



Evaluación Sumativa – Unidad 2

1. Datos de Identificación del Estudiante y la Práctica

Nombre del estudiante(s)	Freddy Alexander Matailo Mora Stalin Joel Tapia Pinta
Asignatura	Desarrollo de Plataformas
Ciclo	Quinto
Unidad	1
Título de la Práctica	Aplicaciones Web Tecnologías de Lado del Cliente
Nombre del Docente	Edison Coronel
Fecha	Jueves 8 de enero del 2026

2. Informe Técnico de Desarrollo Frontend: GastanGO

1. Introducción:

El presente informe detalla el desarrollo de la segunda capa del sistema multiplataforma "GastanGO". Se presenta la construcción de una interfaz web moderna, funcional y responsive, diseñada bajo el modelo de Single Page Application (SPA). El objetivo principal es proporcionar a los usuarios una herramienta intuitiva para la gestión de ingresos y gastos, conectada en tiempo real con el servicio backend desarrollado previamente.

2. Diseño e Implementación del Frontend

2.1 Tecnologías Utilizadas

Para cumplir con los requisitos de una interfaz moderna y escalable, se seleccionó el siguiente stack tecnológico:

Framework SPA: React (con Vite para optimización de entorno de desarrollo).

Estilos y Diseño Responsivo: Tailwind CSS, permitiendo una maquetación rápida y adaptativa (Mobile First).

Gestión de Estado: React Context API (AuthContext y TransactionContext) para manejar la sesión del usuario y el flujo de datos globalmente.

Visualización de Datos: Recharts para la generación dinámica de gráficos estadísticos.

Cliente HTTP: Axios para la comunicación con la API REST.

2.2 Diseño de Interfaz y Experiencia de Usuario (UX)



UNL

Universidad
Nacional
de Loja

FEIRNNR - Carrera de Computación

Se ha implementado un diseño limpio y coherente con los objetivos del proyecto.

- **Paleta de Colores:** Se estableció un color primario Azul Institucional (#2563EB) para acciones principales y marca, complementado con un fondo gris claro (#F6F8FB) para reducir la fatiga visual. Se utilizan colores semánticos para las finanzas: Verde Esmeralda (#10B981) para ingresos y Rojo Suave (#EF4444) para gastos.
- **Accesibilidad (WAI-ARIA):** Los formularios cuentan con etiquetas claras, navegación por teclado habilitada y contrastes de color verificados para asegurar la legibilidad.
- **Estructura Visual:**
 - Layout Principal: Navegación superior fija con acceso rápido a módulos y perfil de usuario.
 - Dashboard: Implementación de tarjetas (Cards) para KPIs, gráficos de balance mensual y listado de últimas transacciones.

The screenshot shows the GastaGO dashboard. At the top, there are four cards: Saldo Total (\$143.80), Ingresos (+\$165.00), Gastos (-\$21.20), and an advertisement for the APP MÓVIL. Below these are two main sections: 'Últimos Movimientos' (Recent Movements) and 'Balance Anual' (Annual Balance). The 'Últimos Movimientos' section lists transactions like SALARIO (INGRESOS, +\$150.00), Gasto en Transporte (TRANSPORTE, -\$1.20), Gasto en Comida (COMIDA, -\$5.00), APUESTA (INGRESOS, +\$15.00), and Gasto en Ropa (ROPA, -\$15.00). The 'Balance Anual' section shows a bar chart with a large green bar for 'Ene' and a small red bar for 'Dici'. To the right, there's a 'Top Gastos' chart showing Ropa, Comida, and Transporte.

This screenshot shows the 'Registrar Ingreso' (Register Income) screen. It has a header '← Atrás' and 'Registrar Ingreso'. It displays an 'INGRESOS \$ 10' field and an 'APUESTA' field. Below is a 'Confirmar Transacción' (Confirm Transaction) dialog with fields for Categoría (Detaille), INGRESOS (APUESTA), and values 'Total' and '\$10.00'. Buttons for 'Editar' and 'Guardar' are at the bottom, along with a numeric keypad and a 'Confirmar' button.



UNL

Universidad
Nacional
de Loja

FEIRNNR - Carrera de Computación

← Atrás

Selecciona la Categoría



Comida



Transporte



Hogar



Entretenimiento



Salud



Ropa



Educación



Ahorros



Salario



Inversiones

← Atrás

Registrar Gasto

ENTRETENIMIENTO

\$ 10

Confirmar Transacción

Categoría	Entretenimiento
Detalle	GASTO REGISTRA...
Total	\$10.00

Editar

Guardar

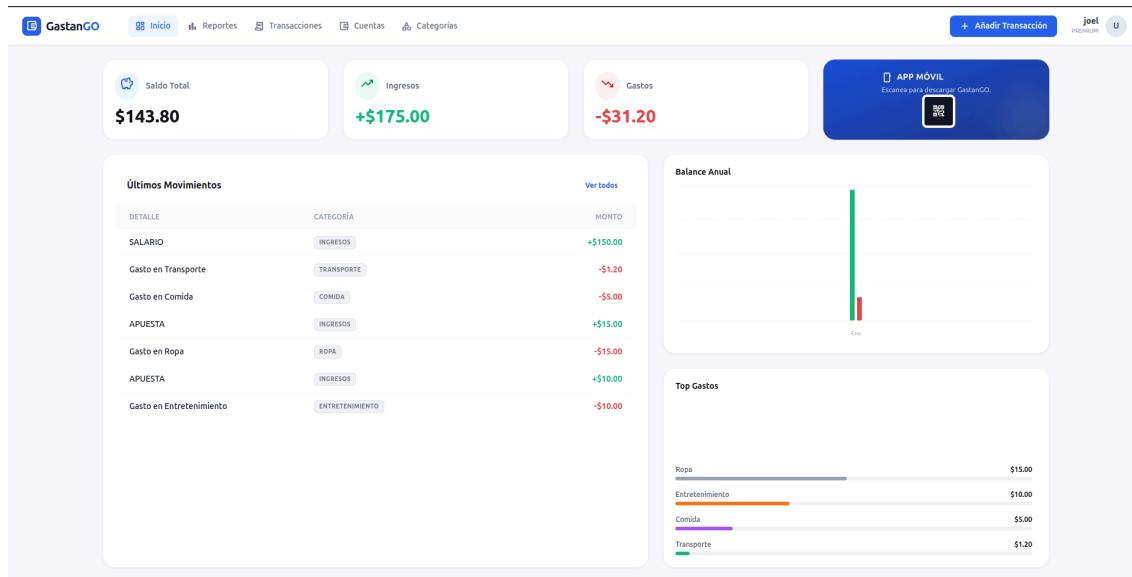
Confirmar



UNL

Universidad
Nacional
de Loja

FEIRNNR - Carrera de Computación



3. Conexión e Integración con el Backend

3.1 Comunicación Cliente-Servidor

La aplicación consume los endpoints REST del backend desarrollado en la Unidad 1. Se implementó una instancia de Axios centralizada (api.js) que maneja:

1. Interceptores: Inyección automática del token JWT en el header Authorization de cada petición.
2. Manejo de Errores: Gestión centralizada de respuestas 401 (No autorizado) y errores de servidor.

3.2 Flujos de Datos Dinámicos

Autenticación: El sistema valida credenciales contra el endpoint /auth/login, almacena el token en localStorage y persiste la sesión mediante AuthContext.

Transacciones: El Dashboard realiza peticiones GET al endpoint /transactions. Los datos recibidos se procesan en el frontend para calcular en tiempo real:

- Saldo Total, Ingresos y Gastos.
- Gráficos de barras por mes.
- Barras de progreso por categoría.

4. Ampliación del Modelo C4 - Arquitectura del Sistema

A continuación, se presentan los diagramas actualizados que integran el frontend en la arquitectura global, cumpliendo con el requisito de documentación arquitectónica.

4.1. Diagrama de Contexto (Nivel 1)

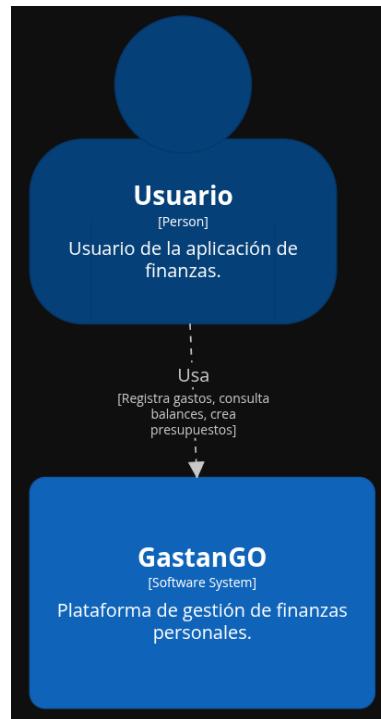


UNL

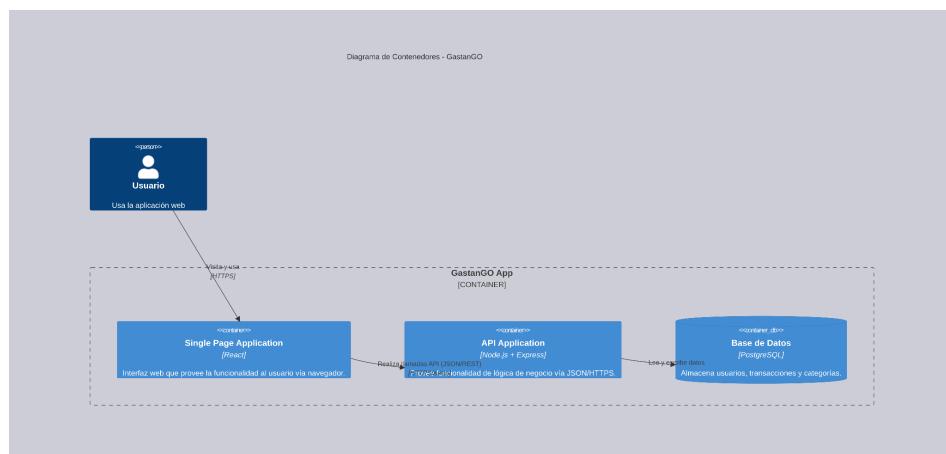
Universidad
Nacional
de Loja

1859

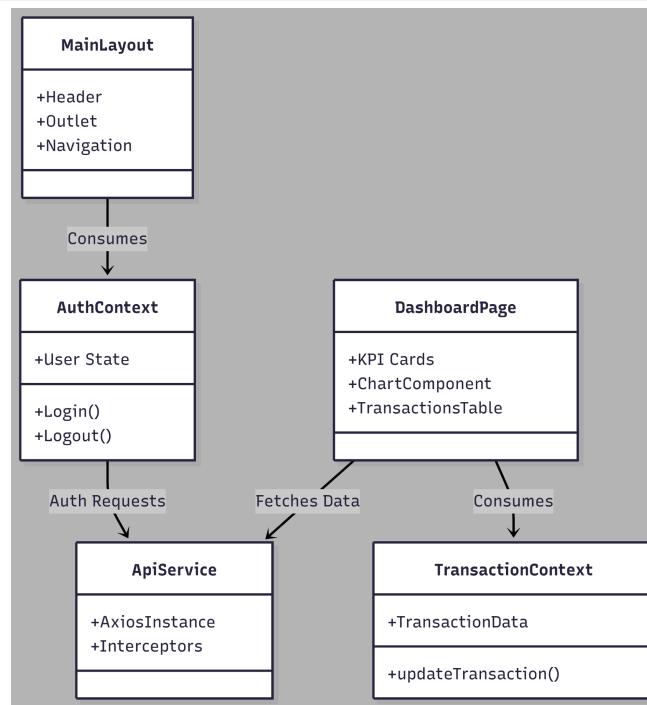
FEIRNNR - Carrera de Computación



4.2 Diagrama de Contenedores (Nivel 2 - Actualizado)



4.3. Diagrama de Componentes del Frontend (Nivel 3)

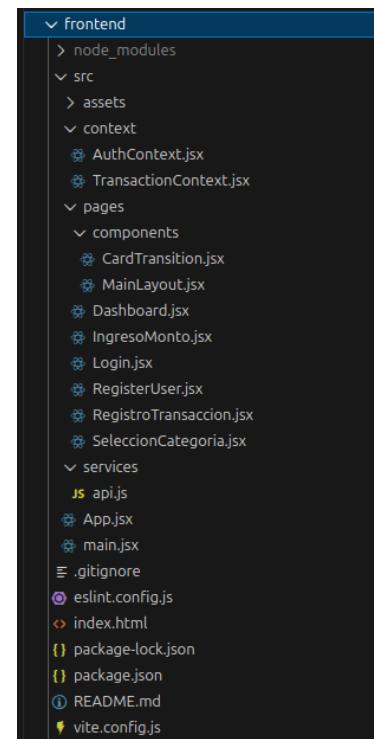


5. Documentación Técnica

5.1. Estructura del Proyecto

El proyecto sigue una arquitectura modular basada en características y responsabilidades:

```
/src
  /assets      # Recursos estáticos
  /context     # Estados globales
  /pages       # Vistas principales
    /components # Componentes reutilizables
  /services     # Configuración de API y peticiones HTTP
  App.jsx      # Enrutamiento principal
  main.jsx     # Punto de entrada
```





UNL

Universidad
Nacional
de Loja

FEIRNNR - Carrera de Computación

6. Conclusión

El desarrollo del frontend de GastanGO ha cumplido satisfactoriamente con los requerimientos funcionales y no funcionales. La integración de React con el backend existente permite una experiencia de usuario fluida y dinámica. La arquitectura C4 documentada demuestra una clara separación de responsabilidades, y el uso de GitFlow asegura la mantenibilidad del código para futuras iteraciones, incluyendo la próxima integración móvil.