



**INSTITUTO POLITÉCNICO NACIONAL**  
**UNIDAD PROFESIONAL INTERDISCIPLINARIA DE INGENIERÍA Y CIENCIAS**  
**SOCIALES Y ADMINISTRATIVAS.**

**Practica 3: Sitio CSS**

**Alumno:**

Gordillo García Mario Yael

**Secuencia:**

5NM58

**Profesor:** Montes Casiano Hermes Francisco.

**Fecha de entrega:**

14 de marzo de 2025



## Índice

<b>Introducción.....</b>	<b>3</b>
<b>Propósito.....</b>	<b>4</b>
<b>Desarrollo.....</b>	<b>5</b>
1. Estructura del HTML.....	5
2. Diseño del Header y la Navegación.....	5
3. Contenido y Organización en el Main.....	6
4. Estilización con CSS.....	7
5. Diseño Responsivo.....	8
6. Propuestas de Maquetación: Flexbox y CSS Grid Layout.....	8
7. Diseño del Footer.....	9
8. Consideraciones Finales.....	9
<b>Resultado.....</b>	<b>11</b>
<b>Conclusión.....</b>	<b>14</b>

## Introducción

En esta práctica, se ha desarrollado un sitio web responsivo que integra diversos elementos fundamentales del diseño y desarrollo web. La estructura del sitio incluye:

- Header: Un encabezado fijo que contiene la navegación principal.
- Footer: Un pie de página con información de autor y enlaces.
- Menú horizontal y vertical (multinivel):
  - El menú horizontal cuenta con cinco secciones principales.
  - Cada sección tiene un menú vertical con cinco enlaces a diferentes páginas HTML.
- Diseño responsivo: Se implementaron media queries para adaptar el contenido a dispositivos móviles.
- Uso de Grid y Flexbox:
  - Se crearon dos propuestas de diseño con Flexbox y dos con CSS Grid Layout.
- Variedad de contenido: Las diferentes páginas HTML incluyen tablas, imágenes, listas, formularios y otros elementos estructurales.

Este proyecto no solo aplica conceptos básicos de HTML y CSS, sino que también refuerza buenas prácticas en la organización del contenido y el diseño adaptable.

### **Propósito**

El propósito de esta práctica es desarrollar un sitio web estructurado y responsivo, aplicando principios fundamentales de diseño y desarrollo web. A través de la implementación de HTML y CSS, se busca:

- **Fortalecer la comprensión** de estructuras semánticas en HTML.
- **Aplicar técnicas modernas de CSS**, como Flexbox y Grid Layout, para organizar el contenido de manera eficiente.
- **Diseñar una interfaz adaptable**, asegurando una experiencia de usuario óptima en distintos dispositivos.
- **Explorar la funcionalidad de menús de navegación**, tanto horizontales como verticales, incluyendo opciones multinivel.
- **Integrar diversos elementos web**, como tablas, listas, imágenes y formularios, para mejorar la presentación y usabilidad.

Este proyecto fomenta el desarrollo de habilidades en diseño web, promoviendo buenas prácticas en accesibilidad, estructura y estilo.

## Desarrollo

A continuación se detalla el proceso y las técnicas empleadas en la creación del sitio web, basándonos en los códigos HTML y CSS proporcionados:

### 1. Estructura del HTML

- **Definición semántica:**

Se utiliza HTML5 para estructurar el documento de forma semántica, lo cual mejora la accesibilidad y la comprensión del contenido tanto para los usuarios como para los motores de búsqueda.

- El documento inicia con la declaración `<!DOCTYPE html>` y se define el idioma con `<html lang="es">`.

- En el `<head>` se incluyen metadatos esenciales: configuración de caracteres (UTF-8), compatibilidad con navegadores, el viewport para dispositivos móviles, y metadatos descriptivos (descripción y palabras clave).

- **Estructura principal:**

El cuerpo del documento se organiza en secciones clave:

- **Header:** Contiene el encabezado fijo y la barra de navegación.
- **Main:** Alberga el contenido principal distribuido en secciones como “Inicio”, “Blog”, “Artista” y “Videojuegos”. Cada sección utiliza elementos semánticos (`<section>`, `<h1>`, `<h2>`, `<p>`, etc.) para organizar la información.
- **Footer:** Incluye datos del autor, enlaces de contacto y derechos de autor.

### 2. Diseño del Header y la Navegación

- **Encabezado fijo y estilizado:**

El `<header>` se posiciona de forma fija en la parte superior de la página con `position: fixed` y un

**z-index** alto para que permanezca visible durante el desplazamiento.

- Se aplica un fondo semitransparente junto con un efecto de desenfoque mediante **backdrop-filter: blur(5px)**, lo que proporciona un aspecto moderno y elegante.
- La altura definida (70px) y el espaciado aseguran que el contenido del header se mantenga legible y organizado.

- **Menú horizontal y vertical (multinivel):**

- **Menú horizontal:**

Utilizado para la navegación principal, se implementa con una lista `<ul>` de clase **.menu-horizontal** que contiene cinco secciones. Cada sección está representada por un `<li>` que incluye un enlace `<a>` y, en la mayoría de los casos, un submenú.

- **Menú vertical:**

Cada elemento del menú horizontal (excepto “Contacto”, en este caso) posee un submenú representado por otra lista `<ul>` con la clase **.menu-vertical**.

- Cada menú vertical contiene cinco entradas, cada una enlazando a diferentes archivos HTML (por ejemplo, **inicio1.html**, **blog1.html**, etc.).

- El despliegue de estos submenús se maneja mediante CSS: al pasar el cursor sobre el elemento padre, se activa la propiedad **display: block** para mostrar el menú oculto.

### 3. Contenido y Organización en el Main

- **Secciones del contenido:**

El `<main>` está dividido en varias secciones que abarcan diferentes tipos de contenido:

- **Inicio:**

Presenta una breve biografía del autor, imágenes y párrafos descriptivos que dan la bienvenida

al usuario.

- **Blog:**

Utiliza listas ordenadas (`<ol>`) para enumerar conceptos y elementos aprendidos, lo que ayuda a estructurar y clarificar la información.

- **Horario de clase:**

Se muestra una tabla (`<table>`) con filas y columnas para detallar los días y horarios de clase.

- **Artista y Videojuegos:**

Cada sección utiliza imágenes, párrafos y videos embebidos mediante `<iframe>` para enriquecer la presentación del contenido y ofrecer una experiencia multimedia.

- **Diversidad en el contenido:**

Cada página enlazada a través de los menús incluye diferentes elementos (tablas, imágenes, formularios, listas) que permiten explorar múltiples técnicas de despliegue y organización de información.

#### 4. Estilización con CSS

- **Reset y configuración global:**

Se inicia con un reset básico para eliminar márgenes y paddings predeterminados, y se establece una tipografía base uniforme para todo el documento.

- **Animación y fondo:**

- Se implementa una animación en el `body` usando `@keyframes gradient` para generar un fondo con gradiente en movimiento.

- La propiedad `background-size` y la animación contribuyen a un efecto visual dinámico que capta la atención del usuario.

- **Estilos del Header y Navegación:**

- El header y el nav se diseñan para ser fijos y responsivos.
- Los enlaces (`<a>`) incluyen transiciones suaves que mejoran la interacción del usuario (por ejemplo, cambios de color y fondo al pasar el cursor).

- **Menús desplegables:**

- Los menús horizontales y verticales están diseñados con propiedades como `position: relative` y `position: absolute` para asegurar que los submenús se desplieguen correctamente sobre el contenido principal.
- Se aplican sombras (`box-shadow`) y bordes redondeados (`border-radius`) para mejorar la estética visual.

## 5. Diseño Responsivo

- **Media Queries:**

Se utilizan diferentes *media queries* para adaptar el diseño a dispositivos de distintos tamaños:

- **Para pantallas de hasta 768px:**

Se ajusta el layout del nav para que los elementos se acomoden en varias filas, se reducen los tamaños de fuente y se adapta la altura de los iframes y tablas.

- **Para pantallas de hasta 480px:**

Se optimiza aún más el diseño, cambiando la dirección del menú a columna y aumentando el padding superior del `body` para compensar el header fijo.

- **Beneficios de la responsividad:**

Estos ajustes garantizan que el sitio ofrezca una experiencia de usuario óptima en dispositivos móviles y tabletas, manteniendo la legibilidad y funcionalidad en todas las resoluciones.

## 6. Propuestas de Maquetación: Flexbox y CSS Grid Layout



- **Flexbox:**

Se incluyeron dos propuestas utilizando Flexbox para organizar el contenido:

- La clase `.flex-grid` se utiliza para distribuir elementos en una línea flexible, permitiendo el ajuste dinámico de los elementos según el espacio disponible.
- Esta técnica es ideal para casos en los que se requiere un diseño fluido y adaptable, manteniendo una separación uniforme entre los elementos.

- **CSS Grid Layout:**

También se ofrecen dos propuestas mediante CSS Grid:

- La clase `.grid-layout` define un sistema de cuadrícula que ajusta automáticamente el número de columnas utilizando la función `repeat(auto-fit, minmax(250px, 1fr))`.
- Este enfoque permite crear layouts complejos y estructurados de manera sencilla, optimizando la presentación de elementos de forma uniforme y escalable.

## 7. Diseño del Footer

- **Elementos y estilo:**

El pie de página está diseñado para complementar la estructura del sitio:

- Utiliza un fondo semitransparente con `backdrop-filter: blur(5px)` similar al header, creando una coherencia visual.
- Contiene información de autoría, enlaces a perfiles (por ejemplo, GitHub) y mensajes de derechos de autor.
- La alineación centrada y el espaciado adecuado aseguran que el footer sea visible y legible en todas las resoluciones.

## 8. Consideraciones Finales

- **Integración y buenas prácticas:**

La práctica combina aspectos estéticos y funcionales del desarrollo web, utilizando una

estructura semántica y técnicas modernas de CSS para asegurar una experiencia de usuario rica y adaptable.

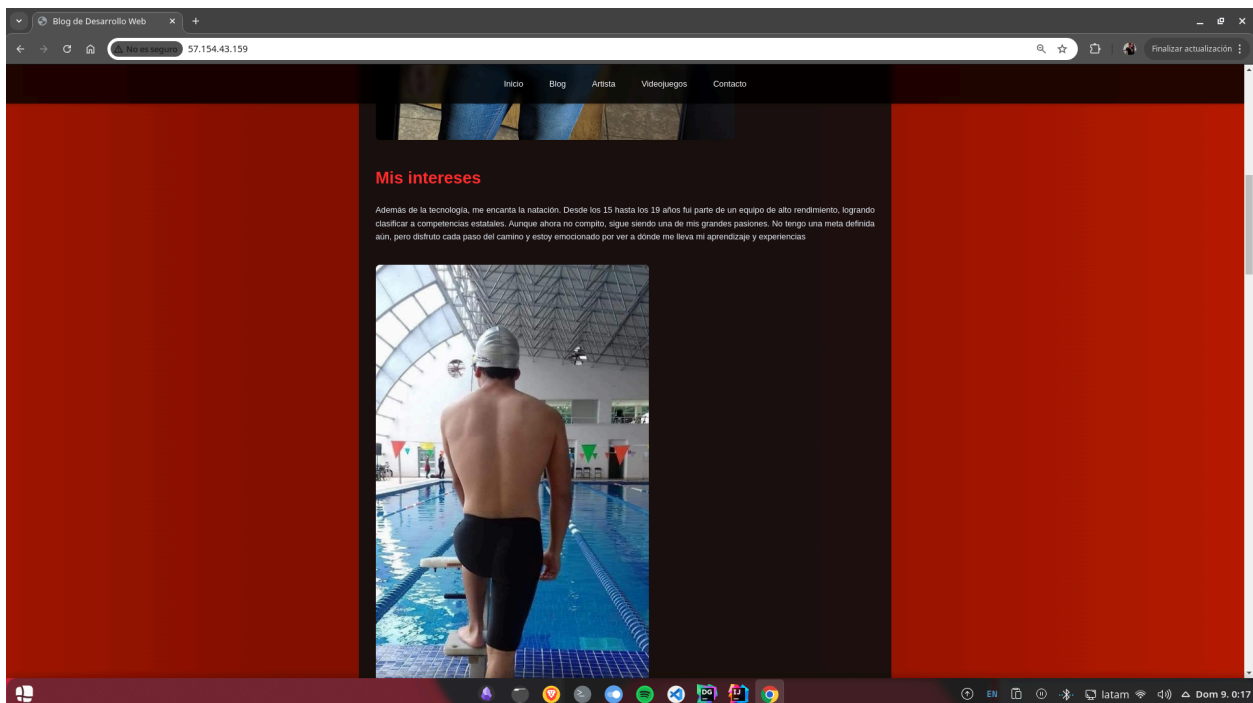
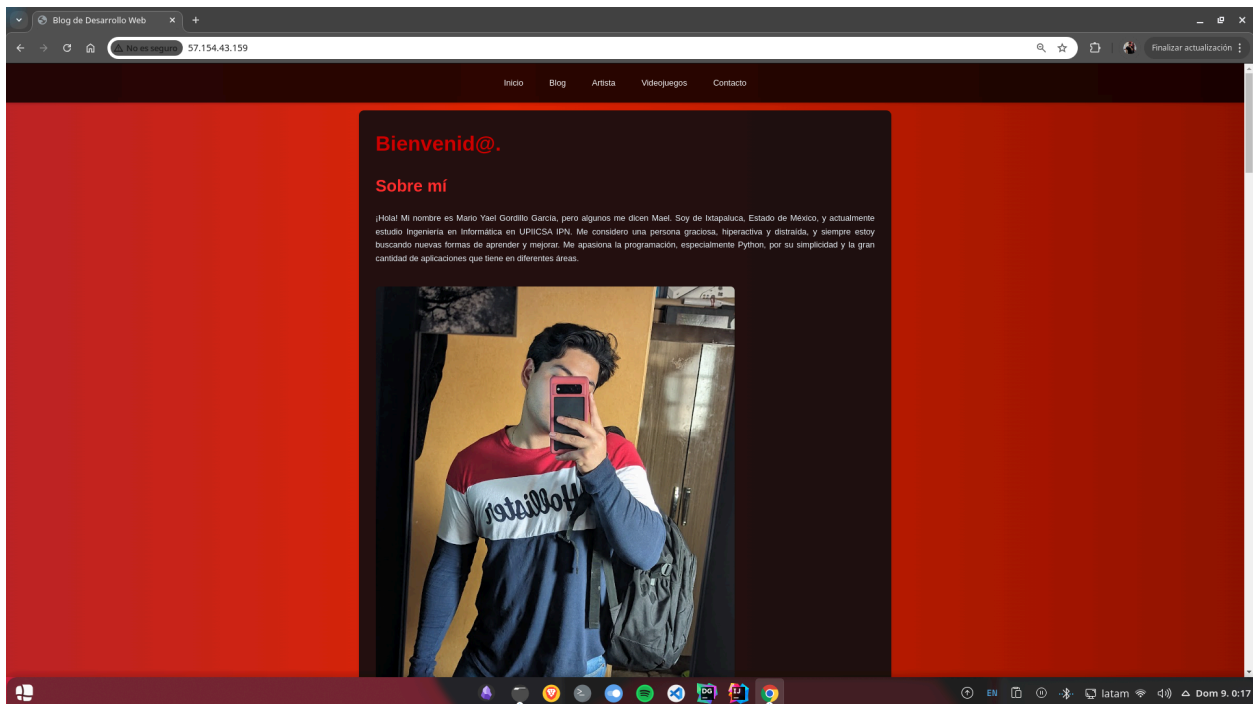
- **Diversidad de elementos:**

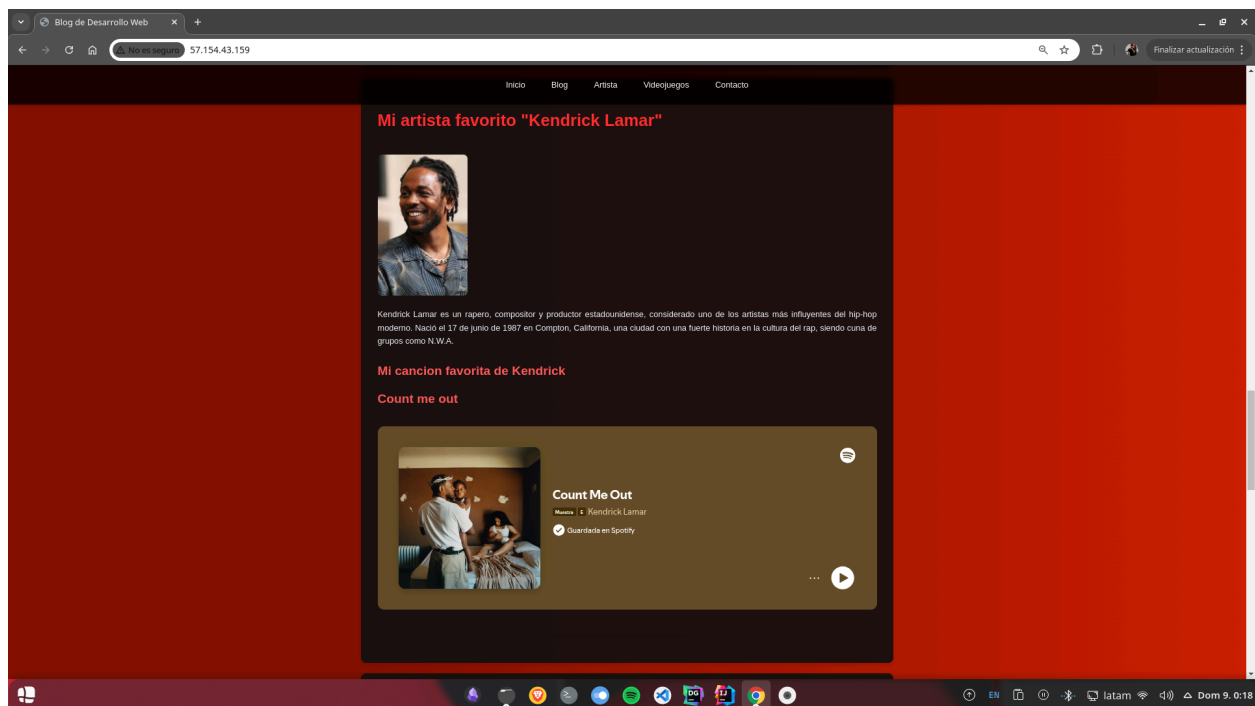
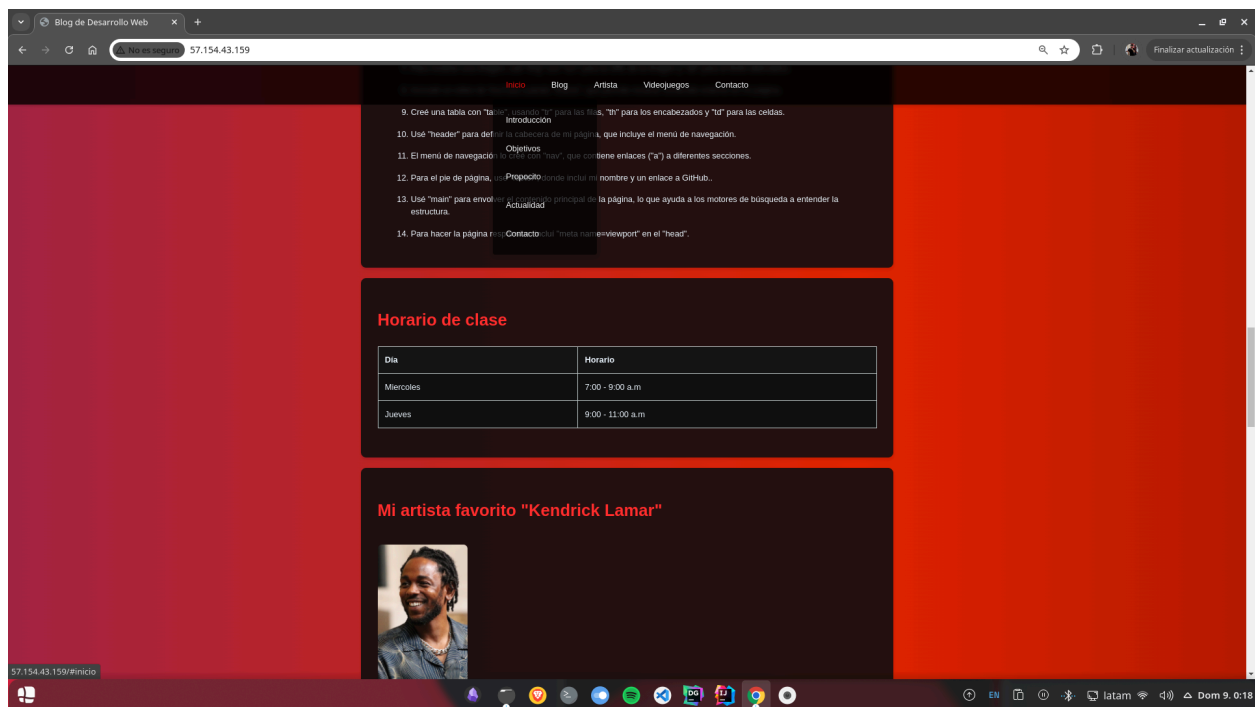
La inclusión de menús multinivel, animaciones de fondo, elementos multimedia (imágenes, videos), y estructuras de contenido variadas (tablas, listas, formularios) permite explorar una amplia gama de técnicas de desarrollo, reforzando los conocimientos sobre HTML y CSS.

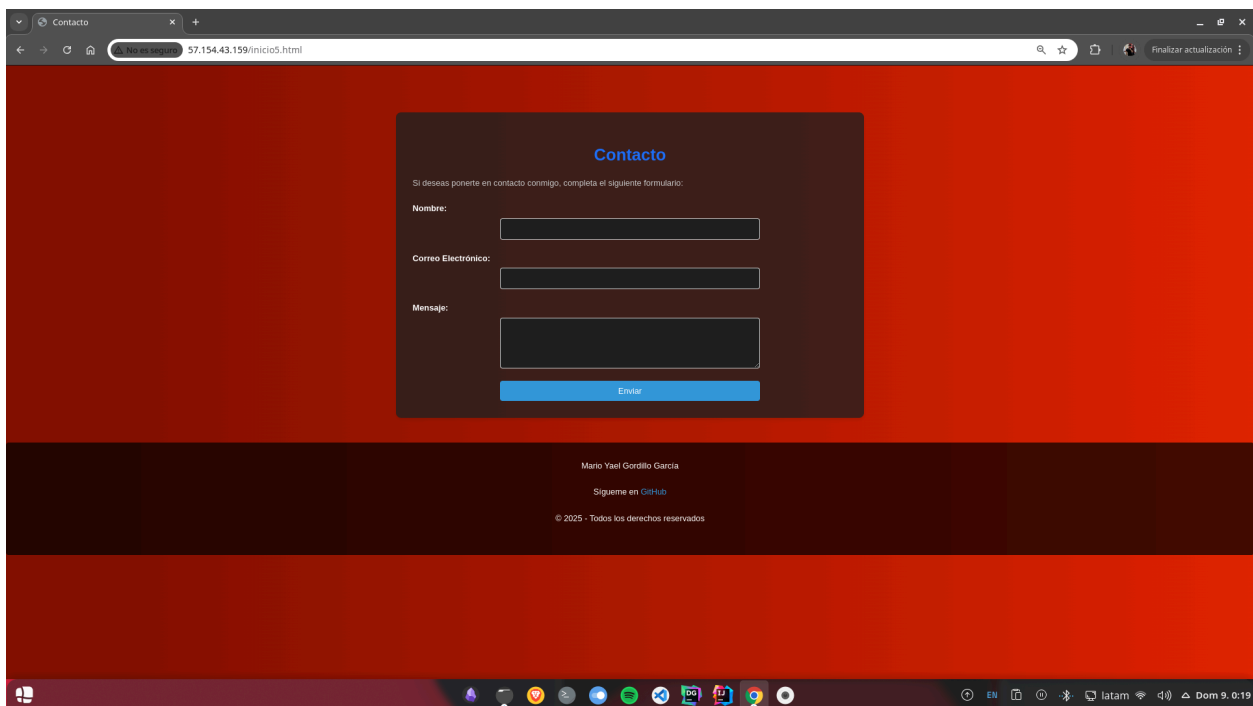
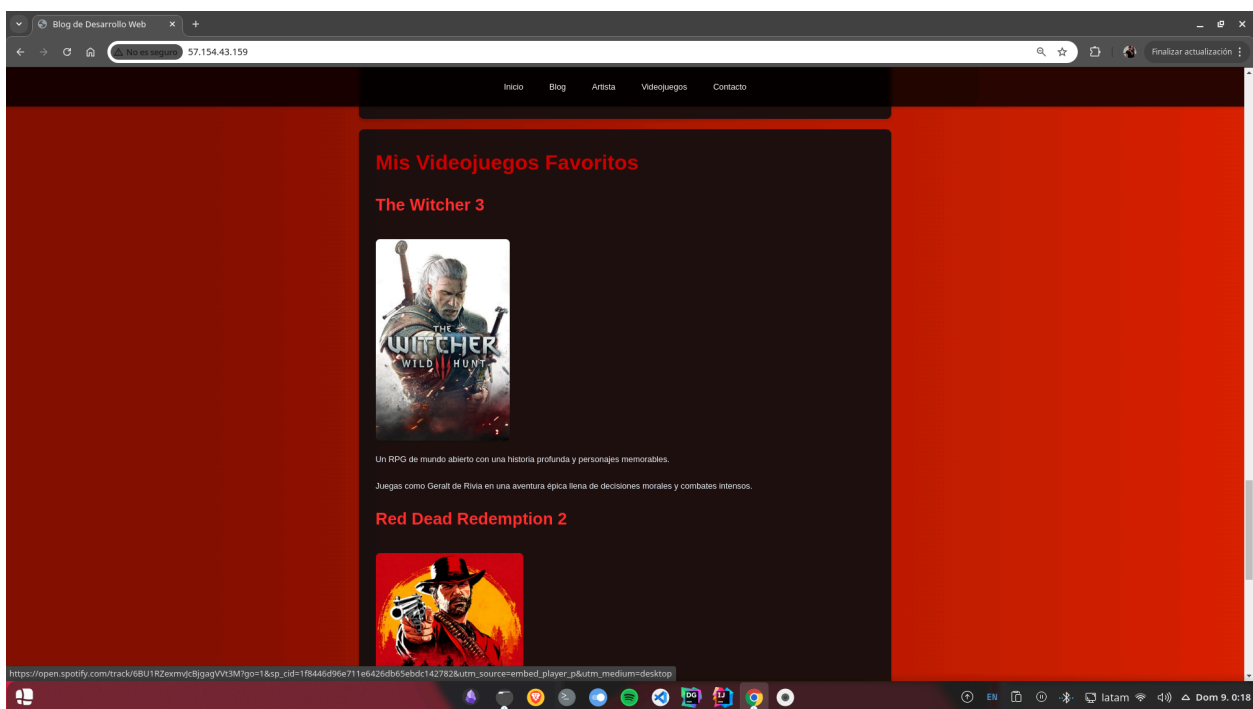
- **Enfoque en la experiencia del usuario:**

Todo el desarrollo está orientado a crear un sitio intuitivo y atractivo, donde la navegación y la presentación del contenido se adaptan de manera óptima a diferentes dispositivos, cumpliendo así con los estándares modernos de diseño web.

## Resultado







## Conclusión

En resumen, este proyecto web integra de manera efectiva técnicas modernas y buenas prácticas en el desarrollo front-end. Se logró:

- **Estructura semántica y organizada:** La implementación de HTML5 con sus etiquetas semánticas asegura que el contenido esté bien estructurado y sea accesible, lo que facilita tanto la navegación del usuario como el posicionamiento en buscadores
- **Diseño interactivo y responsivo:** La combinación de menús horizontales y verticales multinivel, junto con la aplicación de media queries, garantiza una experiencia de usuario óptima en diversos dispositivos, desde pantallas de escritorio hasta móviles.
- **Maquetación flexible:** La utilización de Flexbox y CSS Grid Layout permite experimentar con diferentes propuestas de distribución, mejorando la presentación visual y la adaptabilidad del contenido.
- **Diversidad de elementos de contenido:** La integración de tablas, imágenes, listas y formularios no solo enriquece la información presentada, sino que también demuestra el manejo de múltiples recursos para una interfaz web completa.