

IES NERVIÓN



ÍNDICE

1.	Estudio de viabilidad del sistema.....	3
1.1.	Descripción general del sistema	
1.1.1.	Diagrama de contexto del sistema	
1.2.	Descripción de los sistemas de información actuales	
1.3.	Catálogo de requisitos previo	
1.3.1.	Requisitos funcionales	
1.3.2.	Requisitos de datos	
1.3.3.	Requisitos de interfaz	
1.4.	Alternativas de solución	
1.4.1.	Alternativa I	
1.4.2.	Alternativa II	
1.4.3.	Análisis coste-beneficio	
1.5.	Justificación de alternativa seleccionada	
2.	Gestión del proyecto.....	10
2.1.	Backlog	
2.2.	Sprints	
3.	Análisis de Sistemas de información (ASI).....	16
3.1.	Descripción general del entorno tecnológico del sistema	
3.2.	Catálogo de usuarios	
3.3.	Modelo de casos de uso	
3.4.	Especificación de casos de uso	
3.5.	Modelo de clases de análisis	
3.6.	Interfaces de Usuario	
4.	Diseño de Sistemas de información(DSI).....	32
4.1.	Diseño de la arquitectura del sistema	
4.2.	Modelo de las clases de diseño	
4.3.	Diseño de la interfaz de usuario	
4.4.	Plan de migración y carga inicial de datos	
4.5.	Plan de pruebas	

1. Estudio de viabilidad del sistema (EVS)

Con el presente trabajo de Estudio de Viabilidad del Sistema, se busca entender de manera más completa los puntos más relevantes del desarrollo del proyecto SaveMoney, como son el estudio del mercado, alternativas, herramientas y el desarrollo técnico.

1.1 Descripción general del sistema

SaveMoney ayuda a seguir su actividad financiera de manera eficiente. Su diseño simple lo hace ligero, directo y muy fácil de usar. Solo se necesita un poco de tiempo al día para ahorrar la cantidad y el propósito de cada gasto. No importa dónde se encuentre, solo un par de toques le ahorrarán los gastos.

Principales características:

- Rastreador de gastos e ingresos fácil de usar, limpio e intuitivo.
 - Interfaz de usuario intuitiva y fácil de usar.
- Seguimiento de gastos e ingresos.
 - Registre fácilmente dónde entra y sale su dinero.
- La imagen completa en un solo lugar.
 - Visión clara de su vida financiera. Entienda mejor dónde va y viene su dinero con informes visualizados sobre ingresos, gastos por tiempo y categoría.

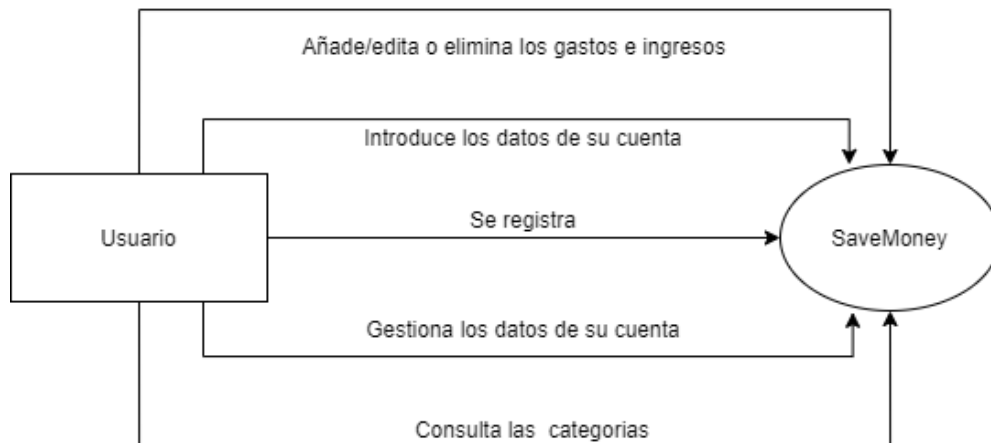
La pantalla principal mostrará el dinero que tienes en la cuenta, y dos secciones ; gastos e ingresos.

En la sección de gastos se mostrará los distintos pagos que que hayas realizado en una semana, día o mes y su categoría. La posibilidad de añadir gastos y distintas categorías según dichos gastos. La sección de ingresos tiene la misma funcionalidad pero en lugar de gastos mostramos y añadimos los ingresos.

En la parte superior izquierda se desplegará un menú hamburguer con distintas opciones de la aplicación. En este menú se encontrarán las opciones de:

- Gráfico : esta será la pantalla principal que hemos descrito anteriormente.
- Categorías : en esta sección aparecerán las distintas categorías que existen
- Resumen: en esta sección se mostrará el resumen de la cuenta del usuario
- Ajustes: se mostrará las opciones de cerrar sesión, idioma, borrar datos...
- Log-out : para cerrar la sesión

1.1.1 Diagrama de contexto del sistema



1.2 Descripción de sistemas de información actuales

Actualmente existen sistemas desarrollados y su uso cada vez es más frecuente. Estos han tenido un gran éxito e impacto últimamente. Debido a esto es un buen momento para sacar uno nuevo.

Los sistemas actuales están orientados a una forma más formal de controlar la cuenta de un usuario, es decir, normalmente pide más datos o los trabaja de una forma más compleja a la hora de manejar los distintos gastos o ingresos.

Aunque hay excepciones las cuales si siguen la misma filosofía de "simplicidad", con un funcionamiento más sencillo y minimalista, pero cumpliendo la misma función.

Esta aplicación abarca varias secciones, pero de un modo más general y fácil de entender para el usuario, por lo que la hace muy intuitiva.

1.3 Catálogo de requisitos previo

1.3.1 Requisitos funcionales

REQUISITOS FUNCIONALES	
Descripción	Prioridad
Debe registrar usuarios y validar los datos	1
Debe permitir iniciar sesión al usuario	1
Debe permitir cerrar sesión al usuario	2
Debe verificar el email del usuario al registrarse	2
Debe permitir recuperar la contraseña a partir del correo registrado anteriormente	2
Debe permitir editar los datos del perfil del usuario	2
Debe permitir añadir gastos	1
Debe permitir eliminar gastos	1
Debe permitir añadir ingresos	1
Debe permitir eliminar ingresos	1
Debe cambiar los datos de la cuenta al añadir un ingreso o un gasto	2
Debe permitir ver las distintas categorías y añadirlas a los gastos o ingresos	2
Debe permitir navegar entre las distintas secciones a través del menú	2

1.3.2 Requisitos de datos

REQUISITOS DE DATOS	
Descripción	Prioridad
Deben guardarse los datos de la cuenta de cada usuario	1
Deben guardarse los gastos obtenidos de cada usuario	2
Deben guardarse los ingresos añadidos de cada usuario	2
Deben guardarse los datos de cada usuario	1
Deben guardarse las categorías	2

1.3.3 Requisitos de Interfaz

REQUISITOS DE INTERFAZ	
Descripción	Prioridad
Debe mostrar los datos de perfil de cada usuario	1
Debe mostrar el balance total de la cuenta	1
Debe mostrar los gastos	1
Debe mostrar los ingresos	1
Debe mostrar las categorías existentes	2
Debe mostrar el resumen de la cuenta	2
Debe mostrar una barra de navegación entre los distintos sectores	2
Debe mostrar la sección de ajustes para cambiar de tema y mostrar la política de privacidad	1
Debe poder cambiarse el tema(claro/oscura)	3

1.4 Alternativas de solución

1.4.1 Alternativa 1

La primera alternativa consiste en desarrollar una aplicación multiplataforma, para Android e IOS, usando Ionic y Angular como framework. Para la base de datos usaremos una base de datos nosql en tiempo real, en la nube de Firebase. La cual ofrece escalabilidad, amplia documentación y comunidad.

1.4.1.1 Requisitos

La aplicación solo puede ser desplegada en dispositivos Android, IOS y en web (en su fase de desarrollo).

1.4.1.2 Tecnologías

- Ionic
- Html
- Scss
- TypeScript
- Node.js
- Firebase Authentication

1.4.1.3 Estudio de riesgo

Riesgo	Porcentaje	Importancia
Falta de experiencia de los desarrolladores	50%	1
Incumplimiento de los plazos	30%	1
Problemas para soportar determinado número de usuarios	10%	2
Incompatibilidad con distintos dispositivos	30%	2
Poco aprovechamiento de la tecnología	40%	2
Poca fiabilidad de las pruebas mono-usuario mientras se desarrolla	30%	2
Poca estabilidad	10%	3

1.4.2 Alternativa 2

La segunda alternativa consiste en desarrollar una aplicación para Android, para usarlo mayoritariamente en dispositivos móviles. Almacenar los datos en una base de datos DOM que el mismo Android implementa y para el login usaremos FireBase.

1.4.2.1 Requisitos

La aplicación solo puede ser desplegada en dispositivos Android, a partir de la versión 5.0 o superior.

1.4.2.2 Tecnologías

- Android Studio 3.0
- Base de datos DOM
- Lenguaje de programación Java
- Firebase Authentication

1.4.2.3 Estudio de riesgo

Riesgo	Porcentaje	Importancia
Falta de experiencia de los desarrolladores	40%	1
Incumplimiento de los plazos	20%	1
Problemas para soportar determinado número de usuarios	10%	2
Incompatibilidad con distintos dispositivos	30%	2
Poco aprovechamiento de la tecnología	40%	2
Poca fiabilidad de las pruebas mono-usuario mientras se desarrolla	30%	2
Poca estabilidad	10%	3

1.5 Justificación de alternativa seleccionada

Escogeremos la primera alternativa por varias razones :

- Los desarrolladores tienen experiencia con ionic y Angular.
- Uso en mayor número de plataformas(Android e IOS)
- No se requiere de hardware complejo y potente para usarlo.
- El uso de dispositivos móviles es muy extendido por lo que tendría mayor visibilidad de cara al público.
- Ofrece una buena estabilidad y sencillez a la hora de usarlo

La segunda alternativa no cambiaría el enfoque de la aplicación ya que seguiría haciendo el uso del móvil, pero en este caso sólo estaría disponible para dispositivos que cuentan con sistema operativo de Android.

2. Gestión del proyecto

Para el desarrollo de este proyecto se ha utilizado la metodología de desarrollo Scrum.



¿Qué es SCRUM?

Se trata de una metodología de trabajo ágil que tiene como finalidad la entrega de valor en períodos cortos de tiempo y para ello se basa en tres pilares: la transparencia, inspección y adaptación.

Los equipos de Scrum son auto-organizados y multifuncionales. Es decir, cada uno es responsable de unas tareas determinadas y de terminarlas en los tiempos acordados. En este caso el equipo es solo el desarrollador Vahagn Hayrapetyan.

La metodología Scrum está basada en sprints. Tenemos un listado de requerimientos, también llamado Backlog. Tomamos uno para desarrollarlo en ese sprint. Lo dividimos en tareas, asignamos esas actividades a los diferentes miembros del equipo. Cada, podemos reunirnos para ir solucionando problemas. Estas reuniones dentro del sprint dependerá de la duración del sprint y del tipo de resultado del proyecto.

Una vez terminado el objetivo de ese sprint, ya tenemos el resultado para incrementarlo en el total del producto terminado. Por eso, tiene tanto sentido utilizarlo en el sector del software. Porque, este producto ya parte de algo que es capaz de funcionar. Así que en cada sprint, iremos consiguiendo mejoras para añadir al producto final.

2.1 Backlog

Proyecto DAM

Tema	ID	User story
Login	LOG1	Como administrador del proyecto Quiero una pagina de inicio de sesión Para poder identificar a los usuarios y que accedan a la aplicación
	LOG2	Como administrador del proyecto Quiero una pagina de registro Para poder registrar los datos de nuevos usuarios.
	LOG3	Como administrador del proyecto Quiero un inicio y registro seguro Para poder administrar los datos de manera segura y cumpliendo la privacidad de los datos-
	LOG4	Como administrador del proyecto Quiero que el usuario pueda cerrar sesión Para poder cambiar de cuenta cuando el usuario quiera
	LOG5	Como administrador del proyecto Quiero verificar el correo electrónico del usuario al registrarse Para poder comprobar que el correo usado es válido
	LOG6	Como administrador del proyecto Quiero que el usuario pueda recuperar su contraseña Para poder cambiar la contraseña en caso de que se le olvide

Presentación	PRE1	Como administrador del proyecto Quiero mostrar una pequeña presentación de como funciona la app Para poder que el usuario tenga una idea de como funciona
	PRE2	Como administrador del proyecto Quiero que el usuario introduzca su cuenta bancaria la primera vez que la use Para poder trabajar dentro de la app sobre esta misma

Operaciones	OPR1	Como administrador del proyecto Quiero mostrar el dinero total que tiene el usuario en todo momento Para poder que el usuario pueda controlar en todo momento el dinero que tiene
	OPR2	Como administrador del proyecto Quiero permitir ver/añadir y eliminar sus gastos al usuario Para poder controlar los gastos del usuario
	OPR3	Como administrador del proyecto Quiero permitir ver/añadir y eliminar sus ingresos al usuario
Categorías	ETI2	Como administrador del proyecto Quiero permitir distintas categorías para los ingresos que realice el usuario Para poder controlar el tipo del ingreso que se realiza
	ETI3	Como administrador del proyecto Quiero permitir distintas categorías para los gastos que realice el usuario Para poder controlar el tipo del gasto que se realiza
	ETI1	Como administrador del proyecto Quiero mostrar las distintas categorías al usuario Para poder controlar las categorías que existen en cada operación
Resumen	CON1	Como administrador del proyecto Quiero mostrar al usuario un resumen de todos los datos de su cuenta Para poder consultar en cualquier momento la situación actual de la cuenta
Settings	SET1	Como administrador del proyecto Quiero que el usuario pueda cambiar de tema oscuro a claro Para poder modificar el tema de la aplicación dependiendo del tema que elija
	SET2	Como administrador del proyecto Quiero mostrar la política de privacidad de la aplicación al usuario Para poder dar a conocerla al usuario

2.2 Sprints

Primer sprint :

- Fecha comienzo : 02/04/2021
- Fecha finalización : 14/04/2021
- Puntos sprint (duración estimada) : 120 = 12 días
- Duración total : 12 días

✓ Sprint 1 stories (completed)		expand
ID: LOG1 Puntos: 34  Accepted	Tema: Login Como administrador del proyecto Quiero una pagina de inicio de sesión Para poder identificar a los usuarios y que accedan a la aplicación	
ID: LOG2 Puntos: 34  Accepted	Tema: Login Como administrador del proyecto Quiero una pagina de registro Para poder registrar los datos de nuevos usuarios.	
ID: LOG3 Puntos: 13  Accepted	Tema: Login Como administrador del proyecto Quiero un inicio y registro seguro Para poder administrar los datos de manera segura y cumpliendo la privacidad de los datos-	
ID: PRE1 Puntos: 8  Accepted more ⌵	Tema: Presentación Como administrador del proyecto Quiero mostrar una pequeña presentación de como funciona la app Para poder que el usuario tenga una idea de como	

Segundo sprint :

- Fecha comienzo : 19/04/2021
- Fecha finalización : 01/04/2021
- Puntos sprint (duración estimada) : 120 = 12 días
- Duración total : 12 días

✓ Sprint 2 stories (completed)

expand

ID: LOG4
Puntos: 5
Accepted

Tema: Login
Como administrador del proyecto
Quiero que el usuario pueda cerrar sesión
Para poder cambiar de cuenta cuando el usuario quiera

ID: PRE2
Puntos: 13
Accepted

Tema: Presentación
Como administrador del proyecto
Quiero que el usuario introduzca su cuenta bancaria la primera vez que la use
Para poder trabajar dentro de la app sobre esta misma

ID: OPR1
Puntos: 21
Accepted

Tema: Operaciones
Como administrador del proyecto
Quiero mostrar el dinero total que tiene el usuario en todo momento
Para poder que el usuario pueda controlar en todo

ID: ET11
Puntos: 21
Accepted

Tema: Categorías
Como administrador del proyecto
Quiero mostrar las distintas categorías al usuario
Para poder controlar las categorías que existen en cada operación

Tercer sprint :

- Fecha comienzo : 09/06/2021
- Fecha finalización : 21/05/2021
- Puntos sprint (duración estimada) : 120 = 12 días
- Duración total : 12 días

✓ Sprint 3 stories (completed)

expand

ID: LOG5
Puntos: 5
Accepted

Tema: Login
Como administrador del proyecto
Quiero verificar el correo electrónico del usuario al registrarse
Para poder comprobar que el correo usado es válido

ID: LOG6
Puntos: 5
Accepted

Tema: Login
Como administrador del proyecto
Quiero que el usuario pueda recuperar su contraseña
Para poder cambiar la contraseña en caso de que se le olvide

ID: OPR2
Puntos: 34
Accepted

Tema: Operaciones
Como administrador del proyecto
Quiero permitir ver/añadir y eliminar sus gastos al usuario
Para poder controlar los gastos del usuario

ID: OPR3
Puntos: 34
Accepted






Tema: Operaciones
Como administrador del proyecto
Quiero permitir ver/añadir y eliminar sus ingresos al usuario
Para poder controlar los ingresos del usuario

Cuarto sprint :

- Fecha comienzo : 28/04/2021
- Fecha finalización : 12/05/2021
- Puntos sprint (duración estimada) : 120 = 12 días
- Duración total : 14 días

✓ Sprint 4 stories (completed)

[expand](#)

ID: ETI2 Puntos: 21  Accepted	Tema: Categorías Como administrador del proyecto Quiero permitir distintas categorías para los ingresos que realice el usuario Para poder controlar el tipo del ingreso que se realiza
ID: ETI3 Puntos: 21  Accepted	Tema: Categorías Como administrador del proyecto Quiero permitir distintas categorías para los gastos que realice el usuario Para poder controlar el tipo del gasto que se realiza
ID: CON1 Puntos: 21  Accepted more ⇓	Tema: Resumen Como administrador del proyecto Quiero mostrar al usuario un resumen de todos los datos de su cuenta Para poder consultar en cualquier momento la situación
ID: SET1 Puntos: 13  Accepted more ⇓	Tema: Settings Como administrador del proyecto Quiero que el usuario pueda cambiar de tema oscuro a claro Para poder modificar el tema de la aplicación dependiendo
ID: SET2 Puntos: 8  Accepted	Tema: Settings Como administrador del proyecto Quiero mostrar la política de privacidad de la aplicación al usuario Para poder dar a conocerla al usuario

3. Análisis de Sistemas de información (ASI)

3.1 Descripción general del entorno tecnológico del sistema

Para realizar el análisis del sistema, vamos a utilizar un SDK de frontend de código abierto para desarrollar aplicaciones híbridas basado en tecnologías web (HTML, CSS y JS), es decir, un framework que nos permite desarrollar aplicaciones para iOS nativo, Android y la web, desde una única base de código. Esta tecnología es Ionic.

Se integra con los principales frameworks de frontend, como Angular, React y Vue, en este caso vamos a utilizar Angular.

Este entorno de desarrollo no posee una filosofía de estructuración de proyectos concreta, por lo que se optará por la clasificación por carpetas.

En cuanto a la base de datos y autenticación usaremos una base de datos nosql en tiempo real, en la nube de Firebase.

3.3 Catálogos de usuarios

CÓDIGO	NOMBRE	DESCRIPCIÓN
USU1	Usuario no registrado	Tiene solo permiso para acceder al área registro para poder identificarse en la aplicación.
USU2	Usuario registrado	Dispone de todas las funcionalidades de la aplicación sin ningún tipo de límite.

3.3 Modelo de caso de uso



3.4 Especificación de los casos de uso

UC1	Registro de Usuario														
Versión	1.0														
Actor principal	USU1 Usuario no registrado														
Personal involucrado e interés	USU1 y el Sistema														
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario anónimo desea registrarse en nuestra aplicación.														
Precondiciones	El correo usado no puede estar ya registrado en la aplicación														
Postcondiciones	El usuario no registrado debe generar un nuevo USU2 usuario registrado														
Secuencia normal	<table> <tr> <th>Paso</th><th>Acción</th></tr> <tr> <td>1</td><td>Si el sistema detecta que un usuario no está registrado ofrece la posibilidad de registrarse al usuario</td></tr> <tr> <td>2</td><td>El usuario solicita el registro al sistema</td></tr> <tr> <td>3</td><td>El sistema muestra un formulario con los datos necesarios para crear un nuevo usuario registrado</td></tr> <tr> <td>4</td><td>El usuario rellena el formulario y lo envía</td></tr> <tr> <td>5</td><td>El sistema registra al nuevo usuario</td></tr> <tr> <td>6</td><td>El sistema manda la verificación al correo registrado</td></tr> </table>	Paso	Acción	1	Si el sistema detecta que un usuario no está registrado ofrece la posibilidad de registrarse al usuario	2	El usuario solicita el registro al sistema	3	El sistema muestra un formulario con los datos necesarios para crear un nuevo usuario registrado	4	El usuario rellena el formulario y lo envía	5	El sistema registra al nuevo usuario	6	El sistema manda la verificación al correo registrado
Paso	Acción														
1	Si el sistema detecta que un usuario no está registrado ofrece la posibilidad de registrarse al usuario														
2	El usuario solicita el registro al sistema														
3	El sistema muestra un formulario con los datos necesarios para crear un nuevo usuario registrado														
4	El usuario rellena el formulario y lo envía														
5	El sistema registra al nuevo usuario														
6	El sistema manda la verificación al correo registrado														
Extensiones o flujos alternativos	<table> <tr> <th>Paso</th><th>Acción</th></tr> <tr> <td>1</td><td>Si alguno de los datos introducidos por el usuario es incorrecto, se informa por pantalla al usuario.</td></tr> <tr> <td>2</td><td>Si el usuario intenta registrarse con unos datos ya registrados, se informa por pantalla al usuario.</td></tr> </table>	Paso	Acción	1	Si alguno de los datos introducidos por el usuario es incorrecto, se informa por pantalla al usuario.	2	Si el usuario intenta registrarse con unos datos ya registrados, se informa por pantalla al usuario.								
Paso	Acción														
1	Si alguno de los datos introducidos por el usuario es incorrecto, se informa por pantalla al usuario.														
2	Si el usuario intenta registrarse con unos datos ya registrados, se informa por pantalla al usuario.														
Frecuencia esperada	15/día														

UC2	Login usuario												
Versión	1.0												
Actor principal	USU2 Usuario registrado												
Personal involucrado e interés	USU2 y el Sistema												
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario anónimo desea registrarse en nuestra aplicación.												
Precondiciones	Los datos usados deben estar ya registrados en la aplicación												
Postcondiciones	El usuario accede a la aplicación con un perfil registrado.												
Secuencia normal	<table> <tr> <th>Paso</th><th>Acción</th></tr> <tr> <td>1</td><td>El usuario solicita acceso al sistema</td></tr> <tr> <td>2</td><td>El sistema muestra un formulario con los datos necesarios para acceder</td></tr> <tr> <td>3</td><td>El usuario rellena el formulario y lo envía</td></tr> <tr> <td>4</td><td>El sistema valida el formulario e identifica al usuario</td></tr> <tr> <td>5</td><td>El sistema lleva al usuario al menú principal mostrando sus datos correspondientes</td></tr> </table>	Paso	Acción	1	El usuario solicita acceso al sistema	2	El sistema muestra un formulario con los datos necesarios para acceder	3	El usuario rellena el formulario y lo envía	4	El sistema valida el formulario e identifica al usuario	5	El sistema lleva al usuario al menú principal mostrando sus datos correspondientes
Paso	Acción												
1	El usuario solicita acceso al sistema												
2	El sistema muestra un formulario con los datos necesarios para acceder												
3	El usuario rellena el formulario y lo envía												
4	El sistema valida el formulario e identifica al usuario												
5	El sistema lleva al usuario al menú principal mostrando sus datos correspondientes												
Extensiones o flujos alternativos	<table> <tr> <th>Paso</th><th>Acción</th></tr> <tr> <td>1</td><td>Si alguno de los datos introducidos por el usuario es incorrecto, se informa por pantalla al usuario.</td></tr> <tr> <td>2</td><td>Si el usuario intenta registrarse con unos datos que no estén registrados, se informa por pantalla al usuario.</td></tr> </table>	Paso	Acción	1	Si alguno de los datos introducidos por el usuario es incorrecto, se informa por pantalla al usuario.	2	Si el usuario intenta registrarse con unos datos que no estén registrados, se informa por pantalla al usuario.						
Paso	Acción												
1	Si alguno de los datos introducidos por el usuario es incorrecto, se informa por pantalla al usuario.												
2	Si el usuario intenta registrarse con unos datos que no estén registrados, se informa por pantalla al usuario.												
Frecuencia esperada	30/día												

UC3	Añadir cuenta bancaria										
Versión	1.0										
Actor principal	USU2 Usuario registrado										
Personal involucrado e interés	USU2 y el Sistema										
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario anónimo desea registrarse en nuestra aplicación.										
Precondiciones	Ninguna										
Postcondiciones	Al usuario se le asigna su cuenta										
Secuencia normal	<table> <tr> <th>Paso</th><th>Acción</th></tr> <tr> <td>1</td><td>El sistema muestra un formulario con los datos necesarios para introducir su cuenta</td></tr> <tr> <td>2</td><td>El usuario rellena el formulario y lo envía</td></tr> <tr> <td>3</td><td>El sistema valida el formulario y asigna la cuenta al usuario</td></tr> <tr> <td>4</td><td>El sistema lleva al usuario al menú principal mostrando sus datos correspondientes</td></tr> </table>	Paso	Acción	1	El sistema muestra un formulario con los datos necesarios para introducir su cuenta	2	El usuario rellena el formulario y lo envía	3	El sistema valida el formulario y asigna la cuenta al usuario	4	El sistema lleva al usuario al menú principal mostrando sus datos correspondientes
Paso	Acción										
1	El sistema muestra un formulario con los datos necesarios para introducir su cuenta										
2	El usuario rellena el formulario y lo envía										
3	El sistema valida el formulario y asigna la cuenta al usuario										
4	El sistema lleva al usuario al menú principal mostrando sus datos correspondientes										
Extensiones o flujos alternativos	<table> <tr> <th>Paso</th><th>Acción</th></tr> <tr> <td>1</td><td>Si alguno de los datos introducidos por el usuario es incorrecto, se informa por pantalla al usuario.</td></tr> </table>	Paso	Acción	1	Si alguno de los datos introducidos por el usuario es incorrecto, se informa por pantalla al usuario.						
Paso	Acción										
1	Si alguno de los datos introducidos por el usuario es incorrecto, se informa por pantalla al usuario.										
Frecuencia esperada	30/día										

UC4	Añadir gastos/ingresos												
Versión	1.0												
Actor principal	USU2 Usuario registrado												
Personal involucrado e interés	USU2 y el Sistema												
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario anónimo desea registrarse en nuestra aplicación.												
Precondiciones	Ninguna												
Postcondiciones	Al usuario se le asigna un nuevo gasto/ingreso												
Secuencia normal	<table> <tr> <th>Paso</th><th>Acción</th></tr> <tr> <td>1</td><td>El usuario pulsa el botón de añadir gasto/ingreso.</td></tr> <tr> <td>2</td><td>El sistema muestra un modal con un formulario el cual contiene los datos necesarios que debe introducir el usuario.</td></tr> <tr> <td>3</td><td>El usuario rellena el formulario y lo guarda</td></tr> <tr> <td>4</td><td>El sistema valida el formulario y asigna el gasto/ingreso y modifica el saldo de la cuenta.</td></tr> <tr> <td>5</td><td>El sistema cierra el modal y vuelve a la pantalla principal mostrando el nuevo gasto/ingreso.</td></tr> </table>	Paso	Acción	1	El usuario pulsa el botón de añadir gasto/ingreso.	2	El sistema muestra un modal con un formulario el cual contiene los datos necesarios que debe introducir el usuario.	3	El usuario rellena el formulario y lo guarda	4	El sistema valida el formulario y asigna el gasto/ingreso y modifica el saldo de la cuenta.	5	El sistema cierra el modal y vuelve a la pantalla principal mostrando el nuevo gasto/ingreso.
Paso	Acción												
1	El usuario pulsa el botón de añadir gasto/ingreso.												
2	El sistema muestra un modal con un formulario el cual contiene los datos necesarios que debe introducir el usuario.												
3	El usuario rellena el formulario y lo guarda												
4	El sistema valida el formulario y asigna el gasto/ingreso y modifica el saldo de la cuenta.												
5	El sistema cierra el modal y vuelve a la pantalla principal mostrando el nuevo gasto/ingreso.												
Extensiones o flujos alternativos	<table> <tr> <th>Paso</th><th>Acción</th></tr> <tr> <td>1</td><td>Si alguno de los datos introducidos por el usuario es incorrecto, se informa por pantalla al usuario.</td></tr> </table>	Paso	Acción	1	Si alguno de los datos introducidos por el usuario es incorrecto, se informa por pantalla al usuario.								
Paso	Acción												
1	Si alguno de los datos introducidos por el usuario es incorrecto, se informa por pantalla al usuario.												
Frecuencia esperada	300/día												

UC5	Eliminar gastos/ingresos										
Versión	1.0										
Actor principal	USU2 Usuario registrado										
Personal involucrado e interés	USU2 y el Sistema										
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario anónimo desea registrarse en nuestra aplicación.										
Precondiciones	Debe existir algún gasto/ingreso.										
Postcondiciones	Al usuario se le borra el gasto/ingreso seleccionado										
Secuencia normal	<table> <tr> <th>Paso</th><th>Acción</th></tr> <tr> <td>1</td><td>El usuario desliza hacia la izquierda sobre el gasto/ingreso que quiera eliminar</td></tr> <tr> <td>2</td><td>El sistema muestra un botón de eliminar en el lado derecho</td></tr> <tr> <td>3</td><td>El usuario pulsa el botón</td></tr> <tr> <td>4</td><td>El sistema muestra un pequeño loading y elimina el gasto/ingreso seleccionado</td></tr> </table>	Paso	Acción	1	El usuario desliza hacia la izquierda sobre el gasto/ingreso que quiera eliminar	2	El sistema muestra un botón de eliminar en el lado derecho	3	El usuario pulsa el botón	4	El sistema muestra un pequeño loading y elimina el gasto/ingreso seleccionado
Paso	Acción										
1	El usuario desliza hacia la izquierda sobre el gasto/ingreso que quiera eliminar										
2	El sistema muestra un botón de eliminar en el lado derecho										
3	El usuario pulsa el botón										
4	El sistema muestra un pequeño loading y elimina el gasto/ingreso seleccionado										
Extensiones o flujos alternativos	Ninguna										
Frecuencia esperada	100/día										

UC6	Consultar datos de la cuenta								
Versión	1.0								
Actor principal	USU2 Usuario registrado								
Personal involucrado e interés	USU2 y el Sistema								
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario anónimo desea registrarse en nuestra aplicación.								
Precondiciones	El usuario debe tener una cuenta asignada								
Postcondiciones	Al usuario se le muestra los datos de la cuenta								
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th><th>Acción</th></tr> </thead> <tbody> <tr> <td>1</td><td>El usuario abre el menú lateral</td></tr> <tr> <td>2</td><td>El usuario entra en el apartado de resumen de la cuenta</td></tr> <tr> <td>3</td><td>El sistema muestra resumen con los datos de la cuenta del usuario registrado(balance, gastos e ingresos)</td></tr> </tbody> </table>	Paso	Acción	1	El usuario abre el menú lateral	2	El usuario entra en el apartado de resumen de la cuenta	3	El sistema muestra resumen con los datos de la cuenta del usuario registrado(balance, gastos e ingresos)
Paso	Acción								
1	El usuario abre el menú lateral								
2	El usuario entra en el apartado de resumen de la cuenta								
3	El sistema muestra resumen con los datos de la cuenta del usuario registrado(balance, gastos e ingresos)								
Extensiones o flujos alternativos	Ninguna								
Frecuencia esperada	50/día								

UC7	Consultar datos del usuario
Versión	1.0
Actor principal	USU2 Usuario registrado
Personal involucrado e interés	USU2 y el Sistema
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario anónimo desea registrarse en nuestra aplicación.
Precondiciones	Ninguna
Postcondiciones	Al usuario se le asigna un nuevo gasto/ingreso

Secuencia normal		
	Paso	Acción
	1	El usuario abre el menú lateral
	2	El usuario entra en el apartado de resumen del usuario
	3	El sistema muestra resumen con los datos del usuario registrado (email, nombre)
Extensiones o flujos alternativos	Ninguna	
Frecuencia esperada	50/día	

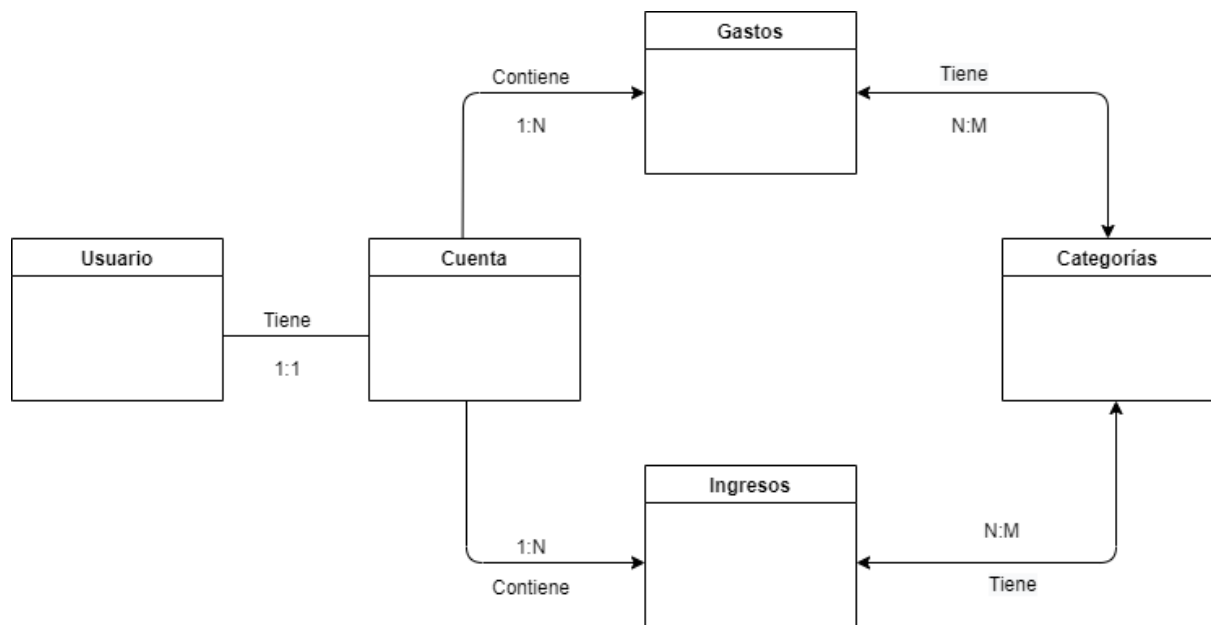
UC8	Ver categorías	
Versión	1.0	
Actor principal	USU2 Usuario registrado	
Personal involucrado e interés	USU2 y el Sistema	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario anónimo desea registrarse en nuestra aplicación.	
Precondiciones	Ninguna	
Postcondiciones	Al usuario se le muestra las distintas categorías	
Secuencia normal		
	Paso	Acción
	1	El usuario abre el menú lateral
	2	El usuario entra en el apartado de categorías
	3	El sistema muestra las distintas categorías existentes tanto para los gastos como para los ingresos
Extensiones o flujos alternativos	Ninguna	
Frecuencia esperada	80/día	

UC9	Cambiar tema												
Versión	1.0												
Actor principal	USU2 Usuario registrado												
Personal involucrado e interés	USU2 y el Sistema												
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario anónimo desea registrarse en nuestra aplicación.												
Precondiciones	Ninguna												
Postcondiciones	El sistema cambia el tema de la aplicación entre claro u oscuro.												
Secuencia normal	<table> <tr> <th>Paso</th><th>Acción</th></tr> <tr> <td>1</td><td>El usuario abre el menú lateral</td></tr> <tr> <td>2</td><td>El usuario entra en el apartado de ajustes</td></tr> <tr> <td>3</td><td>El sistema muestra la opción de cambiar el tema de la aplicación a través de un botón de activación</td></tr> <tr> <td>4</td><td>El usuario pulsa el botón de activación.</td></tr> <tr> <td>5</td><td>El sistema cambia el tema a oscuro/claro dependiendo de la selección del usuario</td></tr> </table>	Paso	Acción	1	El usuario abre el menú lateral	2	El usuario entra en el apartado de ajustes	3	El sistema muestra la opción de cambiar el tema de la aplicación a través de un botón de activación	4	El usuario pulsa el botón de activación.	5	El sistema cambia el tema a oscuro/claro dependiendo de la selección del usuario
Paso	Acción												
1	El usuario abre el menú lateral												
2	El usuario entra en el apartado de ajustes												
3	El sistema muestra la opción de cambiar el tema de la aplicación a través de un botón de activación												
4	El usuario pulsa el botón de activación.												
5	El sistema cambia el tema a oscuro/claro dependiendo de la selección del usuario												
Extensiones o flujos alternativos	Ninguna												
Frecuencia esperada	30/día												

UC9	Ver política de privacidad												
Versión	1.0												
Actor principal	USU2 Usuario registrado												
Personal involucrado e interés	USU2 y el Sistema												
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario anónimo desea registrarse en nuestra aplicación.												
Precondiciones	Ninguna												
Postcondiciones	Al usuario se le muestra en pantalla la política de privacidad de la aplicación												
Secuencia normal	<table> <tr> <th>Paso</th><th>Acción</th></tr> <tr> <td>1</td><td>El usuario abre el menú lateral</td></tr> <tr> <td>2</td><td>El usuario entra en el apartado de ajustes</td></tr> <tr> <td>3</td><td>El sistema muestra la opción de abrir la política de privacidad.</td></tr> <tr> <td>4</td><td>El usuario pulsa dicha opción</td></tr> <tr> <td>5</td><td>El sistema muestra en pantalla la política de privacidad.</td></tr> </table>	Paso	Acción	1	El usuario abre el menú lateral	2	El usuario entra en el apartado de ajustes	3	El sistema muestra la opción de abrir la política de privacidad.	4	El usuario pulsa dicha opción	5	El sistema muestra en pantalla la política de privacidad.
Paso	Acción												
1	El usuario abre el menú lateral												
2	El usuario entra en el apartado de ajustes												
3	El sistema muestra la opción de abrir la política de privacidad.												
4	El usuario pulsa dicha opción												
5	El sistema muestra en pantalla la política de privacidad.												
Extensiones o flujos alternativos	Ninguna												
Frecuencia esperada	10/día												


UC9	Cerrar sesión										
Versión	1.0										
Actor principal	USU2 Usuario registrado										
Personal involucrado e interés	USU2 y el Sistema										
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario anónimo desea registrarse en nuestra aplicación.										
Precondiciones	Ninguna										
Postcondiciones	El sistema cierra la sesión actual de usuario registrado en la aplicación										
Secuencia normal	<table> <tr> <th>Paso</th><th>Acción</th></tr> <tr> <td>1</td><td>El usuario abre el menú lateral</td></tr> <tr> <td>2</td><td>El sistema muestra la opción de cerrar la sesión actual del usuario</td></tr> <tr> <td>3</td><td>El usuario hace click en la opción de cerrar sesión</td></tr> <tr> <td>4</td><td>El sistema cierra la sesión actual de usuario registrado en la aplicación</td></tr> </table>	Paso	Acción	1	El usuario abre el menú lateral	2	El sistema muestra la opción de cerrar la sesión actual del usuario	3	El usuario hace click en la opción de cerrar sesión	4	El sistema cierra la sesión actual de usuario registrado en la aplicación
Paso	Acción										
1	El usuario abre el menú lateral										
2	El sistema muestra la opción de cerrar la sesión actual del usuario										
3	El usuario hace click en la opción de cerrar sesión										
4	El sistema cierra la sesión actual de usuario registrado en la aplicación										
Extensiones o flujos alternativos	Ninguna										
Frecuencia esperada	20/día										

3.5 Modelo de clases de análisis



3.6 Interfaces de usuario

3.6.1 Aspectos comunes de la interfaz de usuario

 MENÚ	SECCIÓN
<ul style="list-style-type: none">OperacionesCategoríasResumenAjustesCerrar sesión	CONTENIDO

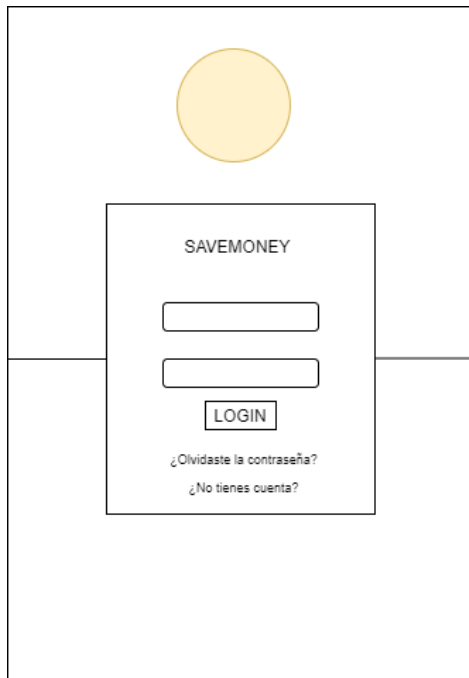
La interfaz mostrará aspectos comunes. Por ejemplo, el menú hamburger con los diferentes iconos de cada sección para navegar a través de estas.

En cada sección, se muestra el nombre de ella en la parte de arriba y ocupando el resto de la pantalla se distribuyen cada característica que contenga la sección.

3.6.2 Especificación de pantallas y ventanas

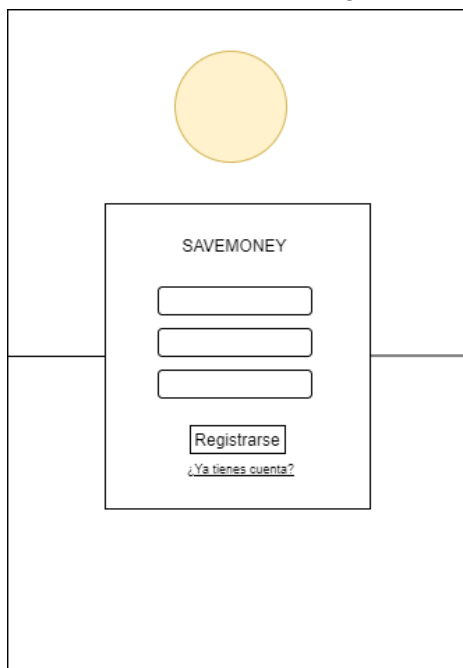
Pantallas:

- Login: Será una pantalla con el logo de la aplicación y el formulario que habrá que rellenar para poder acceder a la aplicación.



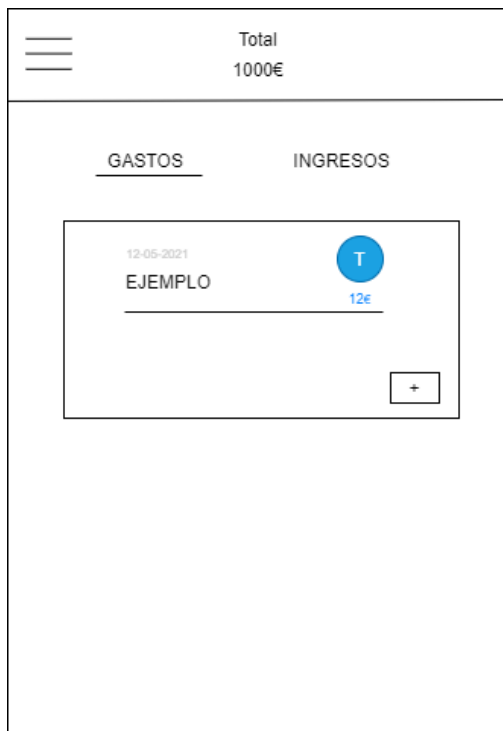
The login screen features a yellow circular logo at the top center. Below it, the text "SAVEMONEY" is displayed. The form consists of two input fields for username and password, followed by a "LOGIN" button. At the bottom, there are two links: "¿Olvidaste la contraseña?" and "¿No tienes cuenta?".

- Registro : Será una pantalla con el logo de la aplicación y el formulario que habrá que rellenar para poder registrarte en la aplicación.



The registration screen features a yellow circular logo at the top center. Below it, the text "SAVEMONEY" is displayed. The form consists of three input fields for name, email, and password, followed by a "Registrarse" button. At the bottom, there is a link: "¿Ya tienes cuenta?".

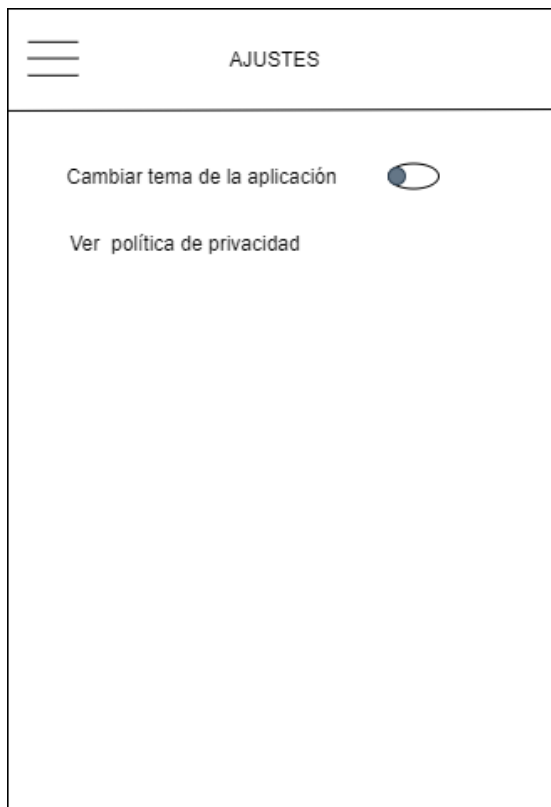
- Main : Será la pantalla principal donde se mostrará el total del saldo de la cuenta y la opción de añadir y de eliminar gastos e ingresos



- Categorías : En esta pantalla se mostrarán todas las categorías disponibles tanto para gastos como para ingresos.



- Ajustes : En esta sección se mostrarán los distintos ajustes de la aplicación, en este caso son : cambiar el tema de la aplicación y mostrar la política de privacidad.



4.Diseño de sistemas de información

4.1 Diseño de la arquitectura del sistema

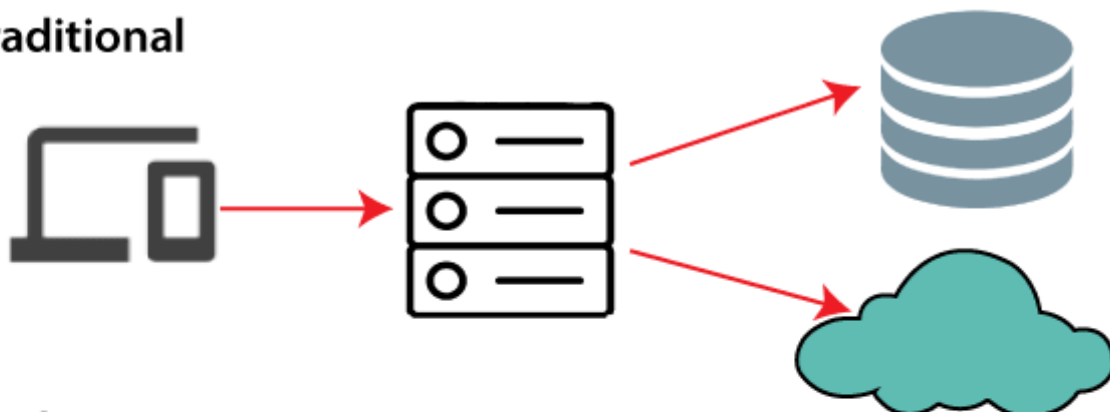
4.1.1 Descripción general del entorno tecnológico del sistema

Los usuarios con sistema operativo Android o IOS, podrán descargar e instalar SaveMoney desde Play Store en el caso de Android y App Store en el caso de IOS.

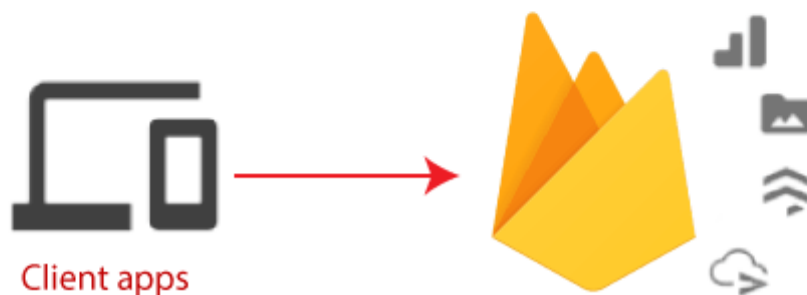
En el sistema para la parte de la gestión del almacenamiento y disponibilidad de la información usaremos una base de datos NoSQL en tiempo real de Firebase .

- Firebase Auth y Database: Una plataforma que ofrece servicios en la nube para aplicaciones web y móviles. En el presente sistema se usa en ambos, una aplicación Single Page Application con Node.js y para la aplicación android.

Traditional



Firebase



4.1.2 Catálogo de requisitos de diseño

En este caso Ionic no tiene un patrón de diseño específico, por lo tanto usaremos un patrón de diseño basado en la división por carpetas, que es uno de los que más se suele usar en este tipo de proyectos. El proyecto se podrá dividir en las siguientes carpetas:

- **Pages:** esta carpeta contendrá las pantallas que se utilizarán en la aplicación ('login', 'main', 'register'...), tanto su diseño como su funcionalidad.
- **Components:** esta carpeta contendrá componentes específicos que se usarán en distintas pantallas del proyecto.
- **Services:** esta carpeta contendrá los distintos servicios que se utilizan en la aplicación, es decir la parte que interactúa con la parte del servidor
- **Shared:** esta carpeta contendrá componentes comunes para casi todas las pantallas, como el 'header'
- **Models:** esta carpeta contendrá los distintos modelos que se usarán en la aplicación
- **Guards:** esta carpeta contendrá las restricciones a distintas rutas dependiendo de distintas condiciones.

Requisitos del modelo de datos:

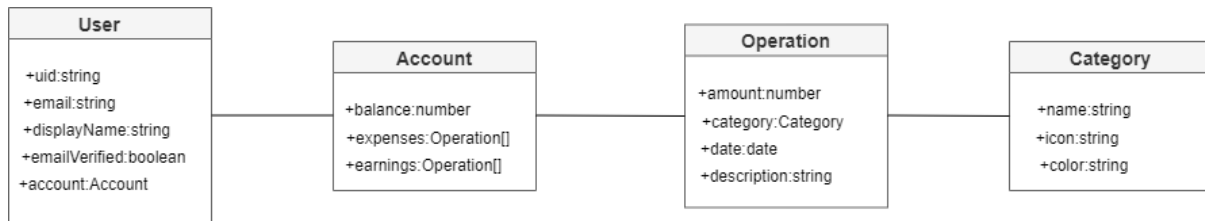
Código	Descripción	Fecha
RD1	El modelo donde se guardan los datos debe responder a un modelo de datos no relacional y en tiempo real	10/05/2021
RD2	El motor de la base de datos que se utilizará será Firestore	10/05/2021
RD3	Los datos no deben ser redundantes	10/05/2021
RD4	Ningún Usuario salvo el equipo de desarrollo y un administrador podrá tener permisos de administración sobre el SGBD	10/05/2021

Requisitos de vistas:

Código	Descripción	Fecha
RD5	Los cambios de datos a mostrar serán controlados mediante observables que se actualicen en tiempo real	10/05/2021
RD6	Los datos de entrada deben ser válidos en esta capa para evitar el acceso al modelo de datos con valores inválidos.	10/05/2021
RD7	La interfaz debe ser intuitiva y fácil de entender para el usuario	10/05/2021
RD8	La interfaz debe contener información personalizada según el usuario logado.	10/05/2021

4.2 Modelo de las clases de diseño

Entidades:



En este caso en lugar de usar clases estamos usando Interfaces, ya que es el mecanismo más común que usa Typescript para definir tipos en las clases. Aquí no se hace uso de getters y setters, ya que las variables de cada interfaz son públicas. Por lo tanto en el diagrama de clases no aparecerán estos métodos.

4.3 Modelo físico de datos

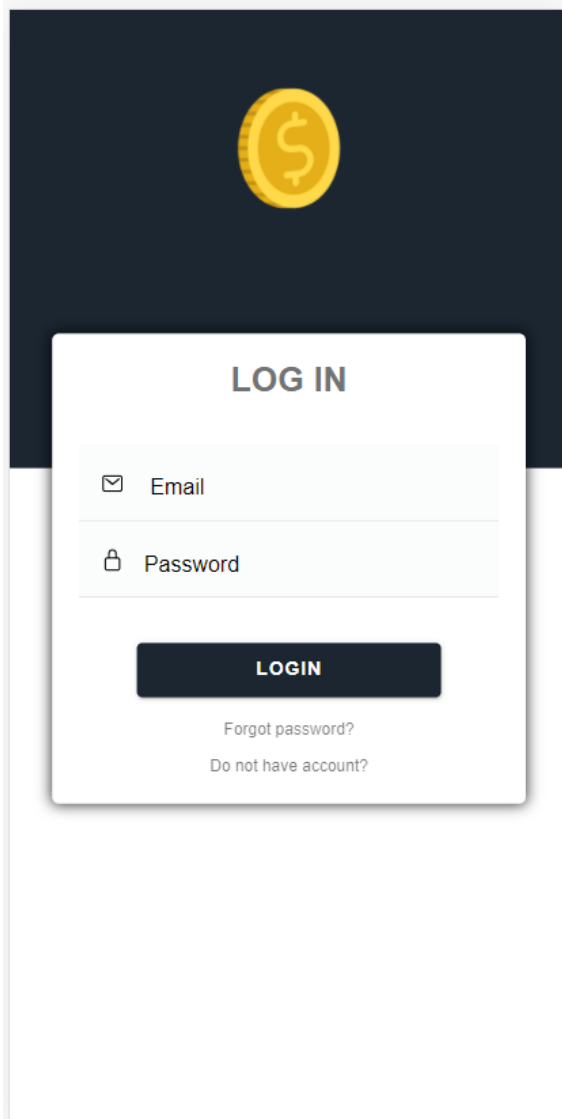
La aplicación utiliza una base de datos Firebase para almacenar información de los objetos más reconocidos y el número de veces que usuarios han iniciado el periodo de prueba gratis o iniciado la suscripción mensual. Por lo tanto, se conecta a la base de datos tanto en la aplicación android como en el monitor web en tiempo real.

A su vez, se tiene una tabla para usuarios administradores que pueden acceder al monitor en tiempo real de la aplicación. Se almacena el correo electrónico, la contraseña encriptada y la fecha en que se efectuó el registro.

Firebase Realtime es una base de datos no relacional, alojada en la nube con la que Los datos se almacenan como JSON y se sincronizan en tiempo real con cada cliente conectado. Se pueden crear tablas con JSON, con una interfaz gráfica o de programáticamente desde clientes en aplicaciones web o móviles.

Las tablas tienen configuradas las llamadas “rules”, las cuales son reglas que seguir al escribir o leer de estas. En las reglas se pueden implementar condiciones como sólo permitir acceso si se encuentra autenticado o no.

4.4 Diseño de la interfaz de usuario



The login form is centered on a dark blue background featuring a large yellow coin icon at the top. The form itself is a white card with a dark blue header labeled "LOG IN". It contains two input fields: "Email" with an envelope icon and "Password" with a lock icon. Below these is a dark blue "LOGIN" button. At the bottom, there are two links: "Forgot password?" and "Do not have account?".

LOG IN

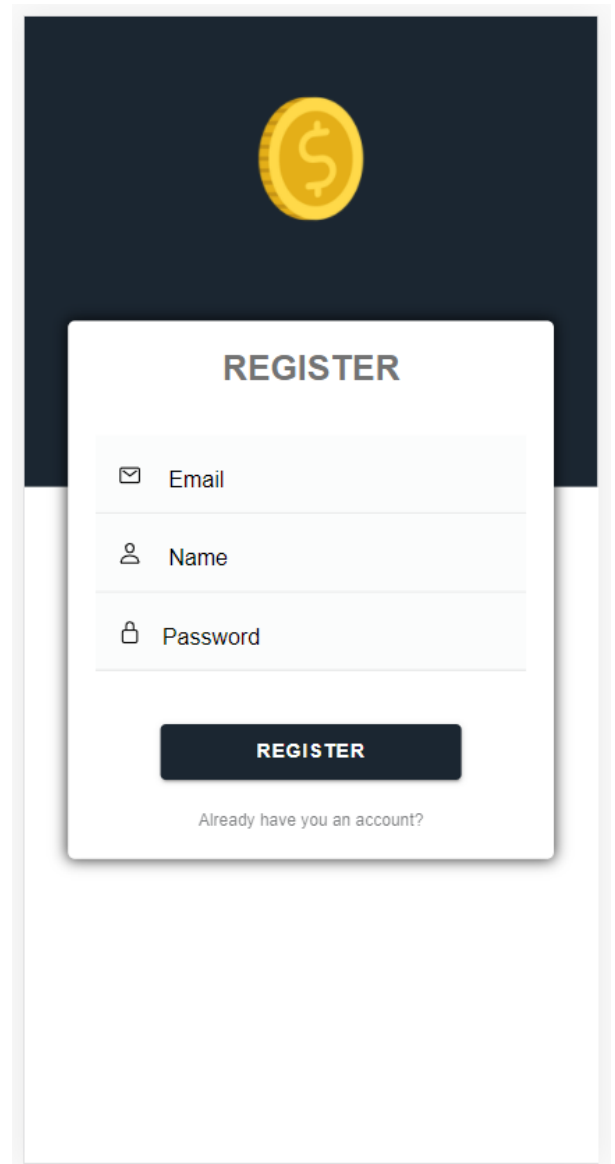
✉ Email

🔒 Password

LOGIN

[Forgot password?](#)

[Do not have account?](#)



The register form is centered on a dark blue background featuring a large yellow coin icon at the top. The form is a white card with a dark blue header labeled "REGISTER". It contains three input fields: "Email" with an envelope icon, "Name" with a person icon, and "Password" with a lock icon. Below these is a dark blue "REGISTER" button. At the bottom, there is a link: "Already have you an account?".

REGISTER

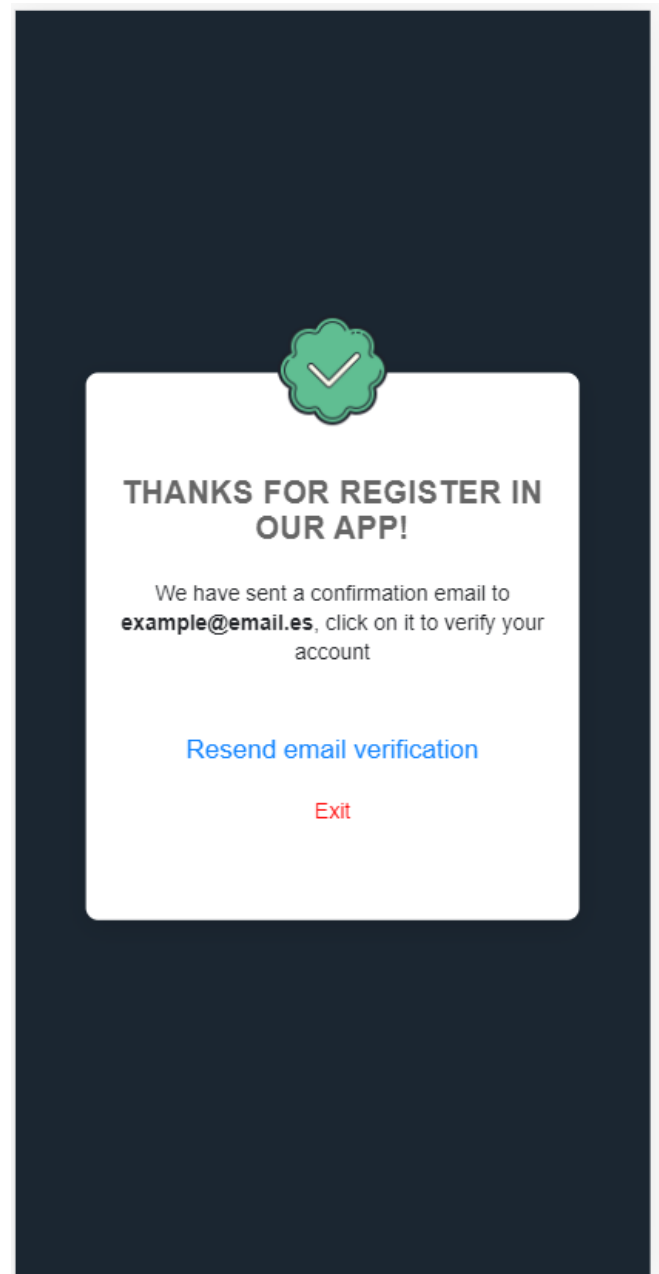
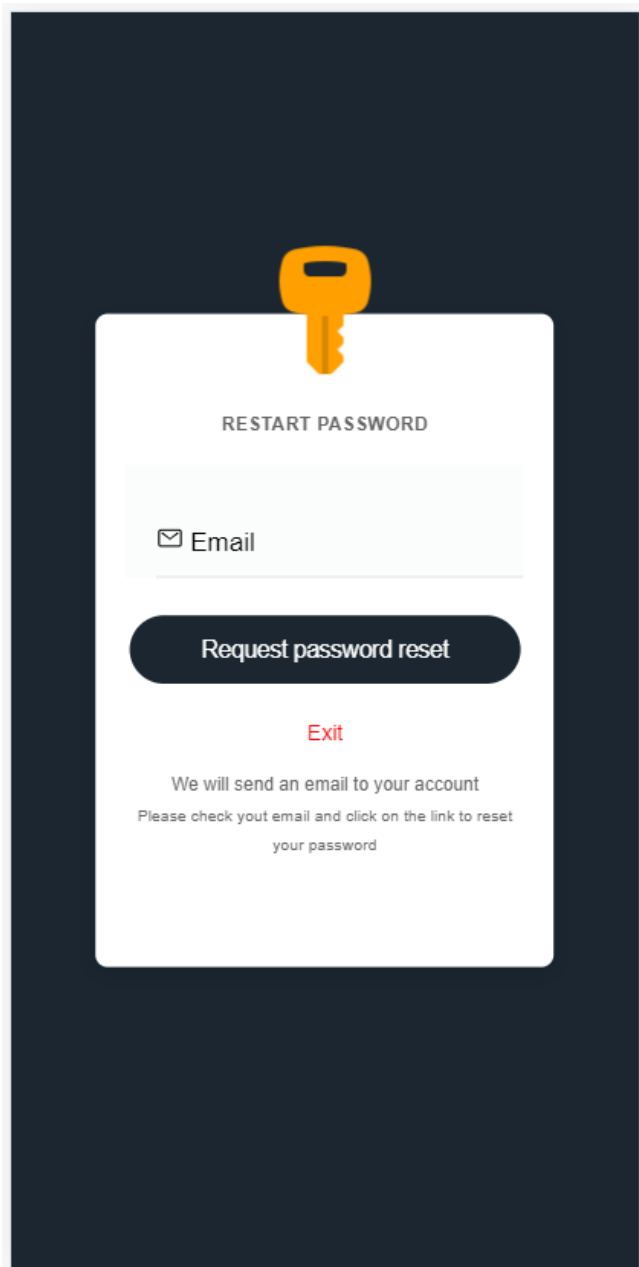
✉ Email

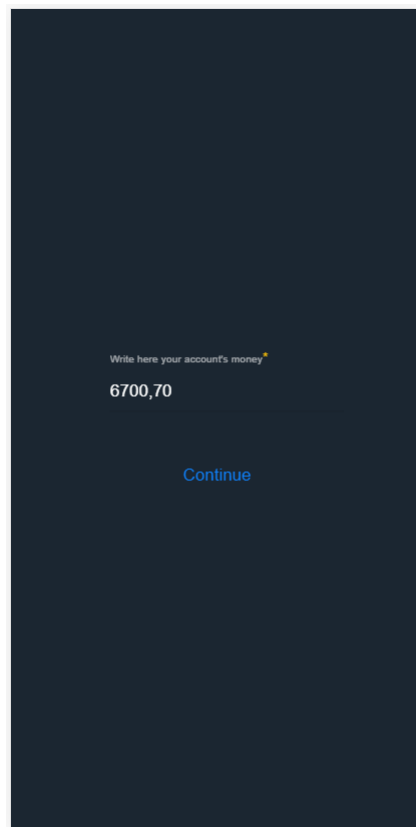
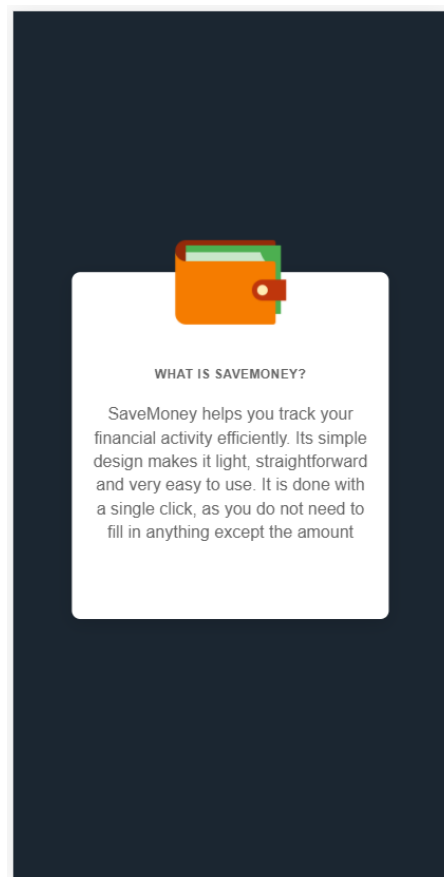
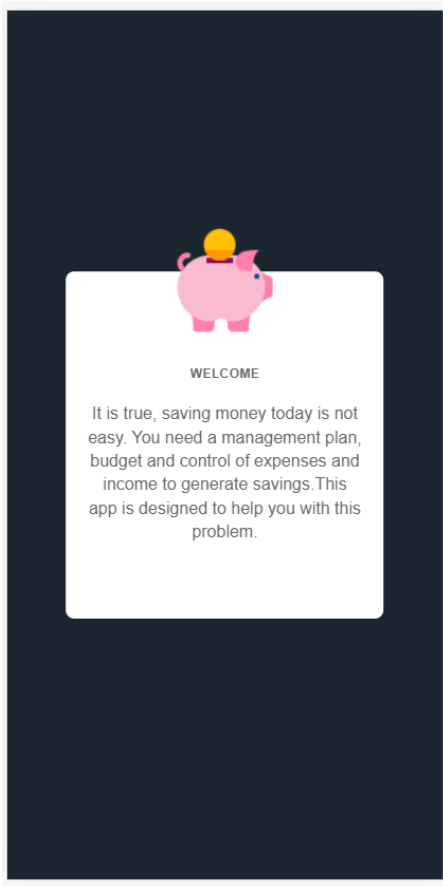
👤 Name

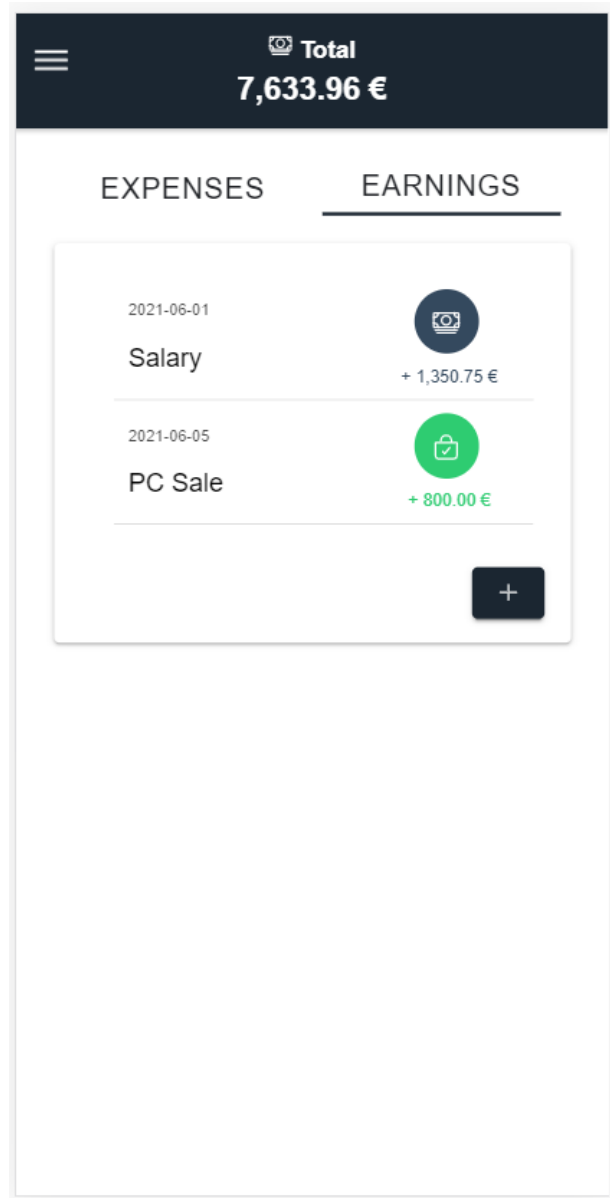
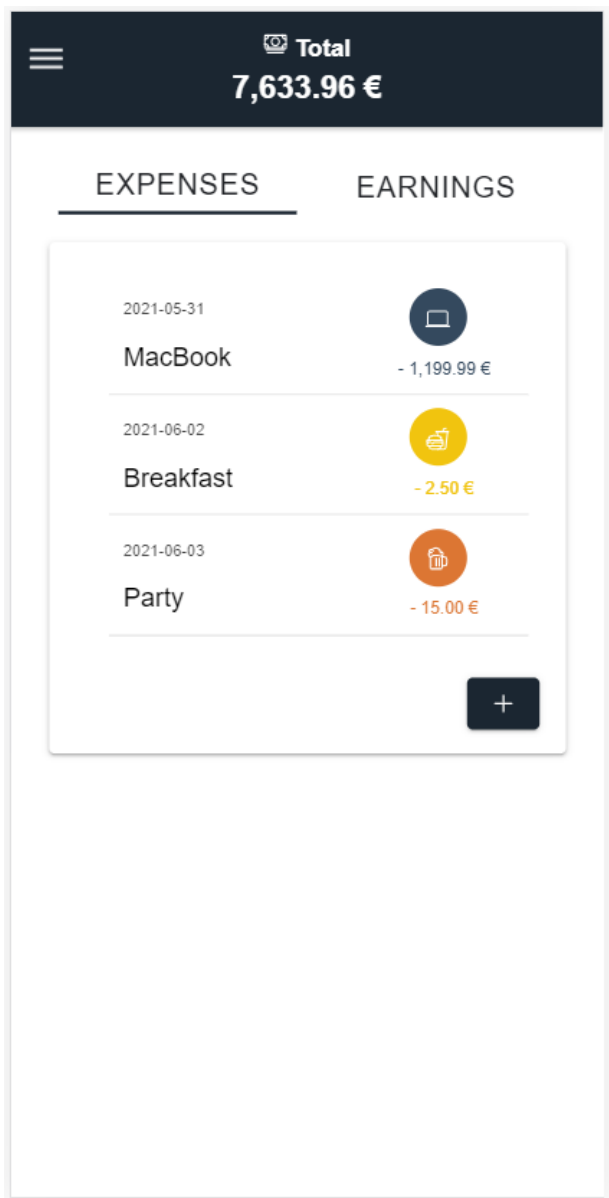
🔒 Password


REGISTER

[Already have you an account?](#)











Total
7,633.96 €


ADD EXPENSES 


Ammount (€):


Description:


Date: dd/mm/aaaa 


Category:



Transport



Gifts



Leisure



Food


Health


Shopping

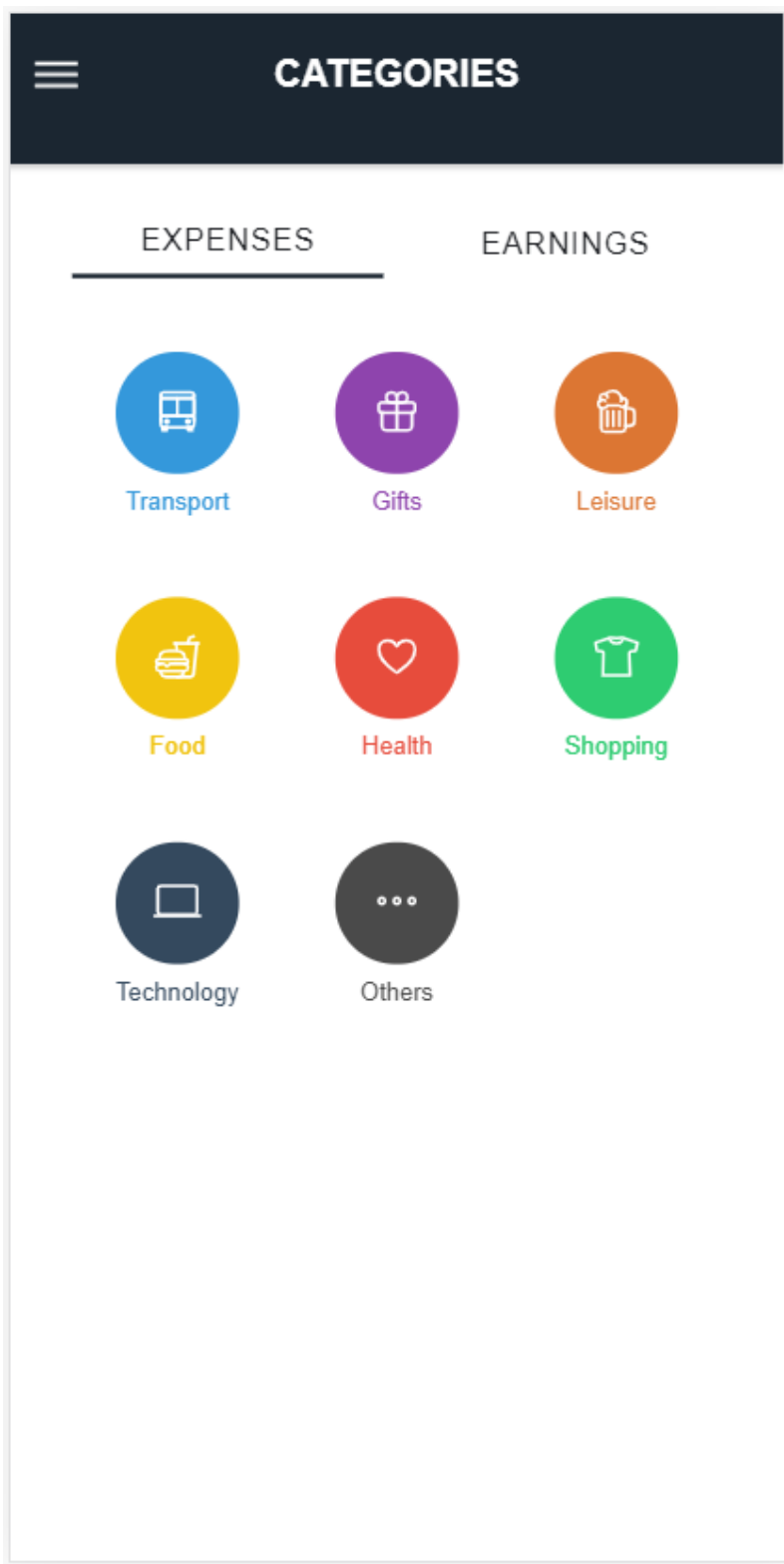

Technol...


Others



39

Vahagn Hayrapetyan Shashikyan
Curso 2020-2021





RESUME

BALANCE

7,633.96 €

EXPENSES

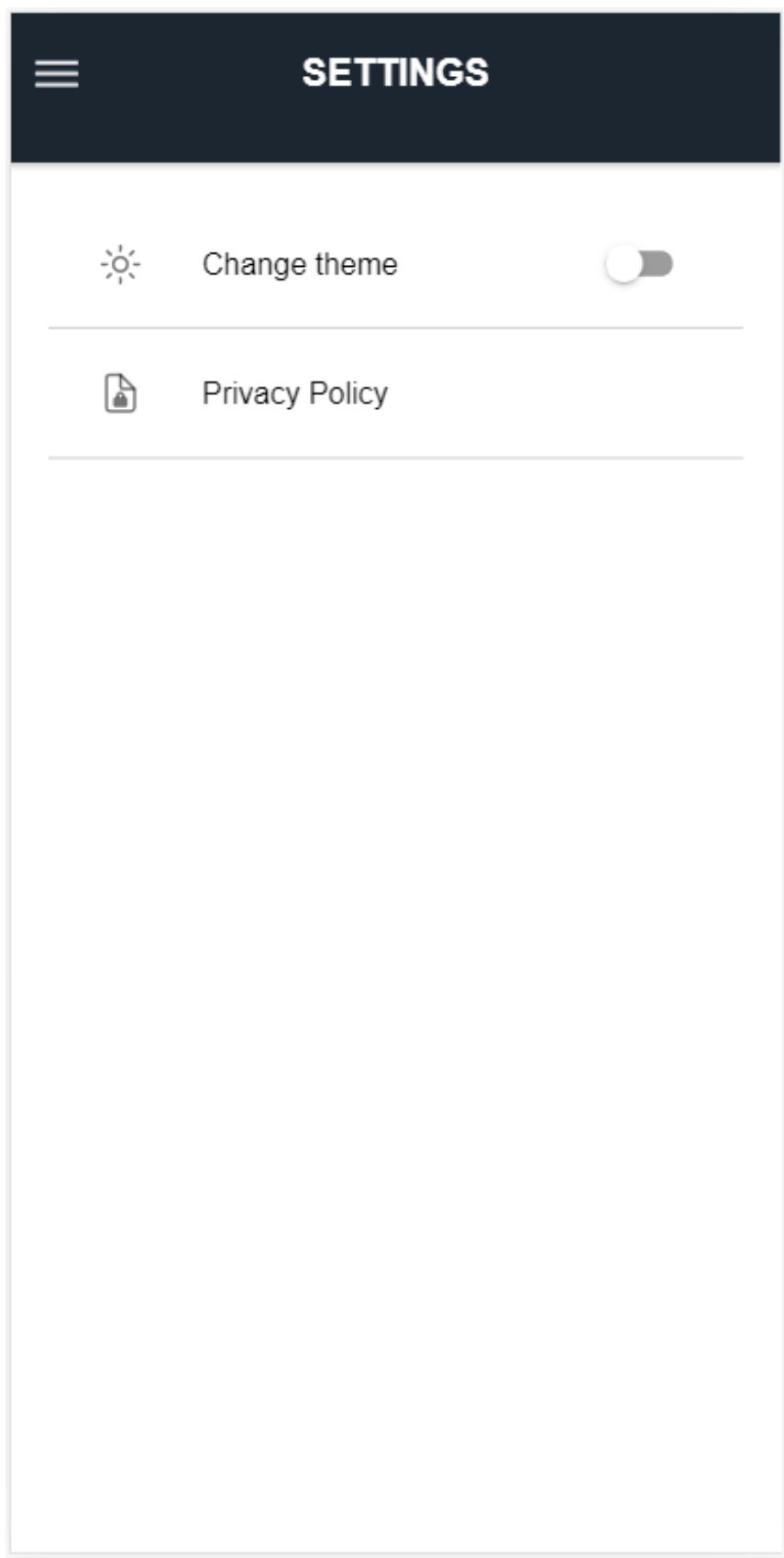
1,217.49 €

EARNINGS

2,150.75 €

NAME : Vahagn

EMAIL : vahagn015@gmail.com



4.5 Plan de migración

La aplicación no importa datos de alguna aplicación anterior. Sin embargo, es necesario que se carguen algunas dependencias antes de ser usada y estos son descargados automáticamente al abrirse por primera vez.

En la aplicación web, una vez desplegada y por consecuente, ya haber implementado todas las dependencias, se cargaran los datos de la base de datos a los gráficos de la ventana de inicio. Estos datos se actualizarán cada vez que se realicen cambios y actualicen con el uso de la aplicación móvil, en tiempo real.

4.6 Plan de pruebas

Se han desarrollado muchas más pruebas, imposible contabilizar todas.

Código	Descripción	Tipo
PR1	Prueba inicio sesión usuario registrado (verificado)	V
PR2	Prueba inicio sesión usuario no registrado	F
PR3	Prueba registro con correo no existente	V
PR4	Prueba registro con correo existente	F
PR5	Prueba registro con contraseña invalida	F
PR6	Prueba recuperación contraseña usuario registrado	V
PR7	Prueba inicio sesión usuario registrado (no verificado)	F
PR8	Prueba de registro con campos del formulario no válidos	F
PR9	Prueba de inicio de sesión con contraseña incorrecta	F
PR10	Prueba de añadir gastos con los datos del formulario válidos	V
PR11	Prueba de añadir gastos con los datos del formulario inválidos	F
PR12	Prueba de añadir ingresos con los datos del formulario válidos	V
PR13	Prueba de añadir ingresos con los datos del formulario inválidos	F
PR14	Prueba de eliminar gastos existentes	V
PR15	Prueba de eliminar ingresos existentes	V
PR16	Prueba redirección a página 'main' sin estar registrado	F
PR17	Prueba añadir saldo válido a la cuenta del usuario	V

PR18	Prueba añadir saldo inválido a la cuenta del usuario	F
PR19	Prueba cambiar el tema de claro a oscuro	V
PR20	Prueba a cambiar el tema de oscuro a claro	V
PR21	Prueba de cerrar sesión con el usuario registrado	V
PR22	Prueba de mostrar política de privacidad	V