**Descripción del proyecto: Gestor de economía para familias, amigos o trabajo.**

1. **Descripción general del proyecto**

El proyecto trata de una aplicación multiplataforma orientada a la gestión de economías y presupuestos en distintos ámbitos, como el laboral, familiar o grupos de amistades. La aplicación proporcionará una interfaz intuitiva y herramientas avanzadas para que los usuarios puedan administrar sus finanzas de manera sencilla y centralizada. A través de funcionalidades como la creación y seguimiento de presupuestos, el registro de transacciones y la generación de reportes financieros, los usuarios podrán tener un mayor control sobre sus ingresos y gastos. Además, se ofrecerán roles diferenciados dentro de los grupos, permitiendo que ciertos miembros actúen como administradores para gestionar permisos, aprobar transacciones y garantizar una gestión financiera colaborativa y organizada.

1. **Justificación de la aplicación**

Hoy en día, muchas personas no tienen en cuenta la importancia de una gestión financiera adecuada en el ámbito familiar. Según el Informe Europeo de Pagos de Consumidores de Intrum, [el 26% de los españoles no se siente seguro de su capacidad para gestionar sus finanzas personales](https://communityofinsurance.com/2024/10/01/1-de-cada-4-espanoles-se-siente-inseguro-para-gestionar-sus-finanzas-personales/). Además, [el 51% de los españoles desea gestionar mejor su presupuesto](https://www.fundssociety.com/es/formate-a-fondo/solo-el-32-de-espanoles-ve-mejoras-en-su-economia-personal-en-2024/) y reducir gastos cotidianos, lo que refleja una necesidad significativa de herramientas que faciliten esta labor.

La falta de habilidades en la administración de las finanzas personales puede llevar a errores comunes, como no ser realista con los ingresos y gastos, lo que puede resultar en presupuestos que no reflejan la verdadera situación financiera de la familia. Estos desafíos resaltan la necesidad de soluciones que ayuden a las familias a planificar y controlar sus finanzas de manera efectiva.

Por lo tanto, desarrollar una aplicación que permita a las familias gestionar sus presupuestos, monitorear sus gastos y planificar sus finanzas puede ser una herramienta valiosa para mejorar su bienestar económico y reducir el estrés asociado a la incertidumbre financiera.

1. **Objetivos de la aplicación**

**Objetivo general**

Desarrollar una aplicación multiplataforma que sirva de herramienta para que las personas puedan gestionar de manera cómoda y efectiva su vida financiera.

**Objetivos específicos**

* Permitir gestionar de manera eficiente los grupos a los que el usuario pertenece
* Autenticación en la aplicación haciendo uso de JWT
* Seguridad en la aplicación mediante encriptación de datos sensibles
* Creación sencilla de presupuestos, gastos, facturas… a un grupo
* Ofrecer a los miembros de un grupo gráficos para la visualización sencilla de la economía del grupo
* Implementación de un sistema de correos en tiempo real

1. **Opciones y funcionalidades de la aplicación**

**Opciones de administrador**

* Gestión de miembros en los grupos
* Permisos para borrar, aceptar o denegar peticiones de gastos o facturas
* Permisos para añadir, borrar o editar presupuestos del grupo
* Histórico del grupo

**Opciones de miembro**

* Petición para añadir un gasto o factura

**Opciones generales**

* Registro e inicio de sesión, implementando el inicio de sesión con Google
* Creación de un grupo personal para cada usuario registrado
* Filtros de búsqueda
* Lista de contactos
* Perfil personal con datos del usuario
* Envío de correos cuando se excedan límites puestos por el usuario

1. **Tecnologías a usar**

**Frontend:**

* **Angular** → Framework popular para el desarrollo de frontends, que permite crear aplicaciones web dinámicas y escalables con una arquitectura modular y reutilizable.

**Backend:**

* **Node.js** → Entorno de ejecución de JavaScript basado en el motor V8 de Chrome, ideal para construir aplicaciones escalables y eficientes.
* **Express.js** → Framework minimalista para Node.js que facilita la creación de API RESTful, gestionando rutas y middleware de manera eficiente.

**Base de datos:**

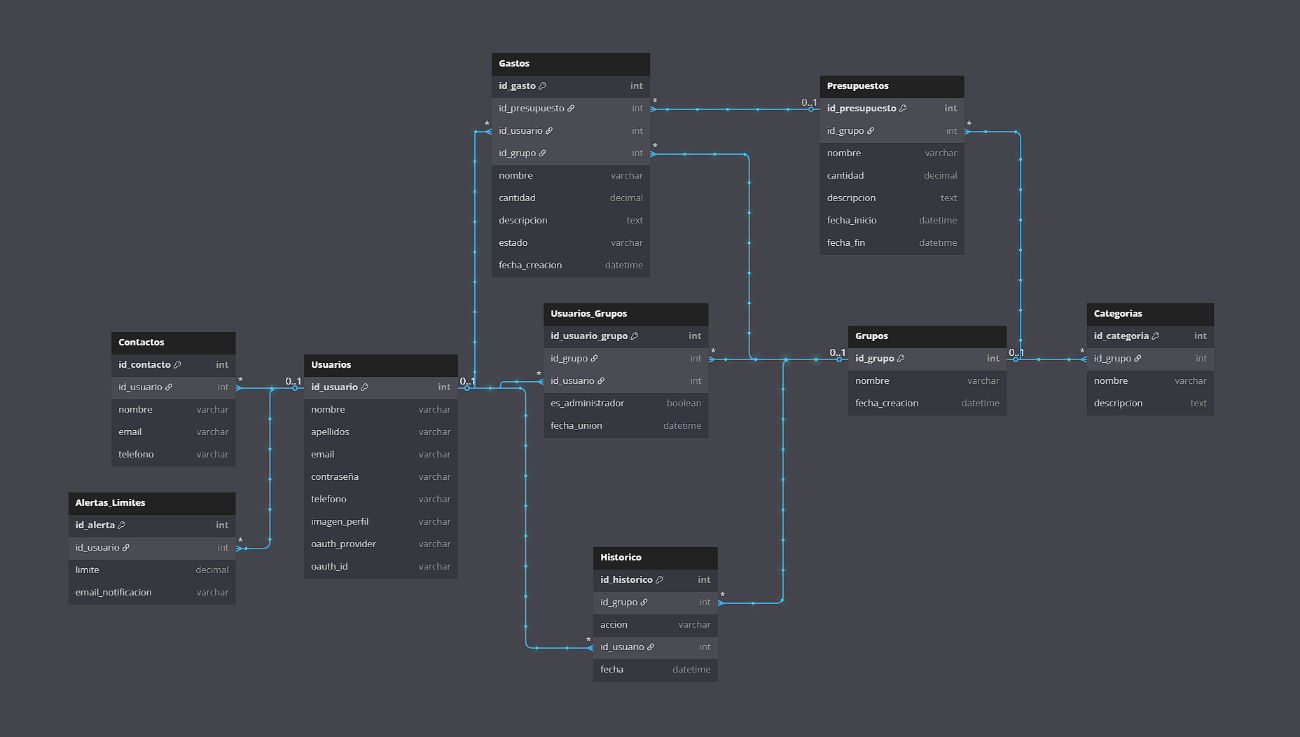
* + **MySQL** → Sistema de gestión de bases de datos relacional utilizado para almacenar y gestionar la información de la aplicación.
  + **Aiven** → Plataforma de alojamiento en la nube para bases de datos, que proporciona escalabilidad, seguridad y facilidad de gestión.
  + **MySQL Workbench** → Herramienta gráfica que permite diseñar, modelar, administrar y consultar bases de datos MySQL de manera visual.
  + **dbdiagram.io** → Herramienta de diseño de modelos entidad-relación

**Extras:**

* **JSON Web Tokens (JWT)** → Método de autenticación segura basado en tokens para gestionar sesiones de usuario.
* **GitHub** → Plataforma para el control de versiones y colaboración en el desarrollo del proyecto.
* **Vercel** → Servicios de hosting para el backend y frontend, permitiendo el despliegue continuo de la aplicación.
* **Google Charts** → API para mostrar los gráficos de los grupos
* **OAuth Client ID** → API para integrar el login con Google

1. **Base de datos**

**Modelo entidad-relación:**

****

(Archivo de imagen disponible en la entrega)

**Explicación del modelo**

* **Usuarios**

Almacena información básica y autenticación de los usuarios registrados, incluyendo nombre, email, contraseña (cifrada), teléfono e imagen de perfil. Permite también el inicio de sesión a través de proveedores OAuth (como Google).

* **Grupos**

Representa distintos grupos (familia, trabajo, amigos) que crean los usuarios para gestionar finanzas de forma colaborativa. Guarda datos esenciales como nombre y fecha de creación.

* **Usuarios\_Grupos**

Tabla intermedia que establece una relación de muchos a muchos entre usuarios y grupos. Permite asignar roles específicos (administrador o miembro) y llevar registro de cuándo un usuario se unió a un grupo.

* **Presupuestos**

Almacena presupuestos establecidos por cada grupo, especificando cantidad, descripción y periodo de vigencia (fechas de inicio y fin).

* **Gastos**

Permite registrar gastos o facturas realizados dentro de un presupuesto. Los gastos tienen un estado (pendiente, aceptado o denegado) que debe ser aprobado por administradores del grupo.

* **Categorías**

Contiene categorías para organizar presupuestos y gastos dentro de un grupo específico.

* **Contactos**

Permite que cada usuario mantenga una lista personal de contactos, facilitando la gestión de invitaciones y comunicación interna.

* **Alertas\_Limites**

Permite a cada usuario configurar alertas automáticas cuando se excedan ciertos límites de gasto, enviando notificaciones a través de correo electrónico.

* **Histórico**

Guarda un registro de las acciones importantes realizadas en un grupo (ej: creación de presupuestos, aprobación de gastos), ofreciendo trazabilidad y transparencia en la gestión financiera.

**Modelo relacional**

**Usuarios**(id\_usuario(PK), nombre, apellidos, email[UNIQUE], contraseña, telefono, imagen\_perfil, oauth\_provider, oauth\_id)

**Grupos**(id\_grupo(PK), nombre, fecha\_creacion)

**Usuarios\_Grupos**(id\_usuario\_grupo(PK), id\_usuario(FK → Usuarios.id\_usuario), id\_grupo(FK → Grupos.id\_grupo), es\_administrador, fecha\_union)

**Presupuestos**(id\_presupuesto(PK), id\_grupo(FK → Grupos.id\_grupo), nombre, cantidad, descripcion, fecha\_inicio, fecha\_fin)

**Gastos**(id\_gasto(PK), id\_presupuesto(FK → Presupuestos.id\_presupuesto), id\_usuario(FK → Usuarios.id\_usuario), id\_grupo(FK → Grupos.id\_grupo), nombre, cantidad, descripcion, estado, fecha\_creacion)

**Categorias**(id\_categoria(PK), id\_grupo(FK → Grupos.id\_grupo), nombre, descripcion)

**Contactos**(id\_contacto(PK), id\_usuario(FK → Usuarios.id\_usuario), nombre, email, telefono)

**Alertas\_Limites**(id\_alerta(PK), id\_usuario(FK → Usuarios.id\_usuario), limite, email\_notificacion)

**Historico**(id\_historico(PK), id\_grupo(FK → Grupos.id\_grupo), accion, id\_usuario(FK → Usuarios.id\_usuario), fecha)