

1-1-2024

## Capítulo 12 Introducción a HTML y al desarrollo de sitios web

Shambhavi Roy

Clinton Daniel  
*University of South Florida*

Manish Agrawal  
*University of South Florida*

Pablo Brescia  
*University of South Florida*

Clara Olivia Ocampo

*See next page for additional authors*

Follow this and additional works at: [https://digitalcommons.usf.edu/dit\\_tb\\_spa](https://digitalcommons.usf.edu/dit_tb_spa)

---

### Scholar Commons Citation

Roy, Shambhavi; Daniel, Clinton; Agrawal, Manish; Brescia, Pablo; Ocampo, Clara Olivia; and Labrador, Sonia, "Capítulo 12 Introducción a HTML y al desarrollo de sitios web" (2024). *FUNDAMENTALS OF INFORMATION TECHNOLOGY: Textbook – Spanish*. 12.  
[https://digitalcommons.usf.edu/dit\\_tb\\_spa/12](https://digitalcommons.usf.edu/dit_tb_spa/12)

This Book Chapter is brought to you for free and open access by the The Modernization of Digital Information Technology at Digital Commons @ University of South Florida. It has been accepted for inclusion in FUNDAMENTALS OF INFORMATION TECHNOLOGY: Textbook – Spanish by an authorized administrator of Digital Commons @ University of South Florida. For more information, please contact [digitalcommons@usf.edu](mailto:digitalcommons@usf.edu).

---

**Authors**

Shambhavi Roy, Clinton Daniel, Manish Agrawal, Pablo Brescia, Clara Olivia Ocampo, and Sonia Labrador

## CONTENIDOS DEL CAPÍTULO

<b>Panorama</b>	<b>270</b>
<b>¿Qué es HTML?</b>	<b>271</b>
Estructura de una página HTML	273
Elementos en HTML	274
Agregar enlaces	276
Agregar imágenes	277
Crear listas	277
Crear tablas	278
Crear formularios	279
<b>Hojas de estilo en cascada (CSS)</b>	<b>283</b>
<b>Diseño de sitios web y el guion gráfico</b>	<b>284</b>
<b>La estructura de un sitio web</b>	<b>286</b>
Hospedaje web	288
Editores de sitios web	290
<b>Extensiones HTML</b>	<b>291</b>
HTML dinámico (DHTML)	292
Lenguaje de marcado extensible (XML)	293
XML and JSON	296
<b>Términos y definiciones del capítulo</b>	<b>298</b>
<b>Caso del capítulo: ¡Wrenley salva a las mascotas!</b>	<b>300</b>

*Al principio puede que no sea rentable para las empresas estar en línea, pero ciertamente no será rentable no estar en línea.*

—Esther Dyson, inversora, filántropa.

## Panorama

La Internet de principios de los años setenta sólo admitía texto sin formato y su uso requería que los usuarios aprendieran la sintaxis precisa de varios comandos informáticos. Por ejemplo, para enviar un correo electrónico a `jane@example.com`, los internautas debían escribir el comando: `mail -s "asunto del correo" jane@example.com <<< "Cuerpo del correo"`. La mayoría de las personas encontraron esto demasiado desafiante y en los primeros tiempos los únicos cibernautas eran investigadores de ciencias informáticas en las universidades.

Esto cambió en 1989 cuando Sir Tim Berners Lee propuso y demostró un sistema de gestión de información basado en hipervínculos.<sup>157</sup> Los usuarios podían hacer clic en enlaces para navegar a otras páginas.<sup>158</sup> En 1993 Tim Berners Lee<sup>159</sup> concretó la primera versión de HTML, haciendo posible compartir información de manera mediante un clic en un enlace en una página. Entre diciembre de 1992 y marzo de 1993, Marc Andreessen (estudiante) y Eric Bina (graduado) de la Universidad de Illinois en Urbana-Champaign (UIUC) construyeron el navegador web Mosaic<sup>160</sup> mientras trabajaban en el Centro Nacional de Aplicaciones de Supercomputación (NCSA, siglas en inglés de *National Center for Supercomputing Applications*) en Urbana-Champaign. También agregaron la etiqueta `img` a `html` para manejar las imágenes<sup>161</sup> y lanzaron el navegador web a la Internet. El navegador pronto se hizo muy popular y adquirió el estatus de aplicación asesina que popularizó la Internet.<sup>162</sup> Sitios como Yahoo ofrecían directorios para que los internautas encontraran contenido en la emergente *World Wide Web*.

---

157 Véase la descripción general del CERN, el lugar de nacimiento de HTML y HTTP <https://home.cern/science/computing/birth-web/short-history-web> (consultado en enero del 2024).

158 La primera página web, creada por Sir Tim Berners Lee, se encuentra en <http://info.cern.ch/hypertext/WWW/TheProject.html> (consultado en enero del 2024).

159 La página de Wikipedia sobre Tim Berners Lee está en [https://en.wikipedia.org/wiki/Tim\\_Berners-Lee](https://en.wikipedia.org/wiki/Tim_Berners-Lee) (consultado en enero del 2024). En español [https://es.wikipedia.org/wiki/Tim\\_Berners-Lee](https://es.wikipedia.org/wiki/Tim_Berners-Lee) (consultado en enero del 2023).

160 NCSA también publicó uno de los primeros tutoriales de HTML. Es un documento sencillo que quizá sea la mejor introducción para aprender sobre HTML. Toda una generación de entusiastas, incluidos los autores de este libro, aprendieron HTML a partir de este documento. Está archivado en <http://seismo.berkeley.edu/~rallen/resources/html/HTMLPrimerAll.html> (consultado en enero del 2024).

161 El mensaje de Marc Andreessen del 25 de febrero de 1993 proponiendo la etiqueta `img` está en, <http://1997.webhistory.org/www.lists/www-talk.1993q1/0182.html> Algo interesante de este foro es que Tim Berners Lee responde a la propuesta sugiriendo que la etiqueta era innecesaria (consultado en enero del 2024).

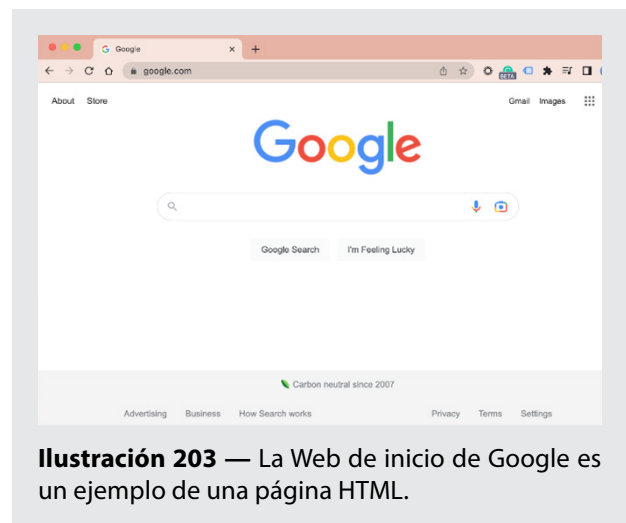
162 Información sobre Eric Bina en Wikipedia en [https://en.wikipedia.org/wiki/Eric\\_Bina](https://en.wikipedia.org/wiki/Eric_Bina) (consultado en enero del 2024), y en el MIT en <https://lemelson.mit.edu/resources/marc-andreessen-and-eric-bina> (consultado en enero del 2024).

Desde entonces, la Internet se ha vuelto cada vez más fácil de usar a medida que HTML ha mejorado,<sup>163</sup> agregando apoyos para imágenes y videos. Esto ha convertido a la Internet y a la *World Wide Web* en una fuerza poderosa para el intercambio y el comercio de información. Ya no se limita a usuarios especializados familiarizados con comandos informáticos exótico; en la actualidad, más de 5 mil millones de personas, más del 60% de la población mundial, utilizan la Internet, principalmente porque HTML y las tecnologías relacionadas han hecho que el acceso a la Web sea fácil, económico y significativo para cualquier persona en cualquier parte del mundo.

Este capítulo presenta HTML, la principal tecnología de usuario final utilizada en la Web.

## ¿Qué es HTML?

HTML (**Lenguaje de marcado de hipertexto**), sigla en inglés de *Hypertext Markup Language*, es el lenguaje de marcado estándar utilizado para crear páginas web y aplicaciones web. Es un conjunto de códigos informáticos que se utiliza para aplicar convenciones de diseño y formato a un documento de texto. El HTML hace el texto de las páginas web más interactivo y dinámico y nos permite especificar la ubicación de imágenes, tablas, enlaces, etc. en las páginas web mediante descripciones de texto simples. En otras palabras, HTML proporciona la estructura y el contenido de una página web, enseñando cómo se deben mostrar el texto, las imágenes, los hipervínculos, las tablas y otros medios en un navegador. Cuando se visita un sitio web, digamos <http://www.google.com>, lo que vemos es una página HTML (**Ilustración 203**). El navegador comprende las especificaciones de la página definidas en HTML y la representa como una página web que se puede ver y con la que es posible interactuar.



**Ilustración 203** — La Web de inicio de Google es un ejemplo de una página HTML.

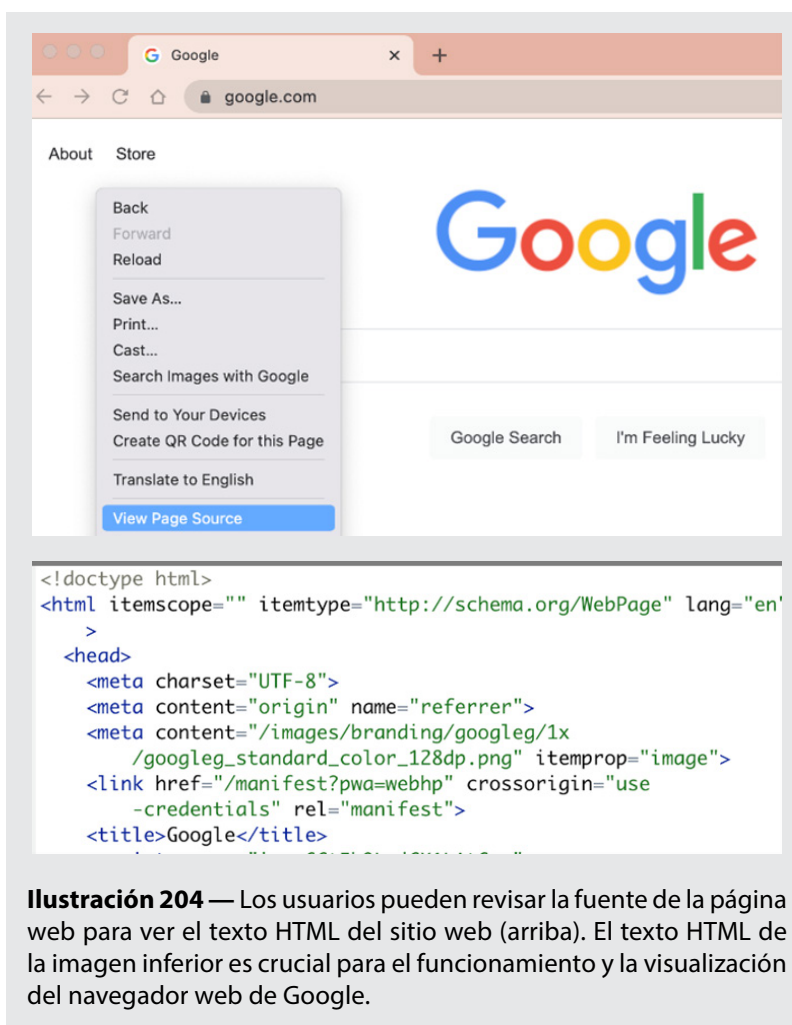
---

163 Una descripción general rápida de la evolución de HTML se encuentra en [https://www.washington.edu/accesscomputing/webd2/student/unit1/module3/html\\_history.html](https://www.washington.edu/accesscomputing/webd2/student/unit1/module3/html_history.html) (consultado en enero del 2024).

Hipertexto<sup>164</sup> significa “texto dentro del texto”.<sup>165</sup> En el contexto de la *World Wide Web* el hipertexto se refiere a un texto con un enlace a otra página dentro de él. El hipertexto es la forma que tiene la Internet para vincular dos o más páginas web (documentos HTML) entre sí.

Para ver el código HTML que define la página, se hace clic derecho en la página web (en la mayoría de los navegadores como Google Chrome o Firefox de Mozilla) y se elige “Ver código fuente de la página”. Esto mostrará la especificación de texto de esa página web. (Ilustración 204).

HTML ha sido un estándar de la Internet desde 1993 y todos los navegadores más comunes (por ejemplo, Google Chrome, Apple Safari, Microsoft Internet Explorer y Firefox de Mozilla) entienden HTML. Cuando se visita cualquier página escrita en HTML, digamos <http://www.google.com>, en cualquier navegador, todos reciben el mismo



164 El término “hipertexto” tiene una asociación con la Florida. Fue acuñado por Ted Nelson en 1965. En 1963, cuando se le ocurrió la idea, Nelson estaba documentando una investigación en el laboratorio John C. Lilly Dolphin en Miami (<https://web.archive.org/web/20150506080854/http://hyperland.com/TNvita>) (consultado en enero del 2024). El hipertexto se definió como “un conjunto de material escrito o pictórico interconectado de una manera tan compleja que no podría presentarse o representarse convenientemente en papel”. Nelson previó que “[el hipertexto] podría crecer indefinidamente e incluir gradualmente una gran parte del conocimiento escrito del mundo”. El hipertexto es visto como un sistema de archivos que, a diferencia de los utilizados en Windows y Mac, “[permite] organizar el contenido de los archivos en cualquier forma y de varias maneras a la vez”. Desafortunadamente, la falta de experiencia técnica de Nelson le impidió implementar la idea. Los hipervínculos modernos se consideran versiones unilaterales de los vínculos bilaterales propuestos por Nelson. La idea del hipertexto de Ted Nelson puede haber influido mucho en el diseño de las computadoras personales modernas, <https://archive.nytimes.com/bits.blogs.nytimes.com/2007/12/11/when-big-blue-got-a-glimpse-of-the-future/> (consultado en enero del 2024).

165 Nelson, Theodor Holm (Agosto 1965). “Complex Information Processing: A File Structure for the Complex, the Changing and the Indeterminate”. *ACM '65: Actas de la 20ª Conferencia Nacional de 1965*, 84–100. doi:10.1145/800197.806036, <https://dl.acm.org/doi/10.1145/800197.806036> (consultado en enero del 2024).

texto y la página se ve igual en todos estos navegadores. Por ejemplo, la página de inicio de Google tiene el mismo aspecto en todos ellos.<sup>166</sup>

HTML es una parte crucial de la Internet y constituye la base de todas las páginas web. Para definir elementos, HTML utiliza etiquetas (*tags*) que tienen atributos para describir elementos con más detalle. Estas etiquetas y atributos se utilizan para crear la estructura visual y semántica de una página web. Aunque HTML está diseñado principalmente para que lo interpreten las máquinas, como se verá en este capítulo su simplicidad hace que sea fácil incluso para que los seres humanos comprendan el contenido.

Prácticamente toda la información y las herramientas se presentan a los usuarios finales en forma de páginas web. Es decir, tú accedes a los recursos de tu escuela en la web, consultas el correo electrónico en la Internet, realizas un seguimiento de las cuentas bancarias en línea, los profesores ingresan sus calificaciones en la página web de la escuela, etc. Todos estos sitios web se crean utilizando HTML. Por lo tanto, HTML es una tecnología fundamental y útil para todos los profesionales modernos.

En este capítulo veremos los elementos básicos necesarios para crear páginas HTML y revisaremos los editores facilitadores de la creación de páginas web. Si encuentras interesante este capítulo y te gustaría aprender más sobre HTML y la creación de páginas web, también deberías aprovechar los tutoriales gratuitos en línea como W3Schools para informarte sobre HTML con más detalle,<sup>167</sup> ya que no es posible cubrir todas las etiquetas y atributos en un capítulo.

## Estructura de una página HTML

HTML es un lenguaje simple. Básicamente, se encierran los textos dentro de etiquetas para indicarle al navegador cómo representar el texto. Por ejemplo, el navegador mostrará `<b>este es texto en negrita</b>` como: “**este es texto en negrita**”. `<i>este es texto en cursiva</i>` se representará como: “*este es texto en cursiva*”. No necesitas editores sofisticados para crear una página HTML. Puedes crear una página HTML escribiendo comandos HTML en un bloc de notas y guardando el archivo como .html (un archivo con extensión html). Todas las páginas HTML siguen la misma estructura básica. Contienen una [declaración DOCTYPE](#) en la parte superior, seguida de una sección HTML. La sección HTML se compone de cabecera (*head*) y cuerpo (*body*). La [Ilustración 205](#) muestra un ejemplo básico.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>This is a Heading</h1>
    <p>This is a paragraph.</p>
  </body>
</html>
```

**Ilustración 205** — La estructura básica de una página HTML incluye DOCTYPE, cabecera (*head*) y cuerpo (*body*).

El contenido de la [Ilustración 205](#) es interpretado por el navegador de la siguiente manera:

---

166 Los navegadores introducen pequeños ajustes individuales, por ejemplo, en la elección de fuentes predeterminadas. Se prueba el rendimiento de los sitios web comerciales en todos los navegadores para tener en cuenta estos ajustes. Pero para los propósitos de este libro, tratamos a todos los navegadores como idénticos.

167 El tutorial HTML de W3Schools se encuentra en <https://www.w3schools.com/html/default.asp> (consultado en enero del 2024).

**Declaración DOCTYPE:** Esta es la primera línea de una página HTML y le dice al navegador qué tipo de documento esperar, en particular qué versión de HTML.

**Etiqueta HTML:** todo el código HTML se declara mediante etiquetas que son palabras claves especiales encerradas entre <>. La **etiqueta HTML** define el inicio y el final de un documento HTML. Todo el contenido del documento HTML está anidado dentro de las etiquetas HTML de apertura y cierre, es decir, entre <html> y </html>. Las etiquetas se cierran con el símbolo “/” en la etiqueta. Por lo tanto, </html> indica el final de la sección que comenzó con la etiqueta <html>.

**Etiqueta cabecera (head):** La **etiqueta cabecera** contiene información sobre el documento HTML que generalmente no se muestra al usuario. Esta sección incluye el título de la página, cualquier información sobre ella, como el autor y el historial de versiones (esto a menudo se denomina metainformación), referencias sobre la apariencia (las llamadas hojas de estilo) y datos sobre las interacciones del usuario (especificadas mediante un lenguaje llamado **JavaScript**). También se incluye una descripción adicional y palabras claves relevantes sobre la página web dentro de la etiqueta principal para que los motores de búsqueda puedan identificar la página con facilidad. El único contenido dentro de la cabecera que se muestra al usuario es la información dentro de la etiqueta del título, que se expresa como el título de la pestaña del navegador.

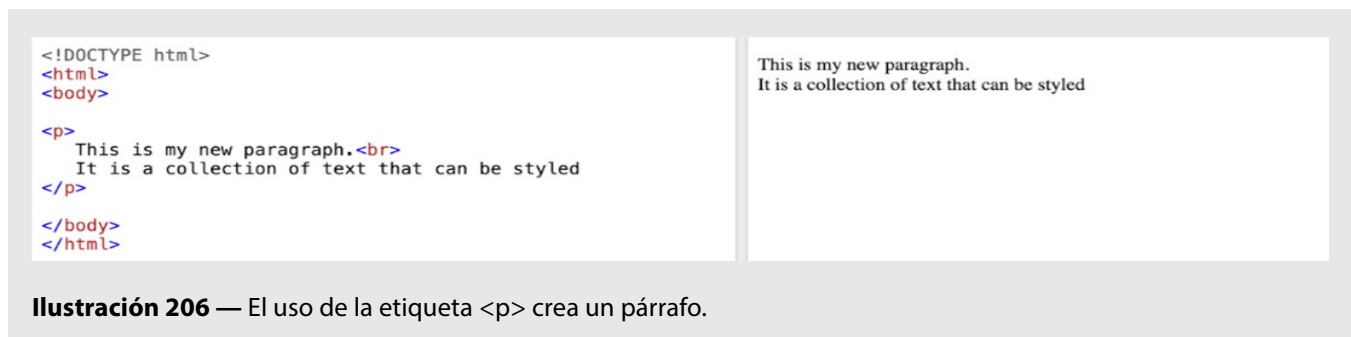
**Etiqueta del cuerpo (body):** Tiene todo el contenido que se muestra al usuario. Esto incluye texto, imágenes, enlaces y otros elementos HTML. Cuando construyas tu página HTML, pasarás la mayor parte del tiempo en esa parte. Esta es también la sección que más analizamos en este capítulo.

## Elementos en HTML

Como la mayoría de los documentos electrónicos, las páginas web tienen texto, imágenes, tablas y enlaces a otras páginas. El texto puede estar en negrita o cursiva y utilizar diferentes tamaños de fuente. Se muestran ideas como viñetas o listas numeradas. Existen formularios con diferentes elementos de entrada para completar. Es posible representar cada uno de estos elementos con etiquetas HTML y crear una página HTML compleja que solicite a los usuarios que ingresen datos y luego los envíen a un servidor para realizar más acciones.

Aquí hay algunas etiquetas esenciales para comenzar.

- La **etiqueta de párrafo** <p> se utiliza para definir un bloque de texto como un párrafo. El texto dentro de la etiqueta <p> se muestra como un párrafo separado, con un salto de línea antes y después (**Ilustración 206**). El texto dentro de la etiqueta de párrafo puede incluir cualquier tipo de contenido como texto, imágenes, enlaces u otros elementos HTML.



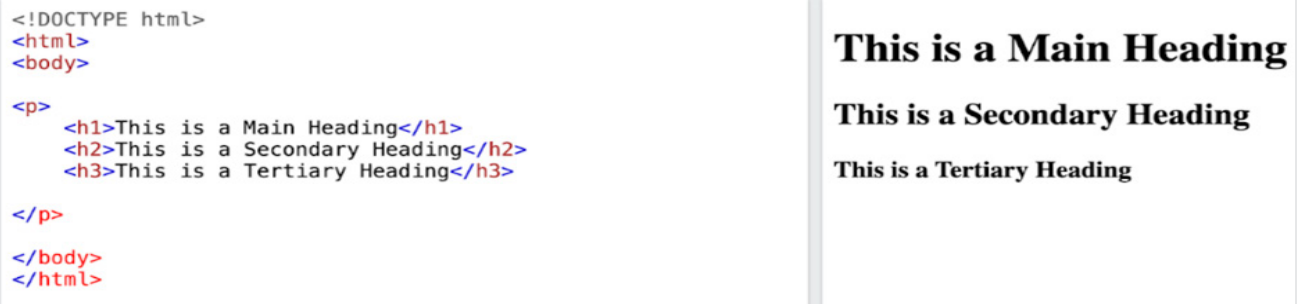
**Ilustración 206** — El uso de la etiqueta <p> crea un párrafo.



Al crear un párrafo, puedes utilizar atributos para especificar qué tipo de párrafo deseas. Para crear un párrafo con texto en color rojo, prueba esto:

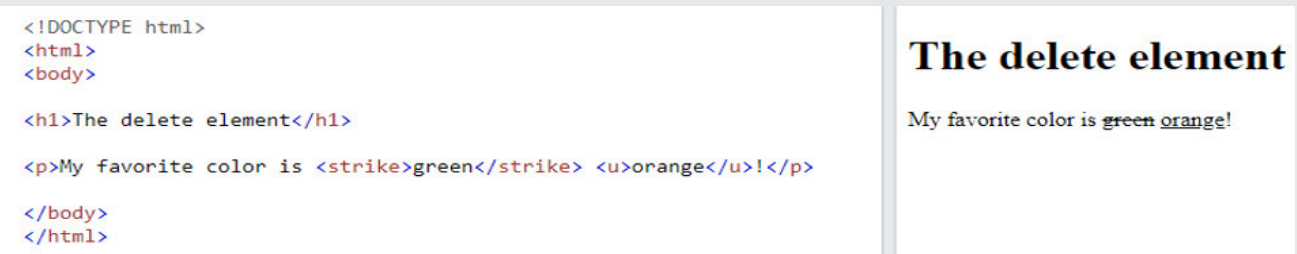
```
<p style="color:rojo;">Este es un párrafo rojo.</p>
```

- **Etiquetas de encabezado:** Son el mecanismo esencial para crear estructura dentro de las páginas. HTML admite seis niveles de encabezados, que van desde `<h1>` a `<h6>`.<sup>168</sup> La etiqueta `<h1>` representa el encabezado principal de una página y normalmente se usa para el título de la página, mientras que los otros encabezados `<h2>` a `<h6>` se utilizan para subtítulos. Es una buena práctica no saltarse los niveles de encabezado, es decir, las subsecciones de `<h1>` deben usar `<h2>`, no `<h3>` o `<h4>`, etc. La [Ilustración 207](#) muestra un ejemplo.



**Ilustración 207** — Las etiquetas de encabezado son útiles para separar títulos y subtítulos.

- La **etiqueta en negrita** `<b>` se utiliza para poner el texto en negrita.
- La **etiqueta de subrayado** `<u>` se emplea para subrayar texto.
- La **etiqueta de tachado** `<del>` o `<strike>` sirve para representar visualmente un texto como tachado ([Ilustración 208](#)).



**Ilustración 208** — Las etiquetas tachadas son útiles para visualizar textos que ya no son relevantes.

- La **etiqueta subíndice** `<sub>` y la **etiqueta superíndice** `<sup>`, como sus nombres lo indican, se utilizan para mostrar texto como subíndice o superíndice ([Ilustración 209](#)). Estas etiquetas pueden resultar útiles para mostrar fórmulas científicas.

168 Los encabezados tienen otro uso: indicar la organización de la página para las tecnologías de asistencia. Puedes leer sobre esto en <https://www.w3.org/WAI/tutorials/page-structure/headings/> (consultado en enero del 2024).

```
<!DOCTYPE html>
<html>
<body>

<p>
  <p>This is text with a subscript: H<sub>2</sub></p>
</p>

</body>
</html>
```

This is text with a subscript: H<sub>2</sub>O

**Ilustración 209** — Las etiquetas de subíndice son útiles cuando se muestran fórmulas químicas.

## Agregar enlaces

Los hipervínculos son la característica distintiva única de HTML. Permiten hacer clic en un enlace de una página e inmediatamente navegar a otra página que podría estar alojada en cualquier parte del mundo. La `<a>` significa **etiqueta de anclaje** y permite agregar hipervínculos a una página HTML.

La etiqueta `<a>` tiene un atributo obligatorio para especificar la página de destino. La página de destino se especifica mediante el atributo `href`, proporcionando la URL de la página o la fuente que se desea vincular. Por ejemplo, para crear un enlace a la página de inicio del sitio web del *New York Times* se utiliza el código que se muestra en la **Ilustración 210**.

```
<!DOCTYPE html>
<html>
<body>

<p>
  Clicking on
  <a href="http://www.nytimes.com">The New York Times </a>
  will take you to the news paper's website.
</p>

</body>
</html>
```

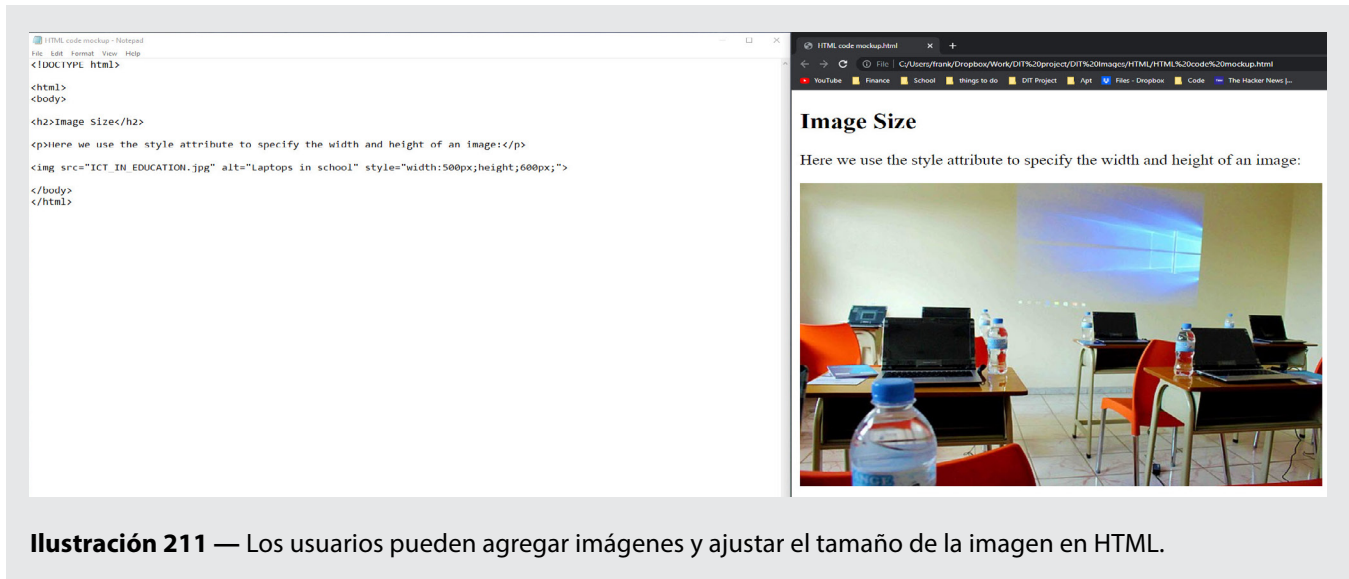
Clicking on [The New York Times](http://www.nytimes.com) will take you to the news paper's website.

**Ilustración 210** — Una etiqueta de anclaje permite a los usuarios vincular direcciones web.

Cuando este código se representa en un navegador web, el usuario puede hacer clic en “*The New York Times*” para ir al sitio web.

## Agregar imágenes

Las imágenes se agregan a una página HTML con la **etiqueta de imagen** `<img>`. Al igual que la etiqueta `<a>`, la etiqueta `<img>` también tiene un atributo obligatorio, que especifica el archivo de imagen que se mostrará. Se debe contar con el atributo de la fuente (`src`) para proporcionar la ubicación de la imagen; de ese modo el navegador puede obtener la imagen y mostrarla en la página web. Es opcional tener otros atributos como parámetros de ancho y alto para indicarle al navegador qué tan grande debe ser la imagen en la página (**Ilustración 211**).



Las imágenes llaman la atención de los visitantes. Por eso, los logotipos, las imágenes de fondo y los anuncios de las empresas están diseñados con cuidado y buen gusto para atraer la atención de los visitantes y transmitirles el mensaje correcto sobre el sitio.

## Crear listas

Para crear listas en HTML, como por ejemplo una lista de cosas para comprar, lugares para visitar o personas para invitar, HTML ofrece dos opciones: listas con viñetas y listas numeradas. Tiene la etiqueta `<ul>` para listas con viñetas que no requieren un orden preciso y la etiqueta `<ol>` para listas numeradas u ordenadas. Se pueden indicar los elementos de la lista con la etiqueta `<li>` y anidar listas dentro de las listas (**Ilustración 212**).

```

<!DOCTYPE html>
<html>
<body>

<h2>A Nested List</h2>
<p>Movies to watch with family</p>

<ul>

<li>Watch with Family
  <ul>
    <li>Gray Hound</li>
    <li>Saving Private Ryan</li>
  </ul>
</li>
<li>Watch with Friends
  <ul>
    <li>Lord of the rings</li>
    <li>Game of Thrones</li>
  </ul>
</li>
<li>Watch with Friends
  <ul>
    <li>13 Assassins</li>
    <li>Zombieland</li>
  </ul>
</li>

</ul>

</body>
</html>

```

## A Nested List

### Movies to watch with family

- Watch with Family
  - Gray Hound
  - Saving Private Ryan
- Watch with Friends
  - Lord of the rings
  - Game of Thrones
- Watch with Friends
  - 13 Assassins
  - Zombieland

**Ilustración 212** — Las listas anidadas pueden ser numeradas o con viñetas en HTML y facilitan a los usuarios la identificación de secciones.

### Crear tablas

Para mostrar datos estructurados como listas de precios, membresías de clubes, inscripciones en clases, etc., HTML incluye **tablas** que se crean usando la etiqueta `<table>`. Las tablas tienen algunos elementos adicionales necesarios para especificar filas y celdas. Las etiquetas `<tr>` y `<td>` definen las filas y celdas de la tabla respectivamente. También se utiliza la etiqueta `<th>` para definir los encabezados de la tabla, que normalmente están en negrita y centrados. Una `<table>` tendrá una o varias etiquetas `<tr>` para indicar las filas de la tabla. Cada `<tr>` a su vez incluye muchas etiquetas `<td>`, para denotar las celdas dentro de la fila. La etiqueta `<td>` representa una celda y comprende el contenido de la celda ([Ilustración 213](#)).

```

<!DOCTYPE html>
<html>
<style>
table, th, td {
border:1px solid black;
}
</style>
<body>
<table style="width:100%">
<tr>
<th>Rank</th>
<th>Name</th>
<th>Points</th>
<th>Team</th>
</tr>
<tr>
<td>1</td>
<td>Domestic</td>
<td>88,110</td>
<td>dcode</td>
</tr>
<tr>
<td>2</td>
<td>Sally</td>
<td>72,400</td>
<td>Students</td>
</tr>
<tr>
<td>3</td>
<td>Nick</td>
<td>52,300</td>
<td>dcode</td>
</tr>
</table>
</body>
</html>

```

Rank	Name	Points	Team
1	Domestic	88,110	dcode
2	Sally	72,400	Students
3	Nick	52,300	dcode

```

<!DOCTYPE html>
<html>
<head>
</head>
<body>
<table>
<tr>
<th>Rank</th>
<th>Name</th>
<th>Points</th>
<th>Team</th>
</tr>
<tr>
<td>1</td>
<td>Sally</td>
<td>72,400</td>
<td>dcode</td>
</tr>
<tr>
<td>2</td>
<td>Nick</td>
<td>52,300</td>
<td>amped</td>
</tr>
</table>
</body>
</html>

```

Rank	Name	Points	Team
1	Sally	72,400	dcode
2	Nick	52,300	amped

**Ilustración 213** — Los usuarios pueden crear tablas en HTML para organizar datos.

## Crear formularios

Los formularios son el mecanismo principal para obtener entradas de los usuarios en las páginas web. Crear formularios en páginas web es importante porque permite a los usuarios interactuar con la web y enviar información para su procesamiento. Casi todas tus páginas web favoritas tienen formularios donde se pueden ingresar datos. A través de los formularios, los usuarios pueden ingresar texto, números, fechas y cargar archivos. También se puede presionar botones, marcar casillas de verificación y elegir botones de opción. La página de inicio de Google.com, probablemente la más famosa del mundo, es sólo un formulario simple con un campo de entrada donde se ingresa la

consulta de búsqueda. La página es esencialmente un formulario con un elemento de entrada de texto y dos botones ([Ilustración 214](#)).<sup>169</sup>

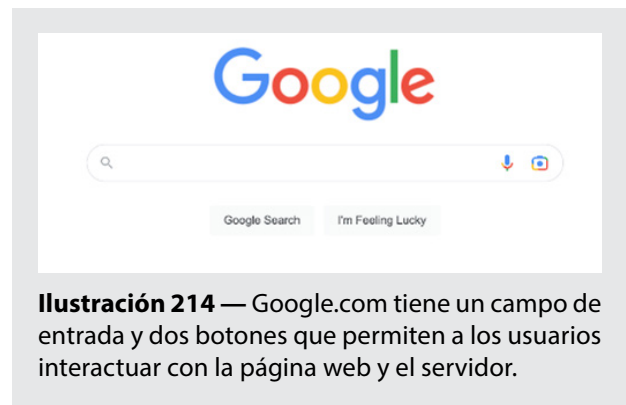
Los formularios se crean utilizando código HTML. Más adelante en este capítulo, veremos cómo se les puede diseñar y personalizar utilizando Hojas de estilo en cascada CSS (la sigla en inglés de *Cascading Style Sheets*). Cuando un usuario envía un formulario en una página HTML, los datos se mandan a un servidor para su procesamiento o almacenamiento en una base de datos. Por ejemplo,

cuando se busca “Películas para ver” en Google.com, la entrada en el formulario se toma y se envía a un servidor de Google. Luego, el servidor responde con la respuesta a la consulta.

Los formularios se utilizan en una variedad de contextos. Los más comunes son: de inicio de sesión y registro, de búsqueda, de contacto, de comentarios y de pago de comercio electrónico. Son una parte esencial del desarrollo web y permiten que las páginas web recopilen datos, proporcionen una experiencia de usuario personalizada y realicen diversas acciones basadas en las aportaciones del usuario.

Los formularios se crean usando la etiqueta `<form>`. Cada formulario también tiene múltiples etiquetas `<input>` para especificar el tipo de entrada que desea el usuario. Estos elementos `<input>` pueden ser texto, número, archivo o imagen. Cada uno de ellos se puede especificar eligiendo el valor correcto para el atributo “tipo” de la etiqueta `<input>`. Los datos recopilados en un formulario se envían al servidor para su procesamiento utilizando el atributo de acción de la [etiqueta del formulario](#).<sup>170</sup>

En el ejemplo de la [Ilustración 215](#) tenemos un formulario simple para recopilar el nombre y apellido del usuario. Cuando se presiona el botón “Enviar” después de ingresar el nombre y apellido, los datos del formulario se enviarán al procesador ubicado en `server_submit.php`. El procesador responderá después de procesar el nombre, por ejemplo guardándolo en la base de datos y personalizando la respuesta con el nombre, etc.<sup>171</sup>



**Ilustración 214** — Google.com tiene un campo de entrada y dos botones que permiten a los usuarios interactuar con la página web y el servidor.

---

169 El botón “Voy a tener suerte” en la página de inicio de Google tiene una historia interesante. La mayoría de los usuarios no lo utilizan, pero es uno de los favoritos de los fundadores. Se puede leer sobre esto en <https://www.howtogeek.com/847170/googles-im-feeling-lucky-explained/> (consultado en enero del 2024), y <https://content.time.com/time/business/article/0,8599,685801,00.html> (consultado en enero del 2024).

170 Los formularios generalmente utilizan la capacidad de HTTP, sigla en inglés de *Hypertext Transfer Protocol* (Protocolo de Transferencia de Hipertexto) para cargar datos desde el navegador al servidor. Esto se especifica utilizando el atributo del método `POST` de la etiqueta `FORM`. Ignoramos este detalle en este libro.

171 Los identificadores (ID único) de los elementos que se ven en el formulario ayudan al procesador a asociar las entradas con los campos. Por lo tanto, estos identificadores deben ser únicos para que los formularios funcionen como se espera.

```

<!DOCTYPE html>
<html>
<body>

<h2>HTML Forms</h2>

<form action="/server_submit.php">
  <label for="firstname">First name:</label><br>
  <input type="text" id="fname" name="firstname" value="John"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lastname" value="Doe"><br><br>
  <input type="submit" value="Submit">
</form>

<p>If you click the "Submit" button, the form-data will be sent to a
page called "/server_submit.php".</p>

</body>
</html>

```

### HTML Forms

First name:  
John

Last name:  
Doe

If you click the "Submit" button, the form-data will be sent to a page called "/server\_submit.php"

**Ilustración 215** — Se pueden crear múltiples formularios en HTML para recopilar diferentes tipos de entradas.

Para crear un elemento de entrada de varias líneas se utiliza la etiqueta "área de texto (*textarea*)". También se especifica el ancho y el alto que debe tener este cuadro de texto utilizando los atributos "filas" y "columnas".

Para crear un menú desplegable que permita a los usuarios seleccionar de una lista se utiliza la etiqueta de selección ([Ilustración 216](#)).

```

<!DOCTYPE html>
<html>
<body>

<h2>The Select (Dropdown) Element</h2>

<p>The select elemnt defines a drop-down list:</p>

<form action="/action_page.php
<label for="cars">Choose a car:</label>
<select id="cars" name="cars">
  <option value="gm">GM</options>
  <option value="chevy">Chevy</options>
  <option value="ford">Ford</options>
  <option value="chrysler">Chrysler</options>
</select> <br><br>
<input type="submit">
</form>
</body>
</html>

```

### The Select (Dropdown) Element

The select elemnt defines a drop-down list:

Choose a car:

**Ilustración 216** — Usando la etiqueta de selección, se crean menús desplegables en HTML.

Hay tipos de entrada de formulario adicionales para manejar escenarios de entrada comunes. Un botón de opción es excelente si se desea brindarles a los usuarios múltiples opciones pero se quiere que elijan solo una. El concepto de botón de radio se remonta a las radios que tenían botones mecánicos para seleccionar una estación. Sólo se podía seleccionar un botón a la vez. Si se presiona otro botón, el botón presionado con anterioridad se saldrá automáticamente. Este concepto se ha trasladado a los formularios HTML y dichos selectores se denominan botones de opción. Entonces, aunque se vean muchas opciones, al seleccionar una nueva opción se anulará la selección anterior. La [Ilustración 217](#) muestra un ejemplo.







```

<!DOCTYPE html>
<html>
<body>
<h2>Date Field</h2>
<p>The <strong>input type="Date"</strong> is used for input fields that should contain a date.</p>
<form action="/action_page.php"
<label for="Birthday">Birthday:</label><br>
<input type="date" id="birthday" name="birthday">
<input type="submit" value="Submit">
</form>
<p>Note: <strong>input type="date"</strong> is not supported in Internet Explorer 11 or prior, Safari 14.1.</p>
</body>
</html>

```

## Date Field

The **input type="Date"** is used for input fields that should contain a date.

Birthday:

Note: **input type="date"** is not supported in Internet Explorer 11 or prior, Safari 14.1

**Ilustración 219** — Los campos de fecha incluyen entradas para el día, mes y año.

## Hojas de estilo en cascada (CSS)

El HTML que hemos creado hasta ahora es funcional, pero insulso. Ahora veamos **CSS**, que es la forma estándar de hacer que las páginas web se destaquen con color y diseño. CSS significa **Hojas de estilo en cascada** y se utiliza para describir cómo se mostrarán los elementos HTML en la pantalla, el papel, los teléfonos, los iPads o en otros medios. El diseño de CSS ahorra tiempo al programador al permitir controlar el diseño de varias páginas web a la vez. En algunos ejemplos hemos especificado el alto y el ancho de la imagen. Observa la **Ilustración 211**. Si bien esto fue conveniente para el ejemplo, imagina tratar de decidir cómo debe verse cada elemento en varias páginas de un sitio web. Cuando agregas etiquetas como los atributos `<font>` y `<color>` a HTML, tendrás una pesadilla al intentar recordar el color y otros atributos de otros elementos similares en la página. Tu página web puede terminar pareciendo inconsistente si no tienes cuidado. Además, el archivo HTML será enorme y desordenado. Se pone peor. Si decides cambiar los colores o estilos de fuente en tu sitio, deberás actualizar la apariencia de cada uno de esos elementos en tu página.

Para evitar estos problemas, la mejor práctica para los programadores es definir todos sus estilos en un archivo CSS que luego vinculan a sus documentos HTML. De esta manera, cada botón "Enviar" en todas las páginas web del sitio utilizará el mismo tamaño y color, y cada encabezado tendrá el mismo tamaño y estilo de fuente.

Para ver cómo funciona CSS, comencemos intentando agregar algunos efectos de estilo básicos a nuestro documento HTML dentro de la etiqueta de estilo, como se muestra en la **Ilustración 220**.

```

<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: lightblue;
}
h1 {
  color: white;
  text-align: center;
}
p {
  font-family: verdana;
  font-size: 20px;
}
</style>
</head>
<body>

<h1>My First CSS Example</h1>
<p>This is a paragraph.</p>

</body>
</html>

```

## My First CSS Example

This is a paragraph.

**Ilustración 220** — La utilización del estilo en HTML permite a los usuarios personalizar las páginas web. Aquí está el HTML (arriba) y su resultado (abajo).

El W3 School es un recurso excelente para aprender sobre estilos.<sup>172</sup> Puedes modificar aspectos sobre la apariencia o el ambiente de tu página web más de lo que imaginas. Una vez que te sientas más cómodo con el CSS, el próximo paso para mejorar la productividad como programador de web es crear un archivo CSS separado (Ilustración 221) y luego vincularlo a tu archivo HTML como se ilustra en la línea 4 de la Ilustración 222.

```
body {  
  background-color: lightblue;  
}  
  
h1 {  
  color: navy;  
  margin-left: 20px;  
}
```

**Ilustración 221** — Los archivos CSS incluyen estilos y atributos que afectan los elementos normales HTML

```
<!DOCTYPE html>  
<html>  
<head>  
<link rel="stylesheet" href="mystyle.css">  
</head>  
<body>  
  
<h1>This is a heading</h1>  
<p>This is a paragraph.</p>  
  
</body>  
</html>
```

**Ilustración 222** — Se puede vincular hojas de estilo al HTML para reducir el desorden en el texto del HTML.

## Diseño de sitios web y el guion gráfico

Los sitios web son colecciones de páginas web individuales que cumplen un propósito específico. Por ejemplo, el sitio de tu distrito escolar, digamos en el condado de Hillsborough, <https://www.hillsboroughschools.org/>, o una escuela, como la secundaria King, <https://kingib.org/> tienen varias páginas que ofrecen toda la información y la interacción que tanto los estudiantes como sus padres esperan de la escuela o del distrito escolar.

Cuando visitas un sitio (por ejemplo, [www.google.com](http://www.google.com)), hay varios servidores (que, además de ser dueños, proveen el mantenimiento) esperando a que los contactes. Para darle apoyo a todos los usuarios que necesitan de sus servicios, los sitios web tienen una colección de páginas web conectadas entre sí a través de hipervínculos. De manera que cuando vas de una página de Google a otra tienes una experiencia sin interrupciones.

Por ejemplo, cuando buscas un libro en Google, obtienes un conjunto de resultados. Si haces clic en un anuncio patrocinado por Google después de revisar los resultados, se abre otra página con otro. Todo el flujo está gestionado por múltiples páginas de Google y programas de aplicaciones que manejan las solicitudes de estos usuarios.

Para lograr todo esto, los sitios deben de ser coherentes, fáciles de navegar y cumplir con lo que se propone. La misma necesidad de coherencia y facilidad de uso se aplica a sitios mucho más grandes cuando escribes un blog, vendes tus productos, o promocionas tu compañía. Construir un sitio web es un ejercicio intenso en término de recursos. Hay que planear, construir y lanzar tu sitio. De la misma

---

172 Puedes encontrar un tutorial para W3Schools y CSS en <https://www.w3schools.com/css/default.asp> (consultado en enero del 2024)

manera que no empezarías a construir una casa sin un plano, no querrás construir un sitio sobre la marcha. Por lo tanto, una vez que aprendes los elementos de las páginas web incluyendo HTML y CSS y estás listo para construir un sitio, es una buena idea empezar con un **guion gráfico**.

Los guiones gráficos son representaciones visuales de proyectos de creación.<sup>173</sup> Cuando se usan en un sitio web, los guiones gráficos ayudan al equipo que lo desarrolla a identificar a todos los usuarios del sitio y a las interacciones que se espera que tengan.<sup>174</sup> Una vez que los guiones se han desarrollado, normalmente se comunican los conceptos usando un *wireframe*, que son ilustraciones de una página web enfocada en la distribución de espacio para los elementos de la página y el comportamiento planeado.<sup>175</sup> Los wireframes están destinados a ser simples y descartables y típicamente no incluyen estilos, colores ni gráficos.

Los wireframes, también conocidos como esquema de página, y los guiones gráficos son herramientas importantes en el proceso de diseño y desarrollo de los sitios web pero sirven un propósito diferente y se usan en contextos diferentes. Los guiones gráficos se dedican a contar una historia y sus componentes visuales y los wireframes se enfocan en los aspectos de interfaz e interacción del sitio web. Para proyectos a gran escala, debes de hacer primero guiones gráficos antes de pasar a usar las herramientas de wireframing.

Mientras el guion gráfico surge del cine y el drama, el concepto ha encontrado un nuevo hogar en el desarrollo de los sitios web. Es útil para entender el panorama y el flujo de alto nivel de las páginas web y para visualizar la experiencia del usuario. Una vez que tú y tu equipo estén de acuerdo acerca del flujo, entonces pueden pasar al diseño en wireframing.

El wireframing bosqueja la estructura y el contenido del sitio web sin incluir elementos de diseño como colores, topografía o imágenes. Los wireframe se crean durante la fase de planificación para ayudar a los diseñadores y programadores a identificar y atender problemas potenciales del diseño y su utilidad.

Típicamente, los wireframing usan cuadros y líneas para representar el lugar donde se ubicarán los distintos elementos en la página, como encabezamientos, menús de navegación, secciones de contenido y llamadas de acción. Balsamiq<sup>176</sup> es un programa de wireframing muy conocido que permite hacer el esquema de tus páginas web y luego compartirlo con tu equipo y tus clientes para que te den su opinión antes de invertir esfuerzo para crear el sitio web (**Ilustración 223**). Las herramientas de wireframing como Balsamiq ofrecen plantillas para recoger, arrastrar y soltar diferentes elementos

---

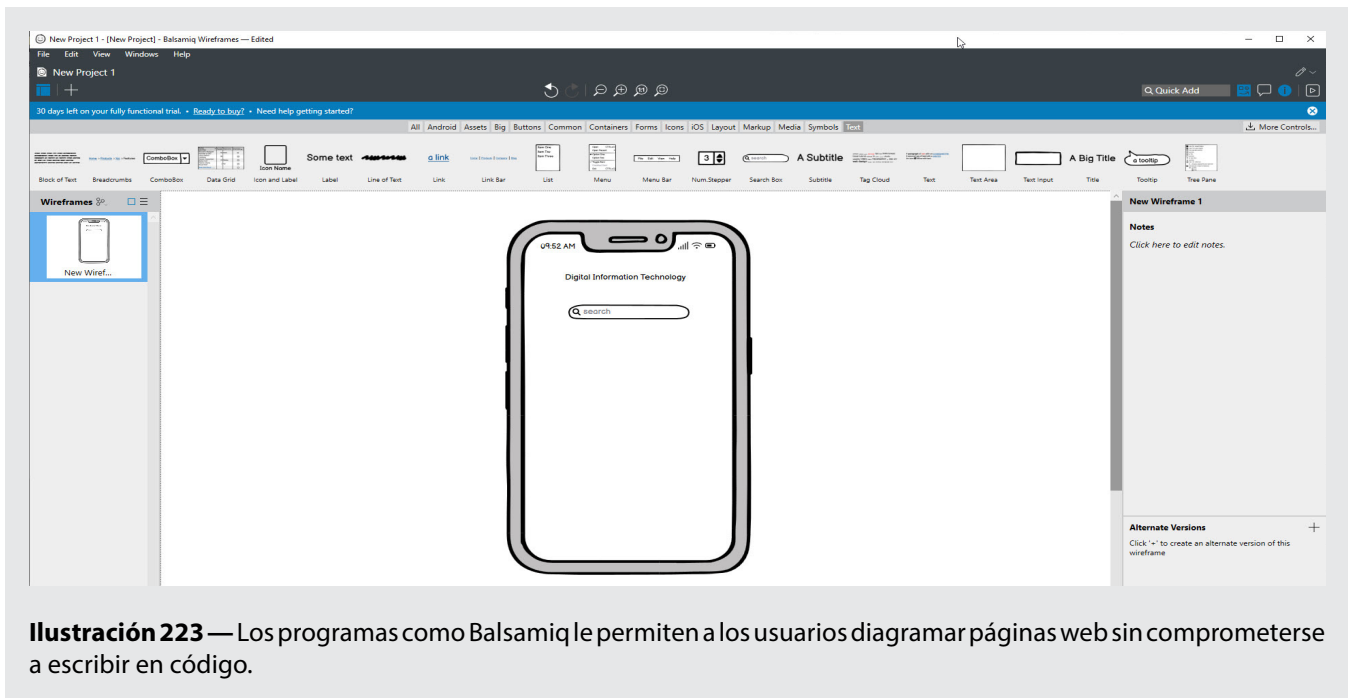
173 Puedes leer acerca de los orígenes de los guiones gráficos y de cómo Walt Disney y su equipo crearon el concepto en 1933 para expresar la “Historia de los tres cerditos” en <https://www.instructionaldesign.org/concepts/storyboarding/> (consultado en enero del 2024)

174 Para una buena descripción del guion gráfico y su uso en el desarrollo de páginas web, veáse <https://www.capterra.com/resources/what-is-a-storyboard-in-web-design/> (consultado en enero del 2024)

175 El gobierno federal de los Estados Unidos tiene un sitio bajo la página de “usability” con una buena introducción para wireframing. [www.usability.gov/how-to-and-tools/methods/wireframing.html](http://www.usability.gov/how-to-and-tools/methods/wireframing.html) (consultado en enero del 2024)

176 <https://balsamiq.com/> tiene un tutorial excelente sobre cómo hacer esquemas de páginas y diseño de interfaz de usuario en <https://balsamiq.com/learn/courses/intro-to-ui-design/> (consultado en enero del 2024).

de página web y ponerlos en la pantalla para poder desarrollar el esquema rápidamente. Puedes usar wireframes para diseñar páginas web para abrir en un navegador o en aplicaciones para móviles como IOS o Androide. Construir wireframes para refinar conceptos para una aplicación es una manera excelente de probar nuevas ideas con poco esfuerzo y sin costo alguno.



**Ilustración 223** — Los programas como Balsamiq le permiten a los usuarios diagramar páginas web sin comprometerse a escribir en código.

Figma es otra herramienta de wireframing de gran popularidad; fue adquirida por Adobe en septiembre del 2022 por 20 billones de dólares.<sup>177</sup> Figma no solo permite hacer el esquema de una página y colaborar con otros colegas, sino que también convierte tu diseño de wireframe en un sitio Web en HTML en funcionamiento; este sitio puede luego ser alojado en tu hospedaje de sitio web.

## La estructura de un sitio web

Una vez que aprendas HTML para construir páginas web y crear guiones gráficos y wireframes para tus ideas, quizás te interese crear tu sitio web para que pueda verlo cualquier persona en el mundo. Pasemos ahora a un panorama de los elementos con los que cuenta un sitio web en funcionamiento.

Para crear un sitio web, el administrador del sitio (o tú) va a necesitar tener un **servidor web** (p. ej. el servidor Apache)<sup>178</sup> para alojarlo.<sup>179</sup> Necesitarás guardar tus páginas HTML y otros archivos de apoyo

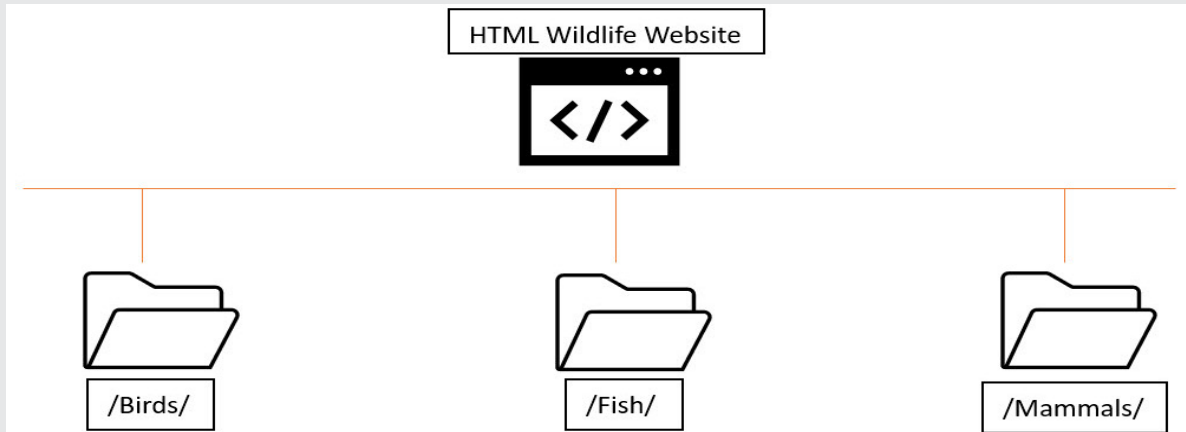
177 Puedes leer más acerca de la noticia en <https://news.adobe.com/news/news-details/2022/Adobe-to-Acquire-Figma/default.aspx> (consultado en enero del 2024).

178 El servidor web Apache, el cual tiene su origen en el primer servidor web que construyó Tim Berners Lee en CERN, es extremadamente robusto y fácil de instalar. Si tienes acceso a una computadora personal instalar el servidor de web Apache y familiarizarte con su uso sería para ti una experiencia enriquecedora. Para saber más sobre Apache y cómo descargarlo, ir a <https://httpd.apache.org/> (consultado en enero del 2024).

179 Puedes leer más detalles en línea, incluyendo la página GoDaddy's, <https://www.godaddy.com/garage/how-to-build-a-website-with-html-and-css/> (consultado en enero del 2024).

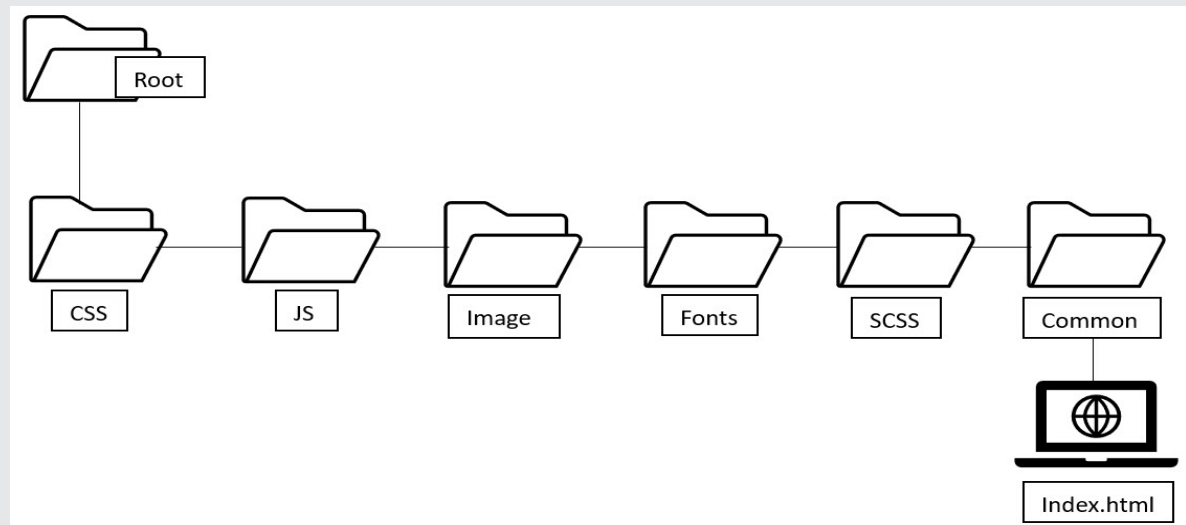
organizados en una estructura de carpetas en el servidor web. Discutamos ahora la estructura básica de un sitio web de manera que puedas entender cómo todos los elementos del sitio están integrados.

Es una buena idea incluir la estructura de tipo carpeta de tu sitio en tu wireframe; así será más fácil ver cómo estará organizado en tu servidor. La [Ilustración 224](#) muestra un ejemplo de una jerarquía de carpetas de un sitio web y cómo se enlazan con diferentes secciones.



**Ilustración 224** — Los wireframes permiten visualizar proyectos, dependencias y estructuras en la fase de planeamiento.

Es útil mantener una organización consistente para los archivos de cada carpeta. La [Ilustración 225](#) muestra una estructura que se usa mucho para organizar archivos en una carpeta. El punto de partida de tu aplicación o sitio web típicamente se llama la carpeta “Raíz” de tu sitio. Un archivo llamado “index.html” aparece automáticamente si alguien visita tu sitio y no especifica el nombre del archivo.<sup>180</sup>

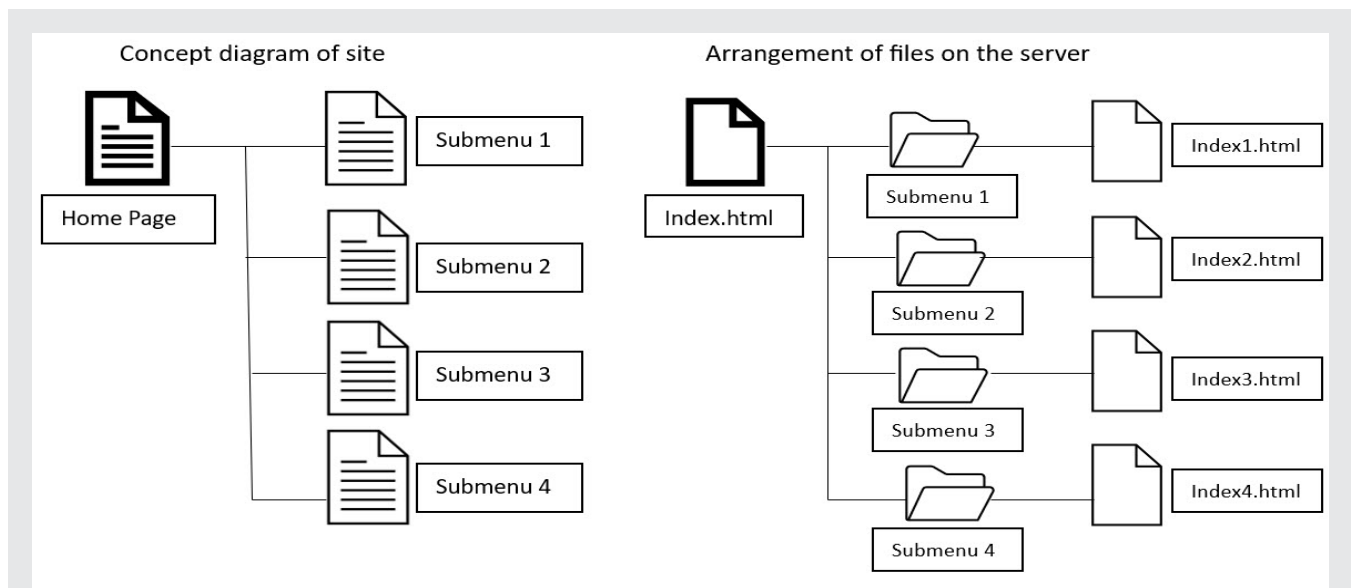


**Ilustración 225** — La estructura tipo carpeta de los servidores de web varía dependiendo del contenido y propósito del sitio web.

180 Para la discusión de la convención del nombre de este archivo, puedes ir a StackOverflow <https://stackoverflow.com/questions/32408259/why-do-people-name-their-files-index-html> (consultado en enero del 2024).

Organizar las carpetas por Imágenes, Fuentes, Bibliotecas comunes, JavaScript y Hojas de estilo (Ilustración 225) facilita el manejo y mantenimiento de los archivos, y también hace más fácil que los buscadores y los usuarios puedan navegar por el sitio. Además, muchos sitios web usan un Sistema de gestión de contenido (Content Management System, CMS, por su sigla en inglés) que provee una interfaz gráfica para organizar y manejar los archivos y contenidos de sitios web.

Puedes tener todas tus páginas HTML en carpetas separadas o dividir tu sitio web en diferentes áreas operativas y tener una página web por cada tópico dentro de sí. En la Ilustración 226, se puede apreciar cómo los submenús de un sitio web corresponden a una carpeta en el servidor que se llama de manera similar. Por ejemplo, si un usuario visita tu página de inicio, el servidor va a mostrar el archivo de máximo nivel index.html. Sin embargo, si un usuario hace clic en el Submenú 1, entonces lo que se va a mostrar es el Index.html dentro de la carpeta Submenú 1.

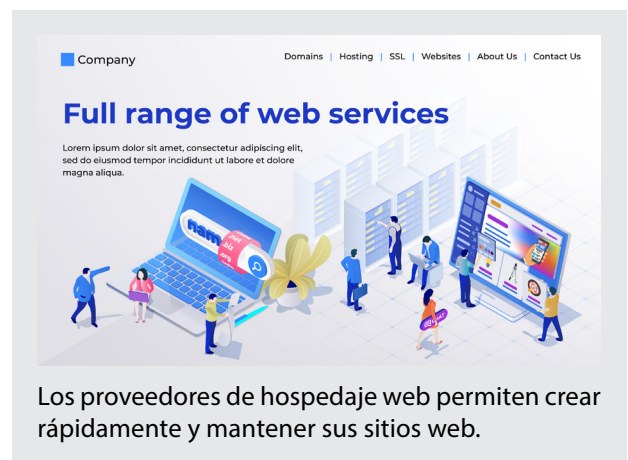


**Ilustración 226** — Los conceptos y diagramas ayudan a los usuarios a desarrollar sus sitios web y contenido.

## Hospedaje web

Hospedar un sitio web significa hacerlo accesible al público en la Internet guardando en un servidor los archivos y datos del sitio. Un servidor web es una computadora conectada a la Internet que está diseñada para almacenar y distribuir páginas web a los usuarios que así lo pidan a través de sus buscadores.

Cuando obtienes un hospedaje web, esencialmente alquilas un espacio en un servidor de una compañía anfitriona o proveedora, lo cual permite guardar tu sitio web en un servidor y hacerlo accesible en la Internet. La compañía que hospeda también provee una serie de servicios tales como mantenimiento del servidor y la seguridad y el apoyo técnico para asegurar que tu sitio esté siempre accesible y opere eficientemente. Se necesita subir los



archivos al servidor y hacer la configuración del sitio de manera que los usuarios puedan encontrar tu sitio en la web.<sup>181</sup>

Una vez que tu página esté alojada, los usuarios pueden tener acceso a ella escribiendo el nombre de tu dominio (p. ej. [www.cyberflorida.org](http://www.cyberflorida.org)) en su buscador. Esta acción envía una solicitud al servidor que aloja tu sitio web, el cual responde enviando las páginas solicitadas al buscador del usuario; luego aparecerán en su pantalla.

### Hospedar desde tu computadora en casa

De hecho, puedes hospedar tu sitio desde la computadora que usas en casa. Para hacerlo, tendrás que conseguir una dirección IP permanente de tu proveedor de Internet, comprar un nombre de dominio (es decir, el nombre de tu sitio web) de un registro DNS y conectar el nombre del sitio a la dirección IP de tu computadora. Además, necesitas asegurarte de lo siguiente: que tu computadora tenga suficiente capacidad para este tipo de tarea, que esté protegida, que esté siempre encendida y que haga una copia de seguridad regularmente.

Como puedes ver, hacer esto desde tu casa añade complejidad y responsabilidad. Por lo tanto, es más popular optar por los servicios de un proveedor de hospedaje de sitios web.

Hay muchos sitios disponibles que hospedan web, cada uno con servicios, precios y capacidades propias. Algunos de los más populares son: GoDaddy,<sup>182</sup> Bluehost, SiteGround, HostGator,<sup>183</sup> DreamHost, A2 Hosting, InMotion Hosting, and Wix.<sup>184</sup> Es importante investigar y comparar las diferentes opciones para encontrar la que mejor sirve a tus necesidades y presupuesto.<sup>185</sup>

---

181 La primera configuración es la adquisición de un nombre para el dominio (como [www.ejemplo.com](http://www.ejemplo.com)) la dirección a tu sitio. Como los proveedores de hospedaje se encargan de este paso no discutimos este aspecto detalladamente en este libro. Para más detalles puedes leer <https://www.hivelocity.net/kb/dns-configuration-everything-you-need-to-know/> (consultado en enero del 2024). Para información acerca de cómo se vería la configuración en el proveedor del servicio de hospedaje, puedes ir a la página de GoDaddy's en <https://www.godaddy.com/garage/configuring-and-working-with-domains-dns/> (consultado en enero del 2024).

182 La información de los planes de GoDaddy's están en <https://www.godaddy.com/hosting/web-hosting> (consultado en enero del 2024).

183 Sus planes aparecen en <https://www.hostgator.com/web-hosting> (consultado en enero del 2024). HostGator se creó en un dormitorio estudiantil de la Florida Atlantic University y por esta razón tenemos preferencia por él.

184 La información sobre sus servicios se encuentra en <https://www.wix.com/> (consultado en enero del 2024).

185 Hay varios recursos en línea que puedes consultar para comparar opciones, p. ej. <https://geekflare.com/best-static-site-hosting-platform/> (consultado en enero del 2024)

## Editores de sitios web

Hay múltiples formas de construir páginas HTML. Una de ellas es codificar desde cero usando los elementos de HTML como `<table>`, `<text>`, `<img>` entre otros. Otra forma es utilizar una herramienta de wireframe como Figma y luego usar la opción de conversión para pasar el wireframe al código HTML.

Otra forma que se usa mucho es el editor WYSIWYG (en español se pronuncia uisiuig). WYSIWYG es un acrónimo para la frase en inglés “lo que ves es lo que obtienes”, *What You See Is What You Get*. En el contexto de editores HTML, un editor WYSIWYG es un programa de software que le permite a los usuarios crear y editar documentos HTML sin tener que saber codificar en HTML. Con un editor WYSIWYG los usuarios arrastran y sueltan elementos de un sitio web, como imágenes y tablas, de manera similar a un lienzo de dibujo, y ven la página que quieren crear como va a aparecer en el buscador sin tener que codificar manualmente el HTML.<sup>186</sup>

Los editores de WYSIWYG HTML hacen que la tecnología web sea accesible para todos. Son populares entre usuarios no-técnicos, como los blogueros, creadores de contenido, y dueños de empresas pequeñas, quienes quizás no tengan el conocimiento técnico para crear páginas web usando códigos HTML. Algunos de los editores WYSIWYG HTML gratuitos<sup>187</sup> que más se usan incluyen Brackets de Adobe,<sup>188</sup> editor de código Phoenix de la comunidad de programadores de Brackets<sup>189</sup> y SeaMonkey de la comunidad Mozilla.<sup>190</sup> También hay editores WYSIWYG comerciales, como por ejemplo Dreamweaver de Adobe.<sup>191</sup>

Los [proveedores de hospedaje web](#) como Wix.com también te dan acceso a generadores y editores de sitios web. Un generador de sitios es típicamente una plataforma WYSIWYG en la cual puedes arrastrar y soltar elementos en una página. Cuando estás listo, puedes hacer clic en el botón de “Publicar” para lanzar tu sitio web sin tener que escribir o editar líneas de código.

Los Sitios de Google ([www.sites.google.com](http://www.sites.google.com)) es otro proveedor de hospedaje para sitios web que tiene su propio generador de páginas web WYSIWYG ([Ilustración 227](#)).

---

186 La página de Wikipedia en inglés para WYSIWYG contiene una simpática historia sobre los editores de WYSIWYG, <https://en.wikipedia.org/wiki/WYSIWYG> (consultado en enero del 2024). En español hay una página de Wikipedia con información muy básica pero útil en <https://es.wikipedia.org/wiki/WYSIWYG> (consultado en septiembre del 2023)

187 Hay otras opciones y reseñas disponibles en línea, incluyendo, por ejemplo un artículo en <https://www.thoughtco.com/best-windows-wysiwyg-editors-3471324> (consultado en enero del 2024).

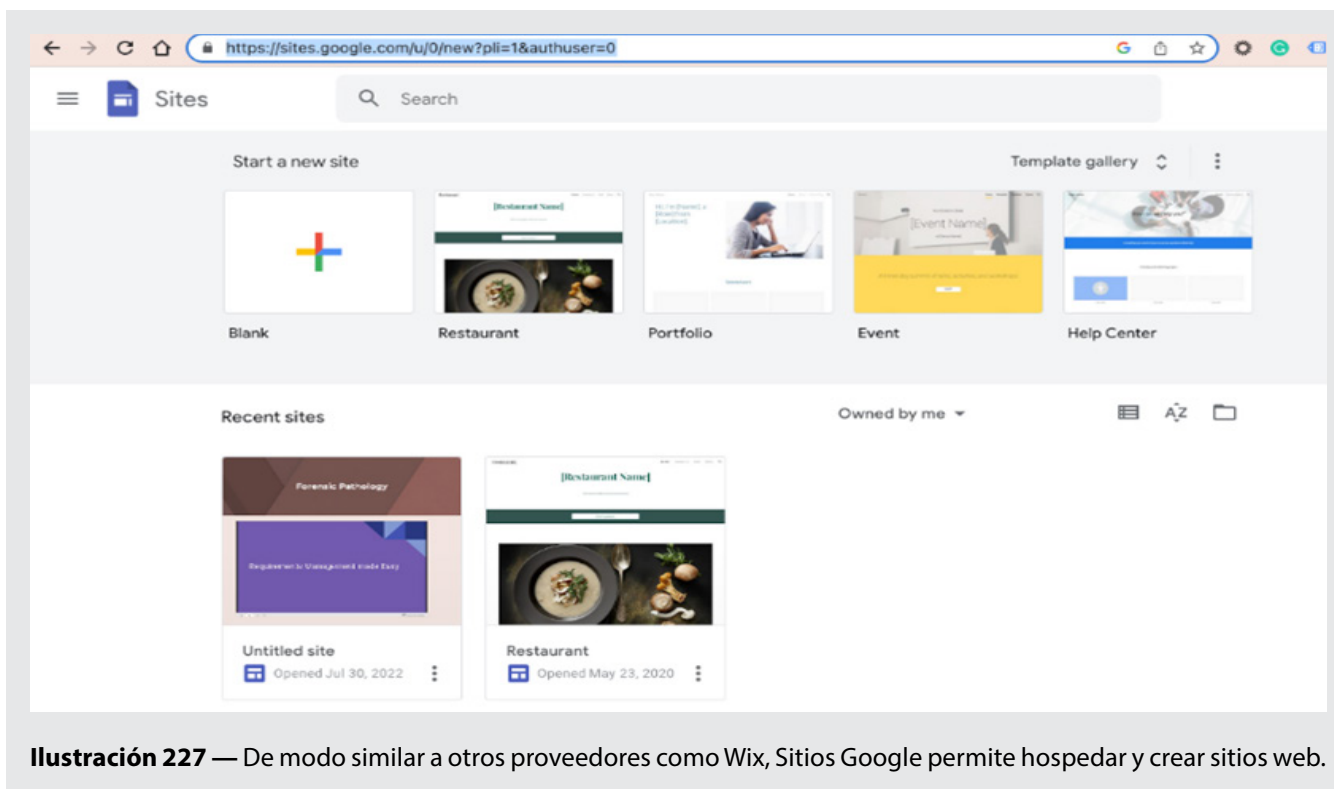
188 Puedes descargar Brackets en <https://brackets.io/> (consultado en enero del 2024).

189 El editor Phoenix está en línea y puedes conseguirlo en <https://phcode.dev/> (consultado en enero del 2024).

190 Puedes descargar SeaMonkey en <https://www.seamonkey-project.org/> (consultado en enero del 2024) SeaMonkey continúa el concepto de una aplicación de Internet “todo en uno” popularizado por el explorador Netscape, la cual los autores de este libro usaban en los primeros años de la Internet.

191 Disponible en <https://www.adobe.com/products/dreamweaver.html> (consultado en enero del 2024).





**Ilustración 227** — De modo similar a otros proveedores como Wix, Sitios Google permite hospedar y crear sitios web.

Las capacidades de los editores HTML evolucionan rápidamente. Mientras los editores comparten la mayoría de las mismas capacidades básicas, también pueden ofrecer otras herramientas dependiendo de las necesidades del público al que está dirigido, estructura de precios, etc. Por ejemplo, ofrece una biblioteca para mantener tus temas más comunes, íconos, un sistema de colaboración (para que puedas obtener la aprobación para tu proyecto antes de que se publique) y un sistema de control de versiones incorporado, de manera que siempre puedas ir a una versión previa. Fíjate en lo que ofrecen los distintos editores y en los precios para que puedas encontrar el que mejor se adecue a tus necesidades.

Aprender a generar páginas y sitios web es una buena manera de entrar en el fascinante mundo de la informática moderna. Puedes continuar tu viaje a través de los siguientes tutoriales disponibles gratuitamente en línea: Mozilla Developer Network (MDN) es un gran recurso para aprender las tecnologías presentadas en este capítulo<sup>192</sup> al igual que W3Schools.

## Extensiones HTML

Concluimos este capítulo presentando otras tecnologías relacionadas que potencian la web y el comercio de hoy. La popularidad de la web ha hecho posible las innovaciones en el diseño de web, llamadas colectivamente DHTML. El concepto de etiquetas o tags también se ha extendido a XML, el cual generaliza la publicación e inclusive permite que las máquinas se intercambien información directamente.

---

192 Para tener acceso a tutoriales para HTML, CSS, JavaScript para programadores y otros de este tipo de tecnología de acuerdo con tu nivel puedes ir a <https://developer.mozilla.org/en-US/docs/Web/Tutorials> (consultado en enero del 2024).

## HTML dinámico (DHTML)

Al comenzar a programar tus propias páginas web, vas a notar que los sitios más populares tienen más capacidades de las que presentamos en este capítulo. Una página que usa un HTML estático puede lucir atractiva con textos en diferentes fuentes, formas y tamaños e imágenes coloridas por todas partes. Sin embargo, en el mundo de los teléfonos móviles esperamos que las páginas que usamos sean además interactivas. Queremos que con un solo clic tengamos acceso rápido a lo que buscamos; que cuando vayamos a un sitio como Amazon podamos añadir inmediatamente los productos al carrito de compra con tan solo un clic. Los programadores utilizan una multiplicidad de tecnologías de web para producir este tipo de interactividad. A dicha colección de tecnologías para crear sitios web interactivos, que incluye HTML, CSS, y JavaScript, la llamamos HTML dinámico, o DHTML, por sus siglas en inglés.<sup>193</sup>

Una forma de entender los distintos tipos de tecnologías DHTML es pensar en cómo cada código respondería a una pregunta diferente. HTML preguntaría “¿Qué mostrar?”, CSS preguntaría “¿Cómo debería lucir lo mostrado?” y JavaScript “¿Cómo debería interactuar con el usuario la página?”

Con DHTML puedes crear páginas web que respondan a acciones del usuario, como hacer clic y desplazar el ratón sin necesidad de que la página tenga que refrescarse o que necesite procesamiento del lado del servidor.

Otro desarrollo reciente, popularizado por los sitios de medios sociales como Facebook y Twitter, es la “Arquitectura de una sola página” o *Single Page Architecture* (sigla en inglés SPA), que es otro acercamiento a la programación en el cual el contenido y la funcionalidad de una aplicación web se contiene en una sola página HTML. En vez de navegar por páginas separadas al hacer clic en un enlace, el usuario se queda en la misma página mientras las secciones de la página se actualizan dinámicamente según la interacción que el usuario tenga con el sitio. La página HTML inicial se carga desde el servidor y luego JavaScript maneja las subsecuentes interacciones con la aplicación. El código JavaScript actualiza las secciones de la página según las interacciones del usuario sin que necesite recargar la página entera, ofreciendo así una experiencia sin interrupciones a los visitantes.<sup>194</sup>

Los SPAs ofrecen varias ventajas para aquellas personas que visitan una página, incluyendo una experiencia más fluida para el usuario, con cargas de páginas más rápidas y desarrollo y mantenimiento más fácil debido a la separación de los códigos entre el lado del cliente y el lado del servidor.<sup>195</sup> Entre los ejemplos más conocidos de SPAs podemos mencionar a Gmail, Facebook y Twitter. Pero su

---

193 JavaScript está más allá del alcance de este libro, pero quienes estén interesados pueden probar el tutorial de JavaScript disponible en Mozilla Developer Network (MDN) citado en la nota anterior.

194 La tecnología para cargar secciones de una página se llama AJAX (por sus siglas en inglés Asynchronous JavaScript and XML). AJAX pide partes de datos al servidor y actualiza pequeñas secciones en la página con los datos sin perturbar la interacción con el usuario. Un ejemplo de cómo se usa AJAX es la forma en que Gmail mueve la vista hasta de una hilera entera cuando se elimina un correo electrónico sin tener que refrescar la página entera. Para saber más acerca del tutorial de W3School AJAX ir a [https://www.w3schools.com/xml/ajax\\_intro.asp](https://www.w3schools.com/xml/ajax_intro.asp) (consultado en enero del 2024)

195 No obstante, SPA también presenta algunos retos, como por ejemplo, dificultades con la optimización de los buscadores de internet (o *Search Engine Optimization*, SEO por sus siglas en inglés), y la necesidad de un cuidadoso manejo de la memoria y el desempeño.

conveniencia es también una fuente de crítica a las aplicaciones de una sola página – los usuarios siguen desplazándose por los SPAs, esperando encontrar novedades, un fenómeno conocido como desplazamiento tonto o “doom-scrolling”.<sup>196</sup>

## Lenguaje de marcado extensible (XML)

HTML está diseñado específicamente para la publicación de páginas web. Por su popularidad se ha creado interés en la idea de estandarizar todas las formas de publicaciones electrónicas, incluyendo documentos y récords. La idea central era que, si los elementos de la información se pudieran estandarizar, las computadoras podrían modificar la información según el propósito con el clic de un botón. Por ejemplo, con tan solo un clic se puede formatear la misma información con hojas de estilo para web y para un impreso, o también guardar como información estructurada en una base de datos relacional para almacenamiento de archivos con otro clic o, finalmente, como información relevante extraída que se reporta al gobierno para propósito de impuestos con otro clic.

El lenguaje de marcado extensible (XML por sus siglas en inglés) es la generalización de etiquetas para publicaciones electrónicas.<sup>197</sup> Detrás de las operaciones de muchas tecnologías de publicaciones electrónicas se encuentra XML. Por ejemplo, la [Ilustración 228](#) muestra el XML creado por Microsoft Word para el título y la oración introductoria de esta sección.<sup>198</sup> Además puedes apreciar los estilos y el contenido para el título y el párrafo.<sup>199</sup>

Como puedes notar en la [Ilustración 228](#), XML codifica los documentos en un formato legible tanto para humanos como para máquinas. XML puede especificar cómo necesita interpretarse cada elemento de un documento electrónico. Esa es información con la cual las computadoras pueden transformar datos fácilmente y con precisión para diferentes usos como publicaciones y registros.

---

196 El “doomscrolling” no es solamente una pérdida de tiempo, sino también y más importante aún, un peligro a la salud. Puedes empezar por leer en español, <https://observatorio.tec.mx/edu-news/doomscrolling/> y en inglés <https://www.webmd.com/balance/what-is-doomscrolling> (consultado en enero del 2024)

197 La página del equipo de trabajo de tecnología es <https://www.w3.org/XML/> (consultado en enero del 2024).

198 Para ver el XML de un documento en Microsoft Word, primero necesitas cambiar la extensión del archivo de “.docx” a “.zip”. Luego extrae el archivo y busca el archivo “document.xml” dentro de la carpeta de Word.

199 La especificación XML para los documentos de Microsoft Office se llama Office Open XML. Puedes leer más información en [https://en.wikipedia.org/wiki/Office\\_Open\\_XML](https://en.wikipedia.org/wiki/Office_Open_XML) (consultado en enero del 2024).

```

▼ <w:p w14:paraId="3DE2CE7B" w14:textId="068E7512" w:rsidR="00880F3C" w:rsidRDefault="00E2563A" w:rsidP="003105C2">
  ▼ <w:pPr>
    <w:pStyle w:val="Heading2"/>
  </w:pPr>
  <w:bookmarkStart w:id="1081" w:name="_Toc132104541"/>
  ▼ <w:r>
    <w:t>Extensible markup language (</w:t>
  </w:r>
  ▼ <w:r w:rsidR="00880F3C">
    <w:t>XML</w:t>
  </w:r>
  ▼ <w:r>
    <w:t></w:t>
  </w:r>
  <w:bookmarkEnd w:id="1081"/>
</w:p>
▼ <w:p w14:paraId="6DD9495F" w14:textId="60E7BD61" w:rsidR="00E2563A" w:rsidRDefault="005D6A3C" w:rsidP="00880F3C">
  ▼ <w:pPr>
    <w:pStyle w:val="BodyText"/>
  </w:pPr>
  ▼ <w:r>
    <w:t>HTML is designed for the specific case of publishing web pages.</w:t>
  </w:r>
  ▼ <w:r w:rsidR="00B9518F">
    <w:t xml:space="preserve"> The popularity of HTML </w:t>
  </w:r>

```

**Ilustración 228** — Las etiquetas XML formatean el texto en un lenguaje que tanto los seres humanos como las computadoras puedan interpretar.

XML es muy útil si la compañía donde trabajas intercambia datos, por ejemplo, con compradores, vendedores, compañías de tarjetas de crédito, bancos y otras organizaciones. El intercambio entre compañías y clientes es un aspecto crucial del comercio en línea, lo cual hace XML un lenguaje clave que debes conocer.

Una característica especial de XML es que le permite a los usuarios crear sus propias etiquetas para representar estructuras específicas de datos e identificar elementos dentro de un documento.

Los estándares XML, que especifican etiquetas para usos precisos, se han desarrollado para la mayoría de los contextos de intercambio de información más importantes. Por ejemplo, el XML para lenguaje de información financiera, o *Business Reporting Language* (XBRL, por su sigla en inglés) permite que las compañías hagan informes del desempeño económico en un formato universal.<sup>200</sup> Desde el 2018, U.S. Securities and Exchange Commission, la principal reguladora de compañías negociadas públicamente en los Estados Unidos, ha requerido que las empresas sometan sus reportes anuales usando el lenguaje XBRL.<sup>201</sup> La [Ilustración 229](#) presenta un buen ejemplo.<sup>202</sup> Una característica clave que puedes observar en el ejemplo es que los números están claramente asociados con los correspondientes significados especificados por los *International Financial Reporting Standards* (IFRS por su sigla en inglés).<sup>203</sup> Por

200 Para más información sobre XBRL en español puedes visitar <https://www.gestiopolis.com/xbrl-estandarizacion-tecnologica-de-datos-financieros/> y en inglés, <https://www.xbrl.org/> (consultado en enero del 2024).

201 Puedes leer más sobre este asunto en <https://www.sec.gov/structureddata/osd-inline-xbrl.html> (consultado en enero del 2024).

202 El ejemplo proviene de <https://www.xbrl.org/ixbrl-samples/gleif-annual-report-2017/gleif-annual-report-2017-viewer.html#f-f6>. Puedes encontrar este ejemplo si buscas “inline xbrl example” (consultado en enero del 2024).

203 La página inicial para los estándares está en <https://www.ifrs.org/> (consultado en enero del 2024).

ejemplo, el XML subyacente comunica a cada tecnología de publicaciones financieras en el mundo que la cantidad resaltada de \$3,404,031 representa gastos salariales para la organización. Esto hace más fácil y preciso que los inversores comparen el desempeño de diferentes organizaciones, y que los reguladores se aseguren de que los números se reportan correctamente. Por último, los visores XML pueden formatear documentos para que sea más legible para los seres humanos, según se aprecia en la [Ilustración 229](#).

**Statement of Comprehensive Income** for the Period from January 1 to December 31, 2017

	Notes	Jan. to Dec. 2017	Jan. to Dec. 2016
		US\$	US\$
Fee revenue	3.1	7,786,002	6,245,640
Wages and salaries		-3,404,031	-2,700,694
Social contributions and expenses for pensions and care		-456,337	-376,780
Personnel expenses	3.2	-3,860,368	-3,077,474
Other operating expenses	3.3	-4,058,493	-3,348,252
Other operating income	3.4	539,527	201,019
Amortization and depreciation expense	4.5/4.6	-214,955	-165,030
<b>Operating surplus</b>		<b>191,713</b>	<b>-144,097</b>
Subsidies and donations	3.5	61,454	158,354

**Fact Properties**

- Concept: (ifrs) Wages and salaries
- Dimensions:
  - Date: 31 Dec 2016 to 30 Dec 2017
  - Fact Value: US \$ 3,404,031
  - Accuracy: 0 (ones)
  - Change: 26.0% increase on 31 Dec 2015 to 30 Dec 2016
  - Entity: [LEI] 506700GE1G29325QX363
  - Concept: ifrs:WagesAndSalaries

**Ilustración 229** — XBRL asiste a los usuarios de empresas a organizar datos financieros.

Una de las características más importantes de XML para informes financieros, comercio electrónico, asistencia médica, y otras aplicaciones industriales es que las computadoras pueden validar datos en relación con un esquema definido o una definición de tipo de documento (*Document Type Definition*, DTD, por su sigla en inglés). Las computadoras pueden repasar cada elemento de un documento y asegurarse de que los datos estén formateados adecuadamente y que satisfagan todas las propiedades necesarias. Por ejemplo, un DTD puede especificar que las ganancias o pérdidas sean en números, o, como en el ejemplo en la [Ilustración 230](#), que una persona deba tener un nombre. Esto ayuda a evitar errores e inconsistencias en los documentos electrónicos.

```
<?xml version="1.0" encoding="UTF-8"?>
<person>
  <name>John Smith</name>
  <age>35</age>
  <address>
    <street>123 Main St</street>
    <city>Anytown</city>
    <state>CA</state>
    <zip>12345</zip>
  </address>
</person>
```

**Ilustración 230** — Las etiquetas XML permiten a los usuarios transferir y validar datos de manera eficiente.

La [Ilustración 230](#) introduce la estructura de los documentos XML. En este ejemplo el documento comienza con una declaración que especifica la versión de XML que se está usando ("1.0") y la codificación de caracteres que se usó para codificar el documento ("UTF-8").<sup>204</sup>

204 Puedes leer acerca del formato de codificación de caracteres o Unicode Transformation Format (UTF por su sigla en inglés) y UTF-8 en <https://blog.hubspot.com/website/what-is-utf-8> (consultado en enero del 2024).

El contenido principal del documento se encuentra en el elemento `<person>`, el cual a su vez contiene varios otros elementos que representan distintos aspectos de la información de la persona. El elemento `<name>` contiene el nombre de la persona, el elemento `<age>` contiene la edad, y el de `<address>` contiene la calle, la ciudad y el código postal. El DTD puede especificar las propiedades de cada una de estas etiquetas, incluyendo si son requeridas u opcionales.

Hay que notar que cada elemento tiene una etiqueta de cierre correspondiente (p. ej. `</name>`), que marca el final del contenido del elemento, igual que HTML. Observa también que los elementos se pueden guardar dentro de otro elemento para crear una estructura jerárquica, igual al elemento `<address>` de este ejemplo.

A continuación, presentamos varias razones porqué el concepto esencial de XML se usa ampliamente:

- XML puede ser leído fácilmente por los seres humanos y puede ser analizado y procesado por computadoras.
- XML no está atado a ninguna plataforma o lenguaje de programación específico, lo que lo hace muy flexible para el intercambio de datos en la Internet.
- XML está diseñado para ser expandible. Los usuarios pueden definir sus propias etiquetas para describir los datos de una manera específica a sus necesidades. Esto permite la creación de estructuras de datos específicos para un caso en particular.
- Los documentos XML se pueden validar usando un esquema definido o una definición de tipo de documento (DTD). Se puede asegurar con facilidad que los datos estén formateados y estructurados adecuadamente para evitar errores e inconsistencias.
- XML se usa para el intercambio de datos entre sistemas y aplicaciones. Esto se debe en parte a su flexibilidad, extensibilidad y su capacidad de representar una amplia gama de tipos de datos y estructuras.
- XML se usa para representar datos, no para formatearlos o para estilo, lo que permite una clara separación entre contenido y presentación y hace más fácil reusar los datos en diferentes contextos o formatos.

Si recibes datos en un formato XML, es posible que necesites un lector XML para leer y analizar los documentos XML. Un lector XML lee documentos XML y extrae la información que contienen; de esta manera la hacen disponible para su uso en aplicaciones. Los lectores XML también se pueden usar para validar documentos XML según un esquema o definición de tipo de documento y para transformar documentos XML a otros formatos.

## XML and JSON

La idea principal de XML de etiquetas generalizables es extremadamente útil. Sin embargo, la rapidez con que se desenvuelve la tecnología ha hecho posible tratar una limitación clave de la tecnología XML -su verbosidad. De las 28 palabras en la [Ilustración 230](#), solo 9 palabras (~33%) expresan información única. El resto de las 19 palabras (67%) son solo etiquetas de protocolo que no comunican información única sobre John Smith, lo que resulta ineficiente, particularmente cuando un importante uso es, precisamente, intercambiar información.

Por esta razón, los programadores han reemplazado cada vez más XML con JavaScript Object Notation (JSON), que sustituye las etiquetas con pares de clave/valor, eliminando de este modo las etiquetas de cierre y reduciendo la cantidad de palabras necesarias para convertir la misma información. La [Ilustración 231](#) muestra un ejemplo de un archivo JSON que especifica cómo una aplicación web escrita con la aplicación de web Flask de Python debe ejecutarse. Puedes ver que JSON elimina las etiquetas de cierre como `</configurations>`, `</name>` etc.

En un futuro cercano, es muy probable que XML continúe siendo popular en ambientes altamente regulados como en el contexto de reportes financieros, en los cuales el enfoque es, precisamente, especificar el tipo de documento y sus propiedades. Para la mayoría de los otros contextos, en que el enfoque es el intercambio simplificado de información, es probable que XML se convierta en tecnología de sistema heredado, siendo reemplazado cada vez más por JSON.



**Ilustración 231** — Los archivos JSON se diferencian de XML por el uso de pares clave-valor.



## Términos y definiciones del capítulo

**CSS (CSS):** Hojas de estilo en cascada corresponde a las siglas en inglés de *Cascading Style Sheets*. Es un estándar de marcado web que permite a los diseñadores web definir la apariencia y posición de una página web utilizando efectos dinámicos especiales.

**Declaración DOCTYPE (*DOCTYPE declaration*):** Primera línea de una página HTML que le indica al navegador qué tipo de documento esperar, particularmente qué versión de HTML.

**Etiqueta cabecera (*Head tag*):** Contiene información sobre el documento HTML que en su mayoría no es visible para el usuario.

**Etiqueta de anclaje (*Anchor tag*):** La etiqueta `<a>` permite agregar enlaces a una página HTML, también conocidos como hipervínculos.

**Etiqueta de encabezado (*Heading tag*):** Son para títulos y subtítulos. La etiqueta `<h1>` representa el encabezado principal en una página y generalmente se usa para el título de la página, mientras que los otros encabezados `<h2>` a `<h6>` se usan para subtítulos.

**Etiqueta de formulario (*Form tag*):** Se crea utilizando la etiqueta `<form>`. Cada formulario también tiene varias etiquetas `<input>` para especificar el tipo de entrada que se desea del usuario.

**Etiqueta de imagen (*Image tag*):** La etiqueta `<img>` se usa para agregar imágenes a la página HTML. Requiere el atributo de fuente (`src`) para dar la ubicación de la imagen, de modo que el navegador pueda obtener la imagen y mostrarla en la página web.

**Etiqueta de párrafo (*Paragraph tag*):** La etiqueta `<p>` se usa para definir un bloque de texto como un párrafo.

**Etiqueta de subíndice (*Subscript tag*):** La etiqueta `<sub>` se utiliza para mostrar texto como subíndice.

**Etiqueta de subrayado (*Underline tag*):** La etiqueta `<u>` se utiliza para subrayar texto.

**Etiqueta de superíndice (*Superscript tag*):** La etiqueta `<sup>` se utiliza para mostrar texto como superíndice.

**Etiqueta de tachado (*Strike tag*):** Las etiquetas `<del>` o `<strike>` se utilizan para tachar visualmente el texto.

**Etiqueta en negrita (*Bold tag*):** La etiqueta `<b>` se utiliza para poner texto en negrita.

**Etiqueta HTML (*HTML tag*):** Una etiqueta es una palabra clave especial que está encerrada por `<>`. La etiqueta HTML define el inicio y el fin de un documento HTML.

**Guión gráfico (*Storyboard*):** Son representaciones visuales de proyectos de creación de sitios web.

**HTML (Lenguaje de marcado de hipertexto) [*HTML (Hypertext Markup Language)*]:** Es el lenguaje de marcado estándar utilizado para crear páginas web y aplicaciones web.

**JavaScript (*JavaScript*):** Un lenguaje de programación que permite potenciar un HTML con animación, interactividad y efectos visuales dinámicos.



**Proveedores de hospedaje web (Website host):** Típicamente una empresa o proveedor que permite a los usuarios mantener, asegurar y respaldar dominios de sitios web en servidores web.

**POST (POST):** Un método HTML que permite a los usuarios enviar datos a un servidor.

**Servidor web (Web server):** Servidores de computadoras diseñados y programados específicamente para alojar sitios web y facilitar la comunicación entre otras computadoras a través de redes de internet.

**Tablas (Tables):** En HTML se crean utilizando la etiqueta <table>, y las etiquetas <tr> y <td> para definir las filas y celdas de la tabla.

**Wireframe (Wireframe):** Es también conocido como esquema de página. Es un esquema visual aproximado de una aplicación propuesta. Es un tipo específico de prototipo generado durante el proceso de diseño centrado en el usuario para recopilar comentarios y retroalimentación de los usuarios potenciales antes de finalizar los diseños.



## Caso del capítulo

### ¡Wrenley salva a las mascotas!

Recientemente, a Wrenley se le dio la oportunidad de desarrollar un sitio web para una organización local sin fines de lucro dedicada a proteger a los animales alojados en un refugio. El objetivo del sitio web es encontrar un hogar permanente para las mascotas mediante la adopción. Wrenley realmente tiene pasión por las mascotas y quiere ayudarlas a encontrar nuevos hogares en cualquier forma que pueda.

Es tu trabajo ayudar a Wrenley a planificar el nuevo sitio web.

.....

**Pregunta 1:** Escribe una lista detallada de todas las páginas web que Wrenley debería incluir en su sitio web (ejemplos: “Acerca de nosotros”, “Contáctenos”, “Donar”). Al enumerar cada página web, incluye una pequeña muestra del contenido que creas útil para ellas.

**Pregunta 2:** ¿Qué lenguajes de desarrollo de páginas web recomendarías para el sitio web de Wrenley? Explica por qué esos lenguajes son necesarios.

**Pregunta 3:** Busca un buen proveedor de alojamiento de sitios web que potencialmente pueda alojar el sitio de Wrenley. Enumera la URL y proporciona una descripción del proveedor de alojamiento del sitio web.

**Puntos extra:** Desarrolla un prototipo de HTML en el sitio web propuesto para Wrenley. Puedes aprender más sobre cómo desarrollar un sitio web usando HTML en W3Schools (<https://www.w3schools.com/html/default.asp>).