

Gestión de Finanzas Personales

**Grado en Ingería Informática de
Gestión y Sistemas de Información**

TRABAJO FIN DE GRADO **Memoria**



BILBOKO
INGENIARITZA
ESKOLA
ESCUELA
DE INGENIERÍA
DE BILBAO

Alumno: Cuesta Alario, David
Director: López Cuadrado, Javier
Curso: 2020 – 2021

Fecha: 2021-01-10

Resumen

Castellano

Esta aplicación permitirá a los usuarios realizar un seguimiento de su patrimonio con la finalidad de simplificar la gestión de sus finanzas personales e inversiones de forma conjunta.

Este proyecto pretende unificar dos herramientas ampliamente extendidas que habitualmente se presentan por separado.

Inglés

This application will allow users to track their assets in order to simplify the management of their personal finances and investments together.

This project aims to unify two widely used tools that are usually presented separately.

Valenciano

Aquesta aplicació permetrà als usuaris fer un seguiment del seu patrimoni amb la finalitat de simplificar la gestió de les seves finances personals i inversions de manera conjunta.

Aquest projecte pretén unificar dues eines àmpliament esteses que habitualment es presenten per separat.

Euskera

Aplikazio honi esker, erabiltzaileek beren aktiboen jarraipena egin ahal izango dute, beren finantza pertsonalen eta inbertsioen kudeaketa batera errazteko.

Proiektu honek bereizita aurkeztu ohi diren asko erabiltzen diren bi tresna bateratu nahi ditu.

Índice de contenido

MEMORIA	1
MOTIVACIÓN	1
ANTECEDENTES.....	2
<i>Finanzas personales</i>	2
<i>Gestión de activos</i>	9
OBJETIVOS.....	18
REQUISITOS	18
<i>Funcionales</i>	18
<i>No Funcionales</i>	19
ALCANCE	19
ESTADO DEL ARTE	20
DEFINICIONES.....	20
<i>Informáticas</i>	20
<i>Financieras</i>	20
CÁLCULOS	21
<i>Beneficio</i>	21
<i>Precio Medio</i>	21
<i>Rentabilidad Anualizada</i>	21
<i>Rentabilidad Ponderada por tiempo</i>	21
HERRAMIENTAS	22
<i>Técnicas</i>	22
<i>Lenguajes de programación</i>	25
<i>Librerías</i>	26
<i>Documentación</i>	28
<i>Programas</i>	29
<i>Servicios</i>	30
METODOLOGÍA	32
PLANIFICACIÓN	32
<i>Evaluación del Trabajo</i>	32
<i>Evaluación Temporal</i>	37
<i>Evaluación Económica</i>	39
<i>Evaluación de Riesgos</i>	42
DISEÑO	49
<i>Jerarquía de actores</i>	49
<i>Casos de Uso</i>	49
<i>Modelo de Dominio</i>	61
<i>Diseño de la interfaz</i>	64
<i>Diagrama Clases</i>	75
<i>Diagrama Secuencia</i>	80
<i>Estructura de los Directories</i>	83
CONCLUSIONES.....	85
RESULTADO	85

EVOLUCIÓN.....	85
DIFICULTADES	86
<i>Técnicas</i>	86
<i>Sociales</i>	87
<i>Diseño</i>	87
<i>Implementación</i>	88
<i>Despliegue</i>	88
APRENDIZAJE	89
TRABAJO FUTURO.....	90
TAREAS PENDIENTES.....	90
<i>Puesta en servicio</i>	90
<i>Pruebas de campo</i>	90
<i>Monetización</i>	90
MEJORAS	90
<i>Herramientas de Gestión</i>	90
<i>FeedBack</i>	90
<i>Tipos de Imputaciones</i>	91
<i>Cache permanente</i>	91
DESARROLLO FUTURO	91
<i>Informes fiscales</i>	91
<i>Automatización</i>	91
<i>Notificaciones</i>	92
<i>Estilo artístico</i>	92
<i>Gestión de Objetivos</i>	92
<i>Herramientas de Análisis</i>	92
BIBLIOGRAFÍA	93
TUTORIALES	93
REFERENCIAS.....	93

Índice de Ilustraciones

RESUMEN DE GOOGLE PLAY DE PRESUPUESTO RÁPIDO	3
VENTANA PRINCIPAL Y SUBMENÚS DE CONFIGURACIÓN EN PRESUPUESTO RÁPIDO.....	3
VENTANA DE TRANSACCIONES EN PRESUPUESTO RÁPIDO.....	4
PARTICULARIDADES DE LAS TRANSACCIONES EN PRESUPUESTO RÁPIDO	5
VENTANA DE DEUDAS EN PRESUPUESTO RÁPIDO	5
VENTANA DE GESTIÓN DE CATEGORÍAS EN PRESUPUESTO RÁPIDO	6
VENTANA DE GRÁFICOS EN PRESUPUESTO RÁPIDO	6
RESUMEN DE GOOGLE PLAY DE REGISTRO CONTABLE.....	7
VENTANA PRINCIPAL EN REGISTRO CONTABLE	7
DATOS EXPORTADOS A EXCEL DE REGISTRO CONTABLE	8
VENTANA DE TRANSACCIONES EN REGISTRO CONTABLE.....	8
VENTANA DE GRÁFICOS EN REGISTRO CONTABLE	8
PÁGINA DE BIENVENIDA DE MI CARTERA EN RANKIA.....	9
PÁGINAS DE REGISTRO E IDENTIFICACIÓN EN MI CARTERA EN RANKIA	10
PÁGINAS DE INICIO EN MI CARTERA EN RANKIA.....	10
VENTANA DE ADMINISTRACIÓN DE ACTIVOS Y TRANSACCIONES EN MI CARTERA EN RANKIA.....	11
VENTANA PARA AGREGAR ACTIVOS Y TRANSACCIONES EN MI CARTERA EN RANKIA	12
VENTANA PARA AGREGAR UNA NUEVA CARTERA EN MI CARTERA EN RANKIA.....	12
VENTANA DE RESÚMENES DE LA CARTERA EN MI CARTERA EN RANKIA	13
VENTANA DE RANKIN DE USUARIOS EN MI CARTERA EN RANKIA	14
VENTANA DE DESCARGA DE PORTFOLIO PERFORMANCE	14
CONTENEDORES DE INFORMACIÓN EN PORTFOLIO PERFORMANCE	14
TABLA DE TRANSACCIONES PARA LA CUENTA DE DEPÓSITO DE INDEXA CAPITAL EN PORTFOLIO PERFORMANCE	15
TABLA DE VALORES SUSCRITOS A LA CUENTA DE VALORES DE BOGLEHEAD EN PORTFOLIO PERFORMANCE	15
GRAFICO PRECIOS HISTÓRICOS EN PORTFOLIO PERFORMANCE	16
VENTANA ALL TRANSACTIONS EN PORTFOLIO PERFORMANCE	16
VENTANA RESUMEN DE ACTIVOS EN PORTFOLIO PERFORMANCE	16
VENTANA GRÁFICOS, GRAFICO DE ACTIVOS EN PORTFOLIO PERFORMANCE	17
VENTANA GRÁFICOS, GRAFICO DE EVOLUCIÓN TEMPORAL EN PORTFOLIO PERFORMANCE	17
ENCRIPCIÓN CON CLAVE SIMÉTRICA	22
ENCRIPCIÓN CON CLAVE ASIMÉTRICA	22
ALMACENAMIENTO DE CLAVES MEDIANTE HASH.....	23
IDENTIFICACIÓN DE CLAVES MEDIANTE HASH.....	24
ALMACENAMIENTO DE CLAVES MEDIANTE HASH + SAL	24
IDENTIFICACIÓN DE CLAVES MEDIANTE HASH.....	24
EJEMPLO DE UTILIZACIÓN DE CHARTS.JS	27
DIVISIÓN DE TAREAS – PARTE 1.....	32
DIVISIÓN DE TAREAS – PARTE 2	33
ESTRUCTURA DE REQUISITOS	36
ESTIMACIÓN DE DURACIONES.....	37
DIAGRAMA DE GANTT	38
EJEMPLO MONETIZACIÓN CON GOOGLE ADSENSE.....	39
FACTURA DEL EQUIPO INFORMÁTICO	40
JERARQUÍA DE ACTORES.....	49
CASOS DE USO DE CUALQUIERA	49
CASOS DE USO DEL USUARIO NO IDENTIFICADO	50

CASOS DE USO DEL USUARIO ADMINISTRADOR	51
CASOS DE USO DEL USUARIO IDENTIFICADO – AÑADIR, ELIMINAR, EDITAR Y FILTRAR INFORMACIÓN.....	53
CASOS DE USO DEL USUARIO IDENTIFICADO – ADMINISTRAR SESIÓN Y REALIZAR CONSULTAS.....	57
MODELO DE DOMINIO – USUARIOS	61
MODELO DE DOMINIO – TRANSACCIONES	61
MODELO DE DOMINIO – ACTIVOS, COMISIONES Y APORTACIONES	62
ESTRUCTURA GENERAL DE LA INTERFAZ	64
MENÚ GESTIONAR EXPORTACIONES E IMPORTACIONES.....	67
MENÚS DE REGISTRO, INICIO DE SESIÓN Y EDITAR REGISTRO	67
MENÚS DE ADMISIONES	68
MENÚ GESTIONAR	69
FORMULARIOS PARA AÑADIR Y EDITAR CUENTAS CATEGORÍAS ESTRATEGIAS O PRODUCTOS	70
FORMULARIOS PARA AÑADIR Y EDITAR APORTACIONES, COMISIONES, TRANSACCIONES O TRASPASOS.....	70
FORMULARIOS DE ACTIVOS, ALLOCATION Y CLASIFICACIONES.....	71
MENÚ GESTIONAR DISTRIBUCIONES	72
MENÚ GESTIONAR EXPORTACIONES E IMPORTACIONES.....	72
MENÚ DE EVOLUCIÓN TEMPORAL PARA GASTOS, INGRESOS Y ACTIVOS	73
MENÚS DE FLUJO DE CAJA	74
MENÚ DE PATRIMONIO PARA COMPOSICIONES Y DISTRIBUCIONES	74
DIAGRAMA DE CLASES – BASE DATOS – ADMISIÓN Y USUARIOS	75
DIAGRAMA DE CLASES – BASE DATOS – COMISIONES, APORTACIONES Y FINANZAS	76
DIAGRAMA DE CLASES – APLICACIÓN.....	77
DIAGRAMA DE FLUJO – REGISTRARSE	80
DIAGRAMA DE FLUJO – IDENTIFICARSE	81
DIAGRAMA DE FLUJO – GENERAR CONTENIDO	82
ESTRUCTURA DE LOS DIRECTORIOS.....	83

Memoria

Este apartado de la documentación está destinado a recopilar sus orígenes. De este modo se analizarán las causas que llevaron a su ejecución, la situación inicial de la que se parte y los objetivos que se pretenden alcanzar.

Motivación

Este proyecto surge de mí, ya no tan reciente, primer contacto con el mundo de la gestión de la economía personal.

Rondaba el año 2015 cuando, por una serie de afortunadas coincidencias comencé a darme cuenta de lo importante que es realmente llevar un control exhaustivo del dinero que pasa por nuestras manos.

La mayoría de las personas ven el dinero como un tema tabú, del que no se puede hablar. No obstante, las personas más ricas y poderosas del mundo consideran que no es más que un juego.

Si bien es cierto que el dinero solo es una ilusión, una invención humana basada en la confianza en el sistema con la única finalidad de simplificar las transacciones comerciales, asumir que no es importante sería un error.

Para comprender realmente la naturaleza del dinero, la definición más acertada sería compararlo con una herramienta, como un martillo. Como cualquier otra herramienta se podría utilizar para hacer cosas buenas, como clavar clavos, o cosas malas, como partir cráneos. La diferencia entre el dinero y cualquier otra herramienta es su versatilidad, de ahí su mayor potencial para construir o destruir.

El dinero es una herramienta que tiene la capacidad de imitar las cualidades de cualquier otra herramienta, e incluso de proporcionarnos habilidades de las que de otro modo nunca dispondríamos. Por ejemplo, si no sabes nada de carpintería, el dinero te da la capacidad de contratar a un carpintero concediéndote su experiencia y capacidades de forma temporal.

Muchos son los ejemplos que se podrían poner acerca de cómo el dinero ha arruinado la vida de personas. Como famosos que se dieron a las drogas o personas a las que les toca la lotería y termina más endeudadas de lo que estaban al principio.

Pero la solución no pasa por ignorar el problema, sino por aprender a gestionarlo.

Lo que sucede es que las herramientas con mayor potencial son más difíciles de utilizar correctamente. Combinado con la gran cantidad de desinformación que existe al respecto y la falta de educación financiera, hacen que tener dinero sea la receta perfecta para el desastre.

De este modo, mi propuesta para este proyecto es crear una herramienta, lo más sencilla posible, que permita a cualquiera con pocos conocimientos financieros realizar un análisis exhaustivo de su situación económica personal.

Porque el primer paso para efectuar un buen análisis es tener un buen sistema de control que permita visualizar todas las variables que influyen de la forma más simple posible.

Antecedentes

Mi primer contacto con el mundo de la gestión de finanzas personales fue crear plantillas Excel, cada vez más complejas, y que realizaban las funciones necesarias. Mentiría si dijera que con estas plantillas no era suficiente para llevar a cabo un control patrimonial a nivel individual. No obstante, debido a mis reiterados fracasos porque gente de mi entorno utilizara dichas plantillas para analizar su situación financiera, comprendí que no todo el mundo tiene la motivación necesaria para utilizar un sistema que, si no se sabe utilizar, puede resultar bastante complejo.

A día de hoy existen muchas aplicaciones que realizan estas funcionalidades. No obstante:

- Las mejores herramientas son de pago, y como muchos se estarán imaginando, tener que pagar por este servicio supone una barrera de entrada tan grande como el hecho de que sea demasiado complejo.
- Entre las que son de uso gratuito, no he encontrado ninguna que ofrezca simultáneamente todas las funcionalidades que considero necesarias.

En función de las funcionalidades que estén presentes podemos diferenciar claramente entre dos tipos de aplicaciones.

A continuación, describiré detalladamente el funcionamiento de las características más relevantes de las que, en mi opinión, son las mejores aplicaciones para realizar estas funcionalidades de forma independiente.

Finanzas personales

Este tipo de aplicaciones se centran en analizar el flujo de caja del usuario.

El usuario introduce sus datos de gastos e ingresos, generalmente distribuidos en un conjunto de categorías y subcategorías fácilmente personalizables.

La aplicación utiliza estos datos para calcular un conjunto de indicadores que organiza gráficas y tablas para mostrárselos al usuario.

Cabe destacar que existe una gran cantidad de aplicaciones de uso gratuito que cubren estas funcionalidades. A pesar de que he analizado un gran número de ellas, con la finalidad de reunir las características más relevantes, me he centrado en las aplicaciones para Android que contaban con una mayor valoración por parte de los usuarios.

Las características que se repiten con mayor frecuencia son:

- Posibilidad de anotar transacciones de gastos e ingresos distribuidos en categorías.
- Posibilidad de disponer de varias cuentas independientes.
- Posibilidad de crear categorías personalizadas.
- Posibilidad de visualizar los datos introducidos mediante listas ordenadas y graficas con diferentes periodos.

Otras características adicionales son:

- Disponer de un acceso rápido para introducir las transacciones más habituales.
- Permitir la programación periódica de transacciones.
- Descargar los datos a un fichero con formato editable.

Características que me parecen especialmente útiles, pero no he encontrado:

- Categorías comunes para gastos e ingresos.
- Importar los datos desde un fichero con formato editable.

Presupuesto Rápido



Presupuesto Rápido - Gestor de Gastos

Info. de la app



Administre sus finanzas personales y controle fácilmente su dinero y sus gastos

Finanzas

4,6 ★

91 mil reseñas

+1 M

Descargas

3

PEGI 3

Valoraciones y opiniones



4,6

★★★★★

91.139

5

4

3

2

1

Ilustración 1 Resumen de Google Play de Presupuesto Rápido

Lo primero que encontramos al acceder a la aplicación es conjunto secciones fácilmente personalizables donde se muestran algunos de los resúmenes representativos. Podemos encontrar:

Vista General

- Resumen**: Total (Ingresos: 500,00 €, Gastos: -98,00 €, Total: 402,00 €); Últimos 7 días (Ingresos: 200,00 €, Gastos: -28,00 €, Total: 172,00 €).
- Cuentas**: Cartera (Última vez usado: 01/01/2021, 200,00 €); Personal (Última vez usado: 08/01/2021, 202,00 €).
- Gastos - Ingresos Últimos 7 días**: Gráfico de barras que muestra ingresos (rojo) y gastos (verde) por día.

Balance: Cantidad total que posee: 402,00 €. Gráfico de líneas que muestra el balance diario.

Presupuestos: Todas las categorías: Cartera (01/01/2021, 0,00 €), Bares (07/01/2021, -3,00 €), Deportes (05/01/2021, -15,00 €), Salario (01/01/2021, 200,00 €), Transportes (01/01/2021, -10,00 €), Ingresos financieros (01/12/2020, 300,00 €).

Transacciones: Viajes (01/01/2021, -10,00 €), Bares (07/01/2021, -3,00 €), Deportes (05/01/2021, -15,00 €), Salario (01/01/2021, 200,00 €), Transportes (01/01/2021, -10,00 €), Ingresos financieros (01/12/2020, 300,00 €).

RESUMEN: Número de ítems mostrados: 2. Seleccionar: Hoy, Últimos 7 días. Botones: CANCELAR, GUARDAR.

CUENTAS: Número de ítems mostrados: 2. Seleccionar: Cartera, Personal. Botones: CANCELAR, GUARDAR.

PRESUPUESTOS: Número de ítems mostrados: 2. Seleccionar: Todas las categorías - Presupuest., Viajes - Presupuesto mensual. Botones: CANCELAR, GUARDAR.

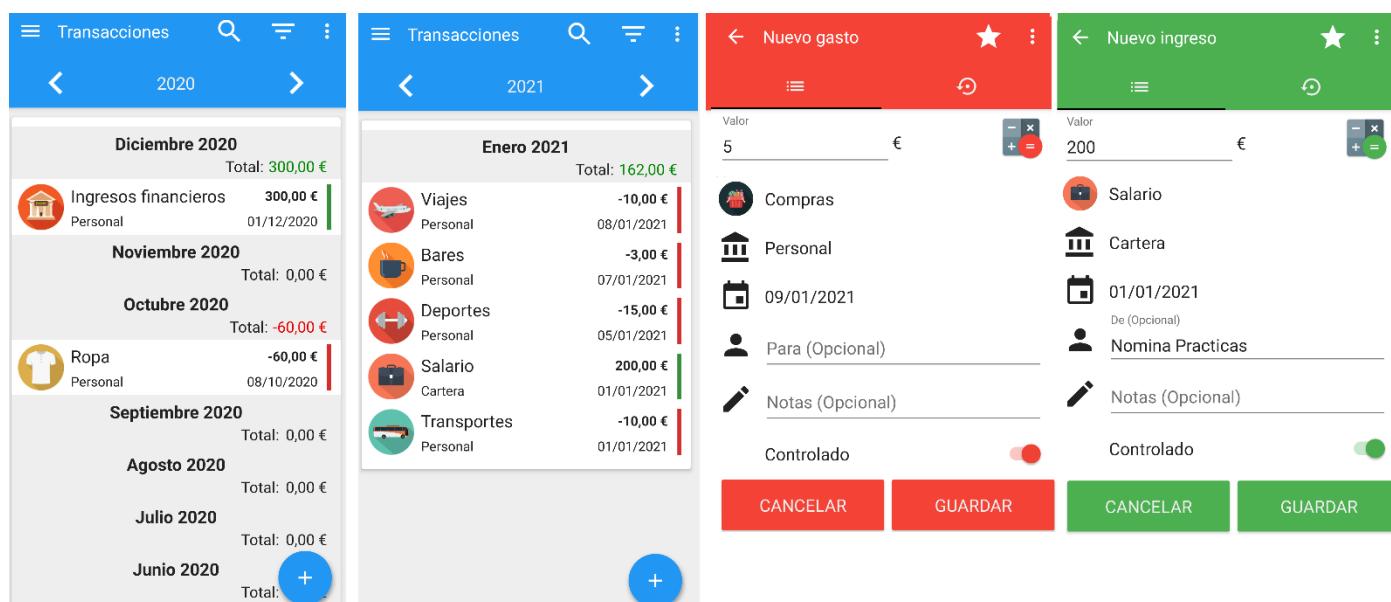
Ilustración 2 Ventana principal y submenús de configuración en Presupuesto Rápido

- La sección **Resumen** muestra la proporción de gastos frente a ingresos mediante un gráfico de sectores que se puede configurar para mostrar varios períodos de tiempo simultáneamente.
- La sección de **Cuentas** muestra el capital disponible en cada una de las cuentas que hallamos registrado.
- La sección de **Gastos – Ingresos** muestra la periodicidad y volumen de los gastos e ingresos mediante un gráfico de frecuencias que se puede configurar para mostrar varios períodos de tiempo.
- La sección de **Balance** muestra la variación del capital disponible en el tiempo mediante un gráfico de líneas por puntos que se puede configurar para mostrar una o varias cuentas simultáneamente en diversos períodos de tiempo.
- La sección de **Transacciones** muestra las transacciones más recientes.

Desde la ventana de transacciones

- Se pueden ver todos los gastos e ingresos de cada año agrupados por meses.
- Se permite añadir nuevos gastos e ingresos especificando:
 - El valor de la transacción.
 - La categoría a la que pertenece.
 - La cuenta desde la que se efectúa.
 - La fecha de la transacción.
 - Dos comentarios diferenciados opcionales.
 - Un Check de verificación que permite clasificar a una operación como no confirmada.

(No existe ninguna diferencia entre una operación confirmada y una operación no confirmada, salvo que la aplicación te recuerda mediante notificaciones acerca de aquellas operaciones pendientes de confirmar)



The image displays four screens from the Presupuesto Rápido application:

- Left Panel (Transactions View):** Shows transaction history from June 2020 to December 2020. Examples include "Ingresos financieros Personal" (300,00 €) and "Ropa Personal" (-60,00 €).
- Middle Panel (Transactions View):** Shows transaction history from July 2020 to December 2020. Examples include "Viajes Personal" (-10,00 €) and "Bares Personal" (-3,00 €).
- Right Panel (Nuevo gasto - New Expense):** A red-themed form for entering a new expense. It includes fields for "Valor" (Value) with a numeric input (5), a dropdown for "Categoría" (Category) with "Compras" selected, and a date input "09/01/2021". Buttons for "CANCELAR" and "GUARDAR" are at the bottom.
- Bottom Right Panel (Nuevo ingreso - New Income):** A green-themed form for entering a new income. It includes fields for "Valor" (Value) with a numeric input (200), a dropdown for "Cuenta" (Account) with "Salario" selected, and a date input "01/01/2021". Buttons for "CANCELAR" and "GUARDAR" are at the bottom.

Ilustración 3 Ventana de transacciones en Presupuesto Rápido

Algunas particularidades del sistema de inserción de transacciones son que se permite:

- Añadir gastos para una fecha que todavía no ha sucedido.
- Configurar transacciones que se repetirán un determinado número de veces cada cierto intervalo de tiempo.
- Señalar transacciones como favoritas para facilitar su inserción futura.

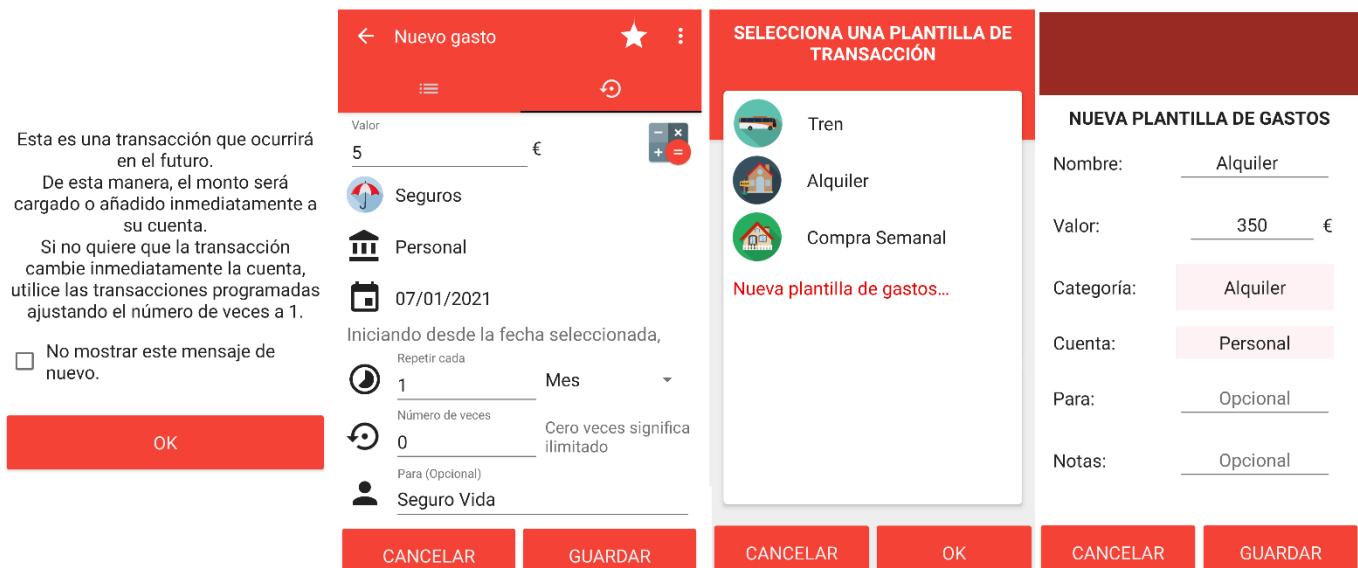


Ilustración 4 Particularidades de las transacciones en Presupuesto Rápido

Esta aplicación contempla las deudas como un tipo especial de transacciones:

- Estas transacciones quedan anotadas por separado salvo que se marque el Check para crear una transacción.
- Una vez confirmada la deuda, se crea un acceso rápido para su devolución total o parcial.

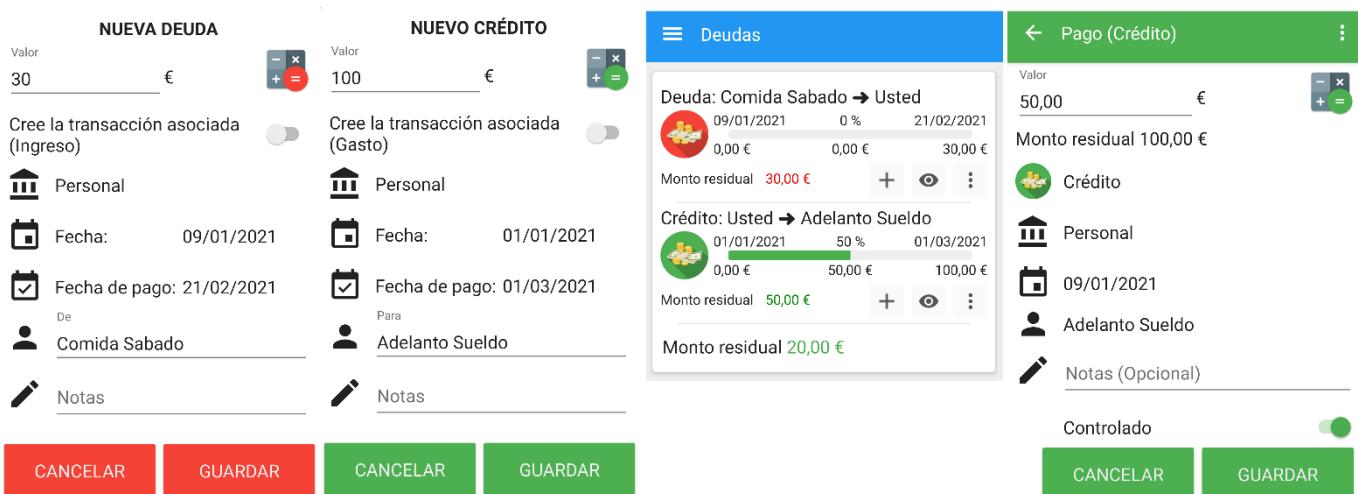


Ilustración 5 Ventana de deudas en Presupuesto Rápido

A pesar de que la aplicación ya cuenta con un gran número de categorías y subcategorías, es posible añadir nuevas desde la ventana de gestión de categorías. Algunas particularidades de este sistema son:

- Las categorías de gastos e ingresos están diferenciadas.
- Al crear nuevas categorías y subcategorías se puede seleccionar como logo alguna de las imágenes que ofrece la aplicación o cargar nuevas de la galería.

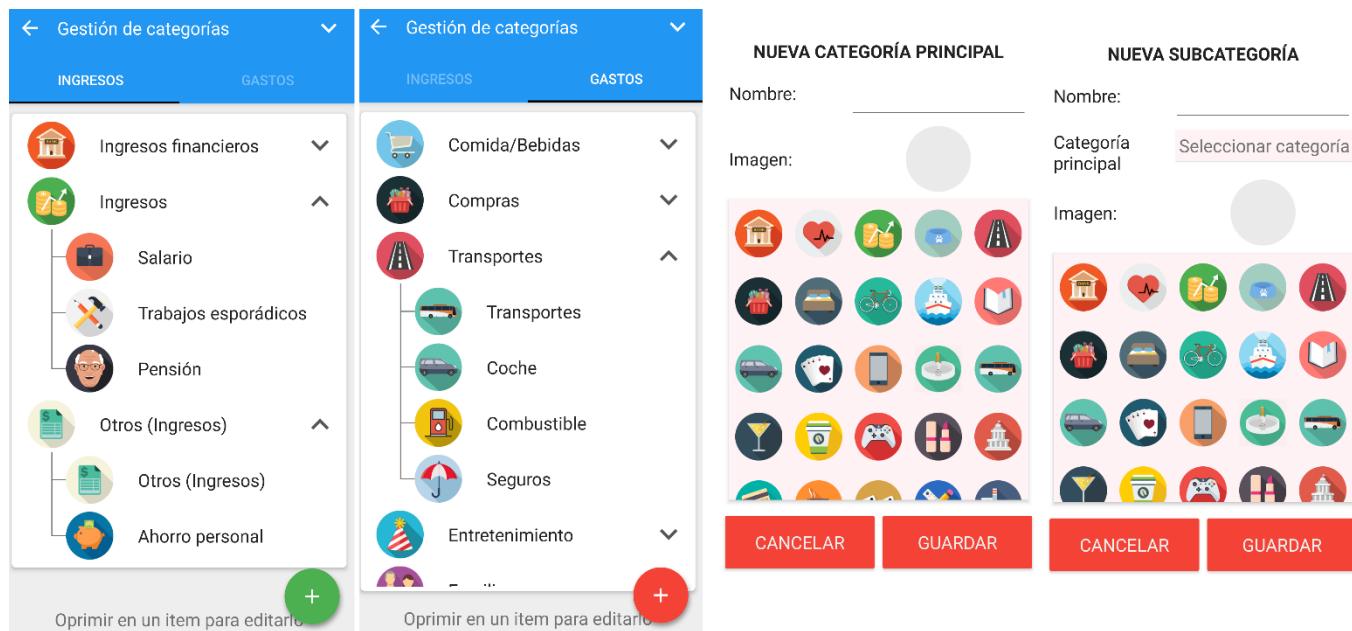


Ilustración 6 Ventana de gestión de categorías en Presupuesto Rápido

Desde la ventana de gráficos se pueden configurar una gran cantidad de resúmenes, más avanzados que los que se muestran en la ventana de inicio.



Ilustración 7 Ventana de gráficos en Presupuesto Rápido

Registro Contable



Registro Contable

Realbyte Inc.

Contiene anuncios •
Compras en la aplicación

Info. de la app



Registro Contable es una aplicación que le ayuda a controlar el dinero personal.

Finanzas

Play Pass

4,7 ★

258 mil reseñas

+10 M

Descargas

3

PEGI 3 ⓘ

Valoraciones y opiniones



4,7

★★★★★

258.936

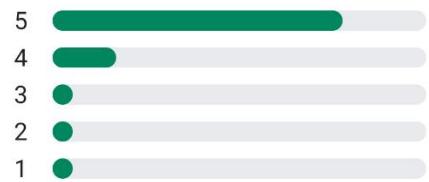


Ilustración 8 Resumen de Google Play de Registro Contable

Lo primero que encontramos al acceder a la aplicación es la ventana del balance, donde podemos ver todas las transacciones ordenadas por fecha en una lista o un calendario.

The screenshot displays the main interface of the Registro Contable app. It features three main sections:

- Left Panel (Diario):** Shows a list of transactions. The first few entries are:
 - 09 2021.01: Salario (€ 100,00) - Balance: € 215,00
 - 08 2021.01: Entretenimie... (€ 10,00) - Balance: € 205,00
 - 07 2021.01: Educación (€ 30,00) - Balance: € 175,00
 - 04 2021.01: Comida (€ 10,00) - Balance: € 165,00
- Middle Panel (Calendario):** A calendar for January 2021. The days of the week are labeled dom., lun., mar., mié., jue., vie., sáb. The dates are numbered 27 through 31, followed by 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 1.1, 2.1. Below the calendar, there are four icons: Trans., Estad., Cuentas, and Ajustes.
- Right Panel (Summary):** Displays a summary of expenses for the month. It includes a table with columns for Ingresos, Gastos, and Balance, and a breakdown by category like Transporte and Mantenimie... It also shows a comparison of last month's expenses (170%), a list of expenses (Gastos), and a button to export to Excel. At the bottom, there are four icons: Trans., Estad., Cuentas, and Ajustes.

Ilustración 9 Ventana principal en Registro Contable

Podemos exportar todos los datos a un fichero Excel (.xls)

A	B	C	D	E	F	G	H	I	J	K	
1	Fecha	Cuenta	Categoría	Subcategorías		Ingreso/Gasto		Nota	Importe	Moneda	Cuenta
2	2021/09/01	Cuentas	Salario			100	Ingreso		100	EUR	100
3	2021/08/01	Efectivo	Entrenamiento			10	Gasto		10	EUR	10
4	2021/07/01	Efectivo	Educación			30	Gasto	Matrícula uni	30	EUR	30
5	2021/07/01	Cuentas	Comida			10	Gasto	comida	10	EUR	10
6	2021/04/01	Efectivo	Productos de belleza	Champú		15	Gasto		15	EUR	15
7	2021/01/01	Efectivo	Regalos			20	Gasto	estrenos	20	EUR	20
8	2021/01/01	Efectivo	Salario			200	Ingreso		200	EUR	200
9	2020/01/12	Efectivo	Salud			50	Gasto		50	EUR	50

Ilustración 10 Datos exportados a Excel de Registro Contable

Se pueden añadir nuevas transacciones de gastos e ingresos especificando:

- La fecha de la transacción.
- La cuenta desde la que se efectúa.
- La categoría a la que pertenece.
- El valor de la transacción.
- La periodicidad con la que se repite la transacción.
- Una descripción opcional.



Ilustración 11 Ventana de transacciones en Registro Contable

Desde la ventana de gráficos se pueden ver tres tipos de gráficos para distintos períodos.

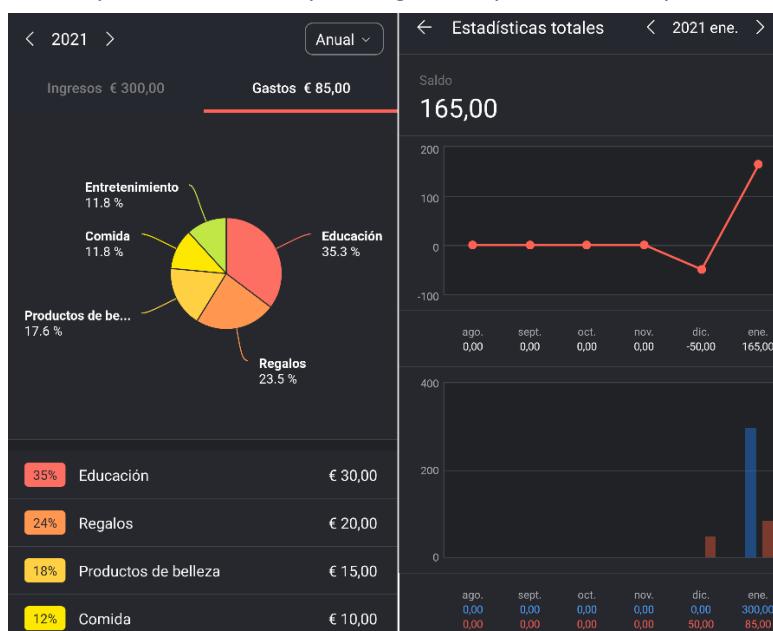


Ilustración 12 Ventana de gráficos en Registro Contable

Gestión de activos

Este tipo de aplicaciones se centran en analizar la evolución del patrimonio del usuario.

El usuario introduce los datos de los activos que tiene en propiedad indicando el precio y la fecha en la que se efectuó la adquisición y la aplicación obtiene los datos de cotización más recientes.

La aplicación utiliza estos datos para calcular un conjunto de indicadores que organiza gráficas y tablas para mostrárselos al usuario.

La mayoría de las aplicaciones capaces de realizar este tipo de análisis son de pago. Sin embargo, existen algunas de uso gratuito.

Las características que se repiten con mayor frecuencia son:

- Posibilidad de anotar transacciones de compra y venta de activos.
- Posibilidad de anotar transacciones de comisiones y dividendos.
- Posibilidad de ver varias estadísticas acerca de la rentabilidad, volatilidad y composición de los activos.
- Posibilidad de visualizar los datos introducidos mediante listas ordenadas y gráficas con diferentes períodos.
- Posibilidad de visualizar la variación temporal de los activos.

Otras características adicionales son:

- Descargar un informe con los datos más relevantes en un formato de texto.
- Permitir la comparación entre diversas carteras.

Características que me parecen especialmente útiles, pero no he encontrado.

- Permitir importar y exportar los datos a un fichero con formato editable.
- Obtener una mayor personalización de las posibles composiciones de los activos.
- Permitir distribuir los activos en categorías personalizadas.

Rankia Mi Cartera

Rankia es una comunidad financiera dirigida a inversores cuyo propósito es aportar transparencia al sector financiero con la finalidad de que los consumidores particulares resuelvan dudas e intercambien opiniones.

Para lograrlo, ofrecen de manera gratuita múltiples herramientas entre las que se encuentran foros, blogs, libros y el servicio web de [Mi Cartera](#) entre otros.



Descubre "Mi Cartera" en Rankia

Mi cartera es la herramienta que Rankia pone a disposición de sus usuarios para gestionar sus inversiones.

Para empezar a usarla de forma gratuita,

Identifícate

O

Regístrate

Ilustración 13 Página de bienvenida de Mi Cartera en Rankia

Lo primero que encontramos al acceder a la web son las ventanas de registro e identificación:

Crea tu cuenta

Y disfruta de todo el contenido de [la comunidad](#)

 Regístrate con Google

 Regístrate con Facebook

— O regístrate con una cuenta en Rankia —

*Nombre de usuario —

La comunidad te conocerá con este nombre

*Contraseña —

*Correo electrónico —

Te enviaremos un enlace para validar tu cuenta

*Nombre —

*Apellidos —

He leído y acepto los [términos y condiciones legales de Rankia](#).

[Crea tu cuenta](#)

Entra en Rankia

¿Aún no estás registrado? [Crea tu cuenta](#)

 Acceder con Google

 Acceder con Facebook

— O accede con tu cuenta de Rankia —

*Nombre de usuario o email —

*Contraseña —

[¿Has olvidado tu contraseña?](#)

[Iniciar sesión](#)

Ilustración 14 Páginas de registro e identificación en Mi Cartera en Rankia



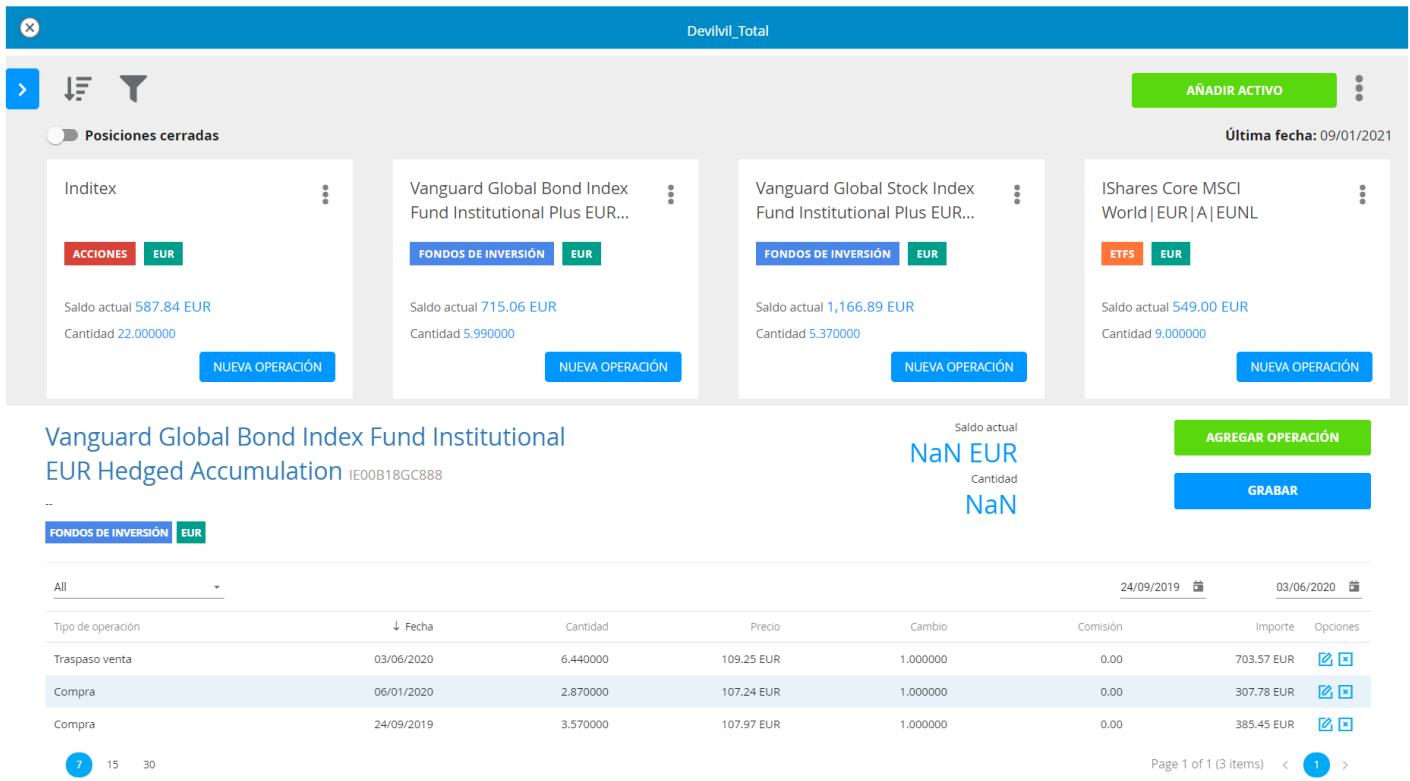
The screenshot shows the Rankia website's user interface. At the top, there is a navigation bar with links for 'MI PERFIL', 'RANKINGS', 'TUTORIAL', 'Buscar' (Search), and a 'Nueva Cartera' (New Portfolio) button. On the right, a user profile for 'David Cuesta Alario' (@DavidCuesta) is displayed. The main content area features a dark-themed dashboard. On the left, a summary section shows a total portfolio value of '3.018,79 €', a risk level of 'Agresivo', and performance metrics for the last week, month, and year. On the right, there is a circular performance indicator showing '7,07' and a line chart showing portfolio performance from March 2020 to January 2021.

Ilustración 15 Páginas de inicio en Mi Cartera en Rankia

Una vez registrados e identificados podemos acceder a la aplicación web.

En la página inicial podemos encontrar:

- Un resumen para cada una de las carteras que incluye:
 - o Varios indicadores de rentabilidad para distintos períodos de tiempo.
 - o Una gráfica que representa la variación porcentual de capital en un periodo de tiempo personalizable.
 - o Una estadística que mide la relación riesgo/rentabilidad sobre 10.
- El botón del candado permite cambiar la privacidad de la cartera de pública a privada.
- El botón de la papelera elimina por completo la cartera.
- El botón del lapicero permite ver y editar las transacciones que conforman la cartera.
 - o Para agregar un nuevo activo escribimos su ISIN o TIKER ([Clave identificativa oficial del producto](#)). Se mostrarán todos los productos que coincidan con dicho símbolo, de entre los cuales, elegiremos la opción que más se ajuste al producto realmente adquirido.
 - o Para agregar una nueva operación completamos los datos de:
 - Tipo de transacción: Compra, venta, traspaso, comisión, dividendo, etc....
 - Fecha de la operación.
 - Precio de compra: Se autocompletará con el precio medio que tenía para la fecha de la operación.
 - Tipo de cambio en caso de que el producto cotice en una moneda distinta a la moneda local: Se autocompletará con el precio medio que tenía para la fecha de la operación.
 - Importe de la transacción y número de acciones adquiridas.



The screenshot shows the Rankia Mi Cartera application interface. At the top, there's a header bar with a close button, a search icon, a filter icon, and a green 'AÑADIR ACTIVO' button. To the right, it says 'Última fecha: 09/01/2021'. Below the header, there are four cards representing different investments:

- Inditex**: Type 'ACCIONES', currency 'EUR'. Saldo actual: 587.84 EUR, Cantidad: 22.000000. Buttons: 'NUEVA OPERACIÓN' and three dots.
- Vanguard Global Bond Index Fund Institutional Plus EUR...**: Type 'FONDOS DE INVERSIÓN', currency 'EUR'. Saldo actual: 715.06 EUR, Cantidad: 5.990000. Buttons: 'NUEVA OPERACIÓN' and three dots.
- Vanguard Global Stock Index Fund Institutional Plus EUR...**: Type 'FONDOS DE INVERSIÓN', currency 'EUR'. Saldo actual: 1,166.89 EUR, Cantidad: 5.370000. Buttons: 'NUEVA OPERACIÓN' and three dots.
- iShares Core MSCI World|EUR|A|EUNL**: Type 'ETFs', currency 'EUR'. Saldo actual: 549.00 EUR, Cantidad: 9.000000. Buttons: 'NUEVA OPERACIÓN' and three dots.

Below these cards, there's a section for the **Vanguard Global Bond Index Fund Institutional EUR Hedged Accumulation** (IE00B18GC888). It shows the current value in 'NaN EUR', quantity 'NaN', and a 'GRABAR' button. A 'FONDOS DE INVERSIÓN' tab is selected.

At the bottom, there's a table of transactions:

Tipo de operación	↓ Fecha	Cantidad	Precio	Cambio	Comisión	Importe	Opciones
Traspaso venta	03/06/2020	6.440000	109.25 EUR	1.000000	0.00	703.57 EUR	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/>
Compra	06/01/2020	2.870000	107.24 EUR	1.000000	0.00	307.78 EUR	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/>
Compra	24/09/2019	3.570000	107.97 EUR	1.000000	0.00	385.45 EUR	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/>

Page 1 of 1 (3 items) < 1 >

Ilustración 16 Ventana de administración de activos y transacciones en Mi Cartera en Rankia

Agregar activo

Nombre:

Nombre	Tipo	Mercado	Precio Última Fecha
Alphabet (US02079K3059)	Acciones	NASDAQ	1,797.83 USD 08/01/2021

Alphabet

Tipo de operación: Compra	Fecha: 09/01/2021
Precio: 1,797.830000	USD
Importe: 0.00	EUR
Tipo de cambio: 0.817929 USD	
Cantidad: 0.000000	

Opciones avanzadas ▾

CANCELAR GRABAR

Ilustración 17 Ventana para agregar activos y transacciones en Mi Cartera en Rankia

Se pueden crear y administrar varias carteras simultáneamente.

Para crear una nueva cartera se introduce su nombre y la divisa local.

Nueva cartera

Nombre	<input type="text"/>
Alias	<input type="text"/>
Divisa	EUR ▼
Pública	<input checked="" type="checkbox"/>
Opciones avanzadas ▾	
CONTINUAR	

Ilustración 18 Ventana para agregar una nueva cartera en Mi Cartera en Rankia

Si entramos al menú de inicio de alguna de las carteras podemos ver algunos resúmenes más detallados:

- Resumen del perfil público de la cartera, desde el que se nos permite ver datos como el número de seguidores o las medallas obtenidas.
- Listado de todas las posiciones con algunos indicadores como el porcentaje sobre el total o el riesgo del producto.
- La distribución de la cartera en cuanto al tipo de activos que la componen.
- Indicadores de Riesgo, Rentabilidad y Diversificación.



Posiciones	Peso Actual	Val./Part.	Gan./Perd.	Último precio	Riesgo
Vanguard Global Stock Index Fund Institutional Plus EUR Accumulation	38,65% 	1.166,89 EUR 5,37	146,18 EUR (14,32%)	217,30 EUR	35,31% Muy Alto
Vanguard Global Bond Index Fund Institutional Plus EUR Hedged Accumulation	23,69% 	715,06 EUR 5,99	12,25 EUR (1,74%)	119,38 EUR	4,59% Muy Bajo
Inditex	19,47% 	587,84 EUR 22,00	47,26 EUR (8,74%)	26,72 EUR	45,06% Muy Alto
iShares Core MSCI World EUR A EUNL	18,19% 	549,00 EUR 9,00	180,00 EUR (48,78%)	61,00 EUR	29,03% Alto
					
TIPO DE ACTIVO	DIVISA	CATEGORÍAS			
<ul style="list-style-type: none"> ● Fondos de Invers... 62,34% ● Acciones 19,47% ● ETFs 18,19% 	<ul style="list-style-type: none"> ● EUR 100,00% 	<ul style="list-style-type: none"> ● RV Global 56,84% ● RF Global 23,69% ● RV Europa 19,47% 			

Ilustración 19 Ventana de resúmenes de la cartera en Mi Cartera en Rankia

La aplicación permite descargar un informe en PDF que contiene los mismos datos que muestra la aplicación para cada una de las carteras.

En la ventana de Social es posible ver las carteras que otros usuarios hayan marcado como públicas.

- Las carteras públicas permiten acceder a la misma información que se ha comentado para las carteras propias.
- Es posible seguir a usuarios concretos para ver la actividad de sus carteras públicas.

Puntos Usuario		Seguidores Usuario	Recomendaciones Usuarios	Score Carteras
Puesto	Usuario	Puntos	Reputación	Seguidores
1	 Fortuna	117.600	⊕⊕⊕⊕⊕ 22 Ratings	724 
2	 @jrpintos	93.100	⊕⊕⊕⊕⊕ 4 Ratings	198 
3	 Manager	60.800	⊕⊕⊕⊕⊕ 32 Ratings	633 

Ilustración 20 Ventana de Rankin de usuarios en Mi Cartera en Rankia

Portfolio Performance

Enlace de descarga para la aplicación de escritorio de [Portfolio Performance](#).



Ilustración 21 Ventana de descarga de Portfolio Performance

Lo primero que encontramos al acceder a la aplicación son un conjunto de tablas que nos permiten crear tres tipos de contenedores de información.

Valores (Estándar)			
Nombre	ISIN	Símbolo	
Inditex	IE00B4L5Y983	ITX.MC	
iShares Core MSCI World UCITS	IE00B18GC888	SWDA.SW	
Vanguard Global Bonds	IE00BGCZ0B53	OP00012169.F	
Vanguard Global Bonds Institucional	IE00B03HD191	OP0001216E.F	
Vanguard Global Stoks	IE00BFPM9N11	OP00000WLG.F	
Vanguard Global Stoks Institucional		OP00011HBM.F	

Cuentas de Depósito		
Cuenta	Balance	Divisa
Colchon	10.000,00	EUR
DeGiro	31,00	EUR
IndexaCapital	18,11	EUR
ClickTrade	15,22	EUR
Gastos	3.000,00	EUR

Cuentas de Valores		
Cuenta de valores	Cuenta de referencia	Volume
BogleHead	Colchon	2.407,83
DGI	Colchon	587,84

Ilustración 22 Contenedores de información en Portfolio Performance

- **Valores:** representan los distintos activos que se pueden comprar y vender.

Para crear un nuevo valor se debe introducir una fila adicional con los datos de:

- o Nombre: Clave identificativa de la fila que debe ser único.
- o ISIN: Clave identificativa oficial del producto al que representa el valor.
- o Símbolo: Clave identificativa que utiliza Yahoo Finance para el producto al que representa el valor.

- **Cuentas de depósito:** Representa las cuentas donde se registran las transacciones

Para crear un nuevo depósito se debe introducir una fila adicional donde sólo es necesario indicar el nombre que será la clave identificativa de la fila por lo que debe ser único

En un depósito se permite registrar varios tipos de transacciones:

- Aportaciones o retiradas de efectivo.
- Compra o ventas de valores.
- Cobros de plusvalías o pagos de comisiones o impuestos.

Para crear una nueva transacción es necesario introducir una fila adicional donde se indicará:

- La fecha de la transacción.
- El tipo de transacción: Podemos encontrar las siguientes opciones en una lista predeterminada: Depósito, Compra, Venta, Comisión, Impuesto, Intereses, devolución, etc....
- La cantidad de capital invertido.
- El activo que se suscribe: Debe ser uno de los valores que hemos registrado.
- El número de acciones que se adquieren.

Transacciones		Gráfico de saldo de la cuenta					
Fecha	Tipo	Cantidad	Balance	Activo	Acciones	por acción	
20 sept. 2019	Depósito	1.000,00	1.000,00				
24 sept. 2019	Compra	-385,46	614,54	Vanguard Global Bonds	3,6	107,972	
25 sept. 2019	Compra	-603,63	10,91	Vanguard Global Stoks	22,7	26,6151	
2 ene. 2020	Depósito	750,00	760,91				
6 ene. 2020	Compra	-307,77	453,14	Vanguard Global Bonds	2,9	107,2369	
7 ene. 2020	Compra	-435,02	18,12	Vanguard Global Stoks	15,3	28,4699	
3 jun. 2020	Compra	-702,81	-684,69	Vanguard Global Bonds Institucional	6	117,3306	
3 jun. 2020	Venta	703,55	18,86	Vanguard Global Bonds	6,4	109,2469	
3 jun. 2020	Compra	-1.020,73	-1.001,87	Vanguard Global Stoks Institucional	5,4	190,0801	
3 jun. 2020	Venta	1.021,14	19,27	Vanguard Global Stoks	38	26,9004	
4 nov. 2020	Comisiones	-1,16	18,11	Vanguard Global Stoks Institucional			

Ilustración 23 Tabla de transacciones para la cuenta de depósito de Indexa Capital en Portfolio Performance

- **Cuentas de valores:** Permiten agrupar varios valores en un único contenedor independientemente de las cuentas en las que se hayan hecho suscripciones:

Resumen de activos		Transacciones					
Acciones	Nombre	Símbolo	Precio	Valor de mer...	Valor en %	Nota	
	Sin clasificación			2.407,83	100,00		
6	Vanguard Global Bonds Institucional	OP00012I6E.F	119,5...	716,30	29,75		
5,4	Vanguard Global Stoks Institucional	OP00011HBM.F	212,5...	1.141,46	47,41		
9	iShares Core MSCI World UCITS	SWDA.SW	USD 7...	550,07	22,85		
	Total			2.407,83	100,00		

Ilustración 24 Tabla de valores suscritos a la cuenta de valores de BogleHead en Portfolio Performance

En la ventana de valores, seleccionando un valor, podemos ver una gráfica donde se muestran

- Los precios históricos de dicho valor.
- Todas las transacciones efectuadas sobre dicho valor (en verde las compras y en rojo las ventas)

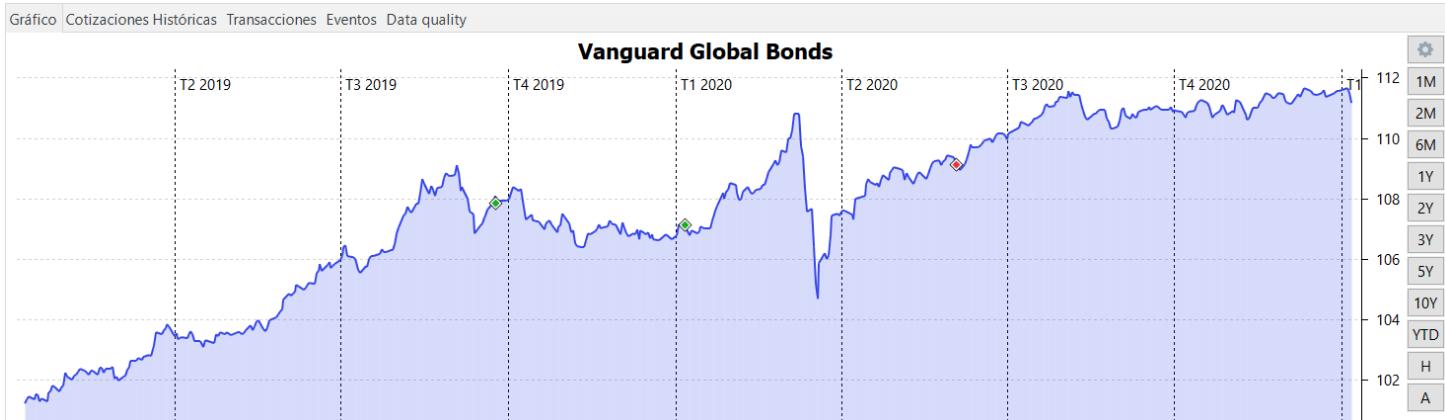


Ilustración 25 Grafico precios históricos en Portfolio Performance

La ventana de All Transactions permite:

- Ver todas las transacciones que se han efectuado.
- Añadir nuevas transacciones a cualquier cuenta creando una nueva fila con los datos pertinentes.

All transactions

Fecha	Tipo	Activo	Acciones	Precio	Cantidad	Comisio...	Impuestos	Valor Ne...	Cuenta	Cuenta de contrap...
4 nov ...	Comisiones	Vanguard Global Stoks Institucional			1,16			1,16	IndexaCapital	
3 nov ...	Dividendo	Inditex	22		7,70		1,46	6,24	ClickTrade	
7 oct ...	Depósito				3.000,00			3.000,00	Gastos	
29 jun ...	Compra	Inditex	22	23,91	526,02	15,00		541,02	DGI	ClickTrade
3 jun ...	Compra	Vanguard Global Stoks Institucional	5,4	190.0801	1.020,73			1.020,73	BogleHead	IndexaCapital
3 jun ...	Compra	Vanguard Global Bonds Institucional	6	117.3306	702,81			702,81	BogleHead	IndexaCapital
3 jun ...	Venta	Vanguard Global Stoks	38	26.9004	1.021,14			1.021,14	BogleHead	IndexaCapital
3 jun ...	Venta	Vanguard Global Bonds	6,4	109.2469	703,55			703,55	BogleHead	IndexaCapital
28 ma ...	Depósito				550,00			550,00	ClickTrade	
17 ma ...	Compra	iShares Core MSCI World UCITS	9	41,00	369,00			369,00	BogleHead	DeGiro
17 ma ...	Depósito				400,00			400,00	DeGiro	
7ene ...	Compra	Vanguard Global Stoks	15,3	28.4699	435,02			435,02	BogleHead	IndexaCapital
6ene ...	Compra	Vanguard Global Bonds	2,9	107.2369	307,77			307,77	BogleHead	IndexaCapital
2ene ...	Depósito				750,00			750,00	IndexaCapital	
25 sep ...	Compra	Vanguard Global Stoks	22,7	26.6151	603,63			603,63	BogleHead	IndexaCapital
24 sep ...	Compra	Vanguard Global Bonds	3,6	107.972	385,46			385,46	BogleHead	IndexaCapital
20 sep ...	Depósito				1.000,00			1.000,00	IndexaCapital	
1ene ...	Depósito				10.000,00			10.000,00	Colchon	

Ilustración 26 Ventana All Transactions en Portfolio Performance

La ventana de resumen de activos muestra una tabla donde se recogen todos los valores registrados ordenados por cuentas, indicando para cada uno un conjunto de indicadores fácilmente personalizables:

Resumen de activos (Estándar)

Nombre	Acciones	Precio	Valor de mer...	Valor en %	ΣDiv 1 año	Div% 1 año	TIR 1 año	Rent. Abs. 1 año
Sin clasificación			16.060,00	100,00	7,70			367,19
IndexaCapital			18,11	0,11				
Vanguard Global Bonds Institucional	6	119.582	716,30	4,46	0,00	0,00%	3,20%	13,49
Inditex	22	26,72	587,84	3,66	7,70	1,42%	19,90%	53,06
Colchon			10.000,00	62,27				
Gastos			3.000,00	18,68				
ClickTrade			15,22	0,09				
Vanguard Global Stoks Institucional	5,4	212.562	1.141,46	7,11	0,00	0,00%	20,17%	119,57
DeGiro			31,00	0,19				
iShares Core MSCI World UCITS	9	USD 74,87	550,07	3,43	0,00	0,00%	63,07%	181,07
Total			16.060,00	100,00	7,70			367,19

Ilustración 27 Ventana Resumen de Activos en Portfolio Performance

En la ventana de gráficos podemos encontrar:

- Un gráfico de sectores donde muestra el peso de cada activo sobre el total.

Es posible configurar el gráfico para que muestre únicamente una cuenta de valores.

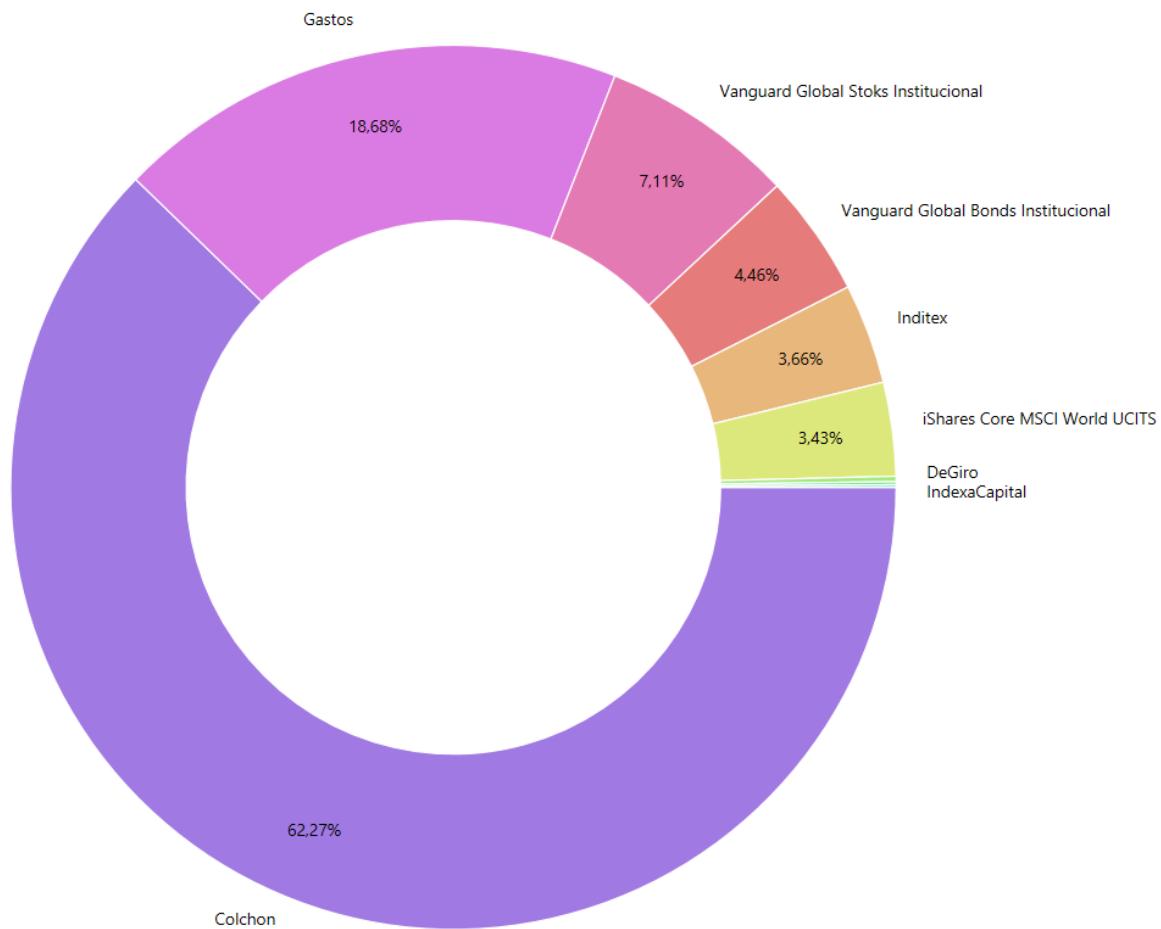


Ilustración 28 Ventana Gráficos, Grafico de Activos en Portfolio Performance

- Un gráfico de líneas que muestra la evolución temporal de cada una de las cuentas de valores.

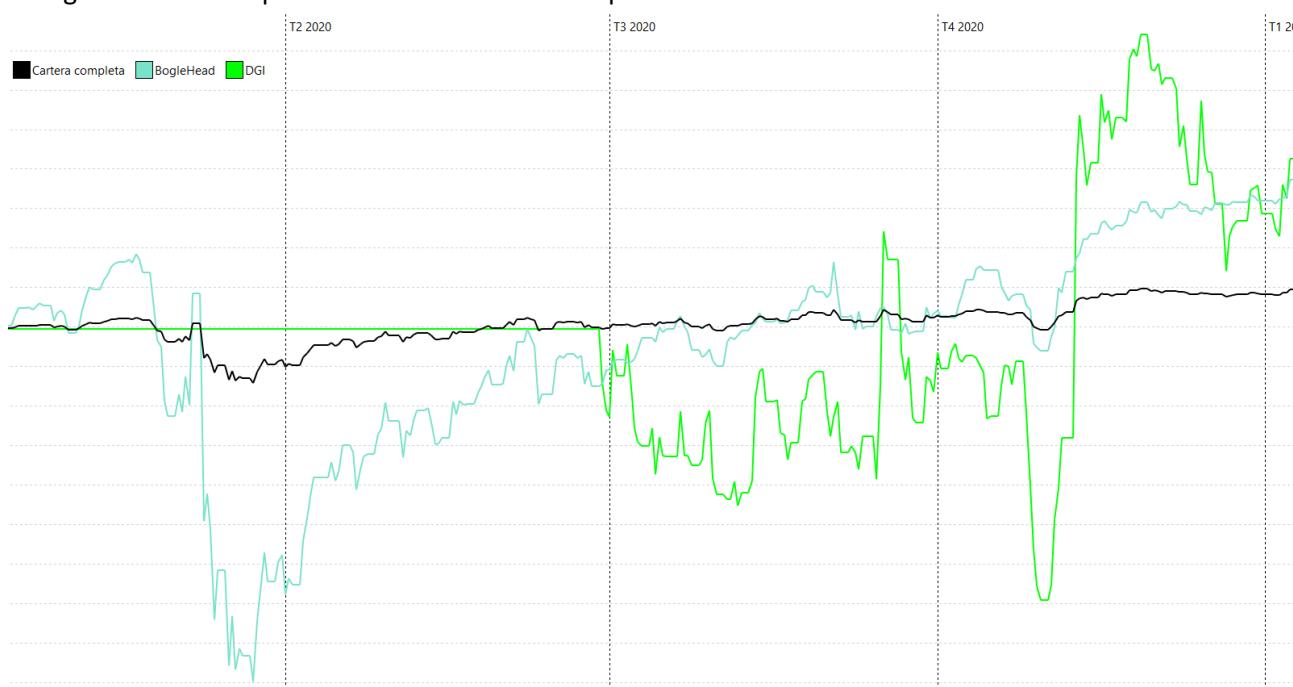


Ilustración 29 Ventana Gráficos, Grafico de Evolución temporal en Portfolio Performance

Objetivos

El principal objetivo de este TFG es diseñar, implementar y probar una aplicación web que ofrezca a personas sin ningún tipo de conocimiento sobre finanzas personales, una única herramienta que les permita simplificar el proceso de toma de decisiones financieras.

Para ello deberemos obtener un resumen patrimonial que represente la situación financiera de los usuarios unificando dos herramientas ampliamente extendidas, que habitualmente se presentan por separado.

Requisitos

Funcionales

Las funcionalidades que se pretenden incluir para obtener la aplicación son:

- Implementar una interfaz gráfica intuitiva que incluya:
 - o Ventanas que permitan la gestión de usuarios:
 - Registro y creación de una nueva cuenta.
 - Iniciar y cerrar sesión.
 - Modificar privilegios.
 - Eliminar la cuenta.
 - o Ventanas que permitan la gestión de la información financiera:
 - La importación de datos mediante formularios y ficheros CSV o XML.
 - La eliminación de datos mediante formularios.
 - La exportación de datos mediante ficheros CSV o XML.
 - o Ventanas que permitan la comparación dinámica mediante tablas y gráficas de:
 - El balance de gastos e ingresos.
 - El balance de activos y pasivos.
 - La evolución de los activos que se tienen en propiedad.
 - Un resumen del estado patrimonial actual.
 - o Ventanas que permitan gestionar la presentación de la información:
 - Creación de categorías de gastos e ingresos.
 - Creación de distribuciones y estrategias para activos.
- Implementar un Scraper capaz de obtener para cualquier activo:
 - o Su cotización actual.
 - o El historial de cotizaciones para un periodo de tiempo determinado.
 - o Determinadas estadísticas acerca de la distribución geográfica y la composición.
- Implementar una calculadora que permita obtener los ratios financieros de cualquier combinación de activos.
- Implementar un sistema que permita almacenar la información financiera de cada usuario.
- Implementar un sistema de usuarios que permita asignar distintos privilegios a cada usuario.

No Funcionales

Aparte de las funcionalidades que se pretenden incluir, la aplicación debe:

- Garantizar la integridad y confidencialidad de la información mediante:
 - o Restricción del acceso a las funcionalidades mediante un sistema de registro obligatorio.
 - o Técnicas de encriptación suficientes para el contenido sensible.
 - o Cierre de sesión automático por inactividad.
 - o Registro de intentos de acceso.
 - o Copias de seguridad que se efectuaran de manera automática y transparente para el usuario.
- Respetar la legislación europea que sea de aplicación.
- Cumplir los estándares de accesibilidad mediante una interfaz gráfica que sea comprensible para todos.

Alcance

Debido a las limitaciones de tiempo y esfuerzo que se espera de un trabajo fin de grado quedarán fuera del alcance del proyecto las siguientes funcionalidades:

- No se garantiza el correcto funcionamiento si no se dispone de una conexión a internet estable.
- No se garantiza el correcto funcionamiento en ningún navegador excepto para el que fue diseñado.
- No se garantiza el diseño responsivo, por lo que la aplicación sólo se visualizará correctamente en la disposición de pantalla para el que fue diseñado.

Estado del Arte

Este apartado de la documentación está destinado a facilitar todos los conceptos necesarios para comprender el funcionamiento de la aplicación, así como el motivo por el que se han aplicado determinadas decisiones de diseño. De este modo se recopilarán una gran variedad de información relacionada con el proyecto.

Definiciones

Este apartado de la documentación del proyecto está destinado a recopilar aquellos términos que se hayan mencionado en este documento y puedan ser desconocidos para aquellos lectores que no estén especializados en las ramas del conocimiento que se tratan. De este modo se expondrá una breve definición que permita al lector entender las bases de aquellos conceptos que no han sido tratadas en el apartado que se mencionan.

Informáticas

Web Scraping: Técnica utilizada para recopilar información de páginas web sin interacción del usuario.

CSV "Comma Separated Values": Se trata de un archivo de texto utilizado para almacenar datos mediante una estructura de tabla en la que se utiliza el carácter punto y coma para separar las columnas y los saltos de línea para separar las filas.

XML: Se trata de un archivo de texto utilizado para almacenar datos mediante una estructura dinámica especificada a través de un lenguaje marcado extensible que utiliza una serie de etiquetas personalizadas con la finalidad de describir la estructura del documento.

Framework: Se trata de un conjunto de herramientas que proporcionan un esquema de trabajo y una serie de utilidades utilizadas en el desarrollo de aplicaciones web dinámicas. En general los Frameworks están asociados a lenguajes de programación.

Financieras

Balance: documento que resume el patrimonio de una persona o institución en un momento dado.

Patrimonio: Conjunto de todos los bienes, derechos y obligaciones cuantificables de una persona o institución.

Activo: Conjunto de bienes y derechos de una persona o institución por los que se espera obtener beneficios económicos como resultado de su explotación.

Pasivo: Conjunto de bienes y obligaciones de una persona o institución por los que se espera contraer gastos o deudas como resultado de su explotación.

Cálculos

Este apartado de la documentación está destinado a recopilar todos los conceptos necesarios para comprender los cálculos que deberá llevar a cabo la aplicación, así como la interpretación de las tablas y graficas que ofrece como resultado.

Beneficio

Se calcula como el valor actual de los activos menos el valor que tuvieron en el periodo inicial añadiendo las comisiones o rentas que se pagaron o cobraron durante todo el periodo.

$$\text{Beneficio} = \text{ValorActual} - \text{ValorAportaciones} + \text{Rentas} - \text{Comisiones}$$

El valor actual de las aportaciones se calcula como el sumatorio de las cotizaciones de cada aportación multiplicadas por el número de títulos de la operación y el tipo de cambio con la divisa de destino en el momento en que se realizó la operación.

- Si la operación es de venta el número de títulos estará en negativo por lo que el valor final se sustraerá de la operación:

$$\text{ValorAportaciones} = \sum \text{PrecioCompraVenta} * \text{NúmeroTitulos} * \text{TipoCambioCompraVenta}$$

Precio Medio

Se trata un método de valoración que pretende estimar el precio que se habría pagado por un activo si se hubieran adquirido todos los títulos en una única operación.

Se calculará como el valor total de las aportaciones del periodo dividido el número de títulos en posesión.

$$\text{PrecioMedio} = \frac{\text{ValorAportaciones}}{\text{NúmeroTotalTitulos}}$$

Rentabilidad Anualizada

Se trata de un método de valoración que pretende estimar los beneficios que habría obtenido una inversión en un año genérico basándose en los beneficios que se han obtenido en un periodo de tiempo determinado.

Se utiliza para hacer estimaciones a futuro o para compararlas con otras inversiones.

Para calcular la rentabilidad anualizada en una inversión donde hubiera movimientos de capital ([entradas o salidas de dinero](#)) utilizaríamos la fórmula del interés compuesto.

No obstante, que no haya entradas y salidas de capital no es lo habitual sino la excepción.

Rentabilidad Ponderada por tiempo

La rentabilidad ponderada por tiempo elimina el efecto de las aportaciones y los reembolsos en el cálculo de la rentabilidad anualizada.

En primer lugar, calculamos las rentabilidades parciales para cada transacción que se haya realizado:

$$TWR = \frac{\text{ValorFinal} - \text{ValorInicial} + \text{Reembolsos} - \text{Suscripciones}}{\text{ValorInicial}}$$

Finalmente calculamos la rentabilidad acumulada de todo el periodo multiplicando el TWR de cada aportación:

$$\prod (1 - TWR_{Aportacion})$$

Para ajustarlo al periodo anual estándar lo multiplicamos por el número de días de todo el periodo y lo dividimos entre el número de días de un año. ([365](#))

Herramientas

Este apartado de la documentación está destinado a presentar las diferentes herramientas y tecnologías que han sido necesarias para el desarrollo de la aplicación. De este modo se describirán las características por las que han sido escogidas y se indicará el uso que se les ha dado en el proyecto.

Técnicas

Criptografía

Trata el estudio y aplicación de los algoritmos, protocolos y sistemas que permiten ocultar la información de un mensaje mediante un código que permite alterar las representaciones lingüísticas de ciertos mensajes con el fin de hacerlos ininteligibles a receptores no autorizados. De esta forma se consigue proteger la información y dotar de seguridad a las comunicaciones y a las entidades que se comunican independientemente de quien acceda al canal e intercepte los mensajes.

Los grandes avances producidos en el mundo de la criptografía se basan en técnicas matemáticas, esto ha sido posible gracias a la evolución que se han producido en el campo de la matemática y la informática.

Sistemas simétricos

Se trata de un método criptográfico en el cual se usa una misma clave para cifrar y descifrar mensajes en el emisor y el receptor.

Las dos partes que se comunican han de ponerse de acuerdo de antemano sobre la clave a usar. A continuación, el remitente cifra el mensaje usando la clave y lo envía al destinatario.

Finalmente, el destinatario puede descifrar dicho mensaje utilizando la misma clave con la que se cifró originalmente.

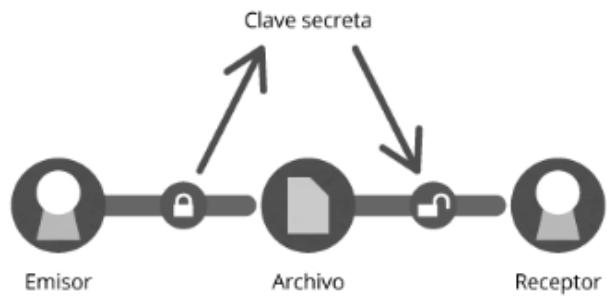


Ilustración 30 Encriptación Con Clave Simétrica

Sistemas asimétricos

Se trata de un método criptográfico basado en el uso de dos claves:

- Una clave pública que será difundida públicamente a todas las personas que necesiten mandar el mensaje.
- Una clave privada que sólo será conocida por el receptor del mensaje.

De este modo el receptor comunica su clave pública al emisor por un canal que no tiene por qué ser seguro. A continuación, el emisor cifra el mensaje utilizando dicha clave pública y lo envía al destinatario. Finalmente, el receptor descifra el mensaje utilizando su clave privada.

En este sistema las claves se generan por pares, de modo que todo lo que una clave cifre sólo su pareja lo puede descifrar.



Ilustración 31 Encriptación Con Clave Asimétrica

Ventajas:

- Sólo el destinatario puede leer el mensaje.
- Sólo hay que almacenar una clave.
- Cualquiera puede cifrar con la clave pública.
- No son necesarios canales seguros para comunicar la clave.

Desventajas:

- El cifrado y descifrado son más lentos que en los sistemas de clave única.
- Debemos certificar la autenticidad de la clave pública del destinatario para evitar suplantaciones de identidad
- No debería ser posible obtener la clave privada a partir de la clave pública
- Un mensaje cifrado con la clave privada lo podría descifrar cualquiera

AES

Advanced Encryption Standard es un esquema de cifrado por bloques que utiliza la criptografía simétrica

Encriptación de claves

No sirve de nada almacenar los datos de un usuario encriptados mediante una contraseña segura, si a continuación se almacena dicha contraseña sin ningún tipo de protección. De este modo, cualquiera que acceda a la información, tendrá la contraseña disponible para descifrarla.

No obstante, uno de los principales problemas a la hora de encriptar la información de distintos usuarios es reconocer cuál es el usuario que se está identificando si la contraseña no está almacenada junto a dicha información.

Para solucionar este problema es posible utilizar funciones de dispersión Hash. Se trata de un algoritmo que genera un criptograma de un tamaño determinado que representa todo el contenido original de un mensaje de tal modo que sea imposible:

- Obtener el mensaje original partiendo del criptograma.
- Obtener dos criptogramas iguales partiendo de mensajes diferentes por muy parecidos que sean.

Mediante estas funciones podemos calcular el resumen criptográfico de la contraseña y almacenarlo junto a la información encriptada, sin tener que preocuparnos de que quien acceda a la base de datos pueda recuperar la contraseña. Al mismo tiempo, es posible reconocer si la contraseña introducida por un usuario es correcta comparando los resúmenes hash de la contraseña recién introducida y la previamente almacenada.

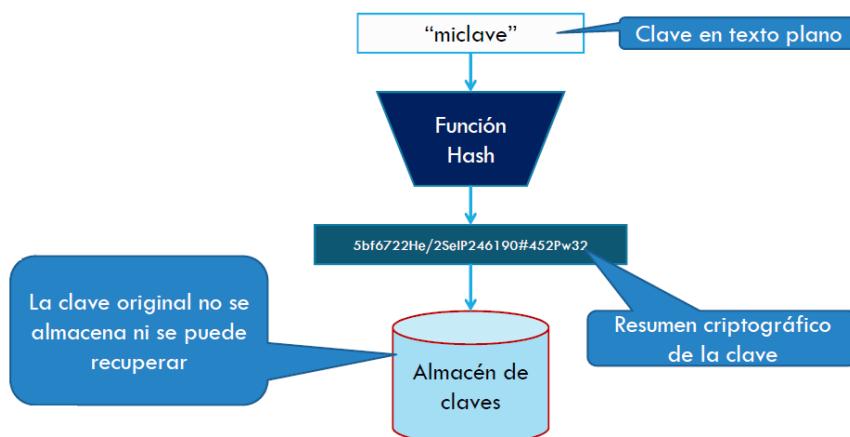


Ilustración 32 Almacenamiento de claves mediante Hash

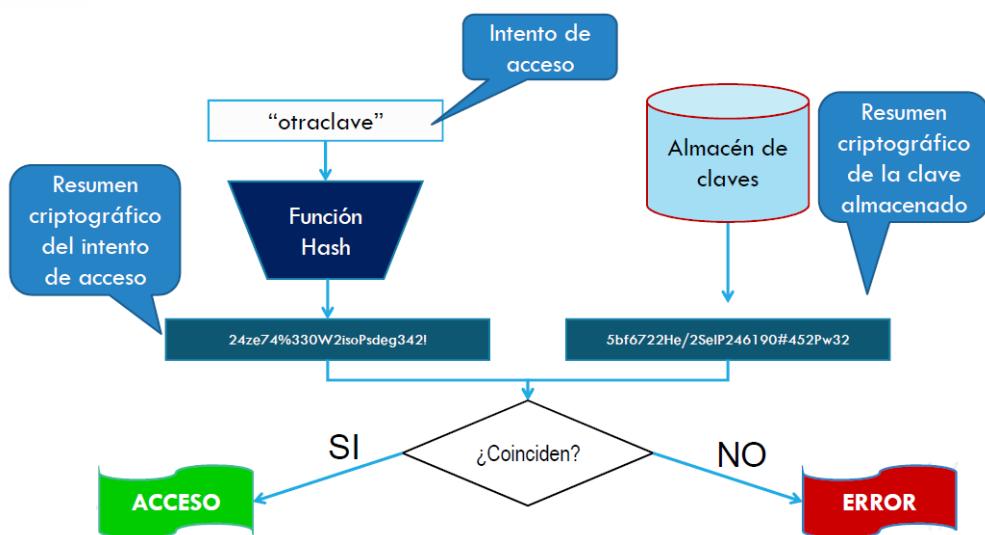


Ilustración 33 Identificación de claves mediante Hash

Este método plantea un nuevo problema, cualquier usuario que utilice la misma contraseña obtendrá el mismo resumen Hash. Como consecuencia sería posible reconocer ciertas contraseñas pre calculando todo el espacio de claves ([calculas cuál sería el Hash de una clave y miras si algún usuario tiene ese mismo Hash](#))

Este problema se soluciona añadiendo un parámetro aleatorio a la contraseña de cada usuario que se almacenará junto al resumen criptográfico de la contraseña y los datos encriptados.

Con esto se consigue que el Hash tenga una codificación diferente por cada usuario, aún teniendo la misma clave. Además, tener disponibles la base de datos, el resumen Hash y la sal no aporta ninguna información sobre la clave porque no es posible reconocer qué parte del resumen Hash pertenece a la clave y cuál a la sal.

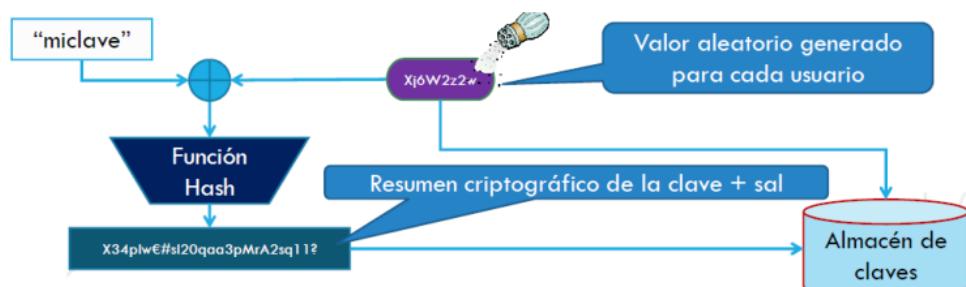


Ilustración 34 Almacenamiento de claves mediante Hash + Sal

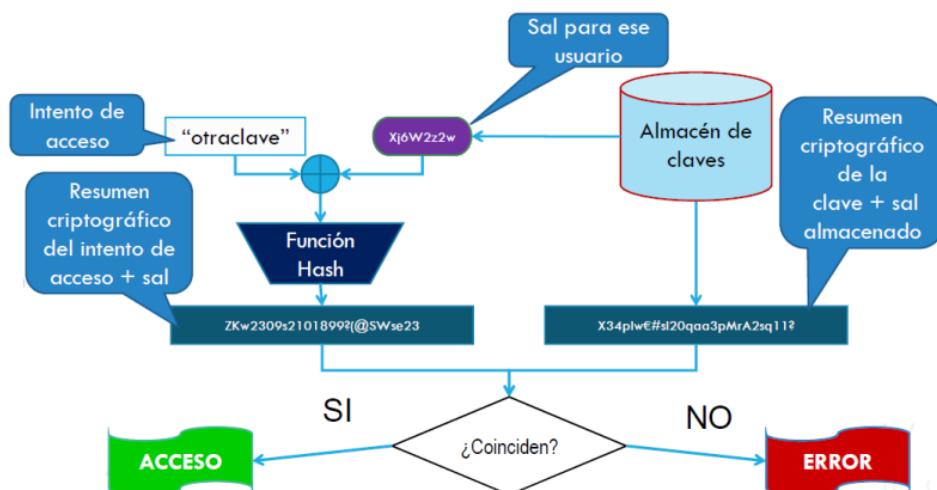


Ilustración 35 Identificación de claves mediante Hash

Lenguajes de programación

En este proyecto se han utilizado los siguientes lenguajes de programación:

Python

Se trata de un lenguaje de programación cuya filosofía es maximizar la legibilidad de su código.

Sus principales características son:

- Admite varios métodos de programación incluyendo la orientación a objetos.
- Es un lenguaje interpretado, por lo que no se necesita compilar el código fuente para poder ejecutarlo.
- Permite realizar cualquier tipo de programa, incluyendo aplicaciones de escritorio y aplicaciones Web.
- El propio lenguaje incorpora una gran cantidad de librerías que simplifican las tareas habituales, además existen muchas librerías externas que permiten tratar temas específicos.
- Se trata de una herramienta gratuita, incluso para propósitos empresariales.
- Tiene una sintaxis muy visual gracias a una notación identada de obligado cumplimiento que obliga a todos los programadores a utilizar la misma notación.

MariaDB

Se trata de un proyecto de licencia abierta que permite gestionar bases de datos relacionales.

Sus principales características son:

- Es sistema potente y versátil.
- Se puede utilizar para proyectos en la web.
- Para comunicarse con las bases de datos utiliza el estándar SQL mantenido por ANSI
- Permite el Cifrado por AES mediante los comandos
 - o `aes_decrypt (Campo , Clave)`
 - o `aes_encrypt (Campo , Clave)`

HyperText Markup Language HTML

Se trata de un lenguaje de etiquetado basado en hipertexto que permite mostrar la información de manera estructurada a través de un explorador web.

Sus principales características son:

- Es un estándar mantenido por World Wide Web Consortium
- Permite definir una estructura común para la elaboración de páginas web en sus diferentes versiones
- Es un lenguaje de marcado interpretado:
 - o El código solamente contiene texto.
 - o Nos permite indicar la estructura de nuestro documento mediante etiquetas.
 - o El navegador es el encargado de compilar todos los elementos para visualizar la página final.
- Ofrece una estructuración lógica fácil de interpretar tanto por humanos como por máquinas.

Cascading Style Sheets CSS

Se trata de un lenguaje de diseño gráfico que permite definir la presentación de un documento estructurado mediante un lenguaje de marcado mediante un conjunto de reglas que se aplican en cascada manteniendo el orden de aparición de las etiquetas con la finalidad de evitar la superposición de estilos.

Sus principales características son:

- Es un estándar mantenido por World Wide Web Consortium
- Se utiliza principalmente para interfaces de usuario escritas en HTML, aunque puede ser aplicado a cualquier documento XML.
- Es aconsejable separar el fichero de estilos del documento de marcado con la finalidad de maximizar su reutilización.

JavaScript

Se trata de un lenguaje de programación implementado como parte de un navegador web cuya finalidad es conseguir mejoras en la interfaz de usuario y páginas web dinámicas.

No se debe confundir con Java porque a pesar de que tienen una sintaxis similar, ambos lenguajes tienen semánticas y propósitos diferentes.

Librerías

Flask

[Flask](#) es un micro framework para Python que permite diseñar aplicaciones web de manera sencilla y ágil

- Incluye un servidor de desarrollo por lo que no se necesita una infraestructura para probar las aplicaciones
- Incluye un depurador y soporte integrado para pruebas unitarias
- Es compatible con Python3.
- Soporta de manera nativa el uso de cookies seguras.
- Se trata de código Open Source amparado bajo una licencia BSD.

En este proyecto se ha utilizado con la finalidad de construir la estructura HTML que forma la interfaz gráfica de aplicación.

BeautifulSoup

[BeautifulSoup](#) es una librería para Python que permite extraer datos de páginas web diseñadas mediante HTML. Para utilizarlo sólo hay que importar su librería y crear un objeto de la clase BeautifulSoup que representa el documento como una estructura de datos anidada por la que se puede navegar de la misma forma que se haría con cualquier estructura anidada.

- importación de la librería:

```
from bs4 import BeautifulSoup
```

- Transformación del documento:

```
estructuraAnidada = BeautifulSoup( URI , 'html.parser' )
```

- Algunos ejemplos de navegación:

```
estructuraAnidada . title  
estructuraAnidada . title . name  
estructuraAnidada . title . parent . name
```

En este proyecto se ha utilizado con el propósito de extraer información acerca de los activos registrados por el usuario, tales como la cotización o las distribuciones.

Pandas DataReader

[Pandas DataReader](#) es una librería para Python que permite crear un marco de datos a partir de varias fuentes de datos de Internet entre las que se encuentra incluida Yahoo Finance

- importación de la librería:

```
from pandas_datareader import data
```

- Obtención de los datos:

```
Datos = data.DataReader( producto , data_source = 'yahoo' , start = FechaInicio , end = FechaFin )
```

- Obtención de los datos:

En este proyecto se ha utilizado con el objetivo de extraer información acerca de los activos registrados por el usuario, tales como la cotización o las distribuciones.

Charts.js

Charts.js es un plugin de JavaScript diseñado para incrustar gráficas en las páginas Web de forma simple y flexible.

Para utilizarlo sólo hay que importar su librería y crear un objeto Chart que recibirá como parámetros el objeto canvas en el que se pretende insertar el gráfico y una estructura JSON que contendrá la información que se pretende representar, así como las opciones de configuración del gráfico estructuradas conforme se indica en la documentación del tipo de gráfica concreto que se pretende crear:

- Importación de la librería:

```
< script src = js/Chart.bundle.js > </script >
```

- En la siguiente imagen se representa sin ninguna alteración el ejemplo más simple representado en la página web oficial de la herramienta:

```

<canvas id="myChart" width="400" height="400"></canvas>
<script>
  var ctx = document.getElementById('myChart').getContext('2d');
  var myChart = new Chart( ctx,
  {
    type: 'bar',
    data:
    {
      labels: ['Red', 'Blue', 'Yellow', 'Green', 'Purple', 'Orange'],
      datasets:
      [
        {
          label: '# of Votes',
          data: [12, 19, 3, 5, 2, 3],
          backgroundColor:
          [
            'rgba(255, 99, 132, 0.2)',
            'rgba(54, 162, 235, 0.2)',
            'rgba(255, 206, 86, 0.2)',
            'rgba(75, 192, 192, 0.2)',
            'rgba(153, 102, 255, 0.2)',
            'rgba(255, 159, 64, 0.2)'
          ],
          borderColor:
          [
            'rgba(255, 99, 132, 1)',
            'rgba(54, 162, 235, 1)',
            'rgba(255, 206, 86, 1)',
            'rgba(75, 192, 192, 1)',
            'rgba(153, 102, 255, 1)',
            'rgba(255, 159, 64, 1)'
          ],
          borderWidth: 1
        }
      ],
      options:
      {
        scales:
        {
          yAxes: [{ ticks: { beginAtZero: true } }]
        }
      }
    });
</script>

```

Ilustración 36 Ejemplo de utilización de Charts.js

En este proyecto se ha utilizado con el fin de construir todos los gráficos que se muestran en la aplicación.

Documentación

Microsoft Word

Se trata de un software de edición de texto que permite al usuario la creación y edición de documentos de texto.

Forma parte de Microsoft Office, un paquete de programas que permite la realización de actividades ofimáticas.

Sus principales características son:

- La edición del texto es sencilla, se puede utilizar conociendo únicamente los conceptos básicos.
- Dispone de muchas herramientas para editar el formato y los estilos.
- Da la posibilidad de visualizar el resultado final antes de ser impreso.
- Cuenta con un corrector ortográfico integrado.
- Permite utilizar atajos de teclado para agilizar el trabajo.

El principal inconveniente de esta herramienta es que requiere de una licencia que no es gratuita.

En este proyecto se ha utilizado con la finalidad de redactar la memoria.

Adobe Acrobat Reader

Se trata de una familia de programas desarrollados por Adobe diseñados con la finalidad de permitir visualizar documentos de texto con múltiples sistemas operativos sin que se modifique el aspecto ni pierda la estructura del documento original.

Para ello utilizan el formato Portable Document Format que se caracteriza porque:

- Puede contener cualquier combinación elementos de texto y multimedia.
- No pierden el formato con el envío a otros usuarios.
- Es uno de los formatos más extendidos en Internet para el intercambio de documentos.
- La mayoría de las herramientas de edición de texto permiten convertir los documentos a este formato.
- Puede cifrarse para proteger su contenido e incluso firmarlo digitalmente.
- Es el estándar ISO para ficheros contenedores de documentos electrónicos con vistas a su preservación.

En este proyecto se ha utilizado con el propósito de presentar la versión final de la documentación, así como para consultar múltiples ficheros en este formato a modo de referencias.

Draw.io

[Draw.io](#) es una aplicación web que permite crear diagramas online sin necesidad de descargar ningún software ni de adquirir licencias.

Se caracteriza porque:

- Es una herramienta muy intuitiva.
- Incluye múltiples plantillas que se pueden utilizar como guía.
- Dispone de una gran variedad de figuras para diseñar los diagramas.
- Permite insertar imágenes externas en los diagramas.
- Permite exportar y guardar el diagrama en diferentes formatos de mapa de bits y en XML para seguir editando más adelante.
- Lo haces todo online, sin necesidad de descargar e instalar ningún software.
- La aplicación es gratuita, por lo que no necesitas ninguna licencia.

En este proyecto se ha utilizado con el objetivo de efectuar los diagramas de clases y secuencia que se encuentran en el apartado de diseño de la documentación.

Paint

Se trata de un programa editor de imágenes que permite abrir y guardar archivos en formato de mapa de bits con la finalidad de crear dibujos sencillos o efectuar ediciones simples sobre imágenes preexistentes.

Esta herramienta de dibujo ha acompañado al sistema operativo Microsoft Windows desde sus primeras versiones de forma gratuita.

Su principal característica es su simplicidad frente a otras herramientas del mismo tipo.

En este proyecto se ha utilizado para retocar algunas de las imágenes presentes en la memoria y diseñar las imágenes que se encuentran en la aplicación.

Programas

XAMPP

[XAMPP](#) es un paquete gratuito de software libre con licencia GNU que se utiliza como un servidor web libre.

Está constituido por:

- El sistema de gestión de bases de datos MariaDB.
- El servidor web Apache.
- Los intérpretes para lenguajes de script PHP y Perl.

Se caracteriza porque es fácil de usar y capaz de interpretar páginas dinámicas.

En este proyecto se ha empleado con la finalidad generar un servidor virtual que permitiera ejecutar la aplicación en localhost. De este modo no era necesario subir la aplicación al servidor tras cada modificación, simplificando considerablemente el proceso de depuración de errores

Anaconda

[Anaconda](#) es una distribución de paquetes de código abierto para Python que permite instalar y administrar dependencias y entornos de manera sencilla.

Se caracteriza porque:

- Es Libre y de código abierto.
- Tiene una documentación detallada y una gran comunidad.
- Tiene integrada una interfaz gráfica de usuario sencilla, pero versátil.
- Permite eliminar problemas de dependencia de paquetes y control de versiones.
- Permite compilar código.
- Facilita la escritura de complejos algoritmos.

En este proyecto se ha aplicado para redactar y compilar el código de la aplicación.

Spyder

[Spyder](#) es un entorno gratuito de código abierto para Python que dispone de funciones avanzadas de edición, análisis, depuración y creación de código.

Es uno de los paquetes que vienen instalados por defecto con Anaconda.

En este proyecto se ha empleado con el fin de redactar y compilar el código de la aplicación.

Notepad++

[Notepad++](#) es un editor de código fuente gratuito de software libre con licencia GNU cuya finalidad es ser un reemplazo del Bloc de notas tradicional.

Esta herramienta dispone de librerías que facilitan la redacción del Código añadiendo funcionalidades como:

- Remarcar de forma distintiva las palabras clave del lenguaje que se esté utilizando.
- Sugerencias automáticas y Autocompletado de palabras clave del lenguaje e incluso nombres de funciones
- Indentación automática del código.

En este proyecto se ha utilizado con el fin de redactar el código de la aplicación que no utilizaba lenguaje Python: SQL, XML,

7Zip

[7Zip](#) es un software gratuito de código abierto bajo la licencia BSD que permite empaquetar y desempaquetar cualquier tipo de ficheros con una elevada relación de compresión

Se caracteriza porque:

- Es compatible con muchos formatos de compresión entre los que se encuentran Zip y Rar.
- Incluye un administrador de archivos.

En este proyecto se ha utilizado con el objetivo de empaquetar el proyecto para su envío y almacenamiento

LogoMakr

[LogoMakr](#) es un software gratuito que ofrece una herramienta online con la que se pueden crear, editar y guardar logotipos.

En este proyecto se ha utilizado para crear el [Logotipo de la página web](#).

Servicios

OneDrive

[OneDrive](#) es la plataforma en la nube de Microsoft. Permite guardar en línea todo tipo de documentos con la finalidad de poder acceder a ellos desde cualquier equipo que disponga de conexión a Internet.

Sus principales características son:

- Recibes 7 GB de espacio de almacenamiento gratuito solo por hacerte una cuenta de Microsoft.
- Se realizan copias de seguridad internas de forma transparente para el usuario.

En este proyecto se ha utilizado con el propósito de almacenar toda la información del proyecto. Tanto la memoria como el código.

Google Crome

[Google Crome](#) es el navegador web gratuito desarrollado por Google.

En este proyecto se ha utilizado:

- Para realizar todas las pruebas y ejecuciones de la aplicación web.
- Como la principal herramienta de navegación por la web.

Gmail

[Gmail](#) es el correo electrónico gratuito desarrollado por Google.

En este proyecto se ha utilizado con la finalidad de comunicarse con el tutor del proyecto.

Yahoo Finance

[Yahoo Finance](#) es un servicio de Yahoo! que proporciona información financiera como las cotizaciones de empresas o índices bursátiles.

En este proyecto se ha utilizado como objetivo del Web Scraper para obtener la última cotización de los Activos solicitados por el usuario.

Alphavantage

[Alphavantage](#) es una API ([Interfaz de programación de aplicaciones](#)) que permite obtener los datos financieros de una gran variedad de activos en formato de series temporales diarias, semanales o mensuales que contienen más de 20 años de datos históricos.

En este proyecto se ha empleado como buscador para obtener las cotizaciones históricas de los Activos solicitados por el usuario.

Patreon

[Patreon](#) es una plataforma que ofrece una alternativa de monetización para proyectos independientes basada en donaciones, donde el creador de contenido pone a disposición de sus fans varios planes de suscripción mensual en los que se les ofrecen diversas bonificaciones, con la finalidad de implicarlos en el proyecto.

Dichas bonificaciones pueden ir desde tener preferencia en el acceso al contenido o disfrutar de contenido exclusivo, hasta participar directamente en la toma de decisiones o la administración del proyecto.

De este modo, Patreon permite a los creadores de contenido obtener cierto grado de estabilidad económica gracias a los ingresos recurrentes que aportan los mecenas. Al mismo tiempo que ofrece un sistema para influir en el desarrollo del proyecto a todos aquellos que estén interesados en patrocinarlo.

En este proyecto se ha utilizado como método de monetización del proyecto.

Google AdSense

[Google AdSense](#) Se trata de un servicio desarrollado por Google que permite a los editores obtener ingresos mediante la colocación de anuncios en sus sitios web. Para seleccionar los anuncios se utilizan una serie de tecnologías que permiten mostrar anuncios relevantes a los visitantes, lo que garantiza a los anunciantes llegar al público que buscan y a los usuarios disfrutar de contenido útil.

En este proyecto se ha aplicado como método de monetización del proyecto.

Azure

[Azure](#) es un servicio de computación en la nube desarrollado por Microsoft que permite construir, probar, desplegar y administrar aplicaciones y servicios web en muchos lenguajes de programación diferentes.

Para ello proporcionan acceso a una red de servidores remotos conectados a internet donde se puede almacenar, administrar y procesar datos, servidores, bases de datos, redes y software.

Entre los distintos servicios que ofrecen está incluido un sistema de copias de seguridad que garantiza el funcionamiento continuo de las aplicaciones y servicios desplegados.

En este proyecto se ha empleado como servicio de computación en la nube para poner el proyecto a disposición de los usuarios.

Avast

[Avast](#) es un software antivirus que dispone de una licencia de uso gratuita que ofrece protección contra el software malicioso, protección de la privacidad en Internet, protección contra el Fising el Spyware y el spam.

En este proyecto se ha utilizado como software antivirus por defecto.

Metodología

Este apartado de la documentación está destinado a exponer el planteamiento inicial que se utilizó como punto de partida. De este modo se incluirán todos los esquemas, hipótesis y decisiones de planificación, análisis y diseño que se tomaron como óptimos para cumplir los objetivos del proyecto.

Planificación

Evaluación del Trabajo

Descomposición del Trabajo

Para poder abordar un proyecto de esta magnitud y conseguir un resultado acorde a las expectativas debemos dividirlo en tareas más pequeñas. Aunque pueda parecer que esto solo añade una mayor carga de trabajo, en realidad nos permite:

- Comprobar y cuantificar nuestro progreso.
- Evitar la procrastinación. ([Las tareas grandes son más difíciles de abordar porque resultan intimidantes](#))
- Solucionar errores antes de que se cometan mediante la detección de las tareas más conflictivas.
- Reorganizar la carga de trabajo ante cualquier incidencia.

Las tareas que se han identificado para este proyecto son:

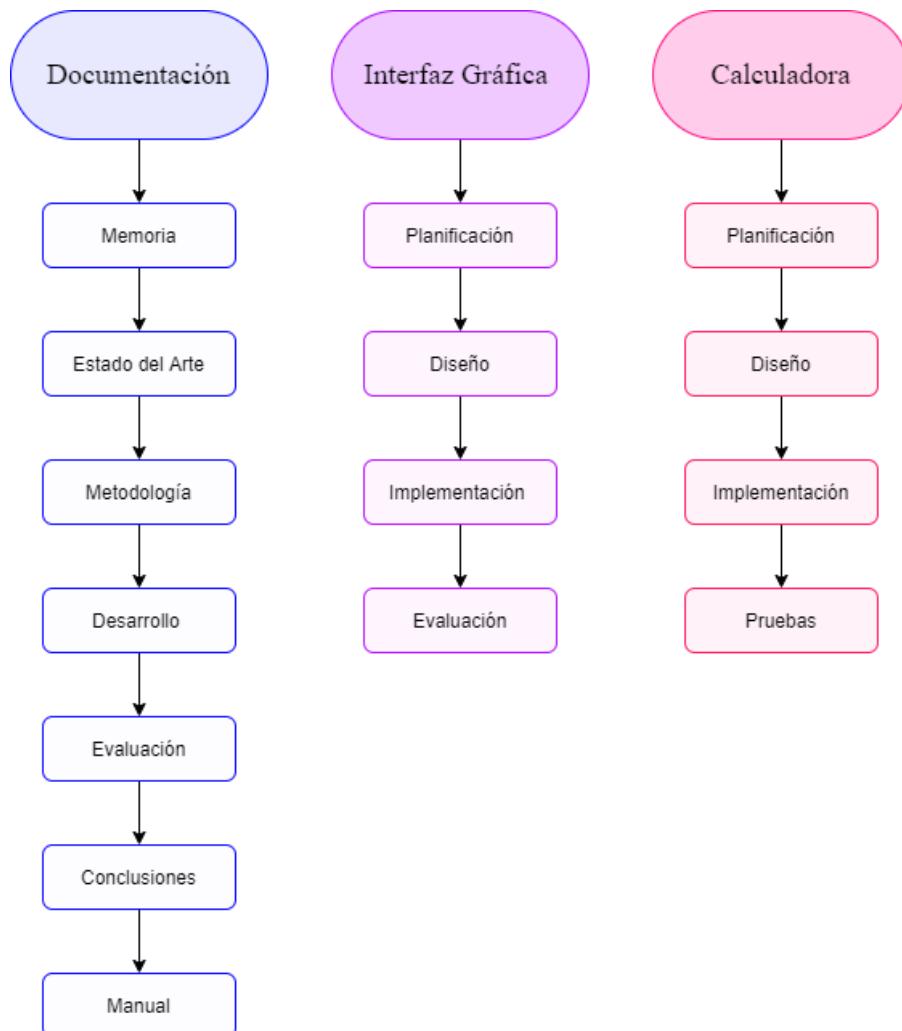


Ilustración 37 División de tareas – Parte 1

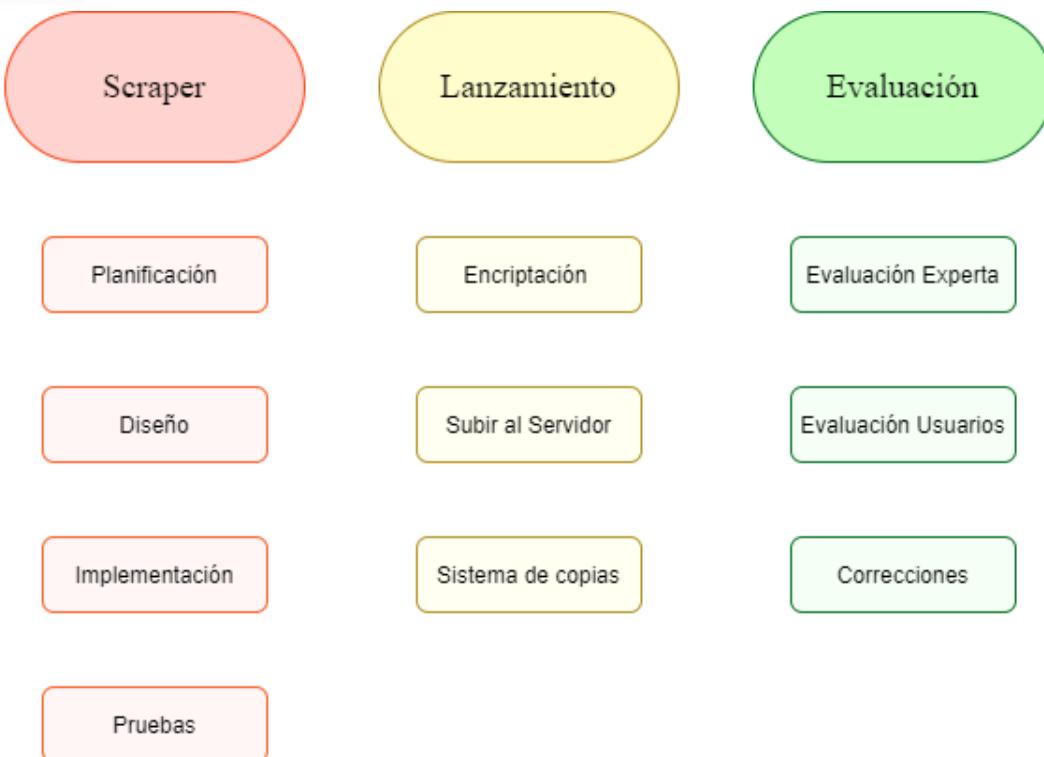


Ilustración 38 División de tareas – Parte 2

Documentación

El objetivo de esta fase es la recopilación de toda la información relacionada con el proyecto. Con la finalidad de garantizar el correcto seguimiento y entendimiento del proceso de desarrollo de esta aplicación.

La documentación está dividida en las siguientes tareas:

- **Memoria:** En este apartado se presenta el proyecto definiendo la situación inicial de la que se parte y los objetivos que se pretenden alcanzar.
- **Estado del Arte:** En este apartado se recopilarán distintos conceptos necesarios para comprender el funcionamiento de la aplicación, así como el motivo por el que se han aplicado determinadas decisiones de diseño.
- **Metodología:** En este apartado se recopilarán los resultados finales de las fases de planificación y diseño de todas las tareas que los requieran.
- **Desarrollo:** En este apartado se describirá cómo ha evolucionado cada fase del proyecto durante su etapa de implementación, analizando cómo han influido los eventos más notables que hayan tenido lugar en las decisiones que se tomaron inicialmente.
- **Evaluación:** En este apartado se recapitularán todas las técnicas empleadas, así como los resultados esperados y obtenidos para cada una de las pruebas que se han realizado.
- **Manual:** En este apartado se redactará una guía en la que se explique con el máximo detalle cómo utilizar cada una de las funcionalidades de la aplicación.
- **Conclusiones:** En este apartado se analizará si la aplicación ha obtenido los resultados esperados y se describirán las principales dificultades que se han presentado y las experiencias que se han extraído de la elaboración de este proyecto.

Interfaz gráfica

El objetivo de esta fase es obtener una interfaz de usuario acorde con las funcionalidades previstas.

La interfaz gráfica debe permitir al usuario acceder a todas las funcionalidades del modelo de forma cómoda e intuitiva sin que este tenga que preocuparse por cuál es su funcionamiento interno.

La Interfaz gráfica está dividida en las siguientes tareas:

- **Planificación:** En este apartado se definirán los requisitos que debe cumplir la interfaz gráfica.
- **Diseño:** En este apartado se definirá la estructura que deberá presentar la interfaz gráfica para satisfacer los requisitos previamente definidos.
- **Implementación:** En este apartado se redactará el código de la interfaz gráfica ateniéndose al diseño previsto en la medida de lo posible.
- **Evaluación:** En este apartado se realizarán las comprobaciones necesarias para verificar que la interfaz gráfica satisface las necesidades para las que había sido diseñada.

Calculadora

El objetivo de esta fase es obtener el modelo de la aplicación acorde con las funcionalidades previstas.

El modelo será el responsable de realizar todas las funcionalidades internas de la aplicación de forma transparente para el usuario.

La calculadora está dividida en las siguientes tareas:

- **Planificación:** En este apartado se definirán los requisitos que debe cumplir el modelo.
- **Diseño:** En este apartado se definirá la estructura que deberá presentar el modelo para satisfacer los requisitos previamente definidos.
- **Implementación:** En este apartado se redactará el código del modelo ateniéndose al diseño previsto en la medida de lo posible.
- **Evaluación:** En este apartado se realizarán las comprobaciones necesarias para verificar que el modelo satisface las necesidades para las que había sido diseñado.

Scraper

El objetivo de esta fase es obtener una herramienta que nos permita obtener de la Web la información necesaria para el correcto funcionamiento del modelo.

El Scraper está dividido en las siguientes tareas:

- **Planificación:** En este apartado se definirán los requisitos que debe cumplir el modelo.
- **Diseño:** En este apartado se definirá la estructura que deberá presentar el Scraper para satisfacer los requisitos previamente definidos.
- **Implementación:** En este apartado se redactará el código del Scraper ateniéndose al diseño previsto en la medida de lo posible.
- **Evaluación:** En este apartado se realizarán las comprobaciones necesarias para verificar que el Scraper satisface las necesidades para las que había sido diseñado.

Lanzamiento

El objetivo de esta fase es desplegar la aplicación en un entorno real para que pueda ser utilizada.

El lanzamiento está dividido en las siguientes tareas:

- **Encriptación:** En este apartado se realizarán las modificaciones necesarias sobre el sistema de almacenamiento de datos con la finalidad de que los usuarios sólo tengan accesibles su propia información.
- **Subir al servidor:** En este apartado se desplegará la aplicación en un servidor web.
- **Sistema de copias:** En este apartado se configurará un sistema de copias de seguridad que permita mantener duplicados los datos con la finalidad de minimizar el riesgo de pérdida de información.

Evaluación

El objetivo de esta fase es verificar que la aplicación en su conjunto satisface correctamente las funcionalidades para las que ha sido diseñada.

La evaluación estará dividida en las siguientes tareas:

- **Evaluación experta:** En este apartado evaluaré personalmente la funcionalidad final de la aplicación.
- **Evaluación de usuarios:** En este apartado se dará a probar la aplicación a varios sujetos de prueba registrando su experiencia.
- **Correcciones:** En este apartado se tratará de corregir todos los errores detectados en la fase de evaluación.

Estructura de Requisitos

Debido a las dependencias que existen entre algunas de las tareas definidas, será necesario establecer un orden de ejecución.

El proyecto comenzará con la memoria de la documentación y seguirá con la redacción del estado del arte.

A continuación, se planificará y diseñará la aplicación ([Interfaz gráfica, Scraper y Modelo](#)) documentando en el apartado de metodología de la documentación las razones por las que se han tomado las distintas decisiones de diseño.

Seguidamente se procederá con la implementación de la aplicación ateniéndose a los diseños previamente planificados y documentando en el apartado de desarrollo de la documentación todos los incidentes que se produzcan describiendo cómo han afectado al diseño previsto.

Una vez terminada cada una de las piezas de la aplicación se efectuarán las pruebas unitarias con su respectiva documentación.

Si todas las partes funcionan correctamente se iniciará la fase de despliegue de la aplicación donde se encriptará la información permanente, se subirá la aplicación al servidor y se configurará el sistema de copias.

Finalmente se realizará la evaluación de la aplicación final por parte de los expertos y los usuarios efectuando las correcciones que se consideren oportunas en función de los resultados obtenidos.

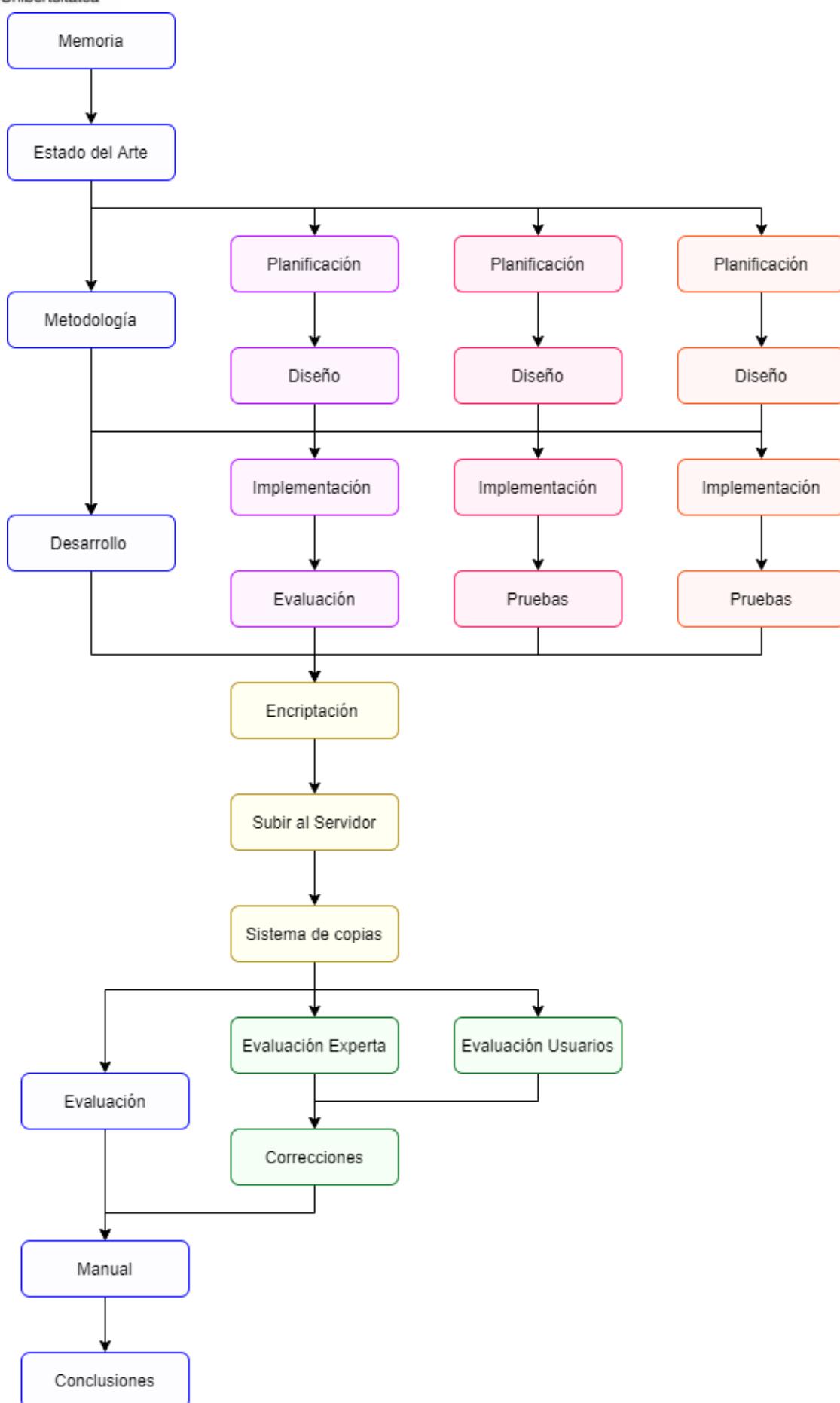


Ilustración 39 Estructura de Requisitos

Evaluación Temporal

El objetivo de la evaluación temporal es definir la duración del proyecto con la finalidad de extrapolar la fecha de finalización y estimar los costes variables.

Carga de trabajo

Determinar la carga de trabajo consiste en valorar el volumen de tareas que una persona puede realizar en un determinado periodo de tiempo. Se trata de una decisión muy importante en la planificación de un proyecto, puesto que una carga de trabajo mal equilibrada entorpecerá el flujo de trabajo.

- Si es demasiado baja retrasará la finalización del proyecto, incrementando los costes variables.
- Si es demasiado alta puede limitar la productividad y creatividad de los trabajadores.

La estimación de la carga de trabajo para este proyecto será de quince horas semanales concentradas uniformemente en tres días. Esta carga de trabajo se coordinará con las prácticas de empleo, que suponen veintiuna horas semanales concentradas uniformemente en tres días diferentes a los asignados para el proyecto.

Estimación de Duraciones

Una vez tenemos definidas todas las tareas que conforman el proyecto, realizaremos una estimación aproximada de la duración de cada tarea basándonos en la experiencia obtenida de proyectos anteriores:

Diagrama Gantt

El diagrama de Gantt es un tipo de gráfico que representa las actividades de forma independiente con el objetivo de tener una imagen general de cómo evolucionan las tareas a través del tiempo.

Para obtener el diagrama de Gantt utilizaremos la estructura de requisitos para repartir las horas estimadas en un calendario que respete la carga de trabajo estipulada.

Num	Tarea	Antecedentes	Duración Estimada
1	Documentación		200
	1.1 Memoria	0	30
	1.2 Estado del Arte	1.1	50
	1.3 Metodología	1.2	50
	1.4 Desarrollo	1.3 / 2.2 / 3.2 / 4.2	30
	1.5 Evaluación	1.4 / 2.3 / 3.3 / 4.3	30
	1.6 Conclusiones	6.3	10
2	Interfaz Gráfica		300
	2.1 Planificación	1.2	50
	2.2 Diseño	2.1	100
	2.3 Implementación	2.2	100
	2.4 Evaluación	2.3	50
3	Calculadora		300
	3.1 Planificación	1.2	50
	3.2 Diseño	3.1	100
	3.3 Implementación	3.2	100
	3.4 Evaluación	3.3	50
4	Scraper		90
	4.1 Planificación	1.2	10
	4.2 Diseño	4.1	50
	4.3 Implementación	4.2	20
	4.4 Evaluación	4.3	10
5	Lanzamiento		230
	5.1 Encriptación	2.4 / 3.4 / 4.4	100
	5.2 Subir al servidor	5.1	30
	5.3 Sistema de copias	5.2	100
6	Evaluación		200
	6.1 Evaluación experta	5.3	50
	6.2 Evaluación Usuarios	5.3	50
	6.3 Correcciones	6.1 / 6.2	100

Ilustración 40 Estimación de duraciones

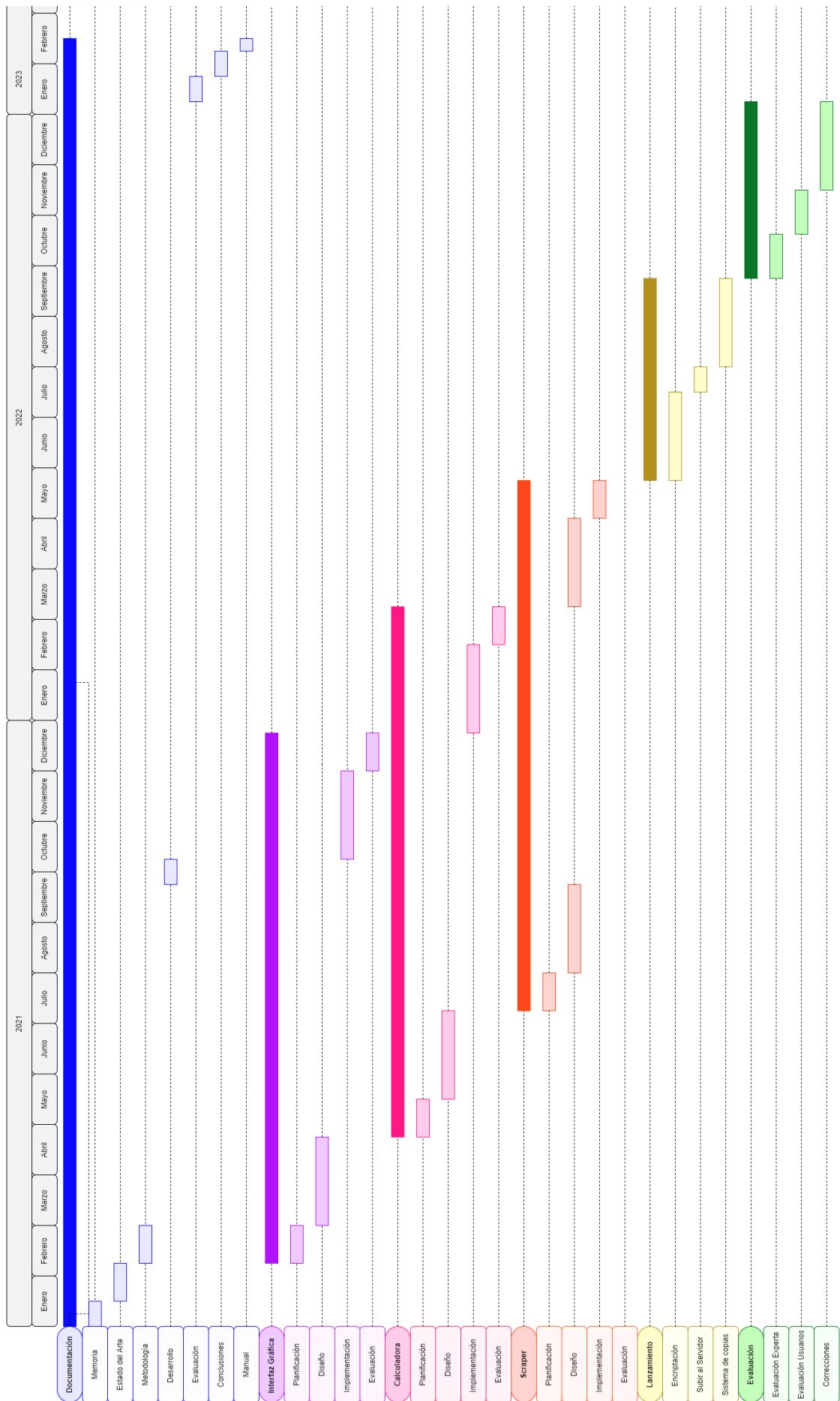


Ilustración 41 Diagrama de Gantt

Evaluación Económica

El objetivo de la evaluación económica es estimar los recursos que se han empleado para la elaboración del proyecto, así como los beneficios potenciales que se obtendrán debido a su explotación.

Modelo de negocio

La monetización de este proyecto no se realizará mediante la tradicional venta de una licencia de uso sobre la aplicación. Debido a que la finalidad de este proyecto es que llegue a aquellas personas que no realizan ningún control sobre sus finanzas personales, imponer un coste de entrada para el uso de la aplicación resultaría ser una medida disuasiva para el público objetivo.

Se proponen las siguientes alternativas para la monetización del proyecto:

- **Patreon:** Gracias al flexible sistema de usuarios que implementa la aplicación, sería posible crear herramientas avanzadas para aquellos usuarios que estuvieran dispuestos a patrocinar el proyecto. Pudiéndose crear los siguientes perfiles de Patreon.

(Nota: Un perfil superior siempre incluye todas las ventajas de los perfiles inferiores)

- **Básico:** Te ha gustado la iniciativa del proyecto y quieres invitar a los desarrolladores a un aperitivo sin implicarte en su desarrollo.
 - Donando 1€ al mes.
 - Tu nombre aparecerá en los créditos del proyecto.
 - No te saldrán anuncios en la aplicación.
- **Avanzado:** Quieres participar activamente en el desarrollo del proyecto.
 - Donando 5€ al mes.
 - Tendrás acceso a un chat donde se debatirá, junto con los desarrolladores, acerca del desarrollo de la aplicación.
 - Tendrás acceso a la aplicación en el servidor de desarrollo donde se testean las nuevas funcionalidades. **(Aviso: Esta versión podría ser inestable)**
- **Premium:** Quieres disfrutar de ventajas adicionales en la aplicación.
 - Donando 10€ al mes.
 - Tendrás un perfil de moderador en la aplicación con acceso a funcionalidades avanzadas.

- **Anuncios:** Mediante herramientas como Google AdSense se pondrán anuncios en la aplicación para todos aquellos usuarios que no sean patrocinadores.

Google AdSense paga en función de la cantidad de clicks que consigas generar en los banners y de su calidad. Según [ingreso digital](#), se podría estimar unos ingresos de 10€ al mes para una web que consiga generar una media de 100 visitas al día.

/s This month Last month Jan 1 – 15, 2021 + add comparison

SITE	↓	Estimated earnings *	Page views	Page RPM	Impressions
🕒 All		€6,276.91	1,811,197	€3.47	5,687,536
Average		€448.35	129,371	–	406,252
🕒 [REDACTED].com		€3,575.93	789,561	€4.53	2,701,400
🕒 [REDACTED].com		€1,556.96	221,239	€7.04	946,354

Ilustración 42 Ejemplo monetización con Google AdSense

Estimación de costes

El coste de la aplicación se calculará mediante el sumatorio de los costes individuales de todos los recursos que se han empleado durante la realización del proyecto:

Deberemos considerar los siguientes costes

- **Mano de obra:** En base al número de horas estimadas en la evaluación temporal del proyecto y considerando un sueldo de 7€/hora para un único empleado.

$$C_{\text{Mano de obra}} = T_{\text{Duracion del proyecto}} * \text{Salario} = 1.320 \text{ horas} * 7\text{€/ hora} = 9.240\text{€}$$

A esta estimación hay que sumarle un 30% como impuestos para la seguridad social del trabajador:

$$9.240\text{€} * 30/100 = 2.772\text{€}$$

- **Licencias:** Las licencias empleadas en este proyecto son casi todas gratuitas a excepción de las siguientes:

- **Microsoft Office:** Venía de serie con el sistema operativo del ordenador, de modo que este gasto estará reflejado en la sección de amortizaciones.
- **OneDrive:** Se ha empleado la licencia familiar de almacenamiento en la nube que ofrece de 1Tb para cinco cuentas personales con un coste de 99€/año. Debido a que en la planificación temporal estima una duración de dos años, el coste incurrido será de 39,60€

$$99\text{ €/año} * 2 \text{ años} \div 5 \text{ usuarios} = 39,60 \text{ €/Usuario}$$

- **Azure:** El servicio de Azure cuenta con una licencia de prueba gratuita con duración limitada que se estima que será suficiente para la fase desarrollo y despliegue inicial de la aplicación.

- **Amortizaciones:** Los periodos de amortización se han obtenido de la [Agencia Tributaria](#)

- **Equipo informático:** La amortización del ordenador con el que se ha efectuado la totalidad del trabajo asciende a 176,50€/año o 14,70€/mes.
 - La amortización de cualquier dispositivo electrónico utilizado para sistemas y programas informáticos no superará el 33% de su valor anual con un máximo de 6 años
 - El equipo fue adquirido el 10/06/2016 por un importe de 1058,99€.

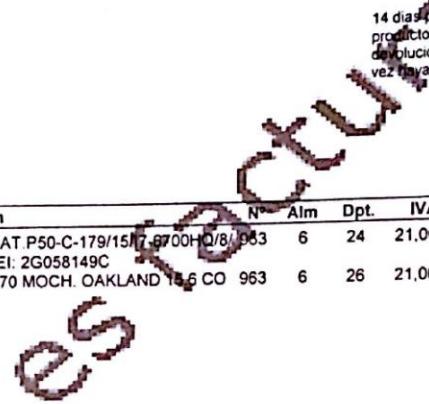
Tel.: 963425600	Fax: 963425609																																																									
Venta al Contado	Nº: 40570430	Fecha: 10/06/2016	Fecha de Entrega: 10/06/2016																																																							
Documento Interno																																																										
Tipo de Entrega: Llevar																																																										
Vendedor: 12556																																																										
																																																										
Dir. Factura: Nº Cliente 900354901 SEÑOR CUESTA ALARIO DAVID X ES X X CIF: 20857516N Móvil: 619700143																																																										
 <i>14 días para cambio o devolución, siempre con producto y embalaje en perfecto estado. No tienen devolución las configuraciones, software o tinta una vez hayan sido desprecintados.</i>																																																										
<table border="1"> <thead> <tr> <th>Pos.</th><th>Cant.</th><th>Nº FDP.</th><th>Nº Art.</th><th>Designación</th><th>Nº Alm</th><th>Dpt.</th><th>IVA</th><th>PVP/Unida</th><th>Total</th></tr> </thead> <tbody> <tr> <td>1</td><td>1</td><td>380</td><td>1305509</td><td>TOSHIBA SAT P50-C-179/15M7-8700HD/8/053</td><td>6</td><td>24</td><td>21,00</td><td>979,00</td><td>979,00 EUR</td></tr> <tr> <td></td><td></td><td></td><td></td><td>Nº Serie/IMEI: 2G058149C</td><td></td><td></td><td></td><td></td><td></td></tr> <tr> <td>2</td><td>1</td><td>3600</td><td>1247377</td><td>PORT 110270 MOCH. OAKLAND 146 CO 963</td><td>6</td><td>26</td><td>21,00</td><td>79,99</td><td>79,99 EUR</td></tr> </tbody> </table> <table border="1"> <tr> <td colspan="5">Registro Mercantil Valencia, Tomo 7375, Folio 15, Sección 8, Hoja V-87572, Libro 4676, Inscripción 2</td> </tr> <tr> <td>IVA</td><td>Total Neto</td><td>Total IVA</td><td>Total Bruto</td><td>Importe Tot</td></tr> <tr> <td>21,00 %</td><td>875,20 EUR</td><td>183,79 EUR</td><td>1058,99 EUR</td><td>1058,99 EUR</td></tr> </table>				Pos.	Cant.	Nº FDP.	Nº Art.	Designación	Nº Alm	Dpt.	IVA	PVP/Unida	Total	1	1	380	1305509	TOSHIBA SAT P50-C-179/15M7-8700HD/8/053	6	24	21,00	979,00	979,00 EUR					Nº Serie/IMEI: 2G058149C						2	1	3600	1247377	PORT 110270 MOCH. OAKLAND 146 CO 963	6	26	21,00	79,99	79,99 EUR	Registro Mercantil Valencia, Tomo 7375, Folio 15, Sección 8, Hoja V-87572, Libro 4676, Inscripción 2					IVA	Total Neto	Total IVA	Total Bruto	Importe Tot	21,00 %	875,20 EUR	183,79 EUR	1058,99 EUR	1058,99 EUR
Pos.	Cant.	Nº FDP.	Nº Art.	Designación	Nº Alm	Dpt.	IVA	PVP/Unida	Total																																																	
1	1	380	1305509	TOSHIBA SAT P50-C-179/15M7-8700HD/8/053	6	24	21,00	979,00	979,00 EUR																																																	
				Nº Serie/IMEI: 2G058149C																																																						
2	1	3600	1247377	PORT 110270 MOCH. OAKLAND 146 CO 963	6	26	21,00	79,99	79,99 EUR																																																	
Registro Mercantil Valencia, Tomo 7375, Folio 15, Sección 8, Hoja V-87572, Libro 4676, Inscripción 2																																																										
IVA	Total Neto	Total IVA	Total Bruto	Importe Tot																																																						
21,00 %	875,20 EUR	183,79 EUR	1058,99 EUR	1058,99 EUR																																																						

Ilustración 43 Factura del equipo informático

- **Gastos corrientes:** En esta sección se incluyen todos los gastos incurridos debido a las actividades ordinarias realizadas a lo largo de la duración del proyecto.

- o **Factura de internet:** $300\text{€/año} * 2 \text{ años} = 600\text{€}$
- o **Factura de luz:** $100\text{€/trimestre} * 4 \text{ Trimestres / año} * 2 \text{ años} = 800\text{€}$

Sumando todos los gastos incurridos en el proyecto, obtenemos unos gastos totales de 13.804,60€ a los que sería prudente sumar un treinta por ciento adicional como margen de seguridad ante posibles imprevistos.

$$13.804,60\text{€} * 30/100 = 4.141,38\text{€}$$

Con esto el coste total estimado asciende a 17.945,98€ que podemos redondear a 18.000€, ampliando el margen de seguridad, con la finalidad de presentar un número redondo a los posibles inversores.

Estimación de Beneficios

En este apartado se analizarán una serie de escenarios hipotéticos, con la finalidad de estimar cuántos ingresos debería generar nuestro modelo de negocio para obtener unos beneficios equivalentes sobre los gastos incurridos.

Para realizar estas estimaciones, se considera razonable obtener una rentabilidad del 10% anual. por lo que nuestro modelo de negocio debería de ser capaz de generar mil ochocientos euros anuales o ciento cincuenta euros mensuales.

$$18.000\text{€} * 8/100 = 1.800 \text{ €/año} \div 12 \text{ mes/año} = 150\text{€/mes}$$

En el escenario hipotético de que todos los ingresos provinieran de mecenas de Patreon, con los tipos de suscripciones consideradas serían necesarios al menos cincuenta suscriptores básicos, diez avanzados y cinco premium.

$$50 Sub_{basicos} * 1\text{€/mes} + 10 Sub_{Avanzados} * 5\text{€/mes} + 5 Sub_{Premium} * 10\text{€/mes} = 150\text{€/mes}$$

En el escenario hipotético de que todos los ingresos provinieran de los anuncios de Google AdSense sería necesario generar un tráfico en la web equivalente a mil quinientas visitas diarias

$$100 Visitas/Dia * 150\text{€/mes} \div 10\text{€/mes} = 1.500 Visitas/Dia$$

En el escenario hipotético de que se utilizasen ambos métodos de monetización de forma simultánea, una proporción que parece razonable sería la siguiente:

- 35 suscriptores básicos.
- 5 suscriptores avanzados.
- 2 suscriptores premium.
- 700 visitas diarias adicionales. ([Los suscriptores están exentos de publicidad](#))

$$35 Sub_{basicos} * 1\text{€/mes} + 5 Sub_{Avanzados} * 2\text{€/mes} + 5 Sub_{Premium} * 10\text{€/mes} = 80\text{€/mes}$$

$$100 Visitas/Dia * 10\text{€/mes} \div 700 Visitas/Dia = 70 \text{ €/mes}$$

Evaluación de Riesgos

En este apartado se analizarán todos aquellos escenarios que podrían perjudicar el desarrollo del proyecto, estimando la probabilidad de que dichos eventos sucedan y valorando el impacto que podrían suponer.

A continuación, se desarrollará un plan de prevención o contingencia que permita minimizar con un coste razonable la probabilidad o el impacto que dichos escenarios podrían suponer.

En el peor de los casos, la gestión de riesgos nos permitirá saber cómo debemos actuar si sucede uno de los riesgos documentados.

Riesgos Sociales

Contraer alguna enfermedad física o psicológica:

- **Riesgo:**
 - Retrasar la realización de las tareas en curso.
 - No completar todos los objetivos del proyecto en el plazo establecido.
- **Impacto:** Reducido.
- **Probabilidad:** Elevada.
- **Prevención:**
 - Mantener las condiciones de higiene recomendadas por los expertos.
 - Respetar los horarios de sueño recomendados por los expertos.
 - Llevar una alimentación saludable.
 - Dedicar tiempo a actividades de ocio para mantener bajos los niveles de estrés.
- **Contingencia:**
 - Se paralizará la actividad del proyecto hasta que se estime conveniente.
 - Se reorganizará la planificación temporal.
- **Costes:**
 - El coste de prevención es mínimo y reduce considerablemente las probabilidades de dicho riesgo.
 - El coste de contingencia es reducido.

Problemas en el entorno social:

- **Riesgo:**
 - Retrasar la realización de las tareas en curso.
 - No completar todos los objetivos del proyecto en el plazo establecido.
- **Impacto:** Reducido.
- **Probabilidad:** Elevada.
- **Prevención:** Se trata de problemas ajenos a mi persona, por lo que no son prevenibles.
- **Contingencia:**
 - Se paralizará la actividad del proyecto hasta que se estime conveniente.
 - Se reorganizará la planificación temporal.
- **Costes:** El coste de contingencia es reducido.

Riesgos de Diseño

Evaluar incorrectamente el coste de implementación de alguna funcionalidad

- **Riesgo:**

- Retrasar la realización de las tareas en curso.
- No completar todos los objetivos del proyecto en el plazo establecido.

- **Impacto:** Medio.

- **Probabilidad:** Media.

- **Prevención:**

- Tener en cuenta mis capacidades y limitaciones a la hora de estimar los costes de implementación.
- Utilizar márgenes de seguridad a la hora de realizar la evaluación temporal y económica.
- No seleccionar objetivos demasiado ambiciosos para un trabajo de fin de grado.
- Realizar una implementación modular, que permita prescindir de ciertas funcionalidades sin afectar a otros módulos de la aplicación.

- **Contingencia:**

- Se prescindirá de aquellos objetivos que se considere oportuno para no exceder el tiempo previsto
- Modificar la planificación del proyecto para excluir las funcionalidades seleccionadas.

- **Costes:**

- El coste de prevención es mínimo y reduce considerablemente las probabilidades y el impacto.
- El coste de contingencia es reducido.

Exceder el tiempo previsto para el desarrollo del proyecto.

- **Riesgo:** No completar todos los objetivos del proyecto en el plazo establecido.

- **Impacto:** Medio.

- **Probabilidad:** Elevada.

- **Prevención:**

- Tener en cuenta mis capacidades y limitaciones a la hora de estimar los costes de implementación.
- Utilizar márgenes de seguridad a la hora de realizar la evaluación temporal y económica.
- No seleccionar objetivos demasiado ambiciosos para un trabajo de fin de grado.
- Realizar una implementación modular, que permita prescindir de ciertas funcionalidades sin afectar a otros módulos de la aplicación.

- **Contingencia:**

- Se prescindirá de aquellos objetivos que se considere oportuno para no exceder el tiempo previsto.
- Modificar la planificación del proyecto para excluir las funcionalidades seleccionadas.

- **Costes:**

- El coste de prevención es mínimo y reduce considerablemente las probabilidades y el impacto.
- El coste de contingencia es reducido.

Riesgos de Implementación

Cambios en las condiciones de uso de los servicios empleados.

- **Riesgo:**
 - Reevaluar la planificación y el diseño del proyecto.
 - Modificar la implementación de las funcionalidades afectadas.
- **Impacto:** Medio.
- **Probabilidad:** Media.
- **Prevención:**
 - Conocer alternativas que permitan sustituir los servicios empleados.
 - Realizar una implementación modular, que permita modificar la implementación de las funcionalidades externas sin afectar a otros módulos de la aplicación.
- **Contingencia:**
 - Buscar alternativas que permitan sustituir los servicios empleados.
 - Evaluar si las alternativas disponibles son mejores que los servicios antiguos con las nuevas condiciones.
 - Modificar la planificación del proyecto para incluir los nuevos servicios.
- **Costes:**
 - El coste de prevención es mínimo y reduce considerablemente el impacto de dicho riesgo.
 - El coste de contingencia es asumible.

Pérdida total o parcial de información del proyecto debido a un fallo en el almacenamiento.

- **Riesgo:**
 - No completar todos los objetivos del proyecto en el plazo establecido.
 - Retrasar la realización de las tareas en curso.
 - Pérdida permanente de información del proyecto.
- **Impacto:** Elevado.
- **Probabilidad:** Reducida.
- **Prevención:**
 - Utilizar un sistema de almacenamiento en la nube.
 - Realizar copias de seguridad periódicas.
- **Contingencia:**
 - Recuperar la copia de seguridad más reciente.
 - Evaluar el alcance de los daños no recuperables
 - Modificar la evaluación temporal.
- **Costes:**
 - El coste de prevención es mínimo y reduce considerablemente la probabilidad de dicho riesgo.
 - El coste de contingencia es elevado.

Riesgos de Despliegue

Tiempos de carga elevados debidos a operaciones computacionalmente muy costosas.

- **Riesgo:**
 - o Modificar la implementación de las funcionalidades afectadas.
 - o Reducción del número de patrocinadores y el volumen de visitas.
- **Impacto:** Elevado.
- **Probabilidad:** Media.
- **Prevención:**
 - o Buscar alternativas que permitan minimizar los tiempos de ejecución para aquellas funcionalidades que se prevea que podrían consumir muchos recursos.
- **Contingencia:**
 - o Diseñar una animación de carga que permita a los usuarios saber que la aplicación no se ha congelado.
 - o Modificar la implementación para utilizar técnicas como las de precargar datos o utilizar el caché.
- **Costes:**
 - o Los costes de prevención y contingencia son bastante elevados, pero el riesgo de perder usuarios activos es demasiado elevado para no tenerlos en consideración.

Problemas con el servidor que ocasione la caída del sistema, impida el acceso a la aplicación por parte de los usuarios o provoque una pérdida de información permanente.

- **Riesgo:** Reducción del número de patrocinadores y el volumen de visitas.
- **Impacto:** Elevado.
- **Probabilidad:** Reducida.
- **Prevención:**
 - o Utilizar un servicio de computación en la nube que garantice el funcionamiento continuo de las aplicaciones y servicios desplegados y realice copias de seguridad continuas para los almacenados.
- **Contingencia:**
 - o Solucionar los problemas con el servidor a la mayor brevedad posible.
 - o Compensar a los usuarios afectados en función de la gravedad y el alcance de la incidencia.
- **Costes:**
 - o El coste de prevención es mínimo y reduce considerablemente la probabilidad de dicho riesgo.
 - o El coste de contingencia podría llegar a ser bastante elevado, pero el riesgo de perder usuarios activos es más costoso.

Riesgos Económicos

Gastos inesperados durante el desarrollo del proyecto o tras su finalización.

- **Riesgo:** No viabilidad económica del proyecto.
- **Impacto:** Elevado.
- **Probabilidad:** Reducida.
- **Prevención:** Utilizar un margen de seguridad razonable al realizar la evaluación económica.
- **Contingencia:**
 - o Reevaluar la viabilidad económica del proyecto.
 - o Reducir los gastos incurridos por otras partidas.
 - o Detener el funcionamiento de la aplicación para limitar las pérdidas.
- **Costes:**
 - o El coste de prevención es mínimo y reduce considerablemente el impacto de dicho riesgo.
 - o El coste de contingencia podría llegar a ser bastante elevado.

No alcanzar las cifras mínimas estimadas durante la planificación económica en cuanto al número de usuarios patrocinadores o al volumen de vistas diarias.

- **Riesgo:** No viabilidad económica del proyecto.
- **Impacto:** Elevado.
- **Probabilidad:** Reducida.
- **Prevención:**
 - o Implicar a los usuarios en el desarrollo del proyecto.
 - Consultar a los usuarios acerca de posibles mejoras en los servicios ofrecidos.
 - Tener en cuenta las opiniones de los usuarios.
- **Contingencia:**
 - o Realizar un plan de marketing que permita atraer nuevos usuarios a la aplicación.
 - o Detener el funcionamiento de la aplicación para limitar las pérdidas.
- **Costes:**
 - o El coste de prevención es mínimo y reduce considerablemente la probabilidad de dicho riesgo.
 - o El coste de contingencia podría llegar a ser bastante elevado e inevitable.

Pérdida, rotura o robo del equipo de trabajo.

- **Riesgo:**
 - Incremento de los gastos del proyecto.
 - Pérdida permanente de información del proyecto.
 - Retrasar la realización de las tareas en curso.
 - No completar todos los objetivos del proyecto en el plazo establecido.
- **Impacto:** Elevado.
- **Probabilidad:** Reducida.
- **Prevención:**
 - Almacenar adecuadamente las herramientas de trabajo cuando no se están utilizando.
 - No utilizar las herramientas de trabajo de forma indebida.
- **Contingencia:**
 - Evaluar el alcance de los daños no recuperables.
 - Reposición del equipo de trabajo perdido.
 - Recuperar la copia de seguridad más reciente.
 - Modificar la evaluación temporal.
- **Costes:**
 - El coste de prevención es mínimo y reduce considerablemente la probabilidad de dicho riesgo.
 - El coste de contingencia podría llegar a ser bastante elevado e inevitable.

Equipo informático infectado por software maligno.

- **Riesgo:**
 - Pérdida permanente de información del proyecto.
 - Retrasar la realización de las tareas en curso.
 - No completar todos los objetivos del proyecto en el plazo establecido.
- **Impacto:** Elevado.
- **Probabilidad:** Media.
- **Prevención:**
 - Realizar análisis completos del sistema de forma periódica.
 - Mantener el antivirus actualizado.
 - No acceder a sitios web sospechosos que pudieran contener software maligno.
 - Utilizar un sistema de almacenamiento en la nube.
 - Realizar copias de seguridad periódicas.
- **Contingencia:**
 - Evaluar el alcance de los daños no recuperables.
 - Reposición del equipo de trabajo perdido.
 - Recuperar la copia de seguridad más reciente.
 - Modificar la evaluación temporal.
- **Costes:**
 - El coste de prevención es mínimo y reduce considerablemente la probabilidad de dicho riesgo.
 - El coste de contingencia podría llegar a ser bastante elevado e inevitable.

Riesgos Legales

Violar derechos de autor.

- **Riesgo:**
 - Modificar la implementación de las funcionalidades afectadas.
 - Incrementar los costes económicos del proyecto al hacer frente a las posibles multas.
- **Impacto:** Reducido.
- **Probabilidad:** Reducida.
- **Prevención:**
 - Disponer de las licencias necesarias para todos los recursos empleados en el proyecto.
- **Contingencia:**
 - Eliminar el material sin licencia.
 - Obtener las licencias necesarias.
- **Costes:**
 - El coste de prevención es mínimo y reduce considerablemente la probabilidad de dicho riesgo.
 - El coste de contingencia es reducido.

Tratar incorrectamente los datos de usuario exponiendo datos sensibles.

- **Riesgo:**
 - Modificar la implementación de las funcionalidades afectadas.
 - Reducción del número de patrocinadores y el volumen de visitas.
 - Incrementar los costes económicos del proyecto al hacer frente a las posibles multas.
- **Impacto:** Alto.
- **Probabilidad:** Media.
- **Prevención:**
 - Seguir correctamente las directrices de la LOPD en cuanto al tratamiento de datos sensibles.
- **Contingencia**
 - Modificar la implementación de las funcionalidades afectadas a la mayor brevedad posible.
 - Compensar a los usuarios afectados en función de la gravedad y el alcance de la incidencia.
- **Costes:**
 - El coste de prevención es elevado, pero reduce considerablemente la probabilidad de dicho riesgo.
 - El coste de contingencia podría llegar a ser bastante elevado pero necesario.

Diseño

Este apartado de la documentación está destinado a describir la estructura que presenta cada una de las piezas de la aplicación, así como la interacción que tienen entre sí.

Jerarquía de actores

La jerarquía de actores permite definir cada uno de los tipos de usuario para los que está previsto que la aplicación pueda funcionar describiendo detalladamente las funcionalidades que podrá realizar:

- **Usuario Cualquiera:** Se trata de un tipo de usuario teórico que permite recopilar aquellas tareas que pueden ser realizadas por todos los tipos de usuario que se hayan definido.
- **Usuario No Identificado:** Se trata de aquel usuario que no se ha identificado en la aplicación utilizando un usuario y contraseña válidos.
 - o Podrá realizar todas las funcionalidades asignadas al usuario cualquiera además de las suyas propias.
- **Usuario Identificado:** Se trata de aquel usuario que si se ha identificado en la aplicación utilizando un usuario y contraseña válidos.
 - o Podrá realizar todas las funcionalidades asignadas al usuario cualquiera además de las suyas propias.
- **Usuario Administrador:** Se trata de aquel usuario que se ha identificado en la aplicación utilizando un usuario y contraseña pertenecientes a un administrador.
 - o Podrá realizar todas las funcionalidades asignadas al usuario identificado y cualquiera además de las suyas propias.

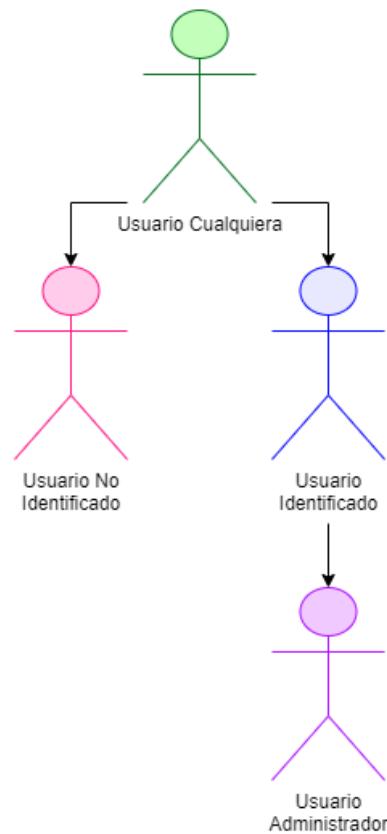


Ilustración 44 Jerarquía de Actores

Casos de Uso

Basándonos en los objetivos definidos podemos especificar las funcionalidades que deberán poder ser realizadas por cada uno de los tipos de actores que se han definido.

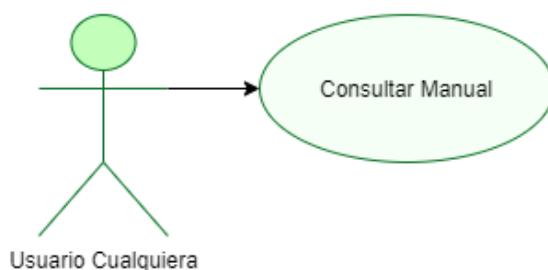


Ilustración 45 Casos de Uso de Cualquiera

Consultar Manual
Descripción: Permite a cualquiera acceder a una guía donde se expone el funcionamiento básico de la aplicación, así como una descripción detallada acerca de una gran variedad de conceptos sobre la gestión de finanzas personales.
Precondiciones: Ninguna
Flujo de eventos:
<ul style="list-style-type: none"> - Cualquiera pulsa el logotipo de la aplicación web para acceder a la página de inicio. - Se mostrarán varias etiquetas que permitirán navegar entre las distintas secciones del manual. - Cualquiera accede a la sección del manual deseada.
Postcondiciones: Ninguna

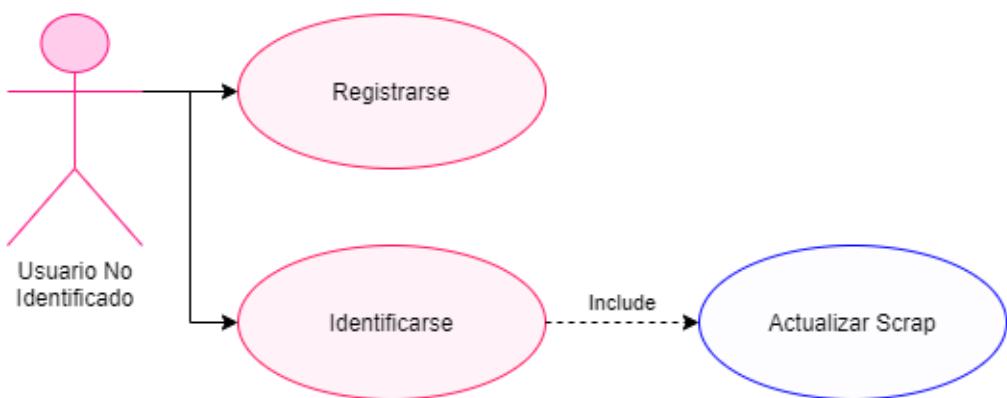


Ilustración 46 Casos de Uso del Usuario No Identificado

Registrarse
Descripción: Permite a un usuario no identificado insertar la información personal necesaria para obtener sus credenciales de acceso.
Precondiciones: El usuario no está identificado ni pendiente de aceptación
Flujo de eventos:
<ul style="list-style-type: none"> - El usuario no identificado accede al menú de registro mediante la pestaña asignada a tal efecto. - El usuario no identificado rellena todos los campos obligatorios del formulario cumpliendo las siguientes restricciones: <ul style="list-style-type: none"> o El DNI debe cumplir las restricciones establecidas entre el número y la letra. o Las contraseñas deben coincidir. o Se debe marcar la casilla de aceptación de los términos de uso. - El usuario no identificado confirma el registro mediante el botón asignado a tal efecto. <ul style="list-style-type: none"> o Si se cumplen todas las condiciones del formulario de registro, los datos son enviados al servidor donde se almacenan para su posterior aceptación. o Si no se cumple alguna de las condiciones del formulario de registro muestra un mensaje de error con información útil para resolver la incidencia.
Postcondiciones: El usuario se añadirá a la lista de usuarios pendientes de aceptación

Identificarse
Descripción: Permite a un usuario no identificado convertirse en un usuario identificado obteniendo acceso a sus funcionalidades.
Precondiciones: El usuario existe y la contraseña es válida.
Flujo de eventos:
<ul style="list-style-type: none"> - El usuario no identificado accede al menú de identificación mediante la pestaña asignada a tal efecto. - El usuario no identificado rellena todos los campos obligatorios del formulario. - El usuario no identificado confirma el acceso mediante el botón asignado a tal efecto. - Se valida si el usuario existe y si la contraseña es válida: <ul style="list-style-type: none"> o En caso afirmativo se cambia el roll del usuario no identificado por el de usuario identificado. o En caso negativo se muestra un mensaje de error con información útil para resolver la incidencia. - Se comprueban los permisos del usuario identificado y se le asignan privilegios en consecuencia. - Se ejecuta el caso de uso de Actualizar Scrap.
Postcondiciones: Se cambia el roll del usuario no identificado por el de usuario identificado.

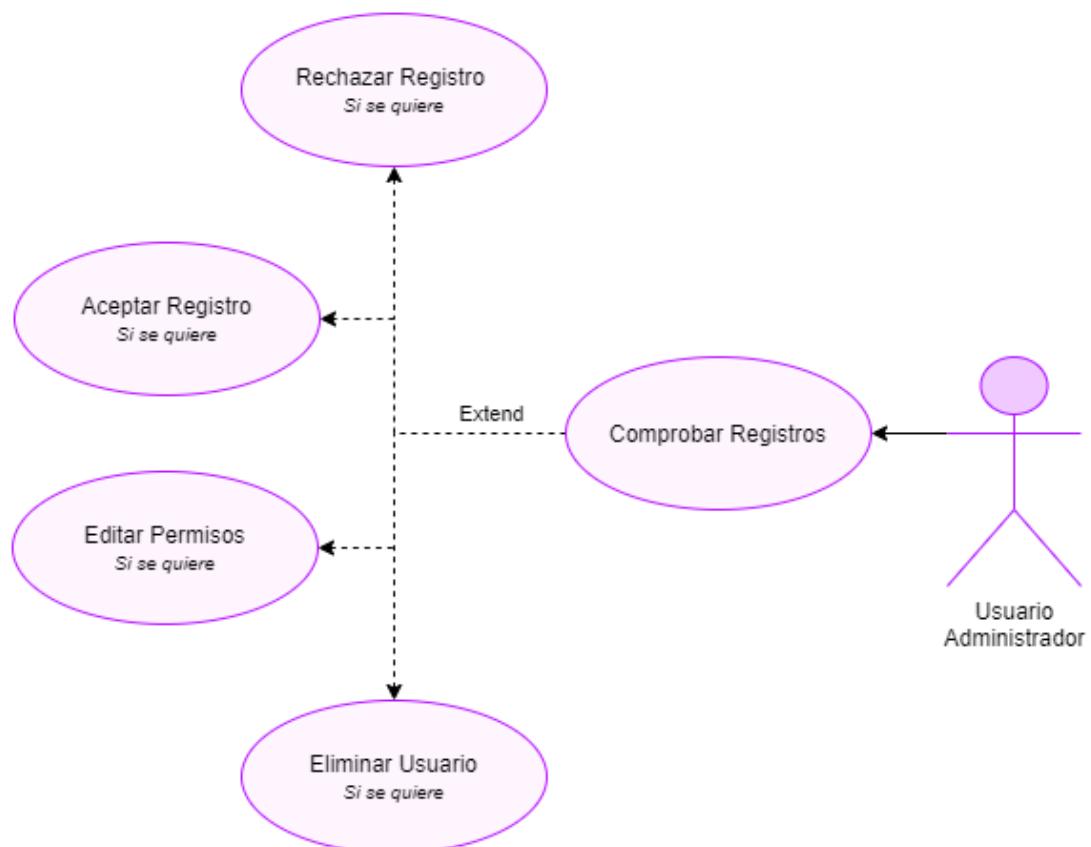


Ilustración 47 Casos de Uso del Usuario Administrador

Comprobar Registros

Descripción: Permite a un usuario administrador acceder a una ventana donde se pueden visualizar los usuarios identificados y pendientes de admisión.

Precondiciones: Ninguna

Flujo de eventos:

- El usuario administrador accede al menú de admisiones mediante la pestaña asignada a tal efecto.
- Se comprueban los permisos del usuario.
- Se cargan los datos de la tabla de admisiones.
- Se cargan los datos de la tabla de usuarios.

Postcondiciones:

Aceptar Registro

Descripción: Permite a un usuario administrador aceptar la solicitud de registro de un usuario no identificado que haya completado correctamente dicho proceso.

Precondiciones: El registro existe.

Flujo de eventos:

- El usuario administrador accede al menú de admisiones mediante la pestaña asignada a tal efecto.
- El usuario administrador selecciona los privilegios que se le asignaran al nuevo usuario.
- El usuario administrador pulsa sobre el botón de confirmar que se encuentra sobre cada solicitud.
- Se crea un nuevo usuario con la información del registro y los privilegios asignados.
- Se elimina el registro de la lista de admisiones.
- Se actualiza la ventana de admisiones.

Postcondiciones: El registro deja de existir y se crea un nuevo usuario.

Rechazar Registro

Descripción: Permite a un usuario administrador rechazar la solicitud de registro de un usuario no identificado que haya completado correctamente dicho proceso.

Precondiciones: El registro existe

Flujo de eventos:

- El usuario administrador accede al menú de admisiones mediante la pestaña asignada a tal efecto.
- El usuario administrador pulsa sobre el botón de rechazar que se encuentra sobre cada solicitud.
- Se elimina el registro de la lista de admisiones.
- Se actualiza la ventana de admisiones.

Postcondiciones: El registro deja de existir

Eliminar Usuario
Descripción: Permite a un usuario administrador eliminar a cualquier usuario.
Precondiciones: El usuario existe.
Flujo de eventos:
<ul style="list-style-type: none"> - El usuario administrador accede al menú de admisiones mediante la pestaña asignada a tal efecto. - El usuario administrador pulsa sobre el botón de eliminar que se encuentra sobre cada usuario. - Se elimina el usuario y todos los datos que tuviera asociados. - Se actualiza la ventana de admisiones.
Postcondiciones: El Usuario deja de existir.

Editar Permisos
Descripción: Permite a un usuario administrador modificar los permisos de cualquier usuario.
Precondiciones: El usuario existe.
Flujo de eventos:
<ul style="list-style-type: none"> - El usuario administrador accede al menú de admisiones mediante la pestaña asignada a tal efecto. - El usuario administrador selecciona los nuevos privilegios que se le asignaran al usuario. - Se modifican los privilegios del usuario. - Se actualiza la ventana de admisiones.
Postcondiciones: Los permisos del usuario han cambiado.

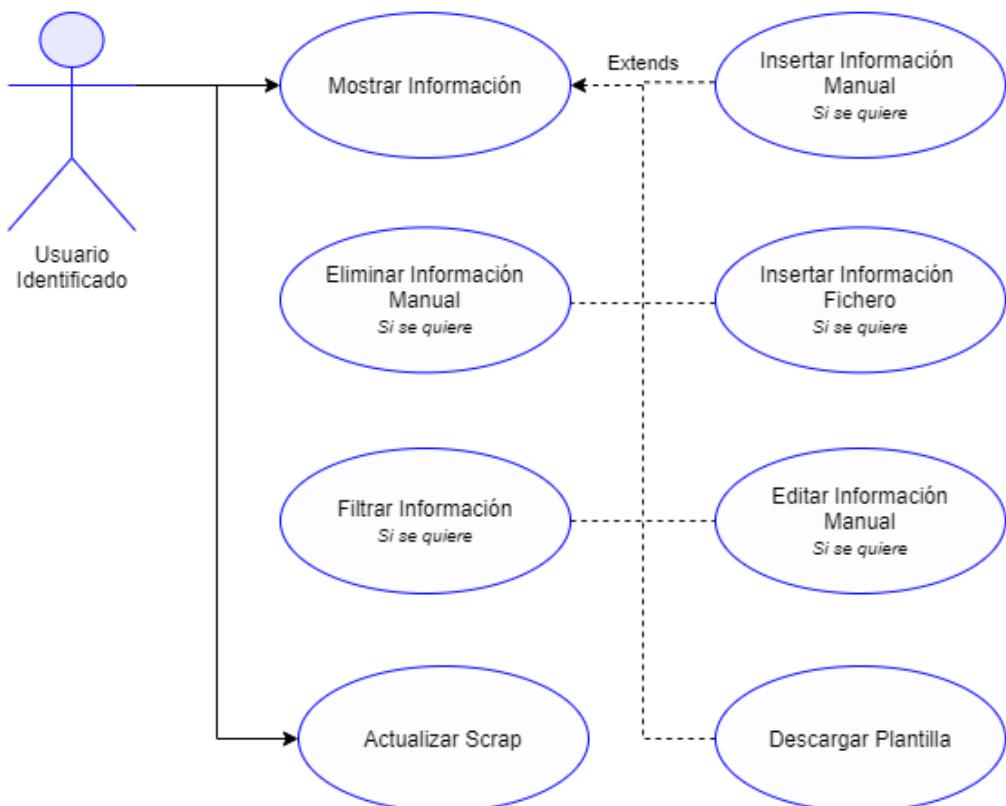


Ilustración 48 Casos de Uso del Usuario Identificado – Añadir, Eliminar, Editar y filtrar información

Mostrar Información

Descripción: Permite al usuario identificado visualizar los cambios producidos por las funcionalidades de edición manual en sus datos económicos. También da acceso a dichas funcionalidades.

Precondiciones: Ninguna.

Flujo de eventos:

- El usuario identificado selecciona alguna de las pestañas y subpestañas asignada a tal efecto:
 - o Subpestaña Gestionar de la Pestaña finanzas para acceder a los datos de finanzas:
 - Subpestaña Gastos/Ingresos para comprobar las imputaciones de gastos e ingresos personales.
 - Subpestaña Categorías para comprobar las categorías registradas.
 - o Subpestaña Gestionar de la Pestaña inversiones para acceder a los datos de activos:
 - Subpestaña Activos para comprobar los productos registrados.
 - Subpestaña Aportaciones para comprobar las transacciones.
 - Subpestaña Comisiones para comprobar las imputaciones de gastos e ingresos provenientes de los activos.
 - Subpestaña Categorías para comprobar las categorías.
 - Subpestaña Distribuciones para comprobar las distribuciones.
- Se obtienen los datos solicitados.
- Se muestran los resultados al usuario identificado.

Postcondiciones: Ninguna

Insertar Información Manual

Descripción: Permite al usuario identificado introducir sus datos económicos mediante formularios.

Precondiciones: Ninguna.

Flujo de eventos:

- El usuario identificado selecciona alguna de las pestañas descritas en el caso de uso mostrar información.
- El usuario identificado completa los campos obligatorios que se encuentran en la primera fila.
- El usuario identificado pulsa el botón añadir.
- Se comprueba que la clave de la información introducida no existiera previamente.
 - o Si no existe se añade la información introducida.
 - o Si existe se consulta al usuario si quiere modificar los datos asociados a dicha clave.
 - Si el usuario accede se modifican los datos de los campos que se han completado.
 - Si el usuario no accede no sucede nada.
- Se actualiza la ventana de Mostrar Información que se estaba utilizando.

Postcondiciones: Se han incorporado los nuevos datos.

Insertar Información desde fichero

Descripción: Permite al usuario identificado introducir sus datos económicos mediante la importación de un fichero externo diseñado para tal propósito.

Precondiciones:

- El fichero de datos debe tener el formato correcto.
- La distribución de la información debe ser la requerida.

Flujo de eventos:

- El usuario identificado selecciona alguna de las pestañas descritas en el caso de uso mostrar información.
- El usuario identificado pulsa el botón importar datos.
- El usuario identificado selecciona el fichero que quiere importar
- Se comprueba si el fichero tiene el formato correcto.
 - Si no tiene el formato correcto:
 - Se notifica al usuario identificado.
 - Se termina el proceso de insertar información desde fichero.
 - Se vuelve a la ventana de Mostrar Información que se estaba utilizando.
 - Si tiene el formato correcto continua el proceso de insertar información desde fichero.
- Se lee cada línea del fichero comprobando que la clave de la información no existiera previamente:
 - Si no existe se añade la información introducida.
 - Si existe se consulta al usuario si quiere modificar los datos asociados a dicha clave:
 - Si el usuario accede se modifican los datos de los campos que se han completado.
 - Si el usuario no accede no sucede nada.
- Se actualiza la ventana de Mostrar Información que se estaba utilizando.

Postcondiciones: Se han incorporado los nuevos datos

Descargar Plantilla

Descripción: Permite a un usuario identificado obtener una plantilla con el formato adecuado donde se indica la distribución que deben tener los datos para que sea aceptado por caso de uso de Insertar información desde fichero.

Precondiciones: Ninguna.

Flujo de eventos:

- El usuario identificado selecciona alguna de las pestañas descritas en el caso de uso mostrar información.
- El usuario identificado pulsa el botón descargar plantilla.
- El usuario identificado selecciona la ubicación de la descarga.
- Se proporciona al usuario identificado la plantilla correspondiente a la información que se inserta en la ventana de Mostrar Información que se estaba utilizando.

Postcondiciones: El usuario identificado obtiene la plantilla.

Editar Información Manual

Descripción: Permite al usuario identificado modificar los datos económicos que ya había introducido previamente mediante formularios.

Precondiciones: La clave de la información no se puede modificar.

Flujo de eventos:

- El usuario identificado selecciona alguna de las pestañas descritas en el caso de uso mostrar información.
- El usuario identificado modifica los campos que quiere modificar de la fila que quiera modificar.
- El usuario identificado pulsa el botón editar de la fila que quiera modificar.
- Se modifican los datos de los campos que se han completado.
- Se actualiza la ventana de Mostrar Información que se estaba utilizando.

Postcondiciones: La información ha sido modificada

Eliminar Información Manual

Descripción: Permite al usuario identificado eliminar los datos económicos que ya había introducido previamente mediante formularios.

Precondiciones: No existen dependencias

Flujo de eventos:

- El usuario identificado selecciona alguna de las pestañas descritas en el caso de uso mostrar información.
- El usuario identificado modifica los campos deseados de la fila que quiera seleccionada.
- El usuario identificado pulsa el botón editar de la fila que quiera modificar.
- Se eliminan los datos de los campos que se han completado.
- Se actualiza la ventana de Mostrar Información que se estaba utilizando.

Postcondiciones: La información ha sido eliminada:

- Si existen dependencias no se eliminará nada.

Filtrar Información

Descripción: Permite al usuario identificado seleccionar únicamente algunos de los datos económicos que ya había introducido previamente en función de ciertos criterios preestablecidos.

Precondiciones: Ninguna.

Flujo de eventos:

- El usuario identificado selecciona alguna de las pestañas descritas en el caso de uso mostrar información.
- El usuario identificado selecciona los filtros deseados de la cabecera de la tabla.
- El usuario identificado pulsa el botón filtrar de la cabecera de la tabla.
- Se actualiza la ventana de Mostrar Información que se estaba utilizando, mostrando únicamente las filas que cumplen las condiciones del filtro.

Postcondiciones: Si ninguna fila cumple el filtro se mostrará la tabla vacía.

Actualizar Scrap
Descripción: Permite al usuario identificado cargar desde la Web la información variable de los activos.
Precondiciones: <ul style="list-style-type: none"> - Hay conexión a internet. - Las referencias a la ubicación de la información son correctas.
Flujo de eventos: <ul style="list-style-type: none"> - El usuario identificado pulsa el botón actualizar. - Para cada activo al que este suscrito el usuario se realizará una consulta mediante la técnica de Web Scraping <ul style="list-style-type: none"> o Si no se puede ejecutar la consulta: <ul style="list-style-type: none"> ▪ Se mantendrán los valores anteriores. ▪ Se notificará al usuario identificado del error producido. o Si la consulta se ejecuta correctamente: <ul style="list-style-type: none"> ▪ Se actualizará la información variable de los activos.
Postcondiciones: Ninguna

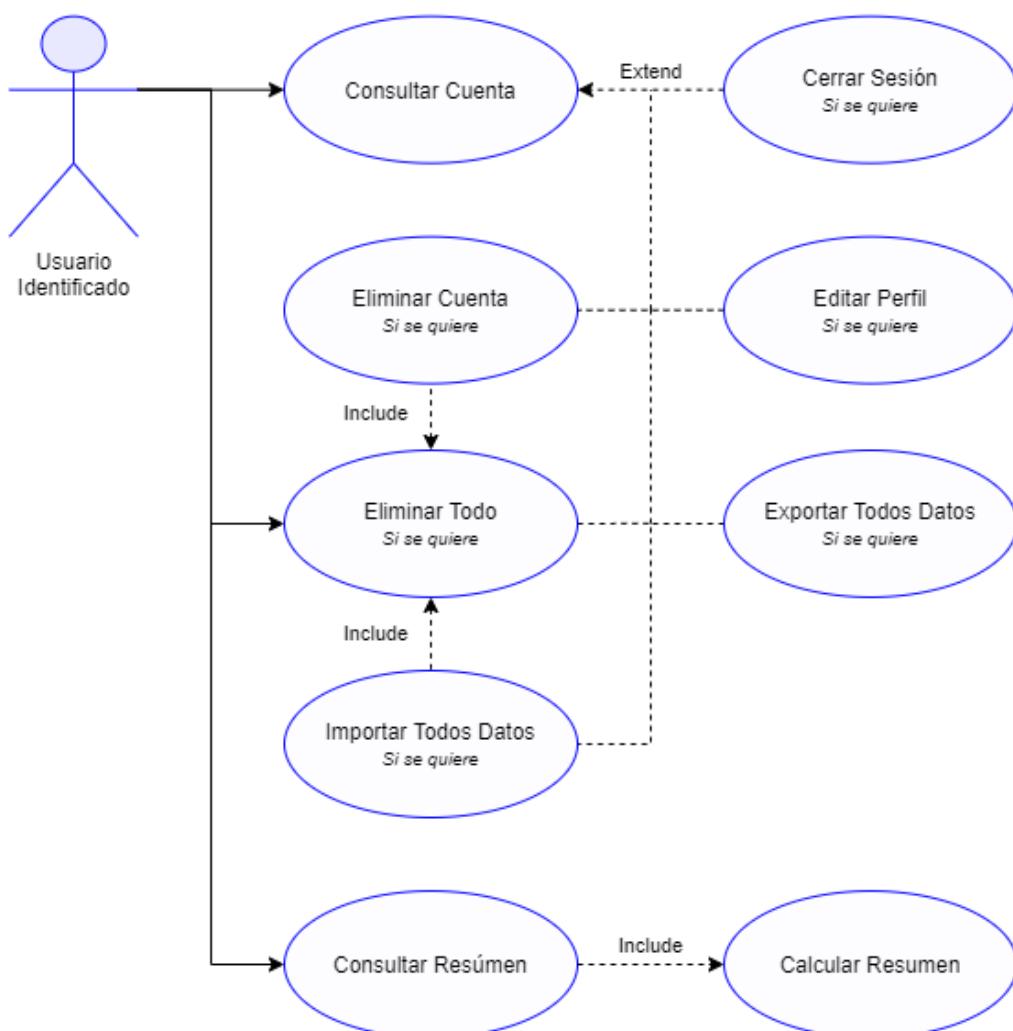


Ilustración 49 Casos de Uso del Usuario Identificado – Administrar sesión y realizar consultas

Consultar Cuenta
Descripción: Permite a un usuario identificado consultar los datos personales que introdujo durante el registro.
Precondiciones: Ninguna.
Flujo de eventos:
<ul style="list-style-type: none"> - El usuario identificado accede al menú de Mi Cuenta mediante la pestaña asignada a tal efecto. - Se mostrarán todos los datos personales del usuario.
Postcondiciones: Ninguna.

Editar Perfil
Descripción: Permite a un usuario identificado modificar los datos personales que introdujo durante el registro.
Precondiciones: Ninguna.
Flujo de eventos:
<ul style="list-style-type: none"> - El usuario identificado accede al menú de Mi Cuenta mediante la pestaña asignada a tal efecto. - El usuario identificado completa aquellos campos del formulario que desee modificar. - El usuario identificado pulsa sobre el botón Editar Perfil. - Se modifican los datos de los campos que se han completado. - Se actualiza la ventana de Mi Cuenta.
Postcondiciones: El usuario ha modificado sus datos personales.

Cerrar Sesión
Descripción: Permite a un usuario identificado bloquear su cuenta con la finalidad de volver a ser un usuario no identificado.
Precondiciones: Ninguna.
Flujo de eventos:
<ul style="list-style-type: none"> - El usuario identificado accede al menú de Mi Cuenta mediante la pestaña asignada a tal efecto. - El usuario identificado pulsa sobre el botón de Cerrar Sesión. - Se bloquea el acceso a los datos del usuario. - Se elimina el estado de identificado del usuario. - Se elimina toda la información temporal almacenada sobre el usuario. (Cookies) - Se regresa al menú principal.
Postcondiciones: Ninguna

Eliminar Cuenta
Descripción: Permite a un usuario identificado eliminar su cuenta de forma permanente.
Precondiciones: Ninguna
Flujo de eventos: <ul style="list-style-type: none"> - El usuario identificado accede al menú de Mi Cuenta mediante la pestaña asignada a tal efecto. - El usuario identificado pulsa sobre el botón de eliminar cuenta. - Se eliminan todos los datos que el usuario tuviera asociados. (Subcaso de uso eliminar todo) - Se elimina el registro del usuario. - Se elimina el estado de identificado del usuario. - Se elimina toda la información temporal almacenada sobre el usuario. (Cookies) - Se regresa al menú principal.
Postcondiciones: El usuario ya no existe.

Importar Todos Datos
Descripción: Permite a un usuario identificado importar todos sus datos económicos mediante la importación de un fichero externo diseñado para tal propósito.
Precondiciones: <ul style="list-style-type: none"> - El fichero de datos debe tener el formato correcto. - La distribución de la información debe ser la requerida.
Flujo de eventos: <ul style="list-style-type: none"> - El usuario identificado accede al menú de Mi Cuenta mediante la pestaña asignada a tal efecto. - El usuario identificado pulsa el botón importar datos. - El usuario identificado selecciona el fichero deseado. - Se eliminan todos los datos que el usuario tuviera asociados. (Subcaso de uso eliminar todo) - Se añade la información del fichero.
Postcondiciones: Se han incorporado los nuevos datos..

Exportar Todos Datos
Descripción: Permite a un usuario identificado exportar todos sus datos económicos a un fichero externo con el formato y distribución adecuados para que sea aceptado por el caso de uso de Importar todos los datos.
Precondiciones: Ninguna.
Flujo de eventos: <ul style="list-style-type: none"> - El usuario identificado accede al menú de Mi Cuenta mediante la pestaña asignada a tal efecto. - El usuario identificado pulsa el botón exportar datos. - El usuario identificado selecciona la ubicación de la descarga. - Se proporciona al usuario identificado el fichero correspondiente.
Postcondiciones: El usuario identificado obtiene el fichero correspondiente.

Eliminar Todo
Descripción: Permite a un usuario identificado eliminar todos los datos que tuviera asociados.
Precondiciones: Ninguna.
Flujo de eventos:
<ul style="list-style-type: none"> - El usuario identificado accede al menú de Mi Cuenta mediante la pestaña asignada a tal efecto. - El usuario identificado pulsa el botón eliminar datos. - Se eliminan todos los datos que el usuario identificado tuviera asociados.
Postcondiciones: El usuario identificado no tiene ningún dato.
Consultar Resumen
Descripción: Permite a un usuario identificado consultar los resúmenes obtenidos por la aplicación.
Precondiciones: Ninguna.
Flujo de eventos:
<ul style="list-style-type: none"> - El usuario identificado selecciona alguna de las pestañas y subpestañas asignada a tal efecto: <ul style="list-style-type: none"> ○ Pestaña de Balance para consultar el resumen general. ○ Pestaña de finanzas para consultar los resúmenes de finanzas: <ul style="list-style-type: none"> ▪ Subpestaña Resumen para consultar los datos ordenados en tablas. ▪ Subpestaña Sectores para consultar los datos ordenados gráficas sectoriales. ▪ Subpestaña Categorías para consultar los datos ordenados gráficas de frecuencias. ○ Pestaña de inversiones para consultar los resúmenes de activos: <ul style="list-style-type: none"> ▪ Subpestaña Histórico para consultar la evolución temporal de los activos. ▪ Subpestaña Resumen para consultar la asignación de activos ordenadas por categorías, estrategias o Brókers. ▪ Subpestaña Distribuciones para consultar las distribuciones de los activos ordenadas por categorías, estrategias o Brókers. - Se obtienen los datos solicitados. - Se calculan los indicadores necesarios. (Subcaso de uso Calcular Resumen) - Se muestran los resultados al usuario identificado.
Postcondiciones: Ninguna.

Modelo de Dominio

Basándonos en los casos de uso definidos podemos especificar los datos permanentes que nuestro sistema deberá ser capaz de almacenar. Mediante el modelo de dominio podemos representar los tipos de objetos que necesita el sistema indicando las relaciones que tendrán entre sí.



Ilustración 50 Modelo de dominio – Usuarios

El sistema de almacenamiento de usuarios.

- Estará dividido en tres grupos.
 - **Admisiones:** recogerá aquellos usuarios que ya se han registrado, pero todavía están pendientes de validar por algún usuario con suficientes permisos.
 - **Usuarios:** recogerá aquellos usuarios cuyos registros ya han sido validados por algún usuario con suficientes permisos.
 - **Accesos:** recogerá la información de todos los intentos de acceso correctos y fallidos de los usuarios, así como los movimientos clave de información con la finalidad de identificar posibles accesos fraudulentos.
- Será independiente de los datos almacenados para cada usuario, los cuales existirán en un fichero independiente que le será vinculado.

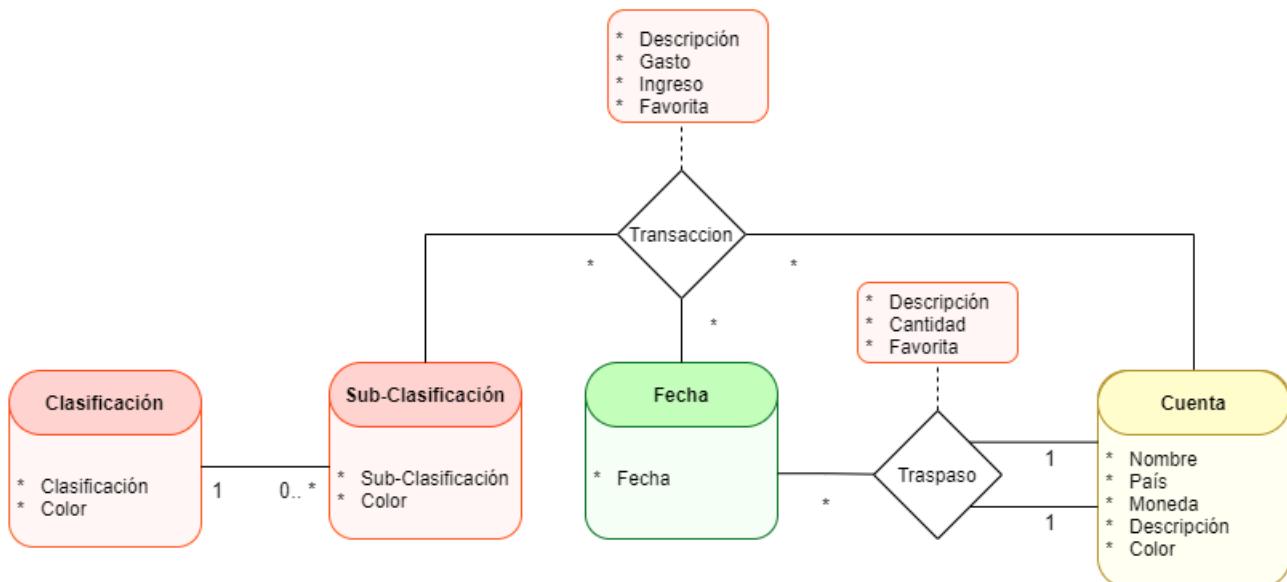


Ilustración 51 Modelo de dominio – Transacciones

El sistema de almacenamiento de finanzas consta de:

- **Clasificaciones:** Se trata de un sistema de clasificación de gastos que permitirá al usuario asignar su propio criterio de organización con la finalidad de poder analizar las implicaciones globales de cada una.
 - o Para cada fecha posible se admite una operación por cada Clase, Asunto y Cuenta.
 - o Por cada clasificación se admite una operación en cada fecha por cada Asunto y Cuenta.
 - o Por cada Asunto se admite una operación en cada fecha por cada Clase y Cuenta.
 - o Por cada Cuenta se admite una operación en cada fecha por cada Clase y Asunto.

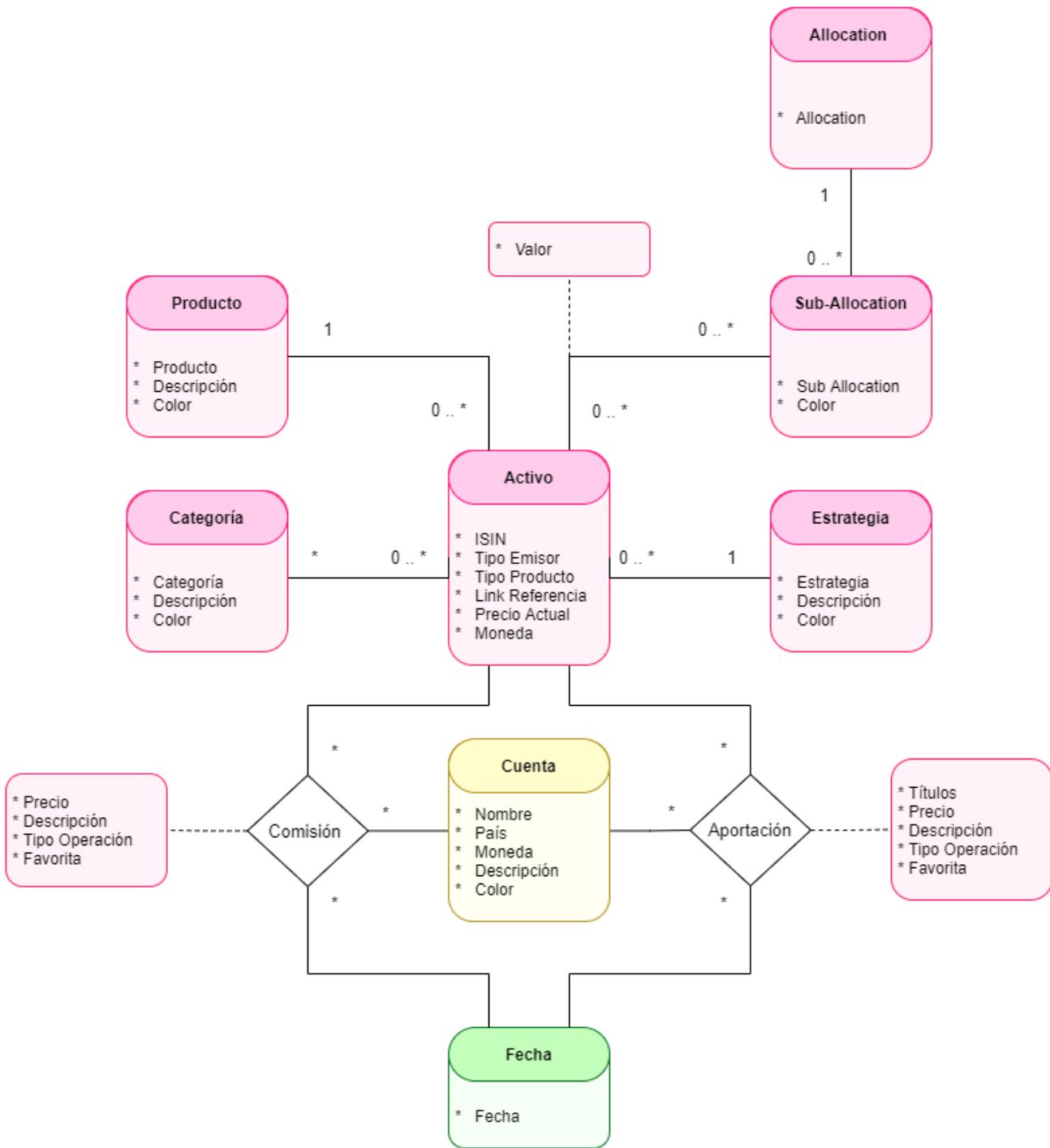


Ilustración 52 Modelo de dominio – Activos, Comisiones y Aportaciones

El sistema de almacenamiento de activos consta de:

- **Categorías:** Se trata de un sistema de clasificación de activos que permitirá al usuario asignar su propio criterio de organización, agrupando distintos activos bajo un mismo criterio. Por ejemplo, podría crearse la categoría de empresas americanas y la de empresas tecnológicas e incluir en ambas los Activos de Google y Amazon.
 - Cada activo puede pertenecer a muchas categorías.
 - Cada categoría puede representar a muchos activos diferentes.
- **Productos:** Se trata de un sistema de clasificación de activos que permitirá agrupar aquellos activos que representen a un mismo producto. Esta clasificación podría ser utilizada, por ejemplo, para agrupar bajo un mismo nombre a todos los activos sutilmente diferentes que permiten replicar el mismo índice.
 - Cada Activo se incluye en un único producto.
 - Cada Producto puede representar a muchos activos diferentes.
- **Estrategia:** Se trata de un sistema de clasificación de activos que permitirá al usuario agrupar aquellos activos que se han comprado con el mismo propósito. Existen muchísimas técnicas de inversión, y por regla general no se suele aplicar una única estrategia simultáneamente. No obstante, es conveniente ser capaz de realizar un seguimiento individual de cada estrategia.
 - Cada activo se incluye en una única estrategia.
 - Cada Estrategia puede representar a muchos activos diferentes.
- **Allocation:** Se trata de un sistema de segmentación de la información correspondiente a un activo con la finalidad de poder analizar las implicaciones globales de cada una de sus partes. Existen muchas segmentaciones posibles que es interesante evaluar a la hora de analizar una cartera. Una de las más habituales, por ejemplo, es la distribución geográfica. El valor de un activo puede estar repartido en diferentes zonas geográficas con diferentes pesos, y resulta de gran utilidad conocer cuánto pesa cada zona geográfica para un determinado conjunto de activos. Por ejemplo, ¿cuál es el peso de empresas europeas en el conjunto de activos que forman parte de una estrategia determinada?
 - Cada activo puede incluir muchos Allocation que podría ser interesante tener en cuenta.
 - Cada Allocation puede representar a muchos activos diferentes.

A su vez, un Allocation está dividido en varias categorías. Por ejemplo, el Allocation de distribución geográfica podría dividirse en Europa, Asia, América, etc...

- Cada Allocation puede incluir muchas categorías.
- Cada Categoría sólo puede pertenecer a un Allocation.
- **Comisión:** Se trata de un registro que permite anotar los costes e ingresos que están asociados a un activo o una Cuenta.
 - Para cada fecha posible se admite una comisión por cada Cuenta y Activo.
 - Por cada Cuenta se admite una comisión en cada fecha por cada Activo.
 - Por cada Activo se admite una comisión en cada fecha por cada Cuenta.
- **Aportación:** Se trata de un registro que permite anotar las subscripciones que se realizan de cada activo en cada Cuenta.
 - Para cada fecha posible se admite una aportación por cada Cuenta y Activo.
 - Por cada Cuenta se admite una aportación en cada fecha por cada Activo.
 - Por cada Activo se admite una aportación en cada fecha por cada Cuenta.

Diseño de la interfaz

Este apartado de la documentación está destinado a describir la distribución espacial de cada una de las piezas de la aplicación.

La ventana se dividirá en cinco secciones:

- **Logo:** Esta sección servirá como enlace a la guía de utilización de la aplicación, además de su función decorativa.
- **Menús de navegación:** Esta sección estará formada por un conjunto de enlaces que permitirán navegar por los diferentes apartados de la aplicación.
- **Informes de error:** Esta sección aparecerá únicamente cuando se produzca algún fallo que sea de interés para el usuario y contendrá información relevante para resolverlo.
- **Cuerpo:** Esta sección contendrá la información del apartado que se esté visualizando actualmente.
- **Pie de página:** Esta sección contendrá la referencia a otros contenidos relevantes del autor.

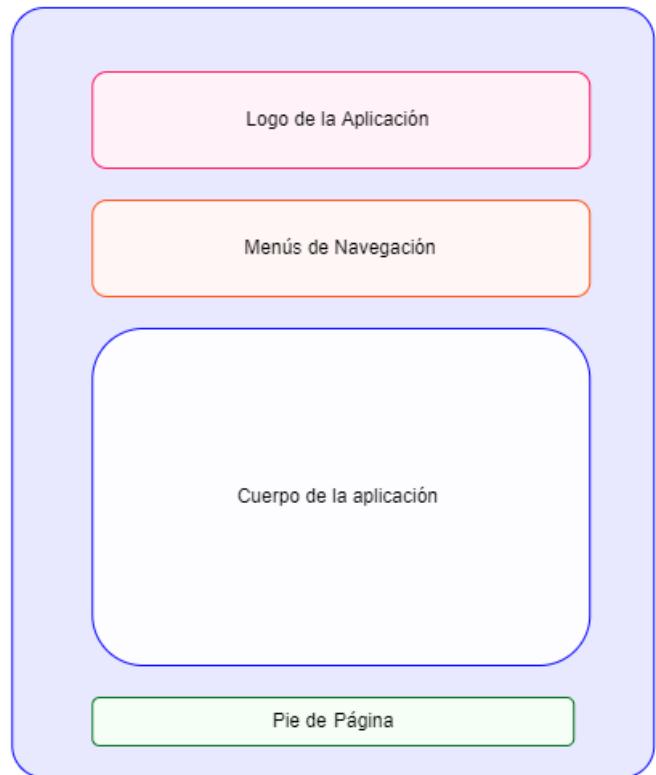


Ilustración 53 Estructura General de la Interfaz

Podemos distinguir los siguientes tipos de apartados:

- **Guía de utilización:** Se trata de un conjunto de ventanas en las que se recogerá información acerca de cómo utilizar la aplicación y cómo interpretar los resultados obtenidos.
- **Opciones de cuenta:** Se trata de un conjunto de formularios que permitirán al usuario eliminar su cuenta, modificar los datos que introdujo durante el registro o cerrar sesión.
- **Inserción de la información:** Se trata de un conjunto de formularios que permitirán al usuario introducir sus datos económicos mediante formularios o la importación de ficheros externos diseñados para tal efecto.
- **Presentación de los resultados:** Se trata de un conjunto de ventanas informativas que mostrarán al usuario los indicadores económicos pertinentes mediante un conjunto de tablas y gráficas.

Arbol de navegación

En esta sección se muestra cómo se han organizado las diferentes ventanas de la aplicación.

- **Guía de utilización**
 - **Manual:** Recoge información acerca del funcionamiento de la aplicación.
 - **Interpretación:** Recoge información acerca de cómo interpretar los resultados de las gráficas.
 - **Formación:** Recoge información acerca de los conceptos más importantes sobre finanzas personales.
- **Registro:** Permite al usuario introducir sus datos personales con la finalidad de crearse una cuenta nueva.
- **Inicio de sesión:** Permite al usuario identificarse en la aplicación introduciendo su usuario y contraseña.
- **Configuración:** Permite al usuario acceder a las opciones de configuración de la cuenta.
 - **Editar Registro:** Permite al usuario modificar los datos personales que introdujo durante el registro.
 - **Cerrar sesión:** Permite al usuario bloquear su cuenta hasta que vuelva a identificar.
 - **Eliminar Cuenta:** Permite al usuario eliminar toda la información de su cuenta de forma permanente.
- **Admisiones:** Permite al usuario acceder a las opciones de administración de usuarios.

Este menú sólo estará disponible si se dispone de los privilegios necesarios.

 - **Gestión Registros:** Permite al usuario gestionar las solicitudes de registro de nuevos usuarios.
 - **Gestión Usuarios:** Permite al usuario gestionar los permisos de los usuarios registrados.
 - **Gestión Accesos:** Permite al usuario comprobar los intentos de acceso de los usuarios.
- **Importaciones:** Permite al usuario introducir sus datos económicos mediante la importación de ficheros externos diseñados para tal efecto, así como exportar los datos existentes y descargar plantillas.
- **Gestión:** Permite al usuario añadir, eliminar y modificar los datos económicos de las siguientes categorías:
 - **Cuentas:** Permite definir cuentas.
 - **Activos:** Permite definir activos.
 - **Categorías:**
 - **Categorías:** Permite definir categorías para clasificar los activos.
 - **Estrategias:** Permite definir estrategias para clasificar los activos.
 - **Productos:** Permite definir productos para clasificar los activos.
 - **Distribuciones:**
 - **Allocation:** Permite definir los tipos de segmentación de activos.
 - **Clasificación:** Permite definir los tipos de segmentación de gastos.
 - **Operaciones:**
 - **Transacciones:** Permite definir gastos e ingresos sobre una cuenta.
 - **Traspasos:** Permite definir un movimiento de capital entre cuentas mediante.
 - **Aportaciones:** Permite definir compras o ventas de activos.
 - **Comisiones:** Permite definir gastos asociados a activos o cuentas.

- **Análisis:** Se trata de un conjunto de gráficas y tablas que mostrarán al usuario los indicadores más relevantes en cuanto a los gastos, ingresos y activos.
 - o **Balance:** Tablas y gráficas que muestran la proporción de activos y pasivos para un periodo determinado
 - o **Evolución temporal:** Conjunto de gráficas que mostrarán la evolución temporal del patrimonio
 - **Gastos e Ingresos:** Los gastos e ingresos asignados a las distintas categorías.
 - **Activos:** Los diferentes conjuntos de activos que se han definido.
 - o **Flujo de caja:** Conjunto de tablas y gráficas analizan los gastos e ingresos.
 - **Resumen:** muestra los datos de gastos e ingresos anualizados distribuidos por categorías.
 - **Categorías:** muestra los datos de gastos e ingresos para un periodo determinado.
 - o **Patrimonio:** Conjunto de tablas y gráficas analizan los activos que se tienen en propiedad.
 - **Composiciones:** muestra los indicadores calculados para los distintos conjuntos de activos.
 - **Distribuciones:** muestra los Allocation definidos para los distintos conjuntos de activos.

Cuerpo de la aplicación

La ventana de Exportaciones permite importar y exportar los datos del usuario identificado de forma masiva mediante ficheros CSV y XML.



Ilustración 54 Menú Gestionar Exportaciones e importaciones

Las ventanas de registro, Inicio de sesión y editar registro están formados por un formulario que permite al usuario introducir los datos necesarios para completar dicho trámite.

Algunas particularidades de estos menús son:

- El formulario de registro permite recordar los últimos datos introducidos durante la misma sesión ([excepto el de la contraseña](#))
- El formulario de editar registro carga los datos actuales del usuario.
- Generan un mensaje de error informativo cuando algún campo no se completa correctamente.
- Una vez establecido el nombre de usuario no se podrá modificar al editar el registro.

Ilustración 55 Menús de Registro, Inicio de sesión y Editar registro

La ventana de admisiones está formada por una tabla que en cada fila contiene:

- La información de cada uno de los usuarios.
- Dos formularios que permiten:
 - o Seleccionar los permisos del usuario y confirmar su admisión.
 - o Rechazar la admisión del usuario.
 - o Seleccionar los permisos del usuario y modificarlos.
 - o Eliminar la cuenta del usuario.

Algunas particularidades de estos menús son:

- Admite una serie de filtros para mostrar sólo las filas cuyos usuarios cumplan todas las condiciones indicadas.

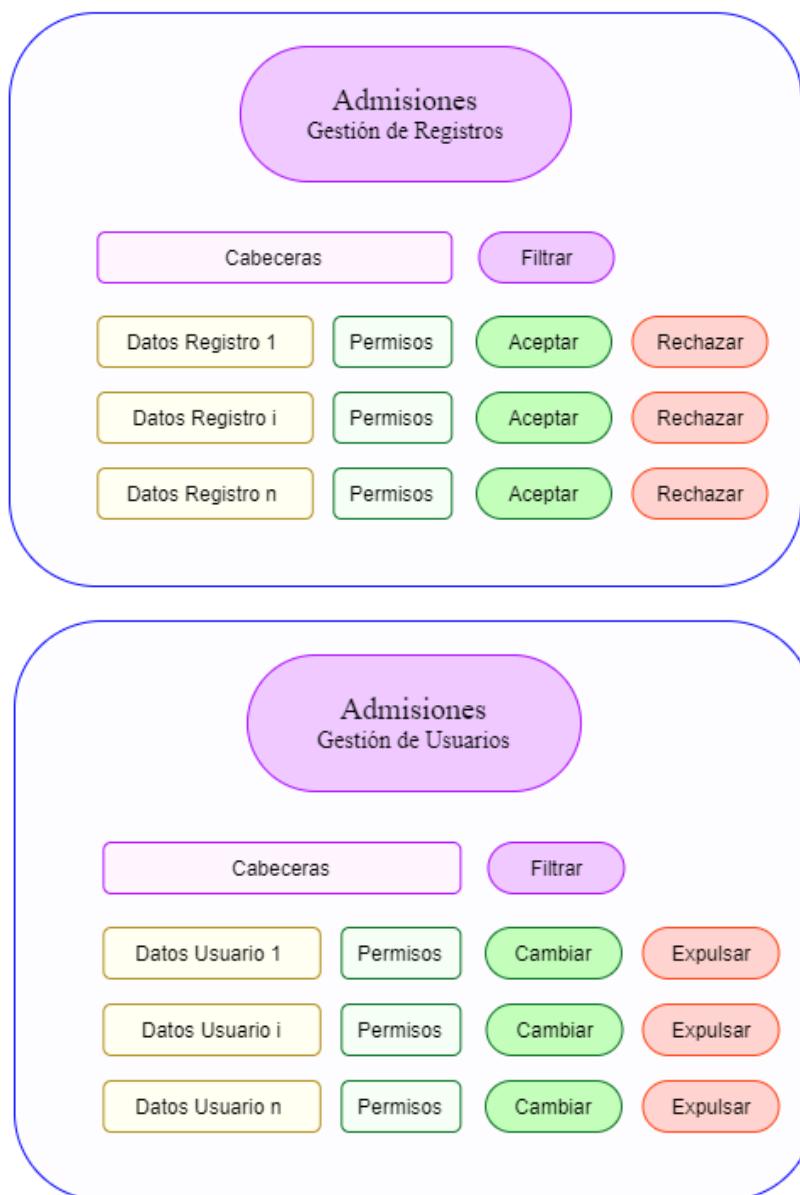


Ilustración 56 Menús de Admisiones

La ventana de gestionar está formada por una tabla que en cada fila contiene:

- La información más relevante de cada una de las entradas que se han introducido para dicho registro
- Tres botones que permiten:
 - o Redireccionar al usuario a un formulario donde puede modificar dicho registro:
 - Se cargará la información de dicho registro.
 - Se modificarán los campos que deban ser alterados.
 - Los datos de los campos modificados del registro seleccionado serán sustituidos.
 - Redirecciona nuevamente a la tabla de registros.
 - La finalidad de este proceso es no tener que introducir los campos que no deban ser alterados.
 - o Redireccionar al usuario a un formulario donde puede insertar un nuevo registro:
 - Se cargará la información de dicho registro.
 - Se modificarán los campos que difieran de los del nuevo registro.
 - Se creará un nuevo registro con los datos de todos los campos.
 - Redirecciona nuevamente a la tabla de registros.
 - La finalidad de este proceso es que el registro de referencia haga de plantilla para no tener que introducir aquellos campos que no difieran con respecto al original.
 - o Eliminar el registro seleccionado.

Algunas particularidades de estos menús son:

- Admite una serie de filtros para mostrar sólo las filas cuyos registros cumplan todas las condiciones indicadas
- Admite un filtro especial que permite mostrar únicamente los registros marcados como favoritos.

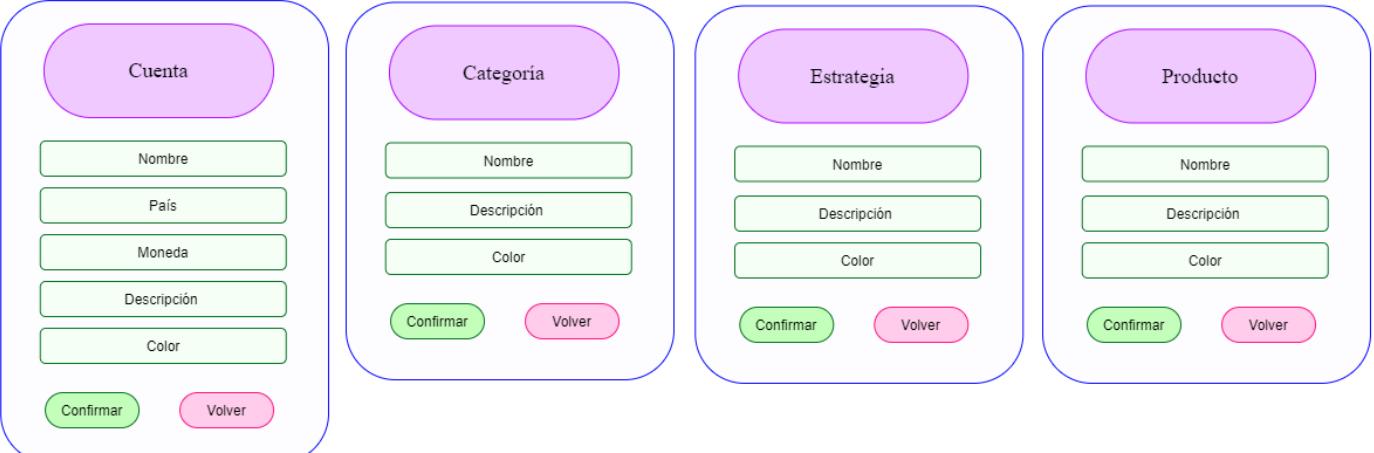


Ilustración 57 Menú Gestionar

El menú de Transacciones permite acceder a las ventanas que permiten añadir y editar los datos económicos del usuario de forma individual.

La ventana de cuentas cuenta con un formulario que permite introducir los datos básicos, como el nombre, el país, la moneda en que se representan las transacciones, la descripción y el color.

Las ventanas de Categorías, Estrategias y Productos cuentan con un formulario que permite introducir los datos básicos, como el nombre, la descripción y el color.



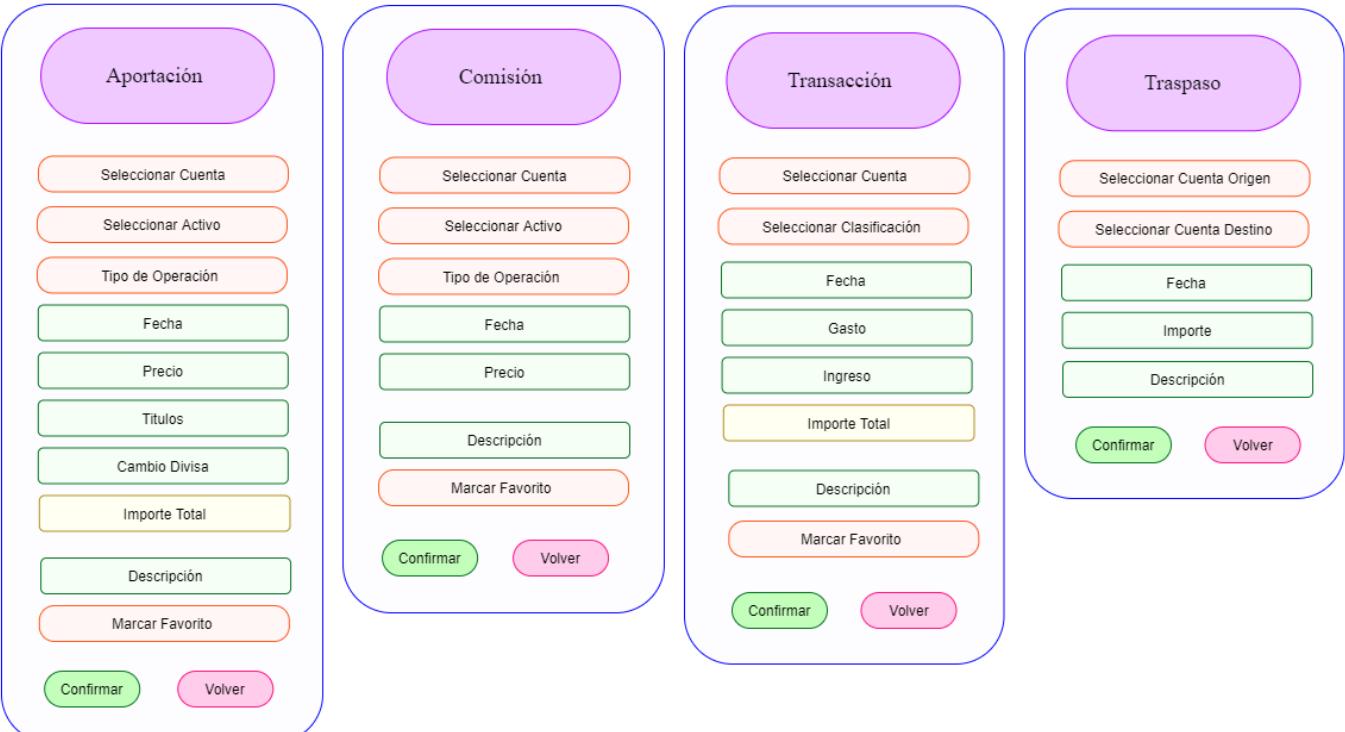
Cuenta	Categoría	Estrategia	Producto
Nombre	Nombre	Nombre	Nombre
País	Descripción	Descripción	Descripción
Moneda	Color	Color	Color
Descripción			
Color			
Confirmar	Confirmar	Confirmar	Confirmar
Volver	Volver	Volver	Volver

Ilustración 58 Formularios para añadir y editar Cuentas Categorías Estrategias o Productos

Las ventanas de Aportaciones, Comisiones, Transacciones y Traspasos cuentan con formularios que permite definir los datos básicos para definir una nueva operación entre las cuentas del usuario identificado.

Cabe destacar que no será posible:

- Definir una Aportación, Comisión, transacción o Traspaso para Cuentas o Activos que no hayan sido previamente definidos.
- Eliminar una Cuenta o Activo que tenga instancias de Aportaciones, Comisiones, Transacciones o Traspasos.



Aportación	Comisión	Transacción	Traspaso
Seleccionar Cuenta	Seleccionar Cuenta	Seleccionar Cuenta	Seleccionar Cuenta Origen
Seleccionar Activo	Seleccionar Activo	Seleccionar Clasificación	Seleccionar Cuenta Destino
Tipo de Operación	Tipo de Operación	Fecha	Fecha
Fecha	Fecha	Gasto	Importe
Precio	Precio	Ingreso	Descripción
Titulos	Descripción	Importe Total	
Cambio Divisa	Marcar Favorito	Descripción	
Importe Total	Confirmar	Marcar Favorito	
Descripción	Volver	Confirmar	
Marcar Favorito		Volver	
Confirmar	Volver	Confirmar	Volver
Volver			

Ilustración 59 Formularios para añadir y editar Aportaciones, Comisiones, Transacciones o Traspasos

Las ventanas de Allocation y Clasificación cuentan con un formulario para los datos básicos, como el nombre, la descripción y el color y una tabla con un número de filas variable que permite definir las sub-Clasificaciones o Sub-Allocation que conforman dicho Allocation o Clasificación.

- El botón de añadir de la cabecera permite insertar una nueva fila.
- El botón de eliminar quita la fila seleccionada.

La ventana de activos cuenta con un formulario para los datos básicos del activo, como el ISIN o el tipo de producto, y dos tablas que permiten gestionar las categorías y los Allocation que están ligados a dicho activo.

- El botón de añadir de la cabecera permite insertar una nueva fila.
- El botón de eliminar quita la fila seleccionada.
- El botón de editar redirecciona al formulario de asignación de pesos para el Allocation seleccionado.

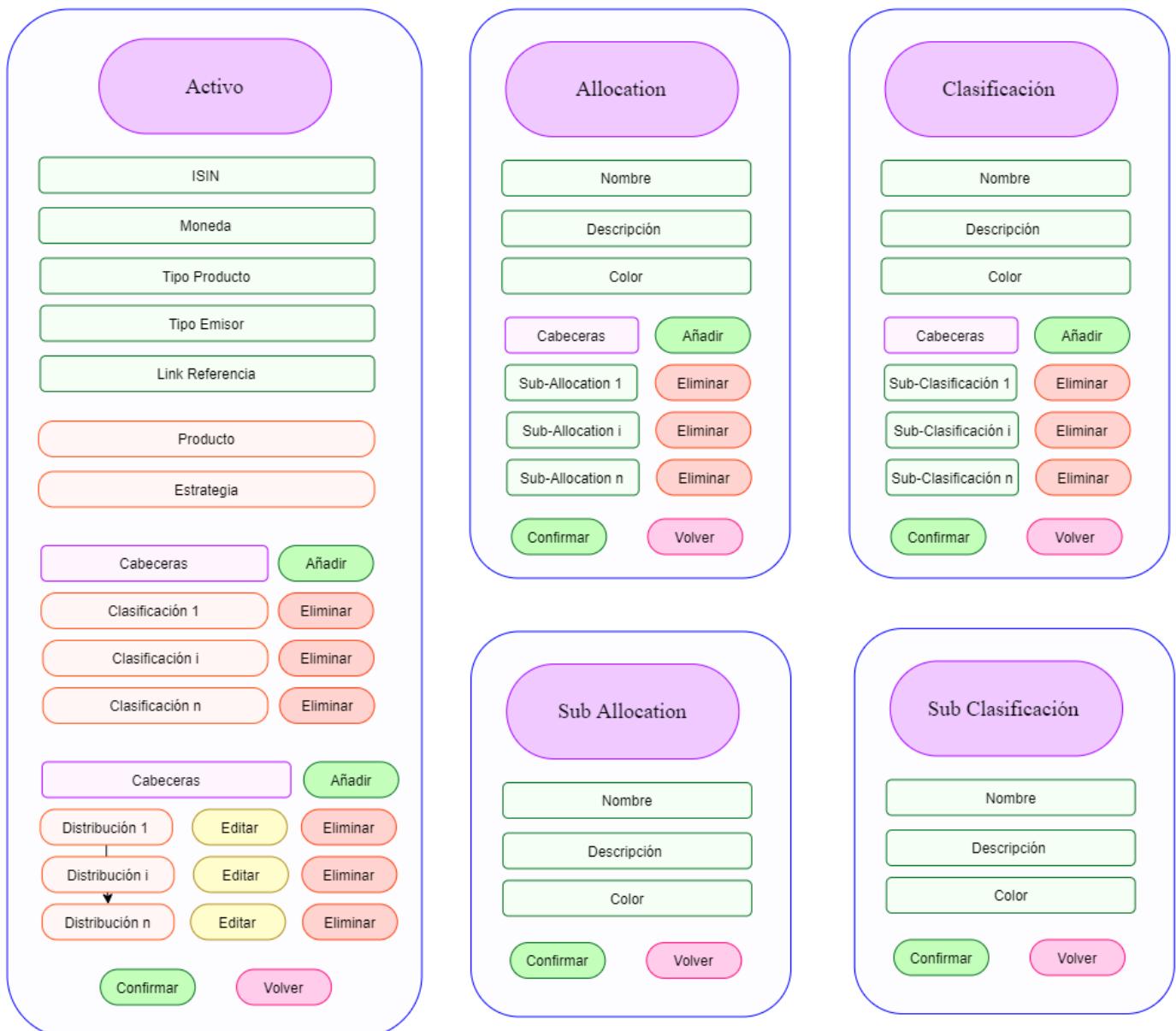


Ilustración 60 Formularios de Activos, Allocation y Clasificaciones

Desde la ventana de activos se puede acceder al formulario de gestionar distribuciones. Este formulario permite asignar valores a los Sub-Allocation de cada Allocation que pertenezca a un activo.

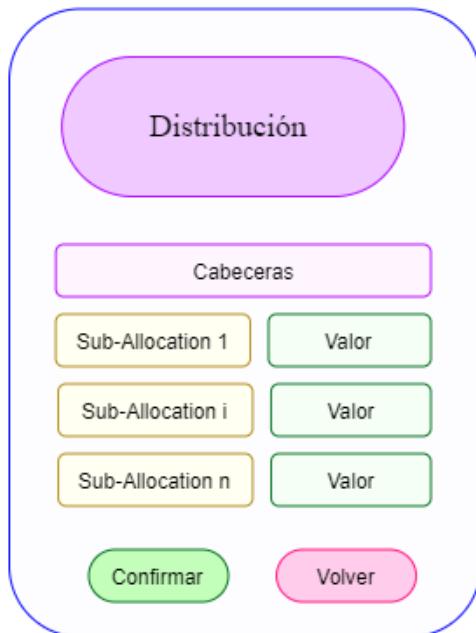


Ilustración 61 Menú Gestionar distribuciones

La ventana de Exportaciones permite importar y exportar los datos del usuario identificado de forma masiva mediante ficheros CSV y XML.



Ilustración 62 Menú Gestionar Exportaciones e importaciones

- **Análisis:** Se trata de un conjunto de gráficas y tablas que mostrarán al usuario los indicadores más relevantes en cuanto a los gastos, ingresos y activos.
 - o **Balance:** Tablas y gráficas que muestran la proporción de activos y pasivos para un periodo determinado.
 - o **Evolución temporal:** Conjunto de gráficas que mostrarán la evolución temporal del patrimonio.
 - **Gastos e Ingresos:** Los gastos e ingresos asignados a las distintas categorías
 - **Activos:** Los diferentes conjuntos de activos que se han definido.
 - o **Flujo de caja:** Conjunto de tablas y gráficas analizan los gastos e ingresos.
 - **Resumen:** muestra los datos de gastos e ingresos anualizados distribuidos por categorías
 - **Categorías:** muestra los datos de gastos e ingresos para un periodo determinado
 - o **Patrimonio:** Conjunto de tablas y gráficas analizan los activos que se tienen en propiedad.
 - **Capital:** Tabla que muestra el estado actual de todas las cuentas.
 - **Composiciones:** muestra los indicadores calculados para los distintos conjuntos de activos.
 - **Distribuciones:** muestra los Allocation definidos para los distintos conjuntos de activos.

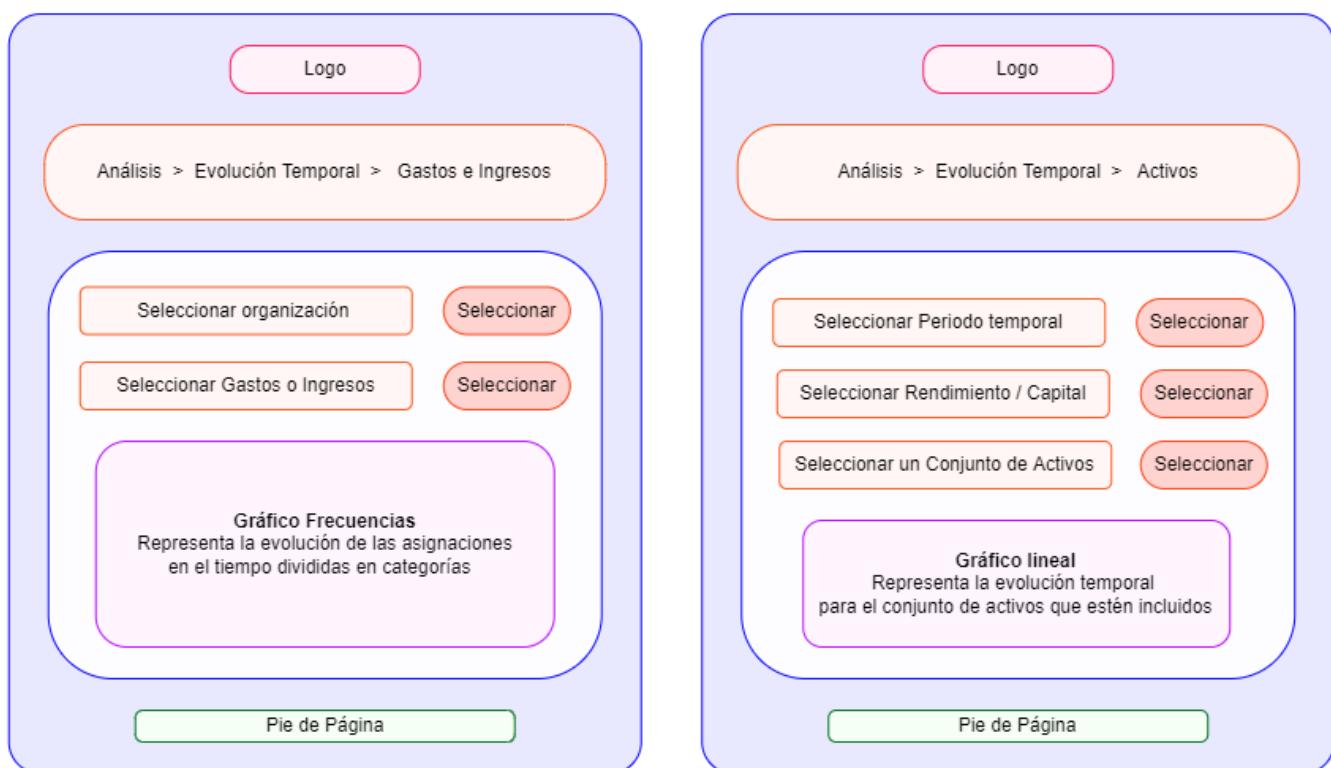


Ilustración 63 Menú de Evolución Temporal para Gastos, Ingresos y Activos

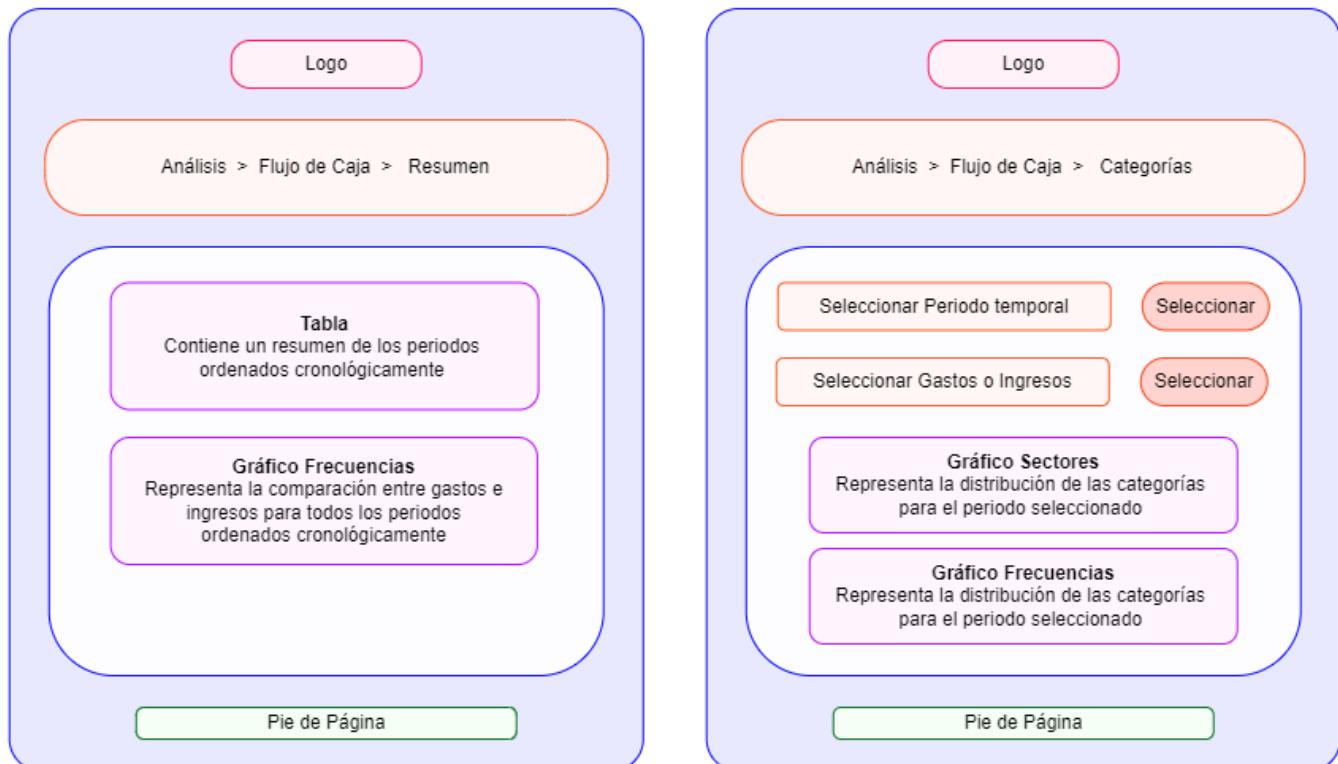


Ilustración 64 Menús de Flujo de Caja

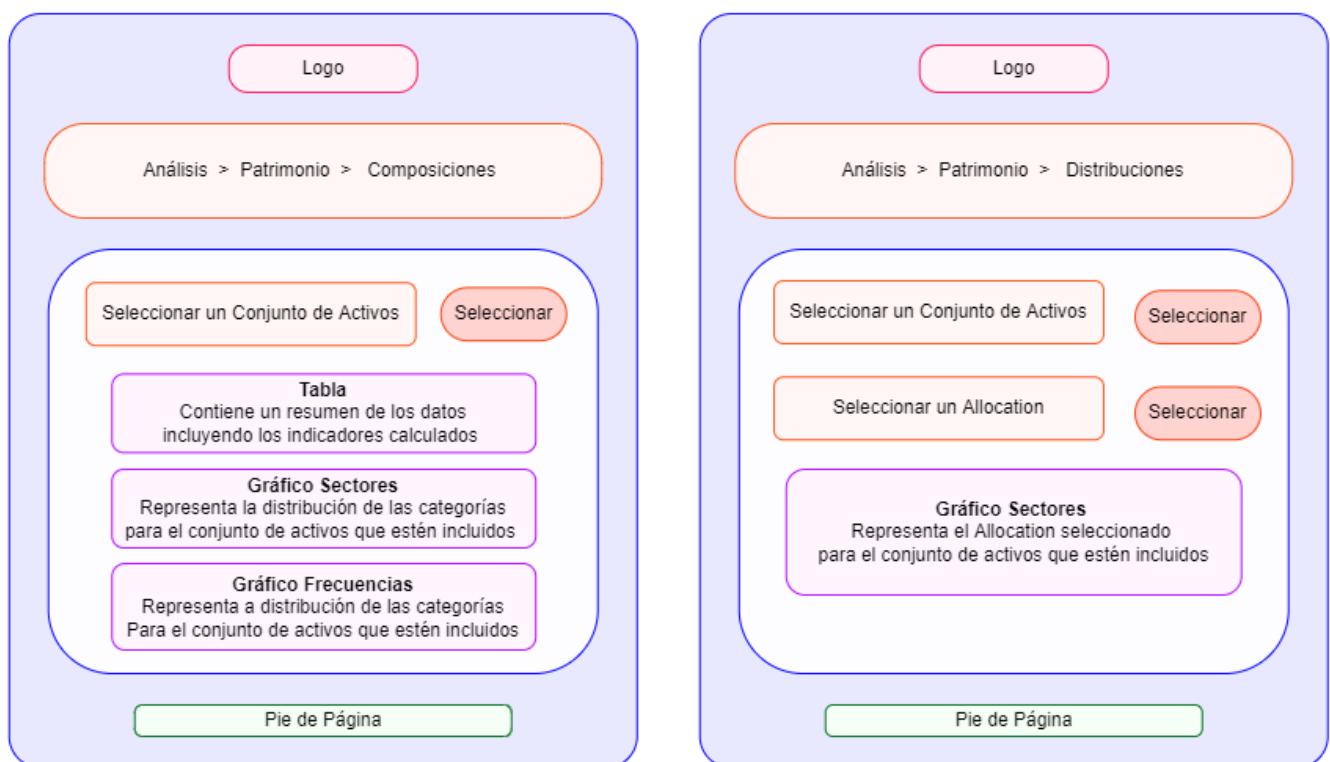


Ilustración 65 Menú de Patrimonio para Composiciones y Distribuciones

Diagrama Clases

Los diagramas de clases permiten representar de manera visual como se relacionan las distintas unidades de información que gestiona la aplicación.

- El diagrama de clases de la base de datos muestra cómo se almacena la información permanente.
- El diagrama de clases de la aplicación muestra cómo se estructura la información temporal.

Base Datos

Basándonos en el modelo de dominio podemos definir la estructura de la base de datos relacional que permitirá almacenar la información permanente de cada usuario.

En cuanto a los ficheros en los que se almacena la información.

- Los Usuarios y las Admisiones se almacenan en un fichero separado de los datos.
- A cada Usuario le corresponderá un fichero en el que se almacenaran todos sus datos de Finanzas y Activos.

En cuanto a la nomenclatura de los diagramas:

- Se resaltan en negrita las claves primarias de cada clase.
- Se utilizan flechas que apuntan de la clave primaria de una clase a las claves extranjeras que le hacen referencia.

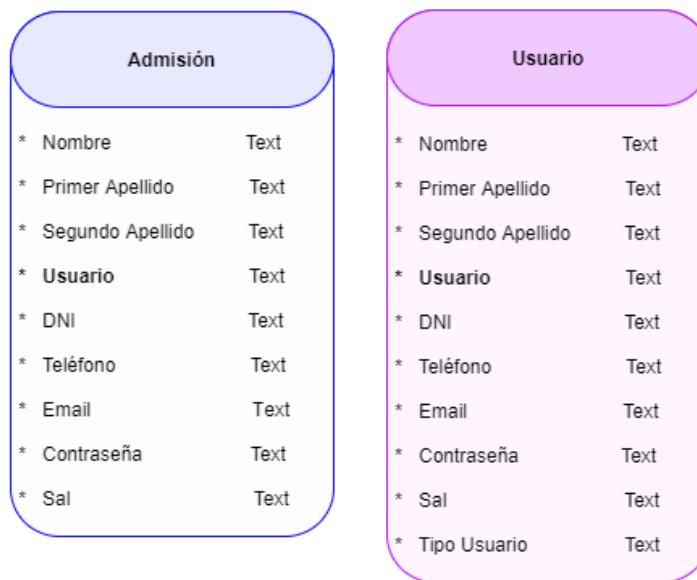


Ilustración 66 Diagrama de clases – Base Datos – Admisión y Usuarios

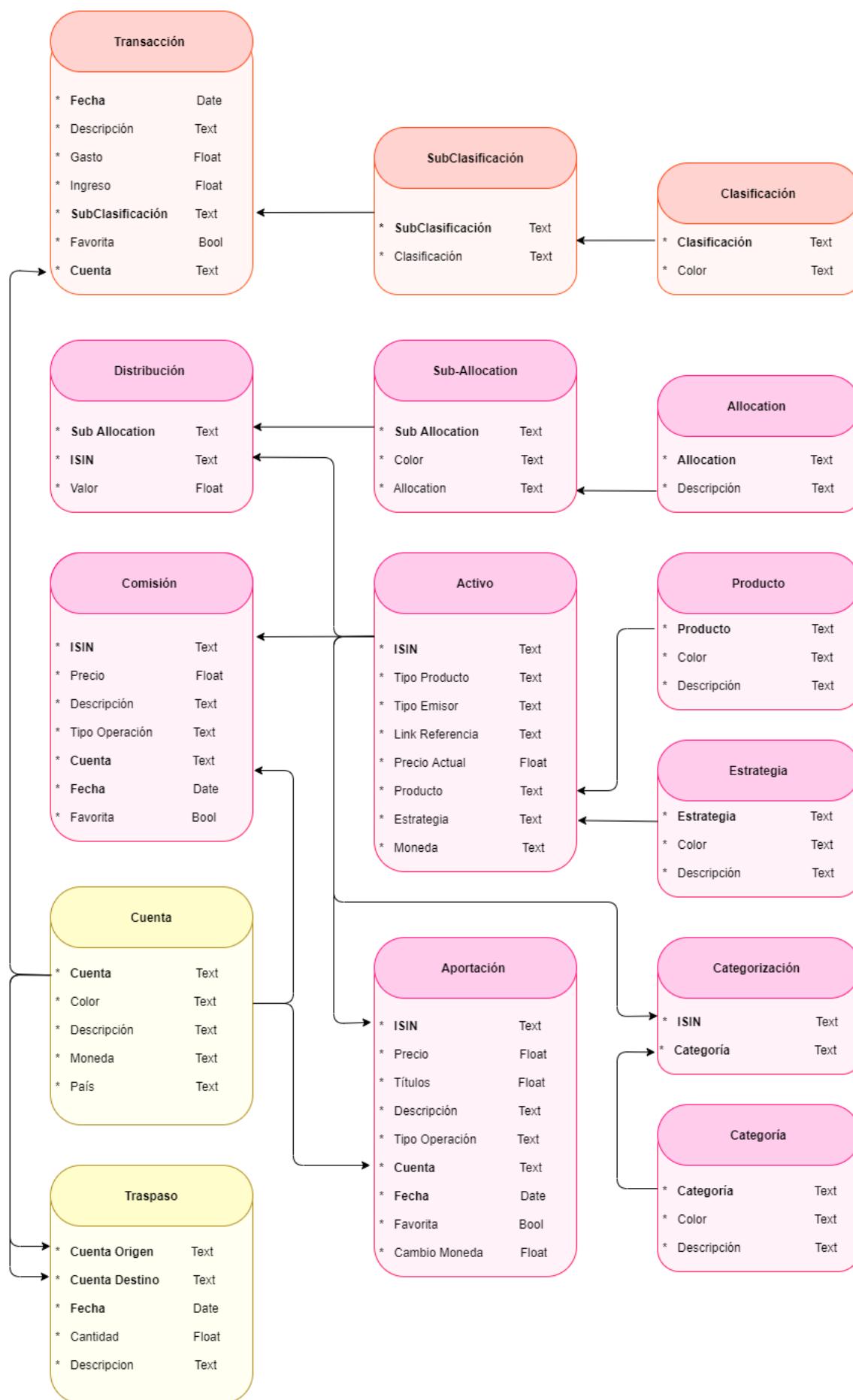


Ilustración 67 Diagrama de clases – Base Datos – Comisiones, Aportaciones y Finanzas

Aplicación

La aplicación trabajará directamente contra la base de datos con la finalidad de que no sea necesario cargar todos los datos permanentes en objetos. De este modo conseguimos:

- Minimizar la cantidad de información temporal que debe ser almacenada.
- Minimizar la carga de trabajo durante el inicio de sesión.

Se almacenará en objetos aquellos cálculos que se realicen durante la ejecución de una sesión con la finalidad de minimizar el número de operaciones evitando volver a realizar las que ya se han ejecutado.

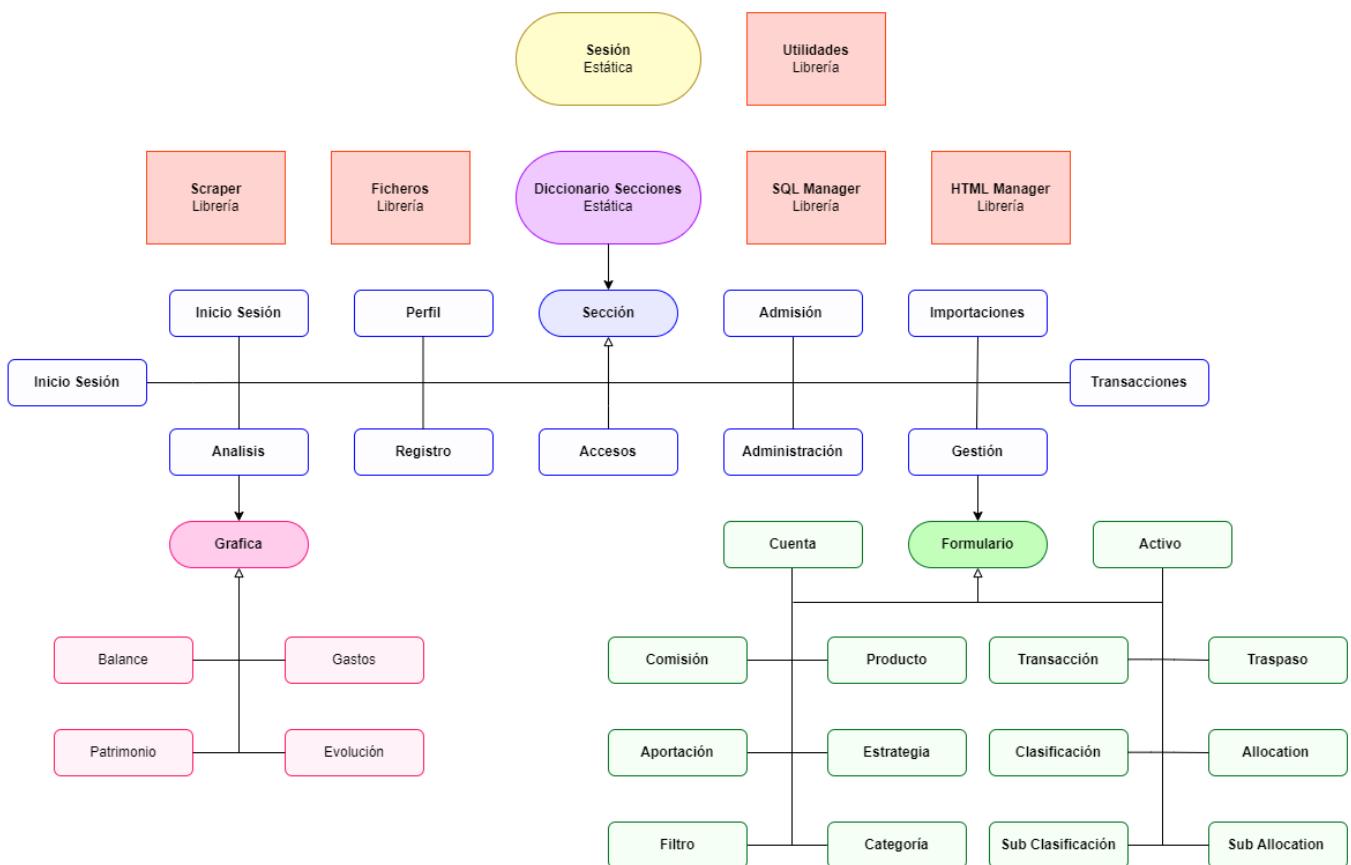


Ilustración 68 Diagrama de clases – Aplicación

A continuación, se proporciona una descripción general de la finalidad de cada clase. Todos los detalles sobre el comportamiento de cada función, así como la descripción de cada atributo se podrán encontrar en el apéndice tres, “Documentación del código”.

Sesión

La clase sesión es una clase estática ([Sólo puede existir una única instancia por cada ejecución del programa](#)) cuya finalidad es almacenar aquellos valores temporales que resultan imprescindibles para acceder a los datos permanentes del usuario que este autenticado.

Además, se encargará de gestionar:

- La autenticación de los usuarios mediante el establecimiento y cierre de sesiones
- El acceso a la información permanente de los usuarios almacenada en la Base de Datos. De este modo, las claves de la sesión nunca saldrán de esta clase.

Secciones

La clase secciones contendrá un diccionario con las diferentes secciones de la aplicación entre las que se puede navegar. Su finalidad es clasificar las funcionalidades de la aplicación conforme a una estructura que facilite su acceso y permita incorporar nuevas funcionalidades de forma modular.

Además, se encargará de gestionar la navegación a través de las diferentes secciones de la aplicación redireccinando al usuario a la sección seleccionada.

Sección

La clase sección es una plantilla genérica que define los atributos y funciones mínimas imprescindibles para que una sección cualquiera pueda ser ejecutada independientemente de su finalidad. Se utilizará la herencia para construir secciones específicas con funcionalidades relevantes.

Esta clase define funciones que deberán ser reescritas por las clases herederas para encargarse de:

- Mostrar al usuario el contenido de la sección seleccionada.
- Evaluar la accesibilidad del usuario a cada sección.

Formulario

La clase formulario deriva de la clase sección, puesto que se trata de una sección en la que el usuario podrá insertar información a través de un conjunto de inputs insertados en formularios prefabricados.

Esta clase es una plantilla genérica que define los atributos y funciones mínimas imprescindibles para que dichos formularios puedan funcionar independientemente de su finalidad. Se utilizará la herencia para construir formularios específicos con funcionalidades relevantes.

Esta clase define funciones que deberán ser reescritas por las clases herederas para encargarse de:

- Detectar los inputs introducidos por el usuario.
- Almacenar la información introducida por el usuario.

Filtro

La clase filtro es deriva de la clase formulario, puesto que se trata de un formulario particular que se inserta como atributo en muchas secciones. Aquellas secciones que lo incorporen mostrarán al inicio de la página un formulario HTML con un conjunto de campos genéricos que permiten acotar la información que se pretende visualizar en dicha sección.

Su funcionamiento consiste en preconfigurar los bloques ORDER BY y WHERE de las consultas que se utilizan para cargar la información que se muestra en cada sección.

Cuando se inicializa el filtro es necesario indicar una lista con los atributos a través de los cuales se podrá filtrar

Gráfica

La clase gráfica deriva de la clase sección, puesto que se trata de una sección en la que el usuario podrá visualizar información a través de un conjunto de tablas y gráficas prefabricadas.

Esta clase es una plantilla genérica que define los atributos y funciones mínimas imprescindibles para que dichas gráficas puedan funcionar independientemente de su finalidad. Se utilizará la herencia para construir gráficas específicas con funcionalidades relevantes.

HTML Manager

Se trata de una librería que contiene funciones genéricas que simplifican la construcción de las etiquetas HTML de la página mediante la generación de plantillas dinámicas que se pueden configurar a través de los atributos de las funciones.

Su finalidad es la de simplificar el código HTML agrupando en funciones aquellos bloques que se utilizan de forma recurrente.

SQL Manager

Se trata de una librería que contiene funciones genéricas que simplifican la construcción de las consultas a la base de datos mediante la generación de plantillas dinámicas que se pueden configurar a través de los atributos de las funciones.

Su finalidad es la de simplificar el código SQL agrupando en funciones aquellos bloques que se utilizan de forma recurrente.

Ficheros

Se trata de una librería que contiene las funciones necesarias para leer un fichero CSV, realizar las comprobaciones necesarias para validar el contenido y delegar al SQL Manager la introducción de los datos recabados.

Scraper

Se trata de una librería que contiene las funciones necesarias para identificar mediante Web Scraping los datos necesarios acerca de las cotizaciones históricas de un activo específico, realizar las comprobaciones necesarias para validar el contenido y devolverlo como resultado en el formato adecuado.

Utilidades

Se trata de una librería utilizada como cajón desastre para múltiples funciones menores cuya finalidad va desde validar si un input tiene un formato específico hasta generar un valor aleatorio para un color.

Diagrama Secuencia

A continuación, se van a representar algunos de los diagramas de secuencia más relevantes para la aplicación. Todos los diagramas de secuencia que no se muestren en este apartado se podrán encontrar en el apéndice tres, "Documentación del código".

Gestión de usuarios

Registrarse

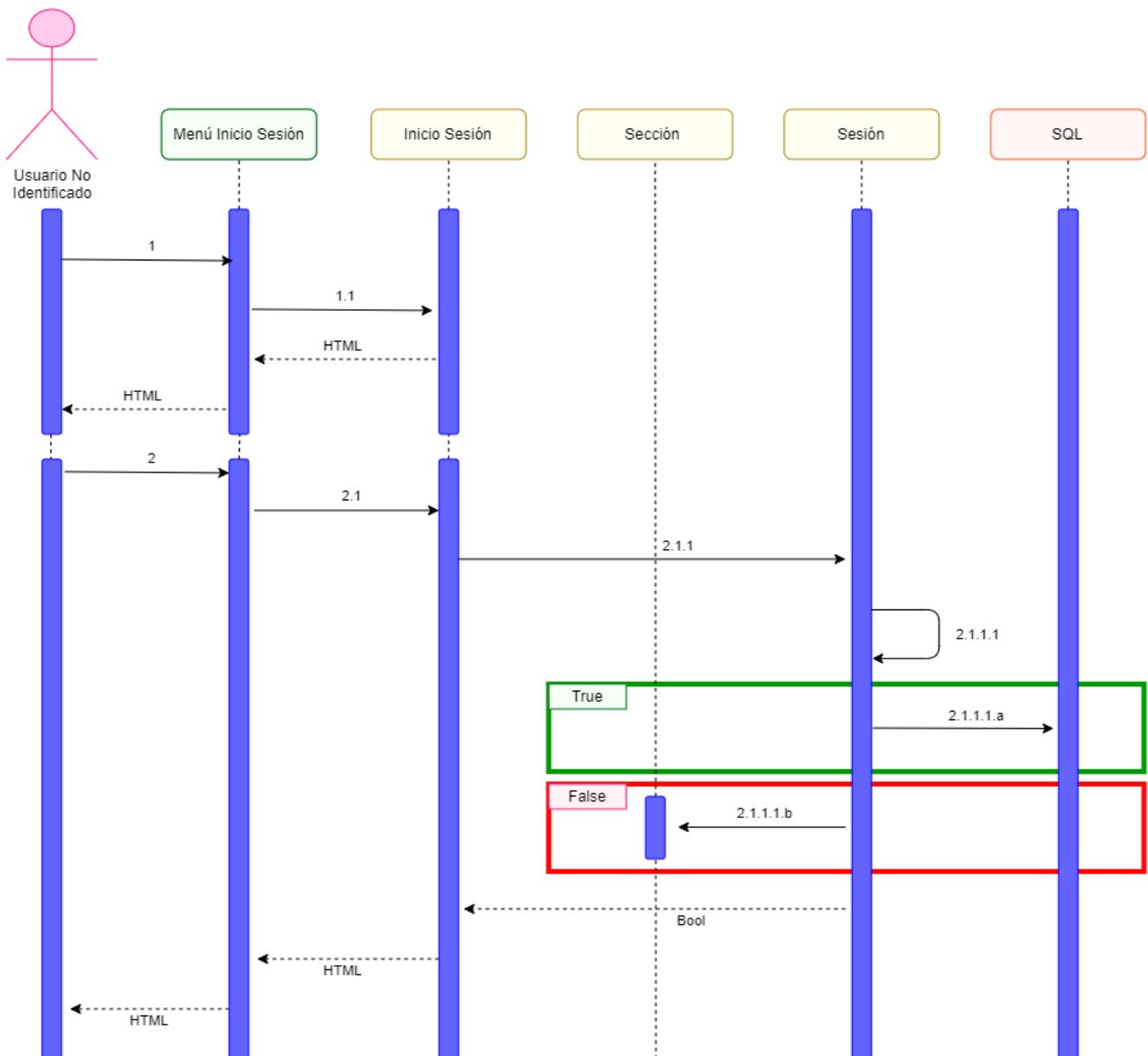


Ilustración 69 Diagrama de flujo – Registrarse

- 1 El Usuario No Identificado accede al formulario de registro
 - 1.1. Se ejecuta la función Generar Contenido de la clase Sección para la subclase Registro
- 2 El Usuario No Identificado rellena el formulario de registro y pulsa el botón de aceptar
 - 2.1. Se ejecuta la función Generar Contenido de la clase Sección para la subclase Registro
 - 2.1.1. Se ejecuta la función Añadir Admisión de la clase Sesión
 - 2.1.1.1. Se ejecuta la función Comprobar admisión de la clase Sesión
 - 2.1.1.1.1. En caso verdadero ejecuta la función Añadir Admisión de la librería SQL
 - 2.1.1.1.2. En caso negativo se ejecuta la función Reportar Error de la clase Sesión

Identificarse

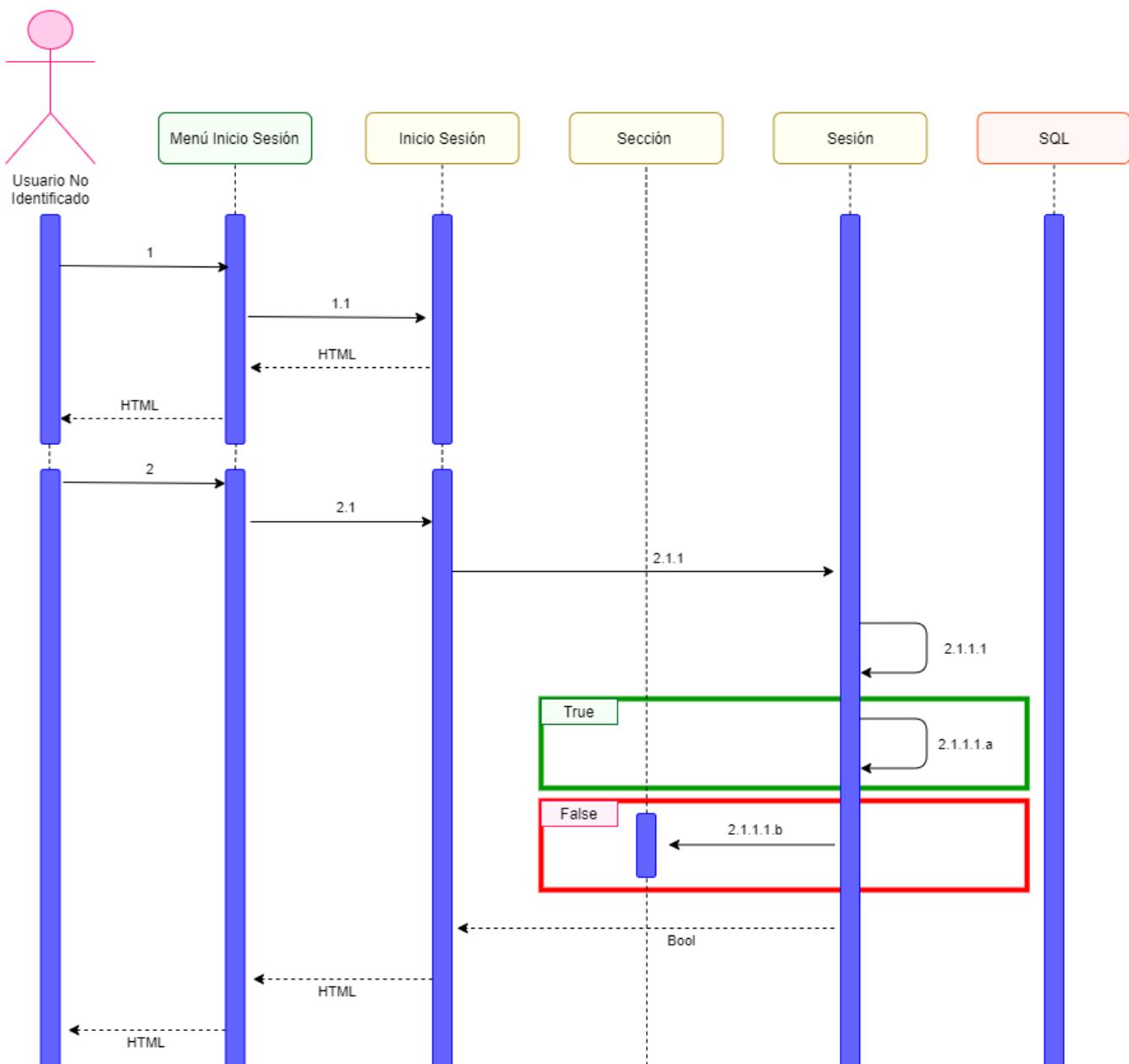


Ilustración 70 Diagrama de flujo – Identificarse

- 1 El Usuario No Identificado accede al formulario de registro
 - 1.1. Se ejecuta la función Generar Contenido de la clase Sección para la subclase Inicio Sesión
- 2 El Usuario No Identificado rellena el formulario de registro y pulsa el botón de aceptar
 - 2.1. Se ejecuta la función Generar Contenido de la clase Sección para la subclase Inicio Sesión
 - 2.1.1. Se ejecuta la función Iniciar Sesión de la clase Sesión
 - 2.1.1.1. Se ejecuta la función Comprobar Credenciales de Sesión de la clase Sesión
 - 2.1.1.1.a En caso verdadero se actualizan los atributos Usuario, Permisos y Contraseña
 - 2.1.1.1.b En caso negativo se ejecuta la función Reportar Error de la clase Sección

Funciones de apoyo

Generar Contenido

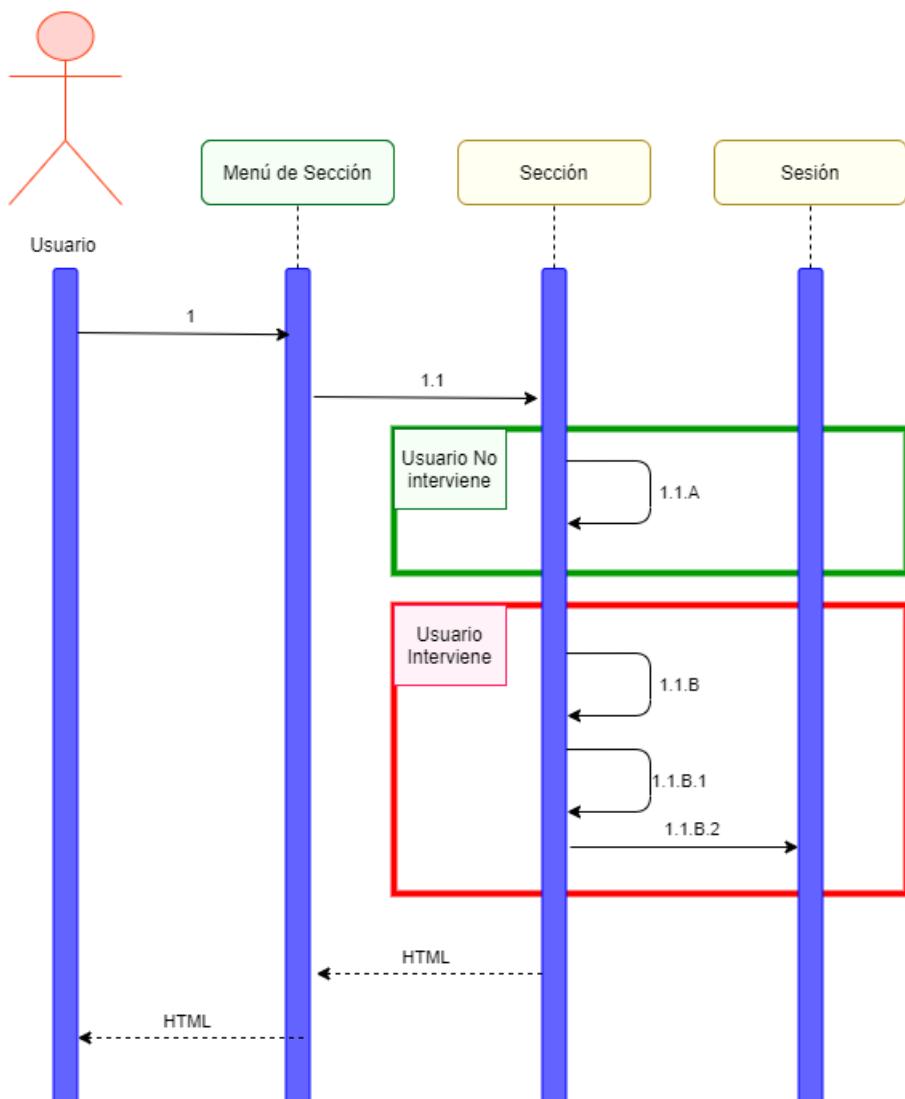


Ilustración 71 Diagrama de flujo – Generar Contenido

- 3 El Usuario No Identificado accede al formulario deseado
 - 3.1. Se ejecuta la función Generar Contenido de la clase sección
 - 1.1.A Si el usuario no ha completado el formulario de la sección Seleccionada
 - 1.1.A.1 Se ejecuta la función Generar Formulario de la subclase sección Seleccionada
 - 1.1.B Si el usuario ha completado el formulario de la sección Seleccionada
 - 1.1.B.1 Se ejecuta la función Capturar Campos de la subclase sección Seleccionada
 - 1.1.B.1.1 Se obtienen los datos de los campos del formulario
 - 1.1.B.1.2 Se ejecuta la función de la clase Sesión Asociada a la sección

Estructura de los Directorios

Cuando el proyecto en el que se trabaja pretende alcanzar ciertas dimensiones, es muy importante que los distintos ficheros que lo componen estén bien organizados en directorios con la finalidad de que sean fácilmente identificables.

En este proyecto se ha utilizado la siguiente distribución:

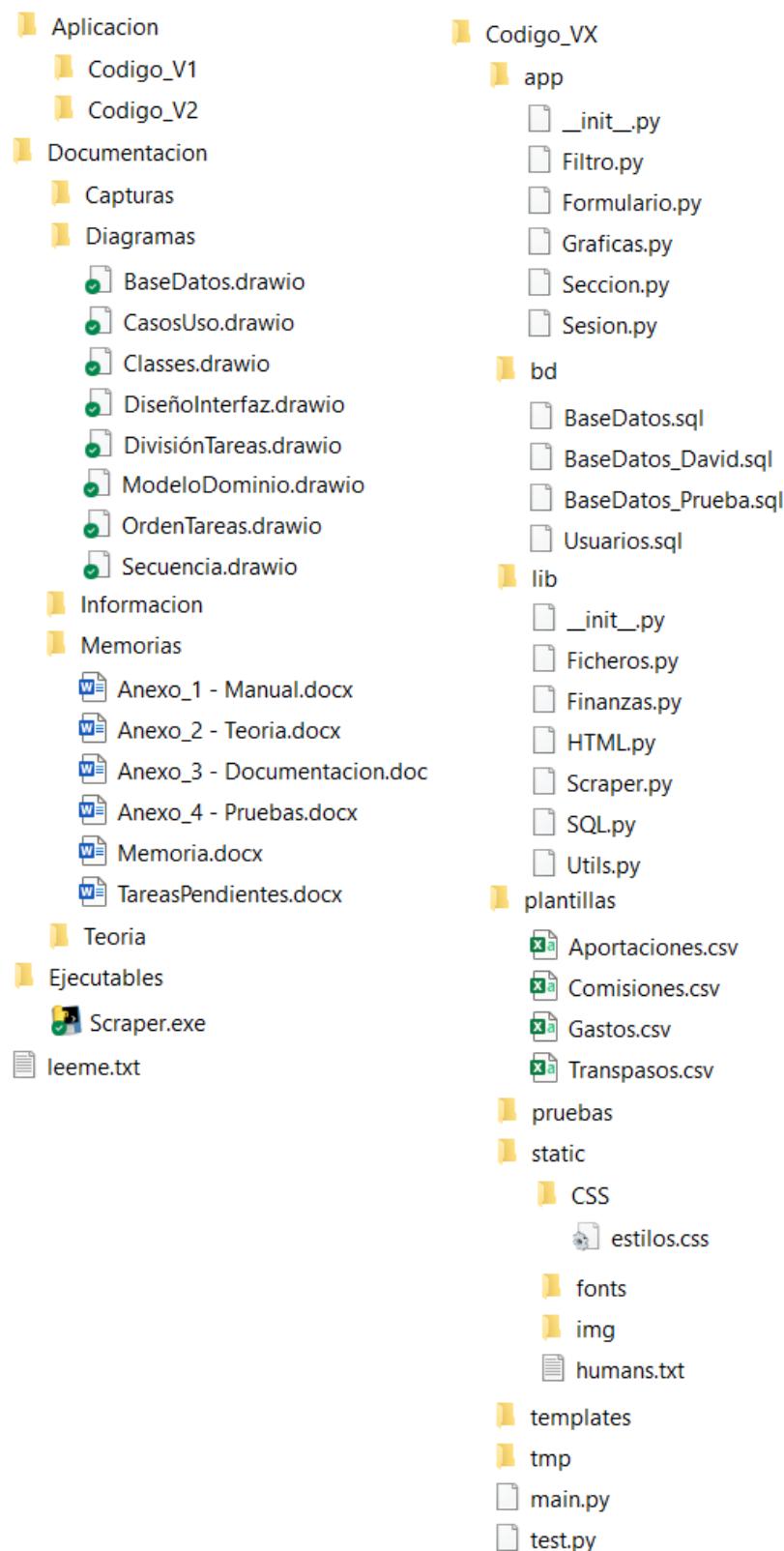


Ilustración 72 Estructura de los Directorios

Aplicación

En el subdirectorio de la aplicación encontraremos las diferentes versiones del código de la aplicación organizadas conforme a la siguiente estructura:

- En la raíz se encuentran los ficheros iniciales, tanto del código como de las pruebas unitarias.
- La carpeta “*app*” contiene las clases principales de la aplicación.
- La carpeta “*bd*” contiene el código de las bases de datos.
 - o Aquellos ficheros con una barra baja indican una versión con diferentes datos iniciales.
- La carpeta “*lib*” contiene las clases de apoyo de la aplicación.
- La carpeta “*plantillas*” contiene los ficheros *csv* utilizados por la librería “Ficheros”.
- La carpeta “*pruebas*” contiene las clases que implementan las pruebas unitarias.
- La carpeta “*static*” contiene todos los ficheros de apoyo para el correcto funcionamiento de la página web. Tales como los ficheros de estilo *css*, las fuentes de texto, las imágenes, los ficheros de configuración y el código JavaScript.
- La carpeta “*templates*” agrupa todas las plantillas HTML que se invocarán desde el código para configurar la vista del usuario.
- La carpeta “*tmp*” es un subdirectorio donde la aplicación podrá generar ficheros temporales que serán eliminados en algún momento de la ejecución antes del cierre de la sesión.

Documentación

En el subdirectorio de documentación encontraremos los registros y las anotaciones de la aplicación:

- La carpeta “*Capturas*” contiene algunas de las capturas de pantalla utilizadas en la documentación.
- La carpeta de “*Diagramas*” contiene los diagramas originales incluidos en la documentación.
- La carpeta de “*Informacion*” contiene algunas anotaciones acerca de funcionalidades o conceptos pendientes de revisión para incluir en la documentación.
- La carpeta de “*Memorias*” contiene los documentos finales de la aplicación.

Ejecutables

En el subdirectorio de ejecutables se almacenan los archivos ejecutables de la aplicación.

Esta sección sólo tenía sentido para la primera versión de la aplicación, que no estaba destinada a ser una página web.

Conclusiones

Este apartado de la documentación está destinado a recapitular la situación final de la aplicación. De este modo, se analizará su funcionalidad tras finalizar el desarrollo, así como las principales dificultades que se han presentado y las experiencias que se han extraído de su elaboración.

Resultado

Aunque la aplicación dista mucho de cumplir todas las expectativas con las que había sido concebida inicialmente. Se han cumplido la mayoría de los objetivos iniciales:

- El usuario puede introducir sus datos de gastos e ingresos.
- El usuario puede introducir los datos de sus activos.
- El usuario puede clasificar sus datos financieros en categorías y subcategorías personalizables.
- La aplicación obtiene los datos de cotización históricos y recientes.
- La aplicación utiliza estos datos para calcular indicadores que organiza gráficas y tablas.
- Se pueden administrar los usuarios registrados, así como las solicitudes de registro.
- Los datos de los usuarios se almacenan encriptados.

Evolución

Este proyecto se inició mucho antes de que fuera propuesto como trabajo de fin de grado. Surgió como aplicación práctica experimental de algunas de las herramientas que habíamos utilizado durante la carrera, junto con mi creciente interés por el mundo de las finanzas.

Inicialmente la aplicación consistía en un Scraper que tomaba de la web las cotizaciones actuales de los activos, un lector que obtenía las transacciones y aportaciones de los ficheros de texto donde estaban almacenadas y una calculadora que realizaba ciertos cálculos que se imprimían por consola.

Debido a la necesidad de encontrar un proyecto que sirviera como trabajo de fin de grado se agregó una base de datos que permitiera almacenar información permanente de varios usuarios y múltiples interfaces construidas con HTML que permitían manejar los datos de forma manual, además de gráficas que representaran dicha información.

Para diseñar esta segunda versión de la aplicación se combinaron los elementos que ya existían con algunas piezas de otros proyectos elaborados durante la carrera. No obstante, debido a la rigidez que presentaba la combinación de ambos sistemas fue necesario reformular la aplicación para que todas las partes encajaran mejor.

Durante esta reformulación, se añadió un sistema de encriptación para la base de datos, se reformuló el Scraper para que utilizara la API Alphavantage en vez de obtener los datos directamente de la Web y se añadió un sistema de caché que permitiera no tener que buscar nuevamente información que se hubiera buscado recientemente.

Dificultades

A lo largo de la ejecución del proyecto han tenido que enfrentarse algunos de los riesgos que se habían previsto durante la planificación del proyecto y otros que no se habían podido prever.

Técnicas

Encriptación

La aplicación consta de una Base de Datos en la que se almacena información sobre aspectos económicos de los usuarios registrados. Como esta información se considera sensible, es necesario implementar un sistema de encriptación que permita proteger dicha información.

Para gestionar el acceso, cada usuario dispone de una contraseña que sólo se almacena durante el tiempo que la sesión está iniciada, siempre en la máquina local y nunca en la base de datos. Con la finalidad de validar los accesos, durante el inicio de sesión, se calcula una función Hash sobre la contraseña y un valor aleatorio almacenado en un campo de la base de datos del usuario. El resultado de esta función se puede almacenar en un campo de la base de datos sin comprometer la contraseña, debido a que la función no se puede aplicar a la inversa sin pagar un alto coste computacional. Si la contraseña es correcta, el resultado de dicha función siempre debería ser el mismo, por lo que se compara el cálculo actual con el que se hizo al registrar la contraseña. Si coinciden, la contraseña es correcta y el usuario puede acceder los datos almacenados.

En la aplicación existen varios tipos de usuarios con diferentes niveles de acceso a ciertas funcionalidades. Uno de esos tipos es el “Administrador”. Una de las capacidades de un usuario administrador es que puede acceder a cierta información, pero no toda, sobre los demás tipos de usuario.

Uno de los principales problemas a resolver fue el conflicto que se generaba por el hecho de que un usuario tuviera que acceder a una información que se almacenaba encriptada bajo una contraseña que no se podía conocer.

La solución a este problema consistía en utilizar [grupos de acceso](#) que permite repartir los privilegios del acceso a los datos mediante un tipo especial de tablas para usuarios.

Esta solución implica otros desafíos:

- Al no encriptar cada dato de forma individual mediante la contraseña, no tengo claro si se está encriptando la información almacenada de forma que si alguien accediera al servidor y se descargara el fichero SQL no pudiera ver la información.
- Para que un usuario pueda llevar a cabo ciertas funcionalidades es necesario darle algunos privilegios que no quiero que puedan tener. Esto se debe a que no encuentro la manera de acotar el alcance algunos permisos:
 - o Para que un usuario pueda eliminarse a sí mismo y cambiar su contraseña es necesario darle el permiso para que pueda hacer eso mismo para cualquier usuario. No encuentro ninguna forma de limitar el alcance de GRANT CREATE USER a un único usuario.
 - o Para que un usuario pueda ver su información en la tabla de usuarios también puede ver la información de todos los demás usuarios. Si bien se puede acotar la información por columnas, no encuentro ninguna forma de acotar la información por filas.

Finalmente se resolvió este problema acotando el alcance de los permisos en función de los distintos tipos de usuario en la aplicación en vez de en la base de datos.

Base de datos

La aplicación consta de una Base de Datos que se implementó mediante la herramienta XAMPP.

Uno de los problemas sin resolver de la aplicación es que el servidor local se bloquea de forma aparentemente aleatoria con cierta frecuencia. Generalmente el problema se arregla reiniciando el servidor, aunque alguna vez ha sido necesario reinstalar el programa.

Sospecho que el problema se debe a la gestión de permisos para los distintos tipos de usuario, dado que el error comenzó a ocurrir después de introducirlos. No obstante, el problema ha quedado sin resolver para esta versión.

Scraper

Para obtener los datos de cotizaciones históricas y actuales se ha implementado una herramienta de Web Scraping que permite obtener dicha información de páginas web.

El Scraper también ha generado varios problemas que han quedado sin resolver para esta versión.

- El proceso de Scraping consume la mayor parte de los recursos de la aplicación, por lo que la aplicación aparece estancada y sin responder cuando se solicitan datos externos. Esto puede tardar incluso varios minutos en función del volumen de información solicitada.
- La conexión a la página de donde se obtiene la información puede devolver el error 500 de conexión “no se ha podido conectar con la fuente”. Este error se produce principalmente cuando no hay conexión a internet o cuando se han hecho demasiados intentos simultáneos y la página objetivo rechaza la conexión. Para solucionarlo basta con esperar unos minutos y volver a intentar, pero la aplicación no está preparada para que el Scraper no devuelva los datos que se le han solicitado.
- Con cierta frecuencia la fuente de los datos modifica la estructura con la que se ordena la información. Esto provoca errores como que se carguen datos en campos incorrectos o directamente que no se puedan cargar.

Sociales

Enfermedad

Debido a la larga duración del proyecto, como consecuencia de la percepción de que el proyecto no avanzaba en determinadas etapas, ha sido inevitable enfrentarse a la desmotivación. Todo esto ha retrasado algunas de las tareas del proyecto impidiendo que se completaran todos los objetivos previstos.

Diseño

Evaluación de Costes

Algunos de los problemas técnicos que se presentaron durante la ejecución de proyecto requirieron reformular el diseño de la aplicación, siendo necesario incluso volver a empezar la implementación de cero en una ocasión.

Gracias a que la aplicación era lo suficientemente modular no fue necesario rehacer todo el proyecto y se pudieron reutilizar gran parte de los formularios reduciendo considerablemente el impacto de tener que volver a empezar la implementación de la aplicación.

Todo esto ha retrasado algunas de las tareas del proyecto impidiendo que se completaran todos los objetivos previstos.

Tiempo de Desarrollo

Debido a varias causas se excedió el tiempo previsto para el proyecto. Esto ha causado que determinadas tareas que estaban previstas no se completaran. Entre estas tareas se encuentra la de subir la aplicación a un servidor e implementar las pruebas de unitarias y de campo que estaban previstas.

Implementación

Condiciones de Uso

La fuente de donde se obtenían los datos externos cambia ocasionalmente su estructura, obligando a reformular el Scraper para que se adapte a la nueva estructura de datos. Gracias al modularidad de la aplicación se ha reducido considerablemente esfuerzo que supone modificar la estructura de datos cada vez que se producen dichos cambios.

Despliegue

Servidor

Originalmente estaba previsto subir la aplicación a un servidor, con la finalidad de presentar la aplicación a un grupo de usuarios previamente seleccionados, facilitándoles una encuesta con determinados ejercicios que les facilitaran la evaluación y preguntas que encauzaran las opiniones a aspectos clave de las funcionalidades.

No obstante, la aplicación no se ha subido a ningún servidor debido a varios aspectos técnicos de la funcionalidad que han quedado sin resolver y que sin lugar a duda dificultarían el funcionamiento de la aplicación con múltiples usuarios funcionando al mismo tiempo.

Coste computacional

Determinadas funcionalidades de la aplicación presentan tiempos de carga elevados debidos a operaciones computacionalmente muy costosas.

Aprendizaje

Durante el periodo de implementación de la aplicación, he tenido que enfrentarme a problemas que se habían trabajado en varias asignaturas de la carrera, como por ejemplo la encriptación de información o el Web Scraping. Pero también se han tenido que enfrentar otros problemas que no se había trabajado con tanta profundidad como la construcción de gráficas o el enrutamiento de peticiones web en Python.

Actualmente existen algunas partes de la aplicación que me gustaría repetir de manera distinta debido a que por el camino he ido descubriendo nuevas herramientas y nuevos métodos de implementación. También existen algunas decisiones de diseño que me he replanteado a lo largo del periodo de implementación de la aplicación

- En la aplicación actual el código SQL está completamente aislado del resto de la aplicación, pero la forma de acceder a las funciones que se comunican con la base de datos son muy estrictas. Recientemente he descubierto una metodología que permitiría que el módulo de la base de datos fuera más independiente del resto de la aplicación.
- En la aplicación actual la interfaz gráfica se construye mediante funciones que concatenan cadenas de texto que representan el código HTML con los parámetros de entrada. A pesar de que estas funciones son muy reutilizables y permiten simplificar la construcción de las interfaces gráficas, la concatenación de cadenas de texto suele ser un proceso con un coste computacional alto. Recientemente he descubierto una metodología que permitiría transferir los parámetros directamente a una plantilla HTML.
- Recientemente he hallado algunas herramientas que permiten simplificar el diseño de aplicaciones web haciendo que el código sea más sencillo, intuitivo y reutilizable:
 - Bootstrap es un conjunto de herramientas de código abierto que ofrece una serie de estilos y clases CSS predeterminadas para cada uno de los componentes HTML.
 - jQuery es una librería de JavaScript que permite simplificar la manera de interactuar con los elementos del documento HTML mediante la manipulación directa del árbol DOM o el manejo de eventos.
 - DataTables es un componente JavaScript que permite simplificar la explotación de datos mediante la generación de tablas dinámicas. Facilitando algunas necesidades básicas que HTML5 no cubre como son la paginación, la ordenación o la obtención de datos de forma dinámica.
- La aplicación actual está diseñada para ser una aplicación web, pero probablemente sería mucho mejor que fuera una aplicación de escritorio debido a que:
 - En ningún momento es necesaria la interacción de los usuarios con otros usuarios ni con el servidor, más allá de las herramientas de gestión de usuarios.
 - Con esta decisión se podrían eliminar todos los problemas derivados del tratamiento de datos sensibles de los usuarios, debido a que estos datos nunca saldrían de sus dispositivos.

Sin duda, cuando retome este proyecto en el futuro, será necesario empezarlo de nuevo. No obstante, no me parece que este proyecto se pueda considerar una pérdida de tiempo, debido a que la experiencia que he ganado con su implementación permitirá mejorar la calidad de otras aplicaciones similares en el futuro.

Trabajo Futuro

Este apartado de la documentación está destinado a recapitular aquellas funcionalidades de la aplicación que se quedaron fuera de la implementación recogida en este proyecto. De este modo, se expondrán brevemente las posibles vías de expansión futura para la aplicación.

Tareas Pendientes

Debido a varias causas se excedió el tiempo previsto para el proyecto.

Esto ha causado que determinadas tareas que estaban previstas no se completaran.

Puesta en servicio

Originalmente estaba previsto subir la aplicación a un servidor. No obstante, debido a varios aspectos técnicos de la funcionalidad que han quedado sin resolver y que sin lugar a duda dificultarían el funcionamiento de la aplicación con múltiples usuarios funcionando al mismo tiempo.

Esta tarea queda como trabajo pendiente para el futuro.

Pruebas de campo

Una vez que la aplicación estuviera en marcha estaba previsto presentar la aplicación a un grupo de usuarios previamente seleccionados, facilitándoles una encuesta con determinados ejercicios que les facilitaran la evaluación y preguntas que encauzaran las opiniones a aspectos clave de las funcionalidades.

Tras finalizar las pruebas será necesario implementar los cambios necesarios para adaptarse más a las expectativas de los usuarios.

Esta tarea queda como trabajo pendiente para el futuro.

Monetización

Una vez que la aplicación estuviera en marcha estaba previsto buscar métodos de financiación como Google AdSense o Patreon.

Esta tarea queda como trabajo pendiente para el futuro.

Mejoras

Durante la planificación del proyecto se limitaron algunas de las funcionalidades que se han implementado con la finalidad de acotar el tiempo de implementación de la aplicación.

Herramientas de Gestión

No se han implementado todas las gráficas y resúmenes que se considerarían imprescindibles para una correcta planificación financiera. Por ejemplo, no se ha incluido ninguna herramienta de planificación temporal.

También sería necesario implementar un sistema que permita una mayor personalización de los datos que se muestran en las gráficas y resúmenes más allá de incluirlos en grupos de Categorías, Estrategias o Allocations.

FeedBack

No se ha implementado ningún sistema que permita al usuario saber cuándo la aplicación está realizando algún cálculo que tardará en ser procesado. Por ejemplo, incluir un elemento gráfico como una barra de carga solucionaría uno de los principales problemas a los que se enfrenta la aplicación actualmente.

Tipos de Imputaciones

No se han implementado todos los tipos de operaciones que se considerarían imprescindibles para una correcta planificación financiera. Por ejemplo, no se ha tratado la gestión de dinero prestado ni la gestión de activos no cotizados como los inmuebles.

Además, por motivos de simplificación, algunos de los eventos financieros se han tratado de forma agrupada debido a que tenían algunas características en común. Por ejemplo:

- La aplicación no diferencia entre recibir un dividendo y pagar una comisión debido a que ambos son una transacción automática derivada de una entidad externa. No obstante, ambas tienen una finalidad y un origen distintos.
- La aplicación no diferencia entre una compraventa y un traspaso de activos, de tal forma que cualquier traspaso se considera una compraventa. Si bien es cierto a que traspasar implica vender un activo para comprar otro utilizando el dinero que se recibió del primero, a nivel fiscal son hechos muy distintos.

Caché permanente

Debido a que el mayor tiempo de ejecución de la aplicación se lleva a cabo en el Scraper cuando accede a datos externos. Sería interesante plantearse el hecho de almacenar de forma permanente algunos de los datos que en la aplicación actual se almacenan de forma temporal.

Creando algunas tablas en la base de datos que almacenen las cotizaciones históricas y actuales de los activos que ya han sido buscados por el Scraper se podría comprobar si estos datos ya han sido buscados con anterioridad.

Esto acortaría considerablemente los tiempos de espera relacionados con el Scraper a costa de aumentar el espacio en disco que necesitaría la aplicación.

Desarrollo futuro

Durante la planificación del proyecto se dejaron fuera del tintero algunas de las funcionalidades que sería muy interesante añadir en el futuro.

Informes fiscales

Debido a que la aplicación reúne una gran cantidad de datos acerca de las finanzas del usuario, podrían adaptarse la aplicación para que redactara automáticamente un borrador de algunos de los informes fiscales solicitados anualmente por hacienda, como por ejemplo serían el modelo D6, el 720 o la declaración de la renta.

Automatización

Podría resultar muy útil a los usuarios que se implementara una herramienta que permitiera gestionar la ejecución automática de transacciones periódicas, para imputar aquellos gastos o ingresos que se producen de forma recurrente, sin que sea necesario que intervenga el usuario.

Por ejemplo, si todos los meses se debe pagar una cuota fija que no suele variar, resulta tedioso tener que imputar dicho dato todos los meses.

También podría ser útil automatizar la imputación automática, sin ninguna intervención del usuario, de determinados eventos financieros externos. Como, por ejemplo, cuando una empresa reparte dividendos que se recoja mediante el Scraper cuál es la cuantía recibida en función de las participaciones que se poseen y se impute automáticamente dicha cuantía.

Notificaciones

Otra funcionalidad que podría resultar muy útil a los usuarios es una herramienta que permita programar notificaciones automáticas cuando se produzcan determinados eventos financieros a los que el propio usuario se ha suscrito. Por ejemplo, el usuario quiere recibir una notificación cuando uno de sus activos reparta dividendos. O cuando uno de sus activos alcance un precio objetivo determinado.

Estilo artístico

Incluir un apartado que permita a los usuarios personalizar el aspecto visual de la aplicación. permitiéndoles modificar, por ejemplo, la fuente del texto o la paleta de colores.

Gestión de Objetivos

Otra funcionalidad que podría resultar muy útil a los usuarios es la de crear un presupuesto de referencia y una asignación de activos objetivo, con la finalidad de ayudar a los usuarios a controlar sus finanzas personales y mantenerse dentro de sus estrategias.

Esta herramienta permitiría establecer límites de gastos por categorías y medir la desviación de los activos del usuario con respecto a su Asset Allocation predefinido. De este modo se notificarían a los usuarios cuando hubieran rebasado dichos límites indicándoles la desviación.

Herramientas de Análisis

Mientras que las herramientas de finanzas personales se centran en permitir a los usuarios gestionar el flujo de caja generado por sus propios activos y los gastos e ingresos. Las herramientas de gestión de activos se centran en analizar la evolución del patrimonio del usuario.

Existe un tercer tipo de herramienta financiera que podría resultar útil a los usuarios. Se trata de herramientas de que permitan comparar las estadísticas de varios activos con fines de investigación, de modo que el usuario pueda utilizar esta aplicación para determinar qué activos le convienen más en función de sus estrategias o necesidades.

Bibliografía

Este apartado de la documentación está destinado a facilitar el entendimiento del resto del documento. De este modo se incluirán las definiciones de los términos clave para su correcto entendimiento, así como las referencias de las que se ha obtenido la información para su elaboración.

Tutoriales

Para alcanzar una correcta comprensión de aquellas herramientas con las que no estaba familiarizado antes de comenzar con la ejecución de este proyecto he recurrido a algunos tutoriales.

Aquellos que me han sido de mayor utilidad se recopilan a continuación:

Autor	Artículo
j2logo	Tutorial de Flask en español
Charts.org	Página oficial
Real Python	Construye un Web Scraper con Python utilizando BeautifulSoup
Invierte Racionalmente	Como funciona Portfolio Performance
Luis Cabrera Benito	Cifrado de MySQL con AES

Referencias

Toda la información utilizada para redactar esta documentación ha sido extraída de las fuentes externas que se exponen a continuación:

Autor	Artículo
Desarrollo Web	¿Qué es Python?
Desarrollo Web	¿Qué es MySQL?
Open Webinars	¿Qué es Flask?
Julián Pérez Porto y María Merino	¿Qué es Word?
Beatrix Calvo	¿Qué es Draw.io?
Luigys Toro	¿Qué es Anaconda?
Página Oficial	¿Qué es Notepad++?
Página Oficial	¿Qué es Anaconda?
Página Oficial	¿Qué es Spyder?
Página Oficial	¿Qué es 7Zip?
Página Oficial	¿Qué es XAMPP?
GCFGlobal	¿Qué es OneDrive?
Rankia	¿Qué es Rankia?
Wikipedia	Definiciones varias
Apuntes de clases	Conceptos varios
Gestión Pasiva	Rentabilidad ponderada por tiempo
Gestión Pasiva	Rentabilidad Media



Gestión de Finanzas Personales

**Grado en Ingería Informática de
Gestión y Sistemas de Información**

TRABAJO FIN DE GRADO **Anexo 1 – Manual de Usuario**



Universidad
del País Vasco
Euskal Herriko
Unibertsitatea

BILBOKO
INGENIARITZA
ESKOLA
ESCUELA
DE INGENIERÍA
DE BILBAO

Alumno: Cuesta Alario, David

Director: López Cuadrado, Javier

Curso: 2020 – 2021

Fecha: 2020-10-19

Índice de contenido

ESTRUCTURA	1
SECCIONES	2
ENCICLOPEDIA	2
GESTIÓN DE USUARIOS	2
<i>Registro</i>	2
<i>Inicio Sesión</i>	3
<i>Cerrar sesión</i>	4
<i>Admisiones</i>	5
<i>Configuración</i>	7
GESTIÓN DE DATOS	10
<i>Transacciones</i>	10
<i>Importaciones</i>	37
ANÁLISIS DE DATOS	43
<i>Balance</i>	43
<i>Patrimonio</i>	45
<i>Flujo de Caja</i>	51

Índice de Ilustraciones

TÍTULO DE LA APLICACIÓN	1
ÍNDICE DE NAVEGACIÓN	1
CONTENIDO DE LA APLICACIÓN	1
REGISTRO DE USUARIOS	2
MENSAJES DE ERROR AL REGISTRAR USUARIOS.....	3
MENSAJES DE USUARIO PENDIENTE DE ADMISIÓN	3
INICIO DE SESIÓN	3
MENSAJES DE ERROR AL INICIAR SESIÓN	4
CERRAR SESIÓN	4
CADUCIDAD POR INACTIVIDAD.....	4
SECCIÓN DE ADMISSIONES	5
FILTRO DE ADMINISTRACIÓN.....	5
ADMINISTRACIÓN DE ADMISSIONES	6
ADMISIÓN DE USUARIOS	6
GESTIÓN DE ACCESOS	7
SECCIÓN DE CONFIGURACIÓN	7
EDITAR PERFIL	8
EDITAR PERFIL	9
SECCIÓN DE GESTIÓN DE TRANSACCIONES	10
SUBSECCIÓN DE DISTRIBUCIONES.....	10
GESTIÓN DE ALLOCATIONS	11
CREACIÓN DE ALLOCATIONS	12
FILTRO DE ALLOCATIONS.....	12
EDICIÓN DE ALLOCATIONS	12
GESTIÓN DE SUB-ALLOCATIONS	13
CREACIÓN DE SUB-ALLOCATIONS.....	13
EDICIÓN DE SUB-ALLOCATIONS	14
FILTRO DE SUB-ALLOCATIONS	14
FILTRO DE ALLOCATIONS	14
GESTIÓN DE CLASIFICACIONES.....	15
CREACIÓN DE CLASIFICACIONES	16
FILTRO DE CLASIFICACIONES.....	16
EDICIÓN DE CLASIFICACIONES	17
GESTIÓN DE SUB-CLASIFICACIONES	17
CREACIÓN DE SUB-CLASIFICACIONES	18
FILTRO DE SUB-CLASIFICACIONES.....	18
GESTIÓN DE ACTIVOS	19
AÑADIR ACTIVO	20
EDICIÓN DE ACTIVOS.....	21
GESTIÓN DE ASIGNACIONES DE SUB-ALLOCATION	21
ASIGNACIÓN DE SUB-ALLOCATION	22
EDICIÓN DE LA ASIGNACIÓN DE UN SUB-ALLOCATION	22
FILTRO DE ACTIVOS	23
SUBSECCIÓN DE CUENTAS	24

GESTIÓN DE CUENTAS	24
AÑADIR CUENTA	25
EDITAR CUENTA	25
SUBSECCIÓN DE OPERACIONES	26
GESTIÓN DE TRANSACCIONES	26
INSERCIÓN DE IMPORTACIONES	27
EDICIÓN DE TRANSACCIONES	28
FILTRO DE TRANSACCIONES	28
GESTIÓN DE APORTACIONES	29
INSERCIÓN DE APORTACIONES.....	30
EDICIÓN DE APORTACIONES.....	30
FILTRO DE TRANSACCIONES	31
GESTIÓN DE TRANSACCIONES.....	31
INSERCIÓN DE COMISIONES	32
EDICIÓN DE COMISIONES	33
FILTRO DE TRANSACCIONES	33
GESTIÓN DE TRANSACCIONES.....	34
INSERCIÓN DE TRASPASOS	35
EDICIÓN DE COMISIONES	35
FILTRO DE TRANSACCIONES	36
SECCIÓN DE IMPORTACIONES	37
IMPORTACIÓN DE DATOS.....	37
FORMATO DE LA PLANTILLA PARA LAS APORTACIONES.....	38
POSIBLES MENSAJES DE ERROR AL IMPORTAR APORTACIONES	38
FORMATO DE LA PLANTILLA PARA LAS COMISIONES	39
POSIBLES MENSAJES DE ERROR AL IMPORTAR APORTACIONES	39
FORMATO DE LA PLANTILLA PARA LAS GASTOS.....	40
POSIBLES MENSAJES DE ERROR AL IMPORTAR APORTACIONES	40
FORMATO DE LA PLANTILLA PARA LAS GASTOS.....	41
POSIBLES MENSAJES DE ERROR AL IMPORTAR APORTACIONES	41
EXPORTACIÓN DE DATOS.....	42
ANÁLISIS DEL BALANCE	43
FILTRO DEL BALANCE.....	43
BALANCE PATRIMONIAL.....	44
RESUMEN DE ACTIVOS.....	44
ANÁLISIS DEL PATRIMONIO.....	45
FILTRO DEL PATRIMONIO	45
GRAFICA DE DISTRIBUCIÓN DE ESTRATEGIAS	46
GRAFICA DE DISTRIBUCIÓN DE ACTIVOS	46
GRAFICA DE DISTRIBUCIÓN DE ALLOCATIONS	47
GRAFICA DE DISTRIBUCIÓN DE GASTOS	48
GRAFICA DE DISTRIBUCIÓN DE INGRESOS.....	49
GRAFICA DE COMPARACIÓN DE GASTOS E INGRESOS	50
TABLA DE DESGLOSE DE GASTOS E INGRESOS	50
ANÁLISIS DEL FLUJO DE CAJA	51
GESTIÓN DE ACCESOS	51

Estructura

Todas las vistas de usuario de la página web de gestión de finanzas personales están divididas en tres bloques:

- **Título:** Se trata del icono de la aplicación, que actúa como enlace de hipertexto a la página principal.



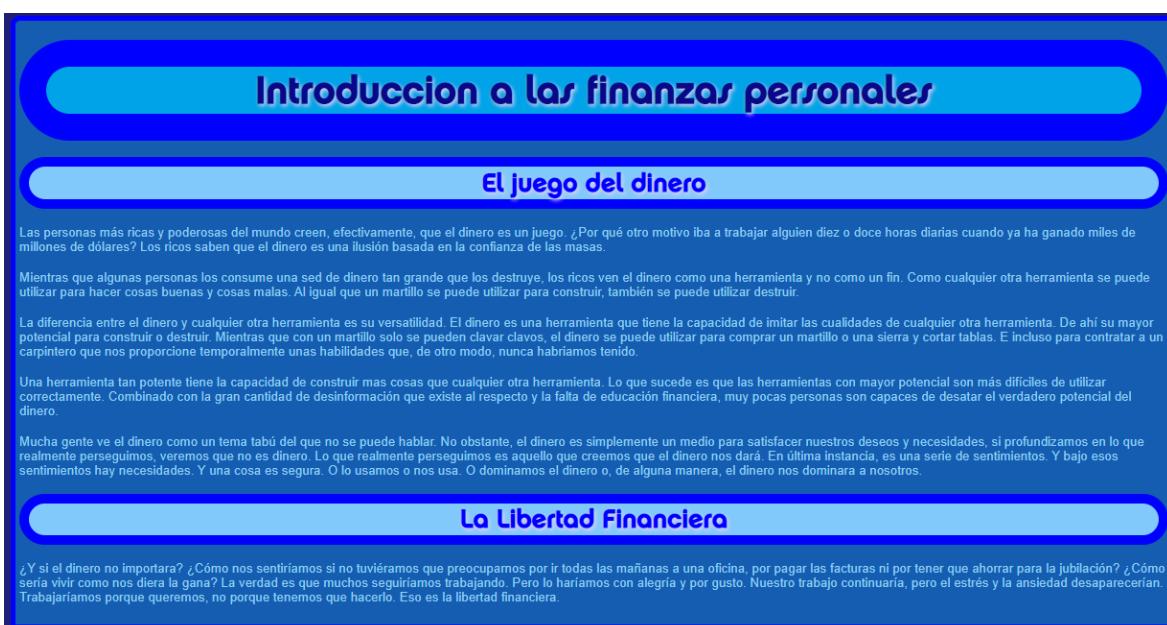
Ilustración 1 Título de la aplicación

- **Navegación:** Consiste en un conjunto de bloques desplegables que permiten navegar entre las distintas secciones de la aplicación.
 - Las opciones disponibles dependen de la ventana que se encuentre abierta en el momento actual.
 - Cada bloque del índice es dependiente del bloque superior.
 - Cada elemento del índice actúa como enlace de hipertexto a la sección indicada.



Ilustración 2 Índice de navegación

- **Contenido:** Mostrará la información de la sección actual. El contenido puede variar desde uno o más formularios hasta un conjunto de gráficos, tablas o bloques de información.



Introducción a las finanzas personales

El juego del dinero

Las personas más ricas y poderosas del mundo creen, efectivamente, que el dinero es un juego. ¿Por qué otro motivo iba a trabajar alguien diez o doce horas diarias cuando ya ha ganado miles de millones de dólares? Los ricos saben que el dinero es una ilusión basada en la confianza de las masas.

Mientras que algunas personas los consumen una sed de dinero tan grande que los destruye, los ricos ven el dinero como una herramienta y no como un fin. Como cualquier otra herramienta se puede utilizar para hacer cosas buenas y cosas malas. Al igual que un martillo solo se puede utilizar para construir, también se puede utilizar destruir.

La diferencia entre el dinero y cualquier otra herramienta es su versatilidad. El dinero es una herramienta que tiene la capacidad de imitar las cualidades de cualquier otra herramienta. De ahí su mayor potencial para construir o destruir. Mientras que con un martillo solo se pueden clavar clavos, el dinero se puede utilizar para comprar un martillo o una sierra y cortar tablas. E incluso para contratar a un carpintero que nos proporcione temporalmente unas habilidades que, de otro modo, nunca habríamos tenido.

Una herramienta tan potente tiene la capacidad de construir más cosas que cualquier otra herramienta. Lo que sucede es que las herramientas con mayor potencial son más difíciles de utilizar correctamente. Combinado con la gran cantidad de desinformación que existe al respecto y la falta de educación financiera, muy pocas personas son capaces de desatar el verdadero potencial del dinero.

Mucha gente ve el dinero como un tema tabú del que no se puede hablar. No obstante, el dinero es simplemente un medio para satisfacer nuestros deseos y necesidades, si profundizamos en lo que realmente perseguimos, veremos que no es dinero. Lo que realmente perseguimos es aquello que creemos que el dinero nos dará. En última instancia, es una serie de sentimientos. Y bajo esos sentimientos hay necesidades. Y una cosa es segura. O lo usamos o nos usa. O dominamos el dinero o, de alguna manera, el dinero nos dominará a nosotros.

La Libertad Financiera

¿Y si el dinero no importara? ¿Cómo nos sentiríamos si no tuviéramos que preocuparnos por ir todas las mañanas a una oficina, por pagar las facturas ni por tener que ahorrar para la jubilación? ¿Cómo sería vivir como nos diera la gana? La verdad es que muchos seguiríamos trabajando. Pero lo haríamos con alegría y por gusto. Nuestro trabajo continuaría, pero el estrés y la ansiedad desaparecerían. Trabajariamos porque queremos, no porque tenemos que hacerlo. Eso es la libertad financiera.

Ilustración 3 Contenido de la aplicación

Secciones

Enciclopedia

Se trata de un conjunto de secciones cuya finalidad es introducir al usuario en el mundo de las finanzas personales, mediante una serie de guías en las que se exponen conceptos fundamentales sobre economía.

Estas secciones se caracterizan porque:

- Son accesibles incluso si no se tiene ningún tipo de permisos sobre la aplicación.
- Se trata de páginas estáticas que no son editables por ningún tipo de usuario.

Gestión de usuarios

Se trata de un conjunto de secciones que permiten a los usuarios interactuar con la información de los usuarios.

Registro

Esta sección permite a los usuarios no identificados, generar solicitudes de acceso que serán posteriormente atendidas por los administradores de la aplicación.



The illustration shows a user registration form titled "Registro". The form consists of several input fields and a button. The fields are labeled with placeholder text: "Usuario: davidcuesta", "Nombre: Pepito", "Primer Apellido: Gonzalez", "Segundo Apellido: Martinez", "DNI: 12345678Z", "Teléfono: 123456789", "Email: miCorreoelectrónico@ho", "Contraseña:", and "Repita la contraseña: *****". Below the fields is a checkbox labeled "He leído y acepto las condiciones de uso". At the bottom is a green "Registrarse" button.

Ilustración 4 Registro de usuarios

La sección consta de un formulario que tras ser completamente cumplimentado genera:

- Un nuevo usuario sin ningún tipo de permisos.
- Un evento que permite a los usuarios administradores asignar algún nivel permisos al nuevo usuario o rechazarlo.



Para que el formulario se considere correctamente cumplimentado se deben cumplir las siguientes condiciones:

- Todos los campos obligatorios han sido completados.
- Los dos campos de contraseña deben coincidir.
- El nuevo usuario no existe y no es un usuario pendiente de admisión.
- Se aceptaron las condiciones de uso.

En caso de que alguna de las condiciones no se cumpla:

- Se mostrará un mensaje con los errores detectados.
- No se completará proceso de registro.

* Las contraseñas no coinciden
* Ese usuario ya existe
* Debe aceptar las condiciones de uso

Ilustración 5 Mensajes de error al registrar usuarios

En caso de que se cumplan todas las condiciones:

- Se genera una nueva admisión con todos los datos del nuevo usuario.
- Se redirige al usuario a la sección de inicio de sesión.
- Se muestra un mensaje indicando que el usuario está pendiente de admisión.

El usuario Pruebas ha sido añadido correctamente a la lista de admisiones

Ilustración 6 Mensajes de usuario pendiente de admisión

Esta sección se caracteriza porque es accesible incluso si no se tiene ningún tipo de permisos sobre la aplicación.

Inicio Sesión

Esta sección permite a los usuarios no identificados generar una sesión durante un tiempo limitado, permitiendo:

- Acceder a las funcionalidades avanzadas de la aplicación.
- Acceder a los datos del usuario que se identifica.

Usuario: davidcuesta

Contraseña: ...

Iniciar Sesión

Ilustración 7 Inicio de sesión



Esta sección consta de un formulario que, tras ser correctamente cumplimentado, modifica el nivel de acceso con los permisos que tuviera asignados el usuario que se identifica.

Para que el formulario se considere correctamente cumplimentado se deben cumplir las siguientes condiciones:

- El usuario está correctamente registrado y ha sido admitido previamente.
- La suma de validación de la contraseña introducida coincide con la almacenada para dicho usuario.

En caso de que alguna de las condiciones no se cumpla:

- Se mostrará un mensaje con los errores detectados.
- Se redirige al usuario a la sección de inicio de sesión.

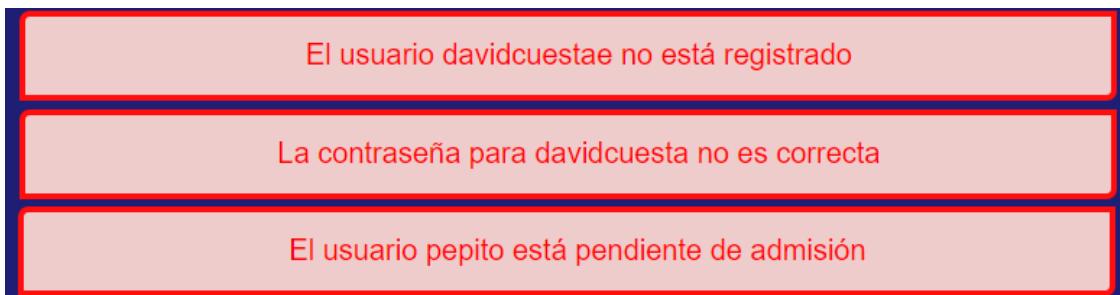


Ilustración 8 Mensajes de error al iniciar sesión

En caso de que se cumplan todas las condiciones:

- Modifica el nivel de acceso del usuario durante un tiempo limitado.
- Se redirige al usuario a la sección principal.

Cerrar sesión

Esta sección permite a los usuarios identificados cerrar la sesión manualmente.

Una vez identificado, el usuario puede cerrar la sesión manualmente accediendo a la sección de cerrar sesión.

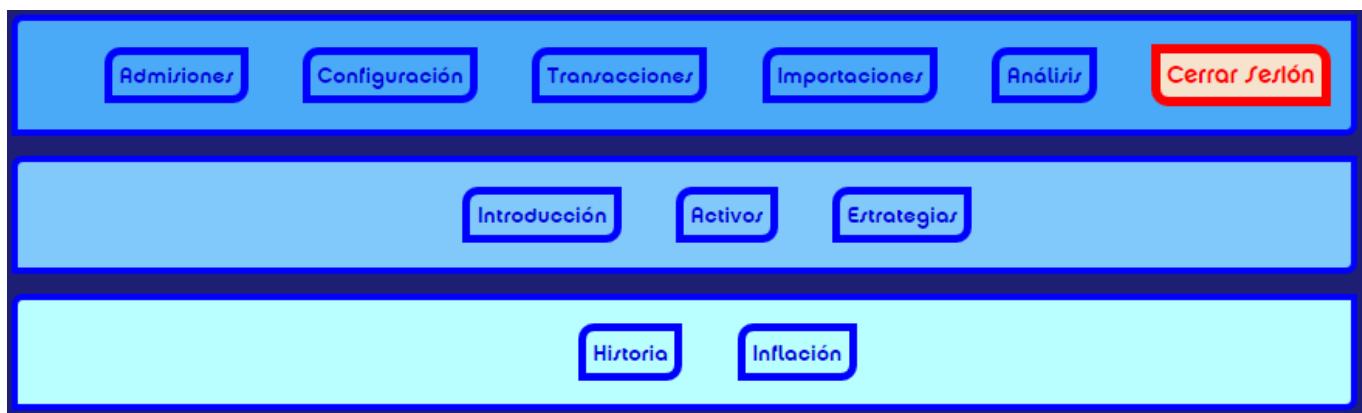


Ilustración 9 Cerrar sesión

Al acceder a esta sección se eliminará toda la información temporal del usuario que estuviera identificado.

Adicionalmente, si el usuario no muestra ninguna actividad durante un periodo de tiempo limitado, se bloqueará la sesión automáticamente.

- En caso de que se cierre la sesión se redirige al usuario a la sección de inicio de sesión.
- Si la sesión se cerró por falta de actividad se muestra un mensaje que así lo indica.



Ilustración 10 Caducidad por inactividad

Admisiones

La sección de admisiones permite administrar los nuevos registros, gestionar los permisos de los usuarios registrados y visualizar los accesos de los usuarios registrados sobre los que tengan permisos.

- Esta sección solo será visible y accesible para usuarios con nivel de administrador o superior.



Ilustración 11 Sección de admisiones

Todas las subsecciones de admisiones disponen de un filtro que permite acotar la lista de usuarios mostrados en función de ciertos campos.

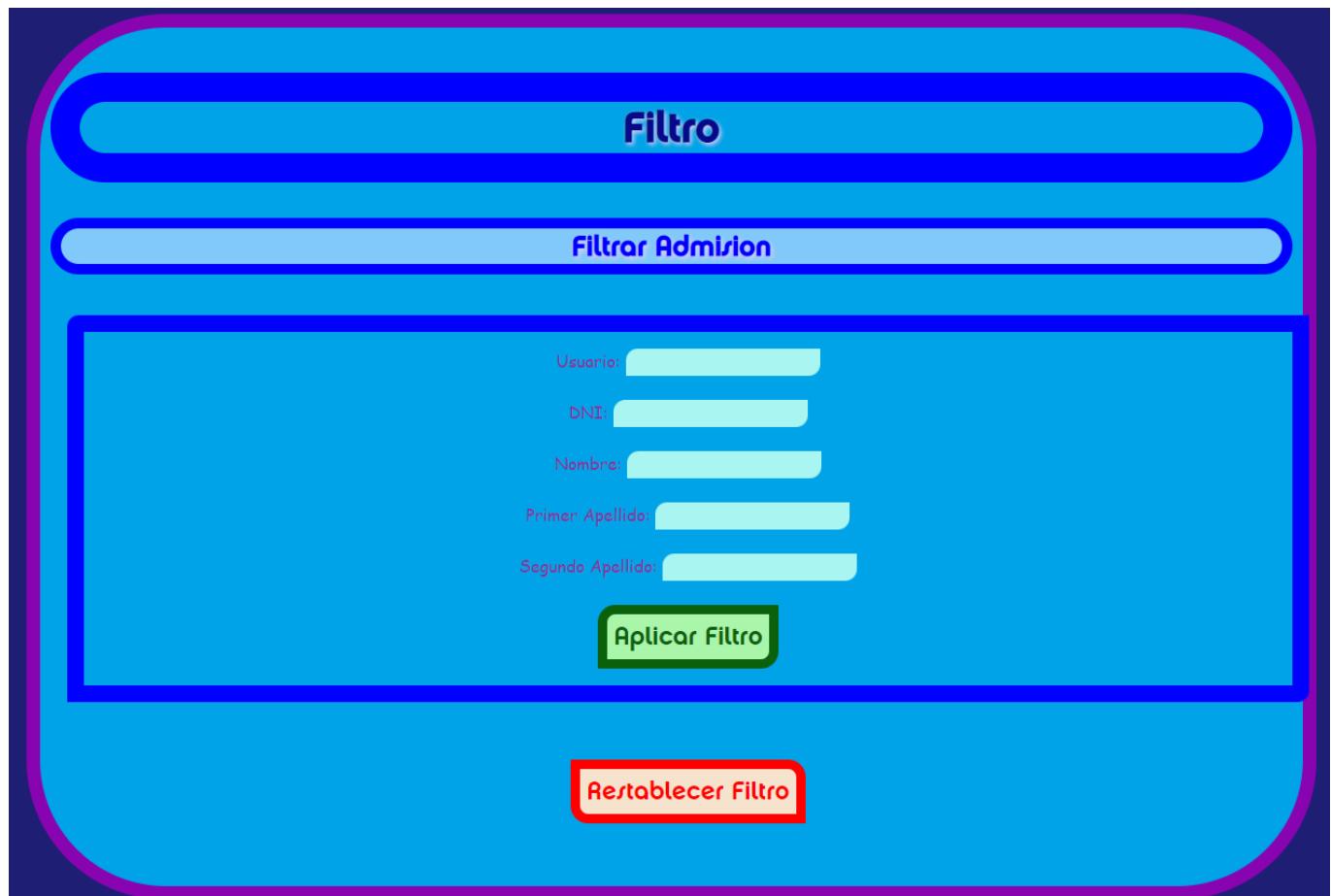


Ilustración 12 Filtro de Administración



Administración

La subsección de administración permite visualizar a todos los usuarios registrados y modificar los permisos o eliminar las cuentas de aquellos usuarios que tengan un nivel de permisos inferior.

Usuario	Permisos	
david cuesta alario ; david	Creator	<button>Eliminar</button>
david cuesta alario ; davidcuesta	Administrador	<button>Eliminar</button>
devilvill muy vil ; devilvill	Usuario	<button>Usuario</button> <button>Cambiar Permisos</button> <button>Eliminar</button>

Ilustración 13 Administración de admisiones

Admisión

La subsección de admisión permite aceptar o rechazar las solicitudes de registro de los nuevos usuarios.

En el caso de aceptar una solicitud hay que indicar el nivel de permisos con el que se creará la cuenta del usuario.

Usuario	
pepito perez perruno ; pepito	<button>Usuario</button> <button>Aceptar</button> <button>Rechazar</button>
David cuesta Alario ; Pruebas	<button>Usuario</button> <button>Aceptar</button> <button>Rechazar</button>
venganito perez perruno ; venganito	<button>Usuario</button> <button>Aceptar</button> <button>Rechazar</button>

Ilustración 14 Admisión de usuarios

Accesos

En la subsección de accesos, se mostrarán los intentos de acceso correctos o fallidos de los distintos usuarios, así como otra información relevante acerca de los cambios en la información de los usuarios, como por ejemplo, cambios de contraseña.

La finalidad de esta sección es facilitar a los administradores la detección de accesos no autorizados.

Esta sección no ha sido implementada.



Ilustración 15 Gestión de accesos

Configuración

Mediante la sección de configuración se pueden gestionar los datos del usuario identificado, así como las distintas opciones de configuración de la aplicación.

Se trata de un conjunto de secciones que permite a los usuarios identificados visualizar y modificar la información de los usuarios registrados sobre los que tengan permisos.



Ilustración 16 Sección de Configuración



Perfil

En la subsección del perfil podremos modificar todos los datos que se introdujeron durante la fase de admisión. A excepción del nombre de usuario, ya que este se utiliza como identificador único y vinculante.

The illustration shows a user interface for editing profile information. At the top, a blue header bar contains the word "Perfil". Below it, another blue header bar contains the text "Editar Perfil". The main content area is a light blue box containing several input fields for personal details:

- Nombre: david
- Primer Apellido: cuesta
- Segundo Apellido: alario
- DNI: 20857516N
- Teléfono: 615755325
- Email: davidcuestaalario@gmail.com

At the bottom of this section is a green button labeled "Aplicar Cambios" (Apply Changes).

Below this, another blue header bar contains the text "Cambiar Contraseña". The main content area is a light blue box containing three input fields for password changes:

- Contraseña Anterior: *****
- Contraseña Nueva: *****
- Repita la Contraseña: *****

At the bottom of this section is a green button labeled "Cambiar Contraseña" (Change Password).

Ilustración 17 Editar Perfil

Los formularios de Editar el perfil y cambiar de contraseña son independientes, para cumplimentarlos correctamente se debe tener en cuenta que:

- Todos los campos son obligatorios.
- Todos los campos deben cumplir las mismas restricciones que en el formulario de registro.
- Todos los campos se encuentran cumplimentados por defecto con el último valor establecido por el usuario.

De este modo:

- Para no modificar un campo simplemente debe no modificar su valor por defecto.
- Para cambiar la contraseña se le exigirá que escriba correctamente la contraseña anterior.

En caso de que alguna de las condiciones no se cumpla se mostrará un mensaje con los errores detectados.

Estilos

En la subsección de estilos se dispone de una serie de herramientas para modificar la configuración gráfica de la aplicación, como por ejemplo los colores de los fondos o la fuente de la letra.

Esta sección no ha sido implementada.



Ilustración 18 Editar Perfil

Gestión de Datos

Se trata de un conjunto de secciones que permiten a los usuarios gestionar su información financiera mediante la inserción, modificación o eliminación de datos.

Transacciones

En la sección de transacciones se podrán visualizar todos los datos previamente introducidos, así como añadir nuevos, editar o eliminar instancias específicas de forma individual.



Ilustración 19 Sección de gestión de Transacciones

Distribuciones

La subsección distribuciones permite crear, editar o eliminar, las etiquetas de clasificación, tanto para los activos, como para las transacciones.



Ilustración 20 Subsección de Distribuciones

Se trata de un sistema de segmentación de la información que permiten agrupar las distintas operaciones en subconjuntos con características comunes. La finalidad de estas etiquetas es poder analizar las implicaciones globales de cada operación.

Allocation

Los Allocations permiten administrar las distribuciones de los activos

Existen muchas segmentaciones posibles que es interesante evaluar a la hora de analizar una cartera. Una de las más habituales, por ejemplo, es la distribución geográfica. El valor de un activo puede estar repartido en diferentes zonas geográficas con diferentes pesos, y resulta de gran utilidad conocer cuánto pesa cada zona geográfica para un determinado conjunto de activos. Por ejemplo, ¿cuál es el peso de empresas europeas en el conjunto de activos que forman parte de una estrategia determinada?

Gestión de Allocations			
Allocation	Descripción	Añadir	Editar Filtros
Calidad	None	Editar	Eliminar
Capitalización	None	Editar	Eliminar
Composición	None	Editar	Eliminar
Entidad	None	Editar	Eliminar
Region	None	Editar	Eliminar
Sector	None	Editar	Eliminar
Vencimiento	None	Editar	Eliminar

Ilustración 21 Gestión de Allocations

Esta lista permite gestionar la visualización, inserción, modificación o eliminación de Allocations:

- Presionando el título de la cabecera se pueden ordenar alfabéticamente todas las entradas de la lista en función de la columna seleccionada.
- El botón eliminar de cada fila permite al usuario eliminar el Allocation de la fila seleccionada junto con todas sus dependencias.
- El botón de añadir redirige al usuario a un formulario en el que se pueden insertar nuevos Allocations.

Tras añadirse se creará un nuevo Allocation vacío con el identificador y la descripción asignadas.

En este formulario todos los campos son obligatorios:

- **Allocation:** Identificador único y vinculante del nuevo Allocation.
- **Descripción:** Nombre personalizado que el usuario asigne al Allocation.

Gestión de Allocations

Formulario para Crear Allocations

Allocation: Region

Descripción: Permite agrupar los activos

Anadir

Cancelar

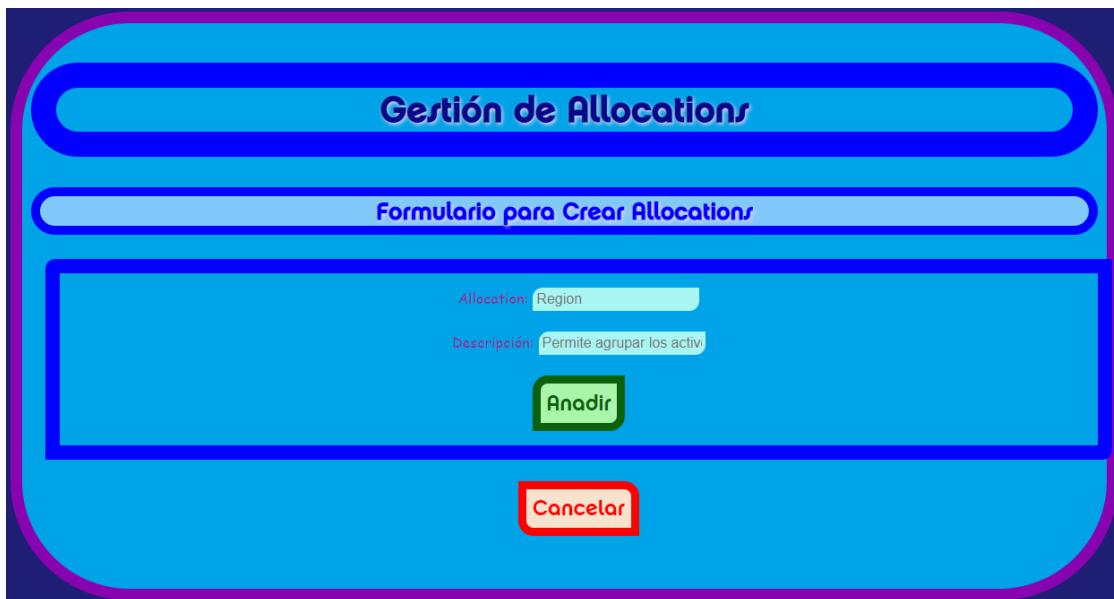


Ilustración 22 Creación de Allocations

- El botón de filtrar permite mostrar únicamente aquellos Allocations que cumplen con los requisitos especificados.

Filtro

Filtrar Allocation

Allocation:

Aplicar Filtro

Restablecer Filtro

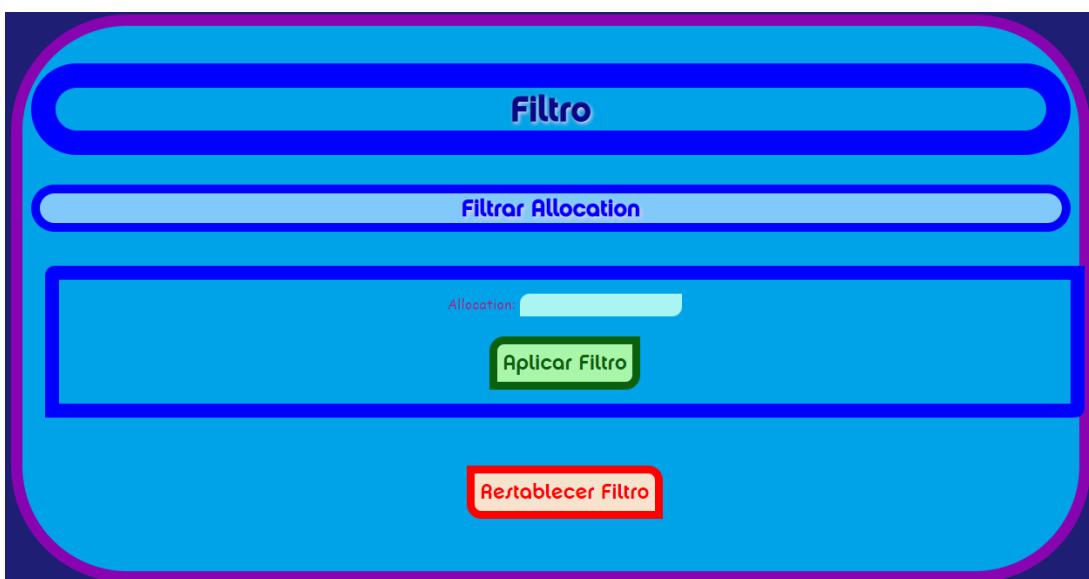


Ilustración 23 Filtro de Allocations

- El botón de editar de cada fila redirige al usuario a un nuevo menú donde puede:
 - o Modificar la descripción del Allocation seleccionado.

Gestión de Allocations

Formulario para Editar el Allocation Calidad

Descripción: None

Editar

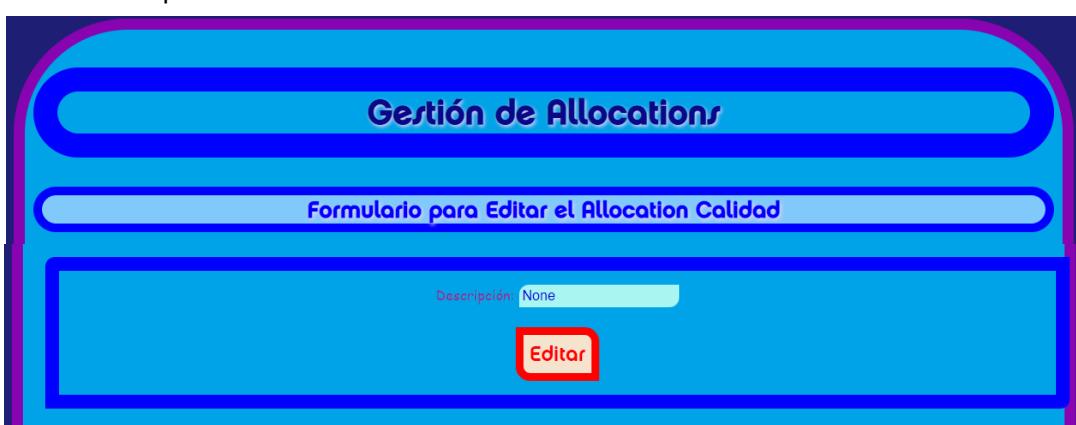


Ilustración 24 Edición de Allocations

Sub-Allocation

Los Allocations se componen de Sub-Allocations que representan cada una de las divisiones en las que se descompone un Allocation.

En el menú de edición de un Allocation encontraremos una lista que permite gestionar la visualización, inserción, modificación o eliminación de los Sub-Allocations del Allocation seleccionado permitiendo:

- Presionando el título de la cabecera se pueden ordenar alfabéticamente todas las entradas de la lista en función de la columna seleccionada.

SubAllocation	Descripción	Añadir	Editar Filtros
A		Editar	Eliminar
AA		Editar	Eliminar
AAA		Editar	Eliminar
B		Editar	Eliminar
BB		Editar	Eliminar
BBB		Editar	Eliminar

Cancelar

Ilustración 25 Gestión de Sub-Allocations

- El botón de añadir redirige al usuario a un formulario en el que se pueden insertar nuevos Sub-Allocations. Tras añadirse se creará un nuevo Sub-Allocation vacío con el identificador y la descripción asignadas.

Gestión de Allocations

Formulario para Crear SubAllocations del Allocation Calidad

SubAllocation: Region

Color:

Anadir

Cancelar

Ilustración 26 Creación de Sub-Allocations

- El botón de editar de cada fila redirige al usuario a un nuevo menú donde puede modificar el color del Sub-Allocation seleccionado.

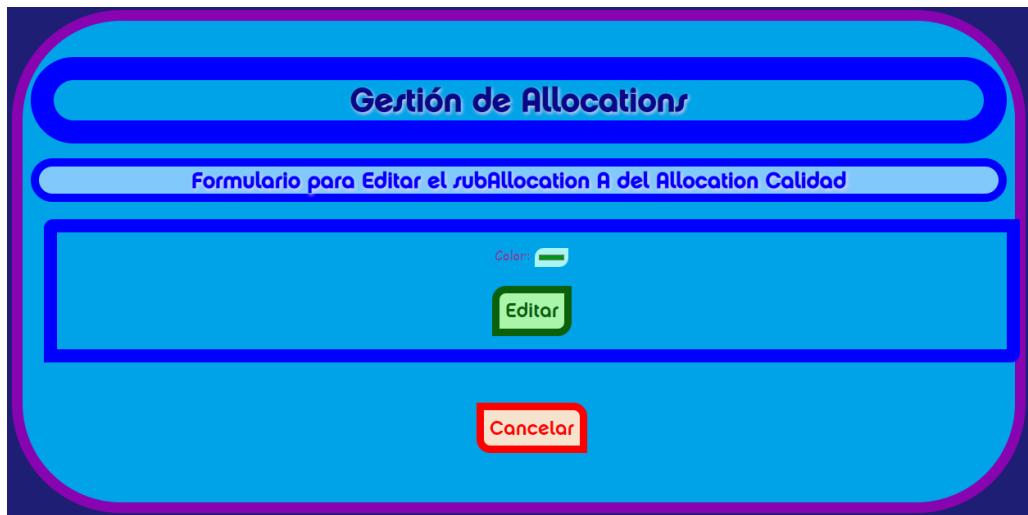


Ilustración 27 Edición de Sub-Allocations

- El botón de Filtrar permite mostrar únicamente aquellos Sub-Allocations que cumplen con los requisitos especificados:

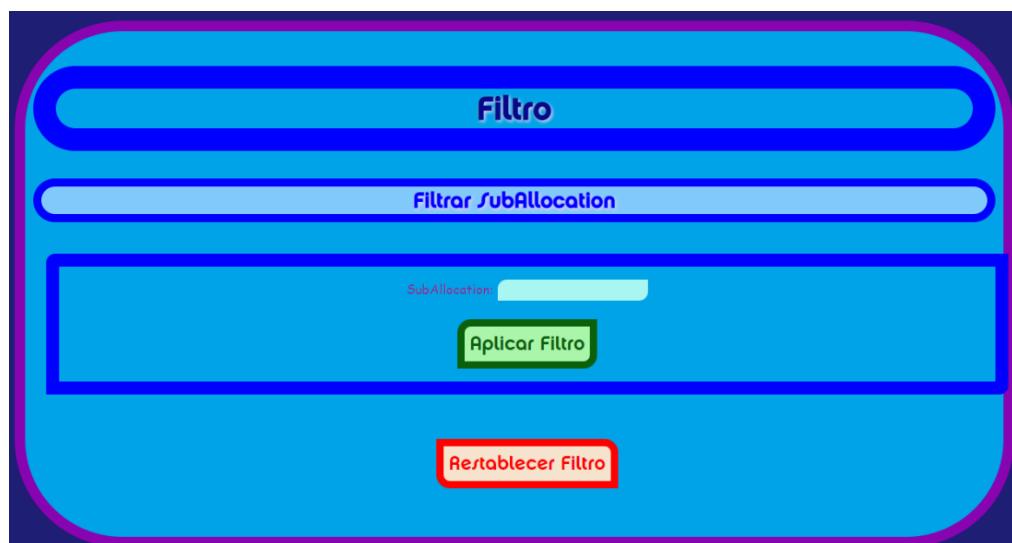


Ilustración 28 Filtro de Sub-Allocations

- El botón del filtro permite mostrar aquellos Allocations que cumplen con los requisitos especificados:

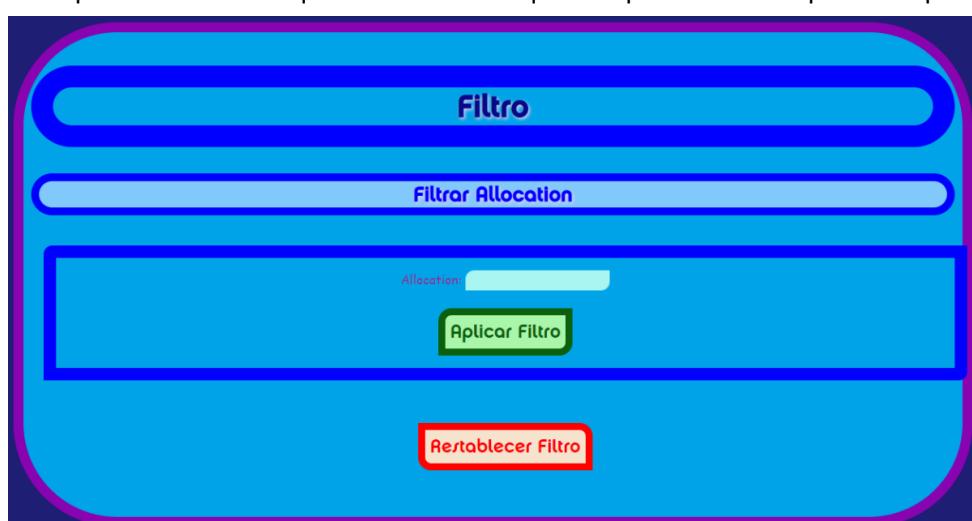


Ilustración 29 Filtro de Allocations

- El botón eliminar de cada fila permite al usuario eliminar el Sub-Allocation de la fila seleccionada junto con todas sus dependencias.

Clasificaciones

Las clasificaciones permiten administrar las distribuciones de las transacciones.

Existen muchas segmentaciones posibles que es interesante evaluar a la hora de analizar los gastos personales. Una de las más habituales, por ejemplo, es la distribución de transporte donde se pueden incluir los gastos de combustible o los bonos de metro.

Para clasificar las transacciones utilizaremos las clasificaciones:

Clasificación	Descripción	Color	Añadir	Editar Filtros
Apuestas	None		<button>Editar</button>	<button>Eliminar</button>
Caprichos	None		<button>Editar</button>	<button>Eliminar</button>
Compras	None		<button>Editar</button>	<button>Eliminar</button>
Estudios	None		<button>Editar</button>	<button>Eliminar</button>
Facturas	None		<button>Editar</button>	<button>Eliminar</button>
Salud	None		<button>Editar</button>	<button>Eliminar</button>
Transporte	None		<button>Editar</button>	<button>Eliminar</button>
Viajes	None		<button>Editar</button>	<button>Eliminar</button>
Vivienda	None		<button>Editar</button>	<button>Eliminar</button>

Ilustración 30 Gestión de Clasificaciones

Esta lista permite gestionar la visualización, inserción, modificación o eliminación de Clasificaciones:

- Presionando el título de la cabecera se pueden ordenar alfabéticamente todas las entradas de la lista en función de la columna seleccionada.



- El botón eliminar de cada fila permite al usuario eliminar La Clasificación de la fila seleccionada junto con todas sus dependencias.

- El botón de añadir redirige al usuario a un formulario en el que se pueden insertar nuevas Clasificaciones.

En este formulario todos los campos son obligatorios:

- **Clasificación:** Identificador único y vinculante de la nueva Clasificación.
- **Descripción:** Nombre personalizado que el usuario asigne a la Clasificación.

Tras añadirse se creará una nueva Clasificación vacía con el identificador y la descripción asignadas.

Gestión de Clasificaciones

Formulario para Crear Clasificaciones

Clasificación: Transporte

Descripción: Permite agrupar los gastos

Color:

Anadir

Cancelar

Ilustración 31 Creación de Clasificaciones

- El botón del filtro permite mostrar aquellos Allocations que cumplen con los requisitos especificados:

Filtro

Filtrar Clasificación

Clasificación:

Aplicar Filtro

Restablecer Filtro

Ilustración 32 Filtro de Clasificaciones



- El botón de editar de cada fila permite redirige al usuario a una lista donde se puede modificar la descripción de la clasificación seleccionada.

The screenshot shows a blue-themed user interface for editing a classification. At the top, a header bar reads "Gestión de Clasificaciones". Below it, a sub-header says "Formulario para Editar la Clasificación Apuestas". The main area contains fields for "Descripción" (set to "None") and "Color" (set to red), with a small color swatch next to the color name. A green "Editar" button is located at the bottom of this section.

Ilustración 33 Edición de Clasificaciones

Sub-Clasificación

Las Clasificaciones se componen de Sub-Clasificaciones que representan cada una de las divisiones en las que se descompone una clasificación.

En el menú de edición de una clasificación encontraremos una lista que permite gestionar la visualización, inserción, modificación o eliminación de los Sub-Clasificación de la Clasificación seleccionada permitiendo:

- El botón eliminar de cada fila permite al usuario eliminar la Sub-Clasificación de la fila seleccionada junto con todas sus dependencias.
- Presionando el título de la cabecera se pueden ordenar alfabéticamente todas las entradas de la lista en función de la columna seleccionada.

The screenshot shows a list of sub-classifications for the "Gastos" category. The header is "Formulario para Editar las SubClasificaciones de la clasificación Gastos". On the left, there's a column for "Sub-Clasificación" with items: Comida, Correos, Deporte, Reprografía, and Vivienda. To the right of each item is a red "Eliminar" (Delete) button. At the top right, there are buttons for "Añadir" (Add) and "Editar Filtros" (Edit Filters). At the bottom center is a "Cancelar" (Cancel) button.

Ilustración 34 Gestión de Sub-Clasificaciones



- El botón de añadir redirige al usuario a un formulario en el que se pueden insertar nuevas Sub-Clasificaciones.

The illustration shows a blue rounded rectangular form titled "Gestión de Clasificaciones" at the top. Below it is a sub-section titled "Formulario para Crear SubClasificaciones de la Clasificación Gastos". Inside this section, there is a text input field labeled "SubClasificación: Region" and a green button labeled "Anadir". At the bottom of the form is a red button labeled "Cancelar".

Ilustración 35 Creación de Sub-Clasificaciones

- El botón del filtro permitirá mostrar únicamente aquellas Sub-Clasificaciones que cumplen con los requisitos especificados:

The illustration shows a blue rounded rectangular form titled "Filtro" at the top. Below it is a sub-section titled "Filtrar SubClasificación". Inside this section, there is a text input field labeled "SubClasificación:" and a green button labeled "Aplicar Filtro". At the bottom of the form is a red button labeled "Restablecer Filtro".

Ilustración 36 Filtro de Sub-Clasificaciones

Activos

En la subsección activos podremos ver una lista con todos los activos registrados por el usuario identificado.

I/SIN	Descripción	Color	Precio	Producto	Estrategia	Añadir	Editar Filtros
AAPL	Apple		None USD	Acción	DGI	<button>Editar</button>	<button>Eliminar</button>
IE00B03HD191	Vanguard Global Stocks Index Fund Acumulation		None EUR	Fondo	BogleHead	<button>Editar</button>	<button>Eliminar</button>
IE00B18GC888	Vanguard Global Bonds Index Fund Acumulation		None EUR	Fondo	BogleHead	<button>Editar</button>	<button>Eliminar</button>
IE00B4L5Y983	iShares Core MSCI World UCITS ETF USD		None USD	ETF	BogleHead	<button>Editar</button>	<button>Eliminar</button>
IE00B4L5YC18	iShares Core MSCI EM UCITS ETF USD		None USD	ETF	BogleHead	<button>Editar</button>	<button>Eliminar</button>
IE00BFPM9N11	Vanguard Global Stocks Index Fund Acumulation Institucional		None EUR	Fondo	BogleHead	<button>Editar</button>	<button>Eliminar</button>
IE00BGCCZ0B53	Vanguard Global Bonds Index Fund Acumulation Institucional		None EUR	Fondo	BogleHead	<button>Editar</button>	<button>Eliminar</button>
ITX	Industria de Diseño Textil		None EUR	Acción	DGI	<button>Editar</button>	<button>Eliminar</button>
MSFT	Microsoft		None USD	Acción	DGI	<button>Editar</button>	<button>Eliminar</button>

Ilustración 37 Gestión de Activos

Esta lista permite gestionar la visualización, inserción, modificación o eliminación de dichos activos:

- Presionando el título de la cabecera de una columna se pueden ordenar alfabéticamente todas las entradas de la lista en función de la columna seleccionada.
- El botón eliminar de cada fila permite al usuario eliminar el activo de la fila seleccionada junto con todas las aportaciones registradas para dicho activo.

- El botón de añadir redirige al usuario a un formulario en el que se pueden insertar nuevos activos.

En este formulario todos los campos son obligatorios:

- **ISIN:** Identificador único oficial del producto que se pretende registrar.
- **Descripción:** Nombre personalizado que el usuario asigne al activo.
- **Color:** Color con el que aparecerá representado el activo en los diagramas que muestra la aplicación.
- **Emisor:** Nombre de la comercializadora que ofrece el producto. No confundir con el banco custodio.
- **Moneda:** Identificador único oficial de la moneda en la que se comercializa el activo. (**EUR, USD**).
- **Fuente de datos:** Nombre de la aplicación de donde se obtendrán los datos del activo que se pretende registrar ([Actualmente sólo está implementado para MorningStar](#)).
- **Enlace a la fuente de datos:** Identificador único que asigna la aplicación de la que se obtendrán los datos.
- **Tipo:** Tipo de activo que se pretende registrar.
- **Categoría:** Categoría, registrada por el usuario, en la que se engloba el activo.
- **Estrategia:** Estrategia, registrada por el usuario, que sigue el activo.



Gestión de Activos

Formulario para Crear Activos

ISIN: IE00B03HD191

Descripción: Nombre completo del pro

Color:

Emisor: Vanguard

Moneda: EUR

Fuente de datos: Morningstar

Enlace a la fuente de datos: F0GBR052TN

Tipo: Acción

Categoría: Acciones EEUU

Estrategia: BogleHead

Anadir

Ilustración 38 Añadir Activo



- El botón de editar de cada fila redirige al usuario a un conjunto de formularios que permiten:
 - o Modificar todos los valores establecidos durante el registro del activo, a excepción del ISIN ya que este se utiliza como identificador único y vinculante. En este formulario:
 - Todos los campos son obligatorios.
 - Se cargará por defecto el último valor establecido por el usuario para cada campo de modo que sólo es necesario alterar los campos que se pretenden modificar.

Formulario para Editar el Activo AAPL

Descripción: Apple

Color: #0072BD

Emisor: Vanguard

Moneda: USD

Fuente de datos: YahooFinance

Enlace a la fuente de datos: AAPL

Tipo: Acción

Categoría: Acciones EEUU

Estrategia: DGI

Editar

Ilustración 39 Edición de Activos

- o Gestionar los sub-Allocation asignados al activo de la fila seleccionada mediante una tabla que permite:
 - Visualizar todos los sub-Allocation que están asignados actualmente al activo seleccionado.

Allocation	Añadir	Editar Filtros
Region	Editar	Eliminar
Composición	Editar	Eliminar
Sector	Editar	Eliminar
Capitalización	Editar	Eliminar

Ilustración 40 Gestión de Asignaciones de Sub-Allocation

- Eliminar el sub-Allocation indicado del activo seleccionado.



- Añadir nuevos sub-Allocation al activo seleccionado. Se mostrará una lista desplegable donde:
 - Aparecerán aquellos sub-Allocation previamente configurados por el usuario, que no estuvieran ya asignados.
 - Se deberá seleccionar el sub-Allocation que se añadirá al activo seleccionado con sus valores por defecto. Para modificar los valores por defecto se debe utilizar la opción de editar sub-Allocation.

Gestión de Activos

Formulario para Añadir Allocations al Activo AAPL

Allocation: Calidad

Anadir

Cancelar

Ilustración 41 Asignación de Sub-Allocation

- Editar los valores de los sub-Allocation que están asignados actualmente al activo seleccionado.

Formulario para Editar el Sub-Allocation Region del Activo AAPL

Africa:	0,0
Asia:	22,0
Australia:	0,0
Canada:	0,0
Estados Unidos:	45,0
Europa:	25,0
Japon:	8,0
Latino America:	0,0
Oriente Medio:	0,0
Reino Unido:	0,0

Editar

Ilustración 42 Edición de la Asignación de un Sub-Allocation

- El filtro permite mostrar únicamente aquellos activos que cumplen con los requisitos especificados.

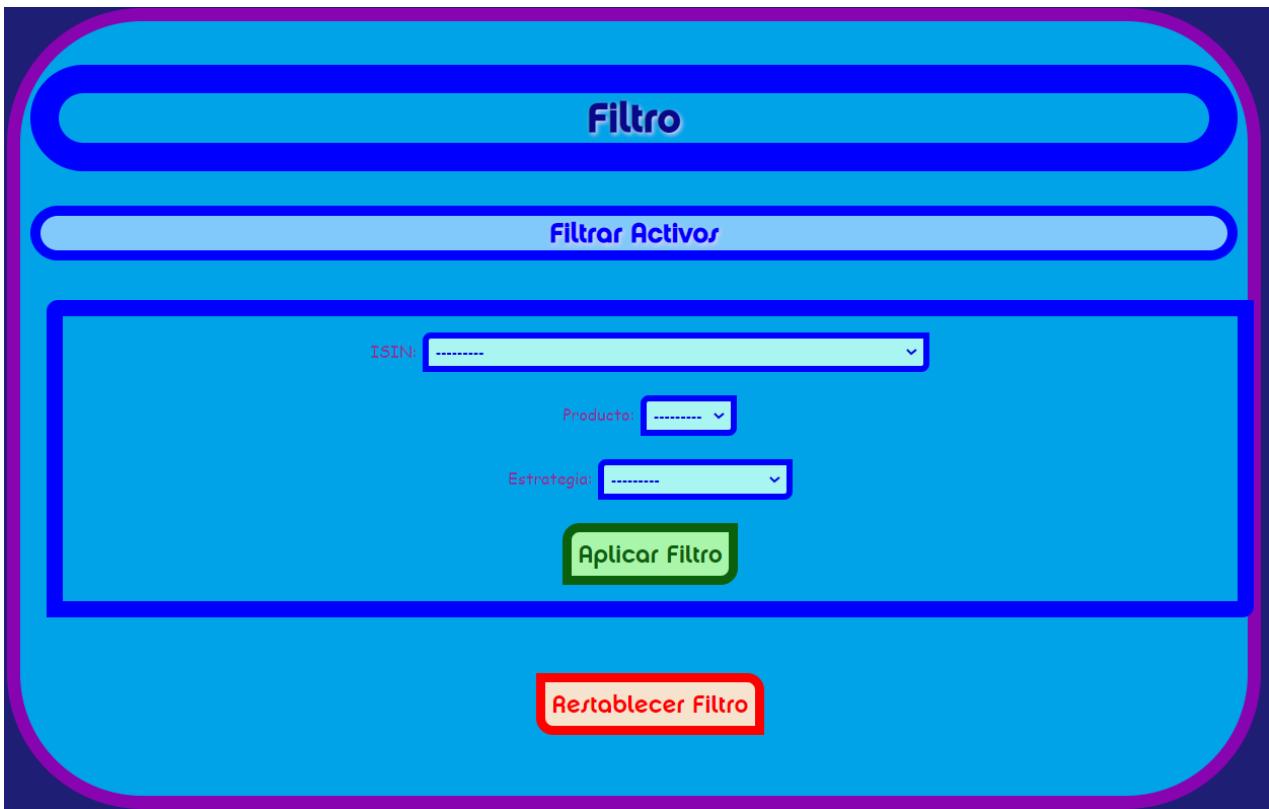


Ilustración 43 Filtro de Activos

Cuentas

En la subsección cuentas podremos ver una lista con todas las cuentas registradas por el usuario identificado.



Ilustración 44 Subsección de Cuentas

Cuenta	Moneda	País	Color	Descripción	Añadir	Editar Filtros
Bankia	EUR	España	■	None	Editar	Eliminar
ClickTrade	EUR	España	■	None	Editar	Eliminar
DeGiro	EUR	Holanda	■	None	Editar	Eliminar
Efectivo	EUR	España	■	None	Editar	Eliminar
IndexaCapital	EUR	España	■	None	Editar	Eliminar
MyInvestor	EUR	España	■	None	Editar	Eliminar

Ilustración 45 Gestión de Cuentas

Esta lista permite gestionar la visualización, inserción, modificación o eliminación de dichos cuentas:

- Presionando el título de la cabecera de una columna se pueden ordenar alfabéticamente todas las entradas de la lista en función de la columna seleccionada.
- El botón eliminar de cada fila permite eliminar la cuenta de la fila seleccionada junto con todas las dependencias registradas para dicha cuenta.



- El botón de añadir redirige al usuario a un formulario en el que se pueden insertar nuevos activos.

En este formulario todos los campos son obligatorios:

- o **Cuenta:** Identificador único de la cuenta que se pretende registrar.
- o **Descripción:** Nombre personalizado que el usuario asigne a la cuenta.
- o **País:** Nombre del país donde está la sede fiscal de la entidad.
- o **Moneda:** Identificador único oficial de la moneda en la que se opera. ([EUR](#), [USD](#)).
- o **Color:** Color con el que aparecerá representada la cuenta en los diagramas que muestra la aplicación.

The form has a blue header bar with rounded corners and a dark blue footer bar. It contains the following fields:

- Cuenta: Bankia
- Descripción: Fondo de emergencia
- País: España
- Moneda: EUR
- Color: (represented by a small color swatch)

At the bottom are two buttons: a green "Anadir" button and a red "Cancelar" button.

Ilustración 46 Añadir Cuenta

- El botón de editar de cada fila redirige al usuario a un formulario que permite modificar todos los valores establecidos durante el registro de una cuenta, a excepción del nombre de la cuenta, ya que este se utiliza como identificador único y vinculante.

The form has a blue header bar with rounded corners and a dark blue footer bar. It contains the following fields:

- Descripción: None
- País: España
- Moneda: EUR
- Color: (represented by a small color swatch)

At the bottom are two buttons: a green "Editar" button and a red "Cancelar" button.

Ilustración 467 Editar Cuenta

Operaciones

En la subsección Operaciones podremos ver una lista con todas las transacciones, aportaciones, comisiones y traspasos registradas por el usuario identificado y añadir o eliminar nuevas operaciones.



Ilustración 478 Subsección de Operaciones

Transacciones

Las transacciones son todas aquellas operaciones que generan gastos o ingresos en el día a día.

Fecha	Cuenta	Sub-Clasificación	Gasto	Ingreso	Descripción	Añadir	Editar Filtros
25/12/2016	Bankia	Abuela	--	200,0	Estrenas abuela AnaMari de navidad	Editar	Eliminar
13/03/2017	Bankia	Concierto	38,2	--	Concierto melendi	Editar	Eliminar
13/05/2017	Bankia	Concierto	3,5	--	Botella de agua	Editar	Eliminar
01/07/2016	Bankia	Correos	14,56	14,56	Envío de los documentos de la matrícula a Bilbao	Editar	Eliminar
06/04/2018	Bankia	Dentista	60,0	60,0	Revisión anual Dentista	Editar	Eliminar
30/07/2019	Bankia	Dentista	60,0	60,0	Revisión anual Dentista	Editar	Eliminar
11/08/2017	Bankia	Deporte	30,0	--	Material Escudo y lanza Soft Combat	Editar	Eliminar

Ilustración 49 Gestión de Transacciones



Esta lista permite gestionar la visualización, inserción, modificación o eliminación de las transacciones:

- Presionando el título de la cabecera se pueden ordenar alfabéticamente todas las entradas de la lista en función de la columna seleccionada.
- El botón eliminar de cada fila permite al usuario eliminar la transacción de la fila seleccionada junto con todas sus dependencias.
- El botón de añadir redirige al usuario a un formulario en el que se pueden insertar nuevas transacciones.

En este formulario todos los campos son obligatorios:

- **Fecha:** Día, mes y año en el que se produce la operación.
- **Cuenta de Origen:** Cuenta sobre la que se aplica el gasto o ingreso.
- **Subclasificación:** Segmentación en la que se engloba esta operación.
- **Gasto:** Cantidad de unidades monetarias que se sustraerán.
- **Gasto:** Cantidad de unidades monetarias que se añadirán a la cuenta.
- **Descripción:** Motivo por el que se ejecuta la operación.

The screenshot shows a web-based application for managing transactions. The main title is "Gestión de Transacciones". Below it, a sub-section title is "Formulario para Crear Transacciones". The form itself has a blue header and a white body. It contains the following fields:

- Fecha: dd/mm/aaaa (with a calendar icon)
- Cuenta de Origen: Bankia (with a dropdown arrow)
- subClasificación: Apuestas ---> Juego (with a dropdown arrow)
- Gasto: 15.5 (text input field)
- Ingreso: 15.5 (text input field)
- Descripción: Motivo de la transacción (text input field)
- Marcar como favorita: (checkbox)
- Anadir: (green button)
- Cancelar: (red button)

Ilustración 50 Inserción de Importaciones



- El botón de editar de cada fila redirige al usuario a un formulario en el que se puede modificar la transacción de la fila seleccionada. Todos los campos se encuentran cumplimentados por defecto con el último valor establecido por el usuario. De este modo, para no modificar un campo simplemente debe no modificar su valor por defecto.

Gasto: 325.0

Ingreso: 325.0

Descripción: Piso de Alex Ye en Bilbao

Marcar como favorito

Editar

Cancelar

Ilustración 51 Edición de Transacciones

- El botón del filtro permite mostrar aquellas Transacciones que cumplen con los requisitos especificados.
 - Para una cuenta específica.
 - Para una subclasificación específica.
 - Para un rango de fechas preestablecido.

Cuenta: -----

SubClasificación: -----

Fecha Inicio: dd/mm/aaaa

Fecha Fin: dd/mm/aaaa

Aplicar Filtro

Restablecer Filtro

Ilustración 52 Filtro de Transacciones

Aportaciones

Las aportaciones son todas aquellas operaciones de compra o venta de activos financieros.

Fecha	Cuenta	I/V/N	Precio	Títulos	Tipo	Descripción	Añadir	Editar Filtros
06/04/2021	ClickTrade	AAPL	126,5	7,0	Compra		Editar	Eliminar
29/05/2020	ClickTrade	FB	225,0	10,0	Compra		Editar	Eliminar
29/11/2020	ClickTrade	FB	276,0	10,0	Compra		Editar	Eliminar
02/01/2022	ClickTrade	FB	313,0	15,0	Venta		Editar	Eliminar
03/06/2020	IndexaCapital	IE00BGCZ0B53	117,33	5,99	Compra		Editar	Eliminar
29/06/2020	ClickTrade	ITX	24,0	30,0	Compra		Editar	Eliminar
04/10/2021	ClickTrade	MSFT	282,37	5,0	Compra		Editar	Eliminar

Ilustración 48 Gestión de Aportaciones

Esta lista permite gestionar la visualización, inserción, modificación o eliminación de las aportaciones:

- Presionando el título de la cabecera se pueden ordenar alfabéticamente todas las entradas de la lista en función de la columna seleccionada.
- El botón eliminar de cada fila permite al usuario eliminar la transacción de la fila seleccionada junto con todas sus dependencias.
- El botón de añadir redirige al usuario a un formulario en el que se pueden insertar nuevas aportaciones.

En este formulario todos los campos son obligatorios:

- **Fecha:** Día, mes y año en el que se produce la operación.
- **Cuenta:** Cuenta sobre la que se aplica la compra o venta del activo financiero.
- **Activo:** Activo financiero sobre el que se opera.
- **Tipo de operación:** “Compra” o “Venta”.
- **Precio:** Precio de ejecución de la operación.
- **Títulos:** Número de participaciones negociadas.
- **Cambio:** Tipo de cambio entre la divisa de la cuenta y la divisa del activo.
- **Descripción:** Motivo por el que se ejecuta la operación.

Gestión de Aportaciones

Formulario para Crear Aportaciones

Fecha: dd/mm/aaaa

Cuenta: Bankia

Activo comenzado: Apple

Tipo de comisión comisión: Venta

Precio: 155

Títulos: 15.5

Cambio Divisa: 1

Descripción: Motivo de la adquisición

Marcar como favorito

Ilustración 54 Inserción de Aportaciones

- El botón de editar de cada fila redirige al usuario a un formulario en el que se puede modificar la aportación de la fila seleccionada. Todos los campos se encuentran cumplimentados por defecto con el último valor establecido por el usuario. De este modo, para no modificar un campo simplemente debe no modificar su valor por defecto.

Gestión de Aportaciones

Formulario para Editar la aportación del 2020-06-03 de IndexaCapital, IEOOB03HD191

Precio: 26.9

Títulos: 37.96

Cambio Divisa: 1.0

Descripción: Motivo de la adquisición

Marcar como favorito

Ilustración 55 Edición de Aportaciones



- El botón del filtro permite mostrar aquellas Aportaciones que cumplen con los requisitos especificados.
 - Para una cuenta o activo específicos.
 - Para un rango de fechas pre establecido.

The form is titled "Filtro" (Filter) at the top. Below it is a section titled "Filtrar Aportaciones" (Filter Contributions). It contains several input fields: "Cuenta:" (Account:), "ISIN:" (ISIN:), "Tipo de Operación:" (Operation Type:), "Fecha Inicio:" (Start Date: dd/mm/aaaa), and "Fecha Fin:" (End Date: dd/mm/aaaa). At the bottom of this section is a red-bordered button labeled "Aplicar Filtro" (Apply Filter). Below this section is another red-bordered button labeled "Restablecer Filtro" (Reset Filter).

Ilustración 56 Filtro de Transacciones

Comisiones

Las comisiones son todas aquellas operaciones que generan gastos o ingresos debido a la conservación de los activos financieros.

Se incluyen en esta sección todas las operaciones del tipo “Comision”, “Interes”, “Dividendo” o “Impuesto”

The table is titled "Gestión de Comisiones" (Commission Management). It has columns for Fecha (Date), Cuenta (Account), ISIN, Gasto (Expense), Ingreso (Income), Tipo (Type), Descripción (Description), and two buttons: "Añadir" (Add) and "Editar Filtros" (Edit Filters). The table lists five transactions:

Fecha	Cuenta	ISIN	Gasto	Ingreso	Tipo	Descripción	Añadir	Editar Filtros
13/06/2016	Bankia	*	--	5,0	Interes	Abono de Intereses	Editar	Eliminar
18/06/2016	Bankia	*	10,0	--	Comisión	Cargos de cuota tarjeta	Editar	Eliminar
26/06/2016	Bankia	*	--	5,0	Interes	Abono de Intereses	Editar	Eliminar
13/07/2016	Bankia	*	--	0,45	Interes	Interes no existe	Editar	Eliminar
12/08/2016	Bankia	*	0,13	--	Comisión	Debería colar	Editar	Eliminar

Ilustración 57 Gestión de Transacciones



Esta lista permite gestionar la visualización, inserción, modificación o eliminación de las comisiones:

- Presionando el título de la cabecera se pueden ordenar alfabéticamente todas las entradas de la lista en función de la columna seleccionada.
- El botón eliminar de cada fila permite al usuario eliminar la transacción de la fila seleccionada junto con todas sus dependencias.
- El botón de añadir redirige al usuario a un formulario en el que se pueden insertar nuevas transacciones.

En este formulario todos los campos son obligatorios:

- **Fecha:** Día, mes y año en el que se produce la operación.
- **Cuenta:** Cuenta sobre la que se aplica el gasto o ingreso.
- **Producto:** Producto sobre la que se aplica el gasto o ingreso.
- **Gasto:** Cantidad de unidades monetarias que se sustraerán.
- **Ingreso:** Cantidad de unidades monetarias que se añadirán a la cuenta.
- **Descripción:** Motivo por el que se ejecuta la operación.

Gestión de Comisiones

Formulario para Crear Comisiones

Fecha: dd/mm/aaaa

Cuenta que cobra la comisión: Bankia

Producto sobre el que se cobra la comisión: La comisión es de la cuenta

Tipo de comisión comisión: Interés

Gasto: 15.5

Ingreso: 15.5

Descripción: Motivo del Traspaso

Marcar como favorita

Anadir

Cancelar

Ilustración 58 Inserción de Comisiones



- El botón de editar de cada fila redirige al usuario a un formulario en el que se puede modificar la comisión de la fila seleccionada. Todos los campos se encuentran cumplimentados por defecto con el último valor establecido por el usuario. De este modo, para no modificar un campo simplemente debe no modificar su valor por defecto.

The screenshot shows a web-based form titled "Gestión de Comisiones". It is specifically for editing a commission entry from June 13, 2016, for Bankia. The form includes fields for "Gasto" (Expense) set to 0.0, "Ingreso" (Income) set to 5.0, and a description "Abono de Intereses". There is also a checkbox for "Marcar como favorita" (Mark as favorite). At the bottom are two buttons: a green "Editar" (Edit) button and a red "Cancelar" (Cancel) button.

Ilustración 59 Edición de Comisiones

- El botón del filtro permite mostrar aquellas Transacciones que cumplen con los requisitos especificados.
 - Para una cuenta o activo específicos.
 - Para un tipo de operación específica.
 - Para un rango de fechas preestablecido.

The screenshot shows a web-based form titled "Filtro" for filtering transactions. It has a header "Filtrar Comisiones". The form contains several dropdown menus and input fields:

- Cuenta: A dropdown menu showing "-----".
- ISIN: A dropdown menu showing "-----".
- Tipo de Operación: A dropdown menu showing "-----".
- Fecha Inicio: An input field with placeholder "dd/mm/aaaa" and a calendar icon.
- Fecha Fin: An input field with placeholder "dd/mm/aaaa" and a calendar icon.

At the bottom are two buttons: a green "Aplicar Filtro" (Apply Filter) button and a red "Restablecer Filtro" (Reset Filter) button.

Ilustración 60 Filtro de Transacciones



Traspasos

Los Traspasos son todas aquellas operaciones que no producen un cambio en el patrimonio del usuario debido a que se consideran ajustes contables entre las propias cuentas del usuario.

Gestión de Traspasos						
Fecha	Origen	Destino	Precio	Descripción	Añadir	Editar Filtros
28/06/2020	Bankia	ClickTrade	651,0	Real incorrecto	<button>Editar</button>	<button>Eliminar</button>
01/04/2021	Bankia	ClickTrade	800,0	Compra Apple	<button>Editar</button>	<button>Eliminar</button>
17/03/2020	Bankia	DeGiro	400,0	Aprovechar rebajas y probar Degiro	<button>Editar</button>	<button>Eliminar</button>
24/12/2020	Bankia	MyInvestor	5.000,0	Colchon Parte 2	<button>Editar</button>	<button>Eliminar</button>
29/12/2020	Bankia	MyInvestor	3.000,0	Colchon Parte 3	<button>Editar</button>	<button>Eliminar</button>
01/01/2016	Efectivo	Bankia	6.485,0	Calculo del dinero Inicial	<button>Editar</button>	<button>Eliminar</button>
01/01/2016	Efectivo	ClickTrade	5.000,0	Calculo del dinero Inicial	<button>Editar</button>	<button>Eliminar</button>
01/01/2016	Efectivo	DeGiro	5.000,0	Calculo del dinero Inicial	<button>Editar</button>	<button>Eliminar</button>
01/01/2016	Efectivo	IndexaCapital	5.000,0	Calculo del dinero Inicial	<button>Editar</button>	<button>Eliminar</button>

Ilustración 61 Gestión de Transacciones

Esta lista permite gestionar la visualización, inserción, modificación o eliminación de los traspasos:

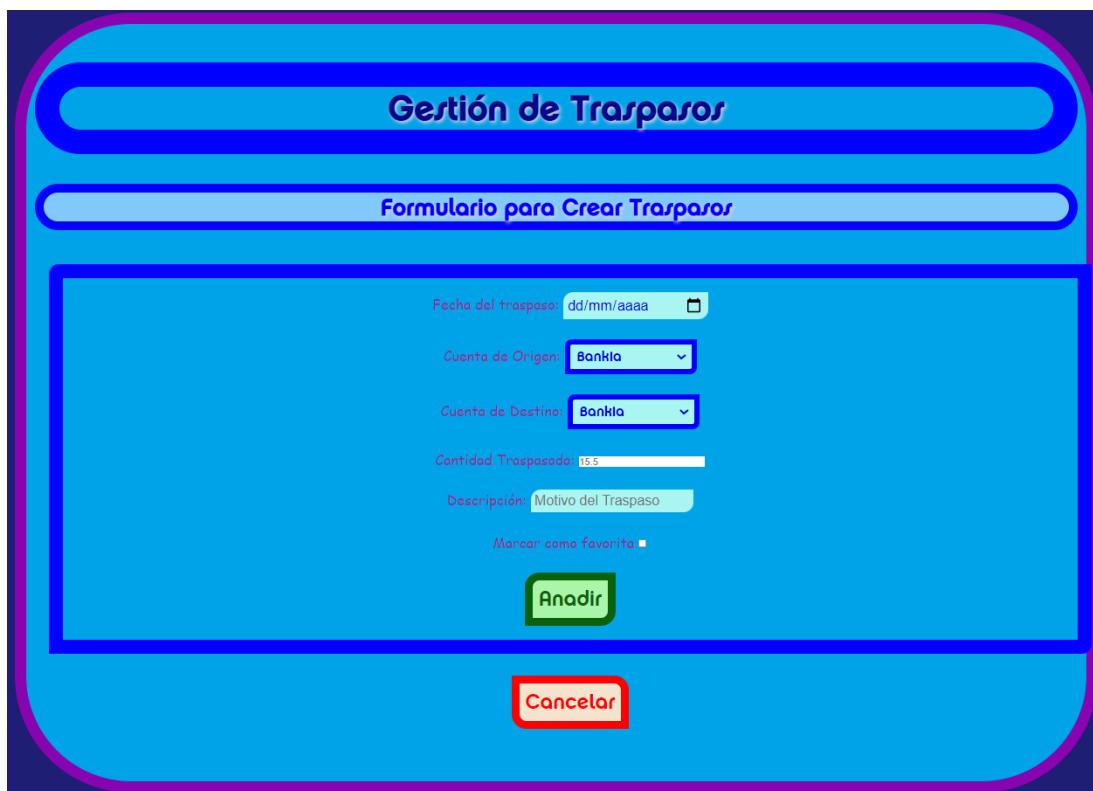
- Presionando el título de la cabecera se pueden ordenar alfabéticamente todas las entradas de la lista en función de la columna seleccionada.
- El botón eliminar de cada fila permite al usuario eliminar la transacción de la fila seleccionada junto con todas sus dependencias.



- El botón de añadir redirige al usuario a un formulario en el que se pueden insertar nuevos traspasos.

En este formulario todos los campos son obligatorios:

- **Fecha:** Día, mes y año en el que se produce la operación.
- **Cuenta de Origen:** Cuenta que inicia la operación.
- **Cuenta de Destino:** Cuenta que recibe el traspaso.
- **Cantidad:** Cantidad de unidades monetarias que se transfieren.
- **Descripción:** Motivo por el que se ejecuta la operación.



The screenshot shows a web-based form titled "Formulario para Crear Traspasos". The form fields are as follows:

- Fecha del traspaso: dd/mm/aaaa (input field)
- Cuenta de Origen: Bankia (dropdown menu)
- Cuenta de Destino: Bankia (dropdown menu)
- Cantidad Traspasada: 15.5 (input field)
- Descripción: Motivo del Traspaso (input field)
- Marcar como favorito:
- Anadir: A green button with white text.
- Cancelar: A red button with white text.

Ilustración 62 Inserción de Traspasos

- El botón de editar de cada fila redirige al usuario a un formulario en el que se puede modificar el traspaso de la fila seleccionada.



The screenshot shows a web-based form titled "Formulario para Editar el traspaso del 2020-06-28 de Bankia a ClickTrade". The form fields are as follows:

- Cantidad Traspasada: 551.0 (input field)
- Descripción: Real incorrecto (input field)
- Marcar como favorito:
- Editar: A green button with white text.
- Cancelar: A red button with white text.

Ilustración 63 Edición de Comisiones



- El botón del filtro permite mostrar aquellas Transacciones que cumplen con los requisitos especificados.
 - o Para una cuenta de destino y origen específicos.
 - o Para un rango de fechas pre establecido.



Ilustración 64 Filtro de Transacciones

Importaciones

En la sección de importaciones los usuarios podrán gestionar su información financiera de forma masiva.



Ilustración 65 Sección de Importaciones

Importaciones

la subsección de importaciones permite a un usuario identificado importar todos sus datos económicos mediante la importación de un fichero externo diseñado para tal propósito.



Ilustración 66 Importación de datos

Para iniciar el proceso de importación se selecciona la plantilla correctamente cumplimentada y se pulsa el botón de importar datos. En caso de que los datos del fichero que se está importando deban sustituir la totalidad de los datos previamente almacenados, se marcará la casilla indicada para tal efecto.

Para que la importación de los datos se complete de forma satisfactoria es necesario seguir el siguiente patrón:

- Los ficheros tendrán formato CSV separando cada elemento por punto y coma “;”
- No se utilizará el símbolo de punto y coma “;” como parte de un dato en ningún campo.
- No se modificará el nombre de la plantilla proporcionada.
- Se respetará el número de columnas.
- No se modificarán las cabeceras de cada columna.
- Se respetará el tipo de dato asignado para cada columna.

En caso de que no se respete alguna de las condiciones:

- No se completará la operación de la columna incorrecta.
- Se mostrará un mensaje de error descriptivo indicando las columnas que no han podido ser procesadas.



A continuación, se especifica el formato que debe tener cada plantilla.

Aportaciones.csv

	A	B	C	D	E	F	G	H	I	J
1	Dia	Mes	Año	Cuenta	ISIN	Operación	Precio	Titulos	Cambio	Descripcion
2	25	9	2019	IndexaCapital	IE00B03HD191	compra	26,62	22,68	1	
3	7	1	2020	IndexaCapital	IE00B03HD191	compra	28,47	15,28	1	
4	3	6	2020	IndexaCapital	IE00B03HD191	venta	26,9	37,96	1	
5	3	6	2020	IndexaCapital	IE00BFPM9N11	compra	190,08	5,37	1	

Aportaciones.csv	
1	Dia;Mes;Año;Cuenta;ISIN;Operación;Precio;Titulos;Cambio;Descripción
2	25;9;2019;IndexaCapital;IE00B03HD191;compra;26,62;22,68;1;
3	7;1;2020;IndexaCapital;IE00B03HD191;compra;28,47;15,28;1;
4	3;6;2020;IndexaCapital;IE00B03HD191;venta;26,9;37,96;1;
5	3;6;2020;IndexaCapital;IE00BFPM9N11;compra;190,08;5,37;1;

Ilustración 67 Formato de la plantilla para las Aportaciones

- Las columnas de Día, Mes y Año, contendrán la fecha de la aportación.
 - o Serán de tipo numérico. Por ejemplo, no serán válidos símbolos o letras.
 - o La fecha debe existir. Por ejemplo, no será válida una fecha con el mes trece.
- La columna de Cuenta contendrá el nombre de una cuenta previamente registrada por el usuario.
 - o Se permite cualquier carácter ([excepto el punto y coma](#)).
 - o La cuenta debe estar registrada por el usuario.
 - o El nombre de la cuenta debe ser exacto, respetando las mayúsculas y los espacios.
- La columna de ISIN contendrá el identificador de un producto previamente registrada por el usuario.
 - o Se permite cualquier carácter ([excepto el punto y coma](#)).
 - o Recuerde que se trata de un identificador oficial asignado al producto. Este identificador será utilizado por el Scraper para obtener de internet la información financiera asociada a dicho producto.
- La columna de Operación contendrá el tipo de operación. “compra” o “venta”.
- Las columnas de Precio, Títulos y Cambio, contendrán el precio de ejecución, el número de títulos negociados y el tipo de cambio al que se ejecutó la operación.
 - o Serán números reales utilizando la coma para separar la parte decimal.
 - o Se permite utilizar el punto para separar los miles ([este separador será eliminado en el post procesado](#)).
 - o Si la operación es en la misma moneda el tipo de cambio será uno.
- La columna de Descripción contendrá una descripción para la operación.
 - o Admite cualquier carácter excepto el punto y coma.

Algunas filas no han podido ser insertadas. A continuación se detallan las posibles causas.

* La fecha 2019/9/2019 no existe

* La fecha 2020/2019/7 no existe

* No tienes registrada la cuenta AAA

* No tienes registrado el activo IE00BFPM9N12

* No tienes registrado el activo AAA

* El tipo de operación debe ser 'Compra' o 'Venta'

* El precio debe ser un número Real. Comprueba que no has añadido ningún símbolo diferente al punto o la coma para separar la parte decimal

* El número de títulos debe ser un número Real. Comprueba que no has añadido ningún símbolo diferente al punto o la coma para separar la parte decimal

Ilustración 68 Posibles mensajes de error al importar Aportaciones



Comisiones.csv

A	B	C	D	E	F	G	H	I	
1	Dia	Mes	Ano	Cuenta	Tipo	ISIN	Gasto	Ingreso	Descripcion
2	13	6	2016	Bankia	Interes	*	0	5	Abono de Intereses
3	18	6	2016	Bankia	Comision	*	10	0	Cargos de cuota targeta
4	26	6	2016	Bankia	Interes	*	0	5	Abono de Intereses
5	13	7	2016	Bankia	Interes	*	0	5	Abono de Intereses

Comisiones.csv	
1	Dia ; Mes ; Ano ;Cuenta;Tipo; ISIN ;Gasto;Ingreso;Descripcion
2	13;6;2016;Bankia;Interes;*;0;5;Abono de Intereses
3	18;6;2016;Bankia;Comision;*;10;0;Cargos de cuota targeta
4	26;6;2016;Bankia;Interes;*;0;5;Abono de Intereses
5	13;7;2016;Bankia;Interes;*;0;5;Abono de Intereses

Ilustración 69 Formato de la plantilla para las Comisiones

- Las columnas de Día, Mes y Año, contendrán la fecha de la aportación.
 - o Serán de tipo numérico. Por ejemplo, no serán validos símbolos o letras.
 - o La fecha debe existir. Por ejemplo, no será válida una fecha con el mes trece.
- La columna de Cuenta contendrá el nombre de una cuenta previamente registrada por el usuario.
 - o Se permite cualquier carácter ([excepto el punto y coma](#)).
 - o La cuenta debe estar registrado por el usuario.
 - o El nombre de la cuenta debe ser exacto, respetando las mayúsculas y los espacios.
- La columna de Tipo contendrá el tipo de operación. “Interés”, “Comisión” “Dividendo” o “Impuesto”.
- La columna de ISIN contendrá el identificador de un producto previamente registrada por el usuario.
 - o Se permite cualquier carácter ([excepto el punto y coma](#)).
 - o Recuerde que se trata de un identificador oficial asignado al producto. Este identificador será utilizado por el Scraper para obtener de internet la información financiera asociada a dicho producto.
 - o Si la operación no se asigna a un producto concreto sino a la cuenta completa se utilizará el “**”
- Las columnas de Gasto e Ingreso contendrán la cuantía de la comisión registrada.
 - o Serán números reales utilizando la coma para separar la parte decimal.
 - o Se permite utilizar el punto para separar los miles ([este separador será eliminado en el post procesado](#)).
 - o Cualquier operación de este tipo se considerará siempre en la misma divisa que la cuenta.
- La columna de Descripción contendrá una descripción para la operación.
 - o Admite cualquier carácter excepto el punto y coma.

Algunas filas no han podido ser insertadas. A continuacion se detallan las posibles causas.
 * La fecha 2016/6/2016 no existe

* La fecha 2016/2016/18 no existe

* No tienes registrada la cuenta AAA

* Ya existe una operacion para * del 2016-10-04 en la cuenta Bankia

* El tipo de operacion debe ser 'Interes', 'Comisión', 'Impuesto' o 'Dividendo'

* Ya existe una operacion para * del 2018-06-24 en la cuenta Bankia

* Ya existe una operacion para * del 2019-10-04 en la cuenta IndexaCapital

* Ya existe una operacion para ITX del 2020-06-29 en la cuenta ClickTrade

* El Gasto y el Ingreso deben ser un numeros Reales. Comprueba que no has añadido ningun simbolo diferente al punto o la coma para separar la parte decimal

Ilustración 70 Posibles mensajes de error al importar Aportaciones

EUSKAL HERRIKO
UNIVERSITATEAUniversidad
del País Vasco

Gastos.csv

	A	B	C	D	E	F	G	H	I
1	Dia	Mes	Año	Cuenta	Clasificacion	Asunto	Gasto	Ingreso	Descripcion
2	29	4	2016	Bankia	Transporte	Autobus	0,7	0	Hospital Ribera - Estacion Renfe
3	29	4	2016	Bankia	Transporte	Tren	1,8	0	renfe alcira a carcaixent
4	27	4	2016	Bankia	Transporte	Tren	7,5	0	Carcaixent - Valencia norte - ida y vuelta Sin familia Num
5	26	4	2016	Bankia	Transporte	Autobus	1,5	0	Carcaixent - Hospital Ribera

Gastos.csv

```

1 Dia;Mes;Año;Cuenta;Clasificacion;Asunto;Gasto;Ingreso;Descripcion
2 29;4;2016;Bankia;Transporte;Autobus;0,7;0;Hospital Ribera - Estacion Renfe
3 29;4;2016;Bankia;Transporte;Tren;1,8;0;renfe alcira a carcaixent
4 27;4;2016;Bankia;Transporte;Tren;7,5;0;Carcaixent - Valencia norte - ida y vuelta Sin familia Num
5 26;4;2016;Bankia;Transporte;Autobus;1,5;0;Carcaixent - Hospital Ribera

```

Ilustración 71 Formato de la plantilla para los Gastos

- Las columnas de Día, Mes y Año, contendrán la fecha de la aportación.
 - o Serán de tipo numérico. Por ejemplo, no serán válidos símbolos o letras.
 - o La fecha debe existir. Por ejemplo, no será válida una fecha con el mes trece.
- La columna de Cuenta contendrá el nombre de una cuenta previamente registrada por el usuario.
 - o Se permite cualquier carácter ([excepto el punto y coma](#)).
 - o La cuenta debe estar registrada por el usuario.
 - o El nombre de la cuenta debe ser exacto, respetando las mayúsculas y los espacios.
- La columna de Clasificación contendrá el nombre de una clasificación previamente registrada por el usuario.
 - o Se permite cualquier carácter ([excepto el punto y coma](#)).
 - o La clasificación debe estar registrada por el usuario.
 - o El nombre de la clasificación debe ser exacto, respetando las mayúsculas y los espacios.
- La columna de Asunto contendrá el nombre de una subclasificación previamente registrada por el usuario para la clasificación indicada.
 - o Se permite cualquier carácter ([excepto el punto y coma](#)).
 - o La subclasificación debe estar registrada por el usuario para la clasificación indicada.
 - o El nombre de la subclasificación debe ser exacto, respetando las mayúsculas y los espacios.
- Las columnas de Gasto e Ingreso contendrán la cuantía de la operación registrada.
 - o Serán números reales utilizando la coma para separar la parte decimal.
 - o Se permite utilizar el punto para separar los miles ([este separador será eliminado en el post procesado](#)).
 - o Cualquier operación de este tipo se considerará siempre en la misma divisa que la cuenta.
- La columna de Descripción contendrá una descripción para la operación.
 - o Admite cualquier carácter excepto el punto y coma.

Algunas filas no han podido ser insertadas. A continuacion se detallan las posibles causas.

* La fecha 2016/4/2016 no existe

* La fecha 2016/2016/29 no existe

* No tienes registrada la cuenta aaaa

* No tienes registrada la clasificacion aaaa - Autobus, añadela antes de insertar esta transaccion

* No tienes registrada la clasificacion Transporte - aaaa, añadela antes de insertar esta transaccion

* El Gasto y el Ingreso deben ser un numeros Reales. Comprueba que no has añadido ningun simbolo diferente al punto o la coma para separar la parte decimal

Ilustración 72 Posibles mensajes de error al importar Aportaciones



Transpasos.csv

	A	B	C	D	E	F	G
1	Dia	Mes	Ano	Origen	Destino	Precio	Descripcion
2	1	1	2016	Efectivo	Bankia	6.485,00	Calculo del dinero Inicial
3	1	1	2016	Efectivo	Indexa Capital	5.000,00	Calculo del dinero Inicial
4	1	1	2016	Efectivo	ClickTrade	5.000,00	Calculo del dinero Inicial
5	1	1	2016	Efectivo	DeGiro	5.000,00	Calculo del dinero Inicial

Transpasos.csv X

```
1 Dia ; Mes ; Ano ;Origen;Destino; Precio;Descripcion
2 1;1;2016;Efectivo;Bankia;6.485,00;Calculo del dinero Inicial
3 1;1;2016;Efectivo;Indexa Capital;5.000,00;Calculo del dinero Inicial
4 1;1;2016;Efectivo;ClickTrade;5.000,00;Calculo del dinero Inicial
5 1;1;2016;Efectivo;DeGiro;5.000,00;Calculo del dinero Inicial
```

Ilustración 49 Formato de la plantilla para los Gastos

- Las columnas de Día, Mes y Año, contendrán la fecha de la aportación.
 - o Serán de tipo numérico. Por ejemplo, no serán validos símbolos o letras.
 - o La fecha debe existir. Por ejemplo, no será válida una fecha con el mes trece.
- Las columnas de Origen y Destino contendrán el nombre de una cuenta previamente registrada por el usuario.
 - o Se permite cualquier carácter ([excepto el punto y coma](#)).
 - o Las cuentas deben estar registradas por el usuario.
 - o Los nombres de las cuentas deben ser exactos, respetando las mayúsculas y los espacios.
- La columna de Precio contendrá la cuantía de la operación registrada.
 - o Serán números reales utilizando la coma para separar la parte decimal.
 - o Se permite utilizar el punto para separar los miles ([este separador será eliminado en el post procesado](#)).
 - o Cualquier operación de este tipo se considerará siempre en la misma divisa que la cuenta.
- La columna de Descripción contendrá una descripción para la operación.
 - o Admite cualquier carácter excepto el punto y coma.

Algunas filas no han podido ser insertadas. A continuacion se detallan las posibles causas.

* La fecha 2016/1/2016 no existe

* La fecha 2019/20/18 no existe

* No tienes registrada la cuenta aaa

* Ya existe un traspaso desde la cuenta Bankia hasta la cuenta IndexaCapital del 2020-01-02

Ilustración 74 Posibles mensajes de error al importar Aportaciones

Exportaciones

Permite a un usuario identificado exportar todos sus datos económicos a un fichero externo con el formato y distribución adecuados para que sea aceptado por el sistema de importación.

Esta sección no ha sido implementada.



Ilustración 75 Exportación de datos

Análisis de Datos

Se trata de un conjunto de secciones que permiten a los usuarios analizar su información financiera mediante la visualización de tablas y gráficas que muestran los datos previamente registrados.

Balance

Esta sección permite a los usuarios identificados visualizar un resumen de su patrimonio.



Ilustración 50 Análisis del Balance

Por defecto se muestran todos los datos disponibles. No obstante, es posible acotar los datos mediante el filtro, seleccionando el periodo temporal.

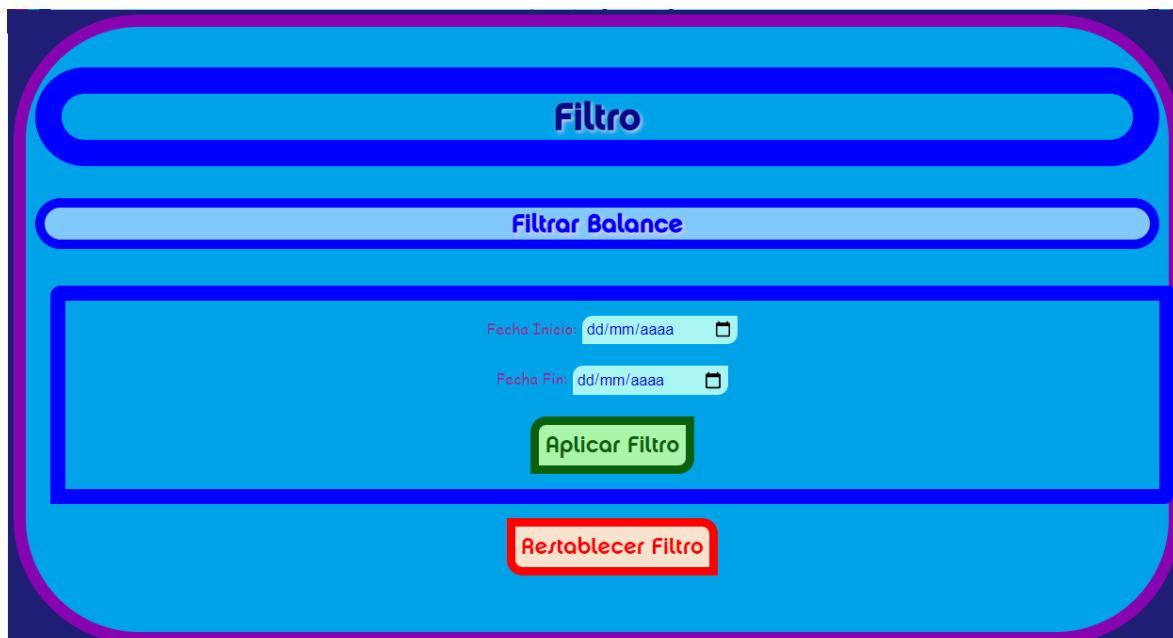


Ilustración 51 Filtro del Balance



La tabla del capital total puede visualizar el desglose del capital para cada cuenta registrada por el usuario en el periodo seleccionado. Mostrando las cantidades totales registradas para los gastos, los ingresos, los traspasos, las comisiones y los rendimientos del capital.

(En esta tabla los números negativos se representan con el color rojo y los positivos mediante el color verde)

Capital Total						
Cuenta	Ingresos	Gastos	Traspasos	Comisiones	Beneficios	Capital
Bankia	37.887,21	25.875,1	9.606,0	128,46	--	2.277,65
ClickTrade	--	--	6.351,0	67,3	1.223,42	7.507,12
DeGiro	--	--	5.400,0	--	3.904,94	9.304,94
Efectivo	20,0	22,0	21.485,0	--	--	21.487,0
IndexaCapital	--	--	9.340,0	17,96	1.616,3	10.374,26
MyInvestor	--	--	10.000,0	30,0	--	10.030,0
Capital	37.907,21	25.897,1		147,8	6.144,67	18.006,98

Ilustración 52 Balance Patrimonial

La tabla del resumen de activos desglosa los rendimientos del capital para cada activo registrado en cualquier cuenta. Mostrando el número total de títulos en propiedad en la fecha de fin del periodo seleccionado, el precio medio de adquisición de dichos títulos, el precio actual, el valor total aportado por el usuario y el valor actual, las comisiones ocasionadas por dicho activo, los beneficios totales generados, la rentabilidad total del periodo seleccionado y la rentabilidad anualizada del periodo seleccionado.

Resumen de Activos										
ISIN	Descripción	Títulos	Precio Medio	Precio Actual	Valor Aportado	Valor Actual	Comisiones	Beneficio	Rentabilidad Total	Rentabilidad Anual
IE00B4L5Y983	iShares Core MSCI World UCITS ETF USD	104,0	39,56	77,3	4.114,0	8.039,31	--	3.925,31	95,41	37,9
IE00B4L5YC18	iShares Core MSCI EM UCITS ETF USD	13,0	36,29	34,72	471,74	451,38	--	20,37	4,32	1,71
IE00BFPM9N11	Vanguard Global Stocks Index Fund Acumulation Institucional	18,05	209,26	275,61	3.777,07	4.974,76	--	1.197,69	31,71	12,39
IE00BGCZ0B53	Vanguard Global Bonds Index Fund Acumulation Institucional	21,09	117,61	109,36	2.480,49	2.306,4	--	174,09	7,02	2,79
FB	Facebook	5,0	50,73	205,81	253,65	1.029,04	--	775,39	395,69	121,41
ITX	Industria de Diseño Textil	30,0	24,0	19,78	720,0	593,4	25,0	151,6	21,06	8,36
AAPL	Apple	7,0	107,52	160,57	752,67	1.124,01	--	371,33	49,34	19,59
MSFT	Microsoft	5,0	242,84	283,5	1.214,19	1.417,49	--	203,3	16,74	6,65
Total	Operaciones Cerradas						25,0	6.119,67	44,37	17,62

Ilustración 53 Resumen de Activos

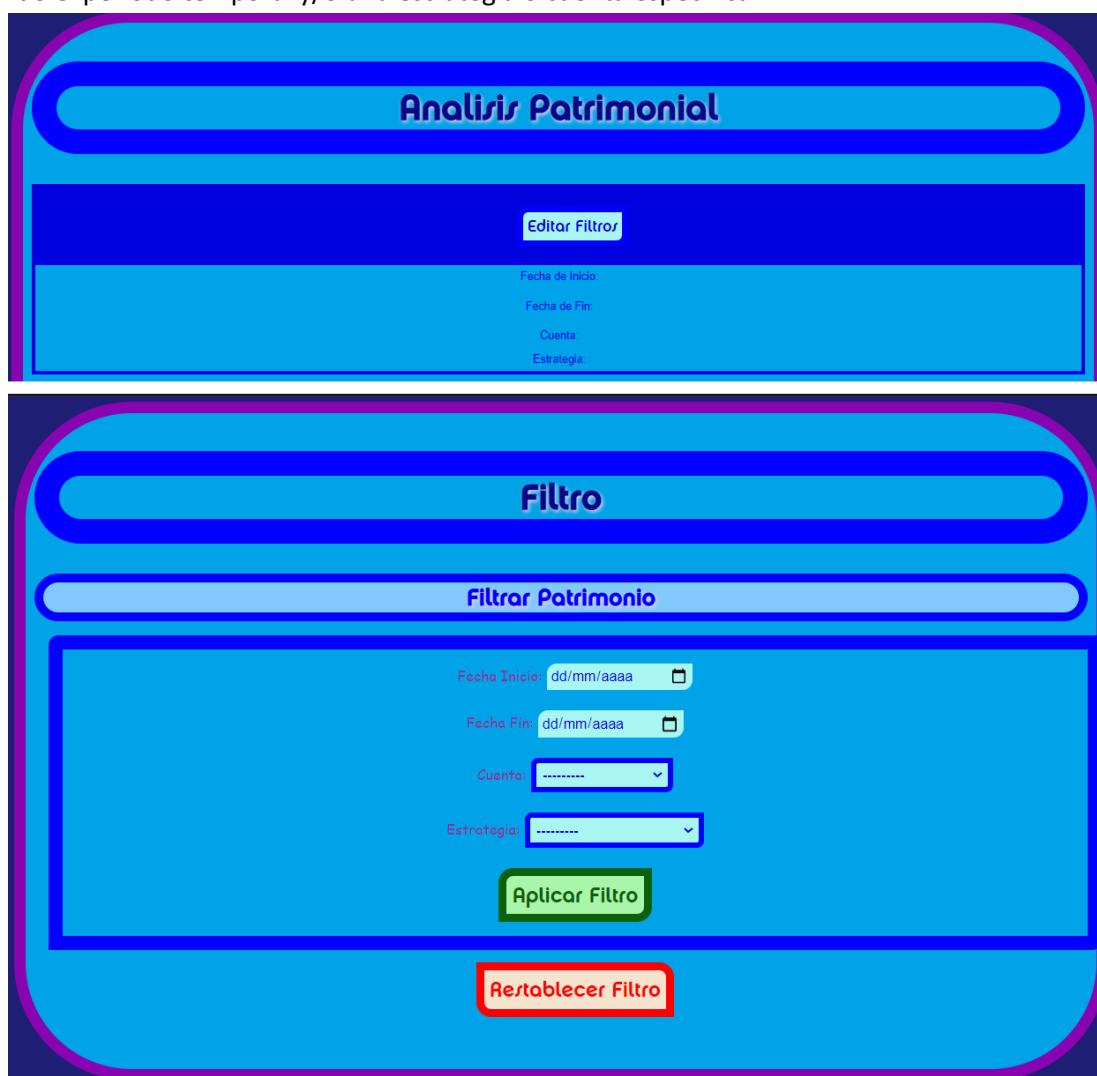
Patrimonio

Esta sección permite a los usuarios identificados visualizar un resumen de su patrimonio.



Ilustración 54 Análisis del Patrimonio

Por defecto se muestran todos los datos disponibles. No obstante, es posible acotar los datos mediante el filtro, seleccionando el periodo temporal y/o una estrategia o cuenta específica



The illustration shows a detailed view of the "Filtro" (Filter) screen. It has a dark blue header with the title "Analisis Patrimonial" and a "Editar Filtros" button. Below this is a dark blue section with fields for "Fecha de Inicio", "Fecha de Fin", "Cuenta", and "Estrategia". The main body of the screen is titled "Filtrar Patrimonio" and contains four input fields: "Fecha Inicio" (dd/mm/aaaa), "Fecha Fin" (dd/mm/aaaa), "Cuenta" (dropdown menu), and "Estrategia" (dropdown menu). Below these is a green "Aplicar Filtro" button and a red "Restablecer Filtro" button at the bottom.

Ilustración 55 Filtro del Patrimonio

La gráfica de la distribución de estrategias desglosa todos los activos en función de las estrategias seleccionadas por el usuario, permitiendo medir el peso de cada una sobre el total.

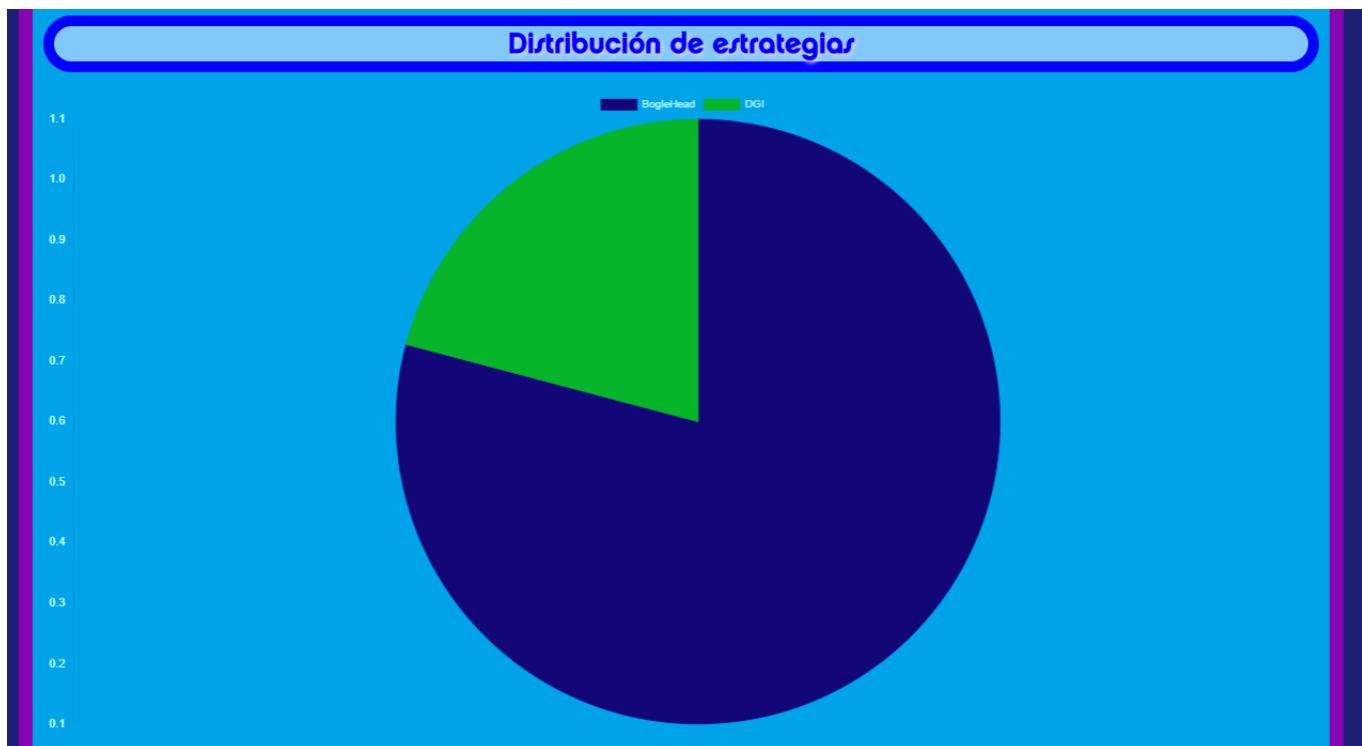


Ilustración 82 Gráfica de Distribución de Estrategias

La gráfica de distribución de activos desglosa todos los activos en función de su peso sobre el total.

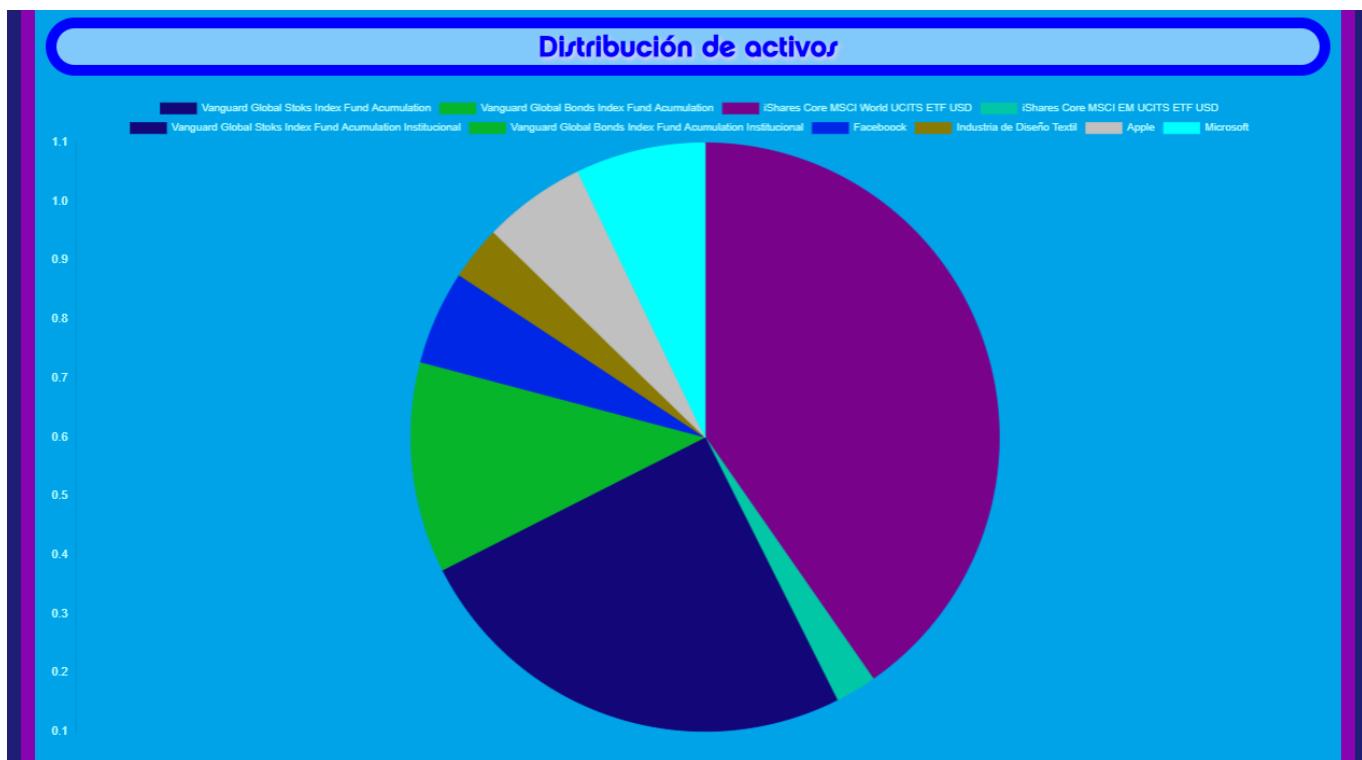


Ilustración 83 Gráfica de Distribución de Activos

Las gráficas de distribución de Allocations descomponen todos los activos en función de los distintos Allocations que ha definido el usuario permitiendo medir el peso de cada una sobre el total.

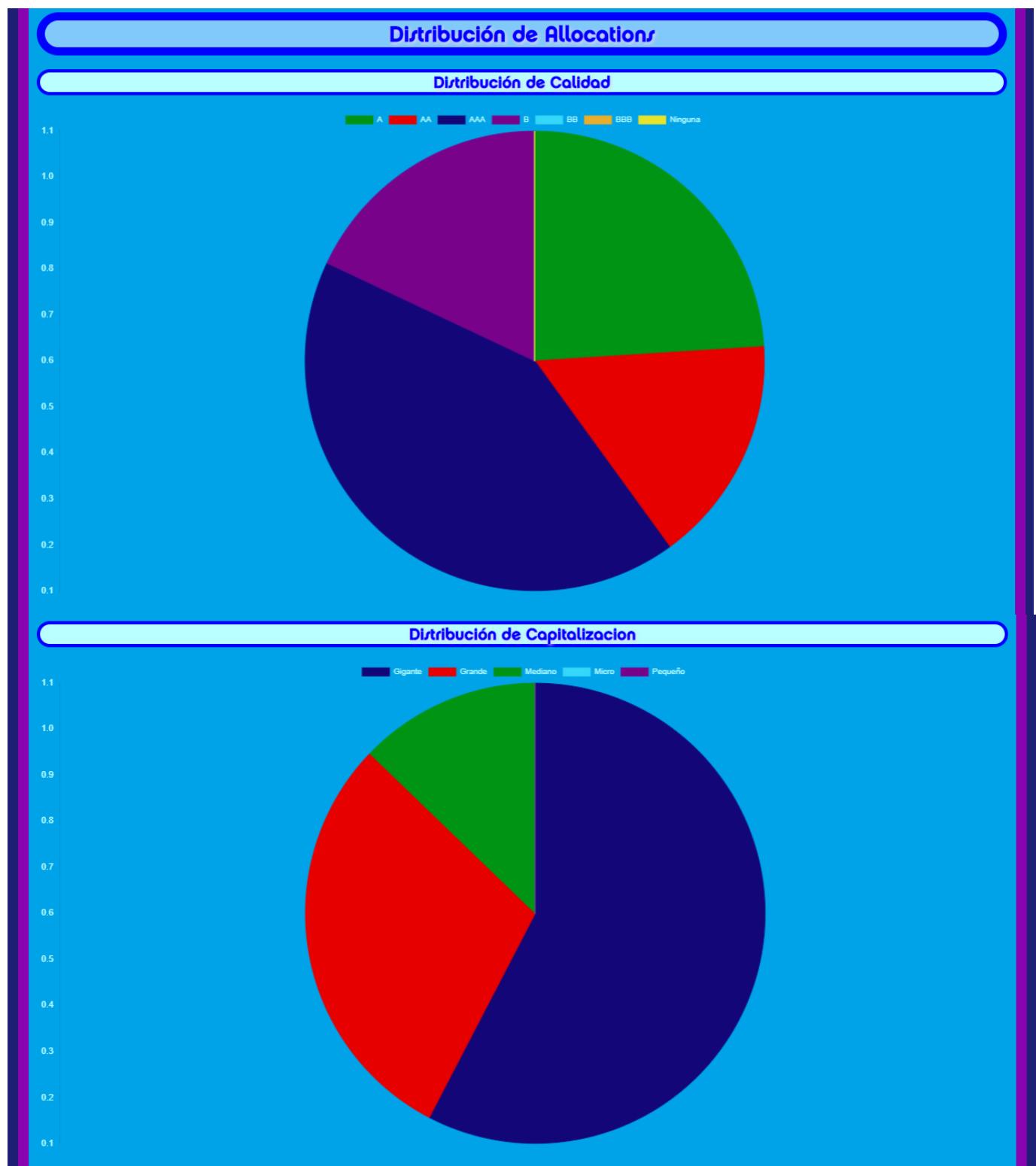


Ilustración 56 Gráfica de Distribución de Allocations

Las gráficas de distribución de Gastos e Ingresos descomponen todas las transacciones en función de las distintas Categorías que ha definido el usuario permitiendo medir el peso de cada una sobre el total.

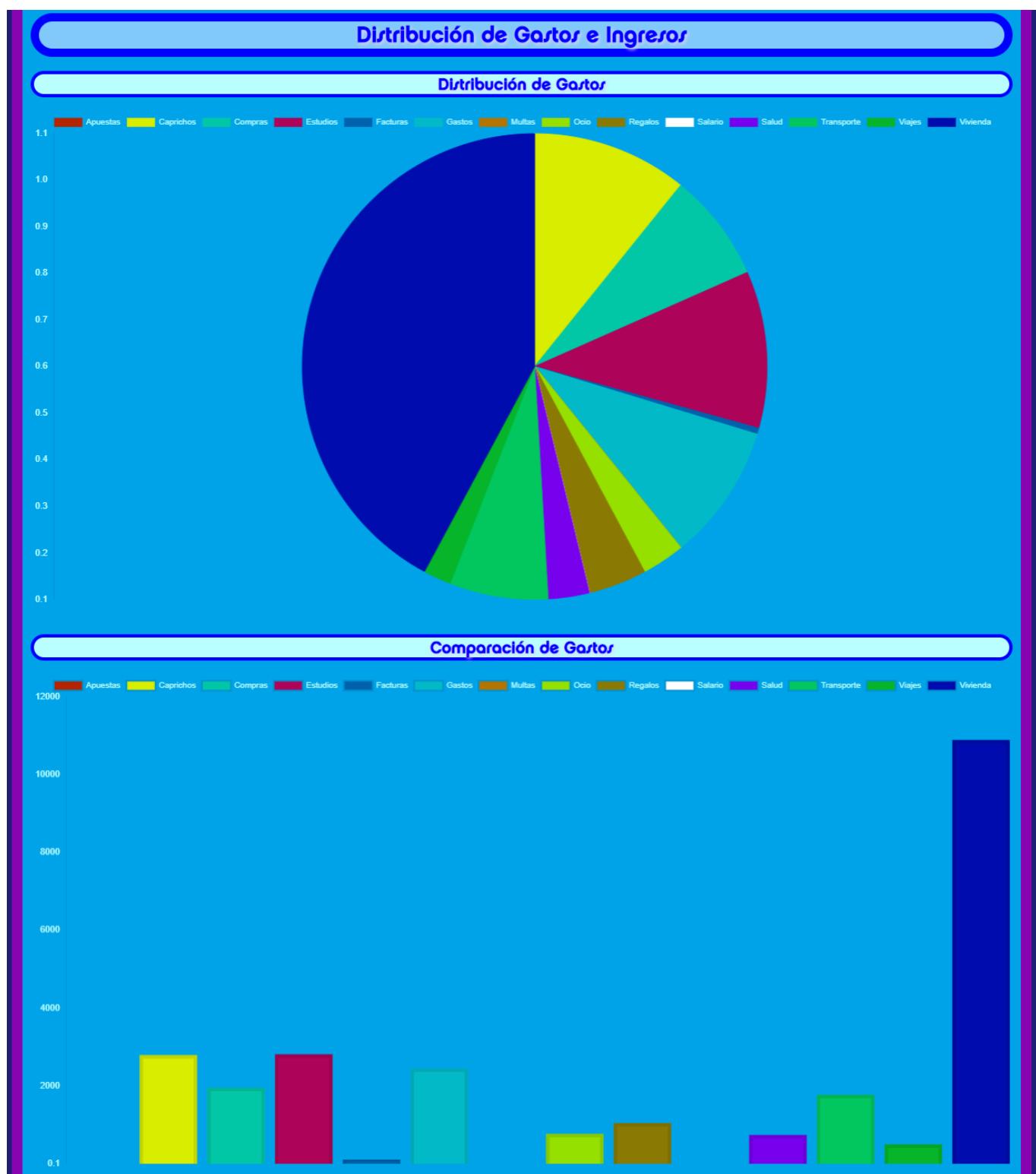


Ilustración 57 Gráfica de Distribución de Gastos

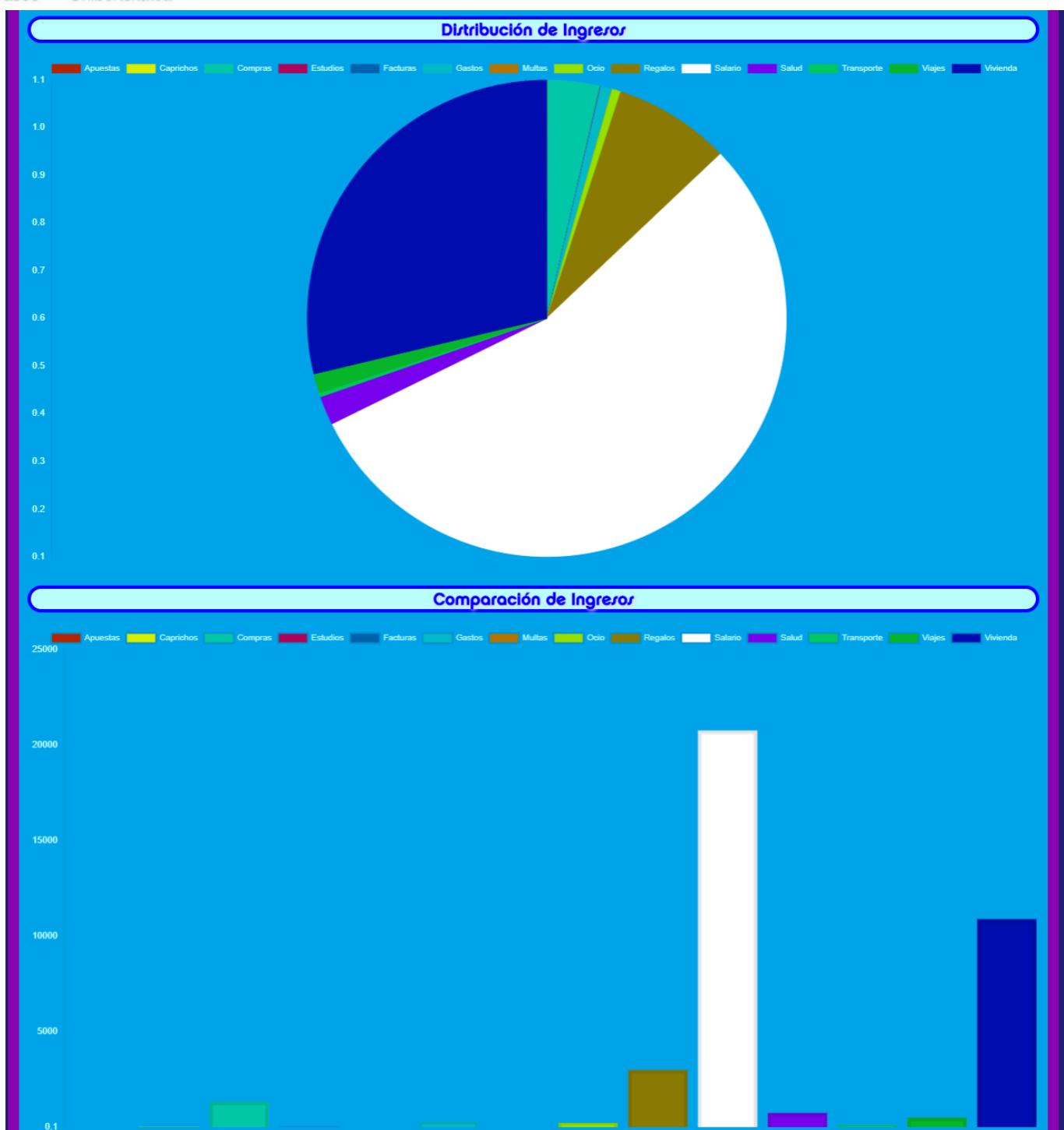


Ilustración 58 Gráfica de Distribución de Ingresos

La gráfica de comparación de ingresos con gastos permite comparar los gastos con los ingresos.

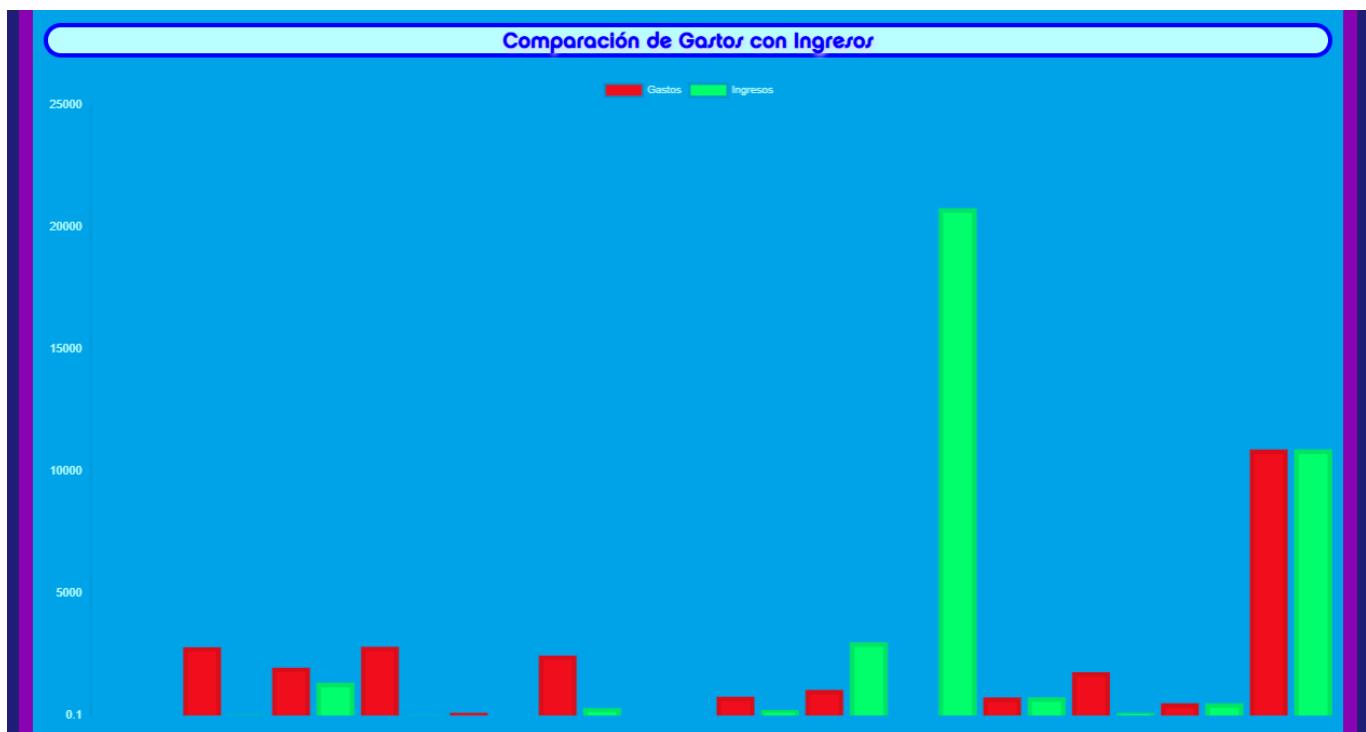


Ilustración 59 Gráfica de Comparación de Gastos e Ingresos

La tabla de Desglose de gastos e ingresos muestra todos los gastos e ingresos asignados a cada subcategoría ordenadas por categorías para el periodo seleccionado.

The table provides a detailed breakdown of expenses (Gasto) and income (Ingreso) across three main categories: Apuestas, Caprichos, and Compras. The data is further broken down by sub-classifications.

Apuestas		
SubClasificación	Gasto	Ingreso
Juego	—	—
Total	—	—

Caprichos		
SubClasificación	Gasto	Ingreso
G2A	94,83	—
GAME	49,95	—
HumbleBumble	169,04	—
IG	97,5	—
LolaMarket	25,8	—
Mojang	23,95	—
Steam	2.312,3	11,0
SuperCell	19,98	10,0
Total	2.793,35	21,0

Compras		
SubClasificación	Gasto	Ingreso
Móvil	564,77	560,77
Muebles	17,7	—
Ordenador	1.213,87	700,0
Ropa	158,9	77,4
Total	1.955,24	1.336,17

Ilustración 60 Tabla de Desglose de Gastos e Ingresos

Flujo de Caja

En la sección de Flujo de caja se muestran la variación del patrimonio, así como la evolución de los gastos e ingresos a lo largo del tiempo.



Ilustración 61 Análisis del Flujo de Caja

Esta sección no ha sido implementada.



Ilustración 62 Gestión de accesos



erian ta zabal zazu

Universidad
del País Vasco Euskal Herriko
 Unibertsitatea

BILBOKO
INGENIARITZA ESKOLA

ESCUELA
DE INGENIERÍA DE BILBAO

Gestión de Finanzas Personales

**Grado en Ingería Informática de
Gestión y Sistemas de Información**

TRABAJO FIN DE GRADO **Anexo 3 – Documentación del Código**



Universidad
del País Vasco
Euskal Herriko
Unibertsitatea

BILBOKO
INGENIARITZA
ESKOLA
ESCUELA
DE INGENIERÍA
DE BILBAO

Alumno: Cuesta Alario, David

Director: López Cuadrado, Javier

Curso: 2020 – 2021

Fecha: 2020-10-19

Índice de contenido

DIAGRAMA DE CLASES	1
SESIÓN.....	2
SECCIONES.....	13
SECCIÓN.....	17
<i>Inicio Sesión</i>	20
<i>Cerrar Sesión.....</i>	21
<i>Registro.....</i>	22
<i>Perfil.....</i>	24
<i>Admisión</i>	26
<i>Administración.....</i>	28
<i>Accesos</i>	30
<i>Gestión.....</i>	31
<i>Análisis.....</i>	34
<i>Importaciones</i>	35
FORMULARIO	36
<i>Activos.....</i>	38
<i>Cuentas</i>	38
<i>Estrategias</i>	39
<i>Categorías.....</i>	39
<i>Productos.....</i>	40
<i>Categorización</i>	40
<i>Allocation.....</i>	41
<i>Sub-Allocation.....</i>	41
<i>Clasificación</i>	42
<i>Sub-Clasificación</i>	42
<i>Distribución.....</i>	43
<i>Transacciones</i>	43
<i>Aportaciones</i>	44
<i>Comisiones</i>	44
<i>Traspasos</i>	45
GRAFICAS	46
<i>Balance</i>	46
<i>Gastos</i>	47
<i>Patrimonio</i>	47
<i>Evolución</i>	47
FILTRO	48
LIBRERÍA SQL.....	49
LIBRERÍA HTML	52
LIBRERÍA FICHEROS.....	56
LIBRERÍA SCRAPER.....	57
LIBRERÍA UTILIDADES	58
DIAGRAMAS DE FLUJO	60

GESTIÓN DE USUARIOS	60
<i>Registrarse</i>	60
<i>Identificarse</i>	61
FUNCIONES DE APOYO.....	62
<i>Generar Contenido</i>	62

Índice de Ilustraciones

DIAGRAMA DE CLASES	1
DIAGRAMA DE FLUJO – REGISTRARSE	60
DIAGRAMA DE FLUJO – IDENTIFICARSE	61
DIAGRAMA DE FLUJO – GENERAR CONTENIDO	62

Diagrama de clases

La aplicación trabajará directamente contra la base de datos con la finalidad de que no sea necesario cargar todos los datos permanentes en objetos. De este modo conseguimos:

- Minimizar la cantidad de información temporal que debe ser almacenada.
- Minimizar la carga de trabajo durante el inicio de sesión.

Se almacenará en objetos aquellos cálculos que se realicen durante la ejecución de una sesión con la finalidad de minimizar el número de operaciones evitando volver a realizar las que ya se han ejecutado.

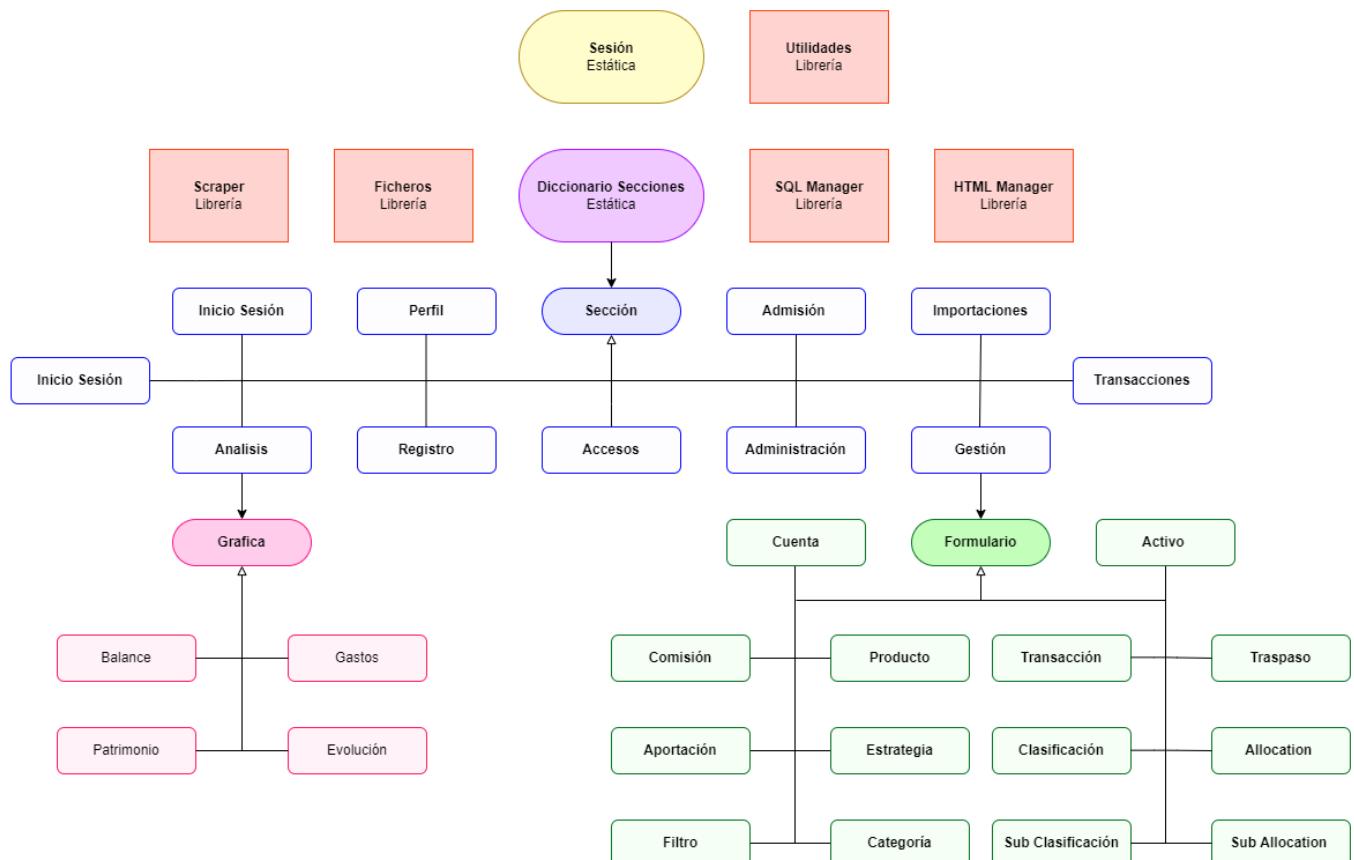


Ilustración 1 Diagrama de clases



A continuación, se explican con detalle los atributos y funciones de cada clase:

Sesión

Finalidad:

- Almacenar aquellos valores temporales que resultan imprescindibles para acceder a los datos permanentes del usuario que este autenticado.
- Gestionar la autenticación de los usuarios mediante el establecimiento y cierre de sesiones.
- Gestionar el acceso a la información permanente de los usuarios.

Características:

- **Estática:** Soló puede existir una única instancia por cada ejecución del programa.

Atributos:

- **Usuario:** Texto cuya finalidad es determinar cuál es el usuario que está identificado.
 - Si el usuario está identificado, su valor será el que aparece en la Base de datos como Usuario.
 - Si el usuario no está identificado, su valor será una cadena de texto vacía.
- **Permisos:** Texto cuya finalidad es determinar qué nivel de acceso tiene el usuario cuando está identificado.
 - Si el usuario está identificado, su valor será el que aparece en la Base de datos como Tipo Usuario.
 - Si el usuario no está identificado, su valor será una cadena de texto vacía.
- **Contraseña:** Texto cuya finalidad es almacenar temporalmente la contraseña para que el resto de la aplicación pueda acceder a la información encriptada.
 - Si el usuario está identificado, su valor será el que el usuario introdujo por teclado.
 - Si el usuario no está identificado, su valor será una cadena de texto vacía.
- **Tiempo Inicio:** Fecha cuya finalidad es almacenar temporalmente la fecha y hora en la que se produjo el inicio de sesión para determinar si el acceso a los datos permanentes ha expirado.
- **Tiempo Permitido:** Número entero cuya finalidad es almacenar la constante que representa el tiempo máximo permitido de inactividad antes de que se fuerce el cierre de la sesión.

Funciones:

- **Sesión:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:** Inicializa los valores de los atributos a sus valores por defecto.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen los valores por defecto.



- **Iniciar Sesión:**

- **Descripción:** Permite a un usuario no identificado obtener la categoría de usuario identificado.
- **Funcionamiento:**
 - Delega la comprobación de las condiciones a la función Comprobar Credenciales de la misma clase
 - Si se cumplen las condiciones:
 - Recupera los permisos del usuario.
 - Actualiza los atributos Usuario, Permisos y Contraseña con los valores del usuario.
 - Actualiza el tiempo de inicio con la fecha y hora actual del sistema.
 - Delega la redirección del usuario a la función Redireccionar de la clase Secciones. ([Balance](#))
 - Delega la generación de un acceso a la función Generar Acceso de la librería SQL.
 - Si no se cumplen las condiciones:
 - Delega la generación de un mensaje error a la función Reportar Error la clase Secciones.
 - Delega la generación de un intento de acceso a la función Generar Acceso de la librería SQL.
- **Entrada:** Valores de cada uno de los campos del formulario de inicio de sesión. ([usuario y contraseña](#))
- **Salida:** Devuelve un booleano que permite confirmar el inicio de sesión.
 - Booleano verdadero si se ha podido iniciar la sesión.
 - Booleano falso si no se ha podido iniciar la sesión.
- **Precondiciones:** Ninguna.
- **Postcondiciones:**
 - Si se cumplen las condiciones:
 - Actualiza los atributos Usuario, Permisos y Contraseña con los valores del usuario.
 - Actualiza el tiempo de inicio con la fecha y hora actual del sistema.

- **Cerrar Sesión:**

- **Descripción:** Finaliza el acceso a los datos permanentes del usuario.
- **Funcionamiento:**
 - Inicializa los valores de los atributos a sus valores por defecto.
 - Delega la redirección del usuario a la función Redireccionar de la clase Secciones. ([Inicio](#))
- **Entrada:** Ninguna.
- **Salida:** Ninguna.
- **Precondiciones:** El usuario debe estar registrado y correctamente identificado.
- **Postcondiciones:** Los atributos tienen los valores por defecto.

- **Añadir admisión**

- **Descripción:** Permite añadir el usuario a la lista de admisiones.
- **Funcionamiento:**
 - Delega la comprobación de las condiciones que se deben cumplir para añadir la admisión a la función comprobar admisión de la misma clase.
 - Calcula la sal y el hash de la contraseña.
 - Delega el proceso de añadir al usuario sin permisos a la función crear usuario de la librería SQL.
 - Delega el proceso de añadir la admisión a la función Añadir admisión de la librería SQL.
- **Entrada:** Valores de cada uno de los campos del formulario de registro.
- **Salida:**
 - Booleano Verdadero si el usuario se ha registrado.
 - Booleano Falso si el usuario no se ha registrado.
- **Precondiciones:** Ninguna.



- **Postcondiciones:** Si el usuario se registra correctamente se le redirecciona a la sección de Inicio de sesión.
- **Añadir usuario**
 - **Descripción:** Permite añadir el usuario a la lista de usuarios.
 - **Funcionamiento:**
 - Obtiene los datos de la admisión que se va a convertir en usuario.
 - Delega el proceso de añadir al usuario a la función Añadir usuario de la librería SQL.
 - Delega el proceso crear la base de datos del usuario a la función Crear BD de la librería SQL.
 - Delega el proceso de agregar permisos al usuario a la función Asignar Permisos al Usuario de la librería SQL.
 - **Entrada:**
 - Nombre de usuario de la admisión que se ha aceptado.
 - Nivel de permisos que se le asignan al nuevo usuario.
 - **Salida:** Booleano Verdadero.
 - **Precondiciones:** La admisión aceptada existe.
 - **Postcondiciones:**
 - Se crea la Base de Datos del nuevo usuario.
 - Se asignan los permisos al usuario.
- **Añadir Aportación:**
 - **Descripción:** Permite al usuario identificado añadir una aportación.
 - **Funcionamiento:**
 - Delega el proceso de añadir la aportación a la función Añadir aportación de la librería SQL. ([Inicio](#))
 - **Entrada:** Valores de cada uno de los campos de la tabla de aportaciones.
 - **Salida:** Ninguna.
 - **Precondiciones:** Se ha comprobado previamente que se cumplen los requisitos para añadir la aportación.
 - **Postcondiciones:** La aportación se añade a la base de datos.
- **Añadir Comisión:**
 - **Descripción:** Permite al usuario identificado añadir una comisión.
 - **Funcionamiento:**
 - Delega el proceso de añadir la Comisión a la función Añadir comisión de la librería SQL. ([Inicio](#))
 - **Entrada:** Valores de cada uno de los campos de la tabla de comisiones.
 - **Salida:** Ninguna.
 - **Precondiciones:** Se ha comprobado previamente que se cumplen los requisitos para añadir la comisión.
 - **Postcondiciones:** La comisión se añade a la base de datos.
- **Añadir Transacción:**
 - **Descripción:** Permite al usuario identificado añadir una transacción.
 - **Funcionamiento:**
 - Delega el proceso de añadir la transacción a la función Añadir transacción de la librería SQL . ([Inicio](#))
 - **Entrada:** Valores de cada uno de los campos de la tabla de transacciones.
 - **Salida:** Ninguna.
 - **Precondiciones:** Se ha comprobado previamente que se cumplen los requisitos para añadir la transacción.
 - **Postcondiciones:** La transacción se añade a la base de datos.



- **Añadir Traspaso:**

- **Descripción:** Permite al usuario identificado añadir un traspaso.
- **Funcionamiento:**
 - Delega el proceso de añadir el traspaso a la función Añadir traspaso de la librería SQL .([Inicio](#))
- **Entrada:** Valores de cada uno de los campos de la tabla de traspasos.
- **Salida:** Ninguna.
- **Precondiciones:** Se ha comprobado previamente que se cumplen los requisitos para añadir el traspaso.
- **Postcondiciones:** El traspaso se añade a la base de datos.

- **Eliminar Usuario:**

- **Descripción:** Elimina toda la información permanente del usuario objetivo.
- **Funcionamiento:**
 - Delega la eliminación de los datos permanentes a la función Eliminar Usuario de la librería SQL.
 - Delega la eliminación del usuario a la función Eliminar Usuario de la librería SQL.
 - Si el usuario objetivo es el usuario identificado.
 - Delega el cierre de la sesión a la función Cerrar sesión de la misma clase.
- **Entrada:**
 - Usuario objetivo o una cadena de texto vacía si el usuario objetivo es el usuario identificado.
- **Salida:** Ninguna.
- **Precondiciones:** El usuario objetivo existe.
- **Postcondiciones:** El usuario y la información permanente asociada ya no existe.

- **Eliminar Admisión:**

- **Descripción:** Elimina el registro de la admisión de un usuario.
- **Funcionamiento:**
 - Delega la eliminación de la admisión a la librería SQL.
- **Entrada:**
 - Usuario objetivo o una cadena de texto vacía si el usuario objetivo es el usuario identificado.
- **Salida:** Ninguna.
- **Precondiciones:** La admisión objetivo existe.
- **Postcondiciones:** La admisión objetivo ya no existe.



- **Actualizar Credenciales:**

- o **Descripción:** Restablece las credenciales del usuario identificado por las recibidas por parámetro.
- o **Funcionamiento:**
 - Comprueba si las contraseñas coinciden.
 - Comprueba si la contraseña anterior coincide con la actual.
 - Si cumple todas las condiciones delega la actualización de la contraseña a la función Actualizar Contraseña de la misma clase.
- o **Entrada:** Valores de cada uno de los campos del formulario de inicio de cambio de contraseña.
(contraseña anterior, contraseña nueva y contraseña repetida)
- o **Salida:**
 - Booleano verdadero si cumple todas las condiciones para cambiar la contraseña.
 - Booleano Falso si no cumple alguna de las condiciones para cambiar la contraseña.
- o **Precondiciones:** El usuario debe estar registrado y correctamente identificado.
- o **Postcondiciones:** Si no se cumple alguna condición modifica el mensaje de error indicando que condición es la que no se cumple.

- **Actualizar Contraseña:**

- o **Descripción:** Restablece las credenciales del usuario identificado por las recibidas por parámetro.
- o **Funcionamiento:**
 - Calcula una nueva sal para el usuario identificado.
 - Calcula el nuevo Hash para el usuario identificado.
 - Sustituye en la Base de Datos los nuevos valores de la Sal y el Hash.
- o **Entrada:** Texto que representa la nueva contraseña para el usuario.
- o **Salida:** Ninguna
- o **Precondiciones:** El parámetro de contraseña puede ser el mismo que la contraseña anterior.
- o **Postcondiciones:** Se actualizan las credenciales del usuario.



- **Actualizar Usuario:**

- **Descripción:** Modifica los datos permanentes del usuario que está actualmente registrado con los datos que recibe por parámetro.
- **Funcionamiento:** Delega el proceso de actualizar los datos permanentes del usuario a la función Actualizar Usuario de la librería SQL.
- **Entrada:** Datos introducidos por teclado en el formulario Editar Perfil: nombre, primer apellido, segundo apellido, DNI, teléfono y email.
- **Salida:** Ninguna.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Se actualizan los datos permanentes del usuario.

- **Comprobar Permisos:**

- **Descripción:** Permite conocer si los permisos del usuario son mayores que los recibidos como parámetro. Esta función se llamará cada vez que el usuario trate de acceder a una nueva sección. Su finalidad es determinar si se le permite o deniega el acceso.
- **Funcionamiento:**
 - Comprueba si el usuario esta registrado.
 - Comprueba si los permisos recibidos como parámetro son válidos.
(No Identificado, Usuario, Administrador, Creador)
 - Compara los permisos del usuario con los recibidos como parámetro
- **Entrada:**
 - Texto que representa si:
 - El nivel de permisos puede ser Estrictamente Igual que el recibido como parámetro. (=)
 - El nivel de permisos puede ser Igual o Superior que el recibido como parámetro. (>=)
 - el nivel de permisos puede ser Igual o Inferior que el recibido como parámetro. (<=)
 - El nivel de permisos puede ser Superior que el recibido como parámetro. (>)
 - el nivel de permisos puede ser Menor que el recibido como parámetro. (<)
 - Texto que representa el nivel de permisos con el que se pretende comparar los permisos del usuario.
- **Salida:**
 - Booleano Falso si no hay ningún usuario registrado.
 - Booleano Falso si los permisos recibidos como parámetro no son válidos.
 - Booleano Falso si los permisos del usuario no son acordes a los recibidos como parámetro.
 - Booleano Verdadero si los permisos del usuario son acordes a los recibidos como parámetro.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.



- **Comprobar Tiempo Sesión:**

- **Descripción:** Permite conocer si se ha terminado el tiempo de sesión por inactividad.
- **Características:** Se ejecutará cada vez que se intente un acceso a la información permanente.
- **Funcionamiento:**
 - Obtiene la fecha y hora actual del sistema.
 - Suma el tiempo máximo permitido de inactividad a la fecha y hora registrada al iniciar sesión.
 - Si el tiempo sesión no supera el máximo permitido.
 - Actualiza la variable Tiempo de Inicio con la fecha y hora actual.
- **Entrada:** Ninguna.
- **Salida:**
 - Booleano Verdadero si el tiempo de sesión ha expirado.
 - Booleano Falso si el tiempo de sesión no ha expirado.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Si el tiempo de sesión no ha expirado actualiza el tiempo de sesión.

- **Comprobar Sesión Iniciada:**

- **Descripción:** Permite conocer si la sesión está iniciada
- **Funcionamiento:** Si el atributo usuario de la sesión está vacío, la sesión no está iniciada.
- **Entrada:** Ninguna.
- **Salida:**
 - Booleano Verdadero si la sesión está iniciada.
 - Booleano Falso si la sesión no está iniciada.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Si el tiempo de sesión no ha expirado, actualiza el tiempo de sesión.

- **Comprobar Contraseña:**

- **Descripción:** Permite conocer si la contraseña de un usuario es correcta.
- **Funcionamiento:**
 - Delega a la librería SQL la obtención de la Sal para ese usuario.
 - Calcula el Hash de la contraseña recibida como parámetro utilizando la Sal.
 - Delega a la librería SQL la comprobación de si el Hash calculado coincide con el almacenado por dicho usuario.
- **Entrada:**
 - Usuario cuya contraseña se quiere verificar.
 - Contraseña que se pretende hacer pasar por la correcta.
- **Salida:**
 - Booleano Verdadero si la contraseña es correcta.
 - Booleano Falso si la contraseña no es correcta.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.



- **Comprobar Credenciales:**

- **Descripción:** Permite conocer si las credenciales de un usuario son correctas.
- **Funcionamiento:**
 - Comprueba si existe algún usuario registrado con ese nombre de usuario.
 - Comprueba si existe alguna admisión con ese nombre de usuario.
 - Delega la comprobación de si la contraseña es correcta a la función Comprobar contraseña de la misma clase.
- **Entrada:** Valores de cada uno de los campos del formulario de inicio de sesión. ([usuario y contraseña](#))
- **Salida:**
 - Booleano Verdadero si se cumplen todas las condiciones.
 - Booleano Falso si no se cumplen alguna condición.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Si no se cumple alguna condición modifica el mensaje de error, indicando que condición es la que no se cumple.

- **Comprobar admisión:**

- **Descripción:** Permite determinar si el usuario puede ser añadido a la lista de admisiones.
- **Funcionamiento:**
 - Comprobar si dicho usuario no existe en las tablas de usuarios y admisiones.
 - Comprobar que las contraseñas coincidan.
 - Comprobar que se haya marcada la casilla de Términos y condiciones.
- **Entrada:** Valores de cada uno de los campos del formulario de registro.
- **Salida:**
 - Booleano Verdadero si el usuario puede registrarse.
 - Booleano Falso si el usuario no puede registrarse.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Genera un mensaje de error enumerando las condiciones que no se cumplen.

- **Existe Cuenta:**

- **Descripción:** Comprueba si el usuario identificado tiene registrada la cuenta cuyo identificador se recibe como parámetro.
- **Funcionamiento:** Delega la comprobación a la función Existe Cuenta de la librería SQL.
- **Entrada:** Identificador de la cuenta que se quiere comprobar si existe.
- **Salida:**
 - Booleano verdadero si la cuenta existe.
 - Booleano falso si la cuenta no existe.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.



- **Existe Activo:**

- o **Descripción:** Comprueba si el usuario identificado tiene registrado el activo cuyo identificador se recibe como parámetro.
- o **Funcionamiento:** Delega la comprobación a la función Existe Activo de la librería SQL.
- o **Entrada:** Identificador del activo que se quiere comprobar si existe.
- o **Salida:**
 - Booleano verdadero si el activo existe.
 - Booleano falso si el activo no existe.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Existe Aportación:**

- o **Descripción:** Comprueba si el usuario identificado tiene registrada la aportación cuyos identificadores se reciben como parámetro.
- o **Funcionamiento:** Delega la comprobación a la función Existe Aportación de la librería SQL.
- o **Entrada:** Identificadores de la aportación que se quiere comprobar si existe.
- o **Salida:**
 - Booleano verdadero si el activo existe..
 - Booleano falso si el activo no existe.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.



- **Existe Comisión:**

- **Descripción:** Comprueba si el usuario identificado tiene registrada la comisión cuyos identificadores se reciben como parámetro.
- **Funcionamiento:** Delega la comprobación a la función Existe Comisión de la librería SQL.
- **Entrada:** Identificadores de la comisión que se quiere comprobar si existe.
- **Salida:**
 - Booleano verdadero si el activo existe.
 - Booleano falso si el activo no existe.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Existe Transacción:**

- **Descripción:** Comprueba si el usuario identificado tiene registrada la transacción cuyos identificadores se reciben como parámetro.
- **Funcionamiento:** Delega la comprobación a la función Existe Transacción de la librería SQL.
- **Entrada:** Identificadores de la transacción que se quiere comprobar si existe.
- **Salida:**
 - Booleano verdadero si el activo existe.
 - Booleano falso si el activo no existe.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Existe Traspaso:**

- **Descripción:** Comprueba si el usuario identificado tiene registrado el traspaso cuyos identificadores se reciben como parámetro.
- **Funcionamiento:** Delega la comprobación a la función Existe Traspaso de la librería SQL.
- **Entrada:** Identificadores del traspaso que se quiere comprobar si existe.
- **Salida:**
 - Booleano verdadero si el activo existe.
 - Booleano falso si el activo no existe.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Existe Clasificación:**

- **Descripción:** Comprueba si el usuario identificado tiene registrada la clasificación cuyos identificadores se reciben como parámetro.
- **Funcionamiento:** Delega la comprobación a la función Existe Clasificación de la librería SQL.
- **Entrada:** Identificadores de la clasificación que se quiere comprobar si existe.
- **Salida:**
 - Booleano verdadero si el activo existe.
 - Booleano falso si el activo no existe.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.



- **Get Usuario:**

- o **Descripción:** Obtiene de la base de datos todos los campos del usuario objetivo.
- o **Funcionamiento:** Delega la obtención de los datos a la librería SQL.
- o **Entrada:** Usuario objetivo o una cadena de texto vacía si el usuario objetivo es el usuario identificado.
- o **Salida:** Todos los datos del usuario objetivo.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Get Usuarios:**

- o **Descripción:** Obtiene de la base de datos todos los campos de todos los usuarios.
- o **Funcionamiento:** Delega la obtención de los datos a la librería SQL.
- o **Entrada:** Ninguna.
- o **Salida:** Lista de usuarios con todos los datos para cada usuario.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Get Admisión:**

- o **Descripción:** Obtiene de la base de datos todos los campos de la admisión objetivo.
- o **Funcionamiento:** Delega la obtención de los datos a la librería SQL.
- o **Entrada:** Nombre de usuario de la Admisión objetivo.
- o **Salida:** Todos los datos de la admisión objetivo.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Get Admisiones:**

- o **Descripción:** Obtiene de la base de datos todos los campos de todas las admisiones.
- o **Funcionamiento:** Delega la obtención de los datos a la librería SQL.
- o **Entrada:** Ninguna.
- o **Salida:** Lista de admisiones con todos los datos para cada admisión.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Get Activos:**

- o **Descripción:** Obtiene de la base de datos todos los campos de todos los activos.
- o **Funcionamiento:** Delega la obtención de los datos a la librería SQL.
- o **Entrada:** Ninguna.
- o **Salida:** Lista de activos.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.



Secciones

Finalidad:

- Clasificar las funcionalidades de la aplicación conforme a una estructura que facilite:
 - o El acceso a las características de cada funcionalidad.
 - o La incorporación futura de nuevas funcionalidades
- Gestionar la navegación a través de las diferentes secciones de la aplicación.
- Podemos diferenciar las siguientes secciones:
 - o **Principal:** Ofrece información general.
 - **Guía:** Sobre el funcionamiento de la aplicación
 - **Blog:** Guía de finanzas personales
 - o **Sesión:** Gestiona la sesión.
 - **Iniciar Sesión:** Permite a un usuario no identificado iniciar la sesión.
 - **Cerrar Sesión:** Permite a un usuario identificado cerrar la sesión.
 - o **Registro:** Permite a un usuario no identificado realizar una solicitud de admisión.
 - o **Configuración:** Permite al usuario identificado personalizar la sesión.
 - **Perfil:** Permite modificar los datos registro.
 - o **Admisiones:** Permite a los usuarios administradores:
 - **Admisión:** Gestionar las solicitudes de admisión.
 - **Administración:** Gestionar los permisos de los usuarios registrados.
 - **Accesos:** Comprobar los accesos de los usuarios registrados.
 - o **Transacciones:** Permite al usuario identificado gestionar sus datos de manera individual.
 - o **Importaciones:** Permite al usuario identificado gestionar sus datos financieros a granel.
 - **Importaciones:** Permite modificar los datos registro
 - **Exportaciones:** Permite modificar los datos registro

Características:

- **Estática:** Sólo puede existir una única instancia por cada ejecución del programa.

Atributos:

- **Secciones:** Diccionario que contiene cada una de las secciones que conforman la aplicación.
- **Sección Actual:** Texto que representa la sección que está visualizando actualmente el usuario.
- **Subsección Actual:** Texto que representa la subsección que se está visualizando actualmente.
- **Parámetros:** Texto que almacena los parámetros que necesiten las secciones que así los requieran.
- **Mensaje:** Texto que representa el siguiente mensaje de error que debe mostrarse al usuario:
 - o Si no se ha producido ningún error que notificar al usuario será una cadena de texto vacío.
 - o Si hay algún error que notificar al usuario será un texto con dicho mensaje.

Funciones:

- **Secciones:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Llama a la constructora de cada una de las secciones que conforman la aplicación.
 - Añade todas las secciones al diccionario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Todas las secciones de la aplicación han sido inicializadas y añadidas al diccionario.
 - **Reportar Error:**
 - **Descripción:** Modifica el valor del atributo Mensaje con el parámetro recibido.
 - **Entrada:** Texto con el mensaje de error que debe mostrarse al usuario.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Se actualiza el valor del atributo Mensaje.
 - **Redireccionar:**
 - **Descripción:** Permite indicar cuál es la sección y subsección que se debe visualizar.
 - **Funcionamiento:**
 - Modifica el valor del atributo Subsección actual.
 - En función de la subsección actual se calcula la sección actual.
 - **Entrada:** Texto con el identificador de la nueva subsección actual que debe mostrarse al usuario.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Se actualiza el valor de los atributos subsección actual y sección actual.



- **Generar Redirección:**

- o **Descripción:** Generar el código HTML que permite efectuar la redirección.
- o **Funcionamiento:**
 - Separa la dirección de la nueva sección de los parámetros.
 - Actualiza el valor de los parámetros.
 - Busca la nueva sección actual en el diccionario de secciones.
 - Si la sección no existe:
 - Delega la generación de un mensaje error a la función Reportar Error la misma clase.
 - Cambia a la sección actual a la principal mediante la función redireccionar de la misma clase.
 - Si la sección existe:
 - Delega la comprobación de los permisos de la nueva sección a la función Comprobar Acceso de la clase Sección de la nueva sección actual.
 - Si el usuario tiene acceso a la nueva sección:
 - Cambia a la sección actual a la indicada por parámetro mediante la función redireccionar de la misma clase.
 - Si el usuario no tiene acceso:
 - Delega la generación de un mensaje error a la función Reportar Error de la misma clase.
 - Cambia a la sección actual a la de inicio de sesión mediante la función Redireccionar de la misma clase.
 - Ejecuta la función redirect de la librería Flask utilizando como parámetro el identificador de la sección actual.
- o **Entrada:** texto con el identificador de la nueva dirección a la que se pretende acceder y los parámetros que requiera la subsección que ha sido llamada separados por el carácter '/'.
- o **Salida:** Texto con formato HTML que representa el código de redirección.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:**
 - El contenido del atributo Redirección es una cadena de texto vacía.
 - El contenido del atributo Sección Actual se actualiza.

- **Generar Índice:**

- o **Descripción:** Genera dinámicamente el índice de navegación de la sección.
- o **Funcionamiento:** Genera un menú de navegación utilizando la función índice de la librería HTML basándose en:
 - El nivel de permisos de la sección actual y del usuario.
 - La sección y subsección actuales.
- o **Entrada:** Ninguna.
- o **Salida:** Texto con formato HTML que representa el índice de navegación de la sección.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.



- **Generar Página:**

- **Descripción:** Genera dinámicamente la página que se muestra al usuario en función de:
 - La sección actual.
 - El último mensaje de error que debe mostrarse al usuario
 - Si existe alguna redirección pendiente.
- **Funcionamiento:**
 - Comprueba si existe alguna redirección pendiente.
 - Existe una redirección pendiente cuando la función Generar Contenido de la clase Sección devuelve una cadena vacía como contenido de la página actual.
 - Si existe delega alguna redirección pendiente delega el proceso de redirección a la función Redirect de la librería Flask.
 - Si no existe ninguna redirección pendiente:
 - Añade la cabecera de la página.
 - Delega la generación del índice de navegación de la sección actual a la función Generar Índice de la misma clase.
 - Delega la generación del mensaje de error a la función Generar Mensaje Error de la librería HTML.
 - Delega la generación del contenido de la sección actual a la función Generar Contenido de la clase Sección para la Sección actual.
 - Añade el pie de la página.
- **Entrada:** Ninguna.
- **Salida:**
 - Si existe alguna redirección pendiente: Texto con formato HTML que efectuara la redirección.
 - Si no existe ninguna redirección pendiente: Texto con formato HTML que concatena el índice de navegación, el mensaje de error y el contenido de la sección actual.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.



Sección

Finalidad:

- Generar el contenido para cada sección de manera universal.
- Evaluar la accesibilidad del usuario a cada sección.

Atributos:

- **Sección:** Texto que almacena una constante que representa distintas las secciones mediante un identificador único permitiendo diferenciarlas.
- **Subsección:** Texto que almacena una constante que representa distintas las subsecciones mediante un identificador único, permitiendo diferenciarlas.
- **Título:** Texto con el que se representa la sección.
- **Estricto:** Texto que representa la relación que debe existir entre los permisos del usuario y de la sección para que el usuario pueda acceder.
 - “=”: Los permisos del usuario y de la sección deben coincidir.
 - “>”: Los permisos del usuario deben ser mayores que los de la sección.
 - “<”: Los permisos del usuario deben ser menores que los de la sección.
 - “>=”: Los permisos del usuario deben ser mayores o iguales que los de la sección
 - “<=”: Los permisos del usuario deben ser menores o iguales que los de la sección.
- **Permisos:** Texto que almacena la constante que representa el nivel de permisos necesario para que el usuario pueda acceder a esta sección. ([No Identificado](#), [Usuario](#), [Administrador](#), [Creador](#))

Funciones:

- **Sección:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:** Inicializa los valores de los atributos con sus valores por defecto.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen los valores por defecto.
- **Comprobar Acceso:**
 - **Descripción:** Permite determinar si el usuario puede acceder a esta sección.
 - **Funcionamiento:**
 - Delega la comparación de los permisos del usuario con los permisos necesarios de la sección a la función Comprobar Permisos de la clase Sesión.
 - Delega la comprobación de Si la sesión esta iniciada a la función Comprobar Sesión iniciada de la clase Sesión.
 - Si la sesión esta iniciada y los permisos de acceso son suficientes delega la comprobación de Si la sesión ha caducado a la función Comprobar Tiempo de Sesión de la clase Sesión.
 - **Entrada:** Ninguna.
 - **Salida:**
 - Booleano Verdadero si el usuario puede acceder a la sección.
 - Booleano Falso si el usuario no puede acceder a la sección.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.



- **Generar Índice:**

- **Descripción:** Genera dinámicamente el índice de la sección.
- **Características:** Esta función debe ser reescrita por sus clases herederas.
- **Funcionamiento:** Genera dinámicamente el índice de la sección.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el índice de la sección.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Generar Contenido:**

- **Descripción:**
 - Genera dinámicamente el contenido de la sección.
 - Está pendiente de los inputs introducidos por el usuario permitiendo:
 - Capturar los campos de los formularios mediante la función Request de la librería Flask
 - Interpretar cuando es necesario efectuar una redirección y notificarlo a la clase Secciones devolviendo una cadena vacía como contenido de la página.
- **Características:** Esta función puede ser reescrita por sus clases herederas si necesitan realizar una funcionalidad más específica.
- **Funcionamiento:**
 - Delega la generación del formulario a la función Generar Formulario de la misma clase.
 - Delega la preparación de la página para recibir los campos del formulario a la función Capturar campos de la misma clase.
- **Entrada:** Ninguna.
- **Salida:**
 - Texto con formato HTML que representa el contenido de la sección.
 - Una cadena vacía en caso de que sea necesario efectuar una redirección.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Generar Formulario:**

- **Descripción:** Permite generar un formulario.
- **Características:** Esta función debe ser reescrita por sus clases herederas.
- **Funcionamiento:** Delega su ejecución a las clases herederas.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario de la sección correspondiente.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.



- **Capturar Campos Global:**

- **Descripción:** Permite capturar los campos que se envían en los formularios de las diferentes secciones.
- **Funcionamiento:**
 - Captura el parámetro “Redirigir” encargado de detectar cuál es la funcionalidad que sea activado.
 - Delega la captura de campos específicos de cada sección a la función capturar campos.
 - Captura el campo “Ordenar” encargado de gestionar el orden de las filas en tablas.
- **Entrada:** Ninguna.
- **Salida:**
 - Booleano falso por defecto.
 - Booleano verdadero si es necesario refrescar la página para efectuar una redirección.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Capturar Campos:**

- **Descripción:** Permite capturar los campos que se envían en los formularios de las diferentes secciones.
- **Características:** Esta función debe ser reescrita por sus clases herederas.
- **Funcionamiento:** Delega su ejecución a las clases herederas.
- **Entrada:**
 - Texto que permite detectar que funcionalidad se ha activado.
- **Salida:**
 - Booleano falso por defecto.
 - Booleano verdadero si es necesario refrescar la página para efectuar una redirección.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.



Inicio Sesión

Finalidad:

- Generar el contenido relacionado con el inicio de sesión.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Sección.
- **Permisos Necesarios:** No Identificado.

Funciones:

- **Inicio Sesión:**

- **Descripción:** Constructora de la clase.
 - **Funcionamiento:** Inicializa los valores de los atributos de su clase predecesora.
 - **Entrada:**
 - **Sección:** Texto que representa el nombre de la sección “Sesión”.
 - **Subsección:** Texto que representa el nombre de la subsección “Inicio de Sesión”.
 - **Título:** Texto con el que se representa la sección.
 - **Permisos:** Número entero que representa el nivel de permisos de la sección.
 - **Estricto:** Texto que representa la relación que debe existir entre los permisos del usuario y de la sección para que el usuario pueda acceder:
 - “=”: Los permisos del usuario y de la sección deben coincidir.
 - “>”: Los permisos del usuario deben ser mayores que los de la sección.
 - “<”: Los permisos del usuario deben ser menores que los de la sección.
 - “>=”: Los permisos del usuario deben ser mayores o iguales que los de la sección.
 - “<=”: Los permisos del usuario deben ser menores o iguales que los de la sección.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.

- **Generar Formulario:**

- **Descripción:** Genera el formulario de Inicio de Sesión que permite a cualquier usuario no identificado introducir sus credenciales para identificarse.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario de inicio de sesión.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

- **Capturar Campos:**

- **Descripción:** Prepara a la página para recibir los campos del formulario.
 - **Funcionamiento:**
 - Si Redirigir es “Iniciar Sesión”:
 - Captura los campos del formulario de Inicio de Sesión.
 - Delega el proceso de inicio de sesión a la función Iniciar Sesión de la clase Sesión.
 - **Entrada:** Texto que permite detectar qué funcionalidad se ha activado.
 - **Salida:**
 - Booleano falso si la sesión no se ha iniciado.
 - Booleano verdadero si la sesión se ha iniciado correctamente.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

Cerrar Sesión

Finalidad:

- Generar el procedimiento que permite cerrar la sesión.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Sección.
 - **Permisos Necesarios:** No Identificado.

Funciones:

- **Cerrar Sesión:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:** Inicializa los valores de los atributos de su clase predecesora.
 - **Entrada:**
 - **Sección:** Texto que representa el nombre de la sección “Sesión”.
 - **Subsección:** Texto que representa el nombre de la subsección “Cerrar Sesión”.
 - **Título:** Texto con el que se representa la sección.
 - **Permisos:** Número entero que representa el nivel de permisos de la sección.
 - **Estricto:** Texto que representa la relación que debe existir entre los permisos del usuario y de la sección para que el usuario pueda acceder.
 - “=”: Los permisos del usuario y de la sección deben coincidir.
 - “>”: Los permisos del usuario deben ser mayores que los de la sección.
 - “<”: Los permisos del usuario deben ser menores que los de la sección.
 - “>=”: Los permisos del usuario deben ser mayores o iguales que los de la sección.
 - “<=”: Los permisos del usuario deben ser menores o iguales que los de la sección.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
 - **Generar Contenido:**
 - **Descripción:** Permite cerrar la sesión que estuviera iniciada.
 - **Funcionamiento:** Delega el proceso de cerrar sesión a la función cerrar sesión de la clase Sesión.
 - **Entrada:** Ninguna.
 - **Salida:** Una cadena de texto vacía si la sesión se ha cerrado correctamente.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.



Registro

Finalidad:

- Generar el contenido relacionado con el registro de nuevos usuarios
- Gestionar la inserción de nuevas admisiones utilizando los datos del formulario de registro.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Sección.
- **Permisos Necesarios:** No Identificado.

Atributos:

- **Nombre:** Texto que permite almacenar temporalmente el último valor introducido en el campo Nombre del formulario de registro.
- **Primer Apellido:** Texto permite almacenar temporalmente el último valor introducido en el campo Primer Apellido del formulario de registro.
- **Segundo Apellido:** Texto almacena temporalmente el último valor introducido en el campo Segundo Apellido del formulario de registro.
- **DNI:** Texto permite almacenar temporalmente el último valor introducido en el campo DNI del formulario de registro.
- **Teléfono:** Texto que permite almacenar temporalmente el último valor introducido en el campo Teléfono del formulario de registro.
- **Email:** Texto que permite almacenar temporalmente el último valor introducido en el campo Email del formulario de registro.

Funciones:

- **Registro:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos a sus valores por defecto.
 - Inicializa los valores de los atributos de su clase predecesora.
 - **Entrada:**
 - **Sección:** Texto que representa el nombre de la sección “Registro”.
 - **Subsección:** Texto que representa el nombre de la subsección “Registro”.
 - **Título:** Texto con el que se representa la sección.
 - **Permisos:** Número entero que representa el nivel de permisos de la sección.
 - **Estricto:** Texto que representa la relación que debe existir entre los permisos del usuario y de la sección para que el usuario pueda acceder.
 - “=”: Los permisos del usuario y de la sección deben coincidir.
 - “>”: Los permisos del usuario deben ser mayores que los de la sección.
 - “<”: Los permisos del usuario deben ser menores que los de la sección.
 - “>=”: Los permisos del usuario deben ser mayores o iguales que los de la sección.
 - “<=”: Los permisos del usuario deben ser menores o iguales que los de la sección.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Todos los atributos tienen los valores por defecto.



- **Generar Formulario:**

- **Descripción:** Genera el formulario de registro que permite a cualquier usuario no identificado añadir su solicitud de admisión.
- **Funcionamiento:**
 - Utiliza el valor de sus atributos para asignar el valor inicial a los inputs del formulario de Registro.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa la ventana del formulario de registro.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Capturar Campos:**

- **Descripción:** Da comienzo al proceso de registro de un usuario.
- **Funcionamiento:**
 - Si redirigir es “Añadir Admisión”:
 - Captura los campos del formulario de Registro.
 - Modifica los valores de los atributos con los campos del formulario.
 - Delega la generación de la nueva admisión a la función Añadir Admisión de la clase Sesión.
- **Entrada:**
 - Texto que permite detectar qué funcionalidad se ha activado.
- **Salida:**
 - Booleano falso si no se puede añadir el usuario a la lista de admisión.
 - Booleano verdadero si se puede añadir el usuario a la lista de admisión.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Si se cumplen las condiciones añade una nueva admisión.



Perfil

Finalidad:

- Generar el contenido relacionado con el registro de nuevos usuarios.
- Gestionar la edición de datos personales con los datos del formulario de Editar Perfil.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Sección.
- **Permisos Necesarios:** Usuario, Administrador o Creador.

Funciones:

- **Perfil:**
 - o **Descripción:** Constructora de la clase.
 - o **Funcionamiento:** Inicializa los valores de los atributos de su clase predecesora.
 - o **Entrada:**
 - **Sección:** Texto que representa el nombre de la sección “Configuración”.
 - **Subsección:** Texto que representa el nombre de la subsección “Perfil”.
 - **Título:** Texto con el que se representa la sección.
 - **Permisos:** Número entero que representa el nivel de permisos de la sección.
 - **Estricto:** Texto que representa la relación que debe existir entre los permisos del usuario y de la sección para que el usuario pueda acceder.
 - “=”: Los permisos del usuario y de la sección deben coincidir.
 - “>”: Los permisos del usuario deben ser mayores que los de la sección.
 - “<”: Los permisos del usuario deben ser menores que los de la sección.
 - “>=”: Los permisos del usuario deben ser mayores o iguales que los de la sección.
 - “<=”: Los permisos del usuario deben ser menores o iguales que los de la sección.
 - o **Salida:** Ninguna.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - o **Descripción:** Genera los formularios de la sección Perfil que permitirán modificar la información permanente del usuario identificado.
 - o **Funcionamiento:**
 - Delega generación del formulario para modificar los datos del perfil a la función Formulario Editar Perfil de la misma clase.
 - Delega generación del formulario para modificar la contraseña a la función Formulario Cambiar Contraseña de la misma clase.
 - o **Entrada:** Ninguna.
 - o **Salida:** Texto con formato HTML que representa los formularios de la sección Perfil.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Ninguna.



- **Formulario Editar Perfil:**

- **Descripción:** Genera el formulario de Editar Perfil que permite modificar los datos personales del usuario identificado.
- **Funcionamiento:**
 - Delega la obtención de los datos personales del usuario identificado a la función Desencriptar Usuario de la clase Sesión.
 - Utiliza el valor de los datos personales del usuario obtenido de la Base de Datos para asignar el valor inicial a los inputs del formulario de Editar Perfil.
 - Incluye un campo oculto con el identificador “Redirigir” que permite identificar este formulario.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario de Editar Perfil.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Formulario Cambiar Contraseña:**

- **Descripción:** Genera el formulario de cambiar contraseña que permite al usuario identificado modificar su contraseña.
- **Funcionamiento:**
 - Sigue al usuario su contraseña anterior y la contraseña que la sustituirá.
 - Incluye un campo oculto con el identificador “Redirigir” que permite identificar este formulario.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario de Editar Perfil.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Capturar Campos:**

- **Descripción:** Da comienzo al proceso de Editar Perfil.
- **Funcionamiento:**
 - Si redirigir es “Editar Perfil”:
 - Captura los campos del formulario de Editar Perfil.
 - Delega el proceso de actualizar los datos del usuario a la función Actualizar Usuario de la clase Sesión.
 - Si redirigir es “Editar Contraseña”:
 - Captura los campos del formulario de Editar Contraseña.
 - Delega la actualización la contraseña a la función Actualizar Credenciales de la clase Sesión.
- **Entrada:**
 - Texto que permite detectar qué funcionalidad se ha activado.
- **Salida:**
 - Booleano falso si no se puede cambiar la contraseña.
 - Booleano verdadero si ha modificado la contraseña.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Los datos del perfil se han actualizado.

Admisión

Finalidad:

- Generar el contenido relacionado con la gestión de las solicitudes de registro.
 - Gestionar la inserción de nuevos usuarios.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Sección.
 - **Permisos Necesarios:** Administrador o Creador.

Funciones:

- **Admisión:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:** Inicializa los valores de los atributos de su clase predecesora.
 - **Entrada:**
 - **Sección:** Texto que representa el nombre de la sección “Admisiones”.
 - **Subsección:** Texto que representa el nombre de la subsección “Admisión”.
 - **Título:** Texto con el que se representa la sección.
 - **Permisos:** Número entero que representa el nivel de permisos de la sección.
 - **Estricto:** Texto que representa la relación que debe existir entre los permisos del usuario y de la sección para que el usuario pueda acceder.
 - “=”: Los permisos del usuario y de la sección deben coincidir.
 - “>”: Los permisos del usuario deben ser mayores que los de la sección.
 - “<”: Los permisos del usuario deben ser menores que los de la sección.
 - “>=”: Los permisos del usuario deben ser mayores o iguales que los de la sección.
 - “<=”: Los permisos del usuario deben ser menores o iguales que los de la sección.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
 - **Generar Formulario:**
 - **Descripción:** Genera el menú que permite la gestión de las solicitudes de registro.
 - **Funcionamiento:**
 - Delega la obtención de los datos de admisiones a la función get Admisiones de la clase Sesión.
 - Utiliza los resultados obtenidos de la Base de Datos para generar una tabla en la que:
 - Se muestran los usuarios pendientes de aceptación.
 - Permite acceder a una funcionalidad para aceptar o rechazar las solicitudes.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el menú de gestión de usuarios registrados.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.



- **Formulario Añadir Usuario:**

- **Descripción:** Genera la funcionalidad que permite aceptar una admisión.
- **Funcionamiento:** Genera un formulario con:
 - Un campo oculto para identificar el botón que se ha pulsado.
 - Una lista desplegable que permite seleccionar los permisos que se le asignarán al nuevo usuario.
 - Un campo oculto para identificar al usuario que se está aceptando.
 - Un botón para aceptar la solicitud.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario para aceptar la solicitud de admisión.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Formulario Eliminar Admisión:**

- **Descripción:** Genera la funcionalidad que permite eliminar una admisión.
- **Funcionamiento:** Genera un formulario con:
 - Un campo oculto para identificar el botón que se ha pulsado.
 - Un campo oculto para identificar la admisión que se está rechazando.
 - Un botón para rechazar la solicitud.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario para rechazar la solicitud de admisión.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Formulario Editar Filtros:**

- **Descripción:** Genera la funcionalidad que permite abrir el menú de edición de filtros.
- **Funcionamiento:** Genera un formulario con:
 - Un campo oculto para identificar el botón que se ha pulsado.
 - Un botón para abrir el menú de edición de filtros.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario Abrir el menú de Edición de Filtros.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Capturar Campos:**

- **Descripción:** Da comienzo al proceso de Aceptar o Rechazar una solicitud de admisión.
- **Funcionamiento:**
 - Si redirigir es “Añadir Usuario”:
 - Captura los campos del formulario de Añadir Usuario.
 - Delega el proceso de añadir al usuario a la función Añadir Usuario de la clase Sesión.
 - Delega el proceso de eliminar la admisión a la función Eliminar Admisión de la clase Sesión.
 - Si redirigir es “Eliminar Admisión”:
 - Captura los campos del formulario de Eliminar Admisión.
 - Delega el proceso de eliminar la admisión a la función Eliminar Admisión de la clase Sesión.
- **Entrada:** Texto que permite detectar qué funcionalidad se ha activado.
- **Salida:**
 - Booleano falso si se no se acepta o rechaza alguna admisión.
 - Booleano verdadero si se acepta o rechaza alguna admisión.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Los datos del perfil se han actualizado.



Universidad

Euskal Herriko
UnibertsitateaBILBOKO
INGENIARITZA ESKOLAESCUELA
DE INGENIERÍA DE BILBAO

Administración

Finalidad:

- Generar el contenido relacionado con la gestión de los usuarios registrados.
- Gestionar el nivel de permisos de los usuarios registrados.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Sección.
- **Permisos Necesarios:** Administrador o Creador.

Funciones:

- **Administración:**
 - o **Descripción:** Constructora de la clase.
 - o **Funcionamiento:** Inicializa los valores de los atributos de su clase predecesora.
 - o **Entrada:**
 - **Sección:** Texto que representa el nombre de la sección “Admisiones”.
 - **Subsección:** Texto que representa el nombre de la subsección “Administración”.
 - **Título:** Texto con el que se representa la sección.
 - **Permisos:** Número entero que representa el nivel de permisos de la sección.
 - **Estricto:** Texto que representa la relación que debe existir entre los permisos del usuario y de la sección para que el usuario pueda acceder.
 - “=”: Los permisos del usuario y de la sección deben coincidir.
 - “>”: Los permisos del usuario deben ser mayores que los de la sección.
 - “<”: Los permisos del usuario deben ser menores que los de la sección.
 - “>=”: Los permisos del usuario deben ser mayores o iguales que los de la sección.
 - “<=”: Los permisos del usuario deben ser menores o iguales que los de la sección.
 - o **Salida:** Ninguna.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - o **Descripción:** Genera el menú que permite la gestión de usuarios registrados.
 - o **Funcionamiento:**
 - Delega la obtención de los datos de usuarios a la función get Usuarios de la clase Sesión.
 - Utiliza los resultados obtenidos de la Base de Datos para generar una tabla en la que:
 - Se muestran los usuarios registrados y sus permisos actuales.
 - Permite acceder a una funcionalidad para modificar sus permisos.
 - Permite acceder a una funcionalidad para eliminarlos.
 - o **Entrada:** Ninguna.
 - o **Salida:** Texto con formato HTML que representa el menú de gestión de usuarios registrados.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Ninguna.



- **Formulario Actualizar Permisos:**

- **Descripción:** Genera la funcionalidad que permite modificar los permisos de un usuario registrado.
- **Funcionamiento:** Genera un formulario con:
 - Un campo oculto para identificar el botón que se ha pulsado.
 - Una lista desplegable que permite seleccionar los nuevos permisos a los que se asignarán usuario.
 - Un campo oculto para identificar al usuario que se está modificando.
 - Un botón para confirmar la solicitud.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario modificar los permisos.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Formulario Eliminar Usuario:**

- **Descripción:** Genera la funcionalidad que permite eliminar un usuario registrado.
- **Funcionamiento:** Genera un formulario con:
 - Un campo oculto para identificar el botón que se ha pulsado.
 - Un campo oculto para identificar el usuario que se va a eliminar.
 - Un botón para confirmar la eliminación del usuario.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario para eliminar al usuario.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Formulario Editar Filtros:**

- **Descripción:** Genera la funcionalidad que permite abrir el menú de Edición de filtros.
- **Funcionamiento:** Genera un formulario con:
 - Un campo oculto para identificar el botón que se ha pulsado.
 - Un botón para abrir el menú de Edición de filtros.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario abrir el menú de Edición de Filtros
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Capturar Campos:**

- **Descripción:** Da comienzo al proceso de Aceptar o Rechazar una solicitud de admisión.
- **Funcionamiento:**
 - Captura el campo oculto “Redirigir” para identificar de qué formulario se trata.
 - Si redirigir es “Actualizar Permisos”:
 - Captura los campos del formulario de Actualizar Permisos.
 - Delega el proceso de modificar los permisos a la función Actualizar Permisos de la clase Sesión.
 - Si redirigir es “Eliminar Usuario”:
 - Captura los campos del formulario de Eliminar Usuario.
 - Delega el proceso de eliminar al usuario a la función Eliminar Usuario de la clase Sesión.
- **Entrada:** Texto que permite detectar qué funcionalidad se ha activado.
- **Salida:**
 - Booleano falso si no se cambian permisos o se eliminan usuarios.
 - Booleano verdadero si se cambian permisos o se eliminan usuarios.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Los datos del perfil se han actualizado.



Accesos

Finalidad:

- Generar el contenido relacionado con los intentos de acceso.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Sección.
- **Permisos Necesarios:** Creador.

Atributos:

- **Usuario:** Variable de tipo texto que representa el último valor seleccionado para el filtro de usuarios del menú de intentos de acceso. Su valor por defecto será una cadena vacía que indica que no se aplica ningún filtro.

Funciones:

- **Accesos:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:** Inicializa los valores de los atributos de su clase predecesora.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Contenido:**
 - **Descripción:** Genera el contenido de la página Accesos.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el contenido de la página Accesos.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.
- **Generar Menú Accesos:**
 - **Descripción:** Genera el menú que permite visualización de los intentos de acceso.
 - **Funcionamiento:**
 - Delega la obtención de los datos de intentos de acceso a la función get Accesos de la librería SQL.
 - Utiliza los resultados obtenidos de la Base de Datos para generar una tabla en la que:
 - Se muestran los intentos de acceso de todos los usuarios ordenados cronológicamente.
 - Permite acceder a una funcionalidad que permite filtrar los intentos de acceso de un usuario.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el menú de los intentos de acceso.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.
- **Capturar Filtros:**
 - **Descripción:** Actualiza los filtros del menú intentos de acceso.
 - **Funcionamiento:**
 - Captura los campos del formulario de Filtrar Accesos generado en el Menú de intentos de acceso.
 - Actualiza el valor del atributo Usuario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** El atributo usuario ha sido actualizado.

Gestión

Finalidad:

- Clasificar los distintos tipos de formularios que permiten a los usuarios introducir sus datos económicos.
 - Gestionar la navegación a través de los diferentes formularios de la aplicación.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Sección.
 - **Permisos Necesarios:** Usuario, Administrador o Creador.

Atributos:

- **Formularios:** Diccionario que contiene cada uno de los tipos de formulario que contempla la aplicación.
 - **Formulario Actual:** Texto que representa el tipo de formulario que se está visualizando actualmente.
 - o Su valor por defecto será “Activos” para acceder a la gestión de los activos.

Funciones:

- **Gestión:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos con sus valores por defecto.
 - Inicializa los valores de los atributos de su clase predecesora.
 - Llama a la constructora de cada uno de los tipos de formulario que contempla la aplicación.
 - Añade todos los tipos de formulario al diccionario.
 - **Entrada:**
 - **Sección:** Texto que representa el nombre de la sección “Transacciones”.
 - **Subsección:** Texto que representa el nombre de la subsección “Transacciones”.
 - **Título:** Texto con el que se representa la sección.
 - **Permisos:** Número entero que representa el nivel de permisos de la sección.
 - **Estricto:** Texto que representa la relación que debe existir entre los permisos del usuario y de la sección para que el usuario pueda acceder.
 - “=”: Los permisos del usuario y de la sección deben coincidir.
 - “>”: Los permisos del usuario deben ser mayores que los de la sección.
 - “<”: Los permisos del usuario deben ser menores que los de la sección.
 - “>=”: Los permisos del usuario deben ser mayores o iguales que los de la sección.
 - “<=”: Los permisos del usuario deben ser menores o iguales que los de la sección.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:**
 - Los atributos tienen los valores por defecto.
 - Todos los tipos de formulario han sido inicializados y añadidos al diccionario.



- **Generar Índice:**

- **Descripción:** Genera dinámicamente el índice de la sección de transacciones.
- **Características:** Esta función debe ser reescrita por sus clases herederas.
- **Funcionamiento:** Genera dinámicamente el índice de la sección.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el índice de la sección de gestión.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Generar Contenido:**

○ **Descripción:**

- Genera dinámicamente el contenido de la sección.
- Está pendiente de los inputs introducidos por el usuario permitiendo:
 - Capturar los campos de los formularios mediante la función Request de la librería Flask.
 - Interpretar cuándo es necesario efectuar una redirección y notificarlo a la clase Secciones devolviendo una cadena vacía como contenido de la página.

○ **Funcionamiento:**

- Delega la actualización del formulario Actual y el subformulario Actual a la función Capturar Redirección de la misma clase.
- Delega la generación del formulario a la función Generar Formulario del formulario actual.
- Delega la preparación de la página para recibir los campos del formulario a la función Capturar campos del formulario actual.

○ **Entrada:** Ninguna.

○ **Salida:**

- Texto con formato HTML que representa el contenido de la sección.
- Una cadena vacía en caso de que sea necesario efectuar una redirección.

○ **Precondiciones:** Ninguna.

○ **Postcondiciones:** Ninguna.

- **Capturar Campos:**

○ **Descripción:** Delega su funcionamiento a la función capturar campos del formulario actual.

○ **Funcionamiento:**

- Delega su funcionamiento a la función capturar campos del formulario actual.

○ **Entrada:**

- Texto que permite detectar qué funcionalidad se ha activado.

○ **Salida:**

- Booleano falso si no se cambian permisos o se eliminan usuarios.
- Booleano verdadero si se cambian permisos o se eliminan usuarios.

○ **Precondiciones:** Ninguna.

○ **Postcondiciones:** Los datos del perfil se han actualizado.



- **Capturar Redirección:**

- **Descripción:** Permite generar el formulario de la sección.
- **Funcionamiento:**
 - Obtiene los parámetros generados durante la redirección desde la clase secciones.
 - Comprueba si los parámetros se corresponden con el identificador de alguno de los formularios.
 - En caso afirmativo:
 - Modifica el valor del atributo subformulario actual con el valor de los parámetros.
 - En función del subformulario actual se calcula el formulario actual.
- **Entrada:** Ninguna.
- **Salida:** Ninguna.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Actualiza los atributos Formulario Actual y subformulario Actual.



Análisis

Finalidad:

- Generar el contenido las ventanas de gráficos y resúmenes.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario, Administrador o Creador.

Atributos:

- **Formulario Actual:** Variable de tipo texto que almacena formulario que se representa actualmente.
- **Subformulario Actual:** Variable de tipo texto que almacena el subformulario que se representa actualmente.
- **Formularios:** Variable de tipo diccionario que almacena todos los formularios disponibles.

Funciones:

- **Análisis:**
 - o **Descripción:** Constructora de la clase.
 - o **Funcionamiento:**
 - Llama a la constructora de cada una de las gráficas que conforman la aplicación.
 - Añade todos los formularios al diccionario.
 - o **Entrada:** Ninguna.
 - o **Salida:** Ninguna.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Todas las gráficas de la aplicación han sido inicializados y añadidos al diccionario.
- **Generar Índice:**
 - o **Descripción:** Genera dinámicamente el índice de las gráficas.
 - o **Funcionamiento:** Genera dinámicamente el índice de las gráficas.
 - o **Entrada:** Ninguna.
 - o **Salida:** Texto con formato HTML que representa el índice de la sección
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Ninguna.
- **Generar Contenido:**
 - o **Descripción:** Delega la generación del contenido a la clase Gráficas.
 - o **Entrada:** Ninguna.
 - o **Salida:** Texto con formato HTML que representa el contenido de la página.
 - o **Precondiciones:** Ninguna
 - o **Postcondiciones:** Ninguna
- **Capturar Campos:**
 - o **Descripción:** Permite capturar los campos que se envían en las gráficas de las diferentes secciones.
 - o **Funcionamiento:** Delega su ejecución a la clase Formulario.
 - o **Entrada:** Texto que permite detectar que funcionalidad se ha activado.
 - o **Salida:**
 - Booleano falso por defecto.
 - Booleano verdadero si es necesario refrescar la página para efectuar una redirección.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Ninguna.



Importaciones

Finalidad:

- Generar el contenido del formulario que permite importar los datos de un usuario mediante un fichero csv.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Importaciones:**
 - o **Descripción:** Constructora de la clase.
 - o **Funcionamiento:** Inicializa los valores de los atributos de su clase predecesora.
 - o **Entrada:** Ninguna.
 - o **Salida:** Ninguna.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - o **Descripción:** Genera el menú que permite la importación de datos mediante ficheros csv.
 - o **Funcionamiento:** Genera el Formulario que permite la importación de datos mediante ficheros csv.
 - o **Entrada:** Ninguna.
 - o **Salida:** Texto con formato HTML que representa el menú de importación de datos mediante ficheros csv.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Ninguna.
- **Capturar Campos:**
 - o **Descripción:** Da comienzo al proceso de Insertar los datos de los ficheros importados.
 - o **Funcionamiento:**
 - Captura el campo oculto “Redirigir” para identificar de qué formulario se trata.
 - Si redirigir es “Importar CSV”:
 - Captura los campos del formulario de Importar CSV.
 - Delega el proceso de interpretar el fichero a la función Importar Datos CSV de la librería Ficheros.
 - o **Entrada:** Ninguna.
 - o **Salida:**
 - Booleano falso si no se ha introducido algún fichero.
 - Booleano verdadero si se ha introducido algún fichero.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Ninguna.

Formulario

Finalidad:

- Generar de forma universal el contenido de cualquier tipo de formulario.

Atributos:

- **Formulario:** Constante de tipo texto que representa los distintos tipos de formularios mediante un identificador único que permite diferenciarlos.
- **Sub Formulario:** Constante de tipo texto que representa los distintos tipos de subformularios mediante un identificador único que permite diferenciarlos.
- **Título:** Constante de tipo texto que con el que se almacena el texto que se mostrará al usuario cuando abra el formulario.
- **Acción:** Variable de tipo texto que permite identificar la acción que ha seleccionado el usuario y permitirá redirigir a la ventana correspondiente a dicha acción.
 - o **Tabla:** Valor por defecto. Redirige a la ventana en la que se muestra la información de aquellos campos que ya han sido registrados mediante este formulario.
 - o **Añadir:** Redirige a la ventana en la que se muestra el formulario que permite añadir nuevos datos.
 - o **Editar:** Redirige a la ventana en la que se muestra el formulario que permite modificar los datos del formulario desde el que se haya seleccionado dicha acción.
 - o **Eliminar:** Redirige a la ventana en la que se muestra el formulario que permite eliminar los datos del formulario desde el que se haya seleccionado dicha acción.
- **Parámetros:** Variable de tipo texto que almacena aquella información que debe mantenerse entre los distintos formularios cuando se navega entre ellos.
- **Filtro:** Instancia de la clase formulario que permite generar un formulario de tipo filtro mediante el cual se pueden configurar ciertas características del formulario al que está asociado.

Funciones:

- **Formulario:**
 - o **Descripción:** Constructora de la clase.
 - o **Funcionamiento:** Inicializa los valores de los atributos con sus valores por defecto.
 - o **Entrada:**
 - Texto que representa el identificador único del formulario que se quiere construir.
 - Texto que representa el identificador único del subformulario que se quiere construir.
 - Texto que representa el título que se mostrará al usuario cuando se abra el formulario
 - o **Salida:** Ninguna.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Los atributos tienen los valores por defecto.
- **Set Acción:**
 - o **Descripción:** Modifica la variable acción del formulario.
 - o **Entrada:** Texto que representa la nueva acción que se debe ejecutar.
 - o **Salida:** Ninguna.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Actualiza la variable acción.



- **Generar Formulario:**

- **Descripción:** Genera el formulario para insertar datos en función del tipo de formulario seleccionado
- **Funcionamiento:**
 - Comprueba el atributo acción
 - Delega la generación de la función a la función que está asociada a dicha acción
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario seleccionado.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Redirigir:**

- **Descripción:** Implementa con formato HTML el formulario de un botón con la capacidad de efectuar una redirección entre formularios.
- **Finalidad:** Se trata de una herramienta para simplificar el código de las clases herederas.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario del botón.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Formulario:**

- **Descripción:**
 - Genera el formulario indicado por las constantes formulario y subformulario y la variable acción.
- **Funcionamiento:** Se delega su ejecución a las clases herederas.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Capturar Campos:**

- **Descripción:** Prepara a la página para recibir los campos del formulario.
- **Funcionamiento:**
 - Si Redirigir es: "Añadir" modifica el valor de la acción a "Añadir" y recarga el formulario.
 - Si Redirigir es: "Editar" modifica el valor de la acción a "Editar" y recarga el formulario.
 - Si Redirigir es: "Eliminar" modifica el valor de la acción a "Eliminar" y recarga el formulario.
 - Si Redirigir es: "Tabla" modifica el valor de la acción a "Tabla" y recarga el formulario.
 - Delega a las clases herederas la posibilidad de añadir más redirecciones.
- **Entrada:** Texto que permite detectar qué funcionalidad se ha activado.
- **Salida:**
 - Booleano falso no es necesario recargar el formulario.
 - Booleano verdadero si se debe recargar el formulario.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

Activos

Finalidad:

- Generar el contenido del formulario de activos para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
 - **Permisos Necesarios:** Usuario.

Funciones:

- **Activos:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
 - **Generar Formulario:**
 - **Descripción:** Genera el formulario indicado por la variable acción p
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

Cuentas

Finalidad:

- Generar el contenido del formulario de cuentas para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
 - **Permisos Necesarios:** Usuario.

Funciones:

- **Cuentas:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
 - **Generar Formulario:**
 - **Descripción:** Genera el formulario indicado por la variable acción para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario.
 - **Precondiciones:** Ninguna.



- **Postcondiciones:** Ninguna

Estrategias

Finalidad:

- Generar el contenido del formulario de estrategias para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Activos:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - **Descripción:** Genera el formulario indicado por la variable acción para este formulario.
 - **Entrada:** Ninguna
 - **Salida:** Texto con formato HTML que representa el formulario.
 - **Precondiciones:** Ninguna
 - **Postcondiciones:** Ninguna

Categorías

Finalidad:

- Generar el contenido del formulario de categorías para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Cuentas:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - **Descripción:** Genera el formulario indicado por la variable acción para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario.



- **Precondiciones:** Ninguna
- **Postcondiciones:** Ninguna

Productos

Finalidad:

- Generar el contenido del formulario de productos para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Activos:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - **Descripción:** Genera el formulario indicado por la variable acción para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna

Categorización

Finalidad:

- Generar el contenido del formulario de categorización para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Cuentas:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - **Descripción:** Genera el formulario indicado por la variable acción para este formulario.
 - **Entrada:** Ninguna.



- **Salida:** Texto con formato HTML que representa el formulario.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

Allocation

Finalidad:

- Generar el contenido del formulario de Allocation para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Activos:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - **Descripción:** Genera el formulario indicado por la variable acción para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

Sub-Allocation

Finalidad:

- Generar el contenido del formulario de Sub-Allocation para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Cuentas:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - **Descripción:** Genera el formulario indicado por la variable acción para este formulario.



- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

Clasificación

Finalidad:

- Generar el contenido del formulario de Clasificación para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Activos:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - **Descripción:** Genera el formulario indicado por la variable acción para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

Sub-Clasificación

Finalidad:

- Generar el contenido del formulario de Sub-Clasificación para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Cuentas:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**



- **Descripción:** Genera el formulario indicado por la variable acción para este formulario
- **Entrada:** Ninguna
- **Salida:** Texto con formato HTML que representa el formulario
- **Precondiciones:** Ninguna
- **Postcondiciones:** Ninguna

Distribución

Finalidad:

- Generar el contenido del formulario de distribución para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario

Funciones:

- **Activos:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - **Descripción:** Genera el formulario indicado por la variable acción para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

Transacciones

Finalidad:

- Generar el contenido del formulario de transacciones para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Cuentas:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.



- **Generar Formulario:**

- **Descripción:** Genera el formulario indicado por la variable acción para este formulario.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

Aportaciones

Finalidad:

- Generar el contenido del formulario de aportaciones para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Activos:**

- **Descripción:** Constructora de la clase.
- **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
- **Entrada:** Ninguna.
- **Salida:** Ninguna.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Los atributos tienen sus valores por defecto.

- **Generar Formulario:**

- **Descripción:** Genera el formulario indicado por la variable acción para este formulario.
- **Entrada:** Ninguna.
- **Salida:** Texto con formato HTML que representa el formulario.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

Comisiones

Finalidad:

- Generar el contenido del formulario de comisiones para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Cuentas:**

- **Descripción:** Constructora de la clase.
- **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora
 - Configura el filtro para este formulario.
- **Entrada:** Ninguna.
- **Salida:** Ninguna.
- **Precondiciones:** Ninguna.



- **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - **Descripción:** Genera el formulario indicado por la variable acción para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

Traspasos

Finalidad:

- Generar el contenido del formulario de traspasos para sus distintas acciones posibles.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Formulario.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Activos:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:**
 - Inicializa los valores de los atributos de su clase predecesora.
 - Configura el filtro para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen sus valores por defecto.
- **Generar Formulario:**
 - **Descripción:** Genera el formulario indicado por la variable acción para este formulario.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

Gráficas

Finalidad:

- Generar de forma universal el contenido de cualquier gráfica y tabla que resuma la información del usuario

Atributos:

- **Formulario**: Constante de tipo texto que representa los distintos tipos de gráficas mediante un identificador único que permite diferenciarlos.
 - **Sub Formulario**: Constante de tipo texto que representa los distintos tipos de gráficas mediante un identificador único que permite diferenciarlos.
 - **Título**: Constante de tipo texto con el que se almacena el texto que se mostrará al usuario cuando abra la ventana de resumen seleccionada.
 - **Filtro**: Instancia de la clase formulario que permite generar un formulario de tipo filtro mediante el cual se pueden configurar ciertas características del formulario al que está asociado.

Funciones:

- **Gráficas:**
 - **Descripción:** Constructora de la clase.
 - **Funcionamiento:** Inicializa los valores de los atributos con sus valores por defecto.
 - **Entrada:**
 - Texto que representa el identificador único del formulario que se quiere construir.
 - Texto que representa el identificador único del subformulario que se quiere construir.
 - Texto que representa el título que se mostrará al usuario cuando se abra el formulario.
 - **Salida:** Ninguna.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Los atributos tienen los valores por defecto.
 - **Generar Formulario:**
 - **Descripción:** Genera el formulario donde se representan los datos especificados.
 - **Funcionamiento:** Delega su ejecución a las clases herederas.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario seleccionado.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

Balance

Finalidad:

- Generar las tablas y gráficas del formulario de Balance.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Gráficas
 - **Permisos Necesarios:** Usuario.

Funciones:

- **Generar Formulario:**
 - **Descripción:** Genera el formulario donde se representan los datos especificados.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario seleccionado.



- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

Gastos

Finalidad:

- Generar las tablas y gráficas del formulario de Gastos.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Gráficas.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Generar Formulario:**
 - **Descripción:** Genera el formulario donde se representan los datos especificados.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario seleccionado.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

Patrimonio

Finalidad:

- Generar las tablas y gráficas del formulario de Patrimonio.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Gráficas.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Generar Formulario:**
 - **Descripción:** Genera el formulario donde se representan los datos especificados.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario seleccionado.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

Evolución

Finalidad:

- Generar las tablas y gráficas del formulario de Evolución.

Características:

- **Herencia:** Hereda todos los atributos y funciones de la clase Gráficas.
- **Permisos Necesarios:** Usuario.

Funciones:

- **Generar Formulario:**
 - **Descripción:** Genera el formulario donde se representan los datos especificados.
 - **Entrada:** Ninguna.
 - **Salida:** Texto con formato HTML que representa el formulario seleccionado.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.



Filtro

Finalidad:

- Acotar los datos que se muestran en cualquier tabla en función de un conjunto de parámetros que puedan ser editados por el usuario.

Atributos:

- **Sección:** Constante de tipo texto que representa los distintos tipos de Filtro mediante un identificador único que permite diferenciarlos.
- **Sub-Sección:** Constante de tipo texto que representa los distintos tipos de Filtro mediante un identificador único que permite diferenciarlos.
- **Orden:** Constante de tipo texto que indica el campo en función del cual se ordenan los elementos del filtro.
- **Orden Tipo:** Constante de tipo texto que indica el criterio de ordenación de los elementos del filtro.
- **Campos:** Diccionario que contiene los campos a partir de los cuales se pueden filtrar.

Funciones:

- **Filtro:**
 - o **Descripción:** Constructora de la clase.
 - o **Funcionamiento:** Inicializa los valores de los atributos con sus valores por defecto.
 - o **Entrada:** Texto que representa el nombre de la sección y la subsección a la que representa el filtro.
 - o **Salida:** Ninguna.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Los atributos tienen los valores por defecto.
- **Añadir Campo:**
 - o **Descripción:** Permite añadir nuevos elementos al filtro.
 - o **Funcionamiento:** Comprueba que dicho elemento no exista previamente
 - o **Entrada:** Textos que representan el identificador, la etiqueta y el título del nuevo elemento del filtro.
 - o **Salida:** Ninguna.
 - o **Precondiciones:** Ninguna.
- **Generar Formulario:**
 - o **Descripción:** Genera el formulario donde se representan los datos para cada campo del filtro.
 - o **Entrada:** Ninguna.
 - o **Salida:** Texto con formato HTML que representa el formulario del Filtro.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Ninguna.
- **Capturar Campos:**
 - o **Descripción:** Recarga la página aplicando los cambios establecidos en el filtro.
 - o **Funcionamiento:**
 - Si redirigir es “Filtrar”: Aplica los cambios del filtro seleccionado.
 - Si redirigir es “Restablecer”: Elimina todos los filtros y aplica los cambios.
 - o **Entrada:**
 - Texto que permite detectar que funcionalidad se ha activado.
 - o **Salida:**
 - Booleano falso si no se debe recargar la pagina.
 - Booleano verdadero si se debe recargar la pagina.
 - o **Precondiciones:** Ninguna.
 - o **Postcondiciones:** Ninguna.



Librería SQL

Finalidad:

- Actuará como intermediario entre la base de datos y la aplicación suministrando los datos necesarios.

Funciones:

- **Conectar:**

- o **Descripción:** Establece la conexión con el servidor.
- o **Entrada:**
 - **Usuario:** Texto que contiene el nombre de usuario.
 - **Contraseña:** Texto que contiene la contraseña del usuario.
 - **Base de Datos:** Texto que contiene el nombre de la base de datos con la que se establece conexión.
- o **Salida:**
 - **Conexión:** Objeto SQL que representa la conexión activa.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Consultar:**

- o **Descripción:** Ejecuta una sentencia SQL del tipo SELECT.
- o **Entrada:**
 - **Conexión:** Objeto SQL que representa la conexión activa.
 - **Consulta:** Texto que contiene las especificaciones de la consulta.
- o **Salida:**
 - **Cursor:** Objeto SQL que representa los resultados de una consulta.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Actualizar:**

- o **Descripción:** Ejecuta una sentencia SQL del tipo UPDATE.
- o **Entrada:**
 - **Conexión:** Objeto SQL que representa la conexión activa.
 - **Consulta:** Texto que contiene las especificaciones de la consulta.
- o **Salida:** Ninguna.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Se actualizan filas correspondientes de la Base de Datos.

- **Existe:**

- o **Descripción:** Comprueba si existe un requerimiento específico en una tabla.
- o **Procedimiento:**
 - Ejecuta una consulta mediante la función Consultar de la librería SQL.
 - Los datos consultados no existen cuando el cursor está vacío.
- o **Entrada:**
 - **Conexión:** Objeto SQL que representa la conexión activa.
 - **Consulta:** Texto que contiene las especificaciones de la consulta.
- o **Salida:**
 - Booleano Verdadero si los datos consultados existen.
 - Booleano Falso si los datos consultados no existen.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.



- **Copiar Tabla:**

- **Descripción:** Dado el identificador de una tabla crea una copia exacta de la tabla de referencia que se utiliza como modelo.
- **Entrada:**
 - **Cursor:** Objeto SQL que representa los resultados de una consulta.
 - **Usuario:** Texto que contiene el nombre de usuario.
 - **Tabla:** Texto que contiene el identificador de una tabla.
- **Salida:** Ninguna.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Se genera la tabla indicada para el usuario indicado.

- **Crear Base de Datos:**

- **Descripción:** Crea una copia exacta de la Base de Datos de referencia para un nuevo usuario.
- **Funcionamiento:** Ejecuta la función Copiar tabla para cada tabla de la Base de Datos de referencia pasando como parámetros el Cursor de la nueva Base de datos y el nombre del usuario Objetivo.
- **Entrada:**
 - **Usuario Administrador:** Texto que contiene el nombre de usuario que genera la base de datos
 - **Contraseña:** Texto que contiene la contraseña del usuario que genera la base de datos.
 - **Usuario Objetivo:** Texto que contiene el nombre de usuario al que se le crea la nueva base de datos.
- **Salida:** Ninguna.
- **Precondiciones:** El usuario administrador tiene los permisos necesarios.
- **Postcondiciones:** Se genera nueva Base de Datos al usuario objetivo.

- **Crear Usuario:**

- **Descripción:** Crea un nuevo usuario y le asigna sus permisos.
- **Entrada:**
 - **Usuario:** Texto que contiene el nombre de usuario.
 - **Contraseña:** Texto que contiene la contraseña del usuario.
- **Salida:** Ninguna.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Se genera una nueva instancia en la tabla de Usuarios con todos los permisos asignados

- **Asignar Permisos:**

- **Descripción:** Modifica los permisos del usuario objetivo
- **Entrada:**
 - **Usuario Administrador:** Texto que contiene el nombre de usuario que modifica los permisos
 - **Contraseña:** Texto que contiene la contraseña del usuario que modifica los permisos
 - **Usuario Objetivo:** Texto que contiene el nombre de usuario al que se le modifican los permisos
 - **Permisos:** Texto que representa el nuevo perfil de permisos del usuario objetivo
- **Salida:** Ninguna
- **Precondiciones:** El usuario administrador tiene los permisos necesarios
- **Postcondiciones:** Se asignan al usuario objetivo los permisos indicados



- **Eliminar Consulta:**

- o **Descripción:** Ejecuta una sentencia SQL del tipo DELETE.
- o **Procedimiento:**
 - Ejecuta una consulta mediante la función Consultar de la librería SQL.
 - Los datos consultados no existen cuando el cursor este vacío.
- o **Entrada:**
 - **Usuario:** Texto que contiene el nombre de usuario.
 - **Contraseña:** Texto que contiene la contraseña del usuario.
 - **Consulta:** Texto que contiene las especificaciones de la consulta.
- o **Salida:** Ninguna.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Se eliminan las filas especificadas de la base de datos del usuario.

- **Eliminar Usuario:**

- o **Descripción:** Ejecuta una sentencia SQL del tipo DROP USER.
- o **Especificaciones:** Si el usuario eliminado tiene permisos de DROP USER puede eliminarse a sí mismo.
- o **Procedimiento:**
 - Ejecuta una consulta mediante la función Consultar de la librería SQL.
 - Los datos consultados no existen cuando el cursor este vacío.
- o **Entrada:**
 - **Usuario Administrador:** Texto que contiene el nombre de usuario que elimina al usuario objetivo.
 - **Contraseña:** Texto que contiene la contraseña del usuario que elimina al usuario objetivo.
 - **Usuario Objetivo:** Texto que contiene el nombre del usuario que va a ser eliminado.
- o **Salida:** Ninguna.
- o **Precondiciones:** El usuario administrador tiene los permisos necesarios.
- o **Postcondiciones:** Se eliminan las filas especificadas de la base de datos del usuario.



Librería HTML

Finalidad:

- Actuará como intermediario entre interfaz gráfica y la aplicación configurando objetos HTML visibles.

Funciones:

- **Título:**

- **Descripción:** Configura el título de una sección.
 - **Entrada:**
 - **ID:** Texto que contiene el identificador único de la etiqueta.
 - **Clase:** Texto que contiene el identificador que se utilizará para asignar los estilos CSS.
 - **Nivel:** Número entero que representa el nivel del título.
 - **Titulo:** Texto que representa la información que se mostrará en el título.
 - **Salida:** Texto que representa al código HTML.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

- **Párrafo:**

- **Descripción:** Configura un párrafo.
 - **Entrada:**
 - **ID:** Texto que contiene el identificador único de la etiqueta.
 - **Clase:** Texto que contiene el identificador que se utilizará para asignar los estilos CSS.
 - **Mensaje:** Texto que representa la información que se mostrará.
 - **Salida:** Texto que representa código HTML.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.

- **Representar Número:**

- **Descripción:** Configura un carácter numérico para que se represente correctamente.
 - **Especificaciones:** Permite configurar características como:
 - El color cuando el número es positivo o negativo.
 - El formato de los decimales.
 - El número de decimales mostrados.
 - **Entrada:**
 - **Número:** Número que se debe mostrar en formato int float o string indiferentemente.
 - **Tiene Color:** Booleano que indica si se debe asignar un color diferente en función del signo.
 - **Salida:** Texto que representa código HTML.
 - **Precondiciones:** Ninguna.
 - **Postcondiciones:** Ninguna.



- **Mensaje Error:**

- o **Descripción:** Configura un mensaje de error.
- o **Especificaciones:** Utilizado para depurar ciertas características del código.
- o **Entrada:** Texto que representa el mensaje de error.
- o **Salida:** Texto que representa código HTML
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna

- **Working:**

- o **Descripción:** Muestra por pantalla la imagen de Trabajo en progreso.
- o **Especificaciones:** Utilizado para indicar al usuario que dicha página no ha sido implementada.
- o **Entrada:** Ninguna.
- o **Salida:** Texto que representa código HTML.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Refrescar:**

- o **Descripción:** Genera un enlace de hipertexto que redirecciona el flujo de ejecución de la página.
- o **Procedimiento:** Se trata de un formulario con submit a la dirección indicada.
- o **Entrada:**
 - **ID:** Texto que contiene el identificador único de la etiqueta.
 - **Clase:** Texto que contiene el identificador que se utilizará para asignar los estilos CSS.
 - **Tipo:** tipo de redirección que se va a ejecutar.
 - **Función:** Nombre de la función que se ejecutará tras el Submit.
 - **Parámetros:** Lista con los parámetros que necesite la función.
 - **Título:** Texto que se mostrará en el enlace.
- o **Salida:** Texto que representa el código HTML.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Índice:**

- o **Descripción:** Genera una sección con un NAV que contiene un conjunto de enlaces de hipertexto dándoles un formato característico.
- o **Procedimiento:**
 - Utiliza la función de refrescar de la librería HTML.
 - Establece el parámetro función a "Redirigir".
- o **Entrada:**
 - **ID:** Texto que contiene el identificador único de la etiqueta.
 - **Clase:** Texto que contiene el identificador que se utilizará para asignar los estilos CSS.
 - **Lista:** Lista con el nombre de todos los elementos que formarán parte del índice.
- o **Salida:** Texto que representa el código HTML.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.



- **Cabecera Tabla:**

- o **Descripción:** Configura la cabecera de una tabla.
- o **Entrada:**
 - **ID:** Texto que contiene el identificador único de la etiqueta.
 - **Clase:** Texto que contiene el identificador que se utilizará para asignar los estilos CSS.
 - **Título:** Texto que representa la información que se mostrará en el título.
- o **Salida:** Texto que representa el código HTML.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Cabecera de Formulario:**

- o **Descripción:** Genera la cabecera de un formulario.
- o **Entrada:**
 - **ID:** Texto que contiene el identificador único de la etiqueta.
 - **Clase:** Texto que contiene el identificador que se utilizará para asignar los estilos CSS.
 - **Acción:** Nombre de la función a la que efectuará la redirección.
- o **Salida:** Texto que representa el código HTML.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Input de Formulario:**

- o **Descripción:** Genera un input para un formulario.
- o **Entrada:**
 - **ID:** Texto que contiene el identificador único de la etiqueta.
 - **Clase:** Texto que contiene el identificador que se utilizará para asignar los estilos CSS.
 - **Etiqueta:** Texto que se muestra en la etiqueta.
 - **Tipo:** Tipo de dato que admite como input.
 - **Place Holder:** Texto que se muestra por defecto cuando el usuario no ha introducido ningún input.
- o **Salida:** Texto que representa código HTML.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Gráfico Simple:**

- o **Descripción:** Genera un canvas y configura el Script para que se genere un gráfico con una única serie.
- o **Entrada:**
 - **ID:** Texto que contiene el identificador único de la etiqueta.
 - **Cabecera:** Lista con las cabeceras de cada serie de datos que se debe mostrar.
 - **Datos:** Lista de listas con los datos que se deben mostrar para cada serie de datos.
 - **Bordes:** Lista con los colores para los bordes de cada serie de datos.
 - **Fondos:** Lista con los colores para los fondos de cada serie de datos.
 - **Tipo:** Tipo de representación de los datos: Tarta, Lineal, Barras, Donut, etc.
- o **Salida:** Texto que representa código HTML y JavaScript.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.



- **Gráfico Agrupado:**

- o **Descripción:** Genera un canvas y configura el Script para que se genere un gráfico compuesto por más de una serie de datos.
- o **Entrada:**
 - **ID:** Texto que contiene el identificador único de la etiqueta.
 - **Cabecera:** Lista con las cabeceras de cada serie de datos que se debe mostrar.
 - **Datos:** Lista de listas con los datos que se deben mostrar para cada serie de datos.
 - **Bordes:** Lista con los colores para los bordes de cada serie de datos.
 - **Fondos:** Lista con los colores para los fondos de cada serie de datos.
 - **Tipo:** Tipo de representación de los datos: Tarta, Lineal, Barras, Donut, etc.
- o **Salida:** Texto que representa código HTML y JavaScript.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

Librería Ficheros

Finalidad:

- Permite leer archivos CSV, validar el formato de los datos e insertar la información en la base de datos.

Funciones:

- ### - Importar Datos CSV:

- **Descripción:** Permite importar datos a la Base de Datos de un usuario directamente desde un fichero CSV.

- ## ○ Funcionamiento

- Carga el fichero en una ruta conocida.
 - Identifica el tipo de fichero. ([Aportaciones](#), [Comisiones](#), [Gastos](#), [Traspasos](#))
 - Lee los datos del fichero.
 - Comprueba que todos los datos cumplen las restricciones necesarias.
 - Carga los datos en la Base de Datos del usuario.

- ### ○ Entrada:

- **Fichero:** Archivo con formato CSV.
 - **Nombre Fichero:** texto que representa el nombre del Fichero.

- #### ○ **Salida:** Ninguna.

- Precondiciones: Ninguna.

- #### ○ Postcondiciones:

- Si los datos cumplen todas las restricciones se cargan en la Base de Datos
 - Se genera un mensaje de error con todas aquellas filas que no cumplen alguna restricción indicando cuál es el requerimiento que no se cumplió.



Librería Scraper

Finalidad:

- Permite obtener información específica sobre los activos mediante la técnica de Web Scraping.

Atributos:

- Se ha implementado un diccionario que permite almacenar las cotizaciones distinguiéndolas por producto y fecha con la finalidad de saltarse el paso de consultar en la Web si dicho dato ya ha sido obtenido para la misma fecha y el mismo producto. Este atributo se restablece cada cierto periodo de tiempo.

El coste computacional del Web Scraping es el más alto en toda la aplicación con bastante diferencia.

Funciones:

- **Cotización Actual:**

- o **Descripción:** Obtiene el dato de la cotización actual de un producto consultando en yahoo Finance
- o **Funcionamiento:** "https://es.finance.yahoo.com/quote/" + pProducto + "?p=" + pProducto
- o **Entrada:**
 - **Producto:** Texto que representa el nombre del producto.
- o **Salida:** Número decimal que representa última cotización registrada en yahoo Finance.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Serie Cotizaciones:**

- o **Descripción:** Obtiene una lista con todas las cotizaciones históricas registradas entre dos fechas en yahoo Finance.
- o **Funcionamiento:** Utiliza la función Request sobre la Api Alphavantage
- o **Entrada:**
 - **Producto:** Texto que representa el nombre del producto.
 - **Fecha Inicio:** Fecha del primer dato que se recupera.
 - **Fecha Fin:** Fecha del último dato que se recupera.
- o **Salida:** Lista de tuplas fecha, cotización registrada en yahoo Finance
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Cotización Fecha:**

- o **Descripción:** Obtiene una lista con la cotización histórica registradas para una fecha concreta en yahoo Finance.
- o **Funcionamiento:** Se delega el procedimiento a la función Serie Cotizaciones de la Librería Scraper utilizando como fecha de fin el día siguiente al de la fecha de inicio requerida.
- o **Entrada:**
 - **Producto:** Texto que representa el nombre del producto.
 - **Fecha Inicio:** Fecha del primer dato que se recupera
- o **Salida:** Tuplas fecha, cotización registrada en yahoo Finance
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.



Librería Utilidades

Funciones:

- **Color Aleatorio:**

- **Descripción:** Genera aleatoriamente un color.
- **Entrada:** Ninguna.
- **Salida:** Tupla con tres números enteros comprendidos entre cero y doscientos cincuenta y cinco.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Degrado:**

- **Descripción:** Genera el siguiente color en la escala RGB.
- **Funcionamiento:** Degrado un color consiste en incrementar el valor de R G o B en tantas unidades como indica la velocidad, si se sobrepasa el límite de 255 de una propiedad se vuelve cero y se itera la siguiente.
- **Entrada:**
 - **Color:** Color en formato RGB del que se parte.
 - **Velocidad:** Número entero que representa la cantidad que se degrada.
- **Salida:** Color en formato RGB aplicando el degradado.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Color De Hexadecimal A RGB:**

- **Descripción:** Convierte el formato del color de Hexadecimal a RGB.
- **Entrada:** Color en formato Hexadecimal.
- **Salida:** Color en formato RGB.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.

- **Color De RGB A Hexadecimal:**

- **Descripción:** Convierte el formato del color de RGB a Hexadecimal.
- **Entrada:** Color en formato RGB.
- **Salida:** Color en formato Hexadecimal.
- **Precondiciones:** Ninguna.
- **Postcondiciones:** Ninguna.



- **Comprobar Fecha:**

- o **Descripción:** Comprueba si una fecha es válida
- o **Entrada:** Fecha en formato día mes año.
- o **Salida:**
 - Booleano Verdadero si la fecha es válida
 - Booleano Falso si la fecha no es válida
- o **Precondiciones:** Ninguna
- o **Postcondiciones:** Ninguna

- **Ajustar Fecha:**

- o **Descripción:** Comprueba el formato de una fecha y lo convierte al formato de la librería DATETIME.
- o **Entrada:** Fecha en cualquier formato. ([incluso texto](#))
- o **Salida:** Fecha en formato DATETIME.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Depurar:**

- o **Descripción:** Dada una cadena de texto elimina todos los caracteres sospechosos.
- o **Finalidad:** Evitar SQL Inyección en los inputs de los formularios.
- o **Entrada:** Cadena de texto que se pretende validar.
- o **Salida:** Cadena de texto depurada.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

- **Generar Sal:**

- o **Descripción:** Genera una cadena de texto tipo Hash en función del DATETIME del sistema.
- o **Entrada:** Ninguna.
- o **Salida:** Cadena de texto Hash.
- o **Precondiciones:** Ninguna.
- o **Postcondiciones:** Ninguna.

Diagramas de Flujo

Este apartado de la documentación está destinado a facilitar el entendimiento del resto del documento. De este modo se incluirán las definiciones de los términos clave para su correcto entendimiento, así como las referencias de las que se ha obtenido la información para su elaboración.

Gestión de usuarios

Registrarse

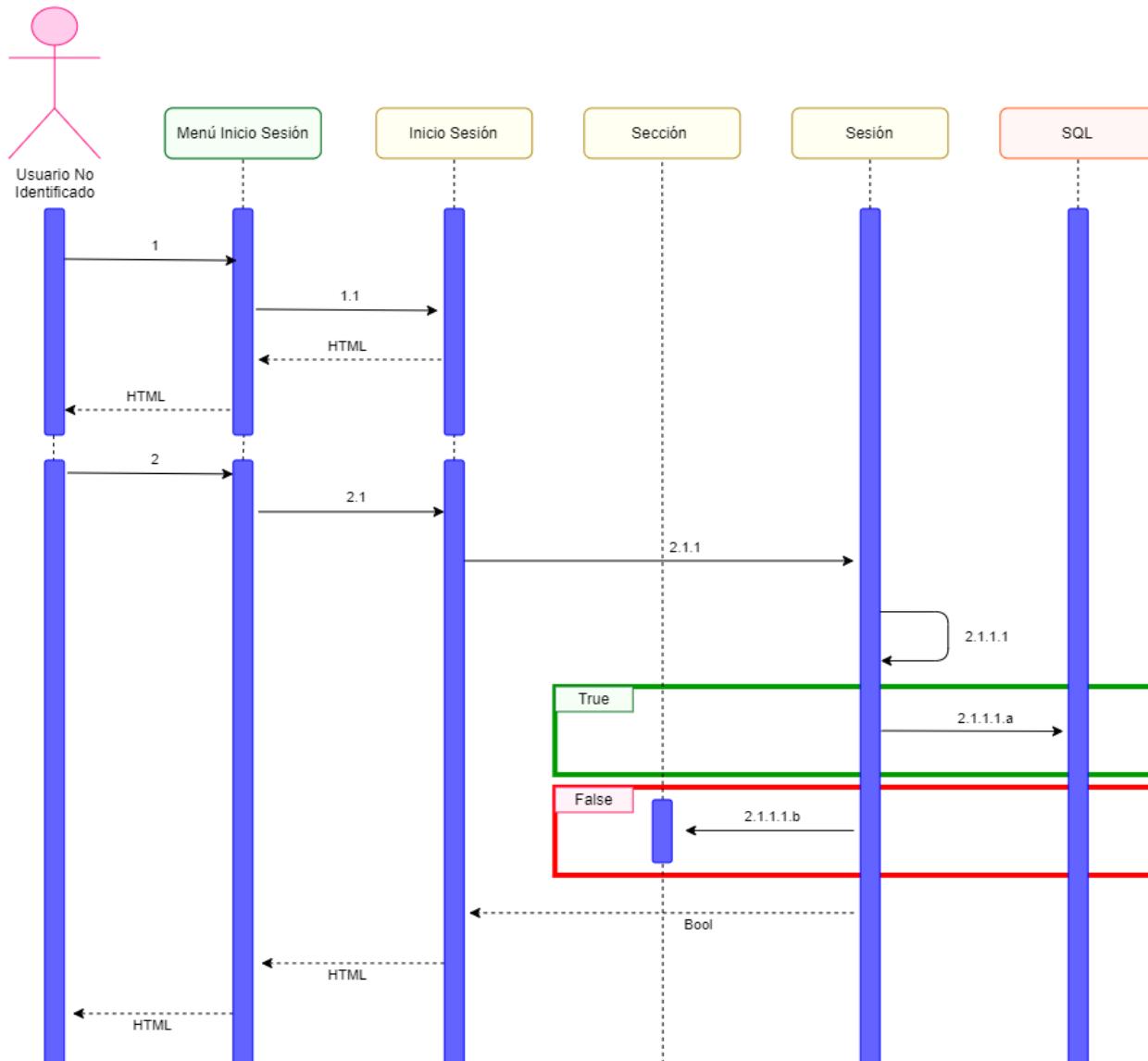


Ilustración 2 Diagrama de flujo – Registrarse

- 1 El Usuario No Identificado accede al formulario de registro.
 - 1.1. Se ejecuta la función Generar Contenido de la clase Sección para la subclase Registro.
- 2 El Usuario No Identificado rellena el formulario de registro y pulsa el botón de aceptar.
 - 2.1. Se ejecuta la función Generar Contenido de la clase Sección para la subclase Registro.
 - 2.1.1. Se ejecuta la función Añadir Admisión de la clase Sesión.
 - 2.1.1.1. Se ejecuta la función Comprobar admisión de la clase Sesión.
 - 2.1.1.1.a. En caso verdadero ejecuta la función Añadir Admisión de la librería SQL.
 - 2.1.1.1.b. En caso negativo se ejecuta la función Reportar Error de la clase Sesión.

Identificarse

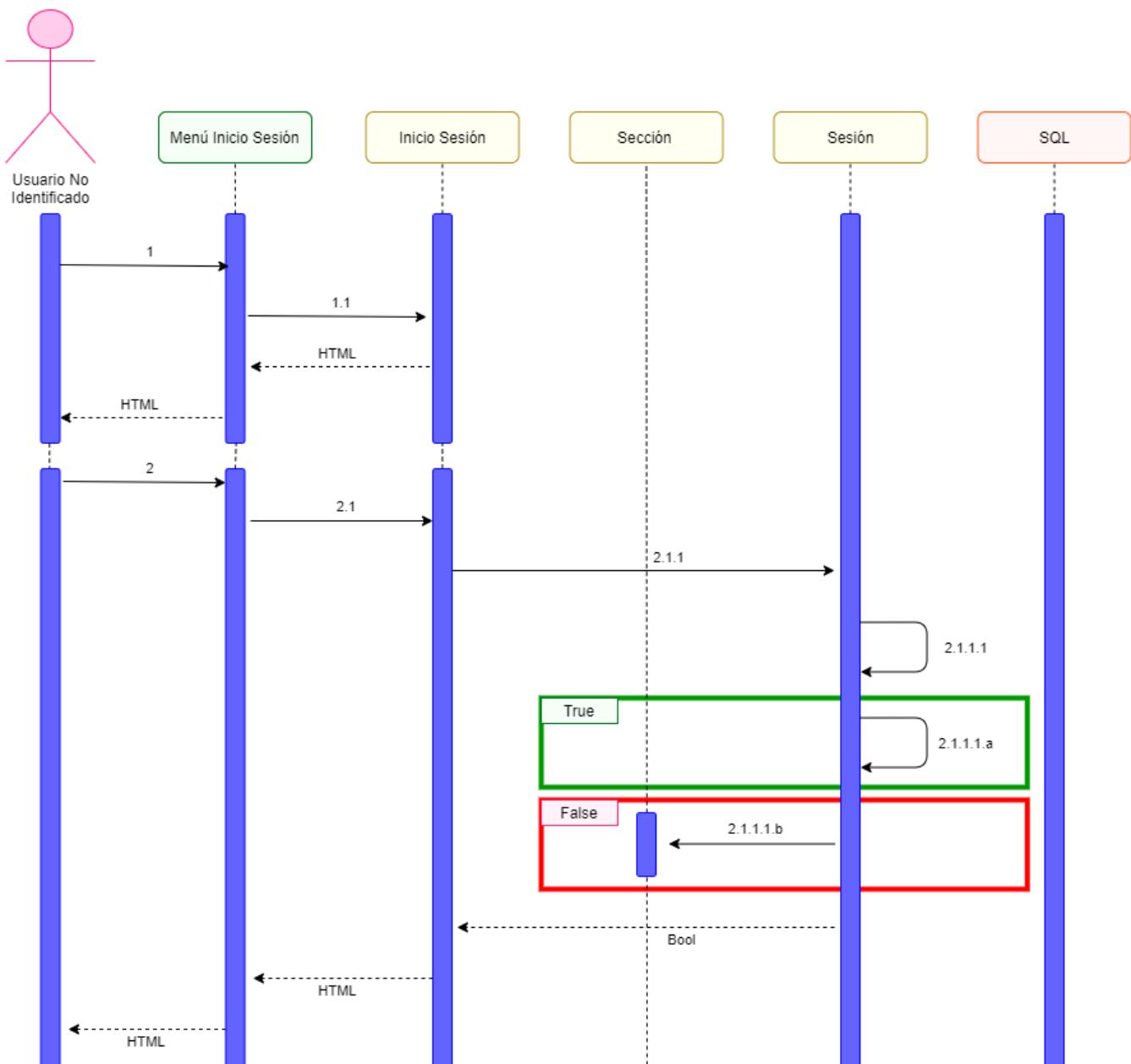


Ilustración 3 Diagrama de flujo – Identificarse

- 1 El Usuario No Identificado accede al formulario de registro.
 - 1.1. Se ejecuta la función Generar Contenido de la clase Sección para la subclase Inicio Sesión.
- 2 El Usuario No Identificado rellena el formulario de registro y pulsa el botón de aceptar.
 - 2.1. Se ejecuta la función Generar Contenido de la clase Sección para la subclase Inicio Sesión.
 - 2.1.1. Se ejecuta la función Iniciar Sesión de la clase Sesión.
 - 2.1.1.1. Se ejecuta la función Comprobar Credenciales de Sesión de la clase Sesión.
 - 2.1.1.1.a En caso verdadero se actualizan los atributos Usuario, Permisos y Contraseña.
 - 2.1.1.1.b En caso negativo se ejecuta la función Reportar Error de la clase Sección.

Funciones de apoyo

Generar Contenido

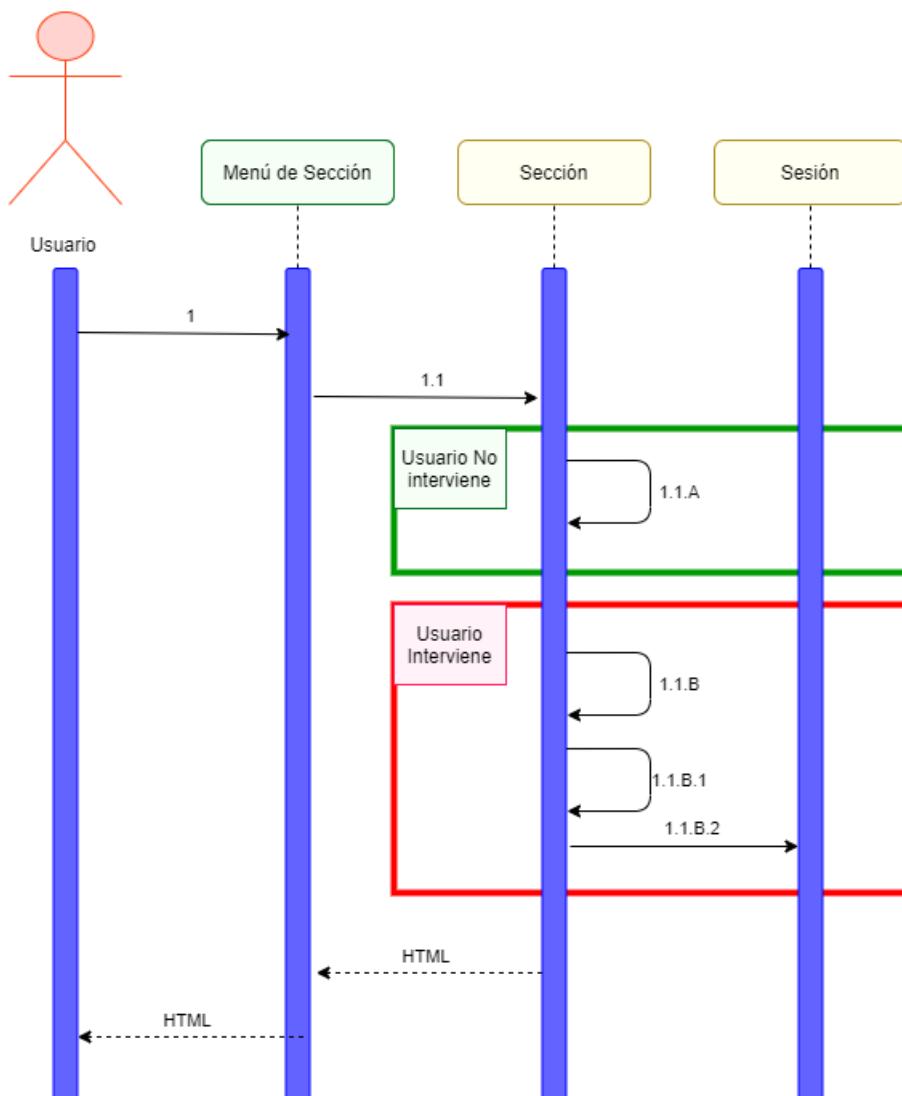


Ilustración 4 Diagrama de flujo – Generar Contenido

- 3 El Usuario No Identificado accede al formulario deseado.

3.1. Se ejecuta la función Generar Contenido de la clase sección.

1.1.A Si el usuario no ha completado el formulario de la sección Seleccionada.

1.1.A.1 Se ejecuta la función Generar Formulario de la subclase sección Seleccionada.

1.1.B Si el usuario ha completado el formulario de la sección Seleccionada.

1.1.B.1 Se ejecuta la función Capturar Campos de la subclase sección Seleccionada.

1.1.B.1.1 Se obtienen los datos de los campos del formulario.

1.1.B.1.2 Se ejecuta la función de la clase Sesión Asociada a la sección.