Un letrero azul con letras blancas

El contenido generado por IA puede ser incorrecto.

# Práctica 1.

# Evaluación Continua.

# Lenguaje de Marcas y sistemas de gestión de la información

Nombre completo: David

Apellidos: García Rodríguez

Curso (DAM/DAW/ASIR): DAW

Promoción: Febrero 2025

**Capturas del diseño en Figma**

* + Bocetos para versión escritorio y versión móvil.

Interfaz de usuario gráfica, Aplicación, Sitio web, Teams

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación, Teams

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación, Sitio web

El contenido generado por IA puede ser incorrecto.

**Enlace al repositorio de GitHub**

* + Debe ser público y mostrar un historial de trabajo progresivo.

<https://github.com/Deika79/Portfolio>

**Enlace al sitio web publicado en un hosting gratuito**

* + Puedes usar servicios como GitHub Pages, Netlify, Vercel, entre otros.

<https://deika79.github.io/Portfolio/>

**Complicaciones encontradas y soluciones aplicadas**

* + Explica al menos **dos problemas reales** surgidos durante el desarrollo y cómo los resolviste.

### 1. Intento de migración del proyecto a **Vite**

Durante el desarrollo del portfolio quise dar un paso más y probar una herramienta moderna de construcción como **Vite**, ya que había leído que mejoraba el rendimiento y la organización del proyecto frente a un HTML/CSS/JS tradicional.  
Sin embargo, al iniciar la migración me encontré con varias dificultades: la estructura del proyecto requería configuraciones adicionales que no conocía, especialmente en la carga de fuentes locales y rutas de imágenes. Además, Vite trabaja con un entorno más cercano a frameworks como React o Vue, y al no tener todavía suficiente nivel en estas tecnologías, no conseguí que la página funcionara correctamente en local.

**Cómo lo resolví:** finalmente decidí mantener el proyecto en su forma estática original (HTML, CSS y JS) para asegurarme de poder entregarlo funcional. Aunque no logré completar la migración, el intento me sirvió para familiarizarme con el proceso de instalación de Vite, su configuración inicial y sus ventajas, lo cual considero positivo como aprendizaje de cara a proyectos futuros.

### 2. Creación de un formulario de contacto real en **GitHub Pages**

Otra funcionalidad que quise implementar fue un **formulario de contacto funcional** (que enviara mensajes a un correo electrónico). El problema principal fue que mi portfolio está desplegado en **GitHub Pages**, que solo permite contenido estático y no admite la ejecución de un backend en PHP, Node.js u otra tecnología de servidor.  
Intenté investigar alternativas como usar servicios externos (ej. Formspree o Netlify Forms), pero no me fue posible integrarlos a tiempo de manera correcta.

**Cómo lo resolví:** opté por dejar el formulario únicamente como simulación de diseño (captura de datos en pantalla, sin envío real). Documenté la limitación de GitHub Pages y exploré las opciones disponibles, lo cual me permitió comprender la diferencia entre proyectos estáticos y dinámicos, así como las limitaciones de cada entorno de despliegue. Aunque no conseguí habilitar el envío real, considero positivo haber detectado la restricción técnica y planteado posibles soluciones.

**Tipografías utilizadas**

* + Especifica el **nombre de cada fuente** utilizada y **su origen** (recordatorio: deben estar **cargadas localmente** y libres de derechos).

Las dos fuentes se han descargado desde <https://www.dafont.com/es/> y son:

* Aldotheapache
* Roze Belinda (regular)

**Plan de pruebas**

* + Detalla un plan básico de pruebas que hayas realizado para validar el correcto funcionamiento de la web.
  + Incluye pruebas como:
    - Visualización en distintos tamaños de pantalla (responsive).
    - Funcionamiento de enlaces y formularios.
    - Visualización en varios navegadores (Chrome, Firefox, etc.).
    - Funcionamiento del menú y del popup de cookies.
    - Carga correcta de fuentes e imágenes.

**Plan de Pruebas**

Con el objetivo de validar el correcto funcionamiento de la web desarrollada, se han realizado una serie de pruebas básicas sobre diferentes aspectos clave: diseño, usabilidad, compatibilidad y carga de recursos. A continuación, se detalla el plan de pruebas llevado a cabo:

**1. Visualización en distintos tamaños de pantalla (Responsive)**

* **Prueba:** Redimensionar la ventana del navegador y utilizar las herramientas de desarrollo (modo responsive).
* **Resultado esperado:** Los elementos se adaptan correctamente, sin desbordamientos ni cortes.
* **Resultado obtenido:** La web mantiene su estructura tanto en escritorio como en tablet y móvil, con el menú y el contenido adaptados al tamaño de pantalla.

**2. Funcionamiento de enlaces y formularios**

* **Prueba:** Verificar que todos los enlaces navegan a la sección o página correcta (incluyendo el enlace a la Política de Privacidad).
* **Resultado esperado:** No aparecen errores 404 y los enlaces funcionan sin problemas.
* **Resultado obtenido:** Todos los enlaces llevan a su destino. Los formularios (si los hay) muestran la interacción prevista.

**3. Visualización en varios navegadores**

* **Prueba:** Abrir la web en **Google Chrome, Mozilla Firefox y Microsoft Edge**.
* **Resultado esperado:** La web se visualiza correctamente en todos los navegadores.
* **Resultado obtenido:** La apariencia y el funcionamiento son correctos en todos los navegadores probados, sin diferencias significativas.

**4. Funcionamiento del menú y del popup de cookies**

* **Prueba:** Comprobar que el menú es accesible en todas las resoluciones (incluido el menú móvil si existe). Verificar que el popup de cookies aparece la primera vez y permite aceptar, rechazar o personalizar las cookies.
* **Resultado esperado:** El menú responde a los clics/taps y el popup de cookies guarda la preferencia del usuario.
* **Resultado obtenido:** El menú funciona correctamente en escritorio y móvil. El popup de cookies aparece y desaparece según la interacción y almacena las preferencias en localStorage.

**5. Carga correcta de fuentes e imágenes**

* **Prueba:** Revisar en las herramientas de desarrollo que las fuentes se cargan de la carpeta local /fonts y que las imágenes se muestran correctamente.
* **Resultado esperado:** No deben aparecer errores de carga (404) y las fuentes locales deben estar aplicadas en los estilos.
* **Resultado obtenido:** Las fuentes locales se aplican según lo definido en el CSS (FuenteUno y FuenteDos). Todas las imágenes se cargan sin errores.

Con este plan de pruebas se puede concluir que la web funciona de forma correcta en los aspectos fundamentales de visualización, compatibilidad, accesibilidad y carga de recursos.

**Conclusiones**