

UNIVERSIDAD TECNOLÓGICA DE TEHUACÁN

Equipo 7:

**Javier Hernández Méndez
Juan Andrés Medina González
Oscar Flores Cerqueda
Carlos Alberto Pacheco Avila
Carlos Eduardo Vázquez
Muños**

Docente:

José Miguel Carrera Pacheco

**ING Desarrollo y
Gestión de Software**

Bloque 1 – Fundamentos del Proyecto

1. Diferencia entre Página Web y Aplicación Web

Característica	Página Web (Sitio Web)	Aplicación Web (Web App)
Objetivo	Informar. El consumo es pasivo (leer, ver).	Realizar tareas. La interacción es activa (crear, manipular datos).
Interactividad	Baja. Menús, enlaces, formularios simples de contacto.	Alta. Edición en tiempo real, cálculos, gestión de usuarios.
Carga de datos	Suele ser estática o con actualizaciones poco frecuentes.	Dinámica. Los datos cambian constantemente según la acción del usuario.
Ejemplo	Un blog personal, el sitio de un restaurante, un portal de noticias.	Gmail, Trello, Spotify Web, Canva.

2. Ejemplos Reales de Aplicaciones Web Profesionales

- Figma: Herramienta de diseño colaborativo. Resuelve la necesidad de diseñar interfaces en equipo sin instalar software pesado.
- Slack: Plataforma de comunicación. Resuelve la fragmentación de correos electrónicos centralizando chats por canales.
- Google Docs: Procesador de texto. Resuelve la gestión de versiones y colaboración simultánea en documentos.

3. ¿Qué tipo de problemas se resuelven con software?

El software profesional generalmente ataca tres tipos de problemas:

- Optimización de procesos: Automatizar tareas repetitivas (ej. facturación automática).
- Centralización de información: Evitar datos dispersos en papeles o excels (ej. un CRM de ventas).
- Conectividad y Comunicación: Romper barreras geográficas o de tiempo (ej. Zoom o plataformas de educación a distancia).

4. Arquitectura General de Aplicaciones Web

- Frontend (Cliente):

- Es la parte visual con la que interactúa el usuario.
- Tecnologías: HTML, CSS, JavaScript (React, Angular, Vue).
- Responsabilidad: UX/UI, capturar inputs, mostrar datos.
- Backend (Servidor):
 - Es el "cerebro" que procesa la lógica.
 - Tecnologías: Python (Django/Flask), Node.js, Java, PHP.
 - Responsabilidad: Autenticación, cálculos, conexión con la base de datos, APIs.
- Infraestructura / Entornos:
 - Base de Datos: Donde se guarda la información (MySQL, PostgreSQL, MongoDB).
 - Servidores/Nube: Donde vive la aplicación (AWS, Azure, Google Cloud, Vercel).
 - Entornos: *Desarrollo* (local), *Staging* (pruebas) y *Producción* (versión final pública).

5. Análisis de Plataformas Similares (Guía de trabajo)

Plataforma 1: Mint

- Control automático de gastos.
- Reportes financieros.
- Enfoque en análisis.

Relación con el proyecto:

Sirve como referencia para la visualización de reportes.

Plataforma 2: YNAB

- Planeación basada en presupuesto.
- Control detallado.
- Enfoque educativo.

Relación con el proyecto:

Inspira la gestión por categorías y presupuestos.

Bloque 2 – Arquitectura de Información y Accesibilidad

1. Arquitectura de Información (AI)

Es el arte y la ciencia de organizar y etiquetar el contenido para que los usuarios encuentren lo que buscan. No es el diseño visual, sino el plano estructural.

- Objetivo: Evitar la carga cognitiva excesiva. Si el usuario tiene que "pensar" dónde está el botón, la AI ha fallado.

2. Jerarquías de Contenido

Define qué es lo más importante en una pantalla. Se logra mediante:

- Tamaño: Los títulos H1 son más grandes que los párrafos.
- Posición: Lo más crítico suele estar arriba a la izquierda (patrón de lectura en F).
- Contraste: Los botones de acción (CTA) deben resaltar sobre el fondo.

3. Patrones de Navegación Web

Son soluciones estándar que los usuarios ya saben usar intuitivamente:

- Barra de navegación superior (Navbar): Común en escritorio.
- Menú de hamburguesa: Estándar en móviles para ocultar opciones.
- Breadcrumbs (Migas de pan): Rastro visual (ej. *Inicio > Hombres > Zapatos > Deportivos*).
- Footer: Para información legal, contacto y mapa del sitio.

4. Accesibilidad: Navegación por Teclado y Tabulación

Un sitio profesional debe poder usarse sin mouse.

- Orden de Tabulación (Tab Order):
 - Al presionar la tecla Tab, el foco debe moverse de manera lógica (generalmente de izquierda a derecha y de arriba a abajo).
 - Nunca debe saltar aleatoriamente de un botón del final a uno del inicio.
- Indicadores de Foco (Focus States):
 - Es el borde (generalmente azul o punteado) que aparece alrededor de un elemento cuando se selecciona con el teclado. Nunca se debe eliminar con CSS (`outline: none`) a menos que se reemplace por otro estilo visual claro.
- Accesibilidad sin Mouse:

- Es vital para personas con discapacidades motoras que usan teclados adaptados o lectores de pantalla.
- Todos los elementos interactivos (botones, formularios) deben ser accesibles vía teclado (usando Enter o Espacio para activar).

Definición del Proyecto

Aplicación Web para el Control de Gastos Personales

1. Definición del problema real

Muchas personas no llevan un registro estructurado y constante de sus gastos diarios, lo que provoca descontrol financiero, dificultad para ahorrar y una mala toma de decisiones económicas.

Este problema es común en estudiantes, trabajadores independientes y personas que administran su dinero de forma informal, ya que suelen depender de métodos poco confiables como notas en papel o aplicaciones complejas que no se adaptan a sus necesidades reales.

La falta de una herramienta sencilla, accesible y centralizada impide que los usuarios tengan una visión clara de sus finanzas personales y de sus hábitos de consumo.

2. Usuarios reales identificados

La aplicación web está dirigida a usuarios reales con necesidades concretas:

- Estudiantes universitarios que necesitan administrar su dinero mensual y controlar gastos recurrentes.
- Trabajadores independientes o freelancers que no cuentan con herramientas formales para el control de sus finanzas personales.
- Personas adultas que desean visualizar y organizar sus gastos de manera sencilla sin conocimientos técnicos avanzados.

Estos usuarios requieren una solución clara, fácil de usar y accesible desde cualquier dispositivo con navegador web.

3. Alcance inicial del proyecto (MVP)

Para asegurar la viabilidad del proyecto y permitir su evolución futura, se define un Producto Mínimo Viable (MVP) con las siguientes funcionalidades:

Funcionalidades incluidas en el MVP:

- Registro e inicio de sesión de usuarios.
- Registro de gastos personales (monto, categoría y fecha).
- Visualización de una lista de gastos registrados.
- Edición y eliminación de gastos.

- Visualización básica del total de gastos.

Funcionalidades excluidas del MVP:

- Integraciones con cuentas bancarias.
- Inteligencia artificial o análisis predictivo.
- Pagos en línea.
- Reportes financieros avanzados.

Esta delimitación permite desarrollar un sistema funcional, mantenible y escalable desde sus primeras versiones.

4. Justificación de la solución como aplicación web

Se opta por desarrollar una **aplicación web** debido a que permite el acceso multiplataforma desde cualquier navegador sin necesidad de instalación, facilitando su uso en computadoras, tabletas y dispositivos móviles.

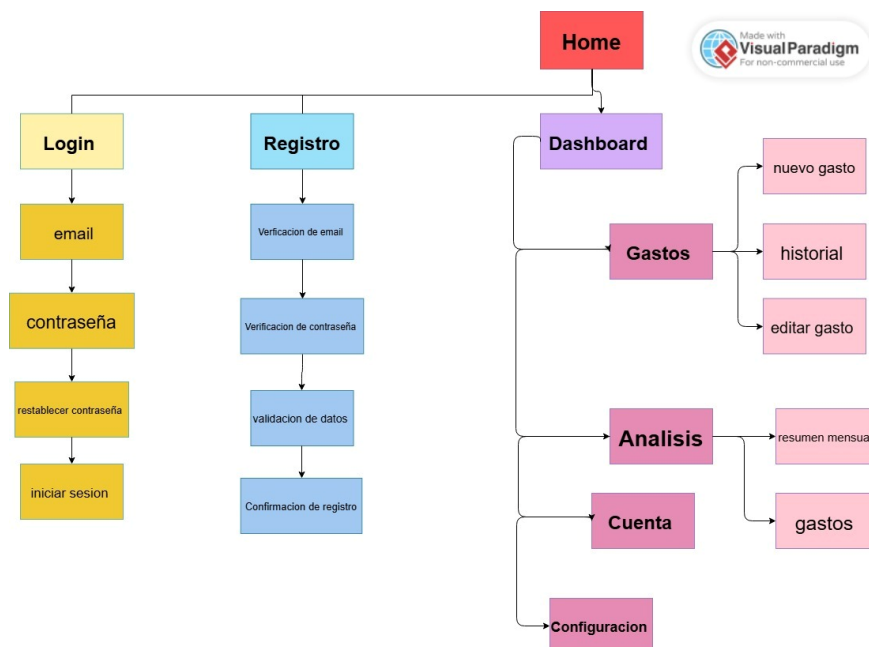
Además, una aplicación web permite:

- Mantenimiento centralizado del sistema.
- Escalabilidad técnica conforme crecen los usuarios.
- Implementación de flujos de autenticación y control de acceso.
- Actualizaciones inmediatas sin intervención del usuario.

Estas características hacen que la aplicación web sea la solución más adecuada para el problema identificado.

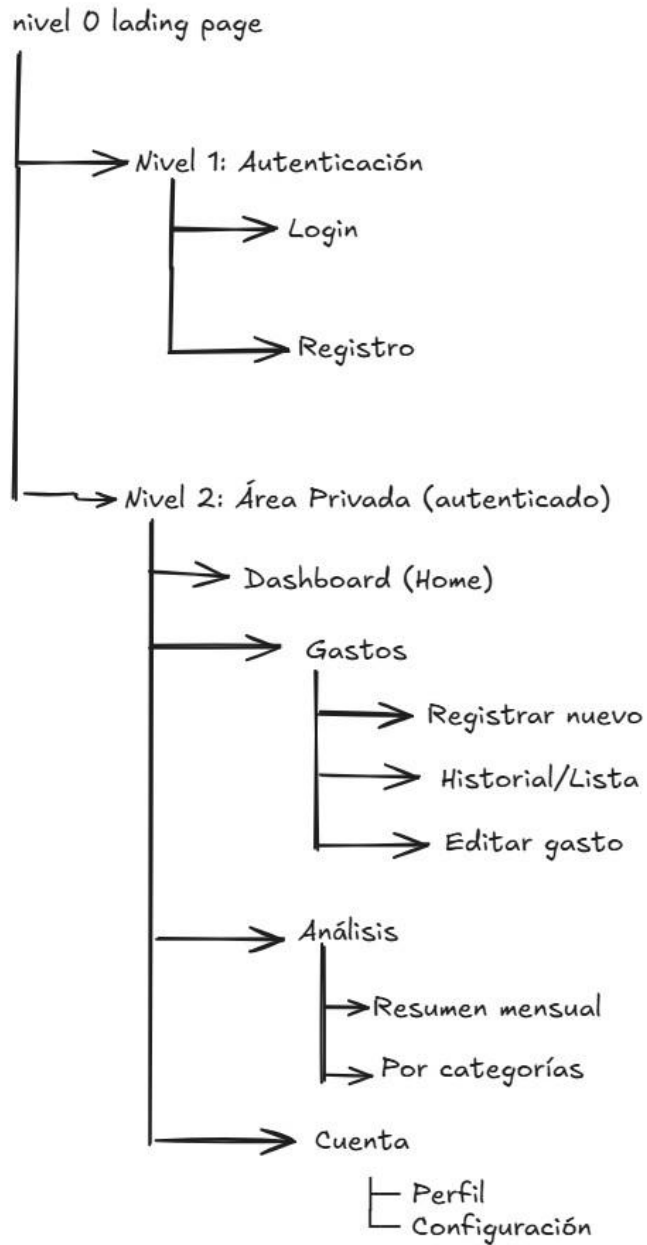
Arquitectura y navegación

- Diseñar:
 - Sitemap completo



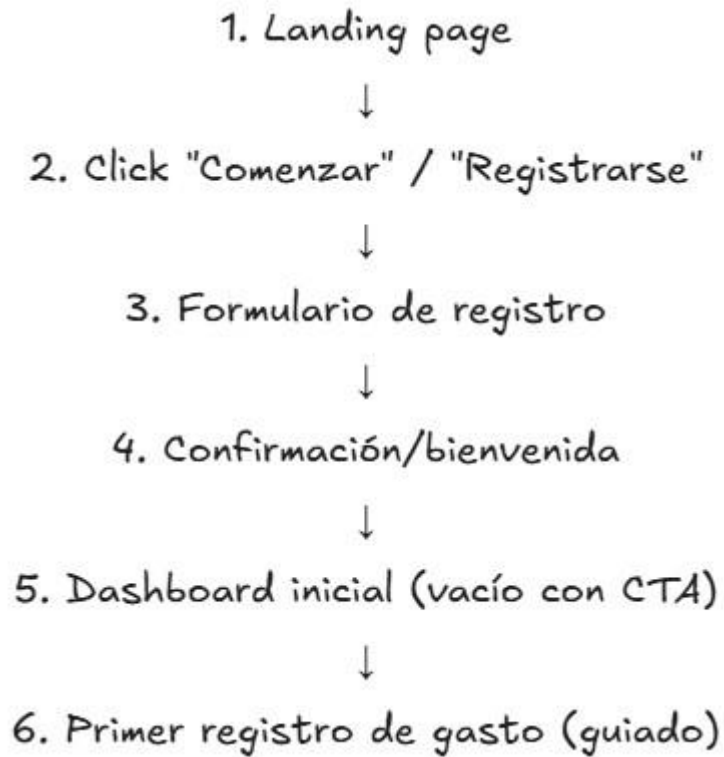
Patrones de Navegación - Aplicación de Control de Gastos

Arquitectura de Información Estructura Jerárquica



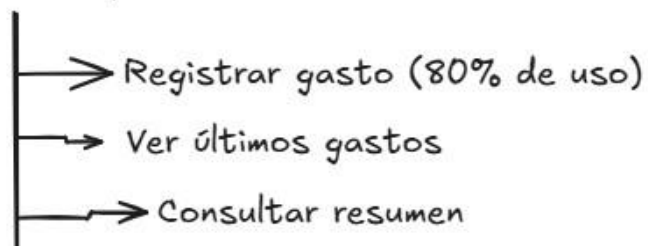
Flujos de usuarios detallados

Primer uso



Uso recurrente

Login → Dashboard → Acción rápida



Registro de gasto (flujo principal)

Dashboard/Cualquier página



Click "+" o "Nuevo Gasto"



Modal/Página de registro

- Campo: Monto
- Campo: Categoría (selector)
- Campo: Fecha (default: hoy)
- Campo: Descripción (opcional)



Click "Guardar"



Confirmación visual



Retorno a origen o Dashboard

Consulta de historial

Navegación → Historial



Lista de gastos

- └ Filtros disponibles:
 - └ Por fecha (rango)
 - └ Por categoría
 - └ Por monto (orden)



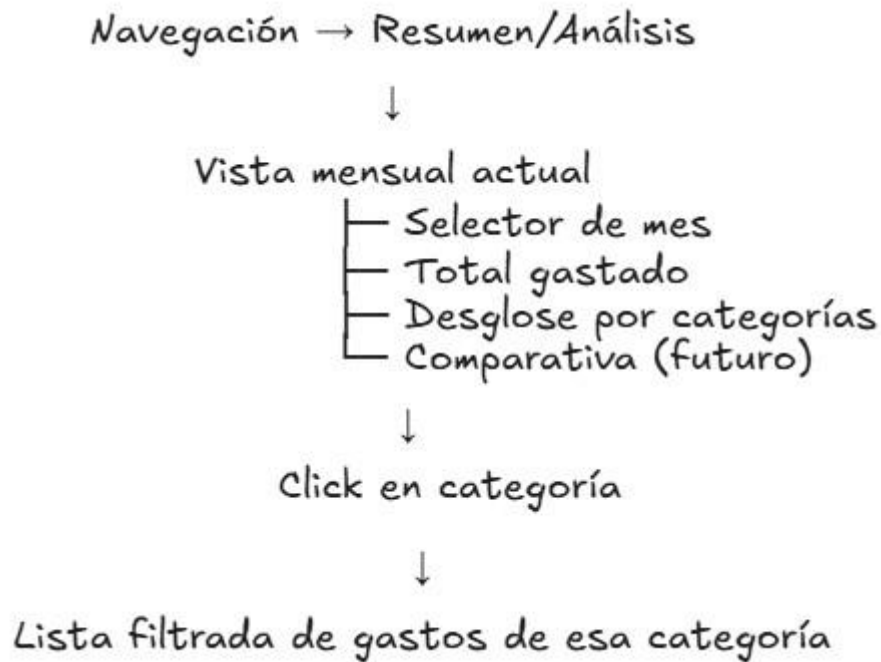
Click en gasto individual



Detalle del gasto

- └ Opción: Editar
- └ Opción: Eliminar

Resumen mensual



3. Componentes de Navegación

Navegación principal (header)

Opción 1: Barra horizontal

[Logo] Dashboard | Gastos | Análisis | [Botón: + Nuevo Gasto] | [Avatar ▼]

Opción 2: Menú lateral colapsable

≡ Menú



Botón de Acción Flotante (FAB)

[+] Siempre visible en esquina inferior derecha

→ Abre modal de registro rápido

Breadcrumbs

Inicio > Gastos > Historial > Editar gasto

Inicio > Análisis > Abril 2025

5. Patrones de Interacción

Registro Rápido (Modal)

- Activación: Botón flotante, atajo teclado (Alt+N), menú principal
- Comportamiento: Overlay sobre página actual
- Cierre: Click fuera, ESC, botón cancelar
- Confirmación: Mensaje toast "Gasto registrado ✓" Urls y rutas

/	→ Landing (público)
/login	→ Inicio de sesión
/registro	→ Crear cuenta
/dashboard	→ Panel principal (privado)
/gastos	→ Historial completo
/gastos/nuevo	→ Formulario registro
/gastos/:id/editar	→ Editar gasto específico
/resumen	→ Análisis mensual actual
/resumen/:año/:mes	→ Análisis de mes específico
/perfil	→ Datos de usuario
/configuracion	→ Ajustes de cuenta

8. Accesos Rápidos y Atajos

Teclado (Desktop)

- Alt + N → Nuevo gasto
- Alt + H → Historial
- Alt + R → Resumen
- ESC → Cerrar modal/cancelar

Accesos Contextuales

Dashboard → "Gastos de esta semana" (link) → Historial filtrado

Resumen → Click en categoría → Historial de esa categoría

Feedback de Navegación

Indicadores Visuales

- Página activa: Resaltado en menú
- Carga: Spinner/skeleton screen
- Sin resultados: "No hay gastos en este período"

- Error: "No se pudo cargar. [Reintentar]"

Mensajes Toast

✓ "Gasto registrado exitosamente"

✓ "Cambios guardados"

⚠ "Completa todos los campos"

✗ "Error al guardar. Intenta de nuevo"

Consideraciones de Accesibilidad

- Skip links: "Saltar al contenido principal"
- Navegación por teclado: Tab index lógico
- ARIA labels: Botones e íconos descriptivos
- Focus visible: Indicadores claros
- Breadcrumbs: <nav aria-label="breadcrumb">

Definición de rutas públicas y rutas privadas

En la aplicación web para el control de gastos personales, las rutas del sistema se dividen en rutas públicas y rutas privadas con el objetivo de controlar el acceso a la información y garantizar la seguridad de los datos financieros de los usuarios.

Rutas públicas

Las rutas públicas son aquellas a las que cualquier usuario puede acceder sin necesidad de autenticarse en el sistema.

Su función principal es permitir el acceso inicial a la aplicación, ya sea para que el usuario cree una cuenta o inicie sesión.

En este proyecto, las rutas públicas incluyen:

- Registro de usuario.
- Inicio de sesión.

Estas rutas no muestran información sensible y son esenciales para que el usuario pueda ingresar de forma segura al sistema.

Rutas privadas

Las rutas privadas son aquellas que requieren que el usuario esté autenticado y autorizado.

Estas rutas protegen la información personal y financiera, asegurando que cada usuario solo pueda acceder y modificar sus propios datos.

En la aplicación web de control de gastos personales, las rutas privadas incluyen:

- Consulta y edición del perfil del usuario.
- Registro de gastos personales.
- Consulta del historial de gastos.
- Edición de gastos.
- Eliminación de gastos.

El acceso a estas rutas está restringido para garantizar la confidencialidad, integridad y control de acceso de la información financiera dentro del sistema.

Sistema en flujos

El sistema fue diseñado considerando flujos de usuario, entendidos como la secuencia de acciones que una persona realiza para completar una tarea, y no únicamente como pantallas aisladas.

Flujo principal del sistema:

El usuario accede a la aplicación → inicia sesión → entra al dashboard → registra un gasto → guarda la información → visualiza el resumen de gastos actualizado.

Flujo secundario:

El usuario inicia sesión → accede al listado de gastos → selecciona un gasto → edita o elimina el registro → el sistema actualiza la información mostrada.

Este enfoque permite una navegación clara, coherente y orientada a objetivos, facilitando la usabilidad y el mantenimiento del sistema.

Validar navegación solo con teclado

Navegación por teclado

La navegación por teclado permite interactuar con la aplicación sin utilizar el mouse.

Elementos que deben ser accesibles:

- Menús de navegación
- Botones
- Formularios
- Enlaces
- Modales

Teclas comunes:

- TAB: moverse entre elementos
- ENTER: activar botones o enlaces
- ESPACIO: seleccionar opciones
- ESC: cerrar ventanas o modales

Accesibilidad sin mouse

Una aplicación accesible debe poder utilizarse completamente sin mouse.

Se valida que:

- Todos los elementos sean alcanzables con el teclado
- Los botones tengan foco visible
- No existan funciones exclusivas del mouse
- Los formularios puedan completarse con teclado

Esto beneficia a personas con discapacidad física, visual o usuarios con dispositivos alternativos.

Validación de navegación accesible

Para validar la accesibilidad se realizan pruebas manuales:

Proceso de validación:

1. Navegar la aplicación usando solo el teclado
2. Verificar el orden correcto de tabulación
3. Confirmar que todos los botones funcionen
4. Revisar que el foco sea visible
5. Asegurar que no existan bloqueos de navegación

Observaciones y Correcciones

Observaciones detectadas

- Elementos sin foco visible
- Orden de tabulación incorrecto
- Botones que no responden con ENTER
- Enlaces no accesibles por teclado

Correcciones propuestas

- Ajustar el orden del DOM
- Implementar estilos de foco visibles