

Examen React-Js

Gestor de Finanzas Multi-usuario

DESCRIPCIÓN GENERAL DE LA APLICACIÓN

El objetivo es construir una aplicación de finanzas personales, capaz de gestionar los datos de **varios usuarios**, cada uno con **múltiples cuentas** (ej: cuenta corriente, tarjeta de crédito, ahorros).

Se entrega un proyecto React + Vite preparado con un diseño básico y 4 archivos JSON con una estructura relacional en `src/data/`:

- `users.json` : Lista de los usuarios de la aplicación (incluye foto y dirección).
- `accounts.json` : Lista de todas las cuentas. Cada cuenta pertenece a un usuario (`userId`).
- `transactions.json` : Lista de todas las transacciones. Cada transacción pertenece a una cuenta (`accountId`).
- `categories.json` : Un **objeto** que mapea IDs de categorías a sus detalles (incluye un icono).

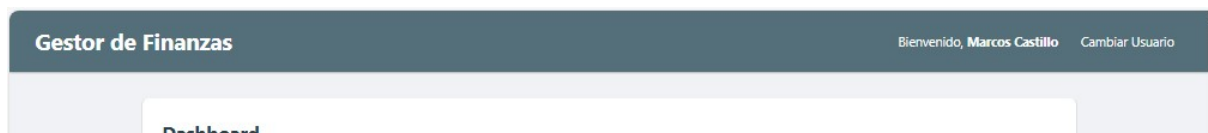
REQUISITOS FUNCIONALES

1. Pantalla de Selección de Usuario

- Al iniciar la aplicación, la primera vista debe ser una **pantalla de bienvenida** que muestre una parrilla de tarjetas, una por cada usuario.
- Cada tarjeta de usuario debe mostrar su **foto, nombre completo y dirección**.
- Cada tarjeta debe tener un botón **"Acceder"**. Al hacer clic en él, la aplicación debe cargar la vista principal de finanzas para ese usuario específico.



- Una vez dentro de la vista principal, la cabecera debe mostrar el nombre del usuario y un botón **"Cambiar Usuario"** para poder regresar a la pantalla de selección.
-



2. Dashboard Interactivo

- El Dashboard es el primer componente que se ve tras seleccionar un usuario.
- Por defecto, debe mostrar **únicamente el "Patrimonio Neto"** del usuario (la suma de los balances de todas sus cuentas).
- También debe mostrar una lista de las cuentas del usuario con su nombre y balance actualizado.

Dashboard	
PATRIMONIO NETO	
12.090,00 €	
Cuentas	
Mi Banco Personal	6490,00 €
Ahorros Futuro	6000,00 €
Tarjeta de Crédito Viaje	-400,00 €

3. Formulario Modal para Nuevas Transacciones

- El formulario para añadir transacciones **no debe ser visible por defecto**.
- Debe aparecer dentro de una **ventana modal**.
- Este modal se activará al hacer clic en un nuevo botón **"Añadir Transacción"** que debe estar situado en la cabecera de la sección "Historial de Transacciones".
- El formulario debe contener los mismos campos (descripción, monto, tipo, cuenta y categoría).
- Las categorías a mostrar dependen de si son Gastos o Ingresos
- Los campos han de estar debidamente validados.
- La fecha será la del día en curso.
- Al añadir la transacción con éxito, el modal debe **cerrarse automáticamente**. El usuario también puede cerrar el modal con un botón de cierre (X) o haciendo clic fuera del diálogo.
-

Nueva Transacción

Descripción

Monto

Tipo

Gasto

Cuenta

Mi Banco Personal

Categoría

Selecciona una categoría

Añadir Transacción

4. Historial de Transacciones Detallado

- * La cabecera de esta sección debe contener el título y el botón **"Añadir Transacción"**.
- * Debe mostrar la lista de transacciones con filtro por **cuenta**.
- * Cada transacción debe mostrar la máxima información posible, incluyendo el **icono de la categoría**, su nombre, la cuenta a la que pertenece y la fecha.
- * Debe mantener la funcionalidad de **eliminar** una transacción, botón (X).

Dashboard

PATRIMONIO NETO

12.090,00 €

Cuentas

Mi Banco Personal

6490,00 €

Ahorros Futuro

6000,00 €

Tarjeta de Crédito Viaje

-400,00 €

Historial de Transacciones

[Añadir Transacción](#)

Todas las Cuentas



Vuelo a Roma

2/12/2025 | Tarjeta de Crédito Viaje

-250,00 € x



Sueldo Marcos Diciembre

1/12/2025 | Mi Banco Personal

+2400,00 € x



Ingreso extra proyecto web

15/11/2025 | Ahorros Futuro

+500,00 € x



Compra semanal Carrefour

8/11/2025 | Mi Banco Personal

-110,00 € x



Sueldo Marcos

1/11/2025 | Mi Banco Personal

+2400,00 € x