

ACTIVIDAD PROYECTO: CREACIÓN DE PÁGINAS WEB. HTML5+CSS.

Hecho por: Jorge González Fuentes



Índice

1. Introducción.....	2
2. Header.	2
3. Footer.	4
4. Elementos comunes.	6
5. Elementos no comunes.....	15
6. Conclusiones	20

1. Introducción.

En este PDF vamos a hablar sobre los elementos HTML usados para la estructura de nuestra página web, pasando por el diseño de estas con los archivos CSS así como de las decisiones tomadas para elegir que elementos usar en nuestra página.

Toda la estructura se va a repetir (como es lógico) en nuestra página, siendo esta la siguiente:

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
  <header>
  </header>
  <main>
  </main>
  <footer>
  </footer>
</body>
</html>
```

2. Header.

Para el header hemos usado la siguiente estructura HTML:

```
<header>
  
  <nav>
    <a href="/index.html" class="activo">Inicio</a>
    <a href="/public/sobre_nosotros.html">Sobre nosotros</a>
    <a href="/public/galeria.html">Galería</a>
    <a href="/public/eventos.html">Eventos</a>
    <a href="/public/reservar.html">Reserva tu cita</a>
  </nav>
</header>
```

Como podemos ver el header se compone por las etiquetas “img”, “nav” y “a”, donde:

- “img” va a contener el logo de la página, que más adelante comentaremos donde se ubica.
- “nav” va a contener el menú de navegación.
- “a” que son los enlaces que nos van a llevar a las páginas, más adelante veremos cómo se colocan.

Con el respectivo CSS donde:

```
header {
  background: #000000;
  position: absolute;
  width: 100%;
  height: 75px;
  left: 0px;
  top: 0px;
}
```

```
header nav {  
  position: absolute;  
  top: 0;  
  right: 0;  
  display: flex;  
  justify-content: center;  
  align-items: center;  
}
```

```
header nav a {  
  font-size: 20px;  
  font-weight: 400;  
  text-decoration: none;  
  width: 184px;  
  height: 75px;  
  color: #efb810;  
  display: flex;  
  align-items: center;  
  justify-content: center;  
  transition: 0.5s;  
}
```

```
header nav a: hover {  
  width: 184px;  
  height: 75px;  
  color: #000000;  
  background-color: #efb810;  
}
```

```
.logoP {  
  width: 75px;  
  height: 75px;  
  margin-left: 25px;  
  padding: 0;  
}
```

```
.activo {  
  background-color: #efb810;  
  color: #000000;  
  height: 65px;  
  padding-top: 5px;  
  border-bottom: 5px solid #000000;  
}
```

```
.activo: hover {  
  height: 65px;  
}
```

En su gran mayoría hemos usado selectores de etiquetas, puesto a que no hay mucho donde distinguir, es decir, todas las etiquetas en el header se van a ver afectadas por ello, y exactamente eso es lo que buscamos, a excepción de algunas que usan las clases creadas.

Primero vamos a hablar de los selectores de etiquetas, donde podemos ver que la etiqueta “header” es la que se encarga de la posición del contenedor, al igual que se encarga del color de fondo y del tamaño del susodicho.

Lo sigue la etiqueta “nav” donde le repetimos la posición que va a tener dentro del contenedor “header”, también le indicamos su posición, la disposición y posición de los elementos dentro de esta etiqueta.

Luego del “nav” pasamos por las etiquetas “a” incluidas dentro de este contenedor. A estas le indicamos el color, el tamaño del contenedor, el tamaño de la fuente y su color, todo esto junto la disposición de dichos elementos dentro del menú de navegación. Para esta etiqueta le hemos agregado también un “hover” que tendrá una transición de medio segundo, donde se cambiará el color de la fuente y del contenedor.

Pasamos ahora por las clases, que nos darán propiedades más exclusivas a ciertos elementos.

Empezamos por la clase “.logoP” que se encarga de la posición del logotipo que le hemos añadido al header.

Luego seguimos con la clase de “.activo” que nos marca en que página nos encontramos actualmente. Esto lo hace cambiándole el color de forma definitiva a la etiqueta que contenga la clase. En esta clase hay un truco, puestos que el “hover” del elemento “a” le seguía afectando, produciendo un efecto donde el contenedor se estiraba, para solucionar esto simplemente hemos hecho que el hover reduzca la altura del contenedor, haciendo que el contenedor parezca estático cuando pasamos el cursor por encima.

El header se mantiene casi igual para todas las paginas, donde la única variación que hay es otorgarle la clase “.activo” al elemento que queremos que destaque para señalar en que página nos encontramos en el momento.

3. Footer.

Para el “footer” hemos seguido casi los mismos pasos que en el header, es decir, hemos optado por selectores de etiquetas que usamos en su gran mayoría, y hacemos uso de los selectores de clase para cosas específicas.

El código HTML del “footer” es el siguiente:

```
<footer>
  <ul class="contactanos">
    <li class="tituloPDP">Contactanos:</li>
    <li><a
      href="https://api.whatsapp.com/send/?phone=34651687717&text=Gracias+por+contactarnos%2C+para+agendar+tu+cita%2C+necesitamos+saber%3B+que+te+quieres+hacer%2C+tama%C3%B1o+en+cent%C3%ADmetros+y+la+zona+donde+lo+quieres.+As%C3%AD+como+imagen+de+referencia+en+caso+de+tenerla.+Gracias%21&type=phone_number&app_absent=0">+34 651 68 77 17</a></li>
    <li><a href="https://maps.app.goo.gl/dGJe1DkXmTmanEBw6">Avda. Cangas 79, Lebrija, 41740, Sevilla</a></li>
  </ul>
  <ul class="horario">
    <li class="tituloPDP">Horario:</li>
    <li>Lunes a Viernes: 11:30-13:30 / 18:00-21:00</li>
    <li>Sábado: 11:30-13:30</li>
    <li>Domingo: Cerrado</li>
  </ul>
```

```
</ul>
<ul class="siguenos">
  <li class="tituloPDP">Siguenos:</li>
    <li><a href="https://www.instagram.com/arttattoo_pmg_">Instagram:
      @arttattoo_pmg_</a></li>
    <li><a href="https://www.tiktok.com/@patrimglez00">TikTok: @patrimglez00</a></li>
  </ul>
</footer>
```

Aquí podemos ver que usamos tres listas no numeradas, donde cada una nos va mostrar información de interés sobre la página. Empezando por la lista que contiene los enlaces para contactar con la empresa, donde la sigue la lista que muestra el horario y por último la lista que contiene los enlaces de las redes sociales.

Pasamos a ver el código CSS del pie de página:

```
footer {
  background: #000000;
  position: absolute;
  width: 100%;
  display: inline-flex;
  height: 75px;
  left: 0;
  bottom: 0;
  color: #efb810;
}
```

```
.tituloPDP{
  font-size: 15px;
  font-weight: 100;
  text-align: left;
}
```

```
footer li{
  font-size: 10px;
  font-weight: 400;
  display: block;
  text-align: left;
  line-height: 1.2;
}
```

```
footer li a{
  color: #efb810;
  cursor: pointer;
  text-decoration: none;
  transition: 0.5s;
}
```

```
footer li a:hover{
  color: #000000;
}
```

```
background-color: #efb810;
text-align: center;
align-items: flex-start;
align-content: flex-start;
text-decoration: none;
}

.contactanos{
margin-left: 30%;
}

.horario{
margin-left: 5%;
}

.siguenos{
margin-left: 5%;
}
```

El selector “footer” al igual que nos pasaba con el selector header, se encarga de señalar la posición del “footer”, su tamaño, el color, y lo más importante, la disposición de los contenedores dentro de la etiqueta “footer”.

Seguimos con el selector “footer li”, que modifica los elementos de la lista para que aparezcan en bloque, también modifica el tamaño del texto contenido en él, su peso, disposición y separación entre líneas.

Pasamos por el selector “footer li a”, que es el encargado de dar color a los enlaces, también modifica el cursor para cuando pases por encima te señale que es un enlace. Este selector contiene una transición de medio segundo, ya que le hemos aplicado una pseudoclase “hover” para que resalte el contenido del enlace, cambiándole el color.

Ahora vamos a ver las clases usadas para el footer, donde “tituloPDP” nos sirve para los títulos de las listas, haciendo que el tamaño de las letras sea distinto a las etiquetas “li” que contienen la información del “footer”. Las siguientes tres clases son muy similares, ya que cada una se encarga de la posición de la lista que contiene dicha clase, haciendo que estas queden centradas en el pie de página.

4. Elementos comunes.

Como ya hemos visto con el header y el footer, hay ciertos elementos que hemos reusado para la creación de nuestras páginas, aunque en este caso no es solo el elemento, sino hablamos de páginas completas donde los cambios son el nombre del archivo, el CSS que usa (más adelante explicaremos por qué), las imágenes que contienen y/o los textos que se muestran.

Estas páginas son las que mostraremos a continuación, de ellas vamos a explicar las similitudes que tienen, así como de sus diferencias.

Empezamos por las páginas más repetidas o que más similitudes comparten entre sí, estas son las páginas que se encargan de darnos información detallada sobre el sitio web o sobre la empresa. Hablamos de “index.html”, “artista.html”, “estudio.html”, “trabajos.html”, “eventoX.html” (siendo X: 1, 2, 3 y 4 respectivamente).

Estas páginas guardan la similitud de dividirse en dos secciones, siendo la primera sección una que ocupa un tercio de la página, que es la encargada de mostrar una foto o el logo de la empresa. La otra sección que ocupa el resto de la página, es la más “importante” puesto que es la que va a contener la información que queremos que el cliente lea, esta va a estar contenida en un cuadro de texto que ocupa cierta parte de la sección, digo cierta parte porque entre paginas cambia el tamaño de dicho cuadro. También la mayoría

de estas páginas contienen un botón que te devuelve al punto donde has podido acceder a la página que contiene lo que hemos mencionado.

Esas son las similitudes más notorias entre páginas, ahora pasamos a ver las diferencias, las cuales en gran parte son cambios en el texto o imágenes, pero como hemos dicho antes, también cambian el tamaño de ciertos elementos como lo pueden ser el cuadro de texto.

Vamos a ver un ejemplo del código del body de la página “artista.html”:

```
<main>
  <div class="estudio">
    
  </div>
  <section class="informacion">
    <div class="infoArtista">
      <h1>La Artista.</h1>
      <p>Conoce a Patri, nuestra tatuadora.<br>
      <br>
      Patri es una joven artista que desde pequeña ha estado presente en el mundo del arte, empezó
      su negocio en este mundillo con retratos y dibujos varios, hasta que un día encontró el mundo del tatuaje
      y lleva desde entonces “x” años usando la piel como lienzo para sus obras.<br>
      <br>
      Comenzó tatuando a domicilio, llevando a cuestas todo su equipamiento para poder realizar sus
      obras y dar un paso mas hacía su sueño. <br>
      Más adelante consiguió asentarse en casa de un familiar para poder realizar de forma mas
      cómoda sus obras, hasta que con esfuerzo y perseverancia, ha conseguido su propio local desde el cual,
      orgullosa de ello, realiza con pasión y dedicación los tatuajes que le encargan. <br>
      <br>
      Patri se especializa en tatuajes de línea fina, microrealismo y puntillismo, pero esto no la detiene
      en realizar todo tipo de tatuajes. <br>
      <br>
      Su formación dentro de este mundillo es la siguiente:<br>
      </p>
      <ul>
        <li>
          Título homologado en maquillaje integral
        </li>
        <li>
          Titulo homologado de tatuajes.
        </li>
        <li>
          Título de piercings proporcionado por la escuela Yanni Piercings
        </li>
      </ul>
      <blockquote>“Todo tiene un comienzo, que esto no es un don ni es fácil, todo es
      aprender, empeñarse y poner esfuerzo en lo que realmente te gusta.”.-
      Patri.</blockquote>
    </div>
    <a href="sobre_nosotros.html" class="boton">Volver</a>
  </section>
```



```
</main>
```

Como hemos dicho, la página se divide en dos secciones, una que va a contener la imagen y otra que va a contener la información, la primera sección es una etiqueta “div” que contiene la imagen (no hemos hecho uso de la etiqueta section porque para usar dicha etiqueta necesitas como mínimo contener un título “h2-h6” para que pase la validación).

La segunda sección si contiene una etiqueta “section” ya que aquí si hacemos uso de un título.

Dentro de la etiqueta “section” tenemos la información que queremos mostrar que va contenida dentro de las etiquetas “p” las cuales por motivos estéticos contienen etiquetas de salto de línea “br”. También hemos metido una lista no numerada para hacer mención, en este caso, de los títulos que posee la trabajadora de la empresa, así como hemos usado una etiqueta “blockquote” ya que queríamos introducir una frase dicha por nuestra profesional.

Ahora pasamos a ver el código CSS de la página, que en las ya mencionadas es muy similar unas con otras, donde los únicos cambios son de tamaño de los contenedores, también cambia el nombre de las clases, para así saber con qué archivo CSS estamos trabajando puestos que cada página contiene su propio archivo CSS, ya que entre paginas como ya hemos dicho, cambian ciertas propiedades, pero la estructura es muy similar. En el código hemos obviado las partes del header y el footer, ya que no tiene sentido mostrarlas en cada código que enseñemos, ya que lo hemos enseñado en sus respectivos apartados.

```
body {  
  background-image: url('../Fotos-Proyecto/Logo-Fondo/fondo.jpeg');  
  background-size: cover;  
  background-repeat: no-repeat;  
  background-attachment: fixed;  
  overflow: hidden;  
}
```

```
img {  
  width: 900px;  
  max-width: 100%;  
  max-height: 100%;  
  height: 100%;  
  object-fit: cover;  
}
```

```
.estudio {  
  position: absolute;  
  height: 800px;  
  top: 75px;  
  left: 0%;  
}
```

```
.informacion {  
  position: absolute;  
  left: 51%;  
  right: 2%;  
  top: 15%;  
  bottom: 13%;  
  background: rgba(255,255,255,0.5);  
}
```

```
}

.infoArtista {
    text-align: justify;
    margin-left: 35px;
    margin-right: 20px;
    align-items: center;
    color: #000000;
}

.infoArtista h1{
    font-family: 'Josefin Slab';
    margin-top: 5%;
    font-size: 35px;
}

.infoArtista p{
    font-family: 'Josefin Slab';
    font-size: 20px;
}

.infoArtista blockquote{
    font-family: 'Josefin Slab';
    font-size: 20px;
    font-style: italic;
}

.infoArtista li{
    font-family: 'Josefin Slab';
    font-size: 20px;
}

.boton {
    position: absolute;
    height: 30px;
    width: 125px;
    margin-left: 75%;
    background: #000000;
    border-radius: 12px;
    border: none;
    text-decoration: none;
    text-align: center;
    font-size: 12px;
    color: #efb810;
    line-height: 30px;
    transition: 0.5s;
}

.boton:hover {
```

```

color: #000000;
background: #efb810;
}

```

Como podemos ver, en este caso se hace mayor uso de los selectores de clase, que de los selectores de etiqueta.

Empezamos por el selector de etiqueta “body” que se encarga del fondo de la página, que va a cubrir todo el tamaño de la ventana, no se repite, es fijo y a la ventana, le hemos quitado las barras de desplazamiento, esto es debido a que no hay contenido más allá del que ocupa la ventana, también se debe a un error con el CSS del pie de página, donde no se quedaba pegado al fondo de la página, sino que estaba un poco desplazado hacia arriba, aunque ese error se corrigió ya, me gusta mantener el atributo de “overflow: hidden” para evitar que se dé de nuevo el error, así como que no muestre la barra de desplazamiento.

Para las clases, tenemos las que se encargan de la posición y el tamaño de las secciones, como lo son el selector de etiqueta “img” o las clases “.informacion”, “.infoArtista” o “.estudio”, también están los tres selectores descendientes de la clase “.infoArtista” que se encargan del tamaño de la letra, del tipo de fuente y del peso de las mismas, y por último el selector de clase “.boton” y su pseudoclase “hover”, que se encarga de la posición, tamaño y color del botón.

Ahora vamos a ver las páginas de “sobre_nosotros.html” y “galería.html”. En dichas paginas los elementos comunes es la división de la página en tres secciones, que veremos en detalle más adelante.

Las diferencias entre las paginas es el cambio de imagen y el título. A continuación, veremos el código HTML de la página “sobre_nosotros.html” de ejemplo:

```

<main>
  <section class="estudio">
    <a href="estudio.html">
      
      <div>
        <h4>El estudio</h4>
      </div>
    </a>
  </section>
  <section class="artista">
    <a href="artista.html">
      
      <div>
        <h4>La artista</h4>
      </div>
    </a>
  </section>
  <section class="trabajos">
    <a href="trabajos.html">
      
      <div>
        <h4>Nuestros trabajos</h4>
      </div>
    </a>
  </section>

```

```
</main>
```

Como podemos ver ahora en el código, vemos las tres secciones que componen la página, donde dentro de cada una hay una etiqueta “a” con un atributo “href” que contiene en enlace de la página a la que nos dirige, dentro de cada etiqueta “a” hay una imagen que cubre un tercio de la página, y dentro de ellas hay un título “h4” que contiene el nombre de la página a la que nos dirigirá la etiqueta “a” al clicar en ella.

A continuación, vamos a ver el código CSS de la página:

```
body {  
    overflow: hidden;  
}  
  
section {  
    position: absolute;  
    width: 640px;  
    height: 800px;  
    top: 75px;  
    display: inline-block;  
}  
  
img {  
    filter: brightness(0.6);  
    max-width: 100%;  
    max-height: 100%;  
    width: 100%;  
    height: 100%;  
    object-fit: cover;  
    transition: 0.5s;  
}  
  
img:hover{  
    filter: brightness(1);  
}  
  
div h4{  
    position: absolute;  
    top: 50%;  
    left: 50%;  
    transform: translate(-50%, 150px);  
    font-size: 40px;  
    font-weight: 100;  
    text-decoration-line: underline;  
    color: white;  
}  
  
.estudio {  
    left: 0%;  
}
```

```
.artista {  
  left: 640px;  
}
```

```
.trabajos {  
  left: 1280px;  
}
```

Como venimos haciendo de costumbre, los selectores de etiqueta componen la mayoría del archivo CSS de esta página (también de la página “galería.html”).

Vemos que el body, al igual que en el código CSS que mostramos antes, tiene la propiedad “overflow: hidden”, propiedad que tiene el mismo propósito que ya comentamos anteriormente. Pasamos por las secciones, que prácticamente se encarga de ajustar la posición, así como del tamaño de las etiquetas “section”.

Lo sigue el selector de etiqueta “img”, del cual, mas allá de las propiedades que ajustan el tamaño y capacitan el tamaño máximo y mínimo de la imagen, vamos a destacar varias propiedades, siendo la primera el “object-fit: cover” que hace que la imagen se ajuste al contenido donde se encuentra, evitando que esta se estire y pierda calidad, la siguiente propiedad sería el “filter: brightness()” el cual se encarga de aplicarle un filtro de brillo a la imagen, esto se lo aplicamos para combinarlo con el hover, para que cuando pases el ratón por encima, esta haga un efecto de iluminación, que va acompañado de una transición para darle un efecto de desvanecimiento.

Pasamos a ver ahora el selector “div h4” el cual es necesario para que el texto se pueda colocar por encima de la imagen, esto lo hacemos haciendo uso de la propiedad “position: absolute” junto el “transform: translate()” que mueve el texto en el eje X e Y dentro del contenedor que se encuentra. Las siguientes propiedades son solo para darle formato al texto (tamaño, color, peso, etc).

Terminando con los selectores, tenemos los selectores de clase que simplemente han sido creados para aplicárselos a cada sección para posicionarlos dentro de la página.

Por ultimo vamos con las páginas de “normales.html”, “pequetatus.html” y “covers.html”, estas páginas actúan como carruseles, comparten prácticamente los mismos elementos, a excepción de “covers.html” que tiene menos botones en la barra de navegación del carrusel, pero su estructura es la misma que el resto de las páginas.

Vamos a ver el código HTML de la página “normales.html”:

```
<main>  
  <div class="carrousel">  
    <div class="grande">  
        
        
        
        
        
      
```

```

        
        
        
        
    </div>
    <ul class="puntos">
        <li class="punto activo"></li>
        <li class="punto"></li>
        <li class="punto"></li>
        <li class="punto"></li>
        <li class="punto"></li>
        <li class="punto"></li>
        <li class="punto"></li>
        <li class="punto"></li>
        <li class="punto"></li>
        <li class="punto"></li>
    </ul>
    <a href="/galeria.html" class="boton">Volver</a>
</div>
</main>

```

Podemos ver que el código se compone de un contenedor “div” y otro contenido dentro de este, estos son los que componen el carrusel, ya que sin ellos no funcionaría el carrusel. Dentro del contenedor interno, tenemos contenidas las imágenes que estarán paralelas unas con otras, esto es así para dar el efecto de carrusel. Fuera del contenedor, tenemos una lista no numerada, que actuará como el menú de desplazamiento del carrusel, así como un enlace que nos hará volver a la página anterior.

Procedemos a ver el código CSS de la página:

```

body {
    background-image: url('../../Fotos-Proyecto/Logo-Fondo/fondo.jpeg');
    background-size: cover;
    background-repeat: no-repeat;
    background-attachment: fixed;
}

.carrousel {
    width: 510px;
    height: 640px;
    margin-top: 120px;
    margin-left: 725px;
    overflow: hidden;
}

.carrousel.grande {
    width: calc(510px * 10);

```

```
display: flex;
flex-flow: row nowrap;
justify-content: space-between;
align-items: center;
transition: all .5s ease;
transform: translateX(0%);
}
```

```
.carrousel .img {
width: 510px;
height: 640px;
background-size: cover;
}
```

```
.carrousel .puntos {
position: absolute;
display: flex;
background-color: #000000;
width: 615px;
height: 50px;
border-radius: 25px;
top: 765px;
left: 665px;
align-items: center;
align-content: center;
align-self: center;
justify-content: center;
align-items: center;
}
```

```
.carrousel .punto {
width: 30px;
height: 30px;
margin-right: 25px;
cursor: pointer;
background-color: black;
border-radius: 25px;
border: 2px solid #efb810;
transition: 0.5s;
}
```

```
.carrousel .punto:hover {
background-color: #efb810;
}
```

```
.carrousel .punto.activo {
background-color: #efb810;
}
```

En este archivo CSS, a diferencia de los ya vistos, predominan los selectores de clase, ya que son necesarios para la creación del carrusel. Para hacerlo de manera concisa y clara, la clase “.carrousel” es para el primer contenedor, en este marcamos el tamaño que va a tener, y lo más importante, el uso de la propiedad “overflow: hidden”. Esto es importante porque actúa como una ventana, es decir, tú al mirar a una ventana, ves solo lo que la ventana permite que veas, pero no ves a través de los muros, solo a través de la ventana, pues aquí es igual.

La siguiente clase que es descendiente, tiene que ser una que contenga el tamaño de la ventana multiplicado por el número de imágenes que queremos meter en el carrusel, que en nuestro caso son diez. Para el cálculo usamos la propiedad “width: calc()”.

El resto de clases no tienen mucho misterio, simplemente se encargan del tamaño y la posición de los elementos dentro del contenedor, siendo las clases descendientes de “.carrousel .puntos” la más destacable entre estas, ya que es la que contiene las propiedades que se le dan a la lista no numerada para que se vea como un menú de desplazamiento, así como los botones que contiene, que de ello se encarga la clase “.carrousel .punto”.

Las últimas clases destacables son las que hacen uso de pseudoclases, como el “hover”.

Como curiosidad, la clase “.carrousel .punto.activo” es una clase que se ejecuta junto con un script ubicado en un archivo JavaScript, que aplica dicha clase al contenedor seleccionado para hacer ver que hemos seleccionado el botón.

En estas tres páginas se hace uso de un script, que no voy a comentar en profundidad porque para ser sinceros lo saque de un tutorial que dejare, en resumen, prácticamente enumera los puntos de la lista y los convierte en botones, así como las imágenes, y que al pulsar el botón hace un cálculo donde aplica una propiedad “transform” al contenedor para modificar su posición y que la imagen se ajuste a la ventana que previamente mencionamos.

A continuación, dejo el link donde se entiende mejor el funcionamiento del carrusel:

<https://www.youtube.com/watch?v=2CEptqw-bSQ>

5. Elementos no comunes

Para terminar, comentaremos a continuación los elementos no comunes, estos son las páginas que no comparten su estructura con otras, dichas páginas son “eventos.html” y “reservas.html”.

Empezamos por “eventos.html”. Esta se trata de una página con scroll, que se divide en cuatro contenedores de texto, donde dentro de estos habrá un título con el titular de la noticia/evento, una foto adjunta que será el cartel promocional del evento en cuestión y por último, un botón que te lleva a saber más sobre el evento.

Su código HTML es el siguiente:

```
<main>
  <div>
    
    <p>Iniciamos Halloween con un día especial en el estudio.</p>
    <a href="/evento1.html">Leer más</a>
  </div>
  <div>
    
    <p>¡Nos vamos a "El rincón de Esteban"!</p>
    <a href="/evento2.html">Leer más</a>
  </div>
```



```
<div>
  
  <p>Nos estrenamos en la "Jerez Tattoo Convention 2023"</p>
  <a href="/evento3.html">Leer más</a>
</div>
<div>
  
  <p>¿Quieres un tatuador en tu boda? ¡Llámanos!</p>
  <a href="/evento4.html">Leer más</a>
</div>
</main>
```

Y su código CSS es el que vamos a ver ahora:

```
body {
  background-image: url('../Fotos-Proyecto/Logo-Fondo/fondo.jpeg');
  background-size: cover;
  background-repeat: no-repeat;
  background-attachment: fixed;
}
```

```
div {
  position: relative;
  width: 1300px;
  height: 280px;
  top: 100px;
  bottom: 75px;
  left: 320px;
  background: rgba(255,255,255,0.6);
  display: flex;
  margin-bottom: 50px;
  align-items: center;
}
```

```
div img {
  position: relative;
  width: 156px;
  height: 280px;
  background-size: cover;
  margin-right: 100px;
}
```

```
div p {
  position: relative;
  font-family: 'Josefin Slab';
  font-size: 40px;
  text-align: center;
  align-content: center;
```

```

}

div a {
  position: absolute;
  height: 30px;
  width: 125px;
  margin-top: 15%;
  margin-left: 85%;
  color: #efb810;
  background: #000000;
  border-radius: 12px;
  border: none;
  text-decoration: none;
  text-align: center;
  font-size: 12px;
  line-height: 30px;
  cursor: pointer;
  transition: 0.5s;
}

div a:hover {
  color: #000000;
  background: #efb810;
}

```

Aquí volvemos al uso mayoritario de los selectores de etiquetas, prácticamente no hay ninguno a destacar, ya que volvemos a lo de siempre, todos contienen las propiedades que modifican tamaño, color, texto, etc. Si acaso lo más destacable es que en este archivo CSS a diferencia del resto, no tiene la propiedad “overflow: hidden” porque como ya hemos comentado de forma reiterada, se arregló el error del footer, y además que es necesario tener en esta página una barra de navegación ya que tiene scroll para poder navegar entre las noticias.

Ahora terminamos con la última página de todas, que sería “reservar.html”. Como su nombre ya indica esta va a tener un formulario que el cliente deberá rellenar para concertar una cita.

El código HTML de dicha página es el siguiente:

```

<main>
  <div>
    <form>
      <label for="nombre">Nombre:*</label><br>
      <input type="text" name="nombre" id="nombre"><br>
      <label for="apellidos">Apellidos:*</label><br>
      <input type="text" name="apellidos" id="apellidos"><br>
      <label for="telefono">Teléfono:*</label><br>
      <input type="text" name="telefono" id="telefono"><br>
      <label for="diseno">Inserte imagen del diseño que te quieres realizar:*</label><br>
      <input type="file" name="diseno" id="diseno"><br>
      <label for="tamanyo">Especifique el tamaño del tatuaje en centímetros (cm):*</label>
      <input type="text" name="tamanyo" id="tamanyo">
      <br>
    </form>
  </div>
</main>

```

```
<label for="fecha">Selecciona la fecha:*</label>
<input type="date" name="fecha" id="fecha"> <br>
<button type="submit">Enviar</button>
</form>
</div>
</main>
```

Como vemos se compone de un solo contenedor y muchos elementos de formulario, en los cuales se incluyen los “input” de tipo texto, los de tipo “file” y el de tipo “date”.

Dentro también del formulario tenemos un botón para enviar los datos del formulario.

Hay que destacar que hay un cambio de diseño respecto al modelo creado en Figma, y es que en el apartado de insertar la imagen, el propuesto en el modelo Figma no es un “input” tipo “file”, sino que es más bien un contenedor “drag and drop” e informándome acerca de ello se hace con herramientas que no hemos visto en el módulo, ya sea porque son herramientas que trae un framework, o porque es código JavaScript puro, hay muchas maneras de hacerlo pero opte por no usar ninguna y cambiarlo por un “input” tipo “file” debido a problemas con el tiempo y falta de conocimiento.

Su código CSS es el que veremos a continuación:

```
body {
  background-image: url('../Fotos-Proyecto/Logo-Fondo/fondo.jpeg');
  background-size: cover;
  background-repeat: no-repeat;
  background-attachment: fixed;
  overflow: hidden;
}

div {
  position: absolute;
  width: 920px;
  height: 750px;
  top: 100px;
  bottom: 75px;
  display: inline-block;
  background: #000000;
  left: 560px;
  border-radius: 20px;
}

div form{
  position: relative;
  font-size: 20px;
  color: #FFFFFF;
  top: 50px;
  margin-left: 40px;
}

div input[type="text"]{
  margin-bottom: 50px;
  width: 765px;
```

```
height: 30px;
font-size: 20px;
}

#tamanyo {
margin-bottom: 40px;
width: 115px;
height: 30px;
font-size: 20px;
}

div input[type="date"] {
margin-bottom: 40px;
width: 166px;
height: 30px;
font-size: 20px;
}

div input[type="file"] {
margin-bottom: 40px;
cursor: pointer;
}

button {
position: absolute;
height: 30px;
width: 125px;
margin-top: 10%;
margin-left: 75%;
background: #efb810;
border-radius: 12px;
border: none;
text-decoration: none;
text-align: center;
font-size: 12px;
color: #000000;
line-height: 30px;
cursor: pointer;
transition: 0.5s;
}

button:hover {
color: #000000;
background: #fbcf71;
}

button:active {
color: #000000;
background: #ad860c;
```

```
}
```

Para sorpresa de nadie, predominan los selectores de etiqueta, todos a excepción de las pseudoclasas, sirven para modificar el tamaño, disposición de los elementos, textos, color, etc.

Destacamos dentro de este archivo el uso del primer y único selector por ID que veremos en todos los archivos, este tiene el único propósito de modificar el “input” donde el cliente inserta el tamaño del tatuaje a realizar.

También como mencionamos antes, hay que destacar las pseudoclasas, ya que aquí no solo hacemos uso del “hover” sino también del “active”, que cambiara el color de fondo del boton al pulsarlo.

6. Conclusiones

El trabajo ha sido arduo y ha requerido bastante tiempo (y quedarse hasta tarde corrigiendo fallos de la página), pero es un trabajo del que me siento orgulloso independientemente de la nota. En un futuro si sale adelante como proyecto, me gustaría ver cómo quedaría la página siendo “responsive” y agregándole mas elementos.