

Aspectos de desarrollo del trabajo practico:

El TP se ha basado en distintos archivos:

- HTML entrelazos con sus respectivos js
- Un archivo index principal la cual tiene la página principal del TP
- Un archivo css con sus estilos referenciados a la página principal(index)
- Un archivo js principal que contiene distintas funciones
- Un archivo HTML para el registro y el ingreso de datos a la página junto con un formulario de contacto validado con un js
- una API que menciona el clima y el lugar actual

-Se pasará a mencionar la estructura y componentes de cada uno basado en secciones:

HTML

-Index.html:

1.	Encabezado:
	<ul style="list-style-type: none">• Incluye botones de navegación para "Ofertas", "Descuentos" y "Combos".• Enlaces a las páginas "Ingresar" y "Contacto".• Un campo de búsqueda con un botón de "Carrito".
2.	Carrousel (Sección "carousel"):
	<ul style="list-style-type: none">• Un área que muestra un carrusel de productos. Los productos se muestran en imágenes, pero el código HTML para las imágenes reales no se proporciona aquí.
3.	Secciones de productos:
	<ul style="list-style-type: none">• "ofertas", "descuentos", "combos": Estas secciones tienen productos con ofertas, productos con descuentos y combos, respectivamente. Cada sección está oculta inicialmente (se establece en display: none).
4.	Próximos productos a traer:
	<ul style="list-style-type: none">• Muestra un título y un carrusel con imágenes de los productos próximos a ser añadidos al sitio.
5.	Pie de página (Sección "pie-pagina"):
	<ul style="list-style-type: none">• Contiene información sobre la empresa "TiendaSnacks".• Enlaces a redes sociales.• Muestra información meteorológica en tiempo real (temperatura, descripción del clima, ubicación y un icono de clima). Esta información parece provenir de un script llamado "apiclimate.js".
6.	Scripts:
	<ul style="list-style-type: none">• Se incluyen varios archivos JavaScript para la funcionalidad de la página: "apiclimate.js", "main.js" y "carousel.js". Estos archivos se encargan de la interactividad y funcionalidad de la página.

-Contacto.html:

Este código HTML crea una página web para un formulario de contacto. El formulario permite a los usuarios ingresar su información de contacto, incluyendo nombre completo, correo electrónico, número telefónico, motivo de contacto y un mensaje. También muestra la ubicación de la empresa en un mapa de Google.

Estructura y componentes:

1. Encabezado:

- No contiene elementos visibles en el encabezado.

2. Artículo (Sección principal):

- Muestra el título "Formulario De Contacto".
- Contiene un formulario de contacto con los siguientes campos:
 - Nombre completo
 - Correo electrónico
 - Número telefónico
 - Motivo de contacto
 - Mensaje
- Un botón para enviar el formulario.
- Muestra la ubicación de la empresa en un mapa de Google.

3. Pie de página:

- Contiene información sobre la empresa "**TiendaSnacks**".
- Enlaces a redes sociales.
- Muestra información meteorológica en tiempo real (temperatura, descripción del clima, ubicación y un icono de clima). Esta información parece provenir de un script llamado "apiclimate.js".

4. Scripts:

- Se incluyen dos archivos JavaScript: "apiclimate.js" y "validar.js". Estos archivos se encargan de la funcionalidad de la página, incluyendo la validación del formulario.

-Ingresar.html:

Resumen: Este código HTML crea una página web para que los usuarios inicien sesión en el sitio web. Los usuarios deben ingresar su dirección de correo electrónico o número de teléfono junto con su contraseña para acceder. También incluye enlaces para recuperar la contraseña o registrarse como nuevo miembro.

Estructura y componentes:

1. Encabezado:

- Contiene un enlace de regreso a la página de inicio ("Home").

2. Contenido principal:

- Muestra el título "Bienvenido".
- Contiene un formulario de inicio de sesión con los siguientes campos:
 - Campo de "Email o Teléfono" para ingresar la dirección de correo electrónico o número de teléfono.

- Campo de "Contraseña" para ingresar la contraseña.
- Un enlace para recuperar la contraseña (¿"Olvidaste La Contraseña?").
- Un botón para iniciar sesión.
- Un enlace para registrarse como nuevo miembro ("No Eres Un Miembro? Regístrate Gratis").

3. Pie de página:

- Contiene información sobre la empresa "TiendaSnacks".
- Enlaces a redes sociales.
- Muestra información meteorológica en tiempo real (temperatura, descripción del clima, ubicación y un icono de clima). Esta información parece provenir de un script llamado "apiclimate.js".

4. Scripts:

- Se incluye un archivo JavaScript llamado "apiclimate.js" que se encarga de proporcionar información meteorológica en tiempo real.

-Registro.html:

Este código HTML crea una página web que permite a los usuarios registrarse de forma gratuita en el sitio web. Los usuarios deben proporcionar su nombre, apellido, dirección de correo electrónico y contraseña para crear una cuenta. También incluye un enlace para iniciar sesión si ya tienen una cuenta.

Estructura y componentes:

1. Encabezado:

- Contiene un enlace de regreso a la página de inicio ("Home").

2. Contenido principal:

- Muestra el título "Bienvenido".
- Contiene un formulario de registro con los siguientes campos:
 - Campo para ingresar el nombre.
 - Campo para ingresar el apellido.
 - Campo para ingresar el correo electrónico.
 - Campo para ingresar la contraseña.
 - Un botón para registrar la cuenta.
 - Un enlace que redirige a la página de inicio de sesión si el usuario ya tiene una cuenta ("¿Ya tienes cuenta?").

3. Pie de página:

- Contiene información sobre la empresa "TiendaSnacks".
- Enlaces a redes sociales.
- Muestra información meteorológica en tiempo real (temperatura, descripción del clima, ubicación y un icono de clima). Esta información parece provenir de un script llamado "apiclimate.js".

4. Scripts:

- Se incluye un archivo JavaScript llamado "apiclimate.js" que se encarga de proporcionar información meteorológica en tiempo real.

En cuanto a los archivos con extensión css----- carousel, main,contacto, ingresar, validación y registro son los encargados de darle estilos a sus páginas HTML referenciadas

los archivos JavaScript(js):

-main.js

Este código JavaScript define un conjunto de productos y luego implementa funcionalidades para mostrar estos productos en una página web y permitir a los usuarios agregar productos a un carrito de compras. A continuación, se presenta un resumen de las características clave del código:

Resumen:

- Se define un array llamado **productos**, que contiene una lista de productos con propiedades como nombre, precio, imagen, ofertas, descuento y combos.
- La página web muestra estos productos y permite al usuario filtrar productos por ofertas, descuentos o combos, y también permite buscar productos por nombre.
- Los usuarios pueden agregar productos al carrito de compras y ver el carrito.

Características del código:

- Se utilizan eventos DOMContentLoaded para asegurarse de que el código se ejecute después de que se haya cargado la página.
- Se definen funciones para mostrar productos y actualizar la visualización en función de los filtros y la búsqueda.
- Se agregan manejadores de eventos a los botones que permiten filtrar productos por ofertas, descuentos o combos.
- Se agrega un manejador de eventos al botón "Carrito" para mostrar una alerta con los productos en el carrito y el total a pagar.
- Se agrega un manejador de eventos para el campo de filtro, que permite buscar productos por nombre y marcar los productos como ofertas si coinciden con el filtro.
- Se permite a los usuarios agregar productos al carrito haciendo clic en el botón "Añadir al Carrito" en cada producto.

-Apiclima.js

Este código JavaScript permite obtener las coordenadas geográficas del dispositivo del usuario a través del navegador y luego utiliza esas coordenadas para obtener datos del clima de la ubicación actual.

Resumen:

- Cuando se carga la página, el código verifica si el navegador del usuario admite la obtención de coordenadas geográficas a través de la API **navigator.geolocation**.
- Si es compatible, el código obtiene la ubicación actual del usuario y utiliza las coordenadas (latitud y longitud) para realizar una solicitud a la API de OpenWeatherMap para obtener datos de clima en esa ubicación.

- Luego, muestra la temperatura actual, la descripción del clima, la ubicación y un ícono de clima en la página web.

Características clave:

- Utiliza el evento **load** para asegurarse de que el código se ejecute después de que la página se haya cargado completamente.
- Comprueba si el navegador admite la geolocalización (**navigator.geolocation**).
- Si se permite la geolocalización, se obtienen las coordenadas (latitud y longitud) del dispositivo del usuario.
- Se construye una URL para realizar una solicitud a la API de OpenWeatherMap, pasando las coordenadas y una clave de API.
- Se utiliza la función **fetch** para realizar la solicitud a la API de OpenWeatherMap y procesar la respuesta.
- La información del clima se muestra en elementos HTML en la página, incluyendo la temperatura, la descripción del clima, la ubicación y un ícono de clima.

-Carousel.js

Este código crea un carrusel 3D interactivo con imágenes y videos. El carrusel es giratorio y tiene características como reproducción automática, fondo musical y control de música, y la capacidad de cambiar el tamaño de las imágenes y videos.

Algunas de las características clave del código son:

1. El carrusel se compone de una serie de imágenes y videos que se organizan en un círculo 3D.
2. Los elementos del carrusel se distribuyen uniformemente alrededor del círculo.
3. El carrusel puede girar automáticamente o pausar la rotación según la configuración.
4. Los usuarios pueden interactuar con el carrusel girándolo manualmente haciendo clic y arrastrando.
5. El fondo musical se reproduce en segundo plano, y los controles de música (reproducir/pausar) son opcionales.
6. Las imágenes y videos se cargan dinámicamente desde una lista de productos o elementos, lo que permite personalizar fácilmente el contenido.
7. El código incluye opciones para configurar el radio, la velocidad de rotación, el tamaño de las imágenes, la URL de la música de fondo y los controles de música.
8. Las imágenes y videos pueden incluir enlaces, lo que permite a los usuarios hacer clic en los elementos para obtener más información.

-Validar.js

se utiliza para la validación de formularios. A continuación, se proporciona una descripción detallada de las partes principales del código:

Constantes y Variables Globales:

- Se declaran las constantes y variables globales utilizadas en el código. Por ejemplo, se definen referencias a elementos del formulario HTML, como **nombre**, **email**, **teléfono**, **motivo**, **mensaje**, y **formulario**.

Evento de Envío del Formulario:

- El código agrega un manejador de evento al formulario (**formulario**) que se dispara cuando se intenta enviar el formulario. El evento se maneja a través de la función **validar Campos**.

Función validar Campos():

- Esta función se ejecuta cuando se intenta enviar el formulario y se encarga de validar los campos del formulario.
- Se definen expresiones regulares (**patternEmail** y **patternTelefono**) para validar el formato del correo electrónico y el número de teléfono.
- La función verifica varios aspectos de los campos del formulario:
 - Comprueba si los campos requeridos están vacíos y, en caso afirmativo, agrega un mensaje de error al **mensajeError** y establece **resulValidacion** en **false**.
 - Verifica si las longitudes de los campos (**nombre**, **motivo**, **mensaje**) están dentro de los rangos permitidos y, en caso contrario, agrega un mensaje de error al **mensajeError** y establece **resulValidacion** en **false**.
 - Valida el formato del correo electrónico y el número de teléfono utilizando las expresiones regulares. Si no son válidos, se agrega un mensaje de error al **mensajeError** y se establece **resulValidacion** en **false**.

Funciones campoRequerido() y longitudCampo():

- Estas funciones son llamadas por **validarCampos()** para verificar si un campo está vacío o si su longitud está fuera del rango permitido, respectivamente.
- Devuelven un mensaje de error si la validación falla.

Fuentes y Estilos:

- En los archivos HTML, se incluyen fuentes de Google Fonts para estilizar el texto. Algunas de las fuentes utilizadas son "Montserrat", "Bebas Neue", "San-Seriff", "Quicksand", "Poppins" etc"

Librerías y Enlaces Externos:

- En los archivos HTML, se enlazan varios archivos externos, que pueden incluir fuentes, íconos y hojas de estilo adicionales.
- La librería Font Awesome se utiliza para incluir íconos en los elementos HTML a través del enlace **<script>**.

Para la API del clima se usó la provista por: <https://openweathermap.org/api>

Para el carrusel se usó la librería:

Author: Hoang Tran (<https://www.facebook.com/profile.php?id=100004848287494>)

Github versión (1 file .html):

<https://github.com/HoangTran0410/3DCarousel/blob/master/index.htm>

Para validación de formulario se usó código JavaScript plano, con expresiones regulares.

Carpetas IMG: poseen las imágenes de los productos, imágenes del slider (próximos productos a traer), imágenes de fondos y demás.