

Alumnos: **Clase: JUEVES-NOCHE 12-02-2026**

- Maria Clara Acosta
- Daniela Escobar (miércoles-noche)
- Ema Luna Salazar

PROMPTS

Prompt 1: Tengo que hacer un proyecto con las siguientes reglas.

Tema: Sistema de Gestión Financiera Multiperfil

Objetivo General: Crear una aplicación web que permita administrar ingresos, gastos, presupuestos y reportes financieros utilizando almacenamiento en localStorage.

Nivel: Últimos semestres – Simulación profesional sin backend.

Tecnologías Obligatorias:

- HTML5
- CSS Framework (Bootstrap o Tailwind)
- JavaScript Vanilla
- localStorage Roles del Sistema:

Usuario:

- Registrar ingresos y gastos.
- Crear categorías personalizadas.
- Definir presupuesto mensual.
- Visualizar balance general.
- Consultar historial filtrado por fecha o categoría.

Administrador Financiero:

- Visualizar datos de todos los usuarios.
- Generar reportes generales.
- Identificar usuarios con sobre presupuesto.
- No puede modificar registros personales.

Funcionalidades Obligatorias:

- CRUD completo de movimientos financieros.
- Cálculo automático de balance.
- Filtros dinámicos.
- Panel de estadísticas.
- Persistencia total en localStorage.
- Protección de vistas según rol.

necesitamos tener:

1-una estructura visual limpia, pero saliendo de los matices básicos, manteniendo un tono empresarial

2-funcionalidad a largo plazo.

3-buenas prácticas, evitar un código espagueti, usando las bases de POO (Programación orientada a objetos).

4-archivos que cada 1 maneje la lógica de una función o clase o en caso de que un archivo necesite manejar más de una lógica que sea todo encapsulado dentro de la misma función que tiene el archivo

Prompt 2: Necesito que te pongas en el rol de un analista de código profesional, revises que problemas está teniendo en este momento que yo he notado varios y nos expliques cada uno, con su respectivo bugfix y como fue el proceso de debugueo.

Prompt 3: Revisa las funcionalidades definidas para cada rol (Usuario y Administrador Financiero) y verifica si están completas, coherentes y funcionales. Identifica qué falta, qué debe mejorarse y qué requisitos adicionales se necesitan para que cada ítem pueda implementarse correctamente en un sistema real.

ANÁLISIS CRÍTICO:

El proyecto “Sistema de Gestión Financiera Multiperfil” cumple con los requisitos funcionales establecidos, ya que permite registrar ingresos y gastos, gestionar categorías, definir presupuestos, calcular balances automáticamente, aplicar filtros dinámicos y diferenciar vistas según rol utilizando localStorage como mecanismo de persistencia. A nivel académico, la aplicación logra simular un entorno profesional sin backend y presenta una estructura organizada basada en Programación Orientada a Objetos y separación de archivos.

Sin embargo, al analizarlo críticamente, se evidencian limitaciones importantes. El uso exclusivo de localStorage implica que no existe seguridad real ni autenticación robusta; el control de roles es únicamente visual y puede ser manipulado desde las herramientas del navegador. Además, no se observa un manejo sólido de validaciones, control de errores o protección ante corrupción de datos, lo que reduce la confiabilidad del sistema en escenarios reales. Aunque se intenta mantener modularidad y evitar código espagueti, la arquitectura no responde claramente a un patrón estructural definido y la lógica de negocio puede quedar acoplada a la manipulación del DOM.

El hecho de haber sido desarrollado 100% con inteligencia artificial permitió cumplir rápidamente con los requerimientos, pero también genera una dependencia tecnológica y una posible falta de comprensión profunda del diseño interno. En conclusión, el proyecto es funcional y adecuado para fines académicos, pero requeriría mejoras en seguridad, validación, arquitectura y escalabilidad para considerarse una solución profesional sólida.