

Hoja de Presentación

Nombre del Proyecto:

Landing Page Interactiva con HTML, CSS, BootStrap y JS

Integrantes del grupo:

Jhon Andres Yate Gueche

Jaider Alejandro Muñoz Ortega

Roger Alejandro Osorio Mondragón

Asignaturas:

Gestión de Bases de Datos I

Introducción a los Lenguajes de Programación

Fecha de Entrega:

[30/09/2024]

Docente:

Jaider Javier De La Rosa Berte

Objetivo de la actividad

El objetivo de este proyecto es desarrollar una landing page interactiva para una tienda gamer que usando HTML, CSS, JAVASCRIPT y que proporcione una experiencia de usuario excepcional. La página se diseñará siguiendo el modelo ER de la base de datos, asegurando que la estructura de contenido sea coherente y organizada. Se busca crear un diseño atractivo y moderno que capte la atención del usuario, una interfaz intuitiva que facilite la navegación y la adaptabilidad a diversos dispositivos, garantizando así una experiencia óptima tanto en desktop como en móviles.

Descripción del Proceso Realizado

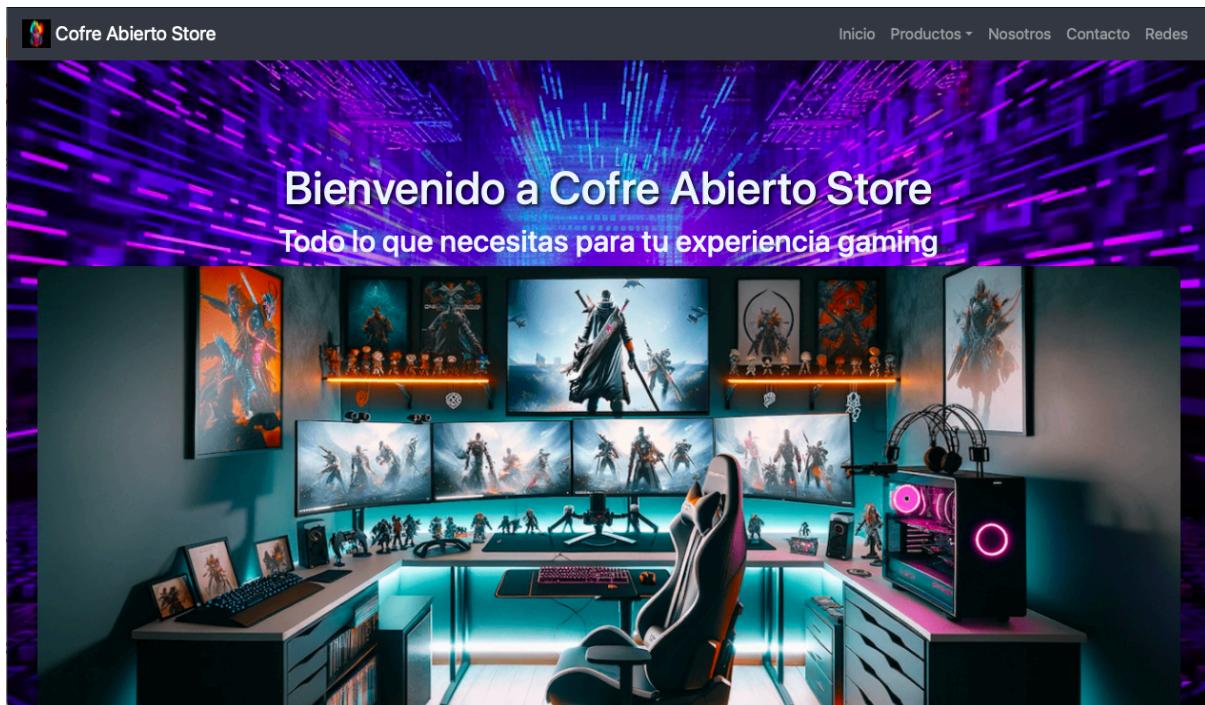
El desarrollo de la landing page interactiva para la tienda gamer se llevó a cabo en varias etapas, cada una de las cuales contribuyó a la creación de un producto final funcional. A continuación se describen las etapas clave del proceso:

- **Investigación y Planificación:**
 - **Definición de Objetivos:** Se establecieron objetivos claros para la página, incluyendo el público objetivo y las funcionalidades deseadas.
- **Diseño del Modelo ER:**
 - **Modelo de Datos:** Se creó un modelo entidad-relación (ER) para organizar la información que se mostraría en la landing page, lo que facilitó la estructura del contenido y la navegación.
- **Prototipado:**
 - **Wireframes:** Se desarrollaron bocetos (wireframes) para visualizar la estructura de la página, incluyendo la disposición de los elementos clave como banners, productos destacados y secciones informativas.
 - **Mockups:** Se diseñaron mockups visuales que incorporaron la paleta de colores, tipografías y elementos gráficos que representarían la identidad de la tienda.
- **Desarrollo Frontend:**
 - **HTML:** Se construyó la estructura básica de la página utilizando HTML, asegurando la semántica y accesibilidad del contenido.
 - **CSS:** Se aplicaron estilos CSS para crear un diseño atractivo y responsive, para una disposición adecuada en diferentes tamaños de pantalla.
 - **JavaScript:** Se implementaron interacciones dinámicas, como carruseles de productos, animaciones al desplazarse y formularios interactivos, mejorando la experiencia del usuario.

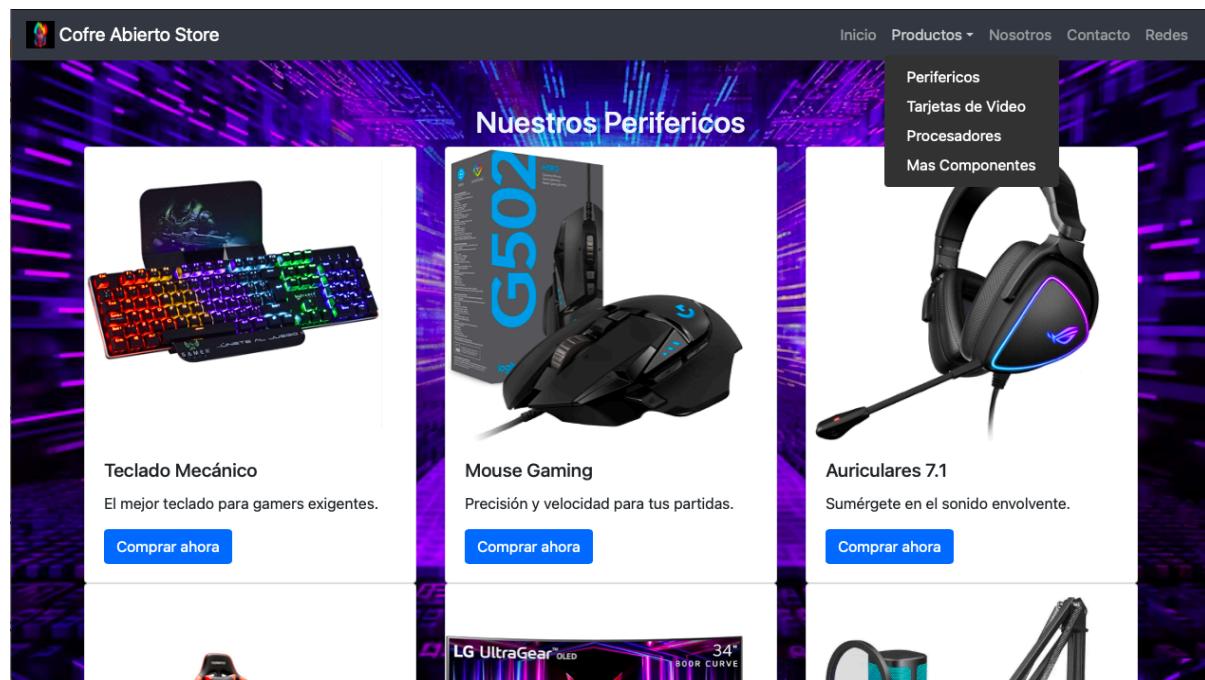
Este proceso integral permitió crear una landing page que no solo cumple con los objetivos establecidos, sino que también ofrece una experiencia atractiva y funcional para los usuarios interesados en productos gamer.

Guia de uso

- **Navegación:** Parte superior, enlaces a todas las secciones: Inicio, Productos, Nosotros, Contacto, Redes sociales.
- **Inicio:** Mensaje de bienvenida



- **Productos:** Todos nuestros productos disponibles divididos en diferentes tipos.



- **Sobre nosotros:** Información acerca de nuestro negocio.
- **Contactanos:** Formularios para que el usuario se contacte directamente con el negocio.
- **Redes sociales:** las redes sociales en las que nos puedes encontrar.

Sobre Nosotros

En Cofre Abierto Store, nos especializamos en ofrecerte el mejor equipo para que lleves tu experiencia de gaming al siguiente nivel.

Contáctanos

Nombre

Email

Mensaje

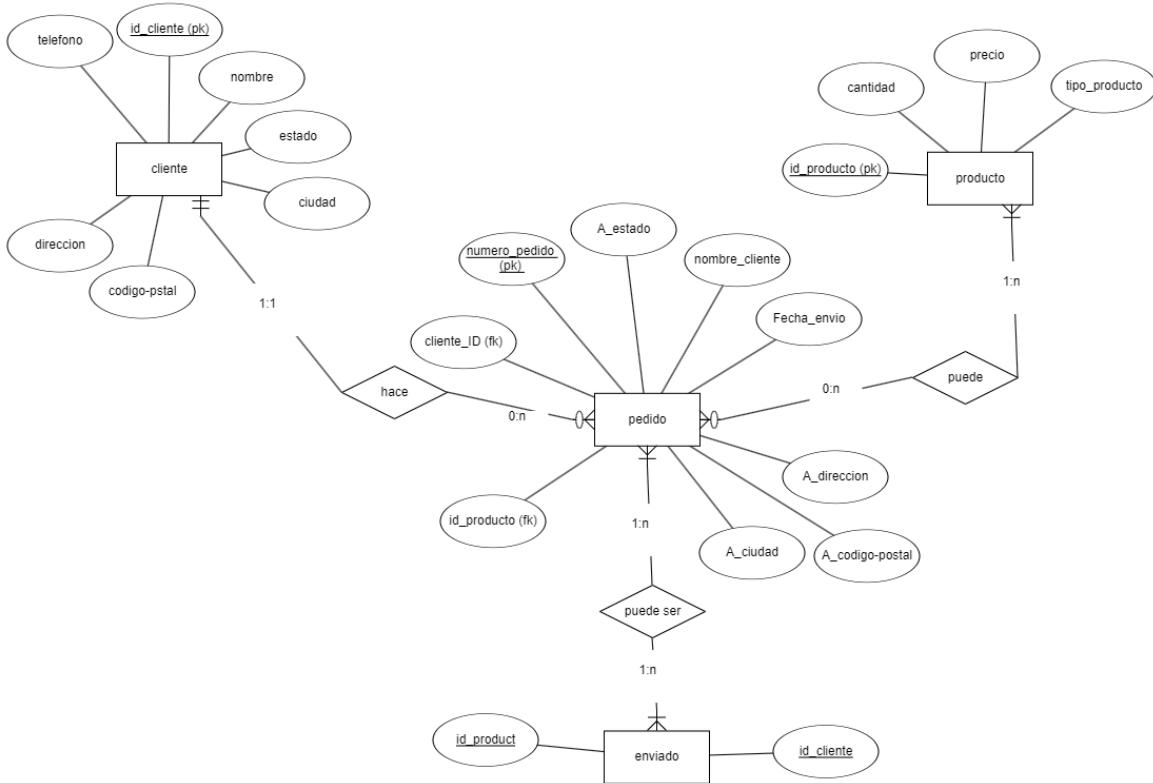
Síguenos en nuestras Redes

Esquema De Relación

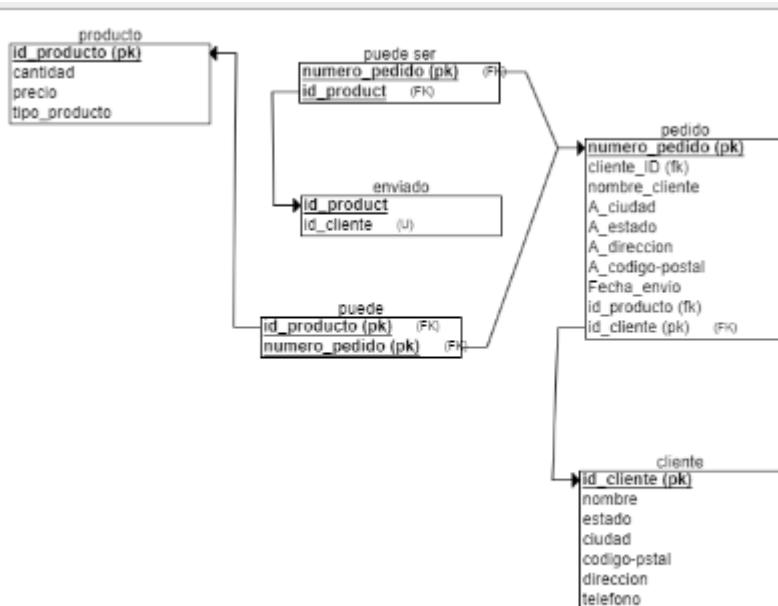
Tablas De relaciones

	Cliente	Pedido	Producto	Enviado
Cliente	X	0-n	-	1-n
Pedido	n-1	X	1-n	1-n
Producto	-	n-1	X	X
Enviado	1-n	1-n	-	X

Modelo Entidad Relación (ER)



Esquema De Relación



Repositorio Git Hub: <https://github.com/Alejo0911/Landing-Page.git>

Estructura de la landing page

Imagen de la estructura de nuestra landing page html:

```
① Proyecto_Final_Ciclo > index.HTML > html > body
1   <!DOCTYPE html>
2   <html lang="es">
3     <head>
4       <meta charset="UTF-8">
5       <meta name="viewport" content="width=device-width, initial-scale=1.0">
6       <meta http-equiv="X-UA-Compatible" content="ie=edge">
7       <title>Cofre Abierto Store</title>
8       <meta name="robots" content="index, follow">
9       <meta name="theme-color" content="#009f">
10      <meta name="description" content="Página dedicada a la venta de componentes y artículos Gamer">
11      <link rel="icon" type="IMG" href="IMG/cofre_abierto2.png">
12
13      <!-- Enlace a CSS personalizado -->
14      <link rel="stylesheet" href="Styles_CSS/styles_pf.css">
15
16      <!-- Enlace a Bootstrap -->
17      <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" rel="stylesheet">
18
```

<!DOCTYPE html>: Esta declaración le dice al navegador que el documento es un archivo HTML5. Ayuda a asegurar que el navegador renderice la página de la manera más estándar posible.

<html lang="es">: Esta etiqueta abre el documento HTML y especifica que el contenido está en español (**lang="es"**). Esto es importante para la accesibilidad y el SEO, ya que indica a los motores de búsqueda y a los lectores de pantalla el idioma principal del contenido.

<head>: Esta etiqueta contiene metadatos sobre el documento, que no se muestran directamente en la página, pero son importantes para el funcionamiento del mismo.

<meta charset="UTF-8" >: Esta etiqueta establece la codificación de caracteres para el documento como UTF-8, que permite representar casi todos los caracteres y símbolos de todos los idiomas.

<meta name="viewport" content="width=device-width, initial-scale=1.0" >: Esta etiqueta es crucial para el diseño responsive. Le dice al navegador cómo controlar las dimensiones y el escalado de la página en dispositivos móviles.

width=device-width establece el ancho del viewport al ancho del dispositivo, y **initial-scale=1.0** establece el nivel de zoom inicial.

<meta http-equiv="X-UA-Compatible" content="ie=edge">: Esta etiqueta es específica para Internet Explorer. Indica al navegador que use la versión más reciente de su motor de renderizado.

<title>Cofre Abierto Store</title>: Esta etiqueta define el título del documento, que aparece en la pestaña del navegador y es importante para el SEO.

<meta name="robots" content="index, follow">: Esta etiqueta indica a los motores de búsqueda que deben indexar la página y seguir los enlaces en ella.

<meta name="theme-color" content="#09f">: Esta etiqueta permite que los navegadores móviles cambien el color de la barra de herramientas del navegador al color especificado cuando el usuario visita la página.

<meta name="description" content="Página dedicada a la venta de componentes y artículos Gamer">: Esta etiqueta proporciona una breve descripción del contenido de la página, que a menudo se muestra en los resultados de búsqueda.

<link rel="icon" type="IMG" href="IMG/cofre_abierto2.png">: Esta etiqueta establece un ícono (favicon) que se muestra en la pestaña del navegador. Sin embargo, el tipo debería ser **image/png** en lugar de **IMG**.

<link rel="stylesheet" href="Styles_CSS/styles_pf.css">: Esta etiqueta enlaza un archivo CSS externo que contiene estilos personalizados para tu página.

<link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" rel="stylesheet">: Esta etiqueta enlaza a la hoja de estilos de Bootstrap, un framework CSS popular que facilita el diseño responsive y la estilización de componentes web.

Segunda imagen continuando la estructura html:

```
<!-- Encabezado / Navegación -->
<header class="navbar navbar-expand-lg navbar-dark bg-dark">
  <a class="navbar-brand" href="#">
    
    Cofre Abierto Store</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="navbarNav">
    <ul class="navbar-nav ml-auto">
      <li class="nav-item">
        <a class="nav-link" href="index.HTML">Inicio</a>
      </li>
      <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="Perifericos.HTML" id="navbarDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">Productos
          <div class="dropdown-menu" aria-labelledby="navbarDropdown">
            <a class="dropdown-item" href="Perifericos.HTML">Perifericos</a>
            <a class="dropdown-item" href="Tarjetas_Video.HTML">Tarjetas de Video</a>
            <a class="dropdown-item" href="Procesadores.HTML">Procesadores</a>
            <a class="dropdown-item" href="Mas_Componentes.HTML">Mas Componentes</a>
          </div>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#nosotros">Nosotros</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#contacto">Contacto</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#redes">Redes</a>
        </li>
    </ul>
  </div>
</header>
```

`<header class="navbar navbar-expand-lg navbar-dark bg-dark">`: Esta etiqueta define el encabezado de la página. Utiliza clases de Bootstrap para crear una barra de navegación:

navbar: Define que se trata de una barra de navegación.

navbar-expand-lg: Indica que la barra se expandirá en pantallas grandes y se colapsará en pantallas más pequeñas.

navbar-dark: Aplica un esquema de color oscuro a la barra.

bg-dark: Establece un fondo oscuro.

``: Este es un enlace que actúa como el logo o nombre de la marca. El **href="#"** indica que actualmente no lleva a ninguna página. Dentro de esta etiqueta, se encuentra la imagen del logo y el nombre de la tienda.

``: Esta etiqueta inserta una imagen en el encabezado:

src: Especifica la ubicación del archivo de imagen.

alt: Proporciona un texto alternativo que se mostrará si la imagen no se carga.

width y **height**: Establecen el tamaño de la imagen.

class="d-inline-block align-top": Clases de Bootstrap que ayudan a alinear la imagen correctamente en la barra de navegación.

`<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">`:

Este botón se utiliza para mostrar y ocultar el menú de navegación en pantallas pequeñas:

class="navbar-toggler": Estilo de Bootstrap para el botón de colapso.

type="button": Indica que es un botón.

data-toggle="collapse": Indica que este botón controlará un elemento colapsable.

data-target="#navbarNav": Especifica qué elemento se colapsará (en este caso, el menú de navegación).

aria-controls, aria-expanded, aria-label: Atributos que mejoran la accesibilidad, indicando a los lectores de pantalla qué hace el botón.

``: Este span se utiliza para mostrar el ícono del botón de colapso (tres líneas horizontales, comúnmente llamado "hamburguesa").

`<div class="collapse navbar-collapse" id="navbarNav">`: Este div contiene el contenido del menú de navegación que se colapsa:

collapse: Clase de Bootstrap que aplica el comportamiento colapsable.

navbar-collapse: Indica que este div es parte de la barra de navegación.

id="navbarNav": ID único que se utiliza para que el botón de colapso lo controle.

`<ul class="navbar-nav ml-auto">`: Esta lista no ordenada contiene los elementos del menú de navegación:

navbar-nav: Clase de Bootstrap que aplica estilos a la lista de navegación.

ml-auto: Clase que aplica margen automático a la izquierda, empujando la lista hacia la derecha.

`<li class="nav-item">`: Cada elemento de la lista representa un ítem en la navegación:

nav-item: Clase de Bootstrap que aplica estilos específicos a los elementos de navegación.

`Inicio`: Este enlace es un elemento del menú de navegación:

nav-link: Clase de Bootstrap que estiliza el enlace para que se ajuste a la barra de navegación.

`<li class="nav-item dropdown">`: Este es un elemento de menú que tiene un submenú (desplegable):

dropdown: Clase que indica que este elemento tiene un menú desplegable.

`Productos`: Este enlace es el que activa el menú desplegable:

dropdown-toggle: Clase que indica que este enlace puede abrir un menú desplegable.

data-toggle="dropdown": Indica que este enlace controla la apertura del menú desplegable.

<div class="dropdown-menu" aria-labelledby="navbarDropdown">: Este div contiene el menú desplegable:

dropdown-menu: Clase de Bootstrap que aplica estilos al menú desplegable.

aria-labelledby: Indica a qué elemento está relacionado el menú desplegable para mejorar la accesibilidad.

Perifericos:

Estos enlaces dentro del menú desplegable representan las opciones del submenú:

dropdown-item: Clase de Bootstrap que estiliza los enlaces del menú desplegable.

Los demás enlaces

Los otros enlaces (**Nosotros, Contacto, Redes**) funcionan de manera similar, cada uno representando una opción en la barra de navegación.

Tercera Imagen de la estructura html:

```
<!-- Hero Section -->
<section class="hero text-center text-black" id="inicio">
  <div class="container-storage">
    <h1 style="color: #aliceblue">Bienvenido a Cofre Abierto Store</h1>
    <h2 style="color: #aliceblue">Todo lo que necesitas para tu experiencia gaming</h2>
    
    <a href="#" class="btn btn-primary">Explorar Productos</a>
  </div>
</section>

<!-- Sección de Productos -->
<section id="perifericos" class="py-5">
  <div class="container">
    <h2 style="color: #aliceblue" class="text-center">Algunos de nuestros Productos</h2>
    <div class="row">

      <!-- Producto 1 -->
      <div class="col-md-4">
        <div class="card">
          
          <div class="card-body">
            <h5 class="card-title">Teclado Mecánico</h5>
            <p class="card-text">El mejor teclado para gamers exigentes.</p>
            <a href="#" class="btn btn-primary">Comprar ahora</a>
          </div>
        </div>
      </div>
    </div>
  </div>
</section>
```

<section>: Esta etiqueta define una sección de contenido dentro de la página. Ayuda a estructurar el documento en partes temáticas.

class="hero text-center text-black": Estas clases aplican estilos de Bootstrap y personalizados.

hero: Usualmente se refiere a una sección destacada, que puede tener un fondo atractivo y gran texto.

text-center: Centra el texto dentro de la sección.

text-black: Aplica un color de texto negro.

<div>: Esta etiqueta define un contenedor genérico.

class="container-storage": Una clase personalizada que probablemente aplica estilos específicos a este contenedor. Es común usar divs para agrupar elementos y aplicar estilos o comportamientos.

class="img-fluid": Esta clase de Bootstrap hace que la imagen sea responsive, es decir, que se ajuste al tamaño del contenedor, manteniendo su relación de aspecto.

<section id="perifericos" class="py-5">

id="perifericos": Asigna un identificador único a esta sección, lo que permite que se pueda vincular a ella directamente desde otras partes del documento.

class="py-5": Esta clase de Bootstrap aplica un margen vertical (padding) de 5 unidades, lo que proporciona espacio adicional arriba y abajo de la sección.

<div class="container">: Este div actúa como un contenedor para alinear el contenido de manera adecuada. La clase **container** de Bootstrap establece un ancho fijo y centrado para el contenido.

<div class="row">: Esta etiqueta se utiliza dentro de un contenedor para crear una fila que alinea columnas de Bootstrap. Las filas se utilizan para organizar el contenido en columnas.

<div class="col-md-4">

col-md-4: Esta clase de Bootstrap define que el elemento ocupará 4 de las 12 columnas disponibles en dispositivos medianos (pantallas a partir de 768px). Esto permite crear un diseño en cuadrícula.

<div class="card">: Esta etiqueta define un componente de tarjeta de Bootstrap, que es una forma de mostrar contenido de manera atractiva. Las tarjetas suelen incluir imágenes, texto y enlaces.

```

```

class="card-img-top": Esta clase de Bootstrap aplica estilos a la imagen para que se ajuste correctamente al área superior de la tarjeta.

<div class="card-body">: Este div contiene el contenido principal de la tarjeta (título, texto, y botón).

<h5 class="card-title">: Este encabezado representa el título del producto dentro de la tarjeta. La clase **card-title** aplica estilos específicos para el título en el contexto de una tarjeta.

<p class="card-text">: Este párrafo proporciona una breve descripción del producto dentro de la tarjeta. La clase **card-text** aplica estilos para el texto dentro de la tarjeta.

Comprar ahora: Este enlace actúa como un botón para comprar el producto.

btn: Clase de Bootstrap que aplica estilos de botón.

btn-primary: Clase que aplica un estilo de botón principal, normalmente un color destacado.

Este tipo de etiquetas se continúa asignando a todos los productos de las diferentes categorías en la página.

Cuarta imagen de la estructura html:

```
<!-- Sección Nosotros -->  
<section id="nosotros" class="py-5 bg-light">  
  <div class="container">  
    <h2 class="text-center">Sobre Nosotros</h2>  
    <p class="text-center">En Cofre Abierto Store, nos especializamos en ofrecerte el mejor equipo para que lleves tu experiencia de gaming al siguiente nivel.</p>  
  </div>  
</section>  
  
<!-- Sección Contacto -->  
<section id="contacto" class="py-5">  
  <div class="container">  
    <h2 style="color: #aliceblue" class="text-center">Contáctanos</h2>  
    <form id="contacto-form">  
      <div class="form-row">  
        <div class="col-md-6 mb-3">  
          <label style="color: #aliceblue" for="nombre">Nombre</label>  
          <input type="text" class="form-control" id="nombre" placeholder="Tu nombre" required>  
        </div>  
        <div class="col-md-6 mb-3">  
          <label style="color: #aliceblue" for="email">Email</label>  
          <input type="email" class="form-control" id="email" placeholder="Tu email" required>  
        </div>  
      </div>  
      <div class="form-group">  
        <label style="color: #aliceblue" for="mensaje">Mensaje</label>  
        <textarea class="form-control" id="mensaje" rows="3" placeholder="Tu mensaje" required></textarea>  
      </div>  
      <button class="btn btn-primary" type="submit">Enviar</button>  
    </form>  
  </div>  
</section>
```

<section id="nosotros" class="py-5 bg-light">

<section>: Define una nueva sección de contenido en la página.

id="nosotros": Proporciona un identificador único a la sección, que se puede utilizar para enlazar directamente a ella.

class="py-5 bg-light":

py-5: Aplica un margen vertical (padding) de 5 unidades.

bg-light: Aplica un fondo claro a la sección.

<h2 class="text-center">Sobre Nosotros</h2>

Este encabezado de nivel 2 se utiliza para el título de la sección.

class="text-center": Centra el texto horizontalmente en la sección.

<p class="text-center">En Cofre Abierto Store, nos especializamos en ofrecerte el mejor equipo para que lleves tu experiencia de gaming al siguiente nivel.</p>

Este párrafo proporciona una breve descripción sobre la tienda.

class="text-center": Nuevamente, centra el texto.

<section id="contacto" class="py-5">

id="contacto": Identificador único para esta sección.

class="py-5": Aplica un margen vertical (padding) de 5 unidades.

<h2 style="color:aliceblue" class="text-center">Contáctanos</h2>

Este encabezado de nivel 2 actúa como el título de la sección de contacto.

style="color:aliceblue": Aplica un color específico (azul claro) al texto.

<form id="contacto-form">

Esta etiqueta define un formulario para la entrada de datos del usuario.

id="contacto-form": Asigna un identificador único al formulario, que se puede usar para referenciarlo en JavaScript o CSS.

<div class="form-row">: Este div se utiliza para agrupar y organizar los elementos de formulario en una fila. Permite una disposición más flexible en las columnas.

```
<div class="col-md-6 mb-3">
```

col-md-6: Esta clase de Bootstrap hace que el div ocupe 6 de las 12 columnas disponibles en dispositivos medianos. Esto significa que dos de estos divs se alinearán en una fila en pantallas medianas o más grandes.

mb-3: Aplica un margen inferior de 3 unidades para separar este div del siguiente.

```
<label style="color:aliceblue" for="nombre">Nombre</label>
```

La etiqueta **label** describe el campo de entrada.

style="color:aliceblue": Establece el color de texto en azul claro.

for="nombre": Este atributo vincula la etiqueta con el campo de entrada correspondiente, mejorando la accesibilidad.

```
<input type="text" class="form-control" id="nombre" placeholder="Tu nombre" required>
```

Este campo de entrada permite al usuario ingresar su nombre.

type="text": Especifica que es un campo de texto.

class="form-control": Aplica estilos de Bootstrap para mejorar la apariencia del campo.

id="nombre": Asigna un identificador único al campo, que se puede usar en el formulario.

placeholder="Tu nombre": Proporciona un texto de ayuda que se muestra dentro del campo hasta que el usuario ingresa su propio texto.

required: Este atributo indica que el campo es obligatorio y debe ser completado antes de enviar el formulario.

```
<input type="email" class="form-control" id="email" placeholder="Tu email" required>
```

Este campo es similar al anterior, pero está diseñado para ingresar una dirección de correo electrónico.

type="email": Especifica que el campo está destinado a recibir un correo electrónico.

```
<div class="form-group">
```

Este div agrupa los elementos de formulario relacionados. En este caso, agrupa la etiqueta y el área de texto para el mensaje.

```
<label style="color:aliceblue" for="mensaje">Mensaje</label>
```

Similar a la etiqueta anterior, describe el campo del mensaje.

for="mensaje": Vínculo con el área de texto correspondiente.

```
<textarea class="form-control" id="mensaje" rows="3" placeholder="Tu  
mensaje" required></textarea>
```

Este elemento permite al usuario ingresar texto más largo.

class="form-control": Aplica estilos de Bootstrap.

id="mensaje": Identificador único para el área de texto.

rows="3": Establece la altura del área de texto en 3 líneas.

placeholder="Tu mensaje": Texto de ayuda para indicar qué se debe ingresar.

required: Indica que este campo también es obligatorio.

```
<button class="btn btn-primary" type="submit">Enviar</button>
```

Este botón envía el formulario.

class="btn btn-primary": Estilos de Bootstrap que hacen que el botón tenga un aspecto prominente y atractivo.

type="submit": Indica que el botón enviará el formulario.

Quinta imagen de la estructura html:

```

<!-- Sección Redes-->
<section id="redes" class="py-5">
    <div class="container">
        <h2 style="color: #aliceblue" class="text-center">Siguenos en nuestras Redes</h2>
        <ul class="list-unstyled d-flex justify-content-center flex-wrap">
            <div class="form-row">
                <li class="nav-item">
                    <a class="nav-link" href="#">
                        Facebook</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="#">
                        Instagram</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="#">
                        TikTok</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="#">
                        Gmail</a>
                </li>
            </div>
        </ul>
    </div>
</section>

<!-- Pie de página -->
<footer class="bg-dark text-white py-3 text-center">
    <p>&copy; 2024 Cofre Abierto. Todos los derechos reservados.</p>
</footer>

<!-- Scripts de Bootstrap y JavaScript personalizado -->
<script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
<script src="Script_JS/script_pf.js"></script>

</body>
</html>

```

<section id="redes" class="py-5">

id="redes": Proporciona un identificador único para esta sección, que permite enlazar a ella desde otras partes del documento.

class="py-5": Aplica un margen vertical (padding) de 5 unidades, proporcionando espacio arriba y abajo de la sección.

<h2 style="color:#aliceblue" class="text-center">Siguenos en nuestras Redes</h2>

Este encabezado de nivel 2 actúa como el título de la sección, informando a los usuarios que pueden seguir a la tienda en redes sociales.

style="color:#aliceblue": Aplica un color azul claro al texto.

<ul class="list-unstyled d-flex justify-content-center flex-wrap">

****: Define una lista desordenada.

class="list-unstyled": Elimina el estilo predeterminado de la lista (como viñetas).

d-flex: Aplica el modelo de diseño flexible (Flexbox) de Bootstrap, permitiendo que los elementos se alineen en una línea horizontal.

justify-content-center: Centra horizontalmente los elementos dentro del contenedor flex.

flex-wrap: Permite que los elementos de la lista se envuelvan en una nueva línea si no hay suficiente espacio horizontal.

```
<div class="form-row">
```

Este div se utiliza para agrupar los elementos de la lista de redes sociales en una fila, aunque aquí parece ser un uso innecesario, ya que normalmente se usa para formularios.

```
<li class="nav-item">
```

Esta etiqueta define un elemento de lista que se usa en el contexto de navegación.

class="nav-item": Clase de Bootstrap que estiliza el elemento como un elemento de navegación.

```
<a class="nav-link" href="#">
```

Este enlace actúa como un vínculo a las redes sociales.

class="nav-link": Clase de Bootstrap que aplica estilos para un enlace de navegación.

```

```

Esta etiqueta inserta una imagen que representa la red social (en este caso, Facebook).

src="IMG/logo_facebook.png": Especifica la ruta de la imagen.

alt="Facebook": Proporciona un texto alternativo para la imagen, útil para accesibilidad.

width="30" y height="30": Establece dimensiones específicas para la imagen.

class="social-icon": Clase personalizada que probablemente aplica estilos específicos a los íconos de redes sociales.

Las siguientes etiquetas **** para Instagram, TikTok y Gmail son similares a las que hemos discutido para Facebook, cambiando solo la imagen y el texto alternativo correspondiente.

```
<footer class="bg-dark text-white py-3 text-center">
```

<footer>: Esta etiqueta define el pie de página de la página.

```
class="bg-dark text-white py-3 text-center":
```

bg-dark: Aplica un fondo oscuro.

text-white: Cambia el color del texto a blanco.

py-3: Aplica un margen vertical de 3 unidades.

text-center: Centra el contenido horizontalmente.

```
<p>&copy; 2024 Cofre Abierto. Todos los derechos reservados.</p>
```

Este párrafo contiene el aviso de copyright.

©: Entidad HTML que representa el símbolo de copyright.

```
<script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
```

Este script carga la biblioteca jQuery, que es útil para realizar tareas de manipulación del DOM y para facilitar la escritura de scripts JavaScript.

```
<script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"></script>
```

Carga Popper.js, que es una dependencia de Bootstrap utilizada para manejar elementos emergentes como menús desplegables.

```
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
```

Carga el archivo JavaScript de Bootstrap, que agrega interactividad a varios componentes de Bootstrap.

```
<script src="Script_JS/script_pf.js"></script>
```

Este script se refiere a un archivo JavaScript personalizado (en este caso `script_pf.js`), donde se pueden incluir funciones específicas para la funcionalidad de la página.

Cierre de las etiquetas

`</body>` Cierra la etiqueta `<body>`, indicando el final del contenido del cuerpo de la página.

`</html>` Cierra la etiqueta `<html>`, indicando el final del documento HTML.

Estructura del archivo CSS:

```
① Proyecto_Final_Ciclo > Styles_CSS > styles_pf.css > ...
1  /* Estilos generales */
2  body {
3      font-family: 'Arial', sans-serif;
4      background-image: url('https://img.freepik.com/fotos-
5      background-size: contain;
6  }
7
8  .hero {
9      height: 125vh;
10     display: flex;
11     align-items: center;
12     justify-content: center;
13     text-align: center;
14 }
15
16 .hero img {
17     width: 95%;
18     height: auto;
19     object-fit: cover;
20     border-radius: 10px;
21 }
22
23 .hero h1 {
24     font-size: 3rem;
25     text-shadow: 2px 2px 5px #rgb(0, 0, 0);
26 }
27
28 .hero p {
29     font-size: 1.5rem;
30     margin-bottom: 20px;
31 }
32
33 #Perifericos .card img {
34     width: 90%;
35     height: 320px;
36     object-fit: contain;
37     object-position: 100%;
38 }
39
40 .card {
41     transition: transform 0.3s;
42 }
```

font-family: 'Arial', sans-serif;: Establece la fuente del texto en todo el cuerpo de la página. Si 'Arial' no está disponible, el navegador usará cualquier fuente sans-serif como respaldo.

background-image: url(...);: Define una imagen de fondo para el cuerpo de la página. Se utiliza una URL que apunta a una imagen específica.

background-size: contain;: Ajusta la imagen de fondo para que se contenga dentro del área del elemento (sin recortar), manteniendo su relación de aspecto.

Regla para la clase .hero

height: 125vh;: Establece la altura de la sección .hero en 125% de la altura de la ventana del navegador (viewport).

display: flex;: Utiliza el modelo de diseño flexible (Flexbox) para organizar el contenido dentro de este contenedor.

align-items: center;: Centra verticalmente los elementos dentro del contenedor flex.

justify-content: center;: Centra horizontalmente los elementos dentro del contenedor flex.

text-align: center;: Centra el texto dentro de la sección.

Regla para imágenes dentro de .hero

width: 95%;: Define que la imagen ocupe el 95% del ancho del contenedor padre.

height: auto;: Mantiene la altura proporcional a la anchura, evitando distorsiones.

object-fit: cover;: Asegura que la imagen cubra completamente el área del contenedor, recortando cualquier parte que sobrepase el área, si es necesario.

border-radius: 10px;: Aplica bordes redondeados a la imagen con un radio de 10 píxeles.

Regla para encabezados dentro de .hero

font-size: 3rem;: Define el tamaño de fuente del encabezado `<h1>` en 3 rem (un rem es igual al tamaño de fuente raíz del documento, por lo general 16px, así que esto sería aproximadamente 48px).

text-shadow: 2px 2px 5px rgb(0, 0, 0);: Aplica una sombra al texto, desplazada 2 píxeles a la derecha y 2 píxeles hacia abajo, con un desenfoque de 5 píxeles y color negro.

Regla para párrafos dentro de .hero

font-size: 1.5rem;: Establece el tamaño de fuente del párrafo en 1.5 rem (aproximadamente 24px).

margin-bottom: 20px;: Aplica un margen inferior de 20 píxeles, separando el párrafo de los elementos que están debajo.

Regla para imágenes dentro de tarjetas en la sección "**Periféricos**"

width: 90%;: Define que la imagen ocupe el 90% del ancho del contenedor padre.

height: 320px;: Establece la altura de la imagen en 320 píxeles.

object-fit: contain;: Mantiene la imagen dentro del contenedor sin recortarla, manteniendo su relación de aspecto.

object-position: 100%;: Posiciona la imagen completamente en la esquina inferior derecha del contenedor, si el espacio no es suficiente.

Regla para las tarjetas

transition: transform 0.3s;: Aplica una transición suave a las transformaciones (por ejemplo, escalado) que ocurren en la tarjeta, durando 0.3 segundos.

Regla para el efecto de hover en las tarjetas

.card:hover: Se aplica cuando el mouse pasa sobre la tarjeta.

transform: scale(1.005);: Escala la tarjeta un 0.5% más grande cuando el usuario pasa el mouse sobre ella, creando un efecto de aumento.

Segunda imagen de la estructura de CSS

```
/* Efectos en botones */
.btn-primary {
    background-color: #ff5722;
    border: none;
}

.btn-primary:hover {
    background-color: #e64a19;
}

/* Sección de Contacto */
form .form-control {
    border-radius: 5px;
}

footer {
    background-color: #111;
}

/* Estilo personalizado para el menú desplegable */
.navbar .dropdown-menu {
    background-color: #333;
    border: none;
}

.navbar .dropdown-item {
    color: #fff;
}

.navbar .dropdown-item:hover {
    background-color: #444;
    color: #ff1cf7;
}

.social-icon {
    margin-right: 10px;
    vertical-align: middle;
}
```

Efectos en botones

.btn-primary: Define el estilo del botón con la clase **btn-primary**.

background-color: #ff5722;: Establece un color de fondo naranja (#ff5722) para el botón.

border: none;: Elimina el borde del botón, dándole un aspecto más limpio.

.btn-primary:hover: Se aplica cuando el mouse pasa sobre el botón.

background-color: #e64a19;: Cambia el color de fondo a un tono más oscuro de naranja (#e64a19) al pasar el mouse, proporcionando un efecto visual interactivo.

Sección de Contacto

form .form-control: Selecciona todos los elementos de entrada dentro de un formulario que tienen la clase **form-control**.

border-radius: 5px;: Aplica bordes redondeados de 5 píxeles a los campos del formulario, suavizando su apariencia.

Estilo del Pie de Página

footer: Aplica estilos al pie de página de la página web.

background-color: #111;: Establece un color de fondo oscuro (#111) para el pie de página, mejorando la legibilidad del texto blanco que probablemente esté en él.

Estilo del Menú Desplegable

.navbar .dropdown-menu: Selecciona el menú desplegable dentro de la barra de navegación.

background-color: #333;: Define un fondo oscuro (#333) para el menú desplegable.

border: none;: Elimina el borde del menú desplegable.

.navbar .dropdown-item: Aplica estilos a los elementos del menú desplegable.

color: #fff;: Establece el color del texto en blanco para los elementos del menú, asegurando que sean legibles sobre el fondo oscuro.

.navbar .dropdown-item:hover: Se aplica cuando el mouse pasa sobre un elemento del menú desplegable.

background-color: #444;: Cambia el fondo a un tono más claro (#444) al pasar el mouse, destacando la opción seleccionada.

color: #ff1cf7;: Cambia el color del texto a un color rosa brillante (#ff1cf7) al pasar el mouse, lo que añade un efecto visual atractivo.

Estilo de Iconos Sociales

.social-icon: Aplica estilos a los iconos sociales (como los de Facebook, Instagram, etc.).

margin-right: 10px;: Agrega un margen derecho de 10 píxeles, separando los iconos unos de otros.

vertical-align: middle;: Alinea verticalmente los iconos en el medio de la línea, asegurando que se vean bien en relación con el texto que los acompaña.

Estructura de JavaScript:

```
// Smooth scroll para navegación

document.querySelectorAll('a[href^="#"]').forEach(anchor => {
  anchor.addEventListener('click', function(e) {
    e.preventDefault();
    document.querySelector(this.getAttribute('href')).scrollIntoView({
      behavior: 'smooth'
    });
  });
});

// Efecto de aparición al hacer scroll

window.addEventListener('scroll', function() {
  const elements = document.querySelectorAll('.card');
  elements.forEach(element => {
    const position = element.getBoundingClientRect().top;
    const screenHeight = window.innerHeight;
    if (position < screenHeight) {
      element.style.opacity = '1';
      element.style.transform = 'translateY(0)';
    } else {
      element.style.opacity = '0';
      element.style.transform = 'translateY(50px)';
    }
  });
});

//Activación del menú desplegable en hover

$(document).ready(function () {
  $('.dropdown').hover(
    function () {
      $(this).find('.dropdown-menu').stop(true, true).delay(100).fadeIn(200);
    },
    function () {
      $(this).find('.dropdown-menu').stop(true, true).delay(100).fadeOut(200);
    }
  );
});
```

Smooth Scroll para Navegación

document.querySelectorAll('a[href^="#"]'): Selecciona todos los enlaces () que apuntan a un identificador en la misma página (es decir, aquellos cuyo href comienza con #).

`.forEach(anchor => {...})`: Itera sobre cada enlace encontrado.

`anchor.addEventListener('click', function(e) {...})`: Añade un evento de clic a cada enlace.

`e.preventDefault()`: Previene la acción predeterminada del enlace (que sería saltar a la sección de destino sin animación).

`document.querySelector(this.getAttribute('href')).scrollIntoView({ behavior: 'smooth' })`: Obtiene el elemento al que apunta el enlace usando su `href` y lo desplaza suavemente a la vista.

Efecto de Aparición al Hacer Scroll

`window.addEventListener('scroll', function() {...})`: Escucha el evento de desplazamiento de la ventana.

`const elements = document.querySelectorAll('.card');`: Selecciona todos los elementos con la clase `card`.

`elements.forEach(element => {...})`: Itera sobre cada elemento `card`.

`const position = element.getBoundingClientRect().top;`: Obtiene la posición superior del elemento con respecto a la ventana.

`const screenHeight = window.innerHeight;`: Obtiene la altura de la ventana del navegador.

`if (position < screenHeight) {...}`: Verifica si el elemento está visible en la ventana.

Si es visible, se establece la opacidad a 1 y se restablece su posición (`transform: translateY(0)`).

Si no es visible, se establece la opacidad a 0 y se mueve el elemento 50 píxeles hacia abajo (`transform: translateY(50px)`).

Activación del Menú Desplegable en Hover

`$(document).ready(function () {...})`: Asegura que el código se ejecute solo después de que el DOM se haya cargado completamente.

`$('.dropdown').hover(...);`: Aplica un efecto hover a los elementos con la clase `dropdown`.

`function () {...}`: Función que se ejecuta al entrar el mouse:

```
$(this).find('.dropdown-menu').stop(true,  
true).delay(100).fadeIn(200);
```

: Muestra el menú desplegable con un retraso y una animación de desvanecimiento.

```
function () {...}
```

: Función que se ejecuta al salir el mouse:

```
$(this).find('.dropdown-menu').stop(true,  
true).delay(100).fadeOut(200);
```

: Oculta el menú desplegable de manera similar.

Función para Establecer una Cookie

```
function setCookie(name, value, days) {...}
```

: Define una función para establecer una cookie.

```
const d = new Date();
```

: Crea un nuevo objeto de fecha

```
d.setTime(d.getTime() + (days*24*60*60*1000));
```

: Calcula la fecha de expiración de la cookie en milisegundos, sumando el número de días especificado.

```
const expires = "expires="+ d.toUTCString();
```

: Convierte la fecha de expiración a un formato que se puede usar en la cookie.

```
document.cookie = name + "=" + value + ";" + expires + ";path=/";
```

: Establece la cookie en el documento, asignándole el nombre, valor, fecha de expiración y el camino en el que es válida

Segunda imagen de la estructura de JavaScript:

```
function getCookie(name) {  
    const nameEQ = name + "=";  
    const decodedCookie = decodeURIComponent(document.cookie);  
    const ca = decodedCookie.split(';');  
    for(let i = 0; i < ca.length; i++) {  
        let c = ca[i];  
        while (c.charAt(0) === ' ') {  
            c = c.substring(1);  
        }  
        if (c.indexOf(nameEQ) === 0) {  
            return c.substring(nameEQ.length, c.length);  
        }  
    }  
    return "";  
}  
  
//Guardar datos del formulario en cookies al enviar  
  
document.getElementById('contacto-form').addEventListener('submit', function(event) {  
    event.preventDefault();  
  
    const nombre = document.getElementById('nombre').value;  
    const email = document.getElementById('email').value;  
    const mensaje = document.getElementById('mensaje').value;  
  
    // Guardar en cookies por 7 días  
    setCookie('nombre', nombre, 7);  
    setCookie('email', email, 7);  
    setCookie('mensaje', mensaje, 7);  
  
    alert('Mensaje enviado y datos guardados en cookies!');  
});  
  
//Cargar los datos de cookies al cargar la pagina  
  
window.onload = function() {  
    const nombre = getCookie('nombre');  
    const email = getCookie('email');  
    const mensaje = getCookie('mensaje');
```

Función para Obtener una Cookie

function getCookie(name): Define una función que toma como argumento el nombre de una cookie y devuelve su valor.

const nameEQ = name + "=";: Crea una cadena que representa el nombre de la cookie seguido de un signo igual (=).

const decodedCookie = decodeURIComponent(document.cookie);: Decodifica la cadena de cookies almacenadas en **document.cookie**.

```
const ca = decodedCookie.split('');: Divide la cadena de cookies en un array usando ; como delimitador.
```

```
for (let i = 0; i < ca.length; i++) {...}: Itera sobre cada cookie en el array ca.
```

```
while (c.charAt(0) === ' ') {...}: Elimina los espacios en blanco al principio de cada cookie.
```

```
if (c.indexOf(nameEQ) === 0) {...}: Comprueba si la cookie actual comienza con el nombre de la cookie que estamos buscando.
```

Si la encuentra, devuelve el valor de la cookie extrayendo el contenido después del signo igual.

```
return "";: Si no encuentra la cookie, devuelve una cadena vacía.
```

Guardar Datos del Formulario en Cookies al Enviar

```
document.getElementById('contacto-form').addEventListener('submit', function(event) {...}): Añade un evento al formulario de contacto que se activa al enviarlo.
```

```
event.preventDefault(): Previene el envío del formulario, lo que permite manejarlo manualmente.
```

```
const nombre = document.getElementById('nombre').value;: Obtiene el valor del campo de nombre del formulario.
```

```
const email = document.getElementById('email').value;: Obtiene el valor del campo de email.
```

```
const mensaje = document.getElementById('mensaje').value;: Obtiene el valor del campo de mensaje.
```

```
setCookie('nombre', nombre, 7);: Llama a la función setCookie para guardar el nombre en una cookie con un tiempo de expiración de 7 días.
```

```
alert('Mensaje enviado y datos guardados en cookies!');: Muestra una alerta confirmando que el mensaje fue enviado y que los datos fueron guardados.
```

Cargar los Datos de Cookies al Cargar la Página

```
window.onload = function() {...}: Define una función que se ejecuta cuando la ventana se carga completamente.
```

```
const nombre = getCookie('nombre');: Llama a la función getCookie para obtener el valor de la cookie del nombre.
```

`const email = getCookie('email');`: Llama a la función `getCookie` para obtener el valor de la cookie del email.

`const mensaje = getCookie('mensaje');`: Llama a la función `getCookie` para obtener el valor de la cookie del mensaje.

`if (nombre) {...}`: Comprueba si existe el valor de la cookie del nombre.

Si existe, establece el valor del campo de nombre en el formulario.

`if (email) {...}`: Realiza la misma comprobación y acción para el campo de email.

`if (mensaje) {...}`: Realiza la misma comprobación y acción para el campo de mensaje.

Este código permite guardar los datos del formulario de contacto en cookies cuando se envía el formulario, y al cargar la página, verifica si hay datos guardados en las cookies y los muestra en el formulario. Esto mejora la experiencia del usuario al facilitar el reingreso de información que ya se había ingresado anteriormente.