

3.1. Guías de estilo. Elementos principales y pautas de diseño

Así como en nuestro día a día existen normas de circulación, protocolos de actuación, etc., también en el diseño web está lo que se denominan *guías de estilo*. La normalización de estilos a través de la creación de guías de estilo, en la que se recogen los criterios y normas que debe seguir un mismo proyecto, permite que la página o sitio web tenga una apariencia mucho más uniforme y atractiva para el usuario. Al igual que se crean guías de estilo para revistas o libros, también se realizan en el desarrollo de las interfaces web.

Este conjunto de normas para el diseño y la redacción de documentos son frecuentes en todo tipo de medios: escritos, orales y gráficos. Su objetivo es lograr un mensaje coherente, eficaz y correcto.

El objetivo principal de las guías de estilo es establecer un estilo uniforme para todo el sitio web; de esta forma se consigue un estilo homogéneo que favorece la navegación del usuario por el sitio. Desde el punto de vista del programador y diseñador, estas guías favorecen el desarrollo del sitio web.

La guía de estilo está dirigida a las personas encargadas del diseño y programación de la interfaz web con la misión de lograr una mejor toma de decisiones tanto en el proceso de diseño como en la fase posterior de mantenimiento del sitio web. Esto es de gran importancia, dado que normalmente el mantenimiento de un sitio web puede ser llevado a cabo por varias personas (figura 3.1). En ella se deben recoger aspectos como la introducción en la interfaz fotografías e imágenes, la gama de colores, los tipos de letra, etc., que se verán con mayor detalle en los siguientes apartados.

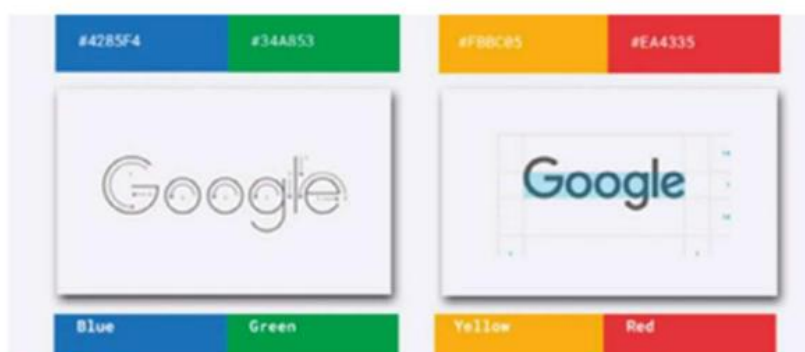


Figura 3.1
Guía de estilo de Google (<https://design.google/library/>).

3.1.1. Tipografías

¿En qué consiste una tipografía? La tipografía hace alusión al tipo de letra que se escoge para un determinado diseño. Según la RAE, su significado es “el modo o estilo en que está impreso un texto” o “la clase de tipos de imprenta”.

El uso de fuentes familiares mejora, en gran medida, la calidad de la lectura. Es muy importante elegir un estilo funcional, que permita su visualización sin problema en todos los navegadores. A la hora de elegirse la tipografía deben tenerse en cuenta la fuente (por ejemplo, la Arial es compatible con todos los navegadores), el estilo (no abusar de la negrita o el subrayado).

¡Cuidado, el subrayado suele usarse para enlaces, y el usuario se puede confundir!, el tamaño de la fuente y el color (se lee mejor un texto oscuro sobre un fondo claro que al revés, y no conviene usar combinaciones arriesgadas, como letra rosa sobre fondo rojo) (figura 3.2).



Figura 3.2

La legibilidad del menú y el estilo que se pretende transmitir al usuario son diferentes.

Las recomendaciones más frecuentes que el diseñador del sitio web debe tener en cuenta, en cuanto a la tipografía se refiere, son las siguientes:

- No es recomendable utilizar más de dos o tres tipografías en un mismo proyecto.
- Usar unas fuentes claras.
- Escoger un buen contraste entre las letras y el fondo.
- No utilizar líneas de texto demasiado largas.
- Utilizar un interlineado de 1,5 puntos adicional al tamaño de la letra del cuerpo.
- Evitar la justificación. Es recomendable la alineación a la izquierda.
- No utilizar demasiados subrayados; los dejaremos para los hipervínculos.

✓ La legibilidad es la cualidad deseable en una familia tipográfica. Se trata de la facilidad de la lectura de una letra. Esta cualidad puede venir determinada por varios parámetros como el interletrado, el interpalabrado o el interlineado.

3.1.2. Colores

En los grandes sitios web suele utilizarse un color asociado a cada sección, de esta forma es fácil identificar en qué apartado se está. El uso de una escala coherente también facilita la navegación. Se debe ser coherente y usar siempre un color para lo mismo, e intentar transmitir armonía y tener en cuenta la psicología del color (cuadro 3.1).

CUADRO 3.1
Colores y significados

Tranquilidad Calma Seriedad Salud	Lujo Elegancia Misterio Vanidad	Dulzura Infancia Delicadeza Sensibilidad	Pasión Energía Fuerza Peligro	Amistad Calidez Confianza Éxito
Felicidad Optimismo Energía Vitalidad	Frescura Naturaleza Esperanza Juventud	Elegancia Sobriedad Clasicismo Poder	Fiabilidad Solidez Equilibrio Templanza	Pureza Perfección Limpieza Bondad

3.1.3. Iconografía

En cuanto a los iconos, estos representan las acciones asociadas a su función. Es deseable que se mantenga una apariencia común entre todos los iconos y entre todas las páginas y secciones del sitio web. Al utilizar estos iconos se evita el uso excesivo de textos. Una buena elección de estos iconos es muy importante, puesto que, si un usuario no es capaz de determinar su significado, no se habrá conseguido el propósito de ahorrarle tiempo en la visualización de la página.

Hay muchas páginas web que ofrecen iconos gratuitos y de pago para descargar y utilizar en proyectos de diseño, incluyendo el diseño web. Algunas páginas populares para descargar iconos son las siguientes:

Font Awesome: Biblioteca de iconos y fuentes vectoriales que ofrece una amplia selección de iconos para utilizar en proyectos de diseño web.

Flaticon: Ofrece una amplia selección de iconos gratuitos y de pago para descargar. Los iconos están organizados por categorías, lo que facilita la búsqueda del icono adecuado para tu proyecto.

Iconfinder: Ofrece una amplia selección de iconos gratuitos y de pago para descargar. Los iconos están disponibles en diferentes formatos y tamaños, lo que los hace ideales para proyectos de diseño web.

Google Fonts Icons: Biblioteca de iconos creada por Google que sigue los principios del diseño Material Design. Los iconos están disponibles en diferentes estilos y se pueden personalizar fácilmente para adaptarse a las necesidades de tu proyecto.

Freepik: Ofrece una amplia selección de recursos gráficos gratuitos y de pago para descargar, incluyendo iconos, ilustraciones, fotos y más. Los recursos están disponibles en diferentes formatos y se pueden personalizar fácilmente para adaptarse a las necesidades de tu proyecto.

3.1.4 Extensiones de utilidad para diseñadores web

Las extensiones son herramientas que pueden mejorar y optimizar el flujo de trabajo de diseñadores y desarrolladores. Hay muchas extensiones disponibles que pueden ayudar a aumentar la productividad y la eficiencia en el trabajo diario. A continuación, se presentan algunas extensiones útiles para diseñadores y desarrolladores.

ColorZilla: Permite a los usuarios seleccionar y copiar colores de cualquier página web. También incluye herramientas para generar degradados y paletas de colores.

Fire Shot o Full Page Screen Capture: Son extensiones para navegadores que permiten a los usuarios capturar capturas de pantalla de páginas web completas. Las capturas de pantalla se pueden guardar en diferentes formatos y se pueden editar y anotar antes de guardarlas.

Responsive Viewer: Permite a los usuarios ver cómo se verá una página web en diferentes dispositivos y resoluciones de pantalla. Esto puede ser útil para probar la capacidad de respuesta de un sitio web.

Page Ruler Redux: Permite a los usuarios medir elementos en una página web. Los usuarios pueden dibujar una regla en la pantalla para medir el tamaño y la posición de los elementos.

WhatFont: Permite a los usuarios identificar las fuentes utilizadas en una página web. Los usuarios pueden pasar el cursor sobre el texto para ver información sobre la fuente, como el nombre y el tamaño.

WebDeveloper: Ofrece una amplia variedad de herramientas y funciones para desarrolladores web. Incluye herramientas para editar CSS, validar HTML, depurar JavaScript y más.

3.1.5 Pasos a seguir para crear una guía de estilo para un sitio web:

Define la identidad de tu marca: Antes de comenzar a crear una guía de estilo, es importante tener una idea clara de la identidad de tu marca y los valores que quieres transmitir a través del sitio web. Esto te ayudará a tomar decisiones sobre el estilo visual y el tono de voz que debes utilizar.

Elige una paleta de colores: Es importante elegir una paleta de colores que refleje la identidad de tu marca y sea coherente con el resto de tus elementos visuales. Asegúrate de incluir información sobre los colores principales y secundarios, así como sobre cómo deben utilizarse en diferentes elementos del sitio.

Selecciona las tipografías: Elige fuentes que sean legibles y estén en línea con la identidad de tu marca. Incluye información sobre las fuentes principales y secundarias, así como sobre cómo deben utilizarse en diferentes elementos del sitio, como títulos, subtítulos y texto del cuerpo.

Establece pautas para el uso de imágenes: Es importante establecer pautas claras sobre cómo deben usarse las imágenes. Incluye información sobre el tipo de imágenes que deben utilizarse, cómo deben editarse y cómo deben presentarse en el sitio.

Define el tono y el estilo de escritura: Establece pautas claras sobre el tono y el estilo que deben seguir los textos, incluyendo aspectos como el uso de la voz activa o pasiva, el nivel de formalidad y el uso de jerga o tecnicismos.

Selecciona los iconos: Los iconos son elementos gráficos que pueden ayudar a mejorar la navegación y la comprensión del contenido. Al crear tu guía de estilo, es importante establecer pautas claras sobre el uso de iconos, incluyendo información sobre el estilo y el tamaño de los iconos, así como sobre cómo deben utilizarse en diferentes partes del sitio.

Diseña botones, formularios y otros componentes: Es importante establecer pautas claras sobre el diseño y el uso de estos componentes, incluyendo información sobre el tamaño, el color, la forma y el estilo de los botones y formularios, así como sobre cómo deben utilizarse en diferentes partes del sitio. Asegúrate de diseñar componentes que sean coherentes con la identidad visual de tu marca y que sean fáciles de usar para tus usuarios.

Una vez que hayas definido todos estos elementos, puedes compilarlos en un documento para crear tu guía de estilo. Asegúrate de presentar la información de manera clara y concisa, utilizando ejemplos visuales para ilustrar tus pautas. Además, ten en cuenta que, una guía de estilo es un documento vivo que puede evolucionar con el tiempo a medida que cambian las necesidades y preferencias de tu marca y tu audiencia.

3.2. Revisión de lenguaje de marcas HTML

Los sitios web están compuestos por páginas que están escritas en algún lenguaje. En la actualidad, debido a que el desarrollo de páginas web está cada vez más presente, existe un creciente desarrollo del sector profesional relacionado con el diseño web. Por lo tanto, comprender, conocer y manejar correctamente el lenguaje de marcas asociado al desarrollo de páginas web es recomendable.

HTML es el lenguaje de marcado de hipertexto utilizado en las páginas web. Este tipo de texto presenta una forma estructurada y agradable, con hipervínculos que conducen a otros documentos y con inserciones multimedia (sonido, imágenes, vídeos...). El fundamento de este lenguaje es la especificación, en el texto, de la estructura lógica del contenido (título, párrafos de texto normal, enumeraciones, definiciones, citas...). Las características principales de este lenguaje son:

- a) Sencillez.
- b) No hay variables.
- c) No se compila.
- d) Se interpreta por el propio navegador.
- e) A las instrucciones se les llama *etiquetas*.
- f) Permite escribir hipertexto.

3.2.1. Estructura básica de un fichero HTML

Un documento en HTML debe estar delimitado en sus extremos por las llamadas *etiquetas*. Estas consisten en palabras reservadas que dividen el documento en secciones y les dan formato. Para crear un documento HTML, debe delimitarse todo el archivo con la etiqueta `<html>`. Dentro de la estructura se distinguen dos partes claramente diferenciadas, la cabeza y el cuerpo, esto es, *head* y *body*. A continuación, se muestran las etiquetas que han de utilizarse para definir un documento HTML y las secciones a las que hacen referencia en su representación web.

- Sintaxis de las etiquetas:

```
<identificador_de_la_etiqueta> ... </identificador_de_la_etiqueta>
```

- Ejemplo:

```
<html>
  <head>
    ...
  </head>
  <body>
    ...
  </body>
</html>
```

UT3.- Guías de estilo y lenguaje de marcas

Un sitio o aplicación web estará compuesto por páginas o documentos html (o bien por scripts que generarán html). Un ejemplo básico de un documento HTML5 (HTML versión 5) es por ejemplo:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Prueba Javascript</title>
</head>
<body>
  |
</body>
</html>
```

Los documentos HTML5, son documentos de texto plano (ASCII), con la extensión .htm o .html.

Dentro contendrán al menos elementos con sus etiquetas (o *tags*) definidos por el estándar HTML5:

- Un declarador DOCTYPE para indicar la versión con la que está codificado el documento. Para HTML5, el declarador es **<!DOCTYPE html>**
- Un elemento padre llamado **<html>...</html>**. Según recomendación del W3C, es prescriptivo usar el atributo **lang="es"** para indicar el idioma del contenido textual del documento. El elemento **<html>** internamente tendrá siempre dos elementos: **<head>...</head>** y **<body>...</body>**.
- Elemento **<head>...</head>**: es la parte **no visible** del documento. Se trata de una «cabecera» previa al contenido del documento.
- Elemento **<body>...</body>**: es el «cuerpo» del documento, o sea, donde se codificará el contenido importante y visible. Dentro de **<body>**, incluiremos todos los elementos que dan contenido y estructura a la página web.

Dentro de **<head>**, incorporaremos diferentes elementos para declarar diferentes cosas:

- **<title>...</title>**: siempre deberemos darle título al documento; este título se mostrará en la pestaña del navegador.
- **<meta ... />**: se trata de metadatos, y sirven para aportar información sobre el documento.
 - charset="utf-8"**: codificación de caracteres utilizada (para el español necesitamos esta).
 - name="robots"**: acciones que deben tener en cuenta los motores de búsqueda e indexación.
 - name="author" | "description" | "keywords" | ...**: metadatos que sirven para indexar el documento.
 - name="theme-color" content="#ff6600"**: color del tema del navegador.
 - name="viewport" content="initial-scale"**: directrices para el viewport (se verá más adelante).
 - http-equiv="refresh"**: metadatos de la propia petición http (p. ej., refrescar cada x segundos).
- **<base ... />**: URL, o dirección base cuando tengamos elementos que carguen cosas desde el servidor (por ejemplo, imágenes, videos, etc.). Si no se pone, la dirección base corresponde a la ruta donde esté alojado el documento.
- **<link ... />**: para vincular hojas de estilo a un documento.

! IMPORTANTE

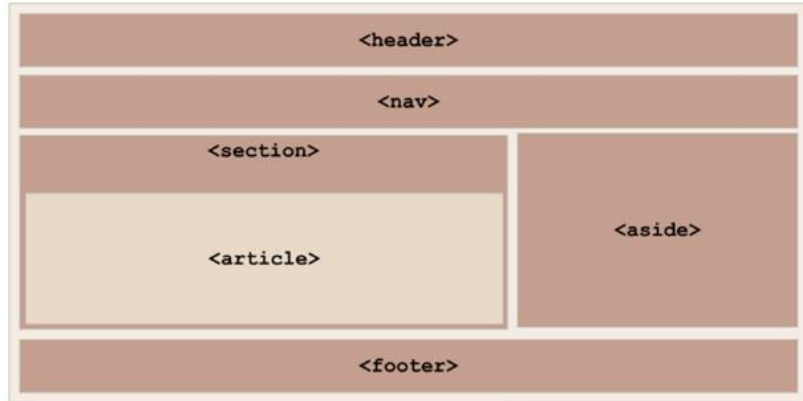
Los elementos **<meta...>**, **<base...>**, **<link...>** y otros que veremos en breve son elementos vacíos (sin contenido) y, por tanto, no les corresponde etiqueta de cierre del tipo **</meta>**.

En HTML5 se pueden autocerrar usando al final **/>** [**<meta... />**] o, simplemente, no cerrarlos.

UT3.- Guías de estilo y lenguaje de marcas

El elemento `<body>...</body>` es el cuerpo del documento, o sea, donde se codificará el contenido importante y visible. Dentro de `<body>` incluiremos todos los elementos que dan contenido y estructura a la página web.

En HTML5 se hace espacialmente énfasis en marcar todos los contenidos del documento mediante etiquetas que le darán un **significado semántico**. A un primer nivel, esto significa utilizar etiquetas que definen las secciones más importantes del documento, tal como vemos en esta imagen, y, más abajo, en el código HTML5:



```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="utf-8" >
    <title>¡Hola, mundo!</title>
  </head>
  <body>
    <header>
      <h1>¡Hola, mundo!</h1>
    </header>
    <nav>El menú, si procediera</nav>
    <section id="noticias">
      <h1>Blog de Noticias</h1>
      <article>
        <h2>Primera noticia</h2>
        <p>Texto de la noticia...</p>
      </article>
    </section>
    <aside>
      <h3>Artículos relacionados</h3>
    </aside>
    <footer>
      <p>Copyright 2022</p>
    </footer>
  </body>
</html>
```

3.2.2. Elementos básicos en HTML: texto, párrafos, estilos y formatos

El texto que se desea mostrar en las páginas web se escribe entre un conjunto de etiquetas que aportan estilo y formato. Una de las más conocidas son las etiquetas `<p>` y `</p>`, las cuales definen un párrafo e introducen un salto y dejan una línea en blanco antes de continuar con el resto del documento.

- Por ejemplo, si en el código HTML se escribe:

```
<p><b>Hola, estamos escribiendo en un párrafo.</b></p>
<p>Ahora ya hemos cambiado de párrafo.</p>
```

- En pantalla, se verá:

Hola, estamos escribiendo en un primer párrafo.
Ahora ya hemos cambiado de párrafo.

Para los contenidos textuales, tenemos las siguientes etiquetas —que ya existían en las primeras versiones de HTML— para marcar los elementos de párrafos y encabezados:

- `<h1>...</h1>`, `<h2>...</h2>`, `<h3>...</h3>`, `<h4>...</h4>`, `<h5>...</h5>`, `<h6>...</h6>`: *headers*. Se usan para marcar los encabezados y títulos de las diferentes secciones.
- `<p>...</p>`: párrafos. Los contenidos de tipo texto siempre deben estar en un elemento `<p>`
- `<div>...</div>`: divisor de bloque. Se trata de un contenedor multiuso para agrupar otros elementos y contenidos del HTML con el objetivo de darles un formato común.
- `...`: agrupador de elementos *in-line*. Parecido a `<div>`, pero, cuando se renderiza en el navegador, no crea un salto de línea, como sí lo hace `<div>`.
- `
` [o cerrándolo: `</br>`]: *break-line*. Fuerza a que se produzca un salto de línea.
- `<hr>` [o cerrándolo: `</hr>`]: *horizontal-row*. Dibuja una línea horizontal y fuerza un salto de línea.
- `<!-- comentario -->`: es una directiva para incorporar comentarios dentro del HTML.

Estos comentarios no son renderizados por el navegador. Se quedan incrustados en código.

Además, HTML5 incorpora una serie de etiquetas para **marcado semántico** de los textos. Hay que usarlas siempre que se pueda, ya que proporcionan más significado al contenido y ayuda a que los motores de búsqueda indexen mejor nuestros documentos html. Algunas de ellas son:

- `...`: texto importante; equivale a algo subrayado en negrita.
- `...`: enfatizar. Se puede anidar; cada nivel indica más énfasis.
- `<mark>...</mark>`: texto marcado o resaltado, como una anotación subrayada con un rotulador.
- `<cite>...</cite>`: referencia a una fuente o autor del texto citado.
- `<small>...</small>`: comentarios al margen, que tienen poca relevancia (letra pequeña).
- `<code>...</code>`: trozos de código fuente o texto monoespaciado.
- `...`: para texto que deseamos que aparezca tachado (inválido u obsoleto).
- `<ins>...</ins>`: para marcar texto que se ha sido añadido al documento más tarde.
- `<dfn>...</dfn>`, `<abbr>...</abbr>`: definiciones y abreviaturas.
- `<date>...</date>`, `<time>...</time>`: para vincular un texto a una fecha u hora concreta.
- `_{...}`, `^{...}`: subíndices, superíndices.

HTML5 mantiene por compatibilidad, ciertas etiquetas de marcado visual, pero, a su vez, la W3C aconseja no usarlas. Estas son:

- `...`: *bold*, texto en negrita; mejor usar `...`.
- `<u>...</u>`: *underline*, subrayar; mejor usar CSS para realizar el estilizado del texto.
- `<s>...</s>`: *stroke*, texto tachado; mejor usar `...`.
- `<i>...</i>`: texto en itálica (cursiva); mejor `...`. Esta etiqueta, de facto, se usa para insertar iconos.

3.2.3. Enlaces

Para la creación de enlaces se utiliza la etiqueta `<a>`. Esta etiqueta tiene dos formas de uso. En primer lugar, indicar la URL (dirección) del recurso con el que se desea enlazar, para ello se utiliza el atributo `'href'`, seguido de la URL. La segunda opción, consiste en nombrar la sección para que se pueda acceder a través de otros enlaces; en este caso se utiliza el atributo `'name'`.

- Sintaxis de los enlaces:

```
<a href=""> ... </a>
<a name=""> ... </a>
```

- Por ejemplo, si con el código HTML se escribe:

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
    <title>PRUEBA ENLACES</title>
  </head>
  <body>
    <p><a href="https://www.sintesis.com">Editorial Síntesis</a></p>
  </body>
</html>
```

Si algo caracteriza al servicio WWW, es la capacidad de enlazar información de una página a otra (incluso de servidores diferentes) mediante el uso de hiperenlaces.

En HTML5, cualquier cosa puede ser un **hiperenlace**: un texto, una imagen, un vídeo... Para generarlo, la etiqueta es bien simple: `<a>` enlace `<a>`. Los atributos más importantes de esta etiqueta son:

- **href="..."**: dirección del recurso al que vincula el enlace. Es imprescindible ponerlo.
 - Vincular elemento de la propia página: `Ir a sección 2`
 - Vincular otro documento html: `Contáctanos`
 - Vincular a otro sitio web: `Ir a buscador`
 - Vincular documentos no html: `Descarga el zip`
 - Vincular otros servicios: `e-mail al Admin`
`Clic para llamarnos`
- **target="..."**: indicamos dónde queremos que se abra el recurso.
 - Abrir en la misma pestaña del navegador: `Contacto`
 Nota: se puede poner `target="_self"`, que es lo mismo que omitirlo.
 - En nueva pestaña: `Contacto`
 - En marco `[iframe]`: `Contacto`
 Nota: se abrirá en `<iframe name="marcol"><iframe>`, se verá más adelante en la formación.

3.2.4. Imágenes

A la hora de diseñar un sitio web, la incorporación de imágenes aporta un mayor atractivo para el usuario, que si, por el contrario, solo se incluye texto; ahora bien, es aconsejable que la calidad de las imágenes sea óptima. Además, cuando se utilizan imágenes, estas deben respetar los derechos de publicación, en el caso de no ser realizadas expresamente para el sitio web donde van a ser usadas.

- Sintaxis de la inserción de imágenes:

(sin atributos)

```

```

(con atributos)

```

```

- Por ejemplo, si con el código HTML se escribe:

```
<html>  
<head>  
  <title>EJERCICIO 2.3. EJEMPLO CON IMÁGENES (CAPÍTULO 3)</title>  
</head>  
<body>  
    
</body>  
</html>
```

3.2.5. Formularios

Los formularios son documentos interactivos que se utilizan para recoger información en un sitio web. Esta información es enviada al servidor donde es procesada. Cada formulario contiene uno o varios tipos de controles, que permiten recolectar la información de varias formas diferentes, desde el ingreso de una palabra en un cuadro de texto, hasta la subida de un archivo.

- Sintaxis de la estructura de un formulario:

```
<form>  
...  
</form>
```

Uno de los controles más importante y utilizado es el elemento `<input>`, que se utiliza para crear controles interactivos en formularios. La forma en que `<input>` funciona viene determinada por el valor del atributo 'type'. El valor predeterminado del control `<input>` es 'text'; este permite el ingreso de una única línea de texto. A continuación, se recogen los valores más utilizados para el atributo 'type'.

A) Text

Se trata de un cuadro de texto vacío donde se puede escribir cualquier texto.

- Ejemplo:

```
<html>
  <head>
    <title>...</title>
  </head>
  <body>
    <form action="" method="get">
      <br>Nombre<br/>
      <input type="text" name="nombre" value=""/>
    </form>
  </body>
</html>
```

B) Button

Debe diferenciarse entre la acción de enviar que realizan los botones creados mediante el tipo 'submit' y otros botones que se deseen crear para el resto de las operaciones, que deben ser implementadas mediante el lenguaje JavaScript y que se pueden programar para realizar cualquier tarea, por muy compleja que sea.

- Por ejemplo, si se pone en el código:

```
<html>
  <head>
    <title>...</title>
  </head>
  <body>
    <form action="" method="get">
      <input type="button" name="guardar" value="Guardar"/>
    </form>
  </body>
</html>
```

C) Checkbox

Es una casilla de selección, se usa junto al atributo 'value' para definir el valor que se enviará. El atributo 'checked' se utilizará para indicar si el elemento está seleccionado.

- Por ejemplo:

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
<title>...</title>
</head>
<body>
  <form action="" method="get">
    <br>Elige una o varias opciones</br>
    <br><input name="op1" type="checkbox" value="op1"/>Opción 1
    </br>
    <br><input name="op2" type="checkbox" value="op2"/>Opción 2
    </br>
    <br><input name="op3" type="checkbox" value="op3"/>Opción 3
    </br>
  </form>
</body>
</html>
```

D) Radio

Este tipo de controles son similares a los 'checkbox', pero son mutuamente excluyentes. Se utilizan cuando solo puede escogerse una de las opciones. Cada vez que se selecciona una opción, se deselecciona cualquier otra que estuviera seleccionada. El parámetro 'name' se utiliza para indicar que dos o más 'radio' están relacionados y que, por tanto, no deben estar seleccionados al mismo tiempo.

- Ejemplo:

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
<title>...</title>
</head>
<body>
  <form action="" method="get">
    <br>Elige una opción</br>
    <br><input type="radio" name="Tipo" value="tipo1"/>Tipo 1
    </br>
    <br><input type="radio" name="Tipo" value="tipo2"/>Tipo 2
    </br>
  </form>
</body>
</html>
```


E) File

Es el control que permite al usuario seleccionar un archivo. Se puede usar el atributo 'accept' para definir los tipos de archivo que el control podrá seleccionar. Al pulsar sobre el botón, se despliega una ventana que permite navegar por el sistema y seleccionar el fichero deseado.

```
<html>
  <head>
    <title>...</title>
  </head>
  <body>
    <form action="" method="get">
      Seleccione un fichero
      <br><input type="file" name="adjunto"/></br>
    </form>
  </body>
</html>
```

F) Submit

Es el botón que envía el formulario con los datos introducidos por el usuario al servidor. Se puede establecer el texto que va a aparecer en el botón; si no se hace, se muestra el valor predeterminado: "Enviar consulta".

- Ejemplo:

```
<html>
  <head>
    <title>...</title>
  </head>
  <body>
    <form action="" method="get">
      <input type="submit" name="buscar" value="Buscar"/>
    </form>
  </body>
</html>
```

G) Select

Se utiliza para definir listas desplegables. Dentro de este control se puede definir el número de elementos que se deseen mediante la etiqueta `<option>`.

- Ejemplo:

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
    <title>...</title>
  </head>
  <body>
    <label for="lista">Lista desplegable</label><br/>
    <select id="lista" name="lista">
      <option value="" selected="selected">- selecciona una opción -</option>
      <option value="opcion1">Opción 1</option>
      <option value="opcion2">Opción 2</option>
      <option value="opcion3">Opción 3</option>
      <option value="opcion4">Opción 4</option>
    </select>
  </body>
</html>
```



H) TextArea

La etiqueta `<textarea>` permite una entrada de texto de varias líneas. El elemento se utiliza a menudo en un formulario para recopilar entradas del usuario, como comentarios o reseñas.

Un área de texto puede contener un número ilimitado de caracteres, y el texto se representa en una fuente de ancho fijo (normalmente Courier). El tamaño de un área de texto se especifica mediante los atributos **cols** y **rows** (o con CSS).

```
<textarea id="txtComen" name="comentario" rows="4" cols="50">
```

En este campo disponemos de varias líneas de texto

Permite introducir textos para descripciones, comentarios, reseñas, etc...

```
</textarea>
```

Existen más tipos de input o controles no descritos aquí. Para más información consultar el atributo **type** de la etiqueta `<input>`. También es muy interesante la etiqueta `<fieldset>`. (www.w3schools.com)

3.3. Tablas

Las tablas permiten mostrar la información de manera más organizada. Para la creación de una tabla, en primer lugar, se definen las características de la tabla y, a continuación, fila a fila, se indica el valor de cada uno de los elementos de cada columna, es decir, el contenido de cada celda.

También en HTML5, podemos generar tablas. Se generan con las etiquetas `<table>...</table>`. Dentro de las tablas, tendremos que ir componiendo filas mediante las etiquetas `<tr>...</tr>` (t. row). Dentro de cada fila, ubicamos celdas mediante etiquetas `<th>...</th>` (t. header) o `<td>...</td>` (t. cell).

```
<!-- Ejemplo de tabla simple de 2x3 -->
<table>
  <caption>Reino Animal</caption>
  <tr> <!-- 1ª fila -->
    <th>Tipo</th>
    <th colspan="2">Ejemplos</th>
  </tr>
  <tr> <!-- 2ª fila -->
    <td>Reptiles</td>
    <td>Serpiente</td> <td>Lagarto</td>
  </tr>
</table>
```

En el ejemplo anterior, solo se ha indicado el valor de cada una de las celdas. En la descripción de la etiqueta `<table>`, se pueden utilizar los atributos de la tabla. En el cuadro 3.3, se

CUADRO 3.3
Atributos principales de las tablas

Etiqueta	Uso
<i>border</i>	Indica el grosor del borde de la tabla y las celdas. Por defecto es cero, es decir, sin borde.
<i>width</i>	Indica el ancho de la tabla. Para designar este valor puede darse el tanto por ciento con respecto al tamaño de la página que tiene que ocupar, por ejemplo, 100% si es el ancho completo de la ventana del navegador, o en píxeles.
<i>align</i>	Indica la alineación de la tabla: <i>left</i> , <i>right</i> o <i>center</i> .
<i>caption</i>	Incorpora en la parte superior de la tabla un título descriptivo.

Las tablas suelen tener un título asociado para poder identificar con mayor facilidad el tipo de contenido que están mostrando. Para añadir un título se utilizará la etiqueta `<caption>` y, a continuación, se indicará el texto que aparecerá en la parte superior de la tabla:

- Por ejemplo, si se escribe el código:

```
<table border=1 width="50%" align=center>
  <caption>
    Título de la tabla
  </caption>
  <tr>
    <td>1,1</td>
    <td>1,2</td>
    <td>1,3</td>
  </tr>
  <tr>
    <td>2,1</td>
    <td>2,2</td>
    <td>2,3</td>
  </tr>
</table>
```

Hay más etiquetas relacionadas con la estructuración de la tabla (**thead**, **tbody**, **tfoot**). Existen, además, atributos para unir celdas (**colspan**, **rowspan**).

3.4 Listas

En HTML5 tenemos elementos para crear listas numeradas y no numeradas. Se trata de las etiquetas `...` y `...`. Estos elementos de lista aceptan atributos, como **start** (número por el que empieza el conteo) o **type** (tipo de numeración, aunque mejor usar propiedades CSS). Dentro del elemento de lista ubicamos los ítems que la conforman mediante las etiquetas `...`.

```
<!-- Ejemplo de listas anidadas -->
<h3>Reino Animal</h3>
<ol start="3">
  <li>Invertebrados</li>
  <li>Vertebrados
    <!-- otra lista dentro del li -->
    <ul type="square">
      <li>Mamíferos</li>
      <li>Reptiles</li>
    </ul>
  </li>
</ol>
```

3.5 Marcos

Aunque HTML5 deja obsoleto el uso de los marcos, a continuación se describirá brevemente su funcionamiento, puesto que, en algún caso, pueden encontrarse sitios web así implementados, sobre los que se haga necesario definir mejoras o en los que deban llevarse a cabo mantenimientos. Los marcos consisten en ventanas independientes incorporadas dentro de la página general; de esta forma, cada página quedará dividida en varias subpáginas, lo que permite realizar un diseño mucho más organizado y limpio a la vista. Este tipo de documentos, en lugar de tener las partes habituales *head* y *body*, incorpora un *frameset* en el lugar del *body*.

Para llevar a cabo la construcción de estos marcos se utiliza la etiqueta `<frameset>` y, a continuación, se indica la disposición del número de nuevas ventanas; en concreto, en primer lugar, se indica el número de columnas, así como su disposición dentro de la página principal. La etiqueta `<frame>` se utiliza para definir las características que va a tomar cada uno de los marcos, de forma individual.

- Sintaxis de los marcos (creación de las columnas):

```
<frameset cols=anchoColumna1, anchoColumna2, ...anchoColumnaN>
  <frame ...>
  <frame ...>
</frameset>
```

De esta forma quedan definidas las columnas, que darán lugar a los *frames* que componen la página web.

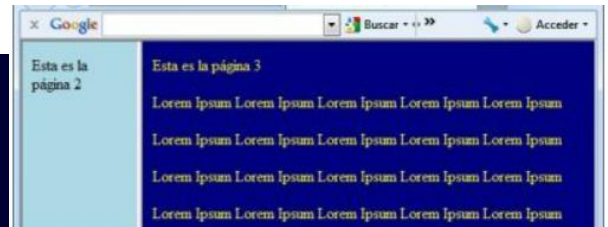
UT3.- Guías de estilo y lenguaje de marcas

```
<frameset cols=10%,20%,70%>
  <frame src="1">
  <frame src="2">
  <frame src="3">
</frameset>
```

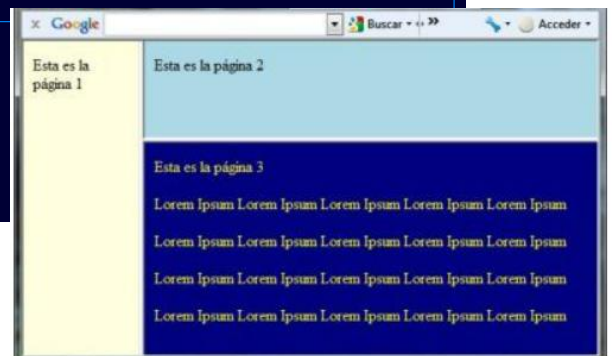
Además de esta distribución por columnas, también se crearán filas, obteniendo con la combinación de ambas casi cualquier tipo de diseño de marcos que se le ocurra al diseñador. La sintaxis utilizada es similar a la anterior, modificando el atributo 'cols' por 'rows' en la etiqueta <frameset>. En el siguiente ejemplo se muestra la combinación de ambas:

```
<frameset cols=30%,70%>
  <frame src="1">
  <frameset rows=40%,*>
    <frame src="2">
    <frame src="3">
  </frameset>
</frameset>
```

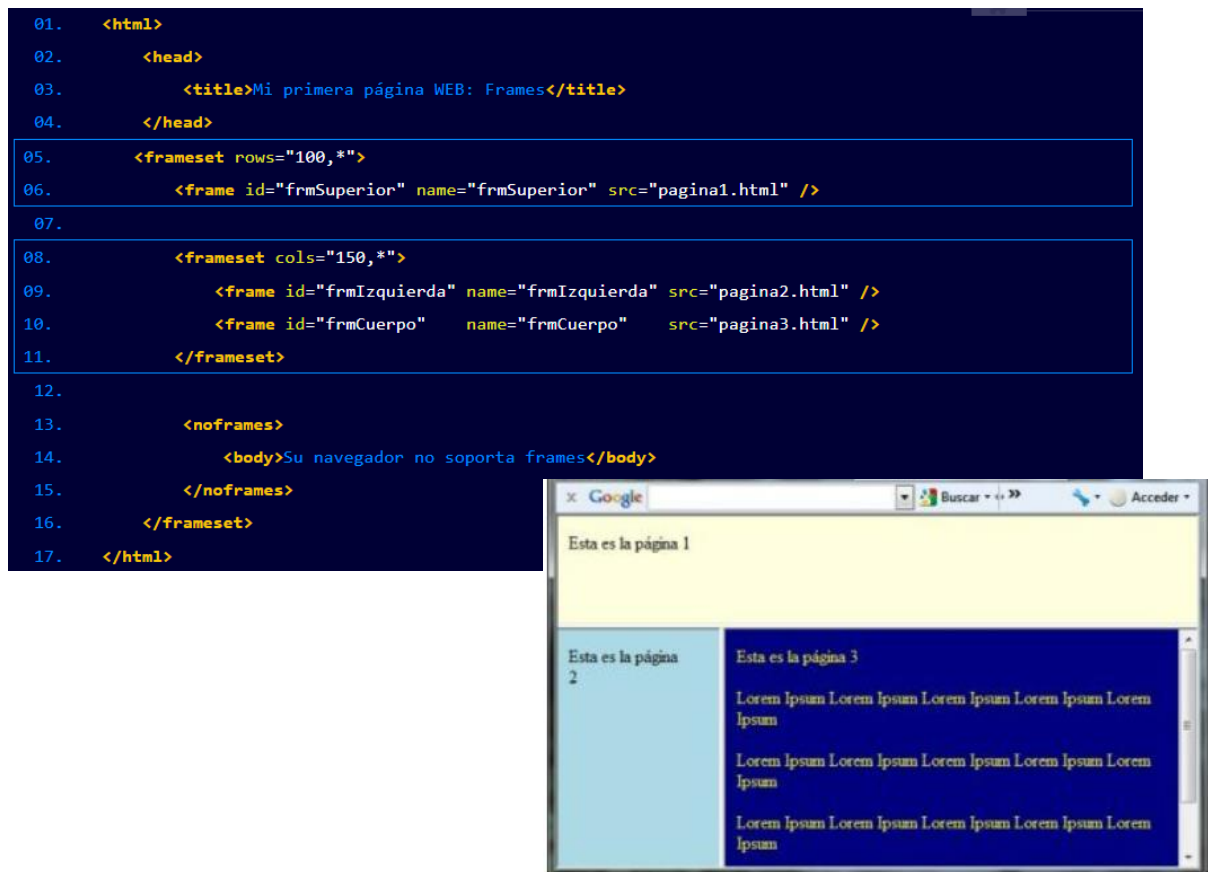
```
01. <html>
02.   <head>
03.     <title>Mi primera página WEB: Frames</title>
04.   </head>
05.
06.   <frameset cols="20%,*">
07.     <frame id="frmIzquierdo" name="frmIzquierdo" src="pagina2.html" />
08.     <frame id="frmDerecho" name="frmDerecho" src="pagina3.html" />
09.
10.   <noframes>
11.     <body>Su navegador no soporta frames</body>
12.   </noframes>
13. </frameset>
14. </html>
```



```
01. <html>
02.   <head>
03.     <title>Mi primera página WEB: Frames</title>
04.   </head>
05.
06.   <frameset cols="120,*">
07.     <frame id="frmIzquierdo" name="frmIzquierdo" src="pagina2.html" />
08.
09.     <frameset rows="100,*">
10.       <frame id="frmSuperior" name="frmSuperior" src="pagina1.html" />
11.       <frame id="frmInferior" name="frmInferior" src="pagina3.html" />
12.     </frameset>
13.
14.   <noframes>
15.     <body>Su navegador no soporta frames</body>
16.   </noframes>
17. </frameset>
18. </html>
```



UT3.- Guías de estilo y lenguaje de marcas



Estas etiquetas **<frameset>** y **<frame>** no están soportadas en HTML5. En cambio, si está soportada la etiqueta **<iframe>** con la que podemos crear un marco en el que cargar otra página web dentro de la principal. Va dentro de las etiquetas **<body></body>**. Entre sus etiquetas de inicio y cierre podemos escribir un texto que será mostrado en caso de que el navegador web no soporte frames.

Sus principales atributos son:

```
<iframe
  align: alineación horizontal del iframe con respecto a la página (bottom | left | middle | right | top).
  height: alto del iframe.
  marginheight: margen entre el contenido del iframe y sus bordes superior e inferior (en píxels)
  marginwidth: margen entre el contenido del iframe y sus bordes izquierdo y derecho (en píxels)

  scrolling: especifica si se mostrarán barras de desplazamiento en el iframe, pudiendo tener valor auto
  (se mostrarán sólo en caso necesario para poder hacer scroll en el contenido del marco), no (no se
  mostrarán nunca) o yes (estarán siempre visibles).

  src: la URL que se cargará en el frame.
  width: ancho del iframe (en píxels o porcentaje).
>
</iframe>
```

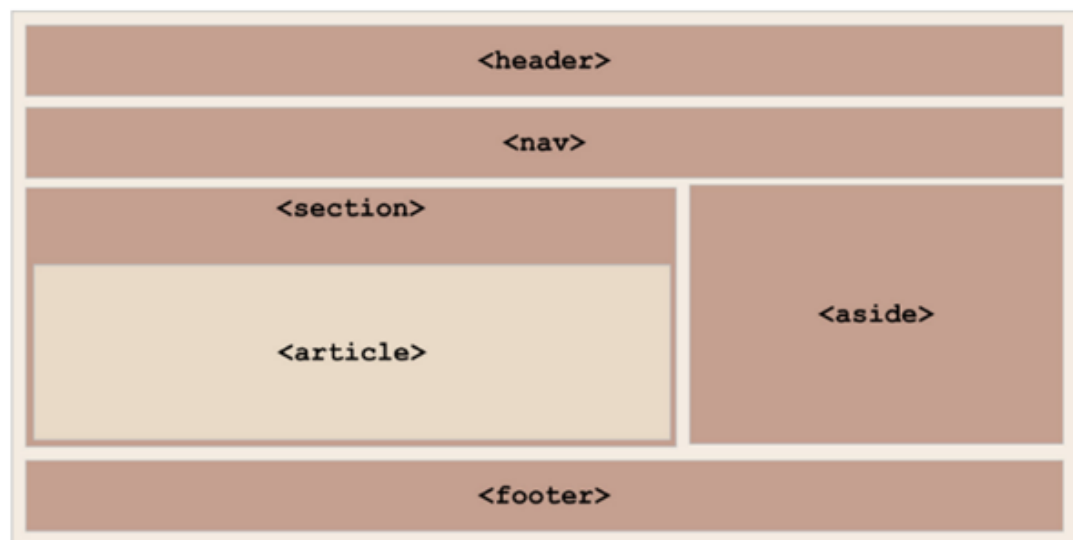
Ejemplo:



3.6 HTML5

HTML5 es la última versión del lenguaje para la programación de páginas web HTML. Los sitios implementados con este lenguaje solo pueden visualizarse correctamente en los navegadores más actuales, puesto que las versiones anteriores no son capaces de interpretar algunas de las nuevas etiquetas que se incluyen y que se verán en este apartado.

Una de las características principales de HTML5 son los elementos semánticos, también llamados *marcados semánticos*. Estos generan divisiones dentro del documento, pero, a diferencia de la habitual etiqueta `<div>`, la cual también define una división, sí describen al navegador claramente su propósito. Los nuevos elementos semánticos en HTML5 se describen a continuación:



`<header>`

Cabecera del sitio, define la parte superior de la página web. Suele mostrar algún elemento representativo que caracterice a la marca del sitio web. En la mayor parte de los casos, se sitúa el menú de navegación `<nav>`.

`<nav>`

La etiqueta `<nav>` crea un menú de navegación, que enlaza con las diferentes partes del sitio web. Normalmente se sitúa justo debajo del encabezado o a la izquierda de la página.

`<section>`

Se podría decir que esta etiqueta es la sustituta de `<div>`, y se utiliza para crear diferentes secciones dentro de una misma página web. Estas secciones pueden contener, a su vez, varios elementos de tipo `<article>`.

`<article>`

La etiqueta `<article>` se utiliza para describir las unidades de contenido. Se pueden definir tantos elementos de este tipo como información vaya a contener la página web. Pueden estar subdivididas en más partes: cabecera ('header'), cuerpo ('section') y pie de artículo ('footer').

`<aside>`

Este elemento define la barra lateral de una página web, aunque puede contener cualquier tipo de contenido, normalmente se utiliza para mostrar elementos que suponen un enlace hacia otros sitios web, redes sociales, etc. Ejemplo:

`<aside>`

`<h3>Barra lateral de navegación</h3>`

` Buscador de Google `

`</aside>`

`<footer>`

Equivale al pie de página. Se utiliza para situar los créditos del sitio web, los datos de contacto, el *copyright*, los formularios de contacto, etc. Puede utilizarse de forma general para todo el sitio web, y también para definir el pie de un elemento tipo `<article>`.

`<main>`

Representa el contenido principal del 'body' de un documento. El área principal consiste en el contenido que está directamente relacionado con el tema central de un documento. Este contenido debe ser único para ese documento. Es importante tener en cuenta que solo debe haber un elemento `<main>` en un documento, y no debe descender de un elemento `<article>`, `<aside>`, `<footer>`, `<header>` o `<nav>`. Ejemplo:

`<main>`

`<nav>...</nav>`

`<p>...</p>`

`</main>`

`<time>`

Representa la hora y la fecha en una página web.

`<email>`

En cuanto a los formularios, HTML5 incorpora nuevas etiquetas, tales como `<email>`. Esta crea un campo para introducir el correo electrónico y realiza una validación automática; de esta forma no será necesario implementar código que realice esta función en JavaScript.

`<mark>`

Esta etiqueta se utiliza para representar un texto resaltado, es decir, subrayado; de esta forma queda marcada su relevancia en el contexto. Ejemplo:

```
<main>
  <h2>Bienvenido a <mark>HTML5</mark></h2>
</main>
```

`<figure>`

La etiqueta `<figure>` representa contenido de carácter visual como: imágenes, ilustraciones, diagramas, fragmentos de código, etc. Este elemento puede ser movido a otra página o a un apéndice sin que afecte al flujo principal.

`<figcaption>`

Esta etiqueta añade un título a una figura. Se ubica anidado al elemento de imagen, `<figure>`, al que va asociado.

```

<!DOCTYPE html>
<html lang="es">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Título de la Web</title>
</head>
<body>
  <header>
    <h1><b>Imagen en HTML5</b></h1>
  </header>
  <section>
    <article>
      <h2>Bloque de contenido 1. Article</h2>
      <figure>
        
        <figcaption> Ejemplo de leyenda</figcaption>
      </figure>
    </article>
  </section>
</body>
</html>

```

<details>

Esta etiqueta tiene como finalidad mostrar información adicional de algún elemento de la página web. El funcionamiento de esta etiqueta consiste en un *widget* que oculta y muestra información, similar a un menú con efecto acordeón.

<summary>

Esta nueva etiqueta incorpora un sumario de la etiqueta <details>, es decir, añade información complementaria.

```

</section>
<details open>
  <summary> Prueba de sumario en HTML5</summary>
  <p>TEXTO QUE SE MUESTRA AL ICORPORAR UN
    ELEMENTO DETAILS EN HTML5 Y DESPLEGARLO
  </p>
</details>
</body>
</html>

```

► Prueba de sumario en HTML5

▼ Prueba de sumario en HTML5

TEXTO QUE SE MUESTRA AL ICORPORAR UN ELEMENTO DETAILS EN HTML5 Y DESPLEGARLO