

**UNIVERSIDAD PRIVADA DEL VALLE
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN SISTEMAS INFORMÁTICOS**



**PAGINA WEB – YOUNG SOUL
PROYECTO FINAL PROGRAMACION WEB I**

Alumna:

Melani Quintela Aguilar

Docente: Henry Miranda Ordoñez

Paralelo: A

**La Paz- Bolivia
2-2025**

INDICE

1. Descripción general del proyecto.....	3
2. Objetivos del Sistema.....	3
3. Tecnologías utilizadas	4
3.1 HTML5 (HyperText Markup Language)	4
3.2 CSS3 (Cascading Style Sheets)	4
3.3 JavaScript (Vanilla JavaScript).....	5
3.4 Git / GitHub	6
4. Estructura de carpetas	6
5. Funcionamiento de cada pagina	7
5.1 index.html (Inicio)	7
5.2 sobre.html (Sobre Young Soul)	7
5.3 elenco.html (Elenco)	7
5.4 instructor.html.....	7
5.5 clases.html.....	8
5.6 inscribirse.html	8
5.7 contacto.html	8
6. Diseño y Estilos (CSS)	8
7. Lógica General del Proyecto	9
7.2 Comportamiento del sitio.....	9
7.2.1 Botones y enlaces	9
7.5 Lógica de Responsividad.....	10
8.Codigos	10
8.1. Reset y Normalización del Estilo	10
8.2. Configuración del Header y Video de Fondo.....	11
8.3. Navegación del Header.....	12
8.4. Estilo de los Links de Navegación	13
8.5. Estilos para Secciones.....	14
8.6. Botón con Efecto Hover.....	14
8.7. Contenedor de Dos Secciones con Imagen de Fondo	15
8.7 Footer (Pie de Página)	16
8.8. Videos con Efectos en Hover.....	16

1. Descripción general del proyecto

El sitio web de **Young Soul** es una plataforma informativa desarrollada en **HTML, CSS y JavaScript**, destinada a mostrar:

- Información institucional
- Clases
- Instructores
- Elenco
- Formulario de inscripción con validaciones
- No utiliza backend ni base de datos.

Toda la lógica se ejecuta del lado del cliente mediante JavaScript.

2. Objetivos del Sistema

El sistema fue diseñado para:

- Presentar información institucional de la academia Young Soul.

- Mostrar instructores, clases, elenco y datos de contacto.
- Permitir que el usuario complete un formulario de inscripción con validaciones en el front-end.
- Facilitar la navegación en distintas secciones mediante enlaces.

3. Tecnologías utilizadas

3.1 HTML5 (HyperText Markup Language)

Función:

- Estructura toda la información del sitio web.
- Define la jerarquía y organización de cada página: encabezados, párrafos, botones, formularios, imágenes, menús y secciones.

Uso en el proyecto:

- Construcción de las páginas: Inicio, Instructores, Clases, Inscripción, Contáctanos, Sobre Young Soul y Elenco.
- Formularios con campos para nombre, teléfono, edad y clase
- Inserción del logo y enlaces de navegación.

3.2 CSS3 (Cascading Style Sheets)

Función:

- Define el diseño visual, colores, tamaños, posiciones, tipografías, espaciados y efectos.

- Permite crear una interfaz estética y responsiva.

Uso en el proyecto:

- Diseño completo del sitio:
- Menú superior y barra de navegación.
- Botones “Leer más” e “Información del elenco”.
- Secciones responsivas usando flex.
- Estilos de las secciones de instructores y clases.
- Bordes redondeados, sombras, hovers.
- Ajuste del tamaño de videos/imágenes.

3.3 JavaScript (Vanilla JavaScript)**Función:**

Agregar interactividad al sitio web y ejecutar lógica en el navegador.

Uso en el proyecto:

- Validación del formulario de inscripción:
 - Verificación de nombre no vacío.
 - Validación de teléfono solo numérico.
 - Mensajes de error antes de enviar.
- Eventos de navegación, desplazamiento y botones interactivos.

3.4 Git / GitHub

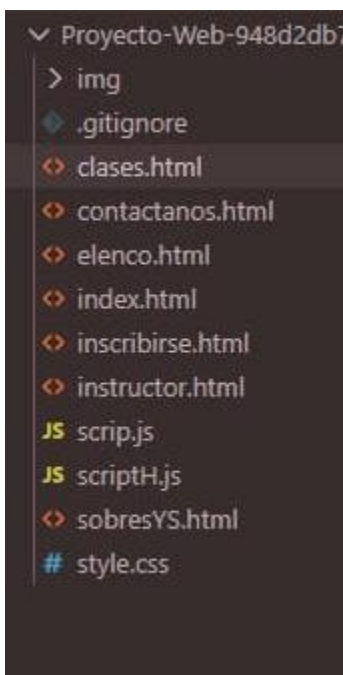
Función:

- Control de versiones y almacenamiento del código.
- Permite guardar cambios y volver a versiones anteriores.

Uso en el proyecto:

- Subida del código fuente del sitio.
- Control de cada actualización del proyecto.

4. Estructura de carpetas



/img

Todas las imágenes usadas

5. Funcionamiento de cada pagina

5.1 index.html (Inicio)

- Contiene el menú fijo superior
- Muestra un banner de presentación
- Incluye el botón **“Leer más”** → redirige a **sobre.html**
- Más abajo contiene el botón **Elenco** → redirige a **elenco.html**

Botones manejados mediante:

```
<a href="sobreYS.html">Leer más</a>
```

5.2 sobre.html (Sobre Young Soul)

- Describe historia, filosofía y objetivo de la academia
- Contiene imágenes y secciones informativas
- Se accede desde el botón **“Leer más”** en el inicio

5.3 elenco.html (Elenco)

- Muestras integrantes del elenco
- Fotografías
- Información de trayectoria

5.4 instructor.html

- Información del instructor
- Organizado con flexbox

5.5 clases.html

- Lista detallada de clases
- Incluye precio y horarios
- Videos con iframe de youtube

5.6 inscribirse.html

- Contiene un formulario con validación
- h.js controla:
 - Campos vacíos
 - Alertas al usuario
 - Evita el envío si hay errores

Ejemplo de función usada:

5.7 contacto.html

- Información de contacto
- Redes sociales
- Mapa

6. Diseño y Estilos (CSS)

- Se utiliza flexbox para acomodar secciones
- Se usa grid para contenido como elenco o clases
- Paleta de colores acorde a la identidad visual de Young Soul

- Uso de media queries para tablet y móvil
- Efectos como hover y transiciones para botones

7. Lógica General del Proyecto

La lógica del proyecto se basa en la interacción entre HTML (estructura), CSS (diseño) y JavaScript (comportamiento) para proporcionar un sitio web funcional, intuitivo y visualmente atractivo para la academia de baile *Young Soul*.

Aunque el sistema no incluye base de datos ni autenticación, toda la lógica se maneja desde el lado del cliente (navegador).

- El sitio está compuesto por varias páginas interconectadas mediante enlaces:
- El menú principal (logo, Instructores, Clases, Inscribirse y Contáctanos) permite acceder directamente a cada sección.
- En la página inicio, dos botones (“Leer más” e “Información del elenco”) redirigen a páginas específicas.
- Toda la navegación se realiza mediante enlaces HTML y no requiere recarga dinámica ni APIs.

7.2 Comportamiento del sitio

7.2.1 Botones y enlaces

Todos los botones del menú y secciones están configurados con la propiedad href para cambiar de página de forma directa.

Ejemplos de acciones:

Click al **logo** - retorna al inicio.

Click en **Instructores** - abre la página con información del equipo.

Click en **Leer más** - abre la página *Sobre Young Soul*.

Click en **Elenco** - abre la página del elenco.

No hay scripts complejos para navegar, todo es HTML estático.

7.5 Lógica de Responsividad

El diseño responde a pantallas pequeñas mediante:

- Media queries (@media (max-width: 768 y 1024))
- Ajustes de tamaño en:
 - imágenes
 - contenedores
 - botones
 - texto
- Uso de unidades relativas (rem, %)

Esto garantiza que la web se adapte correctamente a teléfonos, tablets y computadoras.

8.Codigos

8.1. Reset y Normalización del Estilo

```
* {
  margin: 0;
  padding: 0;
}
```

```

html{
    box-sizing: border-box; /* hace que los márgenes y bordes se incluyan dentro del
tamaño total de un elemento */

    font-size: 62.5%; /* hace que 1 rem = 10px (más fácil de manejar) */
}

*, ::before, ::after{
    box-sizing: inherit; /* garantiza que todos los elementos usen box-sizing: border-box */
}

```

Propósito: Este bloque sirve para resetear los márgenes y el padding por defecto de los navegadores, asegurando que el diseño sea consistente. El `box-sizing: border-box` es una técnica útil para controlar las dimensiones de los elementos, de manera que los márgenes y bordes no aumenten el tamaño total de un elemento.

8.2. Configuración del Header y Video de Fondo

```

.Header {
    position: relative; /* Permite colocar el video detrás sin interferir con otros elementos
*/

    width: 100%;

    height: 100vh; /* 100% de la altura de la ventana */

    overflow: hidden; /* Oculta lo que sobresale del contenedor */
}

.Header video {
    position: absolute; /* Permite mover el video independientemente */

    top: 50%;

    left: 50%;

    min-width: 100%;

    min-height: 100%;

    transform: translate(-50%, -50%); /* Centra el video */

    object-fit: cover; /* Hace que el video cubra toda el área sin deformarse */
}

```

```

    z-index: -1; /* Envía el video al fondo */
}

```

Propósito: Este bloque asegura que el video de fondo cubra toda la pantalla, sin importar el tamaño de la ventana, y que no se deforme. `position: absolute` permite mover el video de manera libre dentro del contenedor `.Header`, centrándolo con `transform: translate(-50%, -50%)`.

8.3. Navegación del Header

```

.Header__nav {
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    padding: 3rem 6rem;
    display: flex;
    justify-content: space-between;
    align-items: center;
    color: white;
    font-family: 'Encode Sans Semi Condensed', sans-serif;
    z-index: 2; /* Asegura que el menú esté sobre el video */
}

.Header__nav__logoText {
    position: fixed;
    color: white;
    font-size: 3rem;
    font-weight: 900;
    text-decoration: none;
    text-shadow: 0px 0px 15px black;
}

```

```
}
```

Propósito: Este bloque configura la barra de navegación fija en la parte superior. Los elementos dentro del `.Header__nav` están alineados usando `flex`, con espacio entre el logo y los enlaces. También se establece un color de texto blanco y una sombra para que el logo sea visible sobre el video.

8.4. Estilo de los Links de Navegación

```
.Header__nav__menu a {
  font-weight: 900;
  color: white;
  text-decoration: none;
  font-size: 2.6rem;
  transition:
    color 0.3s,
    font-size 0.3s;
}

.Header__nav__menu a:hover {
  border-radius: 5rem;
  padding: 0.7rem 1.5rem;
  color: black;
  font-size: 3.2rem;
  background-color: rgb(255, 71, 59);
}
```

Propósito: Se configura el estilo de los enlaces en la navegación, como el tamaño de la fuente y el color. También se agrega una transición para suavizar el cambio de color y tamaño de fuente cuando el usuario pasa el mouse sobre los enlaces.

8.5. Estilos para Secciones

```
.SectionUno {
    border-bottom: solid 0.2rem rgb(255, 255, 255);
    height: 95rem;
    background-color: rgb(18, 18, 18);
    padding: 5rem;
    text-align: center;
    color: white;
}

.SectionUno__lema {
    padding-top: 10rem;
    font-size: 4.5rem;
    font-family: 'Poppins', sans-serif;
    font-weight: bold;
}

.SectionUno__lema span {
    font-size: 12rem;
    font-weight: 900;
}
```

Propósito: En este bloque, la primera sección `.SectionUno` tiene un fondo oscuro y un borde blanco en la parte inferior. Además, la sección del lema `.SectionUno__lema` tiene un gran tamaño de fuente.

8.6. Botón con Efecto Hover

```
.SectionUno__LeerMas {
    display: inline-flex;
    align-items: center;
    font-size: 2.5rem;
```

```

    color: white;

    text-decoration: none;

    font-family: 'Encode Sans Semi Condensed', sans-serif;

    border-bottom: 1px solid white;

    padding-bottom: 1rem;
}

.SectionUno__LeerMas:hover .SectionUno__Flecha {
    transform: translateX(1rem); /* Mueve la flecha al pasar el mouse */
}

```

Propósito: El botón "Leer Más" tiene un efecto de hover que mueve la flecha a la derecha (`translateX(1rem)`), generando un pequeño pero efectivo movimiento visual.

8.7. Contenedor de Dos Secciones con Imagen de Fondo

```

.ContenedorSectDosTres {
    display: flex;

    justify-content: center;

    height: 120rem;

    text-align: center;
}

.ContenedorSectDosTres__SectionDos {
    width: 50%;

    background-image: linear-gradient(rgba(0, 0, 0, 0.5), rgba(0, 0, 0, 0.5)),
    url(img/imgDancer.jpg);

    background-size: cover;

    background-position: center;
}

```

Propósito: La clase `.ContenedorSectDosTres` organiza dos secciones en un contenedor flexbox. La segunda sección (`.ContenedorSectDosTres__SectionDos`) tiene una imagen de fondo con un gradiente para oscurecer la imagen y hacer que el texto sea más legible.

8.7 Footer (Pie de Página)

```
.PiePag {  
    width: 100%;  
    height: 60rem;  
    background: white;  
    display: flex;  
    flex-direction: column;  
    align-items: center;  
    justify-content: center;  
}
```

En resumen, todos los estilos mostrados en esta parte del proyecto ayudan a que la página se vea ordenada, bonita y funcione bien en cualquier pantalla. El reset inicial asegura que todos los navegadores muestren el diseño de la misma manera, y luego cada bloque de código cumple una función importante: el video de fondo ocupa toda la pantalla, el menú se mantiene visible arriba, los enlaces y botones tienen efectos cuando pasas el mouse, y las secciones usan imágenes, colores y tamaños pensados para que el contenido sea fácil de leer.

En conjunto, estos estilos hacen que el sitio tenga una apariencia más profesional, moderna y agradable para el usuario, manteniendo siempre una buena organización y una presentación clara