos que utilizará un hilo o *thread* de ejecución distinto para cada evento publicado. Cuando se reciba una llamada al *webhook*, se retornará directamente a Plaid un HTTP *OK* y se

publicará el evento interno correspondiente, que lanzará un manejador o *handler* desacoplado en otro hilo. Estos manejadores se nombran en el código como casos de uso o *use cases*, dentro de la capa de aplicación.

Cuando el manejador actúe, este se encargará, a su vez, de lanzar un evento por cada transacción detectada, desencadenando que el preprocesamiento de cada una se realice de manera simultánea y sin bloquear ningún otro hilo de ejecución.

Cuando termine el preprocesamiento de cada transacción se publicará un nuevo evento, esta vez externo a través del protocolo AMQP, que llegará al microservicio Core y desencadenará otras operaciones detalladas más adelante.

Sistema de eventos de aplicación en Spring

En cuando a la implementación del sistema interno de eventos, Spring dispone de un sistema propio de gestión de eventos de aplicación que se utiliza en el microservicio como base y API para la publicación de estos. Sin embargo, este sistema es síncrono, lo cual quiere decir que, por defecto, el publicador se queda bloqueado hasta que los receptores terminan de procesar el evento, y esto significaría que la llamada de Plaid quedaría bloqueada hasta que se preprocesaran todas las transacciones actualizadas.

Para poder trabajar de manera asíncrona y no bloqueante, desacoplando del publicador la gestión de los eventos, Spring permite añadir una configuración para ejecutar cada consumidor o *listener* en un hilo nuevo. Esta configuración permite aplicar el patrón de diseño software de *thread pool* o reserva de hilos, que consiste en utilizar un conjunto de hilos de ejecución disponibles para procesar una serie de tareas, en este caso eventos.

La reserva de hilos se puede configurar en función de las necesidades y la carga esperada para fijar un número de hilos de ejecución por defecto o una cantidad máxima de estos, que se pueden crear de forma dinámica a medida que aumenta el nivel de trabajo. En el caso del conector bancario se utiliza la configuración por defecto.

Inicialización del cursor y despacho de las actualizaciones

Como se menciona anteriormente, las llamadas a *webhooks* no contienen un alto volumen de información. En el caso de Plaid, esto quiere decir que las llamadas solo contienen códigos para identificar qué es lo que ha ocurrido y en relación a qué Item. Será el conector quien se encargue de solicitar al agregador la nueva información disponible.

De los distintos códigos de *webhook* que Plaid puede enviar en relación a transacciones (Plaid, 2023c), para el alcance de este proyecto solo interesa el código `SYNC_UPDATES_AVAILABLE`. Este *webhook* se llama por parte del agregador cuando las transacciones de un Item cambian.

Plaid dispone de un *endpoint* de sincronización de transacciones (Plaid, 2023b) que permite conocer de manera fluida y consistente los cambios que se van produciendo en el histórico de movimientos de las cuentas del usuario.

El uso de este *endpoint* se basa en un cursor de paginación que el microservicio debe guardar, asociado al Item, para saber *dónde se quedó* la última vez que *leyó* información sobre transacciones. Este cursor se envía al agregador y este devuelve, junto con la información de sincronización de transacciones, uno nuevo para la siguiente ocasión.

Este funcionamiento facilita el trabajo con notificaciones a través de *webhooks*, ya que permite que el conector solo tenga que consultar los cambios que se han producido desde la última vez que consultó la información, en lugar de descargar el histórico y ver qué transacciones se han añadido, cuáles se han modificado y cuáles se han eliminado.

Así, cuando el conector reciba una llamada al *webhook* con el código `SYNC_UPDATES_AVAILABLE`, utilizará el *endpoint* de sincronización de transacciones para obtener la información.

Cuando el usuario asocia la cuenta bancaria y se crea un Item, el conector debe avanzar el cursor hasta la última posición disponible. Con este fin, se sigue el siguiente proceso:

1. Se llama al *endpoint* de sincronización de transacciones una primera vez para activar el aviso de `SYNC_UPDATES_AVAILABLE`, lo cual es un requisito según especifica la documentación del *webhook* (Plaid, 2023c).
2. Se deben esperar dos llamadas al *webhook* de tipo `SYNC_UPDATES_AVAILABLE` que contendrán una actualización inicial con las transacciones de los últimos treinta días y una actualización histórica de todas las transacciones que Plaid haya podido obtener de la entidad bancaria.
3. Dichas actualizaciones no son de interés para el alcance del proyecto, por lo que el cursor se debe avanzar hasta que se conozca que ambas se han producido. Esta situación se puede conocer a través de la información contenida en la llamada al *webhook*.

Una vez se ha avanzado el cursor hasta la última posición disponible, pasadas las actualizaciones inicial e histórica, cualquier llamada al *webhook* de tipo `SYNC_UPDATES_AVAILABLE` será relevante para el proyecto. En estos casos, el sistema utilizará el *endpoint* de sincronización de transacciones con el cursor almacenado, obtendrá todas las actualizaciones disponibles de transacciones y lanzará su preprocesamiento.

El comportamiento desencadenado por la llamada al *webhook* se valida a través de una serie de tests de integración automatizados dentro del mismo código del microservicio, utilizando respuestas simuladas del *endpoint* de sincronización de transacciones y conectores o *binders* simulados para el envío de eventos a través de AMQP.

Preprocesamiento de la información bancaria

Tras recibir la llamada al *webhook* y obtener todas las actualizaciones disponibles a través del *endpoint* de sincronización, el conector inicia el preprocesamiento de cada una de las transacciones detectadas. Conforme se va completando cada proceso, se envía un mensaje al microservicio Core a través de AMQP con la información útil.

El preprocesamiento de eventos y transacciones está pensado para facilitar el trabajo al microservicio Core en la medida de lo posible, ignorando actualizaciones irrelevantes que generarían una carga inútil o componiendo las transacciones con metadatos de interés.

El *endpoint* de sincronización de Plaid facilita las actualizaciones en tres grupos de transacciones: añadidas, modificadas y retiradas o eliminadas. Para el caso de las

transacciones retiradas, basta con enviar al microservicio Core los identificadores correspondientes.

En los casos de las transacciones añadidas y modificadas, se descartan las transacciones entrantes (para el alcance del proyecto, solo se interpretan pagos salientes) se seleccionan los campos relevantes para el procesamiento y, además, se enriquece el evento con información útil como los identificadores del usuario y del banco asociados.

5.3.2. Microservicio Core

Para implementar el microservicio Core según lo definido en la Subsección 4.2.5, se crea un nuevo proyecto de Kotlin con Spring Framework. El código de este proyecto puede encontrarse en el siguiente repositorio de GitHub:

<https://github.com/balancer-project/balancer-core>

Gestión de gastos

Como responsable de la gestión de gastos, el microservicio Core aglutina la lógica de manejo y persistencia de estos, exponiendo *endpoints* de consulta y manipulación, y realizando accesos a su base de datos.

Recepción y procesamiento de actualizaciones sobre gastos

En cuanto a la parte del dominio relacionada con las transacciones bancarias, el microservicio Core recibe eventos emitidos por parte del conector bancario que contienen actualizaciones y transacciones preprocesadas.

Los eventos se reciben a través de distintas colas AMQP en RabbitMQ, a las que está conectado el microservicio, y se procesan comenzando por los consumidores en la capa de adaptadores primarios, pasando por comandos y casos de uso en la capa de aplicación, y llegando al servicio de asociación transferencia-gasto y al servicio de aplicación de cambios en gastos a partir de transferencias.

Esta lógica es el el segundo punto de interés dentro del alcance de este proyecto, después de la conexión *física* con el agregador bancario, y su implementación se basa directamente en lo definido en la Subsección 4.2.6. Esta implementación, además de los propios servicios de asociación y aplicación de cambios, hace uso del resto de componentes del microservicio, como pueden ser repositorios de gastos y pagos, entidades de dominio, objetos de transferencia de datos (DTO, por sus siglas en inglés) y métodos de transformación.

5.3.3. Aplicación web

Para implementar la aplicación según lo definido en la Subsección 4.2.7, se crea un nuevo proyecto de Node.js con React. El código de este proyecto puede encontrarse en el siguiente repositorio de GitHub:

<https://github.com/balancer-project/balancer-web-app>

Se hace uso de React Router para gestionar las rutas y vistas de la aplicación. Para el alcance del proyecto se las siguientes vistas:

- Inicio: es la página de *landing* de la aplicación, muestra un saludo para el usuario.
- Mis gastos: muestra una lista de los gastos que el usuario tiene registrados.
- Nuevo gasto puntual: permite añadir un gasto puntual a través de un formulario.
- Nuevo gasto recurrente: permite añadir un gasto recurrente a través de un formulario.
- Mis cuentas asociadas: muestra una lista de las cuentas bancarias que el usuario ha asociado con el sistema.
- Asociar cuenta bancaria: permite asociar una cuenta bancaria a través de Plaid Link.
- No encontrado.

Para la comunicación con el backend, se definen dos *wrappers* para las API de gastos y enlace de cuentas bancarias. Estas API, a su vez, hacen uso del cliente HTTP Axios, que realiza las peticiones a más bajo nivel y devuelve las respuestas, que se transforman a los objetos y modelos de dominio a través de *mappers*.

El cliente Axios se incluye como dependencia mediante el uso de Node Package Manager (NPM). De esta manera se hace uso también de otros paquetes como React Bootstrap, React Icons o la implementación de Plaid Link para React, a través de la cual se produce la vinculación y autorización de la cuenta bancaria por parte del usuario, sin necesidad de implementar toda su API en el frontend y proporcionando solo las credenciales necesarias.

Vista de gastos

En la vista de gastos se muestra un listado de los gastos registrados por el usuario, así como un botón para actualizar la información y otro para añadir un gasto.

De manera automática al cargar la página o manualmente haciendo clic en el botón correspondiente, la aplicación web envía una petición al microservicio Core solicitando un listado de los gastos asociados al usuario. Al recibir una respuesta, aplica las transformaciones oportunas y muestra la información actualizada.

Al hacer clic en el botón para añadir un gasto se muestra un formulario para llevar a cabo dicha acción. Al enviar el formulario, la aplicación envía una nueva petición al microservicio Core con los datos introducidos por el usuario.

Las listas, los elementos y los formularios se han definido como componentes de React independientes que pueden ser reutilizados, modificados o sustituidos según sea necesario.

Para dinamizar la actualización de la lista entre las posibilidades de uso de la tecnología WebSocket o la técnica conocida como *polling*, se ha elegido esta última por el nulo acoplamiento que requiere con el backend, al no necesitar mantener una conexión activa. Con esta técnica, cada segundo se envía una petición al servidor para obtener el nuevo listado. En el *frontend* se produce una comparación y, si hay cambios, se generan una alerta y un indicador para el gasto actualizado.

Esta técnica es viable en esta prueba de concepto por la baja carga prevista. Si se

previera una carga moderada o más grande, la técnica de *polling* no sería una opción recomendable por la posible saturación del servidor, y en este caso sería conveniente utilizar la tecnología de WebSocket.

Vistas de nuevo gasto puntual y nuevo gasto recurrente

En estas dos vistas se muestran sendos formularios con los campos necesarios para registrar un gasto personal y un gasto recurrente, tal y como se especifica en el modelo en la Sección 4.3.

Para los formularios se hace uso del paquete React Hook Form, que facilita el registro de cada campo y la validación, mostrando un error al enviar el formulario si existe algún fallo de entrada.

Vista de para asociar una cuenta bancaria

La vista para asociar una cuenta bancaria es de gran simplicidad. Su propósito principal es informar al usuario del procedimiento y las implicaciones para asociar una cuenta. Al cargarse la página, en segundo plano se envía la solicitud correspondiente al *backend* para obtener un token de Link, como se especifica en la Sección 5.3.1.

Una vez el usuario ha comprendido los términos, puede hacer clic en un botón que mostrará la interfaz de Plaid Link, y entonces comenzará el flujo de asociación. Cuando el usuario finalice el proceso y se oculte la interfaz de Plaid Link, la aplicación web habrá recibido un token público que se intercambiará, a continuación, por uno de acceso, según se especifica en la Sección 5.3.1.

Vista de cuentas bancarias conectadas

La vista de cuentas bancarias conectadas facilita al usuario un listado de las cuentas que ha asociado previamente, incluyendo la entidad bancaria, el número de cuenta y algunos datos más.

5.4. Resultados de la implementación

El propósito de esta sección es detallar el sistema obtenido finalmente tras su construcción en la fase de desarrollo, a fin de poder utilizar los detalles como referencia en la sección de conclusiones. Para cada uno de los tres servicios o aplicaciones construidos, se especifica la funcionalidad técnica y los flujos que cubren finalmente en relación a las historias de usuario y el alcance definidos en las subsecciones 3.3.2 y 3.3.3.

5.4.1. Microservicio Core

El microservicio Core se ha constituido como un servicio capaz de recibir peticiones del usuario a través de la aplicación web. Las peticiones gestionadas cubren la siguiente funcionalidad técnica:

- Registrar un gasto puntual. Relacionado con la historia de usuario HU1.
- Registrar un gasto recurrente. Relacionado con la historia de usuario HU1.
- Consultar todos los gastos registrados. Relacionado con la historia de usuario HU2.
- Procesar eventos de actualizaciones sobre transacciones enviados desde el conector bancario. Relacionado con la historia de usuario HU5.

En este punto es necesario destacar que la cobertura del procesamiento entre transacciones y gastos no es completa, sino que se ha implementado parcialmente a modo de prueba de concepto. El flujo de procesamiento implementado es el que se detalla en la Subsección 4.4.1 para la asociación entre una transacción bancaria y un gasto registrado, y en la Subsección 4.4.2 para la aplicación de cambios a un gasto a partir de una transacción. Esta cobertura se discute más adelante en las secciones de conclusiones (6.2) y de trabajo futuro (6.3).

5.4.2. Microservicio conector bancario

El resultado obtenido del microservicio conector bancario es un servicio que recibe peticiones del usuario a través de la aplicación web y, adicionalmente, del agregador bancario. Se cubre la siguiente funcionalidad técnica:

- Asociar una cuenta bancaria. Relacionado con la historia de usuario HU3.
- Consultar todas las cuentas bancarias asociadas. Relacionado con la historia de usuario HU4.
- Recibir y preprocesar eventos de actualizaciones sobre transacciones enviados desde el agregador bancario, y enviar nuevos eventos al microservicio Core. Relacionado con la historia de usuario HU5.

5.4.3. Aplicación web

El desarrollo de la aplicación web ha concluido con la construcción de una interfaz dedicada a la interacción directa con el usuario para transmitir sus peticiones a los sistemas

que las procesarán automáticamente en el *backend*.

Este componente aglutina toda la funcionalidad técnica del resto de microservicios desde el punto de vista de presentación al usuario:

- Registrar un gasto puntual. Relacionado con la historia de usuario HU1.
- Registrar un gasto recurrente. Relacionado con la historia de usuario HU1.
- Consultar todos los gastos registrados. Relacionado con la historia de usuario HU2.
- Asociar una cuenta bancaria. Relacionado con la historia de usuario HU3.
- Consultar todas las cuentas bancarias asociadas. Relacionado con la historia de usuario HU4.
- Recibir actualizaciones sobre los gastos registrados para mostrarlas al usuario. Relacionado con la historia de usuario HU5.

Además, dada la relevancia en la aportación de valor del desarrollo al usuario, se ha estimado oportuno incluir como resultados algunas capturas de pantalla de los flujos disponibles en la aplicación web. Estas capturas, colocadas a continuación y en las siguientes páginas, van anotadas con la funcionalidad con la que se relacionan a partir del listado anterior.

En relación a la funcionalidad sobre recibir actualizaciones acerca de los gastos (HU5), en las capturas figuran los indicadores que se activan cuando un gasto ha recibido una actualización, pero es necesario dejar constancia de que también se ha implementado un sistema de notificaciones mediante alertas del navegador web, que se activa al mismo tiempo que el indicador, pero que no se ha podido mostrar en las capturas de pantalla.

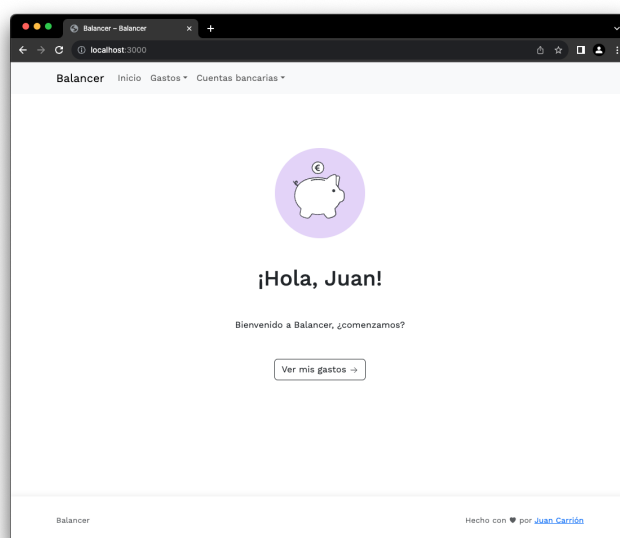
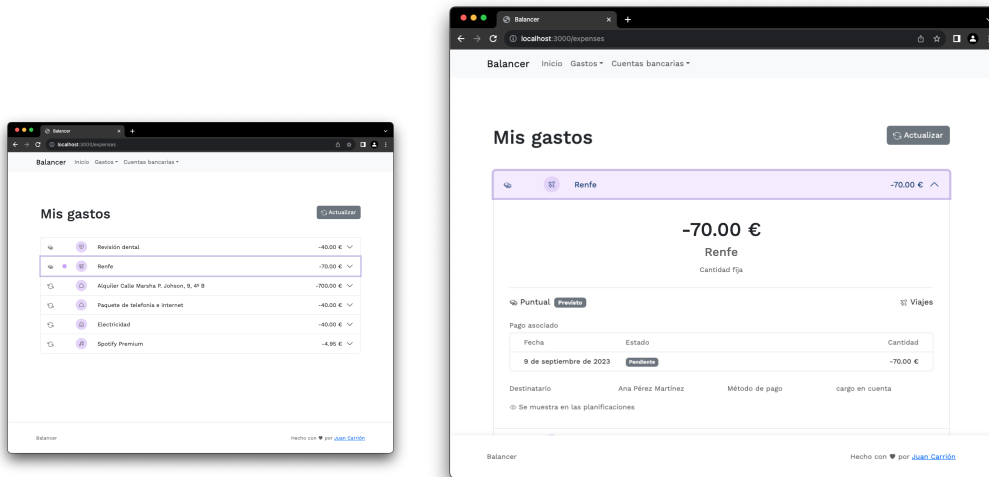
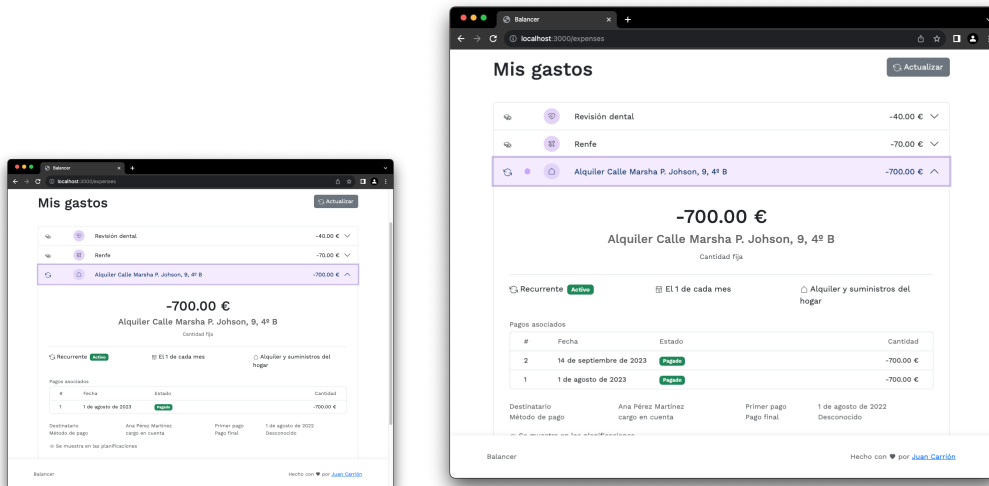


Figura 5.4: Vista de inicio. Esta vista tiene relación con la funcionalidad de consultar todos los gastos registrados (HU2), siendo un punto de entrada a la aplicación con un botón para navegar a la vista del listado.



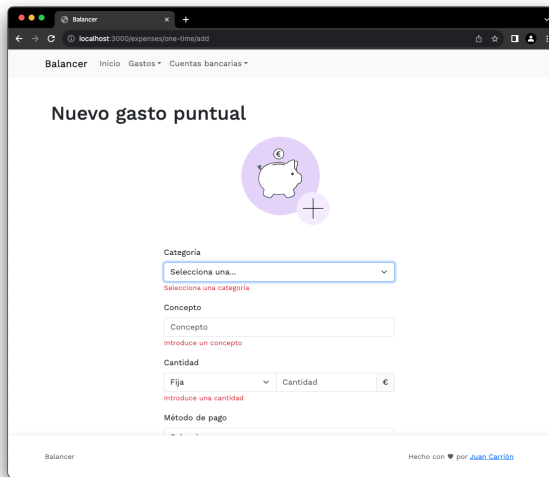
(a) Vista de gastos con un gasto resaltado tras recibir una actualización
 (b) Vista de gastos con un gasto puntual desplegado

Figura 5.5: Vista de gastos con detalles de gastos puntuales. Tiene relación con la funcionalidad de consultar todos los gastos registrados (HU2). Además, la Figura 5.5a muestra un indicador de estado actualizado, relativo a la HU5.

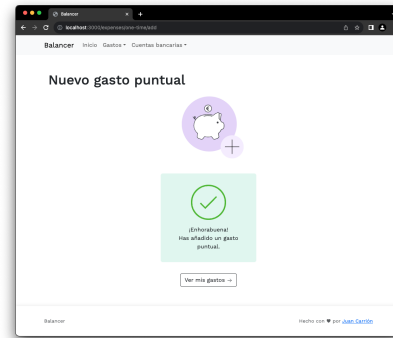


(a) Vista de gastos con un recurrente desplegado
 (b) Vista de gastos con un recurrente desplegado tras recibir una actualización

Figura 5.6: Vista de gastos con detalles de gastos recurrentes. Tiene relación con la funcionalidad de consultar todos los gastos registrados (HU2). Además, la Figura 5.6b muestra un indicador de estado actualizado, relativo a la HU5.

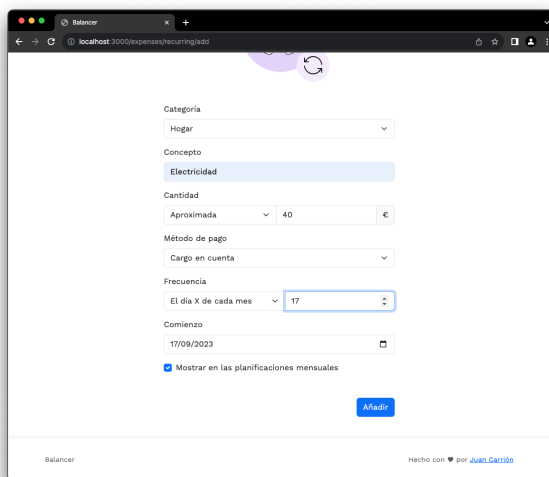


(a) Vista para añadir un gasto puntual mostrando errores en los campos

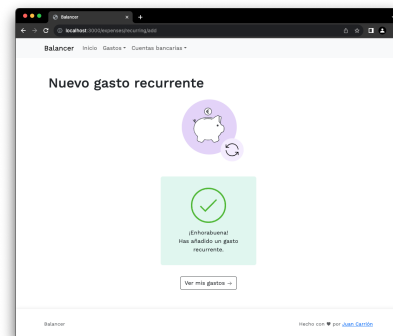


(b) Vista para añadir un gasto puntual mostrando un resultado satisfactorio

Figura 5.7: Vista para añadir un gasto puntual. Tiene relación con la funcionalidad de añadir un gasto puntual (HU1).

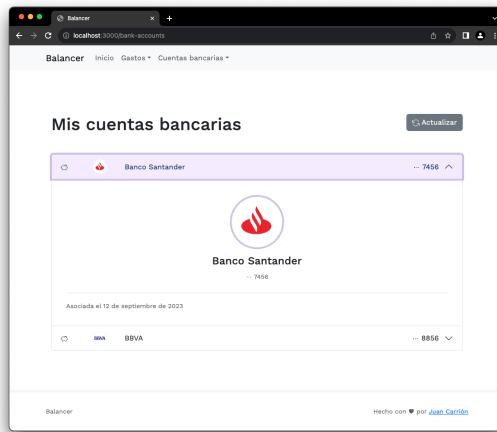


(a) Vista para añadir un gasto recurrente mostrando los campos rellenos

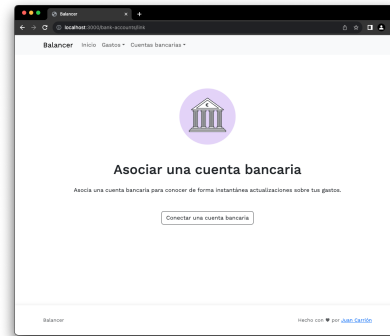


(b) Vista para añadir un gasto recurrente mostrando un resultado satisfactorio

Figura 5.8: Vista para añadir un gasto recurrente. Tiene relación con la funcionalidad de añadir un gasto recurrente (HU2).

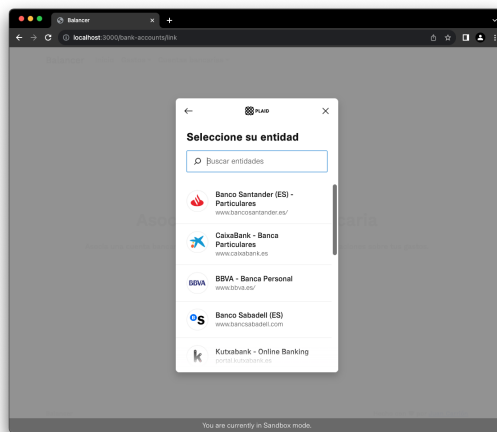


(a) Vista de cuentas bancarias asociadas con una cuenta desplegada

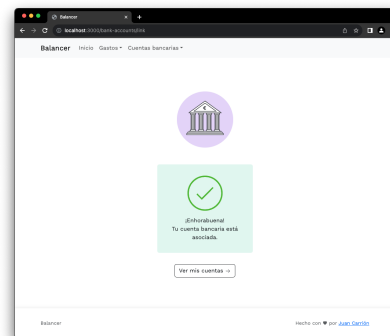


(b) Vista para asociar una cuenta bancaria antes de iniciar el proceso de Link

Figura 5.9: Vistas de cuentas bancarias y para asociar una cuenta bancaria. Tienen relación con la funcionalidad de listar todas las cuentas bancarias asociadas (HU4) y la de asociar una nueva (HU3).



(a) Vista del *widget* de Plaid Link abierto tras hacer clic en el botón correspondiente. Nota: este diseño y desarrollo corresponde a Plaid y no forma parte de este trabajo; se muestra en esta sección como referencia para demostrar la integración con el flujo de enlace del agregador



(b) Vista para asociar una cuenta para asociar una cuenta bancaria mostrando un resultado satisfactorio

Figura 5.10: Vista para asociar una cuenta mostrando Plaid Link y un resultado satisfactorio. Tiene relación con la funcionalidad de asociar cuenta bancaria (HU3).

Capítulo 6

Conclusiones y trabajo futuro

6.1. Planteamiento de las conclusiones sobre objetivos y resultados

La finalidad de este trabajo, como se plantea en la introducción (Capítulo 1), consistía en el desarrollo de un sistema capaz de registrar gastos personales y permitir realizar gestiones automáticas sobre estos con la finalidad de liberar a los usuarios de la importante dedicación temporal que suponía realizar dichas gestiones de forma manual.

En este capítulo final se describen las conclusiones derivadas de los resultados obtenidos al finalizar el trabajo, recogidos en la sección anterior, en comparación con los objetivos definidos en el planteamiento inicial del proyecto.

6.2. Conclusiones

Tras las fases de análisis, una vez obtenido el conocimiento necesario sobre el estado de los agregadores financieros y las necesidades de los usuarios a través de las entrevistas, y después de diseñar y definir la arquitectura del sistema, es posible concluir que el alcance técnico y funcional se ha cumplido, y que la solución software construida se acerca a los objetivos fijados.

Para realizar una comparación con dichos objetivos, se plantea primero un análisis del cumplimiento de los objetivos específicos y, posteriormente, una generalización para el objetivo principal.

Evaluación del cumplimiento del objetivo específico C1: desarrollar un sistema capaz de registrar los gastos del usuario, con sus características determinantes (parámetros temporales, categoría, importe, etc.), y ofrecer listados, informes, previsiones y gráficos en base a estos.

Los resultados mostrados en la Sección 5.4 demuestran que el sistema resultante cubre distintos flujos relacionados con la creación (figuras 5.7a, 5.7b, 5.8a y 5.8b) y el listado (figuras 5.5a, 5.5b, 5.6a y 5.6b) de gastos.

El objetivo específico se cumple parcialmente desde el punto de vista del aporte y la visualización de la información. El usuario podrá registrar sus datos y obtener un listado de los mismos, pudiendo gestionarlos a través de una interfaz visual intuitiva y dinámica.

Sin embargo, no se han incluido en el alcance del trabajo ni en el desarrollo algunos puntos detectados en el análisis funcional que incrementarían el nivel de cumplimiento del objetivo específico, como pueden ser la edición manual de gastos ya registrados, la visualización de planificaciones mensuales o de otros periodos temporales, o la generación de informes de gastos.

Evaluación del cumplimiento del objetivo específico C2: desarrollar un sistema capaz de registrar gastos de tipo tanto puntual como recurrente, pudiendo al mismo tiempo mantener un estado para cada tipo (pendiente y pagado para el puntual; y activo y finalizado para el recurrente).

En la misma Sección 5.4 se puede verificar que el sistema construido dispone de una separación funcional entre el modelo de gasto puntual (figuras 5.5b, 5.7a y 5.7b) y el

de gasto recurrente (figuras 5.6a, 5.6b, 5.8a y 5.8b), cada uno con sus estados asociados, cumpliendo así en gran nivel objetivo fijado.

Podría argumentarse que el objetivo no se cumple de manera completa por la no implementación de aspectos como los periodos de recurrencia para los gastos de tipo recurrente.

Evaluación del cumplimiento del objetivo específico C3: desarrollar un sistema capaz de registrar los pagos asociados a cada gasto del usuario y actualizar el estado de dicho gasto en base a cambios en los pagos.

El sistema se ha implementado tal y como se define en el modelo de datos (4.3), asociando un único pago a los gastos de tipo puntual (figura 5.5b) y un número arbitrario de pagos a los gastos de tipo recurrente (figuras 5.6a y 5.6b).

Por otro lado, aunque no se puede observar en las capturas de pantalla incluidas en los resultados, se ha implementado la aplicación de cambios en el estado de los gastos cuando se añade o modifica un pago. El detalle de esta implementación está descrito en la Subsección 4.4.2.

Evaluación del cumplimiento del objetivo específico C4: desarrollar un sistema dotado de la capacidad automática para integrarse y recibir información bancaria a partir de una fuente de datos, procesarla y actualizar el estado de los pagos y los gastos en función de dicha información.

En relación a este último objetivo específico, se observa también en los resultados detallados en la Sección 5.4 que se ha llevado a cabo cierta implementación relacionada con el objetivo.

En primer lugar, como se puede ver en las figuras 5.9b, 5.10a y 5.10b, se ha realizado la integración con una fuente de información bancaria, concretamente un agregador bancario, y el sistema es capaz de asociar una cuenta bancaria de cuyas transacciones recibir notificaciones.

En segundo lugar, como se puede comprobar en la figura 5.9a, el usuario puede comprobar, tras asociar diferentes cuentas bancarias, un listado con las mismas y los detalles de cada una.

En tercer lugar, se puede observar que la integración con la fuente de información también se ha producido a nivel de recibir dicha información de manera automática. Esta observación se puede realizar, en cierto grado, en las figuras 5.5a y 5.6b, en las cuales se muestra un indicador (un punto rosado) junto a cada gasto indicando que se ha producido una actualización basada en una transacción bancaria. En mayor medida, esta integración se demuestra en las secciones 5.3.1, 5.3.1 y 5.3.1, en las cuales se detallan los flujos técnicos implementados.

En cuarto y último lugar, para terminar de validar la consecución de este objetivo específico, es pertinente referirse a las subsecciones 4.4.1 y 4.4.2, que detallan los algoritmos a seguir para el procesamiento de las transacciones y la actualización de pagos y gastos en base a estas, y que detallan de manera exacta cómo se ha llevado a cabo la implementación final.

De esta manera se concluye que el objetivo se ha cumplido de forma íntegra, disponiendo el sistema resultante de las capacidades de integración con la fuente, recepción y procesamiento de la información bancaria y actualización de pagos y gastos en base a esta.

Conviene destacar, no obstante, que existen ciertos aspectos que se podrían echar en falta para cubrir de manera excepcional el objetivo.

Por un lado, sobre las fuentes de datos bancarios, se han dejado fuera del alcance el uso de las notificaciones por correo electrónico y la adición manual por parte del usuario. Si bien la integración con el agregador bancario es de gran valor tanto para el objetivo como técnico, estas dos alternativas también se identificaban como valoradas por el público objetivo de este trabajo.

En segundo lugar, con respecto al procesamiento de las actualizaciones de transacciones procedentes del agregador bancario y tal y como se especifica en los resultados, este procesamiento se ha implementado solo de manera parcial, a modo de prueba de concepto, y sin cubrir todas las posibilidades ofrecidas por los datos y las propiedades de los modelos. Si bien es cierto que esta prueba de concepto aporta un gran valor por exponer la manera en la que se debería procesar la información, aún quedaría valor por aportar al usuario a través del procesamiento de las entidades completas.

Evaluación del cumplimiento del objetivo principal: liberar a la persona de la burocracia, el tiempo y el esfuerzo dedicados a gestionar gastos personales puntuales y recurrentes para poder dedicar ese tiempo a disfrutar de los productos y servicios asociados a dichos gastos.

Contando con que los objetivos específicos evaluados anteriormente parten en su definición de este objetivo principal, a través de una comparación conjunta y de los aspectos en común se puede analizar el nivel de cumplimiento de dicho objetivo principal.

El nivel en que se consigue "liberar a la persona de la burocracia, el tiempo y el esfuerzo" se consigue, a través de los objetivos específicos, en la medida en que se alcanza la automatización de algunos procesos que el usuario realizaba de forma manual.

El registro de gastos en el sistema a través de la aplicación web, si bien requiere algo de dedicación, habilita la posibilidad de realizar procesamientos automáticos sobre estos gastos. Estos procesamientos, cubiertos en parte, sí producen un alivio en la dedicación del usuario.

Por ejemplo, a nivel de comodidad y experiencia, el usuario dispone ahora de una vista organizada y elaborada para obtener un vistazo rápido de sus gastos. Puede conocer, con una pasada, los gastos puntuales que ha tenido y los gastos recurrentes a los que hace frente.

El usuario puede ahora también recibir notificaciones sobre actualizaciones en sus gastos, de manera automática cuando se produce una transacción en su banco. Esta capacidad libera al usuario de tener que estar comprobando si se ha producido algún pago o cobro en la aplicación de su banco, y en el caso de que esa misma aplicación lo avisara sobre la actualización, el sistema diseñado aún sería de utilidad, pues liberaría al usuario de modificar manualmente el gasto asociado.

A modo de conclusión general, el trabajo ha servido para construir un sistema software funcional y técnicamente completo, capaz de cubrir el alcance definido y alcanzando, de esta manera, una gran parte de los objetivos fijados. Más adelante, en la sección de trabajo futuro (6.3), se plantean algunos puntos para un mayor alcance de los objetivos propuestos y una ampliación de los mismos.

A nivel técnico y en cuanto a la parte de los objetivos relativa a la creación de un sistema informático, se ha cumplido el propósito, obteniendo un sistema compuesto por diferentes servicios y aplicaciones que, en conjunto y trabajando de manera colaborativa, permiten llevar el valor al usuario y cumplir con los objetivos principales del trabajo.

Se trata de un sistema finalmente compuesto por dos microservicios y una aplicación web, pero construido de tal manera que es fácilmente extensible para añadir nuevos servicios y nuevas interfaces, tanto de interacción con el usuario como para comunicarse con otros servicios externos.

6.3. Trabajo futuro

Una vez realizado el análisis y la discusión de los resultados con respecto a los objetivos, surgen distintas áreas en las que el trabajo podría ampliarse y producir una evolución del sistema en el futuro.

Por un lado, con los mismos objetivos fijados para este proyecto, cabría la posibilidad de ampliar el alcance funcional y así conseguir una mayor cobertura de dichos objetivos. Este nuevo alcance podría incluir aspectos de funcionalidad como la visualización de planificaciones mensuales o la generación de informes de gastos, además de mayores opciones de operación sobre los gastos o un mayor grado de personalización.

A nivel de interacción con el usuario, en esa misma ampliación del alcance podría incluirse la construcción de una aplicación móvil, siendo los dispositivos móviles los que más se utilizan durante el día a día y los que el usuario tiene a disposición en todo momento para recibir notificaciones. En cuanto a notificaciones, también podría implementarse una comunicación directa desde el servidor hasta la aplicación web, añadiendo robustez a dicho sistema frente a la implementación resultante actual.

En relación a las fuentes de información bancaria y como ya se ha mencionado antes, podrían añadirse en el futuro el uso de las notificaciones por correo electrónico o la introducción manual de datos a partir de ficheros informáticos ofrecidos por los bancos.

También se podría completar en mayor manera el trabajo puliendo la lógica de procesamiento entre transacciones y gastos, ya sea añadiendo reglas al algoritmo implementado o utilizando sistemas de inteligencia artificial para la asociación y la aplicación de cambios de manera automática.

En lo que respecta a la metodología, especialmente teniendo en cuenta la utilizada para la captación de requisitos y el análisis previo al diseño, con encuestas a un grupo de personas, sería conveniente añadir en un futuro la validación del cumplimiento de los requisitos extraídos. Esta validación consistiría en contar de nuevo con las personas entrevistadas y plantearles el sistema final o un prototipo, obteniendo sus comentarios y opiniones y analizando en qué medida se cumplen sus expectativas.

Es importante destacar, además, en vistas al futuro, que la Unión Europea ya trabaja en una renovación de la normativa relacionada con la información y las operaciones bancarias en el ámbito del *open banking* y de la ya denominada como PSD3. Esta renovación podría traer nuevas posibilidades y cubrir en mayor medida los objetivos fijados u otros adicionales dentro de la temática del proyecto.

Por último queda destacar, a nivel técnico, que podrían aplicarse distintas mejoras que se han *obviado* con el propósito de centrar el desarrollo en el aporte de valor al usuario. Algunos ejemplos de esas mejoras son la seguridad, incluyendo una gestión de usuarios y permisos, y la implementación de una API *gateway* o un *proxy* entre la aplicación web y los microservicios de *backend*; y un mayor aprovechamiento de las capacidades del lenguaje y el *framework* utilizados, como podría ser la programación reactiva.

Chapter 6

Conclusions and future work

6.1. Approach to conclusions over objectives and results

The aim of this work, as defined in the introduction (chapter 1) chapter, consists in the development of a system capable of registering personal expenses and being able to perform automatic management operations over those expenses, with the goal of freeing the users of the important amount of temporal dedication that comes of performing these operations manually.

In this final chapter we present the conclusions derived from the results obtained after finishing the work, captured in the previous section, in comparison with the objectives defined in this project's initial approach.

6.2. Conclusions

After the analysis phases, once the necessary knowledge about the state of financial aggregators has been obtained and the necessities of the users have been captured through the interviews, and after designing and defining the system's architecture, it is now possible to conclude that the technical and functional scope has been reached, and that the resulting software solution is near to the defined objectives.

In order to make a comparison with those objectives, a first analysis is set on how the specific objectives are met, and then a generalization for the main objective takes place.

Evaluation of the achievement of the specific objective C4: to develop a system with the ability to integrate and automatically receive bank information from a data source, process it and update the status of payments and expenses based on this information.

The results shown in section 5.4 proves that the resulting system covers different flows related to expense creation (figures 5.7a, 5.7b, 5.8a y 5.8b) and listing (figures 5.5a, 5.5b, 5.6a y 5.6b).

The specific objective is achieved partially from the point of view of the submission and visualization of the information. The user will be able to register their expenses' data and obtain a list of them, being able to manage them through an intuitive and dynamic visual interface.

However, there are some items that were detected during the functional analysis, that would increase the amount of achievement of the specific object, but have not been included in the work scope, such as the manual editing of already-registered expenses, the visualization of monthly plans, or the generation of expense reports.

Evaluation of the achievement of the specific objective C2: to develop a system capable of registering expenses of both one-time and recurring types, while being able to maintain a status for each type (pending and paid for the one-off; and active and finished for the recurring).

In that same section 5.4, we can verify that the resulting system has a functional separation between the one-time expense model (figures 5.5b, 5.7a and 5.7b) and the recurring expense model (figures 5.6a, 5.6b, 5.8a and 5.8b), each with its associated statuses, which means that the objective is met.

It could be argued that the objective is not largely or completely met due to the non-implementation of aspects such as recurrence periods for recurring expenses.

Evaluation of the achievement of the specific objective C3: to develop a system capable of registering the payments associated with each user expense and updating the status of that expense based on changes in its payments.

The system has been implemented exactly as defined in the data model (4.3), associating a single payment to one-time expenses (figure 5.5b) and a variable number of payments to recurring expenses (figures 5.6a and 5.6b).

Even though it is not observable in the included screenshots, the system implements the application of changes in the status of expenses based on the addition or modification of a payment. A detail of this implementation is described in subsection 4.4.2.

Evaluation of the achievement of the specific objective C1: to develop a system capable of registering the user's expenses, with their specific properties (temporal parameters, category, amount, etc.), and to generate lists, reports, plans and graphs based on these.

In relation to this last specific objective, it is also observed in the results detailed in 5.4 that certain implementation related to the objective has been done.

First, as can be seen in figures 5.9b, 5.10a and 5.10b, integration with a banking information source, specifically a banking aggregator, has been carried out and the system is able to associate a bank account in order to be able to receive notifications about updates on transactions related to that account.

Second, as can be seen in figure 5.9a, the user can check, after associating different bank accounts, a list of them with the details of each one.

Third, we can observe that the integration with the information source has also been carried out at the level of it being received automatically. This observation can be made in figures 5.5a and 5.6b, which show an indicator (a pink dot) next to each expense indicating that an update has occurred based on a bank transaction. To a greater extent, this integration is demonstrated in sections 5.3.1, 5.3.1 and 5.3.1, which detail the implemented technical flows.

Fourth and finally, to finish validating the achievement of this specific objective, it is important to refer to subsections 4.4.1 and 4.4.2, which detail the algorithms for processing transactions and updating payments and expenses based on them, and which detail exactly how the final implementation has been carried out.

Reviewing these last statements, it can be concluded that the objective has been fully met, with the resulting system having the capabilities of integrating with the data source, receiving and processing banking information, and updating payments and expenses based on it.

However, it is worth noting that there are certain aspects that could be missed to cover the objective exceptionally.

On one hand, on banking data sources, the use of email notifications and manual addition through spreadsheets have been left out of scope. While integration with the banking aggregator has great value both for the objective and technically, these two alternatives were also identified as being valued by the target audience of this work.

On the other hand, with respect to the processing of updates to transactions from the banking aggregator and as specified in the results, this processing has only been implemented partially, as a proof of concept, without covering all the possibilities offered by the data and the properties of the models. While it is true that this proof of concept provides great value by showing the way in which the information should be processed, there would still be value to be added to the user through the processing of the complete entities.

Evaluation of the achievement of the main objective: to free the person from the bureaucracy, time and effort required to manage one-time and recurring personal expenses, so that they can dedicate that time to actually enjoying the goods and services associated with those expenses.

Taking into account that the specific objectives evaluated earlier are based on the definition of this unique main objective, through a joint comparison and the common aspects, it is possible to analyze the level of achievement of said main objective.

The level in which "to free the person from the bureaucracy, time and effort" is achieved, through the specific objectives, is represented by the achieved automation of some of the processes that the user would normally perform manually.

The registration of expenses in the system through the web application, although requiring some time and dedication, enables the possibility of performing automatic processing on these expenses. This processing does produce a relief in the time expending of the user.

For example, at the level of comfort and user experience, the user now has an organized and elaborate view to get a glance of their expenses. The user is now able to quickly view the one-time expenses they have had and the recurring expenses they face.

The user can now also receive notifications about updates to their expenses, automatically when a transaction occurs in their bank. This capability frees the user from having to check if any payment or deposit has been made in their bank's application, and in the event that the same application notifies them of the update, our system would still be useful, as it would free the user from manually modifying the associated expense and updating its status.

As a general conclusion, the work has led to the construction of a functional and technically complete software system, capable of covering the defined scope and thus

achieving a large part of the objectives set. Later, in the section on future work (6.3), some points are raised for a greater scope of the proposed objectives and an expansion of those.

At the technical level and as far as the part of the objectives related to the creation of a software system is concerned, the purpose has been fulfilled, obtaining a system composed of different services and applications that, together and collaboratively working, allow to bring value to the user and fulfill the main objectives of the work.

The resulting system is finally composed of two microservices and a web application, built in such a way that it is easily extensible to add new services and new interfaces, both for interaction with the user and to communicate with other external services.

6.3. Future work

Once the analysis and discussion of the results in relation to the objectives has been carried out, there are different areas in which the work could be expanded and generate an evolution of the resulting system in the future.

On one hand, with the same objectives set for this project, it would be possible to expand the functional scope achieving greater coverage of those objectives. This new scope could include aspects of functionality such as the visualization of monthly plans or the generation of expense reports, as well as more operation options on expenses or a higher degree of customization for the data and user-oriented applications.

At the user interaction level, this same scope expansion could include the construction of a mobile application, as mobile devices are the most used in a daily basis, and the ones that the user has at their disposal at all times, being the ideal recipient for notifications. As for notifications, a direct communication technology could also be implemented from the server to the web application, adding robustness to the system compared to the current resulting implementation.

In relation to banking information sources, as mentioned earlier, the use of email notifications or the manual entry of data from spreadsheet files provided by banks could be added in the future.

The work could also be completed in a greater way by polishing the processing logic between transactions and expenses, either by adding rules to the implemented algorithm or by using artificial intelligence systems for the automatic association and application of changes.

As for the methodology, especially taking into account the one used for the capture of requirements and the analysis prior to design, with surveys of a group of people, it would be convenient to add some validation steps related to the fulfillment of the extracted requirements. This validation would consist of interviewing the users again and presenting them with the final system or a prototype of it, obtaining their comments and opinions and analyzing to what extent their expectations are met.

Additionally, it is important to note with a future perspective that the European Union is already working on a renewal of the regulations related to banking information and operations in the field of open banking and the already known as PSD3. This renewal could bring new possibilities and cover the defined objectives set or other additional ones within the project's knowledge area.

Finally, it should be noted at a technical level that different improvements could be applied that have been skipped with the purpose of focusing the development on the value added to the user. Some examples of these improvements are security –including user and permission management– and the implementation of an API gateway or a proxy between the web application and the backend microservices; and a greater use of the capabilities of the language and the framework used, such as the reactive programming paradigm.

Bibliografía

- Bonér, J. et al. (2014). *The Reactive Manifesto*. URL: <https://www.reactivemanifesto.org/> (visitado 31-08-2023).
- Brown, Simon (2018). «The C4 Model for Software Architecture». En: *InfoQ*. URL: <https://www.infoq.com/articles/C4-architecture-model/> (visitado 31-08-2023).
- Cockburn, Alistair (2005). *Hexagonal Architecture*. URL: <https://alistair.cockburn.us/hexagonal-architecture/> (visitado 31-08-2023).
- Congreso de los Diputados (2018). «Real Decreto-ley 19/2018, de 23 de noviembre, de servicios de pago y otras medidas urgentes en materia financiera». En: *Boletín Oficial del Estado* 284. <https://www.boe.es/eli/es/rdl/2018/11/23/19/>, págs. 114474-114568.
- Gomber, Peter, Jascha-Alexander Koch y Michael Siering (2017). «Digital Finance and FinTech: current research and future research directions». En: *Journal of Business Economics* 87, págs. 537-580. DOI: <https://doi.org/10.1007/s11573-017-0852-x>.
- JetBrains (2022). *The State of Developer Ecosystem 2022*. URL: <https://www.jetbrains.com/lp/devecosystem-2022/> (visitado 31-08-2023).
- Kotlin Foundation (2023). *Maven – Kotlin*. URL: <https://kotlinlang.org/docs/maven.html> (visitado 31-08-2023).
- Parlamento Europeo y Consejo de la Unión Europea (2015). «Directiva (UE) 2015/2366 del Parlamento Europeo y del Consejo de 25 de noviembre de 2015 sobre servicios de pago en el mercado interior y por la que se modifican las Directivas 2002/65/CE, 2009/110/CE y 2013/36/UE y el Reglamento (UE) nº 1093/2010 y se deroga la Directiva 2007/64/CE». En: *Diario Oficial de la Unión Europea* L337. <https://eur-lex.europa.eu/eli/dir/2015/2366/oj>, págs. 37-127.
- Plaid (2023a). *Access token – Plaid Docs*. URL: <https://plaid.com/docs/quickstart/glossary/#access-token> (visitado 27-07-2023).
- (2023b). *API reference for the Transactions Sync endpoint – Plaid Docs*. URL: <https://plaid.com/docs/api/products/transactions/#transactionssync> (visitado 31-08-2023).
- (2023c). *API reference for Transactions webhooks – Plaid Docs*. URL: <https://plaid.com/docs/api/products/transactions/#webhooks> (visitado 21-07-2023).
- (2023d). *API reference for webhooks – Plaid Docs*. URL: <https://plaid.com/docs/api/webhooks/> (visitado 20-07-2023).
- (2023e). *Plaid OpenAPI specification*. URL: <https://github.com/plaid/plaid-openapi/blob/master/2020-09-14.yml> (visitado 27-07-2023).
- (2023f). *Token exchange flow – Plaid Docs*. URL: <https://plaid.com/docs/api/tokens/#token-exchange-flow> (visitado 20-07-2023).
- Richardson, Chris (2023a). *Microservice Architecture*. URL: <https://microservices.io> (visitado 31-08-2023).

- Richardson, Chris (2023b). *Pattern: API Gateway / Backends for Frontends - Microservice Architecture*. URL: <https://microservices.io/patterns/apigateway.html> (visitado 31-08-2023).
- Skan, Julian, James Dickerson y Samad Masood (2015). «The future of fintech and banking: digitally disrupted or reimaged?» En: *FinTech Innovation Lab*.
- Spring Framework (2023). *WebClient – Spring WebFlux – Spring Framework*. URL: <https://docs.spring.io/spring-framework/reference/web/webflux-webclient.html> (visitado 27-07-2023).
- Stack Overflow (2023). *Stack Overflow Developer Survey 2023*. URL: <https://survey.stackoverflow.co/2023/> (visitado 31-08-2023).
- Suryono, Ryan Randy, Indra Budi y Betty Purwandari (2020). «Challenges and Trends of Financial Technology (Fintech): A Systematic Literature Review». En: *Information* 11.12. DOI: 10.3390/info11120590. URL: <https://www.mdpi.com/2078-2489/11/12/590>.
- University, Software Engineering Institute - Carnegie Mellon (2021). *Software Architecture*. URL: <https://www.sei.cmu.edu/our-work/software-architecture/> (visitado 31-08-2023).

Apéndice A

Comparativa de agregadores bancarios de tipo proveedor AIS de open banking

Nombre	Web	Cuenta creada	Pruebas realizadas	App/UI de conexión de cliente - SDK front-end para autenticación de usuario final	Sandbox/playground	Pruebas reales (pre-producción)	Requisitos para acceso al entorno de producción	Avisos/eventos vía AMQP o webhooks	Valoración
Klarna Kosma	kosma.com	Sí	Probada obtención transacciones de cuenta demo vía API con Postman: iniciar una sesión, iniciar un flujo, iniciar sesión en el banco a través de la XS2A App en el cliente, obtener los datos de las transacciones y cerrar la sesión.	Sí: XS2A App	Sí (entorno demo)	No	Contacto a través de formulario y envío de información	No	No se ha obtenido respuesta. No ofrecen avisos/eventos y no tiene pruebas reales, por lo que por ahora se descarta.
Salt Edge	saltedge.com	Sí	En cuenta demo vía Postman: creado cliente de prueba, creada sesión de conexión, iniciado sesión en el banco a través del Connect Widget en el cliente, obtenidas conexiones, obtenidas cuentas y obtenidos movimientos.	Sí: Salt Edge Connect	Sí (entorno demo)	Sí, 100 conexiones, bajo aprobación	Contacto a través de formulario y envío de información. Necesarios detalles y aprobación	No	No ofrecen avisos/eventos, pero sí pruebas reales bajo aprobación. No es una opción descartable.
Tink	tink.com	Sí	Creada app sandbox, creado usuario, generada URL para autenticación, autenticado usuario en el banco a través de Tink Link en el cliente, obtenidos datos del usuario, obtenidas cuentas y obtenidos movimientos.	Sí: Tink Link	Sí	Sí, bajo aprobación	Desconocido	Sí	Es una buena opción, aunque no permite probar los envíos de eventos a través de webhooks.
Plaid	plaid.com	Sí	Solicitado Link Token, abierto Link a través de SDK en React, realizada autenticación, intercambiado public token por access token y obtenidas cuentas y movimientos.	Sí: Plaid Link	Sí	Sí, 100 conexiones, bajo aprobación	Contacto a través de formulario y envío de información. Necesarios detalles y aprobación	Sí	Permite probar los envíos de eventos y generar usuarios con datos de prueba (transacciones) personalizados. Además tiene una documentación muy comprensible, vídeos de ayuda y una gran cantidad de ejemplos y demostraciones con código público. Es la mejor opción.

Apéndice B

Formularios de las encuestas del análisis de requisitos

Nótese que se diseñaron y utilizaron tres formularios distintos, uno para cada persona tipo. Sin embargo, por motivos de espacio, en este apéndice se muestra un único formulario unificado, con las partes 2.A, 2.B y 2.C diferenciadas dentro del mismo.

Encuesta sobre gastos personales

¡Hola! En primer lugar, quisiera darte las **gracias** echarme una mano para completar mi Trabajo de Fin de Grado en Ingeniería Informática. A continuación te plantearé algunas preguntas relacionadas con tus gastos personales y cómo los gestionas. Esta información se utilizará para diseñar y definir el proyecto final, y se incorporará, anonimizada, a la memoria.

Por favor, indica "Sí" a continuación para confirmar que comprendes y aceptas la recogida de tus datos y su publicación según se indica arriba.

(Sí/No)

Indicaciones para responder a la encuesta

Es recomendable rellenar esta encuesta en un ordenador, por el tamaño del contenido. A lo largo de la encuesta te encontrarás con algunas preguntas de respuesta libre, otras de selección y otras de datos concretos. Verás las **preguntas en negrita**, la descripción como texto normal, las **guías de respuesta en azul** y las **respuestas de ejemplo en naranja**.

1.1. Introducción personal

1.1.1. En general, actualmente, ¿a qué tipo de gastos tienes que hacer frente?

En este apartado me gustaría que me contaras en qué situación te encuentras y a qué gastos tienes que hacer frente en tu día a día, semana a semana o mes a mes. Debe ser a modo de resumen, aunque puedes explayarte todo lo que quieras. No es necesario que detalles aquí tus gastos, eso viene más adelante. Por ejemplo: **Actualmente vivo en un piso compartido con estudiantes, por lo que mensualmente cubro los gastos de mi casa. Además, tengo varias suscripciones de entretenimiento y hago ejercicio en un gimnasio.**

Tu respuesta:

1.2. Filtro de idoneidad para la encuesta

1.2.1. Mes a mes, ¿realizas gastos¹ relacionados con tu hogar², suscripciones³, financiación, seguros u otros cargos recurrentes?

¹Ya sean domiciliados o manuales.

²Hogar: alquiler, agua, luz, gas, telefonía, internet, etc.

³Suscripciones: servicios de streaming de vídeo o música, servicios de comida a domicilio, etc.

(Sí/No)

1.2.2. ¿Utilizas alguna aplicación (móvil o de ordenador) para la gestión de tus gastos personales?

(Sí/No)

1.2.3. Si no es así, ¿utilizas algún otro sistema (manual o digital) para el mismo fin?

(Sí/No)

(Si en la 1.2.2. respondiste "Sí", no respondas a esta pregunta)

1.2.4. Si no utilizas sistema alguno para gestionar tus gastos, ¿crees que podrías beneficiarte de utilizar alguno?

(Sí/No)

(Si en la 1.2.3. respondiste "Sí" o no respondiste, no respondas a esta pregunta)

Si respondiste "No" en esta última pregunta, por favor, ponte en contacto conmigo.

(SOLO TIPO A) 2. Evaluación sobre el sistema utilizado actualmente

2.1. ¿Qué aplicación de gestión de gastos utilizas actualmente?

2.2. ¿Con qué frecuencia accedes a la aplicación?

2.3. ¿La aplicación te permite obtener una previsión de lo que vas a gastar en el mes siguiente? ¿Es un aspecto interesante para ti?

2.4. ¿La aplicación te permite manejar tus gastos recurrentes (luz, agua, suscripciones, etc.)? ¿Es un aspecto interesante para ti?

2.5. ¿La aplicación te calcula una cantidad a reservar para los gastos recurrentes del mes siguiente? ¿Es un aspecto interesante para ti?

2.6. ¿La aplicación está conectada con tu banco y puedes ver tus movimientos de manera dinámica? ¿Es un aspecto interesante para ti?

2.7. ¿La aplicación te indica cuando se ha producido un cobro? ¿Es un aspecto interesante para ti?

2.8. ¿Recibes notificaciones de la aplicación? ¿Son interesantes para ti?

2.9. ¿La aplicación te ofrece resúmenes de gasto? ¿Son interesantes para ti?

2.10. ¿La aplicación te permite establecer objetivos de ahorro? ¿Es un aspecto interesante para ti?

2.11. ¿La aplicación te permite categorizar los gastos o los categoriza automáticamente? ¿Es un aspecto interesante para ti?

2.12. De las funcionalidades de la aplicación mencionadas, ¿cuáles valoras o utilizas más?

2.13. ¿Qué otras funcionalidades no mencionadas ofrece la aplicación y utilizas habitualmente?

2.14. ¿Has probado otras aplicaciones? ¿Qué aspecto te hizo decantarte por la que usas actualmente?

(SOLO TIPO B) 2. Evaluación sobre el sistema utilizado actualmente

2.1. ¿Qué sistema de gestión de gastos utilizas actualmente? ¿Lo has creado personalmente o has utilizado una plantilla?

2.2. ¿Con qué frecuencia utilizas o accedes a la hoja de cálculo?

2.3. ¿La hoja de cálculo te permite obtener una previsión de lo que vas a gastar en el mes siguiente? ¿Es un aspecto interesante para ti?

2.4. ¿La hoja de cálculo te permite manejar tus gastos recurrentes (luz, agua, suscripciones, etc.)? ¿Es un aspecto interesante para ti?

2.5. ¿La aplicación te calcula una cantidad a reservar para los gastos recurrentes del mes siguiente? ¿Es un aspecto interesante para ti?

2.6. ¿La hoja de cálculo te permite registrar cuando se ha producido un cobro y por cuánto ha sido? ¿Es un aspecto interesante para ti?

2.7. ¿La hoja de cálculo te ofrece resúmenes de gasto? ¿Son interesantes para ti?

2.8. ¿La hoja de cálculo te permite establecer objetivos de ahorro? ¿Es un aspecto interesante para ti?

2.9. ¿La hoja de cálculo te permite categorizar los gastos? ¿Es un aspecto interesante para ti?

2.10. De las funcionalidades de la hoja de cálculo mencionadas, ¿cuáles valoras o utilizas más?

2.11. ¿Qué otras funcionalidades no mencionadas ofrece la hoja de cálculo y utilizas habitualmente?

2.12. ¿Has probado otros sistemas como, por ejemplo, aplicaciones? ¿Qué aspecto te hizo decantarte por la hoja de cálculo?

2.13. ¿Te gustaría que el sistema que utilices pudiera conectarse a tu banco y añadir movimientos de manera dinámica?

2.14. ¿Te gustaría que el sistema que utilices pudiera enviarte notificaciones para estar al tanto de tus gastos? Si puedes, detalla cómo.

(SOLO TIPO C) 2. Evaluación sobre el sistema utilizado actualmente

2.1. ¿Utilizas la aplicación de tu banco para gestionar tus gastos recurrentes? ¿Qué sueles observar?

2.2. ¿Con qué frecuencia utilizas o accedes a la aplicación de tu banco?

2.3. ¿Llevas a cabo una previsión de lo que vas a gastar en el mes siguiente? ¿Te gustaría que una aplicación lo hiciera por ti?

2.4. ¿Reservas alguna cantidad para los gastos recurrentes del mes siguiente? ¿Cómo haces el cálculo?

2.5. ¿Te gustaría que una aplicación te avisara de una manera intuitiva sobre qué gastos recurrentes te cobran?

2.6. ¿Haces resúmenes, balances o análisis de tus gastos? ¿Cómo los llevas a cabo?

2.7. ¿Estableces objetivos de ahorro? ¿En qué te basas?

2.8. En tus gestiones sobre gastos recurrentes, ¿qué parte te parece más tediosa?

2.9. ¿Has probado aplicaciones u hojas de cálculo para llevar tus gastos personales? ¿Qué aspecto te hizo decantarte por no usarlas?

2.10. ¿Qué otros aspectos buscarías en una aplicación para gestionar tus gastos personales?

2.11. ¿Te gustaría que la aplicación pudiera conectarse a tu banco y añadir movimientos de manera dinámica?

2.12. ¿Te gustaría que la aplicación pudiera enviarte notificaciones para estar al tanto de tus gastos? Si puedes, detalla cómo.

3. Evaluación sobre gastos

3.1. Concretamente, ¿a qué gastos haces frente y cuáles son sus características?

Esta pregunta es la más técnica y detallada de todas. Consiste en que indiques, de la manera más concreta que puedas, a qué gastos haces frente actualmente y qué características tienen. Para ello, a continuación verás una tabla con algunos de los gastos más comunes, en la que cada fila es un gasto y las columnas indican características de cada gasto.

Si tienes gastos que no aparezcan en la tabla, puedes **añadir tantas filas como desees**.

Si en algún gasto ves una columna que no tiene sentido para tu caso o que **desconoces**, puedes dejarla en blanco o escribir N/A.

Si alguno de tus gastos viene agrupado con otro, por ejemplo, un paquete de televisión y telefonía, puedes añadirlo de manera independiente.

Puedes dejar **comentarios** sobre los gastos en el campo que hay bajo la tabla.

En **naranja** puedes ver **ejemplos de valores** para cada campo.

Aquí te explico las columnas:

- **Plataforma o empresa:** quién te ofrece el servicio, o a quién pagas el gasto.

- **Importe fijo o variable:** si importe del gasto es fijo (todos los meses igual) o variable (según el consumo o periodo, por ejemplo agua y luz).

- **Domiciliado:** si el gasto se te cobra automáticamente, ya sea por tenerlo domiciliado en una cuenta o asociado a una tarjeta.

- **Frecuencia:** cada cuánto tienes ese gasto (mensual, bimensual, trimestral, otro...).

- **Día de cobro:** si lo sabes, indica cuándo se produce cada gasto (cuando te pasan el recibo, cuándo haces el pago... por ejemplo "el día 5 de cada mes", "el primer viernes de cada mes"...).

Categoría	Gasto	Plataforma o empresa		Importe fijo o variable		Domiciliado (Sí/No)	Frecuencia	Día de cobro
		Ejemplos:	Tu respuesta:	Ejemplos:	Tu respuesta:			
Hogar	Alquiler	Particular, Alquiler Seguro...		Fijo				
	Agua	Ista, Aguas de Murcia...		Variable				
	Luz	Endesa, Iberdrola...		Variable				
	Gas	Iberdrola, Curenergía, Repsol...		Variable				
	Telefonía	Movistar, Vodafone...		Fijo				
	Internet	Movistar, Vodafone...		Fijo				
(↑ Añade más si tienes...)								
Suscripciones	Streaming de vídeo	Netflix, Disney+, Amazon Prime Video, HBO Max, YouTube Premium, Apple TV+...		Fijo				
	Streaming de música	Spotify, Apple Music, Amazon Music, YouTube Music...		Fijo				
	Compras y envíos	Amazon Prime...		Fijo				
	Comida a domicilio	Uber Eats Pass...		Fijo				
	Prensa y noticias	El País, La Vanguardia...		Fijo				
	Almacenamiento en la nube	Google Drive, Dropbox, Mega...		Fijo				
Transporte	Uber Pass...		Fijo					

	Otras suscripciones	Twitter Blue, Google One, iCloud+, Apple One, hosting web, dominios web, plataformas de gaming y videojuegos...	Fijo
	(↑ Añade más si tienes...)		
Financiación	Préstamo bancario		Fijo
	Tarjeta de crédito		Variable
	Compra financiada en Apple	Apple/Cetelem	Fijo
	Compra financiada en IKEA	IKEA/CaixaBank	Fijo
	(↑ Añade más si tienes...)		
Seguros	Seguro de móvil	Apple Care, Apple Care+...	Fijo
	Seguro de ordenador	Apple Care, El Corte Inglés...	Fijo
	Seguro de salud	Sanitas, Quirón...	Fijo
	(↑ Añade más si tienes...)		
Salud y bienestar	Gimnasio	McFit...	Fijo
	Medicamentos		Variable
	Psicólogo		Fijo
	Fisioterapeuta		Variable
	(↑ Añade más si tienes...)		
Otros gastos recurrentes	Abono de transportes		Fijo
	Compras personales muy recurrentes		Variable
	(↑ Añade más si tienes...)		

Comentarios:

3.2. Si tienes algún gasto domiciliado, ¿tu banco te notifica cuando se produce el cobro? ¿Por qué medio lo hace (app, email...)?

Ejemplo: email de Banco Sabadell que indica "el recibo de Endesa se cargará en tu cuenta 3637 el día 2023-06-07 por importe de 20.00 EUR".

3.3. ¿Qué entidades te notifican cuando se va a producir un cobro? ¿Por qué medio lo hace (app, email...)?

Ejemplo: email de Endesa con asunto "Ya puedes descargar tu factura.", que contiene el importe de la factura.

Ejemplo: email de Apple con asunto "Your invoice from Apple.", que contiene el importe de la factura.

3.4. En caso de tener varios gastos domiciliados, ¿dispones de una única cuenta bancaria o tienes varias para distintos propósitos?

4. Agradecimiento final

Te agradezco enormemente tu colaboración en este proyecto, me ayudará a realizar un trabajo de mayor calidad y más cercano a la realidad.

¡Un abrazo!

4.1. ¿Te gustaría conocer el resultado final del proyecto?

Si deseas que contacte contigo para mostrarte el resultado final del proyecto, escribe "Sí" a continuación.

(Sí/No)

Apéndice C

Respuestas a las encuestas del análisis de requisitos

Parte 1: introducción personal y filtro de idoneidad para la encuesta (screener previo)

Pregunta	Persona encuestada	Respuesta
En general, actualmente, ¿a qué tipo de gastos tienes que hacer frente?	N. Hervás	Vivo de alquiler por lo que todos los meses he de pagar la mensualidad así como los gastos referentes al agua, gas, electricidad y wifi. Adicionalmente, hay que hacer que hacer la compra por lo que eso también es una fuente de gastos. Gran parte de los gastos mensuales son en ocio, bien en restauración o en salir a tomar algo y/o de fiesta. Por otro lado, pago el abono de transportes y cada 4 meses el abono de autobús para volver a mi ciudad natal. De manera ocasional también gasto dinero en viajes y ropa. Aparte de eso, no tengo gastos superfluos.
	A. Leiva	Vivo compartiendo piso, con lo que tengo gastos domésticos, el pago mensual por la compra del coche, suscripción a varias plataformas y gastos fijos por mi situación laboral (cuenta de autónomos)
	J. Ferrer	Vivo yo solo, independizado en un piso de dos habitaciones más plaza de garaje, con lo que cubro todos los gastos fijos de casa + energía. Tengo suscripción a varias plataformas de entretenimiento; y, por mi trabajo, tengo gastos importantes en gasolina (no hay otro medio de transporte para llegar) y abono del parking de empleados.
	H. Marco	Vivo en el piso de un familiar con mi pareja. Al año soporto los gastos fijos siguientes: Internet, tlf., letra del coche, Teaming (donación), impuestos del vehículo, suscripción a Amazon Prime. No obstante, aun siendo estos fijos, mis principales gastos son los variables; siendo estos los siguientes: comida (incluyendo las de ocio), mascotas, gastos del hogar (electrodomésticos, muebles, reparaciones, decoración, etc.), ropa, salud y medicinas, gasolina, inversiones...
	P. Bonet	Vivo solo en un piso de alquiler, trabajo y pago mis gastos, que van desde el alquiler hasta Spotify, pasando por los suministros de casa y pagos mensuales de financiación de compras.
	Y. Díaz	Alquiler, alimentación, mantenimiento coche.
	A. Zhou	Actualmente vivo en un piso por lo que tengo que sufragar los gastos que conlleva. Adicionalmente pago algunas suscripciones, relacionadas con entretenimiento o algunas con el hogar.
Mes a mes, ¿realizas gastos ¹ relacionados con tu hogar ² , suscripciones ³ , financiación, seguros u otros cargos recurrentes?	N. Hervás	Sí
	A. Leiva	Sí
	J. Ferrer	Sí
	H. Marco	Sí
	P. Bonet	Sí
	Y. Díaz	Sí
	A. Zhou	Sí
¿Utiliza alguna aplicación (móvil o de ordenador) para la gestión de sus gastos personales?	N. Hervás	Sí
	A. Leiva	No
	J. Ferrer	No
	H. Marco	No
	P. Bonet	No
	Y. Díaz	No
	A. Zhou	No
Si no es así, ¿utiliza algún otro sistema (manual o digital) para el mismo fin?	N. Hervás	N/A
	A. Leiva	Sí
	J. Ferrer	Sí
	H. Marco	Sí
	P. Bonet	Sí
	Y. Díaz	No
	A. Zhou	No
Si no utiliza sistema alguno para gestionar sus gastos, ¿cree que podría beneficiarse de utilizar alguno?	N. Hervás	N/A
	A. Leiva	N/A
	J. Ferrer	N/A
	H. Marco	N/A
	P. Bonet	N/A
	Y. Díaz	Sí
	A. Zhou	Sí

Parte 2.A: preguntas para la persona tipo A sobre el sistema que utiliza

Pregunta	Persona encuestada	Respuesta
¿Qué aplicación de gestión de gastos utilizas actualmente?	N. Hervás	Money Manager
¿Con qué frecuencia accedes a la aplicación?	N. Hervás	Cada vez que hago un gastos para apuntarlo y categorizarlo correctamente
¿La aplicación te permite obtener una previsión de lo que vas a gastar en el mes siguiente? ¿Es un aspecto interesante para ti?	N. Hervás	Permite configurar presupuestos por categorías
¿La aplicación te permite manejar tus gastos recurrentes (luz, agua, suscripciones, etc.)? ¿Es un aspecto interesante para ti?	N. Hervás	Permite gestionarlos como cualquier otro gasto normal. Los gastos van categorizados y se pueden marcar como recurrentes si se desea
¿La aplicación te calcula una cantidad a reservar para los gastos recurrentes del mes siguiente? ¿Es un aspecto interesante para ti?	N. Hervás	La aplicación permite configurar presupuestos para lo que sea, en este caso gastos recurrentes. Personalmente no es una funcionalidad que utilice pero puede ser interesante para otros usuarios
¿La aplicación está conectada con tu banco y puedes ver tus movimientos de manera dinámica? ¿Es un aspecto interesante para ti?	N. Hervás	La aplicación no está conectada al banco y por lo tanto no es dinamica pero prefiero que sea así porque de esta manera puedes contabilizar el dinero en efectivo, cosa que de la otra manera no es posible. A pesar de no estar conectado al banco puedes establecer distintas cuentas bancarias digamos y te hace un resumen del dinero que tendrías y gastos realizados
¿La aplicación te indica cuando se ha producido un cobro? ¿Es un aspecto interesante para ti?	N. Hervás	Al no estar conectado al banco no puede notificar cobros pero si que es interesante tener esa funcionalidad para no depender de la app del banco siempre y cuando exista privacidad con los datos del usuario y banco
¿Recibes notificaciones de la aplicación? ¿Son interesantes para ti?	N. Hervás	No recibo notificaciones, no lo considero necesario puesto que sería un poco invasivo para mi pero otros usuarios que no tengan interiorizado inputar los gastos pueden encontrar util que se les recuerde apuntar sus gastos
¿La aplicación te ofrece resúmenes de gasto? ¿Son interesantes para ti?	N. Hervás	La aplicación tiene una vista por la que mediante de un pie chart te muestra un resumen del gasto por categorías
¿La aplicación te permite establecer objetivos de ahorro? ¿Es un aspecto interesante para ti?	N. Hervás	No lo permite y yo no lo necesito porque no soy una persona con problemas para ahorrar
¿La aplicación te permite categorizar los gastos o los categoriza automáticamente? ¿Es un aspecto interesante para ti?	N. Hervás	La aplicación permite crear categorías y subcategorías para los gastos, es de las mejores funcionalidades para entender en que se gasta el dinero
De las funcionalidades de la aplicación mencionadas, ¿cuáles valoras o utilizas más?	N. Hervás	Tener los gastos en una vista de calendario, diario o mensual. Estadísticas de ingresos y gastos con graficos por categorías para los periodos registrados
¿Qué otras funcionalidades no mencionadas ofrece la aplicación y utilizas habitualmente?	N. Hervás	Poder importar y exportar todos los gastos historicos, poder adjuntar una imagen a los gastos y poner comentarios, poder usar una moneda secundaria para cuando se viaje fuera del EEE
¿Has probado otras aplicaciones? ¿Qué aspecto te hizo decantarte por la que usas actualmente?	N. Hervás	Probé Fintonic y me pareció muy poco flexible, no me permitía hacer ajustes ni tener una vista clara de los gastos historicos

Parte 2.B: preguntas para la persona tipo B sobre el sistema que utiliza

Pregunta	Persona encuestada	Respuesta
¿Qué sistema de gestión de gastos utilizas actualmente? (Hoja de cálculo) ¿Lo has creado personalmente o has utilizado una plantilla?	A. Leiva J. Ferrer H. Marco P. Bonet	Hoja de cálculo de Excel. Hoja de cálculo de Excel. Hoja de cálculo de Excel. Hoja de cálculo de Excel.
¿Con qué frecuencia utilizas o accedes a la hoja de cálculo?	A. Leiva J. Ferrer H. Marco P. Bonet	Intento cada mes, pero realmente cuando puedo Si no es diariamente, al menos una vez cada par de días A diario. Cada vez que realizo uno de los gastos de los que llevo la cuenta, es decir, varias veces por semana.
¿La hoja de cálculo te permite obtener una previsión de lo que vas a gastar en el mes siguiente? ¿Es un aspecto interesante para ti?	A. Leiva J. Ferrer H. Marco P. Bonet	No, la uso guía para saber cuales son mis gastos aprox Totalmente, tras varios meses de introducción de datos he aprendido cuál es mi hábito de gasto estándar, y puedo comparar mes a mes el porcentaje de gasto que he tenido + añadir comentarios en algunas de las celdas para explicar si hubo algún gasto mayor no previsto. Me permite hacer estimaciones con base en los gastos producidos en los meses anteriores y teniendo en cuenta mis necesidades. Sí, lo es. Sí, tengo algunas fórmulas configuradas para tener una previsión de lo que tengo que "bloquear" para el mes siguiente. Para mí es fundamental.
¿La hoja de cálculo te permite manejar tus gastos recurrentes (luz, agua, suscripciones, etc.)? ¿Es un aspecto interesante para ti?	A. Leiva J. Ferrer H. Marco P. Bonet	Sí. Me parece interesante por saber cuanto tengo que cubrir mas o menos Sí, en la columna de gastos hay una celda para cada tipo de gasto fijo y así puedo comprender mejor la variación del total mensual dependiendo de la estación del año Sí y sí. Sí, para cada gasto tengo una fila y voy apuntando de qué tipo es cada gasto, cuando empieza, cuando acaba, cuándo me lo cobran, cuánto espero que me cobren y cuánto me cobran finalmente. Saber cuánto espero que me cobren es importante para tener una previsión, y saber cuánto me cobran finalmente me ayuda a hacer algo de análisis a posteriori.
¿La aplicación te calcula una cantidad a reservar para los gastos recurrentes del mes siguiente? ¿Es un aspecto interesante para ti?	A. Leiva J. Ferrer H. Marco P. Bonet	No En este momento no he creado esa fórmula en mi Excel, cuando empecé me fijé un ahorro fijo mensual aleatorio, pero sí puede ser interesante saber cuánto % de la nómina recibida ahorrar (ya que en mi caso, cada mes cobro algo distinto, no es un ingreso fijo) No, no tiene esa opción. Es por eso que uso 4 hojas de cálculo: gastos, ingresos, presupuesto e inversiones. Sí es un aspecto interesante, pero con la hoja de cálculo lo tengo que hacer yo manualmente mes a mes. Me indica la suma de los gastos previstos. Sí, es importante.

¿La hoja de cálculo te permite registrar cuando se ha producido un cobro y por cuánto ha sido? ¿Es un aspecto interesante para ti?	<p>A. Leiva Sí. Me parece interesante para llevarlo claro y ordenado</p> <p>J. Ferrer Sí, existen dos celdas, la del cobro de la nómina y la de "otros ingresos", en la que suelo dejar un comentario haciendo referencia al origen de ese ingreso. Es interesante en cuanto a que es necesario para la forma en que está diseñado el excel, que si no tuviera en cuenta los ingresos sería imposible calcularme unos porcentajes y unos gráficos de los que también hago uso</p> <p>H. Marco No. Tengo columnas por meses, pero no puedo hacer un registro de la fecha exacta. Sí lo es. Sería muy interesante tener una APP donde se generaran datos exactos, como la fecha del movimiento, a quién se le hizo, etc.</p> <p>P. Bonet Por cada gasto tengo asociado un mes en el que voy anotando cuándo se ha producido el cobro o cuándo lo he pagado, y cuánto me han cargado finalmente.</p>
¿La hoja de cálculo te ofrece resúmenes de gasto? ¿Son interesantes para ti?	<p>A. Leiva Sí. Si lo son para saber en qué gasto más</p> <p>J. Ferrer Creé tres celdas para cada columna de mes: una indica la cantidad en euros de gasto total, otra el porcentaje de gasto que se genera en la primera semana del mes (asociada a los gastos fijos de la vivienda) y otra que indica el porcentaje de gasto con respecto a mi capacidad financiera total. Todos me resultan interesantes, aunque los 2 últimos más; con el "% de gasto primera semana" consigo ver si los gastos que tengo son realmente importantes e imprescindibles o me estoy dejando llevar demasiado por los planes sociales, y con el "% gasto capacidad financiera" puedo ver como, mes a mes, esa ratio va disminuyendo, lo cuál indica que estoy siendo capaz de ahorrar considerablemente cada mes</p> <p>H. Marco No, no ofrece resúmenes de gastos, aunque sí que podría generarlos haciéndola una tabla dinámica y generando gráficos con ella. Sí que es interesante para mí.</p> <p>P. Bonet No me los ofrece, pero podrían ser interesantes.</p>
¿La hoja de cálculo te permite establecer objetivos de ahorro? ¿Es un aspecto interesante para ti?	<p>A. Leiva No. Si lo es pero tengo otras herramientas para el ahorro</p> <p>J. Ferrer No; me parecería interesante porque no cobro lo mismo cada mes, y siendo realista no puedo querer alcanzar cada mes el mismo nivel de ahorro</p> <p>H. Marco Sí, pero los he de establecer manualmente. También es un aspecto interesante para mí</p> <p>P. Bonet No me lo permite, por ahora no me llama mucho la atención.</p>
¿La hoja de cálculo te permite categorizar los gastos? ¿Es un aspecto interesante para ti?	<p>A. Leiva Si. Si, para hacerme una idea de en qué gasto más</p> <p>J. Ferrer Sí. La esencia del Excel era separar los gastos en distintas filas (categorías) para entender mejor dónde se estaba yendo mi dinero, y poder tomar decisiones concretas de recorte (si es que fuese necesario)</p> <p>H. Marco Sí, pero he de generar yo las categorías y subcategorías. Es un aspecto fundamental para llevar un control de los gastos y poder saber dónde estoy invirtiendo o malgastando mi dinero.</p> <p>P. Bonet Sí, cada gasto está dentro de una categoría, me ayuda a ver en qué gasto más. Sería interesante que tuviera también etiquetas.</p>
De las funcionalidades de la hoja de cálculo mencionadas, ¿cuáles valoras o utilizas más?	<p>A. Leiva Resúmenes de gastos y la categoría de los mismos</p> <p>J. Ferrer Las ayudas visuales; en mi caso los gráficos y las escalas de color en determinadas celdas, que me ayudan a ver la evolución durante los últimos meses de mis finanzas</p> <p>H. Marco Lo que más valoro es lo que precisamente le falta: poder introducir los gastos directamente desde una APP de Smartphone en la categoría y mes que yo establezca y, además, contenga metadatos sobre la fecha exacta, negocio/motivo, si se pagó con tarjeta o en efectivo, con qué tarjeta y con qué cuenta bancaria, etc. No obstante, de las mencionadas, lo que más valoro es la facilidad de organizar y categorizar los gastos.</p> <p>P. Bonet Especialmente la previsión y la suma, porque me ayuda a saber qué cantidad de mis ingresos debo tener siempre disponible para hacer frente a los gastos que me van a llegar. Lo más interesante es que dicha suma se vaya actualizando, ya que tengo una cuenta bancaria reservada a este tipo de gastos, y puedo ver en cada momento cuánto dinero debería tener en esa cuenta.</p>
¿Qué otras funcionalidades no mencionadas ofrece la hoja de cálculo y utilizas habitualmente?	<p>A. Leiva Ninguna, es sencillita</p> <p>J. Ferrer El cálculo de promedios para cada categoría de gasto, con lo que suelo hacerme una previsión de en cuánto a qué cantidad rondará el gasto el mes siguiente</p> <p>H. Marco Llevar un control de los gastos totales, tanto por cuantía como por porcentaje, lo que me permite con tener dos columnas que me dan un resumen general de mis gastos durante todo el año.</p> <p>P. Bonet En la hoja puedo indicar manualmente si me han cobrado el gasto o lo he pagado, y detallar qué cantidad me han cobrado finalmente. También es interesante que puedo introducir esa cifra en cuanto la conozco, por ejemplo, si el banco me notifica que me van a pasar un recibo de luz para el que tenía previstos 40 € y ya sé que el importe final será de 25 €, puedo dejarlo anotado y liberar 15 €. Si por el contrario el importe final es mayor, tengo algo más de tiempo para asegurar esa cantidad, si lo necesito (para casos más grandes, es importante, por ejemplo, con el gas en invierno).</p>
¿Has probado otros sistemas como, por ejemplo, aplicaciones? ¿Qué aspecto te hizo decantarte por la hoja de cálculo?	<p>A. Leiva Sí. Sobre todo privacidad, que una aplicación sepa exactamente mis gastos y mis ingresos no me hacía mucha gracia.</p> <p>J. Ferrer Sí, antes utilizaba Fintonic y también la App de mi banco, pero no me ofrecían gráficos tan personalizados como yo quería, ni se podía ver de un vistazo tantas ratios mes a mes para poder comparar, que era lo que me interesaba. Tampoco me ofrecían la evolución neta de mi capacidad financiera</p> <p>H. Marco Sí, he probado alguna aplicación móvil o he intentado guiarme por el gestor de gastos que incorporan algunas APP de las entidades bancarias, pero lo primero presenta muy pocas opciones de personalización y categorización y lo segundo no deja control alguno ya que los gastos se categorizan de forma automática, lo cual puede hacer que realizar dos gastos distintos en el mismo negocio la APP los agregue a categorías distintas.</p> <p>P. Bonet Sí, he probado varias aplicaciones para móvil y escritorio, pero no cubrían mis necesidades, no permitían hacer previsiones, enviaban demasiadas notificaciones y algunas eran de pago.</p>
¿Te gustaría que el sistema que utilices pudiera conectarse a tu banco y añadir movimientos de manera dinámica?	<p>A. Leiva Sí</p> <p>J. Ferrer Lo he pensado alguna vez, pero creo que el hecho de que tenga que introducir yo manualmente las cantidades hace que compruebe los movimientos en mi cuenta y si algo no me cuadra, lo estudie más a fondo.</p> <p>H. Marco A mí banco no, pero sí a una aplicación donde yo tenga la capacidad de generar tablas dinámicas para añadir los gastos con sus respectivo metadatos y posteriormente poder modificar la vista de los mismos por año, categoría, mes, etc. sin necesidad de modificar la plantilla.</p> <p>P. Bonet Sí, sería estupendo no tener que añadir los gastos de forma manual. Quizá tampoco automática del todo, pero sí que la aplicación me pregunte de manera inteligente.</p>
¿Te gustaría que el sistema que utilices pudiera enviarte notificaciones para estar al tanto de tus gastos? Si puedes, detalla cómo.	<p>A. Leiva Sí si por ejemplo un mes he gastado mas en una categoría que en otra para controlarlo.</p> <p>J. Ferrer De algún modo, las escalas de color en las celdas de "ahorro bruto" y "ahorro real" me van avisando cada vez que introduzco algún movimiento</p> <p>H. Marco Sí que me gustaría. Ya sea que el sistema fuere manual o automático, sería bueno que se pudieran poner topes de gasto por categoría o mes y que si me acerco a dicho tope el software me notificara dicha situación. También que me avisara de la previsión de gastos fijos para el mes siguiente durante la última semana del mes presente para no olvidar ningún gasto puntual, así como que me pueda notificar previsiones de gastos para el mes siguiente partiendo de mi histórico de gastos en los meses previos o en el mismo mes durante años anteriores. Otra opción muy interesante sería que me notificara si alguien que no sea yo ha realizado algún gasto con alguna de las cuentas bancarias que comparto con esa persona. O, además de lo anterior, que si dicha plataforma de gastos la comparto con terceras personas, estas sean notificadas cuando alguien que no son ellas mismas realiza o introduce un gasto. Sobre todo que se puedan enviar alertas prioritarias si los gastos superan una cuantía determinada establecida previamente por el usuario. Esto sería estupendo para cuentas de entidades, negocios, clubes, asociaciones, peñas o cualquier otro tipo de agrupación donde varias personas pueden hacer uso del dinero común. Sin perjuicio de lo anterior, sería bueno que el usuario pueda elegir cómo prefiere recibir dicha notificación: notificación push, email, SMS, alerta por WhatsApp/Telegram, etc.</p> <p>P. Bonet Sí, sería interesante que me avisara cuando se me va a cobrar o se me cobra un gasto que tenía previsto y configurado en la aplicación.</p>

Parte 2.C: preguntas para la persona tipo C sobre el sistema que utiliza

Pregunta	Persona encuestada	Respuesta
¿Utilizas la aplicación de tu banco para gestionar tus gastos recurrentes? ¿Qué sueles observar?	Y. Díaz	Si, solo miro el saldo, algunos movimientos y poco más
	A. Zhou	Sí, generalmente reviso los cobros que aparecen para ver si se corresponden con los gastos realizados.
¿Con qué frecuencia utilizas o accedes a la aplicación de tu banco?	Y. Díaz	Varias veces por semana
	A. Zhou	3-4 veces por semana
¿Llevas a cabo una previsión de lo que vas a gastar en el mes siguiente? ¿Te gustaría que una aplicación lo hiciera por ti?	Y. Díaz	Si y Si
	A. Zhou	Actualmente no llevo una previsión de mis gastos futuros, (aunque sí tengo en mente la cantidad de ciertos gastos variables como puede ser la luz de casa)
¿Reservas alguna cantidad para los gastos recurrentes del mes siguiente? ¿Cómo haces el cálculo?	Y. Díaz	No
	A. Zhou	Generalmente no
¿Te gustaría que una aplicación te avisara de una manera intuitiva sobre qué gastos recurrentes te cobran?	Y. Díaz	Si
	A. Zhou	Sí, aunque más que avisar, quizás un resumen de los gastos recurrentes.
¿Haces resúmenes, balances o análisis de tus gastos? ¿Cómo los llevas a cabo?	Y. Díaz	Solo calculo lo que me debería quedar a fin de mes
	A. Zhou	Generalmente no, a veces algún que otro cálculo mental, pero generalmente no.
¿Estableces objetivos de ahorro? ¿En qué te basas?	Y. Díaz	si, me baso en una cantidad al mes que quiero ahorrar
	A. Zhou	Sí pero, al ser una persona dependiente económicamente casi al 100%, el ahorro no se basa tanto en ingresos para gastos sino en "regalos".
En tus gestiones sobre gastos recurrentes, ¿qué parte te parece más tediosa?	Y. Díaz	suscripciones
	A. Zhou	Hacer la transferencia de la comunidad y el agua. La comunidad es mensual y el agua bimestral. Al ir ambas mezcladas en el mismo pago, no se puede programar un pago recurrente (al ser el agua variable).
¿Has probado aplicaciones u hojas de cálculo para llevar tus gastos personales? ¿Qué aspecto te hizo decantarte por no usarlas?	Y. Díaz	Si, me deprimó
	A. Zhou	Fintonic. La he desinstalado porque con el banco N26 requería conceder permisos en la aplicación del banco cada vez que se quería refrescar la página. Además, dado que los ingresos también son variables (en función de la necesidad del gasto) se complicaba la previsión de gastos estimados en función de ingresos (es decir has gastado el X% de tus ingresos).
¿Qué otros aspectos buscarías en una aplicación para gestionar tus gastos personales?	Y. Díaz	Alertas, usabilidad, buena visualización de datos
	A. Zhou	Una aplicación simple, que te notificase de las cosas importantes de una manera clara (sin aturdirte con notificaciones) y que generase por ejemplo resúmenes mensuales de gasto)
¿Te gustaría que la aplicación pudiera conectarse a tu banco y añadir movimientos de manera dinámica?	Y. Díaz	si
	A. Zhou	Sí.
¿Te gustaría que la aplicación pudiera enviarte notificaciones para estar al tanto de tus gastos? Si puedes, detalla cómo.	Y. Díaz	sisisi, limites, objetivos de ahorro
	A. Zhou	Sí, como en la pregunta 2.10





































Parte 3: evaluación de gastos (común a todas las personas tipo)**Aviso: las respuestas a la pregunta sobre detalles de gastos personales se encuentran en las siguientes páginas, por motivos de espacio.**

Pregunta	Persona encuestada	Respuesta
Comentarios	N. Hervás	-
	A. Leiva	La cuota de autonomos es un gasto fijo anualmente hablando pero cambia de año en año. Las compras de material depende un poco del trimestre pero siempre compro cosas que necesito, a veces es mas caro y otras menos. Las dietas tambien dependen de los días que trabaje, pero la mayoría de ellos como fuera, así que supone un gasto importante.
	J. Ferrer	-
	H. Marco	Ya detallé al principio las categorías de gastos, así como si estos son fijos o variables, en la pregunta "1.1.1. En general, actualmente, ¿a qué tipo de gastos tienes que hacer frente?".
	P. Bonet	Con respecto a los gastos de IKEA (CaixaBank P&C) y Apple (Cetelem), aunque los recibos me los cargan juntos, en mi hoja de cálculo los tengo desglosados según la compra a la que corresponden, para poder saber cuánto llevo pagado y cuánto me queda de cada cosa. Por ejemplo, si de CaixaBank P&C me pasan un recibo de 100 €, en mi hoja tengo apuntado que 25 € corresponden a una compra en IKEA, otros 25 € a otra, y los 50 € restantes a una última compra. Tengo una hoja adicional en el Excel en la que hago seguimiento de las financiaciones que tengo, su total, su cuota, lo pagado y lo pendiente.
	Y. Díaz	-
	A. Zhou	Los gastos que aparecen sin rellenar o bien no dispongo de ellos o bien no son sufragados desde mis cuentas bancarias.
Si tienes algún gasto domiciliado, ¿tu banco te notifica cuando se produce el cobro? ¿Por qué medio lo hace (app, email...)?	N. Hervás	Cuando la UCM me cobra la matricula o me ingresan la nomina el banco me manda una notificación push con un mensaje similar a "Se ha ingresado su nomina en la cuenta *****" o "Se ha realizado un cobro de XXX,XX€ en su cuenta *****"
	A. Leiva	Sí, lo hace a través de la aplicación del propio banco
	J. Ferrer	Sí, a través de la App
	H. Marco	Sí, me envían una notificación push unos días previos avisando de que se ha girado un recibo con una determinada cuantía de dinero y la entidad o empresa que lo ha hecho.
	P. Bonet	Sí, me notifica por app móvil y por email.
	Y. Díaz	No
	A. Zhou	N26 avisa por una notificación del movil el importe y la fecha de una domiciliación futura. Abanca no avisa directamente, mas en la app hay un apartado donde se pueden consultar las domiciliaciones.
¿Qué entidades te notifican cuando se va a producir un cobro? ¿Por qué medio lo hacen (app, email...)?	N. Hervás	Apple manda un email para el cobro de la subscripción
	A. Leiva	Principalmente todas las de los suministros a través de emails
	J. Ferrer	Iberdrola + Apple + Aigües d'Elx, a través de email
	H. Marco	Ninguna de las que actualmente tengo gastos domiciliados con ellas.
	P. Bonet	Me notifican Endesa, Google, Amazon y Apple, por email.
	Y. Díaz	Endesa, Netflix, Klarna (eMail)
En caso de tener varios gastos domiciliados, ¿dispones de una única cuenta bancaria o tienes varias para distintos propósitos?	N. Hervás	N/A
	A. Leiva	Tengo varias cuentas bancarias
	J. Ferrer	Una sola cuenta
	H. Marco	Tengo múltiples cuentas bancarias: ingresos y gastos (nómina, recibos domiciliados, pagos a la Administración y Tesorería pública), gastos personales (ocio, compras, así como cualquier otro gasto no fijo que sea personal), ahorros, gastos conjuntos de la vivienda (comida, mascotas, muebles, reparaciones, etc.) y otra para inversiones.
	P. Bonet	Dispongo de una cuenta general, otra como hucha para ahorros, y una tercera solo para mis gastos recurrentes. En esta última tengo domiciliados los gastos (tanto por domiciliación como tal como cargados a una tarjeta asociada a la cuenta).
	Y. Díaz	Solo una para mi y otra compartida con mi pareja.
A. Zhou	Tengo varias con varios propósitos, por ejemplo, N26 la utilizo solo para los gastos del domicilio.	




































Parte 4: agradecimiento final

Pregunta	Persona encuestada	Respuesta
¿Te gustaría conocer el resultado final del proyecto?	N. Hervás	Sí
	A. Leiva	Sí
	J. Ferrer	Sí
	H. Marco	Sí
	P. Bonet	Sí
	Y. Díaz	Sí
A. Zhou	Sí	

Parte 3: evaluación de gastos (común a todas las personas tipo)
3.1. Concretamente, ¿a qué gastos haces frente y cuáles son sus características?

Persona encuestada	Categoría general	Categoría específica	Concepto o nombre del servicio	Empresa o entidad	Importe fijo o variable	Domiciliado	Frecuencia	Día de cobro
Hogar								
N. Hervás	Hogar	 Alquiler	Renta de alquiler	Particular	Fijo	No	Mensual	1
P. Bonet	Hogar	 Alquiler	Renta de alquiler	Particular	Fijo	No	Mensual	1
H. Marco	Hogar	 Alquiler	Renta de alquiler	Particular	Fijo	No	Mensual	1-7
A. Leiva	Hogar	 Alquiler	Renta de alquiler	Alquiler Seguro	Fijo	Sí	Mensual	Último día
Y. Díaz	Hogar	 Alquiler	Renta de alquiler	Particular	Fijo	Sí	Mensual	
A. Zhou	Hogar	 Alquiler	Renta de alquiler	Comunidad	Fijo	No	Mensual	3-27
N. Hervás	Hogar	 Agua	Consumo de agua	Canal de Isabel II	Variable	Sí	Bimensual	Mediados de mes
P. Bonet	Hogar	 Agua	Consumo de agua	Canal de Isabel II	Variable	Sí	Bimensual	1-5
H. Marco	Hogar	 Agua	Consumo de agua	Aigües d'Elx	Variable	Sí	Trimestral	5
A. Leiva	Hogar	 Agua	Consumo de agua	Ista	Variable	No	Bimensual	
Y. Díaz	Hogar	 Agua	Consumo de agua	Aguas de Zaragoza	Fijo	Sí	Mensual	
A. Zhou	Hogar	 Agua	Consumo de agua	Ista	Variable	No	Bimestral	3-27
N. Hervás	Hogar	 Luz	Consumo de electricidad	EDP	Variable	Sí	Mensual	
P. Bonet	Hogar	 Luz	Consumo de electricidad	Repsol	Variable	Sí	Mensual	1-5
H. Marco	Hogar	 Luz	Consumo de electricidad	Iberdrola	Variable	Sí	Mensual	5
A. Leiva	Hogar	 Luz	Consumo de electricidad	Endesa	Variable	Sí	Mensual	
Y. Díaz	Hogar	 Luz	Consumo de electricidad	Endesa	Fijo	Sí	Mensual	
A. Zhou	Hogar	 Luz	Consumo de electricidad	Iberdrola	Variable	Sí	Mensual	25
N. Hervás	Hogar	 Gas	Consumo de gas	EDP	Variable	Sí	Mensual	
P. Bonet	Hogar	 Gas	Consumo de gas	Naturgy, comercializadora regulada	Variable	Sí	Mensual	5
A. Leiva	Hogar	 Gas	Consumo de gas	Curenergía	Fijo	Sí	Mensual	2
Y. Díaz	Hogar	 Gas	Consumo de gas	Iberdrola	Fijo	Sí	Mensual	
A. Zhou	Hogar	 Gas	Consumo de gas	Repsol	Fijo	No	N/A	
P. Bonet	Hogar	 Telefonía	Telefonía	Digi	Fijo	Sí	Mensual	4
H. Marco	Hogar	 Telefonía	Telefonía	Movistar	Fijo	Sí	Mensual	1
J. Ferrer	Hogar	 Telefonía	Telefonía	Yoigo	Fijo	Sí	Mensual	1-5
A. Leiva	Hogar	 Telefonía	Telefonía	Vodafone	Fijo	Sí	Mensual	
N. Hervás	Hogar	 Internet	Internet	Movistar	Fijo	Sí	Mensual	
P. Bonet	Hogar	 Internet	Internet	Digi	Fijo	Sí	Mensual	4
H. Marco	Hogar	 Internet	Internet	Movistar	Fijo	Sí	Mensual	1
J. Ferrer	Hogar	 Internet	Internet	Yoigo	Fijo	Sí	mensual	1-5
A. Leiva	Hogar	 Internet	Internet	Digi	Fijo	Sí	Mensual	Último día
Y. Díaz	Hogar	 Internet	Internet	Digi	Fijo	Sí	Mensual	
A. Zhou	Hogar	 Internet	Internet	Jazztel	Fijo	Sí	Mensual	1
A. Zhou	Hogar	 Alarma	Alarma de hogar	Ring	Fijo	Sí	Anual	15 mayo
P. Bonet	Hogar	 Suministros básicos	Compras corrientes	Carrefour, Mercadona y Lidl	Variable	No	Semanal	N/A

Parte 3: evaluación de gastos (común a todas las personas tipo)
3.1. Concretamente, ¿a qué gastos haces frente y cuáles son sus características?

Persona encuestada	Categoría general	Categoría específica	Concepto o nombre del servicio	Empresa o entidad	Importe fijo o variable	Domiciliado	Frecuencia	Día de cobro
Suscripciones a servicios digitales								
N. Hervás	Suscripciones a servicios digitales	 Almacenamiento en la nube	iCloud+ 50 GB	Apple	Fijo	Sí	Mensual	Mediados de mes
H. Marco	Suscripciones a servicios digitales	 Almacenamiento en la nube	iCloud+ 200 GB	Apple	Fijo	Sí	Mensual	28
J. Ferrer	Suscripciones a servicios digitales	 Almacenamiento en la nube	iCloud+ 200 GB	Apple	Fijo	Sí	Mensual	
Y. Díaz	Suscripciones a servicios digitales	 Almacenamiento en la nube	Mega Pro I	Mega	Fijo	Sí	Mensual	
A. Zhou	Suscripciones a servicios digitales	 Almacenamiento en la nube	iCloud+ 200 GB	Apple	Fijo	Sí	Mensual	
P. Bonet	Suscripciones a servicios digitales	 Streaming de vídeo	Disney+	Disney+	Fijo	Sí	Anual	
Y. Díaz	Suscripciones a servicios digitales	 Streaming de vídeo	Disney+	Disney+	Fijo	Sí	Mensual	
H. Marco	Suscripciones a servicios digitales	 Streaming de vídeo	Amazon Prime Video	Amazon	Fijo	Sí	Mensual	15
J. Ferrer	Suscripciones a servicios digitales	 Streaming de vídeo	Amazon Prime Video	Amazon	Fijo	Sí	Anual	15 de abril
A. Zhou	Suscripciones a servicios digitales	 Streaming de vídeo	Amazon Prime Video	Amazon	Fijo	Sí	Mensual	
Y. Díaz	Suscripciones a servicios digitales	 Streaming de vídeo	Amazon Prime Video	Amazon	Fijo	Sí	Mensual	
Y. Díaz	Suscripciones a servicios digitales	 Streaming de vídeo	Amazon Prime Video	Amazon	Fijo	Sí	Mensual	
H. Marco	Suscripciones a servicios digitales	 Streaming de vídeo	Netflix	Netflix	Fijo	Sí	Mensual	15
A. Zhou	Suscripciones a servicios digitales	 Streaming de vídeo	Netflix	Netflix	Fijo	Sí	Mensual	
Y. Díaz	Suscripciones a servicios digitales	 Streaming de vídeo	Netflix	Netflix	Fijo	Sí	Mensual	
A. Leiva	Suscripciones a servicios digitales	 Streaming de vídeo	YouTube Premium	Google	Fijo	Sí	Mensual	12
A. Zhou	Suscripciones a servicios digitales	 Streaming de vídeo	Atresplayer	Atresmedia	Fijo	Sí	Mensual	
Y. Díaz	Suscripciones a servicios digitales	 Streaming de vídeo	HBO Max	HBO	Fijo	Sí	Mensual	
P. Bonet	Suscripciones a servicios digitales	 Streaming de música	Spotify Premium	Spotify	Fijo	Sí	Mensual	28-29
H. Marco	Suscripciones a servicios digitales	 Streaming de música	Spotify Premium	Spotify	Fijo	Sí	Mensual	12
A. Leiva	Suscripciones a servicios digitales	 Streaming de música	Spotify Premium	Spotify	Fijo	Sí	Mensual	23
Y. Díaz	Suscripciones a servicios digitales	 Streaming de música	Spotify Premium	Spotify	Fijo	Sí	Mensual	
A. Zhou	Suscripciones a servicios digitales	 Streaming de música	Spotify Premium	Spotify	Fijo	Sí	Mensual	
A. Leiva	Suscripciones a servicios digitales	 Cuentas digitales premium	Google One	Google	Fijo	Sí	Mensual	4
Y. Díaz	Suscripciones a servicios digitales	 Cuentas digitales premium	Google One	Google	Fijo	Sí	Mensual	
Y. Díaz	Suscripciones a servicios digitales	 Cuentas digitales premium	Apple One	Apple	Fijo	Sí	Mensual	
A. Zhou	Suscripciones a servicios digitales	 Otras suscripciones	Notability	Notability	Fijo	Sí	Anual	Diciembre
P. Bonet	Suscripciones a servicios digitales	 Otras suscripciones	Taxdown	Taxdown	Fijo	Sí	Anual	
Suscripciones a servicios físicos								
P. Bonet	Suscripciones a servicios físicos	 Compras y envíos	Amazon Prime	Amazon	Fijo	Sí	Anual	
H. Marco	Suscripciones a servicios físicos	 Compras y envíos	Amazon Prime	Amazon	Fijo	Sí	Mensual	15
A. Leiva	Suscripciones a servicios físicos	 Compras y envíos	Amazon Prime	Amazon	Fijo	Sí	Anual	
A. Zhou	Suscripciones a servicios físicos	 Compras y envíos	Amazon Prime	Amazon	Fijo	Sí	Anual	21 septiembre
H. Marco	Suscripciones a servicios físicos	 Transporte	Parking empleados aeropuerto	AENA	Variable	No	Bimensual	
A. Leiva	Suscripciones a servicios físicos	 Transporte	Uber Pass	Uber	Fijo	Sí	Mensual	
A. Leiva	Suscripciones a servicios físicos	 Comida a domicilio	Wetaca	Wetaca	Variable	Sí	Semanal	Jueves

Parte 3: evaluación de gastos (común a todas las personas tipo)

3.1. Concretamente, ¿a qué gastos haces frente y cuáles son sus características?

Persona encuestada	Categoría general	Categoría específica	Concepto o nombre del servicio	Empresa o entidad	Importe fijo o variable	Domiciliado	Frecuencia	Día de cobro
Financiación								
P. Bonet	Financiación	🛒 Compras financiadas	Compra financiada de coche	Cetelem	Fijo	Sí	Mensual	1-10
J. Ferrer	Financiación	🛒 Compras financiadas	Compra financiada de coche	Skoda	Fijo	Sí	Mensual	15
P. Bonet	Financiación	🛒 Compras financiadas	Compra financiada de colchón	Carrefour	Fijo	Sí	Mensual	1
A. Leiva	Financiación	🛒 Compras financiadas	Compra financiada en Apple (varios)	Cetelem	Fijo	Sí	Mensual	5
A. Leiva	Financiación	🛒 Compras financiadas	Compra financiada en IKEA (varios)	CaixaBank Payments & Consumer	Fijo	Sí	Mensual	Último día
Seguros								
P. Bonet	Seguros	💡 Seguros de objetos	Seguro de coche	Mutua	Fijo	Sí	Mensual	15
A. Leiva	Seguros	💡 Seguros de objetos	Seguro de ordenador	Rosellimac (WertGarantie)	Fijo	Sí	Mensual	15
P. Bonet	Seguros	👤 Seguros profesionales	Seguro de baja de autónomos	Previsión Mallorquina	Fijo	Sí	Mensual	15
P. Bonet	Seguros	👤 Seguros profesionales	Seguro de responsabilidad civil	Plus Ultra	Fijo	Sí	Anual	
P. Bonet	Seguros	❤️ Seguros personales	Seguro de vida	Mutua	Fijo	Sí	Mensual	15
Salud y bienestar								
H. Marco	Salud y bienestar	🏋️ Gimnasio		Freeletics	Fijo	Sí	Trimestral	2
Y. Díaz	Salud y bienestar	🏋️ Gimnasio		McFit	Fijo	Sí	Mensual	
H. Marco	Salud y bienestar	🧠 Psicólogo		I. Soriano	Fijo	No	Bono 5 sesiones	
A. Leiva	Salud y bienestar	🧠 Psicólogo		Particular	Fijo	No	Semanal	Jueves
J. Ferrer	Salud y bienestar	💊 Medicamentos		Farmacia	Variable	No	N/A	
Trabajo								
P. Bonet	Trabajo	👤 Gastos profesionales	Cuota de autónomos	Tesorería General de la Seguridad Social	Fijo	Sí	Mensual	28-29
P. Bonet	Trabajo	👤 Gastos profesionales	Pago IVA	Agencia Estatal de Administración Tributaria	Variable	Sí	Trimestral	20
P. Bonet	Trabajo	👤 Gastos profesionales	Pago IRPF	Agencia Estatal de Administración Tributaria	Variable	Sí	Trimestral	20
P. Bonet	Trabajo	👤 Gastos profesionales	Aplicacion de gestoría	Contasimple	Fijo	Sí	Anual	N/A
P. Bonet	Trabajo	👤 Gastos profesionales	Compras de material	Amazon principalmente	Variable	No	N/A	N/A
P. Bonet	Trabajo	👤 Gastos profesionales	Dietas	Varios restaurantes	Variable	No	N/A	N/A
P. Bonet	Trabajo	👤 Gastos profesionales	SPA	Quiron Prevencion	Fijo	Sí	Anual	N/A
Otros gastos recurrentes								
N. Hervás	Otros gastos recurrentes	🚗 Abono de transportes		Consortio Regional de Transportes de Madrid	Fijo	No	Mensual	N/A
P. Bonet	Otros gastos recurrentes	🚗 Abono de transportes		Consortio Regional de Transportes de Madrid	Fijo	No	Mensual	N/A
A. Leiva	Otros gastos recurrentes	🚗 Abono de transportes		Consortio Regional de Transportes de Madrid	Fijo	No	Mensual	N/A
A. Zhou	Otros gastos recurrentes	🚗 Abono de transportes		Consortio Regional de Transportes de Madrid	Fijo	No	Mensual	N/A

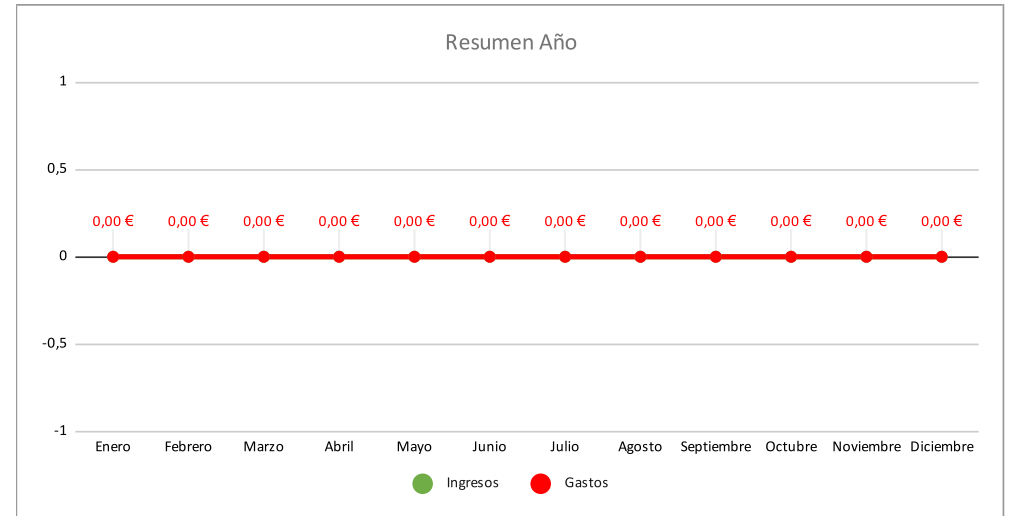
Apéndice D

Hojas de cálculo proporcionadas por las personas encuestadas

										Movimientos consolidados							
Pendiente actual		795,99 €								junio 2023				mayo 2023			
Grupo	Entidad	Concepto	Importe	Total grupo	Tipo	Domicilia	Prever	Día de cobro	Periodo	Fecha	Pagado	Pendiente fijo	795,99 €	Fecha	Pagado	Pendiente fijo	0,00 €
												Final real	Pendiente fijo			Final real	Pendiente fijo
Alquiler			-700,00 €														
	Alquiler Seguro	Alquiler	-700,00 €		Fijo	Sí	Sí	Último día del mes	Mensual			-700,00 €	-700,00 €	(...)	Sí	-700,00 €	0,00 €
Hogar			-201,00 €														
	Digi	Internet fibra Digi	-21,00 €		Fijo	Sí	Sí	?	Mensual			-21,00 €		(...)	Sí	-21,00 €	0,00 €
	Curenergía	Gas	-40,00 €		Fijo	Sí	Sí	2 de cada mes	Mensual	(...)	Sí	-40,00 €	0,00 €	(...)	Sí	-40,00 €	0,00 €
	Endesa	Electricidad	-100,00 €		Variable	Sí	Sí	?	Mensual	(...)		-30,00 €	-30,00 €	(...)	Sí	-20,00 €	0,00 €
	Ista	Agua	-40,00 €		Variable	No	Sí	?	Bimensual			-40,00 €					
Suscripciones			-6,98 €														
	Google	Google One	-1,99 €		Fijo	Sí	Sí	4 de cada mes	Mensual	(...)	Sí	-1,99 €	0,00 €	(...)	Sí	-1,99 €	0,00 €
	Spotify	Spotify Premium	-4,99 €		Fijo	Sí	Sí	?	Mensual	(...)		-4,99 €	-4,99 €	(...)	Sí	-4,99 €	0,00 €
Financiación			-200,00 €														
	Cetelem	Ordenador portátil	-200,00 €		Fijo	Sí	Sí	5 de cada mes	Mensual	(...)	Sí	-200,00 €	0,00 €				
Otros recibos			-27,00 €														
	Rosellimac	Seguro portátil	-27,00 €		Fijo	Sí	Sí	15 de cada mes	Mensual	(...)	Sí	-27,00 €	0,00 €				
	UCM	UCM	-100,00 €		Fijo	Sí	No	10 de cada mes	Mensual					(...)	Sí	-100,00 €	0,00 €
Psicólogo			-70,00 €														
		Psicólogo semana 1	-70,00 €		Fijo	No	Sí			(...)	Sí	-70,00 €	0,00 €	(...)	Sí	-70,00 €	0,00 €
Otros previstos			-50,00 €														
		Wetaca	-50,00 €			Sí	Sí			(...)	Sí	-40,00 €	0,00 €	(...)	Sí	-40,00 €	0,00 €
Total global gastos previsibles/fijos				-1.254,98 €													

	YEAR 2019												TOTAL ANNUAL	ANNUAL PERCENTAGE
	JANUARY	FEBRUARY	MARCH	APRIL	MAY	JUNE	JULY	AUGUST	SEPTEMBER	OCTOBER	NOVEMBER	DECEMBER		
FIXED COSTS	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	
Phone														0,00 €
Gym														0,00 €
F. C. - Others														0,00 €
FOOD	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	
Supermarket														0,00 €
Takeaway														0,00 €
Restaurants														0,00 €
Bars														0,00 €
HOME	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	
Furniture														0,00 €
Electronics & Informatics														0,00 €
Home appliances														0,00 €
CLOTHING	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	
Dress clothes														0,00 €
SUPERMARKET	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	
Personal care														0,00 €
S. - Others.														0,00 €
HEALTH	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	
H. - Others														0,00 €
														0,00%
TRANSPORT	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	
Gas														0,00 €
Transport														0,00 €
T. - Others														0,00 €
PETS	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	
P. - Others														0,00 €
														0,00%
FORMATION	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	
Master														0,00 €
Courses														0,00 €
Congresses, seminars...														0,00 €
Photocopies														0,00 €
Books														0,00 €
ENTERTAINMENTS	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	
Videogames														0,00 €
Movie theater & theater														0,00 €
E. - Others														0,00 €

Mes	Ingresos	Gastos	Saldo Final
Enero	0,00 €	0,00 €	0,00 €
Febrero			
Marzo	0,00 €	0,00 €	0,00 €
Abril			
Mayo	0,00 €	0,00 €	0,00 €
Junio			
Julio	0,00 €	0,00 €	0,00 €
Agosto			
Septiembre	0,00 €	0,00 €	0,00 €
Octubre			
Noviembre	0,00 €	0,00 €	0,00 €
Diciembre			
Total	0,00 €	0,00 €	



	2022												Total Anual	
	Enero	Febrero	Marzo	Abril	Mayo	Junio	Julio	Agosto	Septiembre	Octubre	Noviembre	Diciembre		
Supermercado	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Comida	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Articulos Limpieza	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Carnicería	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Frescos	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
S - Otros	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Gastos_Fijos	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Agua	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Luz	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Gas	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Internet	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Teléfono	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Suscripciones	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A

Categorías

Ocio

Bizum

Compras

Facturas

Resturantes

Gasolina

Seguros

Seguridad SS

Impuestos

Formacion

Suscripciones

Coche

Material

Ropa

Gastos

Supermercado

Comida

Articulos Limpieza

Carnicería

Frescos

S - Otros

Gastos_Fijos

Agua

Luz

Gas

Internet

Teléfono

Suscripciones

G - Otros

Formación

Cursos

Material

Herramienta

Dietas

F - Otros

Ocio

Restaurantes

Fiestas

Compras

Vacaciones

Planes

O - Otros

Transporte

Gasolina

Mantenimiento Coche

Bus / Metro

Taxi

T - Otros

Vivienda

Muebles

Electrodomésticos

Reparaciones

Decoración

Ahorro

V - Otros

Salud

Farmacia

Fisio

Gimnasio

Sa - Otros

Seguros

Coche

Vida

Resp. Civil

Se - Otros

Impuestos

IVA

IRPF

Autónomos

I - Otros

Ingresos

Nóminas

Facturas

Bizum

Transferencias

