

Anexo III: Especificación de diseño

# APLICACIÓN PARA LA GESTIÓN ECONÓMICA PERSONAL



**VNiVERSiDAD  
D SALAMANCA**

CAMPUS DE EXCELENCIA INTERNACIONAL

**Autor:**

Victor Fagúndez Poyo

**Tutor:**

Dionisio Tomás Rodríguez Barrios

Fecha de Adjudicación: Febrero 2020

Fecha de Presentación: Julio 2020

**Grado en Ingeniería Informática en Sistemas de Información**

**Escuela Politécnica Superior de Zamora**



### Historial de Revisiones

<b><i>Versión</i></b>	<b>Fecha</b>	<b>Descripción</b>
<i>1.0.0</i>	05.04.20	Versión inicial del documento
<i>1.0.1</i>	08.06.20	Añadidos diseños de versión móvil

CONTENIDO	
1.	Introducción..... 7
2.	Diseño de datos..... 8
2.1.	Diagrama entidad Relación ..... 8
2.2.	Tablas de la base de datos..... 9
2.2.1.	Tabla user ..... 9
2.2.2.	Tabla Usuario_Cuenta ..... 10
2.2.3.	Tabla Cuenta..... 11
2.2.4.	Movimiento ..... 12
2.2.5.	Usuario_Empresa ..... 13
2.2.6.	Empresa ..... 13
2.2.7.	Sucursal ..... 14
2.2.8.	Producto..... 14
2.2.9.	Empresa_Producto..... 15
2.2.10.	Movimiento_producto ..... 15
2.2.11.	Categoría..... 16
2.2.12.	Categoría_Movimiento..... 16
3.	Diseño de la interfaz..... 17
3.1.	Guía de estilo ..... 17
3.2.	Estructura de la Interfaz..... 18
3.2.1.	La pantalla de inicio ..... 18
3.2.2.	La pantalla de cuentas ..... 19
3.2.3.	La pantalla de movimientos..... 20
3.2.4.	Otras Vistas ..... 20
3.3.	División en componentes..... 23
3.3.1.	Componentes de navegación ..... 23
3.3.2.	Componentes de Información..... 24
3.3.3.	Componentes de acceso a Información ..... 24
3.3.4.	Componente de ayuda al usuario..... 25
3.3.5.	Componentes de creación y modificación..... 25
4.	Arquitectura del software ..... 27
5.	Abreviaturas ..... 29
6.	Bibliografía..... 30

## TABLA DE ILUSTRACIONES

Ilustración 1. Diagrama entidad relación del sistema.....	8
Ilustración 2. Tabla User .....	9
Ilustración 3. Tabla Usuario_Cuenta .....	10
Ilustración 4. Tabla Cuenta.....	11
Ilustración 5. Tabla Movimiento .....	12
Ilustración 6. Tabla Usuario_Empresa .....	13
Ilustración 7.Tabla Empresa .....	13
Ilustración 8. Tabla Sucursal .....	14
Ilustración 9.Tabla Producto.....	14
Ilustración 10. Tabla Empresa_Producto.....	15
Ilustración 11. Tabla Movimiento_Producto.....	15
Ilustración 12. Tabla Categoria .....	16
Ilustración 13. Tabla Categoria_Movimiento.....	16
Ilustración 14.Material-UI. ....	17
Ilustración 15. Pantalla de Inicio .....	18
Ilustración 16.Pantalla de Inicio Móvil .....	18
Ilustración 17. Pantalla de Inicio Móvil2 .....	19
Ilustración 18. Pantalla de Cuentas .....	19
Ilustración 19. Pantalla de Cuentas versión móvil .....	20
Ilustración 20. Pantalla de Movimientos .....	20
Ilustración 21. Vista de Usuario .....	21
Ilustración 22. Pantalla de Usuario versión móvil.....	21
Ilustración 23. Vista de Registrarse .....	22
Ilustración 24.Vista de Inicio de Sesión .....	22
Ilustración 25. Componentes de navegación .....	23
Ilustración 26. Subcomponentes Barra de navegación .....	24
Ilustración 27. Componentes de Información .....	24
Ilustración 28. Componentes de acceso a información .....	24
Ilustración 29. Componente de ayuda al usuario 1 .....	25
Ilustración 30. Componente de ayuda al usuario 2.....	25
Ilustración 31. Componente de Modificación .....	26
Ilustración 32. Componente de creación .....	26

Ilustración 33. Framework React JS .....	27
Ilustración 34.Framework Yii2 .....	28

**TABLA DE TABLAS**

**No se encuentran elementos de tabla de ilustraciones.**

## 1. INTRODUCCIÓN

En el presente anexo se documentará el diseño y proceso de creación del modelo de datos asociado al proyecto. Para ello nos apoyaremos en la documentación vista en el anexo de especificación de requisitos al cual nos ajustaremos para el diseño y elaboración de la base de datos de la aplicación.

Además, se explicará el diseño de la interfaz de usuario de la aplicación con la que interactuarán los usuarios finales.

## 2. DISEÑO DE DATOS

En primer lugar, procedemos a la creación del modelo de datos de la aplicación, la cual almacenará la información relevante del sistema.

### 2.1. DIAGRAMA ENTIDAD RELACIÓN

El primer paso para la creación de la base de datos es la creación del modelo entidad relación de esta (ilustración 1). En este diagrama se muestran las diferentes entidades del sistema con sus correspondientes relaciones y atributos.

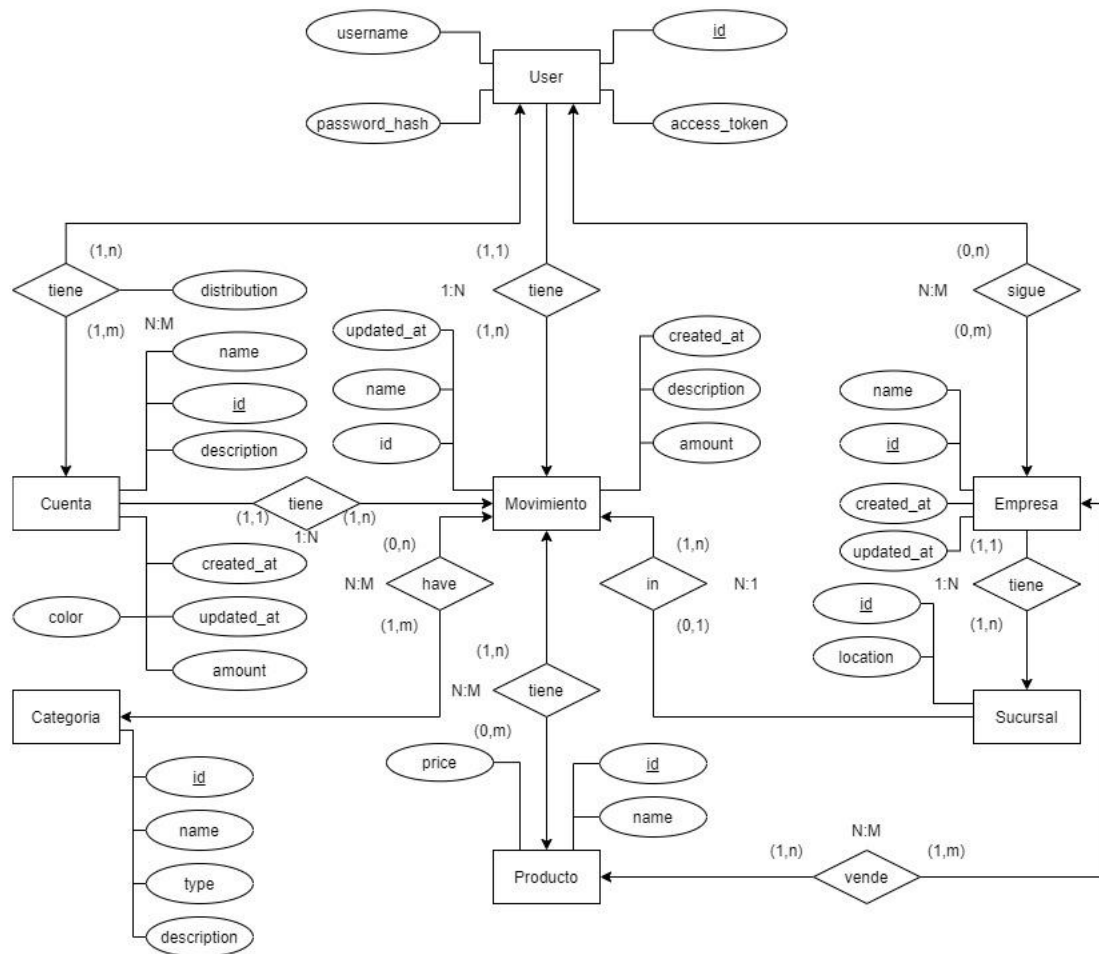


Ilustración 1. Diagrama entidad relación del sistema





## 2.2. TABLAS DE LA BASE DE DATOS

A partir de la normalización de las entidades y relaciones del diagrama entidad-relación hemos obtenido las siguientes 12 tablas:

### 2.2.1. TABLA USER

Esta es la tabla encargada de almacenar la información de los usuarios registrados en el sistema, cuyos atributos son los siguientes:


#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
<input type="checkbox"/> 1	<b>id</b> 	int(11)			No	Ninguna		AUTO_INCREMENT
<input type="checkbox"/> 2	<b>username</b> 	varchar(255)	utf8_unicode_ci		No	Ninguna		
<input type="checkbox"/> 3	<b>password_hash</b>	varchar(255)	utf8_unicode_ci		No	Ninguna		
<input type="checkbox"/> 4	<b>access_token</b>	varchar(32)	utf8_unicode_ci		No	Ninguna		
<input type="checkbox"/> 5	<b>distribution</b>	int(11)			No	100		

*Ilustración 2. Tabla User*

- **Id:** El identificador único de cada usuario y clave primaria de la tabla.
- **Username:** El nombre del usuario
- **Password\_hash:** El hash de la contraseña del usuario obtenido a partir de las funciones de seguridad del Framework PHP Yii2, el cual usa el algoritmo Blowfish.
- **Access\_token:** La clave de acceso para las peticiones API REST.
- **Distribution:** El porcentaje que le queda al usuario por distribuir entre sus cuentas, tiene un valor predeterminado de 100.

### 2.2.2. TABLA USUARIO\_CUENTA

Esta es la tabla encargada de asociar a los usuarios y las cuentas que les pertenecen. Surge de la relación N:M entre las entidades Usuario y Cuenta. Esta relación permite que un usuario pueda tener varias cuentas y que una cuenta pueda tener varios usuarios asociados de forma que se establecen las bases para el cumplimiento del requisito funcional 11 perteneciente a la fase 1 del proyecto.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	<b>id</b> 	int(11)			No	Ninguna		AUTO_INCREMENT
2	<b>idUsuario</b>	int(11)			No	Ninguna	El identificador de usuario asociado a la cuenta	
3	<b>idCuenta</b>	int(11)			No	Ninguna	El identificador de la cuenta asociada al usuario	
4	<b>distribution</b>	int(11)			No	Ninguna	Porcentaje de la cantidad introducida por cada propietario de la cuenta	

*Ilustración 3. Tabla Usuario\_Cuenta*

Esta tabla presenta los siguientes atributos:

- **Id:** El identificador de la relación, auto incremental y clave primaria de la tabla.
- **idUsuario:** El identificador del usuario al que pertenece la cuenta
- **idCuenta:** El identificador de la cuenta perteneciente al usuario
- **Distribution:** El % de ingresos que asocia el usuario a esa cuenta. Esto permite que cada uno de los usuarios a los que pertenece una cuenta pueda aportar una cantidad diferente si así lo desea.

### 2.2.3. TABLA CUENTA

Esta tabla es la encargada de almacenar las diferentes cuentas que se creen los usuarios para gestionar sus gastos e ingresos. Cuenta con los siguientes atributos:


#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
<input type="checkbox"/> 1	<u>id</u> 	int(11)			No	Ninguna	El identificador de la cuenta	AUTO_INCREMENT
<input type="checkbox"/> 2	<u>name</u>	varchar(40)	utf8mb4_general_ci		No	Ninguna	El nombre que el usuario pone a la cuenta	
<input type="checkbox"/> 3	<u>description</u>	varchar(140)	utf8mb4_general_ci		No	Ninguna		
<input type="checkbox"/> 4	<u>amount</u>	float			No	Ninguna	La cantidad total disponible en esa cuenta	
<input type="checkbox"/> 5	<u>distribution</u>	int(11)			No	Ninguna		
<input type="checkbox"/> 6	<u>color</u>	int(11)			No	Ninguna		
<input type="checkbox"/> 7	<u>icon</u>	int(11)			No	Ninguna		
<input type="checkbox"/> 8	<u>created_at</u>	int(11)			No	Ninguna	Fecha de creación de la cuenta	
<input type="checkbox"/> 9	<u>updated_at</u>	int(11)			No	Ninguna	Fecha de modificación de la cuenta	

Ilustración 4. Tabla Cuenta

- **Id:** El identificador único de la cuenta y clave primaria
- **Name:** El nombre que el usuario le da a la cuenta
- **Description:** La descripción que el usuario le da a la cuenta
- **Amount:** La cantidad total disponible en esa cuenta
- **Distribution:** El % de los ingresos del usuario que se asocian a esta cuenta. Se incluye aquí también para los casos en que el usuario al que pertenezca la cuenta sea único.
- **Color:** Un valor numérico entre 1 y 7 que representa el color asociado a la cuenta. Se ha elegido esta forma de guardar el color en base a que se cumplan criterios de accesibilidad y visibilidad de los componentes.
- **Created\_at:** Fecha de creación de la cuenta
- **Updated\_at:** Fecha de modificación de la cuenta

### 2.2.4. MOVIMIENTO

Esta tabla representa a los ingresos y gastos que pueda tener un determinado usuario. Esta tabla cuenta con los siguientes atributos:

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	<u>id</u> 🗝️	int(11)			No	Ninguna		AUTO_INCREMENT
2	<u>description</u>	varchar(100)	utf8mb4_general_ci		No	Ninguna	Un resumen de lo que consiste la transaccion	
3	<u>name</u>	varchar(40)	utf8mb4_general_ci		No	Ninguna	El nombre que el usuario da a la transaccion	
4	<u>amount</u>	float			No	Ninguna	La cantidad asociada a la transacción	
5	<u>created_at</u>	int(11)			No	Ninguna	La fecha en la que se ha creado la transaccion	
6	<u>updated_at</u>	int(11)			No	Ninguna	La fecha de modificación del movimiento	
7	<u>idCuenta</u>	int(11)			No	Ninguna	El identificador de la cuenta asociada a este movimiento	
8	<u>idUsuario</u>	int(11)			No	Ninguna		

*Ilustración 5. Tabla Movimiento*

- **Id:** El identificador único del movimiento, auto incremental y clave primaria de la tabla
- **Description:** Un resumen sobre la transacción
- **Name:** El nombre que el usuario le da a la transacción
- **Amount:** La cantidad asociada a la transacción
- **Created\_at:** La fecha en la que se ha creado la transacción
- **Updated\_at:** La fecha de modificación del movimiento
- **idCuenta:** El identificador de cuenta asociada a este movimiento
- **idUsuario:** El identificador del usuario que ha creado este movimiento

### 2.2.5. USUARIO\_EMPRESA

Esta tabla representa la relación N:M entre las entidades User y Empresa que permiten almacenar las empresas a las que sigue un determinado usuario. Esta tabla cuenta con los siguientes atributos:


#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	<u>id</u> 	int(11)			No	Ninguna		AUTO_INCREMENT
2	<u>idUsuario</u>	int(11)			No	Ninguna	El identificador del usuario que sigue a la empresa	
3	<u>idEmpresa</u>	int(11)			No	Ninguna	El identificador de la empresa a la que sigue el usuario	

Ilustración 6. Tabla Usuario\_Empresa

- **Id:** la clave primaria de la tabla
- **idUsuario:** El identificador del usuario
- **idEmpresa:** El identificador de la empresa a la que sigue el usuario

### 2.2.6. EMPRESA

Esta tabla representa a la entidad Empresa, la cual, representa a los diferentes comercios que pueden operar en la aplicación en base al Objetivo “OBJ-04 Gestionar Empresa” y sus correspondientes requisitos de información, pertenecientes a la fase 1 del desarrollo de la aplicación. Esta tabla tiene los siguientes atributos:



#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	<u>id</u> 	int(11)			No	Ninguna	El identificador de la empresa	AUTO_INCREMENT
2	<u>name</u>	varchar(40)	utf8mb4_general_ci		No	Ninguna	El nombre de la empresa	
3	<u>created_at</u>	int(11)			No	Ninguna	Fecha de creación de la tupla empresa	
4	<u>updated_at</u>	int(11)			No	Ninguna	Fecha de modificación de los valores de la tupla empresa	

Ilustración 7. Tabla Empresa

- **Id:** identificador único para cada empresa y clave primaria de la tabla
- **Name:** El nombre de la empresa
- **Created\_at:** Fecha de creación de los datos de la empresa
- **Updated\_at:** Fecha de modificación de los datos de la empresa

### 2.2.7. SUCURSAL

Esta tabla representa a la entidad Sucursal del modelo entidad relación y cuenta con los siguientes atributos:


#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	<u>id</u> 	int(11)			No	Ninguna	El identificador de la sucursal	AUTO_INCREMENT
2	<u>idEmpresa</u>	int(11)			No	Ninguna	El identificador de la empresa asociada	
3	<u>location</u>	varchar(40)	utf8mb4_general_ci		No	Ninguna	La localización de la sucursal	

*Ilustración 8. Tabla Sucursal*

- **Id:** El identificador único para cada sucursal
- **idEmpresa:** El identificador de la empresa asociada a esa sucursal
- **location:** La localización de la sucursal

### 2.2.8. PRODUCTO

Esta tabla guarda información sobre los productos que las empresas puedan tener y los usuarios puedan comprar. Esta tabla tiene los siguientes atributos para guardar la información básica del producto:


#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	<u>id</u> 	int(11)			No	Ninguna	El identificador del producto	AUTO_INCREMENT
2	<u>name</u>	int(11)			No	Ninguna	El nombre del producto	
3	<u>price</u>	int(11)			No	Ninguna	El precio del producto	

*Ilustración 9. Tabla Producto*

- **Id:** El identificador único del producto
- **Name:** El nombre del producto
- **Price:** El precio del producto

### 2.2.9. EMPRESA\_PRODUCTO

Esta tabla surge a partir de la relación N:M del modelo entidad relación, a partir de la cual, un producto puede estar en varias empresas y una empresa puede tener varios productos. Para esta relación, la tabla tiene los siguientes atributos:


#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	<b>id</b> 	int(11)			No	Ninguna		AUTO_INCREMENT
2	<b>idEmpresa</b>	int(11)			No	Ninguna	El identificador de la empresa que vende el producto	
3	<b>idProducto</b>	int(11)			No	Ninguna	El identificador del producto vendido por la empresa	

*Ilustración 10. Tabla Empresa\_Producto*

- **Id:** El identificador de la relación
- **idEmpresa:** El identificador de la empresa a la que pertenece el producto
- **idProducto:** El identificador del producto

### 2.2.10. MOVIMIENTO\_PRODUCTO

Esta tabla surge a partir de la relación N:M del modelo entidad relación, a partir de la cual, un movimiento puede tener varios productos y un producto puede estar en varios movimientos. En base a esto esta tabla tiene los siguientes atributos:

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	<b>id</b> 	int(11)			No	Ninguna		AUTO_INCREMENT
2	<b>idMovimiento</b>	int(11)			No	Ninguna	El identificador del movimiento que tiene el producto	
3	<b>idProducto</b>	int(11)			No	Ninguna	El identificador del producto que tiene el movimiento	

*Ilustración 11. Tabla Movimiento\_Producto*

- **Id:** El identificador de la relación
- **idMovimiento:** El identificador del movimiento
- **idProducto:** El identificador del producto perteneciente al movimiento

### 2.2.11. CATEGORIA

Esta tabla surge a partir de la entidad categoría la cual permite a los usuarios agrupar sus movimientos en secciones que le permitan organizarse mejor. Para ello esta tabla tiene los siguientes atributos:


#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	<b>id</b> 	int(11)			No	Ninguna	El identificador de la categoría de movimiento	AUTO_INCREMENT
2	<b>name</b>	varchar(40)	utf8mb4_general_ci		No	Ninguna	El nombre de la categoría de movimiento	
3	<b>description</b>	varchar(100)	utf8mb4_general_ci		No	Ninguna	La descripción de la categoría de movimiento	
4	<b>type</b>	tinyint(1)			No	0	Categoría de tipo ingreso (0) Categoría de tipo gasto (1)	

Ilustración 12. Tabla Categoría

- **Id:** El identificador único de la categoría
- **Name:** El nombre de la categoría
- **Description:** La descripción de la categoría del movimiento
- **Type:** Categoría de tipo de movimiento: ingreso o gasto

### 2.2.12. CATEGORIA\_MOVIMIENTO

Esta tabla surge a partir de la relación N:M entre las entidades Categoría y Movimiento dado que un movimiento puede tener varias categorías y una categoría varios movimientos asociados. Para ello esta tabla cuenta con los siguientes atributos:


#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	<b>id</b> 	int(11)			No	Ninguna		AUTO_INCREMENT
2	<b>idCategoría</b>	int(11)			No	Ninguna		
3	<b>idMovimiento</b>	int(11)			No	Ninguna		

Ilustración 13. Tabla Categoría\_Movimiento

- **Id:** El identificador único de la relación
- **idCategoría:** El identificador de la categoría relacionada
- **idMovimiento:** El identificador del movimiento relacionado con la categoría



### 3. DISEÑO DE LA INTERFAZ

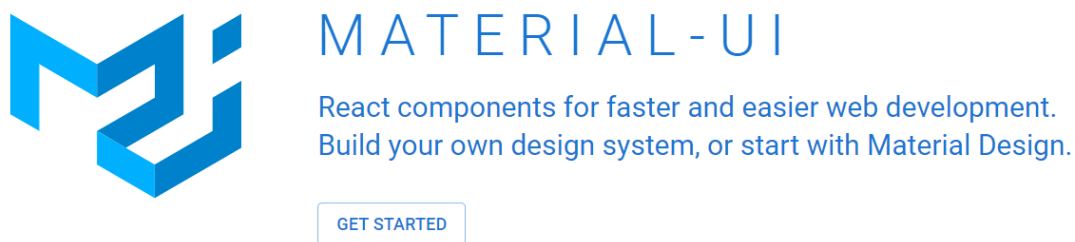
Este sistema es una aplicación de finanzas personales que registra los datos de gastos e ingresos, permitiendo gestionarlos generando informes gráficos y agrupándolos en base a los criterios de los usuarios.

La interfaz de usuario de este sistema debe centrarse en ver, investigar y comprender datos. Las infografías presentadas al usuario deben invitar al usuario a ver un resumen de sus gastos, incluidos los detalles de una sola transacción.

La interfaz de usuario debe ser concisa y directa en cuanto a la información que proporciona. Minimalista en cuanto a la facilidad de uso, siguiendo la base de mostrarle única y exclusivamente al usuario aquello que necesita en ese preciso instante. Adaptable a los diferentes dispositivos en los que se muestre. En concepto, la interfaz, el diseño y la experiencia de usuario de este sistema se basa, fundamentalmente, en una estética basada en datos.

#### 3.1. GUÍA DE ESTILO

Para llevar a cabo los objetivos de diseño de esta interfaz hemos seguido las líneas de estilo de Material Design para lo cual usaremos la librería de Material-UI que proporciona componentes de React JS.



*Ilustración 14. Material-UI.*

### 3.2. ESTRUCTURA DE LA INTERFAZ

La estructura de este sistema dispone de varias vistas a través de las cuales el usuario podrá ver sus finanzas personales. Las vistas principales son las siguientes:

#### 3.2.1. LA PANTALLA DE INICIO

La cual muestra un resumen financiero con diferentes gráficos, que proporcionen información relevante al usuario.



Ilustración 15. Pantalla de Inicio



Ilustración 16. Pantalla de Inicio Móvil

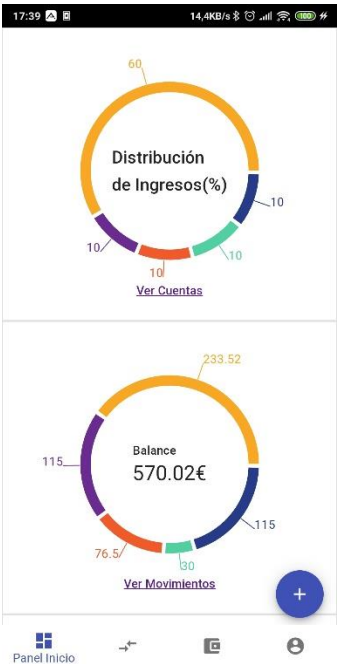


Ilustración 17. Pantalla de Inicio Móvil2

3.2.2. LA PANTALLA DE CUENTAS

La cual muestra información las cuentas que ha creado el usuario

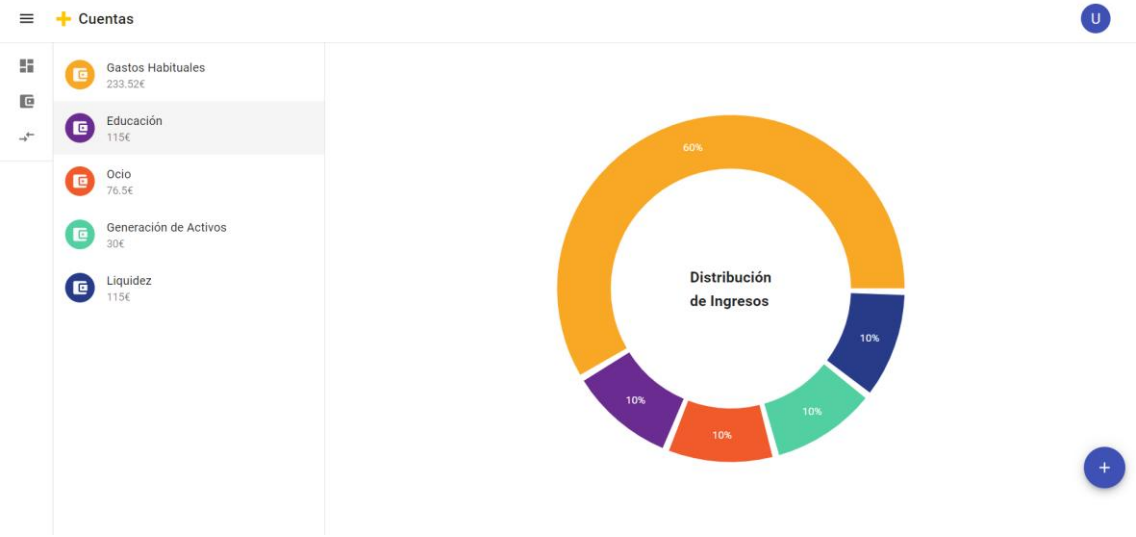


Ilustración 18. Pantalla de Cuentas



Ilustración 19. Pantalla de Cuentas versión móvil

### 3.2.3. LA PANTALLA DE MOVIMIENTOS

La cual muestra información sobre los movimientos, ya sean gastos o ingresos, que haya creado el usuario.

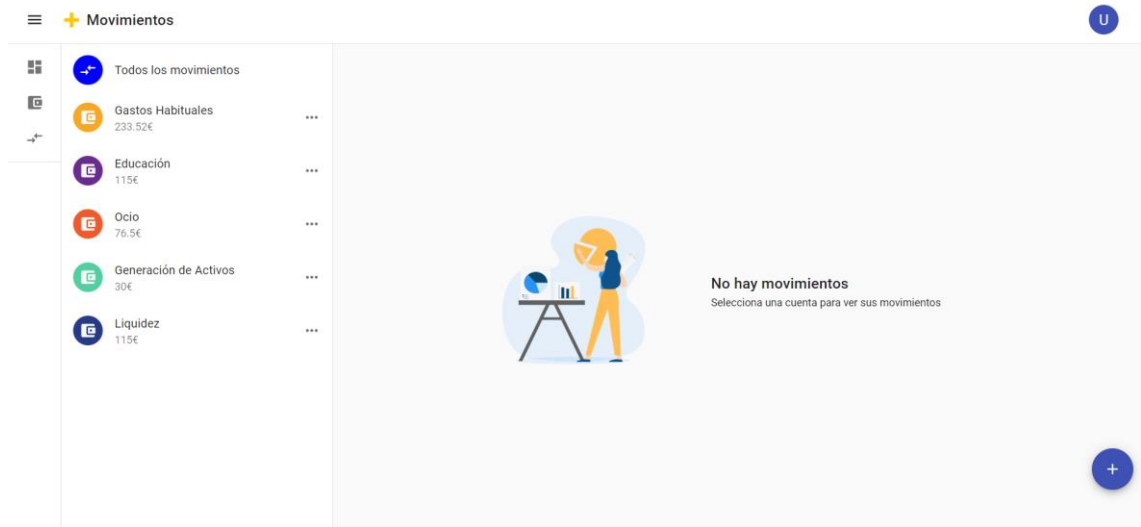


Ilustración 20. Pantalla de Movimientos

### 3.2.4. OTRAS VISTAS

A mayores de estas vistas principales tenemos la vista del usuario donde se muestra la información del usuario.

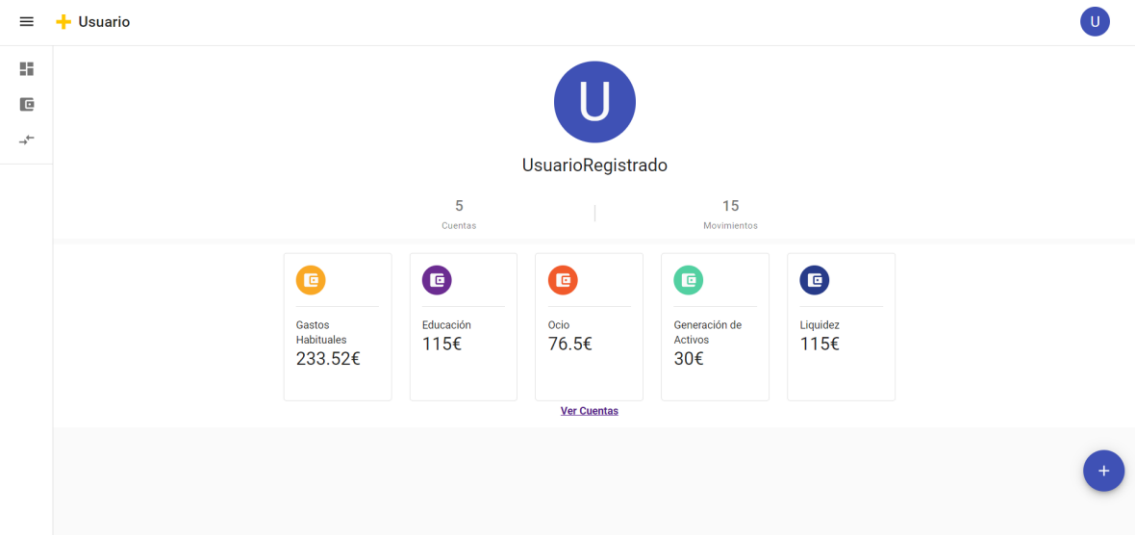
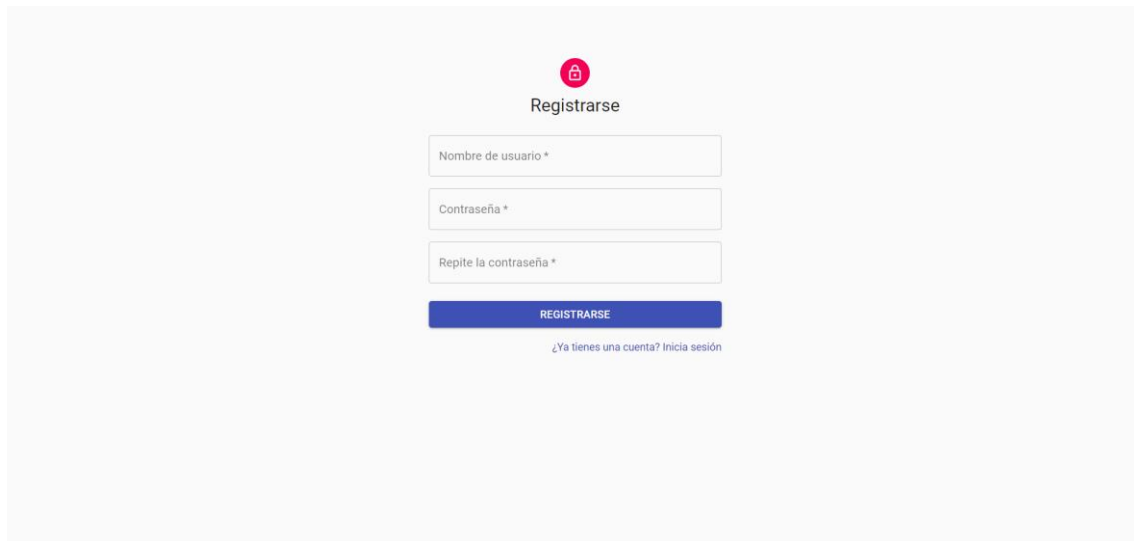


Ilustración 21. Vista de Usuario



Ilustración 22. Pantalla de Usuario versión móvil

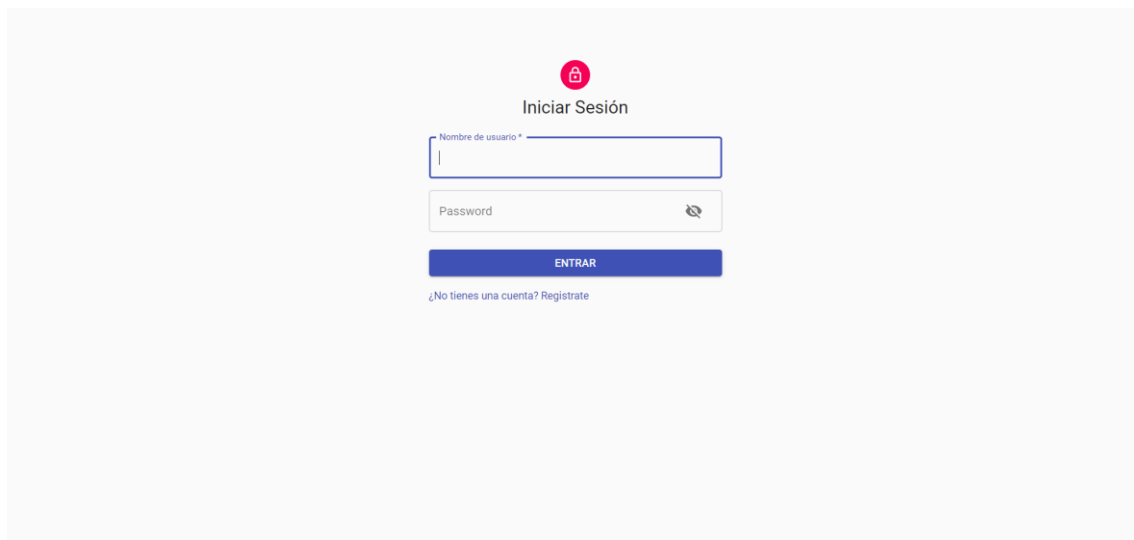
## La vista de Registrarse



The registration form is centered on a light gray background. At the top is a red circular icon with a white padlock. Below it is the title "Registrarse". The form consists of three white input fields with gray borders: "Nombre de usuario \*", "Contraseña \*", and "Repite la contraseña \*". Below these fields is a blue button with the text "REGISTRARSE" in white. At the bottom, there is a link that says "¿Ya tienes una cuenta? Inicia sesión".

*Ilustración 23. Vista de Registrarse*

## Y la vista de inicio de sesión



The login form is centered on a light gray background. At the top is a red circular icon with a white padlock. Below it is the title "Iniciar Sesión". The form consists of two white input fields with gray borders: "Nombre de usuario \*" and "Password". The "Password" field has a small eye icon on the right side. Below these fields is a blue button with the text "ENTRAR" in white. At the bottom, there is a link that says "¿No tienes una cuenta? Regístrate".

*Ilustración 24. Vista de Inicio de Sesión*

### 3.3. DIVISIÓN EN COMPONENTES

Dado que el framework que vamos a utilizar para la realización de la interfaz es un framework orientado a componentes, React JS, una división preliminar en componentes facilitará el desarrollo. De esta forma dividiremos la interfaz en componentes que la compongan y estos a su vez en subcomponentes buscando la mayor reutilización posible de componentes a lo largo del desarrollo.

#### 3.3.1. COMPONENTES DE NAVEGACIÓN

Dado que los elementos de la navegación estarán presentes a lo largo de la página, la mejor forma de reutilizarlos es agruparlos en componentes.

Uno de estos componentes es la barra de navegación, resaltada en amarillo en la ilustración, la cual está presente a lo largo de la mayoría de las vistas, salvo en la versión de móvil que se sustituye por una barra inferior. También tenemos una barra lateral, en rojo en la ilustración, en la que se muestran las diferentes vistas a la que el usuario puede acceder.

En la parte inferior y resaltada en verde, tenemos el botón para crear un nuevo elemento.

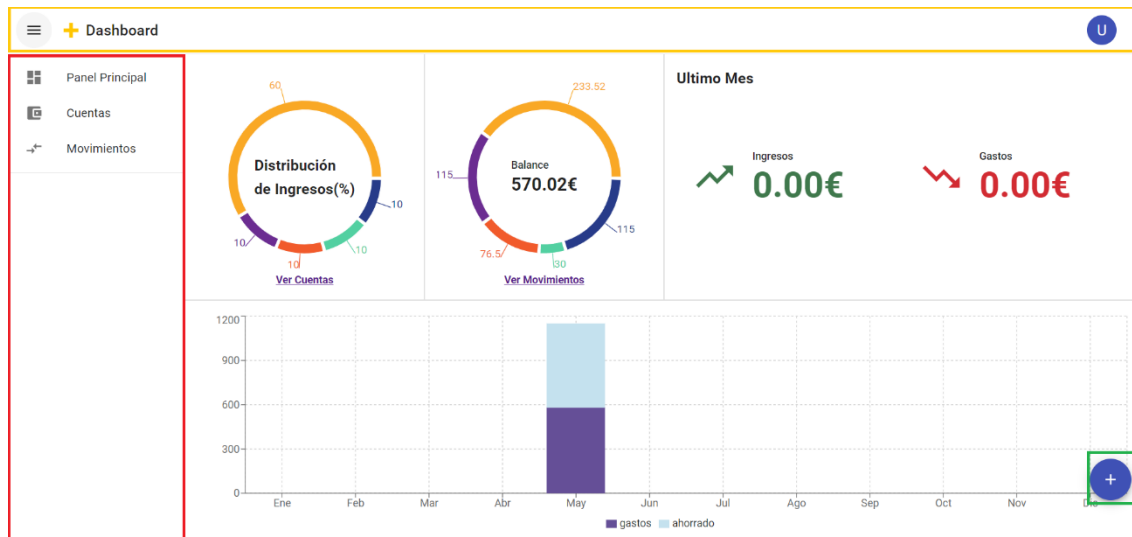


Ilustración 25. Componentes de navegación

A su vez estos componentes están compuestos por otros subcomponentes como podemos apreciar en la siguiente ilustración en la que se resaltan los diferentes subcomponentes que forman la barra de navegación.



Ilustración 26. Subcomponentes Barra de navegación

### 3.3.2. COMPONENTES DE INFORMACIÓN

Para mostrar la información en diferentes gráficos contamos con los componentes de información los cuales se muestran resaltados en la siguiente ilustración.



Ilustración 27. Componentes de Información

### 3.3.3. COMPONENTES DE ACCESO A INFORMACIÓN

Los cuales nos permiten interactuar con la interfaz para acceder a las diferentes cuentas y movimientos, como se muestra en la siguiente ilustración.

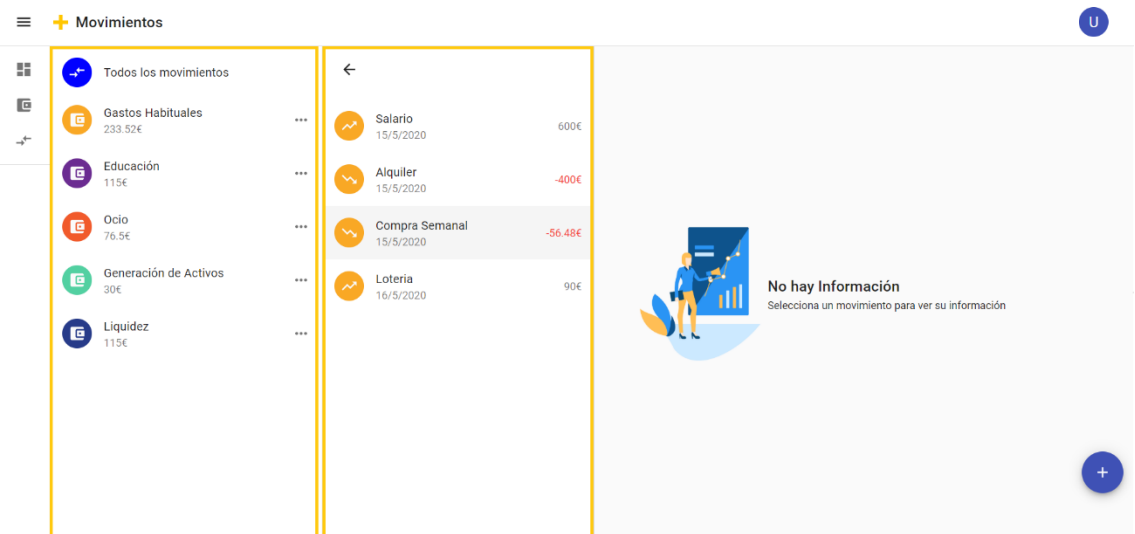


Ilustración 28. Componentes de acceso a información



### 3.3.4. COMPONENTE DE AYUDA AL USUARIO

A veces es necesario informar al usuario de cual es el siguiente paso que puede realizar, para ello están estos componentes de ayuda al usuario que se muestran resaltados en las imágenes inferiores.

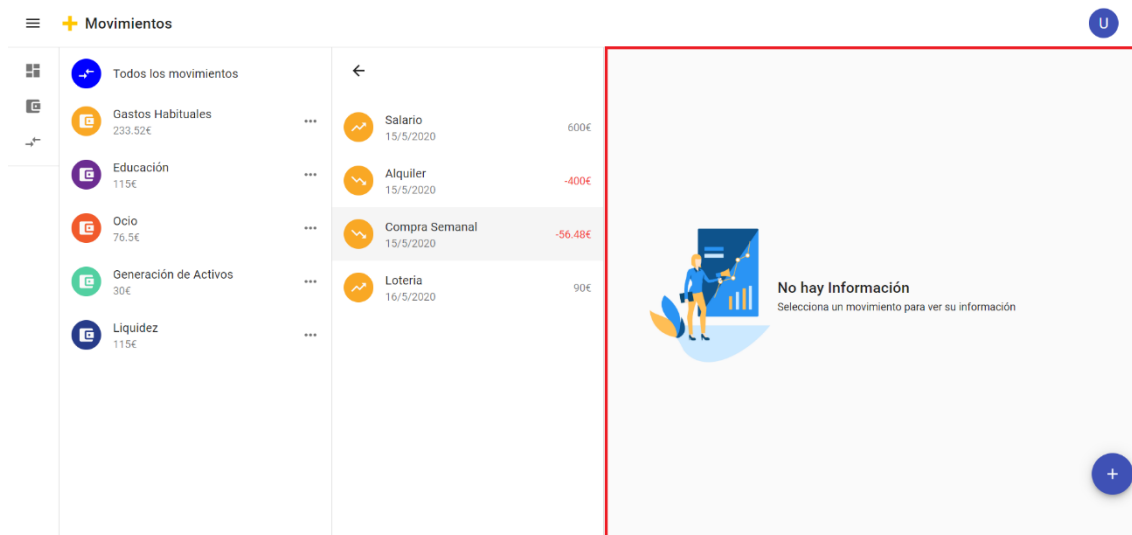


Ilustración 29. Componente de ayuda al usuario 1

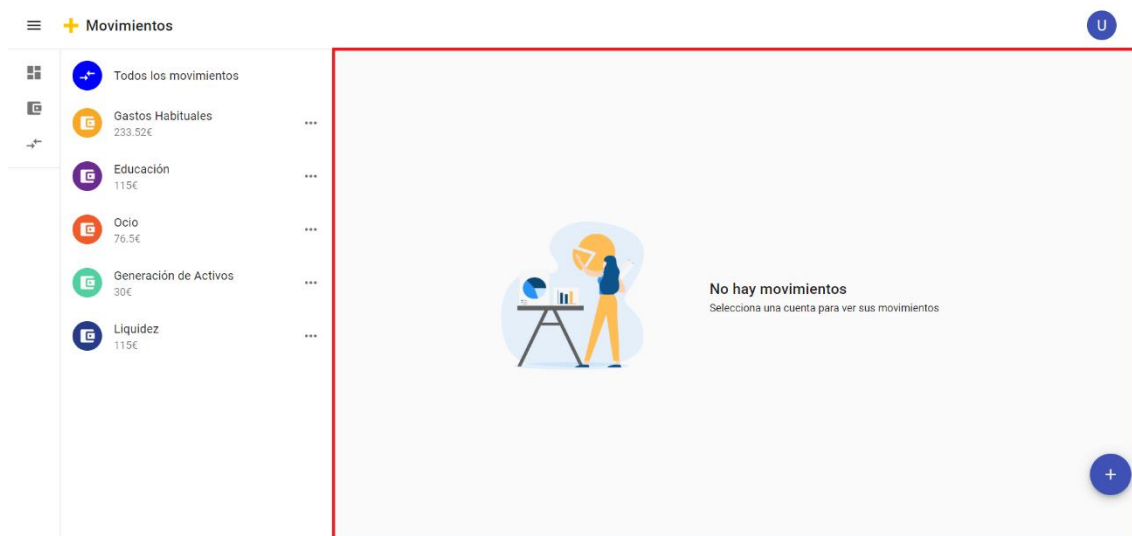


Ilustración 30. Componente de ayuda al usuario 2

### 3.3.5. COMPONENTES DE CREACIÓN Y MODIFICACIÓN

Cuando el usuario quiera crear un nuevo movimiento, una nueva cuenta o incluso modificar una de estas dos tenemos los componentes de creación y modificación.

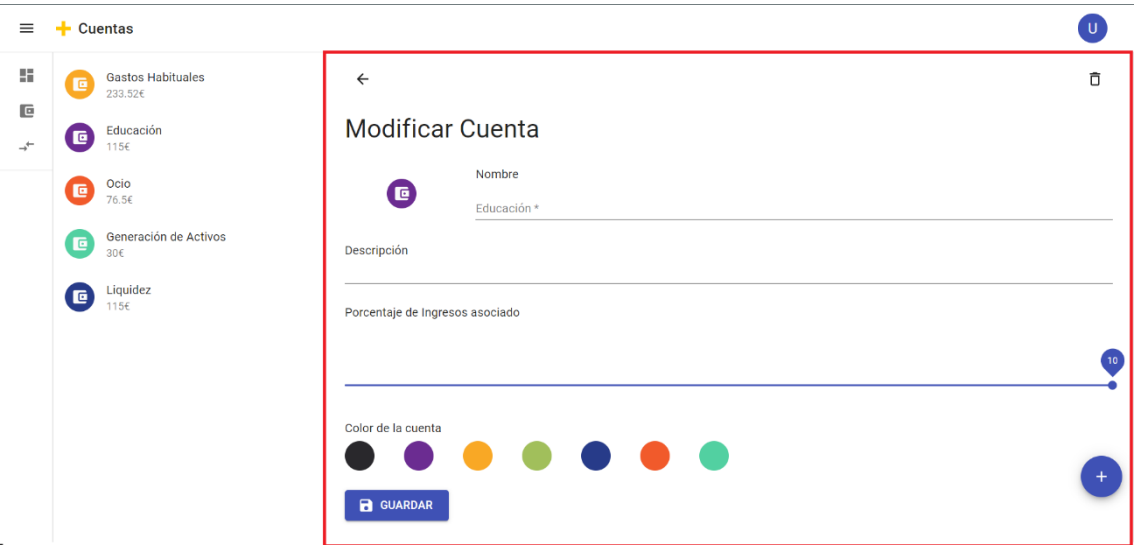


Ilustración 31. Componente de Modificación

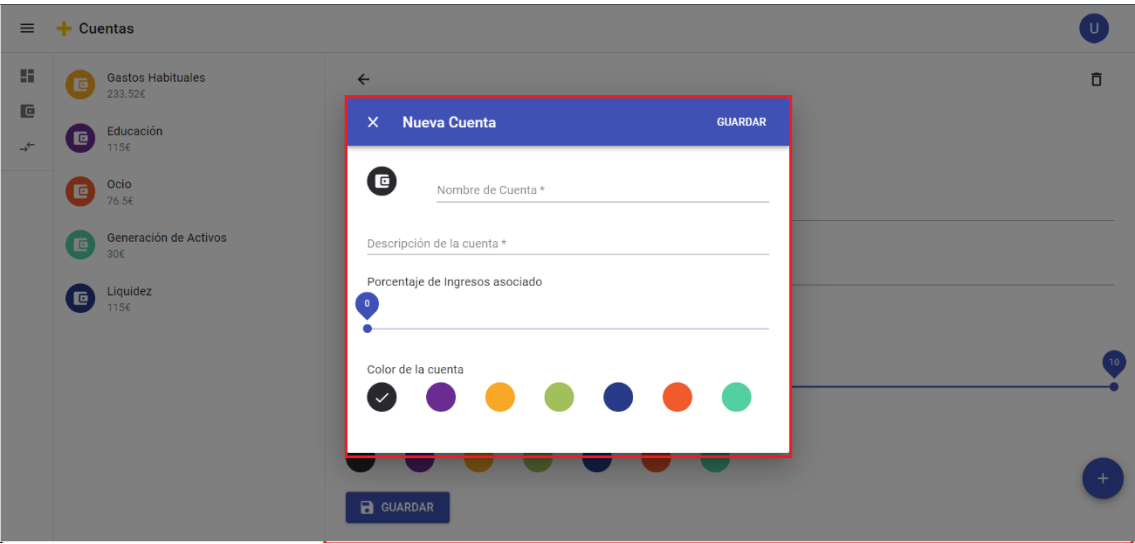


Ilustración 32. Componente de creación

## 4. ARQUITECTURA DEL SOFTWARE

En primer lugar, debemos definir la estructura arquitectónica con la que definiremos nuestro proyecto. “La arquitectura de software se refiere a las estructuras fundamentales de un sistema de software y la disciplina de crear tales estructuras y sistemas. Cada estructura comprende elementos de software, relaciones entre ellos y propiedades de elementos y relaciones.”[1]

La arquitectura software que se usará en este sistema será una arquitectura orientada a servicios (SOA). Esta arquitectura proporciona ventajas sustanciales para los objetivos y requisitos del sistema.

SOA es una arquitectura que prioriza:

- “El Valor del Negocio por encima de la estrategia técnica”[2]
- “Las Metas Estratégicas por encima de los beneficios específicos de los proyectos”[2]
- “La Interoperabilidad Intrínseca por encima de la integración personalizada”[2]
- “Los Servicios Compartidos por encima de las implementaciones de propósito específico”[2]
- “La Flexibilidad por encima de la optimización”[2]
- “El Refinamiento Evolutivo encima de la búsqueda de la perfección inicial”[2]

En base a esto, el sistema software desarrollado con la arquitectura SOA contará con dos partes principales:

1. El frontend del sistema, este será la aplicación que contará con la interfaz de usuario (UI) con la cual los usuarios del sistema interactuarán.

Para llevar esto a cabo se seguirá un diseño basado en componentes utilizando el Framework Javascript React JS.



*Ilustración 33. Framework React JS*

2. El backend del sistema, el cual proporcionará, acceso a la información alojada en la base de datos mediante una API REST.

De esta forma, al contar con una API REST, se proporciona una gran flexibilidad al sistema permitiendo acceder a la información desde diferentes tecnologías. Por ejemplo, esto permite que nuestra aplicación frontend, desarrollada en una determinada tecnología, puede comunicarse y acceder a la información. Pero también permitiría a los sistemas ERP o CRM de las empresas registradas en el sistema acceder a la información del sistema, aunque este desarrollados en otra tecnología diferente.

Para llevar a cabo este desarrollo se utilizará el framework PHP Yii2.



*Ilustración 34. Framework Yii2*

## 5. ABREVIATURAS

<i>Abreviatura</i>	<i>Significado</i>
<i>SOA</i>	Arquitectura Orientada a Servicios ( <i>Service Oriented Architecture</i> )
<i>UI</i>	Interfaz de Usuario ( <i>User Interface</i> )
<i>REST</i>	Transferencia de estado representacional ( <i>Representational State Transfer</i> )
<i>ERP</i>	Sistema de Planificación de Recursos Empresariales ( <i>Enterprise Resource Planning</i> )
<i>CRM</i>	Manejo de la relación con el cliente ( <i>Customer Relationship Management</i> )
<i>API</i>	Interfaz de programación de aplicaciones ( <i>Application programming interface</i> )

## 6. BIBLIOGRAFÍA

- [1] P. Clements, *Documenting software architectures : views and beyond*. Boston: Addison-Wesley, 2011.
- [2] “Manifiesto SOA.” [Online]. Available: [http://www.soa-manifesto.org/default\\_spanish.html](http://www.soa-manifesto.org/default_spanish.html). [Accessed: 03-Jun-2020].