



CREACIÓN DE PÁGINAS WEB CON EL LENGUAJE DE MARCAS (UF1302)

Familia profesional: Informática y Comunicaciones

[Manual de contenidos](#)

Reservados todos los derechos. El contenido de esta obra está protegido por la ley, que establece penas de prisión o multas, además de las correspondientes indemnizaciones por daños y perjuicios, para quienes reprodujeren, plagiaren, distribuyeren o comunicaren públicamente en todo o en parte, una obra literaria, artística o científica, o su transformación, interpretación o ejecución artística fijada en cualquier tipo de soporte o comunicada a través de cualquier medio, sin la preceptiva autorización.

Todos los nombres propios de programas, sistemas operativos, equipos, hardware, programas de afiliación, páginas web, etc. que aparecen en esta publicación son marcas registradas de sus respectivas compañías, organizaciones y propietarios y tan solo se muestran a modo informativo.

Conzepto Comunicación Creativa

ÍNDICE

1. LOS LENGUAJES DE MARCAS

1.1. Características de los lenguajes de marcas

- 1.1.1 Utilización de etiquetas
- 1.1.2 Compatibilidad
- 1.1.3 Editores de texto

1.2. Estructura de un documento creado con lenguaje de marcas

- 1.2.1. Comienzo del archivo
- 1.2.2. Encabezados
- 1.2.3. Título
- 1.2.4. Estilos
- 1.2.5. Cuerpo
- 1.2.6. *Scripts*

1.3. Navegadores web

- 1.3.1. Navegadores modo texto
- 1.3.2. Los navegadores más utilizados
- 1.3.3. Diferencias de visualización

1.4. Marcas para dar formato al documento

- 1.4.1. Marcas de inicio y final
- 1.4.2. Marcas de aspecto
- 1.4.3. Marcas de párrafo
- 1.4.4. Marcas de fuentes y colores
 - 1.4.4.1. Listas ordenadas

1.5. Enlaces y direccionamientos

- 1.5.1. Creación de un enlace
- 1.5.2. Tipos de enlace
- 1.5.3. Los enlaces y la navegación

1.6. Marcos y capas

- 1.6.1. Marcos
 - 1.6.1.1. Creación de marcos
 - 1.6.1.2. Tipos de marcos
 - 1.6.1.3. Situación de los marcos
 - 1.6.1.4. Configuración de los marcos

- 1.6.2. Capas

2. IMÁGENES Y ELEMENTOS MULTIMEDIA

2.1. Inserción de imágenes: formatos y atributos

- 2.1.1. Incluir imágenes en las páginas
- 2.1.2. Atributos de las imágenes
- 2.1.3. Propiedades de ubicación de las imágenes

2.2. Mapas de imágenes

- 2.2.1. Definición de mapa
- 2.2.2. Creación de un mapa con una imagen
- 2.2.3. Establecer diferentes partes en la imagen
- 2.2.4. Vincular las diferentes partes de la imagen

2.3. Inserción de elementos multimedia: audio, vídeo y programas

- 2.3.1. Características y propiedades de los elementos multimedia
- 2.3.2. Recursos necesarios para el funcionamiento de los elementos multimedia
- 2.3.3. Etiquetas y propiedades para la inserción de audio
- 2.3.4. Etiquetas y propiedades para inserción de vídeo
- 2.3.5. Etiquetas y propiedades para la inserción de programas

2.4. Formatos de audio y vídeo

- 2.4.1. Descripción de los formatos de audio
- 2.4.2. Descripción de los formatos de vídeo
- 2.4.3. Configuración de los recursos para audio
- 2.4.4. Configuración de los recursos para vídeo

2.5. Marquesinas

- 2.5.1. Los textos con movimiento
- 2.5.2. Utilización de etiquetas para incluir marquesinas
- 2.5.3. Las marquesinas y los distintos navegadores

3. TÉCNICAS DE ACCESIBILIDAD Y USABILIDAD

3.1. Accesibilidad web, ventajas de la accesibilidad

- 3.1.1. Definición de Accesibilidad
- 3.1.2. Aplicabilidad de la Accesibilidad
- 3.1.3. Descripción de las ventajas de la Accesibilidad

- 3.1.4. Usabilidad web, importancia de la usabilidad
 - 3.1.4.1. Definición de usabilidad
 - 3.1.4.2. Interacción web-individuo
 - 3.1.4.3. Aplicabilidad de la usabilidad
 - 3.1.4.4. Recursos sobre usabilidad
- 3.1.5. Aplicaciones para verificar la accesibilidad de sitios web (estándares)
 - 3.1.5.1. Recursos web estándares
 - 3.1.5.2. Utilización de los recursos en las páginas web
 - 3.1.5.3. Comprobar la accesibilidad en las páginas web
- 3.1.6. Diseño de sitios web usables
 - 3.1.6.1. Descripción de sitios web usables
 - 3.1.6.2. Estudio de la estructura y diseño de los sitios web usables
- 3.1.7. Adaptación de sitios web usables

4. HERRAMIENTAS DE EDICIÓN WEB

4.1. Instalación y configuración de herramientas de edición web

4.2. Funciones y características

- 4.2.1. Descripción de los elementos de las herramientas web
- 4.2.2. Ubicación de los elementos de las herramientas web
- 4.2.3. Propiedades de los elementos de las herramientas web
- 4.2.4. Tareas a realizar con una herramienta de edición web
- 4.2.5. El lenguaje de marcas y las herramientas de edición web

OBJETIVOS

Esta Unidad Formativa está comprendida por cuatro temas. Primeramente, aprenderá sobre cómo utilizar los lenguajes de marca, para qué sirven sus etiquetas y cómo puede hacer que sus documentos HTML tengan compatibilidad con la mayoría de los navegadores que utilizan los usuarios actualmente.

Además, verá los principales editores de texto y cómo van a ayudar en su trabajo como desarrolladores web. Este tipo de herramientas están especializadas para ahorrar tiempo y facilitar el trabajo.

También, va a ver cómo crear la estructura de un documento HTML desde su encabezado, pasando por los títulos, insertando estilos y *scripts*.

Otra cosa que verá serán los tipos de navegadores que existen como, por ejemplo, los de tipo texto, por voz o lo más comunes como Ópera, Chrome, Mozilla, entre otros.

Debido a esta gran diversidad entre los navegadores, los documentos HTML que cree no se verán igual en todos ellos, por lo que, aprenderá varias soluciones a este problema.

Por otra parte, profundizará en los tipos de marcas que existen en HTML para dar aspecto al documento, marcas de párrafo y marcas para añadir tipos de fuente y colores.

Otra parte muy importante que aprenderá para crear webs útiles, será añadir enlaces como los vínculos y las anclas, con ellos podrá enlazar imágenes, vídeos y otros archivos.

Además, verá cómo utilizar las marcas y las capas, cosa que es muy utilizada por los desarrolladores web para crear páginas, ya que, gracias a estos elementos, puede hacer divisiones del espacio de la página para posicionar de manera milimétrica los diferentes elementos.

También, aprenderá a manipular audios, imágenes y vídeos para insertarlos en la web. Para ello, se verán los tipos de formatos que existen de cada uno de ellos, cuán utilizarlos y cómo.

Otra parte muy destacada del desarrollo web serán las técnicas de accesibilidad y usabilidad que, como ya verá, son imprescindibles para crear una página web profesional.

Y, por último, estudiará las diversas herramientas de edición web en profundidad. Cómo pueden ayudar estas herramientas y cuáles se adaptarán más a su forma de programar.

CONTENIDOS

1. LOS LENGUAJES DE MARCAS

Un lenguaje de marcas es una manera de codificar un documento el cual, además de texto, incorpora etiquetas o marcas que contienen información adicional acerca de la estructura del texto o su presentación.

Hay diferentes tipos de lenguajes de marcas:

- Lenguaje de presentación: Como el RTF (formato de texto enriquecido) el cual está diseñado para crear documentos con formato de presentación añadido.
- Lenguajes de procedimientos: HTML es uno de los lenguajes de procedimientos más usados y su objetivo es crear un documento atractivo visualmente para ser compartido en Internet.
- Lenguajes descriptivos: Un ejemplo de lenguaje descriptivo es el XML (*eXtensible Markup Language*), el cual solo indica información sobre la estructura del documento sin añadir información relativa al diseño del mismo.

En el caso del desarrollo web, los lenguajes HTML y CSS son los lenguajes básicos para la organización y visualización de un sitio web.

El HTML (Hyper Text Markup Language) es el lenguaje básico de marcado de hipertexto y es el que indica al navegador cómo visualizar los contenidos de un sitio web. Fue creado en 1989 por Tim Berners-Lee y es el estándar en la creación de páginas web hasta hoy. El lenguaje HTML surge de la evolución del lenguaje SGML, y este a su vez del GML. SGML surgió en 1986 pero, al ser tan complejo y caro a nivel de software, dejó de ser funcional para lo que la evolución de la *World Wide Web* (el Internet) estaba reclamando.

Así, el HTML se convirtió en una versión simplificada de su predecesor SGML, que permite organizar, enlazar y compatibilizar grandes cantidades de información.

El HTML ha ido evolucionando durante todos estos años. Actualmente se usa la versión HTML5, una versión más eficiente, con nuevos elementos, atributos y comportamientos. Su tecnología mejorada, permite su implementación en un amplio abanico de sitios Web y aplicaciones. La semántica del HTML5 permite describir con mayor precisión el contenido del sitio web; su conectividad y rendimiento mejorados establece nuevas formas de comunicación con el servidor y agiliza el intercambio de datos, ganando tiempo de envío-respuesta de los mismos. Soporta nuevos formatos multimedia, gráficos y efectos 2D y 3D. Además, confiere nuevos accesos y salidas a la interacción con los dispositivos mediante APIs, lo que mejora la experiencia de usuario con el entorno digital.

El lenguaje CSS no se ha quedado atrás y ha evolucionado a la par del HTML. Actualmente el estándar más usado es el CSS3 que, junto con HTML5, ofrece una gran variedad de nuevos efectos y recursos para hacer diseños más sofisticados en los sitios Web.

1.1. Características de los lenguajes de marcas

Los lenguajes de marcas, también designados lenguajes de marcado o lenguajes de descripción de documentos, son un tipo de lenguaje que combina información relacionada con el mismo, es decir, la información añadida se mezcla con el texto principal para definir así su estructura y el procesamiento del documento.

Algunas de las características de los lenguajes de marcas son:

- Universalidad: El lenguaje se codifica en texto plano por lo que cualquier persona puede leer y editar la información.
- Lenguaje compacto: Las etiquetas de marca se mezclan en el mismo contenido.
- Independencia: El lenguaje de marca es independiente del dispositivo en el que se muestra.
- Facilidad de procesamiento: Permite desarrollar lenguajes especializados, adaptándose al tipo de documento que quiera procesar.
- Flexibilidad: Posibilidad de combinar con otros lenguajes.

Así, el HTML es el lenguaje de marcas más conocido y usado para desarrollar páginas web hasta el momento. Tiene la flexibilidad de combinar con otros lenguajes como pueden ser el CSS, para dar un formato más visual y atractivo al documento; el JavaScript, que enriquece la interacción web-usuario y PHP, entre otros muchos. También es un lenguaje ligero, fácil de procesar, fácil de mezclar con otros lenguajes en el mismo contenido y es, sobre todo, universal, puesto que cualquier persona puede entenderlo, interpretarlo y editarlo.



TOME NOTA

El lenguaje de marcas HTML permite estructurar el documento en secciones para organizar la información de manera lógica. Para ello, puede usar elementos como párrafos, cabeceras, títulos, listas, etc. Estos elementos serán interpretados por el navegador para que pueda visualizar la información.

En el siguiente vídeo, de cerca de siete minutos de duración, se amplían estas características generales:

RECURSO MULTIMEDIA



1.1.1. Utilización de etiquetas

La Web, tal y como se conce hoy día, se basa en HTML, es decir, un lenguaje de etiquetas. La etiqueta (del inglés *tag*) es la marca que indica al navegador qué estructura tiene el sitio Web y qué tipo de fragmentos de texto contiene, qué disposición tienen, qué tamaño y tipo de letra, si hay tablas, etc. Estas etiquetas hacen que la estructura de la web sea entendible.

Los lenguajes de marcas más conocidos que puede encontrar son el HTML, XML, XHTML y WML.

HTML

El HTML fue uno de los primeros lenguajes de marcas en aparecer y el más extendido por toda la World Wide Web. Las etiquetas de un documento escrito en HTML se escriben de la siguiente manera:

```
<p>Hola mundo</p>
```

Las etiquetas se componen de una apertura y un cierre. La apertura se escribe con estos signos "<>" y el cierre con esos dos signos más una barra "</>". Dentro de estos signos se escribe el código que precise: ya sea un párrafo, un encabezado, etc. También debe indicar que existen etiquetas cuyo cierre se escribe directamente dentro de la apertura, es decir:

```
<br/>
```

Esta es la base elemental para una etiqueta en lenguaje de marcas HTML.

Dentro de estas etiquetas también se pueden añadir los atributos los cuales establecen un valor y un comportamiento a la etiqueta. Un claro ejemplo de ello, muy utilizado, es el atributo style, a través del cual se indica el formato y estilo al contenido de la etiqueta.

```
<p style="color: blue; font-weight: bold;">Hola mundo</p>
```

A través del atributo style se ha indicado que "Hola mundo" tenga la tipografía en negrita y que el texto tenga un color azul.

Por otro lado, se pueden escribir etiquetas anidadas cuya información es acumulable, es decir:

```
<p><i>Hola mundo</i></p>
```

En este caso se indica que el texto "Hola mundo" aparecerá en cursiva.

HTML es el lenguaje de marcas más utilizado hasta ahora, aunque, gracias a su flexibilidad, ha podido evolucionar hacia un lenguaje más económico y adaptativo a otras tecnologías. Esto se traduce al nuevo HTML5, el cual, está ayudando en la utilización de herramientas de gestión y la inclusión de audio y vídeo para enriquecer el contenido. Otro de los cambios fundamentales del HTML5 es la inclusión del estándar DOM (Modelo de Objeto de Documento), la estructura que genera el

navegador cuando carga un documento HTML el cual, a su vez, se puede modificar con JavaScript, lo que hace que el contenido de la página sea más dinámico.

Aunque HTML5 sea una evolución del lenguaje muy reciente del HTML, ya está implementado en la mayoría de los navegadores actuales.

A continuación, observe las diferencias en las estructuras del HTML tradicional y el nuevo HTML5.

Ejemplo de estructura HTML clásica.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
      "http://www.w3.org/TR/html4/strict.dtd">  
  
<html>  
  
  <head>  
  
    <title>Estructura HTML básica</title>  
  
  </head>  
  
  <body>  
  
    <p>¡Hola mundo!</p>  
  
  </body>  
  
</html>
```

Ejemplo de estructura HTML5.

```
<!DOCTYPE html>  
  
<html lang="es">  
  
  <head>  
  
    <meta charset="utf-8">  
  
    <title>Estructura HTML5</title>  
  
  </head>  
  
  <body>
```

```
<header>

</header>

<nav></nav>

<section>

    Primera sección de la web

    <article> Artículo 1 </article>

</section>

<aside>

    <p> Texto 1 del aside </p>

    <p> Texto 2 del aside </p>

</aside>

<footer>

    Información pie de página

</footer>

</body>

</html>
```

XML

El XML o lenguaje de marcas extensible es un lenguaje de marcado genérico desarrollado e implementado por la W3C. Las etiquetas XML son muy similares a las de HTML, pero mucho más flexibles ya que los usuarios pueden definir sus propias etiquetas. Su objetivo es ser un estándar para el intercambio de información estructurada entre plataformas, acogiendo una amplia variedad de aplicaciones como editores de texto, hojas de cálculo, bases de datos. El XML se define como lenguaje de descripción de datos frente al HTML, que es un lenguaje de presentación.

Algunas de las características más destacables de este lenguaje son:

- Estructura jerárquica: Cada etiqueta debe ir anidada dentro de otra correctamente.
- Elemento raíz: Los elementos de un documento XML parten de un único elemento raíz.
- Comillas: Los valores de los atributos deben escribirse dentro de comillas simples o dobles.

- Es sensible al uso de mayúsculas y minúsculas.

XHTML

Este lenguaje de marcas no tuvo mucho éxito en su desarrollo. Este estándar se presentaba como una evolución del HTML, pero, el esfuerzo que suponía su implementación hizo que los desarrolladores desistieran en su desarrollo. Uno de sus objetivos era facilitar la visualización de las páginas web en los diferentes dispositivos y, aunque esto no cuajó, sí asentó la premisa de separar el contenido de la presentación, premisa que ya aporta el XML.

WML

El *Wireless Markup Language* es utilizado para desarrollar las páginas que se cargan en navegadores de los teléfonos móviles y otros dispositivos.

Algunas de sus características son:

- Pueden incluir imágenes y texto con formato.
- Permite manipular variables y formularios para la comunicación entre teléfono y servidor.
- Los elementos son sensibles a mayúsculas/minúsculas.

1.1.2. Compatibilidad

El proceso de crear una página web es sencillo, solo se necesita disponer de un ordenador con sistema operativo instalado, un navegador web y un editor de texto plano.

Los editores de texto plano, como su nombre indica, son programas para escribir texto, sin formato ni archivos multimedia añadidos. El más básico y común que puede usar es el Bloc de notas en Windows y el TextWrangler en Mac. También encuentra editores de texto más complejos, con cantidad de *plug-ins* para añadir que facilitará la tarea de escribir código.



TOME NOTA

Los plug-ins son pequeños programas que amplían y mejoran el funcionamiento de los programas a los que se añaden.

Para escribir código HTML no es necesario conectarse a Internet y tampoco es necesario alojarla en un servidor web. Aprender el lenguaje HTML es muy sencillo ya que no requiere de conocimientos previos de programación.

Por otro lado, HTML es compatible con cualquier plataforma en la que se pueda instalar un navegador web, pero, todos los navegadores no interpretan de la misma manera el código HTML, así que, el mismo contenido no lo verá igual en los diferentes navegadores.

Vea a continuación algunos de los navegadores más usados en la actualidad y sus principales características:

Edge



Logo actual de Edge

Es el navegador por defecto que usa Windows en su sistema operativo, por eso es uno de los más usados.

Hay que revisar, con cuidado, que el código que se escriba lo interprete correctamente este navegador. Añadir que, difícilmente, interpreta HTML5. Aun así, es recomendable testear su código en este navegador.

Mozilla Firefox



Logo actual de Mozilla Firefox

Mozilla Firefox es uno de los navegadores, por no decir el único, más útiles para los desarrolladores web. Dispone de una amplia gama de *plug-ins* que permiten, entre otras cosas, depurar código HTML para así ver los posibles errores cuando corre el código. Soporta sin problemas HTML5.

Google Chrome



Logo actual de Google Chrome

Este navegador fue desarrollado por Google y Google, a su vez, es uno de los principales impulsores del HTML5. Esto se traduce a que este navegador se convierte en una herramienta fundamental para implementar las nuevas funcionalidades del HTML5. Aunque es un navegador muy ligero, consume muchos recursos de CPU.

Safari



Logo actual de Safari

Al igual que Internet Explorer, Safari es el navegador que viene instalado por defecto en los dispositivos Apple.

Opera



Logo actual de Opera

También para los usuarios de Apple, Opera es un navegador rápido y fue un navegador pionero en muchas funcionalidades de HTML5.

Navegadores modo texto

Los usuarios de Linux o Unix, cuyo sistema operativo funciona bajo línea de comandos, usan los denominados navegadores modo texto, los cuales solo permiten visualizar texto. Aunque, también puede utilizar los Sistemas Operativos Windows bajo línea de comando como, por ejemplo, el Sistema Operativo DOS.

Estos navegadores son apropiados para conexiones muy lentas y, aunque, no suelen usarse para el proceso de desarrollar una página web, sí pueden ser útiles para crear sitios más accesibles.

1.1.3. Editores de texto

Los editores de texto son programas informáticos para crear y modificar archivos digitales, en este caso, archivos de texto plano. En un editor de texto puede escribir la base de una página web en HTML. Esto lo puede hacer con un software muy básico como, por ejemplo, el Bloc de notas, instalado por defecto en el sistema de Windows, en él puede escribir con formato plano (solo texto) su página web.

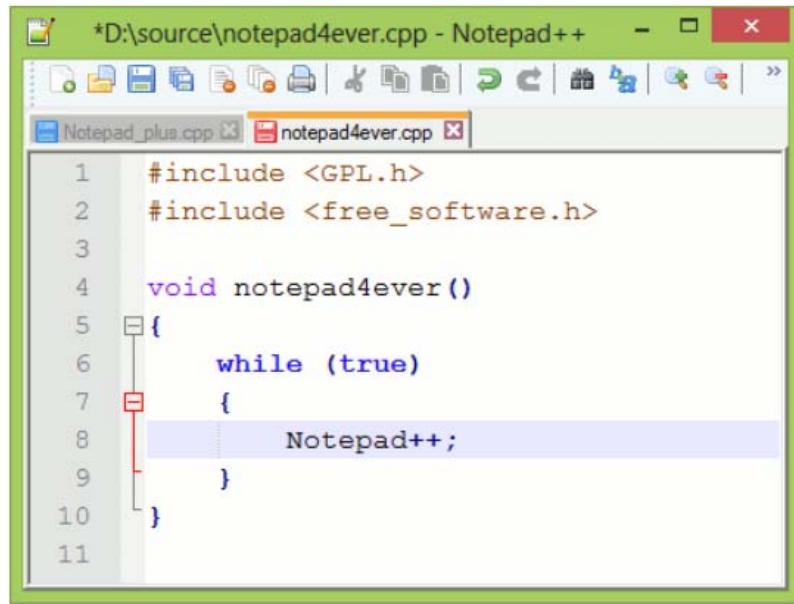
A lo largo del tiempo, se han ido desarrollando multitud de editores de texto que se ajustan a las necesidades de los desarrolladores web, aunque todos tienen en común funciones básicas como estas:

- Buscar y reemplazar: Puede buscar una palabra o conjunto de palabras para reemplazar y corregir.
- Copiar, cortar y pegar.
- Formatear: Dar formato al texto.
- Deshacer y rehacer.
- Filtros: se pueden programar acciones para trabajar de diferentes maneras el texto.

Los editores más actuales tienen una infinidad de *plug-ins* de código abierto que son diseñados por los propios usuarios que trabajan con el editor. Estos *plug-ins* agilizan el trabajo y ahorran tiempo y esfuerzo automatizando procesos básicos y complejos sobre sus códigos. Como hay infinidad de editores, hay algunos más fáciles de usar que otros y algunos que brindan más funcionalidades.

NotePad++

Es la versión actualizada del legendario Bloc de notas y Wordpad pero con funciones avanzadas. Su barra de herramientas se puede personalizar, incluye un mapa del documento para comprobar dónde se encuentra en cada momento, pestañas para trabajar en varios documentos, autocompletado de las etiquetas y funciones, etc. Es gratuito. Su página oficial es <https://notepad-plus-plus.org/>.



The screenshot shows the Notepad++ application window. The title bar reads "*D:\source\notepad4ever.cpp - Notepad++". The menu bar includes File, Edit, View, Insert, Search, Tools, Plugins, Settings, Help, and Exit. The toolbar contains icons for file operations like Open, Save, Print, and Find. Below the toolbar, there are tabs for "Notepad_plus.cpp" and "notepad4ever.cpp". The main text area displays the following C++ code:

```
1 #include <GPL.h>
2 #include <free_software.h>
3
4 void notepad4ever()
5 {
6     while (true)
7     {
8         NotePad++;
9     }
10 }
11
```

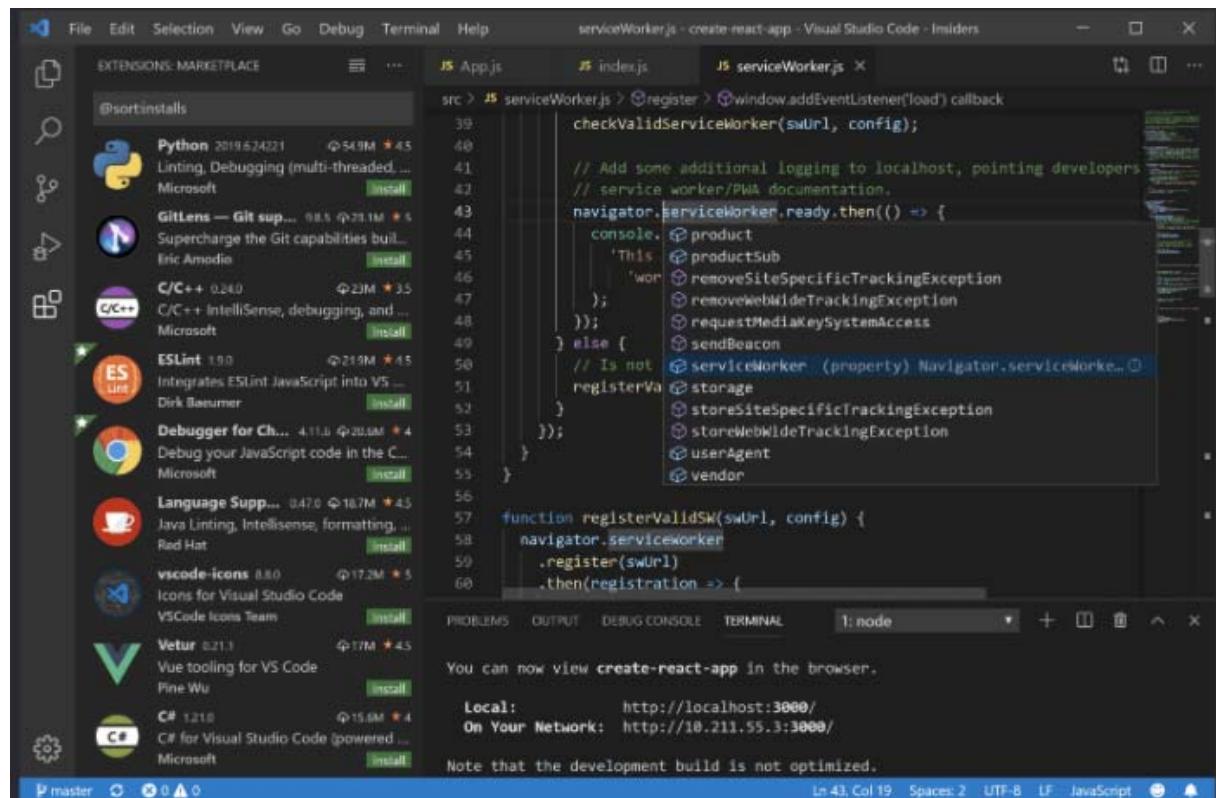
The code uses color-coded syntax highlighting: purple for keywords like `#include`, `void`, `while`, and `NotePad`; blue for comments; and black for strings and variables. A tooltip "NotePad++;" appears over the line `NotePad++;` at line 8. The Notepad++ interface has a light green background and a white text area.

Notepad++

Visual Studio Code

Es desarrollado por Microsoft, gratuito y disponible para Windows, Linuz e iOS. Esto, más que un sencillo editor, es un IDE (Entorno de Desarrollo Integrado), es decir, un software para el diseño de aplicaciones que aúna diferentes herramientas de desarrollo web en una sola interfaz.

Tiene infinitud de extensiones para los nuevos lenguajes de programación y *frameworks*. En definitiva, es un software muy potente, que combina a su vez otras herramientas que ofrecen multitud de combinaciones para trabajar procesos complejos. La página oficial es <https://code.visualstudio.com/>.



The screenshot shows the Visual Studio Code interface. On the left, there's a sidebar titled 'EXTENSIONS MARKETPLACE' listing various extensions like Python, ESLint, and C/C++. The main area is a code editor with three tabs: 'src > serviceWorker.js', 'index.js', and 'serviceWorker.js'. The 'serviceWorker.js' tab contains JavaScript code related to service workers. At the bottom, the terminal shows the command 'node' and the message 'You can now view create-react-app in the browser.' The status bar at the bottom right shows file details: '1: node', 'Ln 43, Col 19', 'Spaces: 2', 'UTF-8', 'LF', 'JavaScript'.

Interfaz de Visual Code.



TOME NOTA

Los frameworks son un concepto de marcos de trabajo (traducido literalmente) donde dispone de herramientas con las que puede ahorrar más tiempo y con las que puede hacer un código más limpio haciendo uso de estructuras ya elaboradas.

Brackets

Este editor de texto elaborado por la empresa Adobe es de código abierto. Se puede utilizar Windows, Linux y Mac. Destaca su función Inline Editors, que permite ver y trabajar varios archivos a la vez. Su función de vista en vivo, permite ver en tiempo real los cambios que aplica a su código. También gracias a su Preprocessor Support hace que sea más fácil encontrar, agregar y editar correlaciones de clases entre los archivos HTML y CSS. Estas pequeñas diferencias hacen que sea muy cómodo de usar y haga ganar tiempo en su trabajo. En su página oficial puede encontrar toda la documentación y descarga del software: <http://brackets.io/>.

Sublime Text

Es otro de los editores de textos más elegidos por los desarrolladores web junto a Visual Code. Tiene multitud de *plug-ins* y complementos para cada tipo de lenguaje de programación, multitud de

comandos de teclado y un modo libre de distracciones que permite centrarse en el trabajo. Se pueden tener varios documentos abiertos al mismo tiempo y una vista en miniatura para visualizar en qué parte del documento se encuentra y moverse rápidamente por él. Este editor está disponible para Windows, Mac y Linux y, aunque se distribuye como software de prueba, se puede comprar una licencia personal y usarlo en varios equipos. Encontrará más información y actualizaciones en su página oficial <https://www.sublimetext.com/>.

1.2. Estructura de un documento creado con lenguaje de marcas

Los documentos creados con lenguaje de marca tienen una estructura base reconocible.

Esta base es la que permite incorporar información en las diferentes secciones del archivo para, posteriormente, visualizarla en el sitio web. Generalmente, está delimitado por tres partes: una línea que contiene la información sobre la versión HTML, pero no siempre es incluida; una cabecera y un cuerpo con el contenido del documento. Todo esto debe ir entre las etiquetas <HTML></HTML>.

En el siguiente vídeo, con una duración de cinco minutos, conocerá la estructura básica de un documento HTML:



1.2.1. Comienzo del archivo

Al comienzo del archivo HTML puede encontrar varios aspectos. Hay que recordar que existen variaciones en cuanto a la escritura del código del antiguo estándar HTML4 o XHTML al nuevo estándar HTML5.

Por tanto, una de las primeras líneas de código del documento HTML es la que indica qué tipo de documento es. Para ello, se usa la etiqueta <!DOCTYPE html>. Esta es la etiqueta que indica al navegador qué tipo de documento es para así adaptarse y poder traducir correctamente la información que contiene de manera visual.

-Una estructura más compleja para HTML4 o XHTML:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html>
  <head>
    <title></title>
  </head>
  <body>

  </body>
</html>
```

-Frente a la simplicidad del HTML5:

```
<!DOCTYPE html>

<html>
  <head>
    <title></title>
  </head>
  <body>

  </body>
</html>
```

1.2.2. Encabezados

Los encabezados son los que, comúnmente, se denominan **head** y las etiquetas que se utilizan en esta sección son `<head></head>`.

Es el lugar donde se especifican los metadatos del documento. Esta parte no se muestra de forma visual en el navegador, es decir, cuando se navega por el sitio web, estos datos no los ve. Sin embargo, es posible verlos a través del código fuente de la página.

Los metadatos que pueden incluirse en esta sección son el título y descripción de la página. Toda la información que se introduzca en esta sección facilita a los buscadores crear y mostrar los resultados de búsqueda según los criterios de búsqueda introducidos por el usuario. También es el lugar indicado para incluir los enlaces de otros documentos externos para que se relacionen con el actual, como por ejemplo documentos para cargar estilos CSS o scripts, que son fragmentos de código con instrucciones que permiten la interacción del usuario con el sitio web.

La etiqueta meta ofrece muchas posibilidades para indicar datos sobre el sitio web. La manera correcta de escribir la etiqueta meta es <meta />, como puede apreciar, el cierre de la etiqueta está dentro de los símbolos mayor y menor. Para incluir esta información debe usar los atributos *name* y *content* que se incluirán dentro de la propia etiqueta meta. En *name* se escribe el atributo reconocible para el navegador, como pueden ser *description*, *keywords*, *autor*, *viewport*... En *content* se escribe lo pertinente a cada atributo, por ejemplo, en *description* se escribe texto, sin embargo, en *viewport* se escriben parámetros. Algunos de los atributos más usados son:

```
<!DOCTYPE html>

<html>
  <head>
    <title>Cómo usar las etiquetas meta</title>
    <meta name="description" content="Añadir descripción" />
    <meta name="keywords" content="Usar palabras clave" />
    <meta name="author" content="añadir autor de la página" />
    <meta name="viewport" content="initial-scale=1, width=device-width" />

  </head>
  <body>

  </body>
</html>
```

- **description:** Con este atributo puede añadir una descripción del sitio web. Si trabaja en una buena descripción que potencie el sitio web, este, mejorará su posicionamiento en los buscadores.
- **keywords:** Este atributo permite añadir palabras clave para mejorar la posición del sitio web.
- **author:** Así puede incluir el autor o autora del sitio web.
- **viewport:** Este atributo permite indicar el comportamiento de la región visible del navegador, es decir, es un parámetro muy recurrente para hacer *responsive* el sitio web.
- **meta:** Otro aspecto muy importante de la etiqueta <meta /> es que permite incluir información para los robots de los buscadores. A través del atributo robots indica si los robots deben o no indexar el sitio web o páginas concretas del mismo.

```
<head>
  <meta name="robots" content="index"/>
</head>
```

Algunos parámetros para usar en el *content* de robots son los siguientes:

- -index: Indica que se indexe el contenido de la página en el buscador.
- -noindex: Indica que no se indexe el sitio web.

- -none: Equivale a indicar los parámetros noindex,nofollow.
- -follow: Indica a los buscadores que siga los enlaces que encuentre en el sitio web.
- -nofollow: Indica que no siga los enlaces del sitio web.
- -nosnippet: Indica a los buscadores que no muestre fragmentos.
- -unavailable_after: Establece una fecha de caducidad para no rastrear más el sitio web.
- -noarchive: Indica que el sitio web no se almacene en caché, lo que haría que el sitio web sea más lento cada vez que cargue.

1.2.3. Título

La etiqueta *title* sirve para añadir un título descriptivo para el sitio web. También se incluye en la sección head y se escribe de la siguiente manera.

```
<title>Título del sitio web</title>
```

Al igual que la etiqueta *<meta />*, esta etiqueta *<title>* facilita información a los buscadores para crear los resultados de búsqueda. Es muy importante elegir las palabras claves adecuadas para el título ya que, un buen título, dará mejores resultados en los buscadores y el sitio web saldrá mejor posicionado.

1.2.4. Estilos

A un documento HTML se le puede dar formato y estilo para hacer más atractivo el documento web. La etiqueta *<style>* define el estilo de los elementos contenidos dentro del documento HTML y permite jerarquizar la información a través de colores, tipografías, tamaños de letra, variar el tamaño de las imágenes, entre otras muchas opciones.

Existen varias opciones para incluir la etiqueta *<style>* y son:

- Dentro de la cabecera del documento. Así definie qué etiquetas HTML tendrán un estilo concreto. En el ejemplo, se define que todas las etiquetas *<p>* (de párrafo) del documento HTML tendrán un color negro y la tipografía será Verdana.

```
<!DOCTYPE html>

<html>

    <head>
        <title>La etiqueta style</title>
        <style type="text/css">
            p { color: black; font-family: Verdana; }
        </style>
    </head>

    <body>

    </body>
</html>
```

Es posible dar estilo a etiquetas concretas dentro del documento HMTL. En este ejemplo se define que el *body* de la página HTML será todo de color *aqua*. Como nota añadida, en el body, además de color, puede incluir imágenes o vídeos.

```
<!DOCTYPE html>

<html>

    <head>
        <title>La etiqueta style</title>
    </head>

    <body style="background-color: aqua;">

    </body>
</html>
```

Puede vincular hojas de estilo externas al documento, o documentos HMTL, de un sitio web mediante la etiqueta <link>.

```
<!DOCTYPE html>

<html>

    <head>
        <title>La etiqueta style</title>
        <link rel="stylesheet" type="text/css" href="css/estilos.css">

    </head>

    <body>

    </body>
</html>
```

La etiqueta `<link>` incluye 3 atributos:

- `rel`: Indica la relación entre el recurso enlazado, en este caso, un archivo CSS y el documento HTML. Siempre se usa el valor `stylesheet`.
- `Type`: Indica el tipo de recurso enlazado. Siempre se indicará `text/css`.
- `Href`: Indica la URL del archivo CSS que contiene los estilos.

La manera más limpia y eficaz de incluir los estilos en un documento es a través de la etiqueta `<link>`. La ventaja de usar hojas de estilo externas es que se puede homogeneizar el estilo en diferentes documentos HTML a la vez y, además, los cambios que precisen realizarse se controlan mejor desde un archivo localizado. Así asegurar que dichos cambios queden efectuados en todos los documentos. Con la etiqueta `<link>` se pueden enlazar tantas hojas de estilo como sean necesarias.

1.2.5. Cuerpo

La etiqueta `<body>` sigue después del encabezado, es la etiqueta que almacena el contenido del documento HTML. Puede albergar múltiples etiquetas de texto, enlaces, vídeo, audio, imágenes, tablas, formularios, botones, etc. En esta etiqueta `<body>` puede confeccionar y organizar cómo se verá la información en el sitio web mientras el usuario navega por el mismo gracias a las múltiples etiquetas que puede usar y a los estilos CSS con los que dar un formato atractivo al contenido.

```
<!DOCTYPE html>

<html>

    <head>
        <title>La etiqueta body</title>

    </head>

    <body>
        <h1>La Etiqueta body</h1>
        <p>En la etiqueta body podemos organizar la información
            que se verá en el navegador.</p>
    </body>
</html>
```

Existen ciertos atributos específicos para la etiqueta `<body>` que, colocados dentro de la propia etiqueta, indicarán qué formato tendrán otras etiquetas dentro de la sección *body* o qué aspecto tendrá el *body* general. Estos son algunos ejemplos:

- **Text:** Indica el color predeterminado para el texto normal dentro del elemento.

Ejemplo: `<body text="#222222">`

- **Bgcolor:** Indica un color para llenar el fondo del elemento.

Ejemplo: `<body bgcolor="#00ccee">`

- **Background:** Indicando una URL a través de este atributo, usa una imagen para cubrir el fondo del elemento.

Ejemplo: `<body background="imagenes/fondo.png">`

- **Link:** Indica el color predeterminado de todos los vínculos dentro del elemento *body*.

Ejemplo: `<body link="#0000ff">`

Es menester mencionar que estos atributos ya han quedado obsoletos con la evolución del HTML hacia el HTML5. Se recomienda usar estos atributos gráficos en las hojas de estilo CSS. En la etiqueta *body* se pueden trabajar otros muchos aspectos, entre ellos, los eventos. Los eventos son pequeños trozos de código, normalmente programados en JavaScript, JQuery y HTML, que permite al usuario interactuar con el contenido de manera más dinámica. Uno de los atributos más útiles es `onload`, el cual permite que inmediatamente después de que en la página hayan cargado todos los elementos se ejecute una función, en este caso, un *pop up* con la suscripción a una newsletter o un banner de publicidad.

```
<!DOCTYPE html>

<html>

    <head>
        <title>La etiqueta body</title>

    </head>

    <body onload="window.prompt('¡Suscríbete a mi web!')">
        <h1>La Etiqueta body</h1>
        <p>Conoce más sobre la etiqueta body.</p>
    </body>
</html>
```

1.2.6. Scripts

Los *scripts* son programas que pueden incluirse en el código HTML. Estos programas son ejecutados por el navegador bien cuando se visualiza el contenido o cuando se hace clic sobre un enlace.

Uno de los lenguajes más utilizados para desarrollar *scripts* es el lenguaje de programación interpretado JavaScript, esto quiere decir que no requiere compilación y las líneas de código son leídas por el navegador en el momento. Con los *scripts* se mejora notablemente la interactividad y funcionalidad de las páginas y de la interacción del usuario con el contenido a través de mensajes emergentes, cambios de aspecto en elementos del documento, animaciones, etc.

Para añadir estos programas se usa la etiqueta `<script>`, la cual puede colocarse en cualquier lugar del documento, aunque lo conveniente es incluirla en la cabecera `<head>`.

Los atributos más comunes para la etiqueta `<script>` son:

- **Type:** Al igual que la etiqueta `<style>`, este atributo especifica el tipo de lenguaje de script que corresponda, en este caso “`text/javascript`”.
- **Src:** Este otro atributo indica la ruta donde se encuentran los archivos externos que contienen código JavaScript, los cuales se quieren incluir en el archivo HTML.

Ejemplo: En este ejemplo se indica qué tipo de archivo es (`type`) y dónde se ubica (`src`).

```
<!DOCTYPE html>

<html>

    <head>
        <title>La etiqueta Script</title>
        <script type="text/javascript" src="code.js"></script>
    </head>

    <body>
    </body>
</html>
```

Una vez vista la estructura básica de esta etiqueta, mencionar que existen varias opciones para añadir código JavaScript al documento HTML.

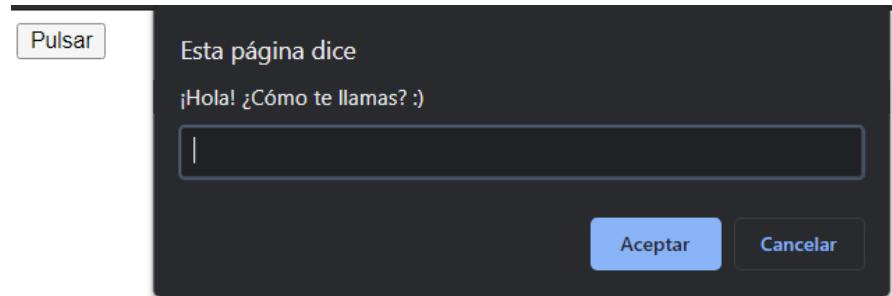
- Una de ellas es la vista en el ejemplo anterior y la más limpia. Se trata de colocar la etiqueta `<script>` en el *head* y vincular un archivo de código JavaScript externo. En este archivo JavaScript externo podrá tener todos los eventos y funciones asociados a los elementos del archivo HTML, de manera ordenada y clara. Trabajar de esta manera ahorrará tiempo a la hora de encontrar fallos o depurar código, además de darle homogeneidad a los cambios para que se apliquen todos por igual. Es esta la manera más recomendada a la hora de vincular código JavaScript al archivo HTML.
- Sin embargo, existen otras prácticas. Otra de las mencionadas es escribiendo código JavaScript dentro de la etiqueta `<script>` y esta, a su vez, dentro del *head*. En este ejemplo ha diseñado un programa que al entrar en el sitio web hay un botón que pone “Pulsar”, programado en el *body* con el evento `onClick`. Al pulsarlo se ejecutará el código JavaScript y aparecerá una ventana emergente que preguntará el nombre. Este es un ejemplo de funcionalidad básica que puede hacer con *scripts* en su sitio web.

```
<!DOCTYPE html>

<html>

    <head>
        <title>La etiqueta Script</title>
        <script type="text/javascript">
            function nombre(){
                window.prompt('¡Hola! ¿Cómo te llamas? :)')
            }
        </script>
    </head>

    <body>
        <button onclick="nombre()">Pulsar</button>
    </body>
</html>
```



- Una tercera opción para incluir código JavaScript en su código HTML es incluyendo el nombre de la función en el atributo *href* en enlaces de esta manera:

```
<!DOCTYPE html>

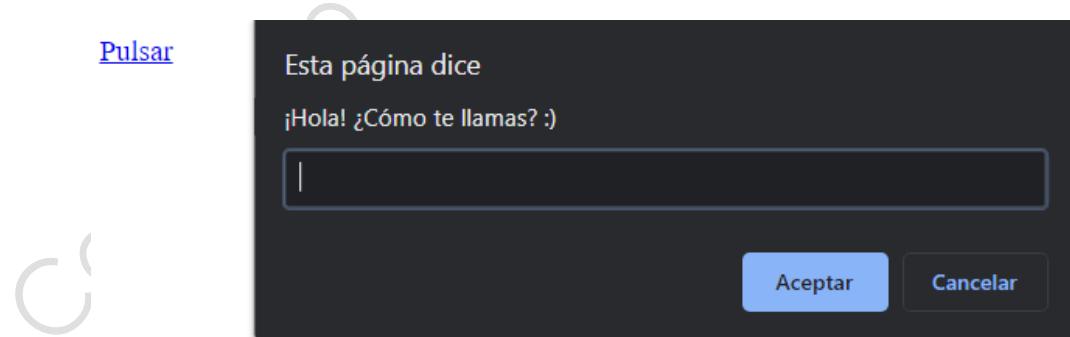
<html>

<head>
    <title>La etiqueta Script</title>
    <script type="text/javascript">
        function nombre(){
            window.prompt('¡Hola! ¿Cómo te llamas? :)')
        }

    </script>
</head>

<body>
    <a href="javascript:nombre()">Pulsar</a>
</body>

</html>
```



Los eventos más interesantes y útiles que puede añadir a su código HTML para hacer un contenido dinámico son los relativos al ratón como, por ejemplo:

- onclick: El evento se ejecuta cuando el usuario hace clic en el elemento programado.
- ondblclick: El evento se ejecuta cuando el usuario hace doble clic sobre el elemento programado.

- onmousedown: El evento se ejecuta cuando el botón del ratón se presiona sobre el elemento programado.
- onmousemove: El evento se ejecuta cuando el usuario move el cursor del ratón sobre el elemento programado.
- onmouseover: El evento se ejecuta cuando el usuario coloca el cursor del ratón sobre el elemento programado.
- onmouseout: El evento se ejecuta cuando el cursor del ratón sale de la zona del elemento programado.

1.3. Navegadores web

Un navegador web es un software que interpreta la información de distintos archivos y sitios web para su visualización e interacción. Es un visor de documentos HTML, los cuales pueden contener archivos multimedia y enlaces a otros sitios web o dentro del mismo sitio web. Estos documentos que muestra el navegador pueden ubicarse en el ordenador del usuario o en cualquier otro dispositivo conectado a ese ordenador a través de Internet.

El primer buscador fue creado en 1990 por el padre del HTML, Tim Berners-Lee, a raíz del proyecto WorldWideWeb (www), un proyecto que consistía la creación de un sistema para compartir información, pero esa información requería de un visor para hacerla accesible a los usuarios. Así, nació el primer navegador que facilitaba el acceso a esa información desde un ordenador personal.

El siguiente navegador con más impacto social y económico fue el Netscape Navigator, el cual funcionaba en casi todos los sistemas Unix y entornos Windows. Aunque el Internet Explorer que conoce hoy día no tardó en conquistar los mercados y desbancar al Netscape el cual, tras este batacazo, liberó el código dando origen al proyecto Mozilla.

En 2003, Apple lanzó el navegador Safari que, en ese momento, acaparó la mayor parte del mercado por su velocidad y continuas actualizaciones.

En 2004, parte del proyecto Mozilla, anteriormente mencionado, da origen a Mozilla Firefox, un navegador muy ligero. Es muy recomendado para el desarrollo web gracias a su gran cantidad de *plug-ins* especializados para facilitar el testeo y trabajo del desarrollador.

Ya en 2008 Google Chrome aparece en el mercado y se convierte en uno de los navegadores más utilizados hasta hoy gracias a su diseño limpio y sencillo y múltiples funciones, superando a sus mayores competidores, como Microsoft Edge o Safari. Aunque sigue siendo uno de los navegadores que consume más recursos de la CPU.

En 2015 Microsoft lanza Microsoft Edge como sucesor de Internet Explorer y, además de estar preparado para adaptarse a dispositivos móviles iOS, Android y Windows 10, tiene otras muchas tecnologías de integración y sincronización Pc-teléfono para enriquecer la experiencia de usuario.

No hay que olvidar la evolución paralela de los *Smartphones*. La evolución de los navegadores se ha visto comprometida a satisfacer la rápida aceptación de otros dispositivos móviles, como los

teléfonos móviles inteligentes (Smartphones), tablets, etc. Esta tendencia ha obligado al desarrollo de navegadores móviles y adaptables a los nuevos dispositivos móviles. Esto también ha hecho que se creen nuevos códigos y recursos dentro del lenguaje de marcas CSS para adaptar los sitios web a estos dispositivos, el llamado diseño responsive. Se trata de adaptar las tecnologías al uso cotidiano del usuario y, en este caso, el navegador web primario ha tenido que adaptarse a los pequeños dispositivos para seguir ofreciendo información al usuario.

Aun así, la comunicación entre el servidor web y el navegador se mantiene bajo los mismos protocolos y procedimientos. Generalmente, la comunicación entre navegador y servidor se realiza mediante el protocolo de comunicaciones HTTP (Hypertext Transfer Protocol) en el que las *peticiones* son enviadas por el cliente, es decir, un navegador web. Esta petición se envía a un servidor, el cual la gestiona y la responde. Entre estos dos procesos se encuentran una serie de intermediarios, denominados proxies, por los que los paquetes de datos viajan y son filtrados. El protocolo HTTP es complejo en su profundidad, pero no deja de ser el protocolo por excelencia gracias a su flexibilidad y facilidad de uso, características que hacen que este protocolo pueda evolucionar con las nuevas aplicaciones de Internet.

Otro tipo de protocolo soportado por los navegadores es el FTP (File Transfer Protocol) o HTTPS (Hypertext Transfer Protocol Secure), la cual añade una capa de seguridad SSL que encripta las interacciones del usuario con el sitio web como introducción de datos sensibles, personales o bancarios, entre otros.



TOME NOTA

En el futuro diseño de sitios web hay que considerar la capa de seguridad SSL. Esta capa, además de dar seguridad al sitio web, confiere confiabilidad al usuario que interactúe con la web a la hora de introducir datos sensibles

Puede conocer algo más sobre los navegadores y su funcionamiento en el siguiente vídeo, de diez minutos de duración:

RECURSO MULTIMEDIA



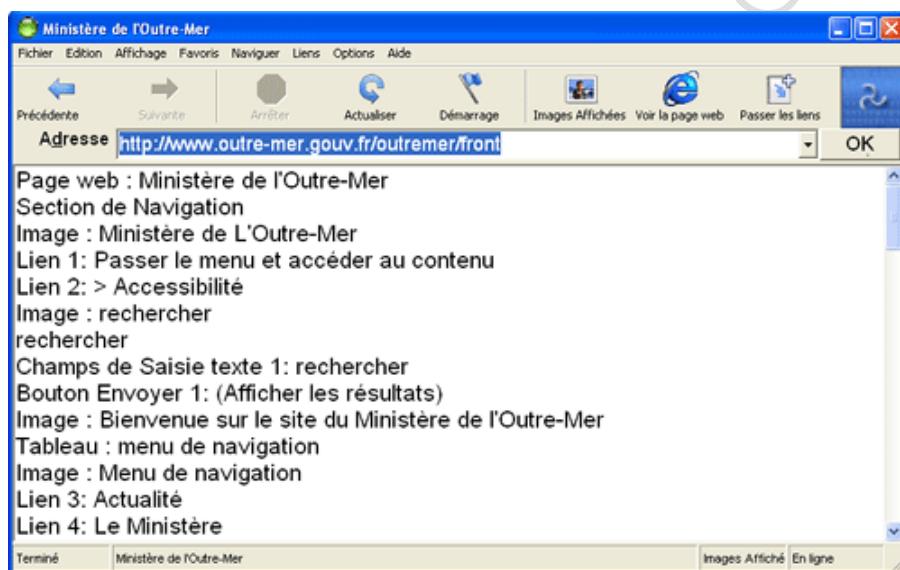
1.3. 1. Navegadores modo texto

Los navegadores web basados en texto es un navegador que solo presenta el texto de las páginas web, obviando los gráficos como imágenes, vídeos, audio y efectos en botones, etc. Es decir, la funcionalidad del CSS y Javascript quedan anulados.

Esto puede deberse a diversos motivos. Uno de ellos puede ser debido a la baja velocidad del ancho de banda que, al no haber suficientes datos para cargar la página en su totalidad, reduce la visualización a la estructura del texto.

Otro motivo es cuando no se tiene un ordenador con suficiente potencia de CPU, lo que implica que el software no tenga suficientes recursos para cargar todos los datos del sitio web.

Y otro de los motivos por el que se usa este tipo de navegadores es por lo útil que pueden llegar a ser para usuarios con discapacidad visual. El software de texto a voz lee el contenido a estos usuarios.



Algunos ejemplos de estos navegadores son:

- Webbie (<https://www.webbie.org.uk/es/>) Es un explorador web para usuarios con deficiencia visual. Está basado en Microsoft Internet Explorer y es compatible con Windows XP/Vista.
- Links (<http://links.twibright.com/>)
- Browsh (<https://www.brow.sh/>) Browsh es un navegador modo texto que va un paso más allá y codifica las imágenes y algunos elementos gráficos de manera muy básica.

1.3.2. Los navegadores más utilizados

No todos los navegadores web ofrecen las mismas funcionalidades ni sirven para las mismas operaciones que se necesita hacer. En los últimos años, Google Chrome ha ido ganando terreno frente a sus competidores. A continuación, se detallan las características más destacadas de cada uno de los navegadores más conocidos.

Google Chrome

Este navegador web se lanzó en 2008 por la compañía Google y es uno de los más utilizados por la mayoría de usuarios. Destaca por ser el navegador más rápido, por su estabilidad y su alta compatibilidad con todos los sistemas, a pesar de su alto consumo de RAM.

A este navegador se le pueden añadir infinidad de complementos que le confieren buena manejabilidad y una gran usabilidad al usuario. Además, por ser un navegador creado por Google, ofrece muchas posibilidades de sincronización con otras herramientas de Google, lo que hace que la experiencia de usuario sea muy fluida.

<https://www.google.es/chrome/>

Edge

A pesar de su sencillez y de ser el navegador por defecto de los sistemas Windows, este navegador web tiene bastantes puntos en contra:

- Microsoft Edge es su sustituto, por lo que Internet Explorer ya se ha quedado desactualizado. La compañía Microsoft pretende con Microsoft Edge mejorar en rendimiento y velocidad para posicionarlo de nuevo en la pole de los navegadores web. Este navegador fue creado en 1995 por Microsoft y fue uno de los primeros en acaparar todo el mercado gracias a su inclusión, por defecto, en todos los sistemas operativos de Windows.
- Es un navegador web de código cerrado y un foco incesante de ataques cibernéticos.
- Este navegador tiene problemas para visualizar las nuevas etiquetas HTML5. Debe tener especial cuidado cuando diseña una web ya que la mayoría de nuevas tecnologías no las soporta, dando como resultado problemas de visualización o, directamente, la web no se verá.

<https://www.microsoft.com/es-es/edge>

Mozilla Firefox

Mozilla Firefox es un navegador web creado y desarrollado como software libre con código abierto. Su objetivo es mejorar la privacidad y experiencia de los usuarios. Es compatible con toda clase de dispositivos como Windows, Linux, Mac, iOS y Android.

Firefox ofrece un gran rendimiento al utilizar menos caché. Gracias a sus continuas actualizaciones ofrece gran seguridad en la navegación y una gran variedad de extensiones para mejorar la experiencia de usuario. Una de las extensiones más importantes que tiene Firefox es su herramienta para desarrolladores y programadores web, Firebug, a través de la cual se puede analizar, editar, monitorizar y depurar el código fuente HTML, CSS y JavaScript instantáneamente, permitiendo ver los cambios al momento.

<https://www.mozilla.org/es-ES/firefox/new/>

Opera

Opera es un navegador creado en 1995 por la compañía Opera Software. A pesar de ser desconocido para la mayoría de los usuarios, Opera ofrece funcionalidades que lo hacen uno de los más innovadores y eficaces en equipos informáticos de bajas prestaciones.

Este navegador ofrece seguridad y rapidez en su navegación, soporta diversos estándares de lenguajes y es ágil en equipos con pocos requisitos. Además, cuenta con un modo llamado turbo que ofrece una conectividad rápida a Internet en conexiones lentas.

Aun así, Opera es un navegador de código cerrado y tiene menos actualizaciones que el resto de sus competidores.

<https://www.opera.com/es>

Safari

Este navegador está desarrollado exclusivamente por Apple. Se lanzó en 2003 y, al igual que Internet Explorer, viene instalado, por defecto, en todos los dispositivos de Apple. Como curiosidad, antes de que Safari fuese el navegador predeterminado en los sistemas Apple, la empresa de Steve Jobs usaba el navegador de Internet Explorer de Windows.

Al igual de Google Chrome, Safari permite sincronizar automáticamente con otros dispositivos Apple a través de iCloud, lo cual mejora la experiencia de usuario.

A pesar de su diseño sencillo, su velocidad y gran usabilidad, tiene más problemas que ventajas: su motor de búsqueda está desactualizado lo que conlleva a fallas de seguridad, no es compatible con la mayoría de sistemas y no ofrece una gran variedad de extensiones y funcionalidades para enriquecer la experiencia de usuario.

<https://www.apple.com/es/safari/>

Existen alternativas a todos estos navegadores estándar que ofrecen otras características interesantes.

Brave

Este navegador es de los mismos creadores del proyecto Mozilla. Es compatible con todos los sistemas operativos y para los dispositivos móviles iOS y Android. Destaca por las opciones de seguridad que ofrece a sus usuarios: bloqueo de publicidad, *trackers* y *phising*, control de cookies y visualización de páginas exclusivamente con el protocolo de seguridad HTTPS. Permite visualizar todos los elementos bloqueados sospechosos o peligrosos y bloquea la publicidad intrusiva.



Es una alternativa rápida, segura, eficiente y además en español.

<https://brave.com/es/>

Maxthon

Este navegador destaca por ser especialmente sencillo, tanto que casi no tiene opciones solo la de navegar. Está pensado para usuarios que solo quieren navegar y no quieren distracciones, esto lo convierte en un navegador muy liviano que consume pocos recursos de sistema.

Maxthon es, además, compatible con todos los sistemas, pero solo está disponible en inglés.

<https://es.maxthon.com/>

Torch

Este navegador está diseñado para visualizar contenido multimedia. Dispone del sistema de descarga de archivos Torrent integrado que permite la descarga de archivos de audio y vídeo. Torch dispone

de una plataforma de videojuegos gratuitos llamada Torch Games para jugar online sin necesidad de descargar.

Además, Torch permite diseñar el perfil personal de Facebook a través de plantillas o creando una personal.



<https://torchbrowser.com/>



TOME NOTA

Es importante elegir un navegador que proteja los datos personales para evitar robos de información sensible a través de técnicas como el phising. El protocolo HTTPS es otro aspecto clave a tener en cuenta para confiar en la seguridad de un sitio web.

1.3.3. Diferencias de visualización

Un mismo sitio web puede tener una apariencia muy diferente en los diferentes navegadores. Un sitio web no deja de ser un código que entra a un servidor y es interpretado a través de un navegador. Por ello, uno de los desafíos del desarrollador web y diseñador web es que el sitio web se visualice de la misma manera en los diferentes navegadores, al menos, en los más importantes.

Las pequeñas diferencias de interpretación del código de los navegadores pueden llegar a hacer que un sitio web no se vea en su totalidad o que ciertas funciones no aparezcan. Es importante que todo el sitio web funcione en su totalidad en los diferentes navegadores para llegar al mayor número de usuarios.

Ejemplo de visualización:

Visualización en Microsoft Edge:



Misma página en Google Chrome



Diseño web

Necesitas una página web para tu comercio, un evento, o tu marca personal, nosotros estamos contigo desde que empiezas con la idea, ideas frescas e innovadoras y los consejos de nuestros expertos, tendrás asesoramiento desde el minuto 1.

[VER PORTFOLIO](#)



Mantenimiento

Terminarlos de hacer tu sitio web ¿Y ahora qué? Nos encargamos de que tengas la versión más actualizada de tu plataforma web.

Tenemos también planes por horas para que hagamos actualizaciones de contenido.

[VER PLANES](#)



Redes sociales

Desarrollamos estrategias de Marketing para que administres tus Redes Sociales y tengas éxito.

O también podemos hacerlo nosotros, nos encanta y además se nos da bien.

[VER PORTFOLIO](#)

1.4. Marcas para dar formato al documento

En este epígrafe, aprenderá cómo dar formato al texto de los documentos web utilizando, para ello, el lenguaje de marcas HTML. Es importante que sepa que desde que aparecieron las hojas de estilo CSS, esta función de dar formato a los elementos de los documentos web, se ha visto cada vez más distanciada del HTML y, para ello, se ha utilizado CSS. Por lo que, hoy en día, el HTML tan solo se utiliza para crear la estructura de sus documentos web, mientras el CSS se utiliza para dar estilo a los elementos.

Sin embargo, si se habla de cómo interpretan los navegadores los estilos si en HTML o CSS, se puede decir que aún muchos navegadores son capaces de interpretar el formato que aporta HTML a los elementos, por lo que no tendrá ningún problema a la hora de emplearlos para ese fin. Por lo tanto, reuniendo estos conocimientos se puede afirmar que una página web es una suma de un documento HTML y un documento CSS.

A continuación, vea cómo insertar saltos de línea, párrafos y espacios en blanco. Algo que es realmente sencillo y ayudará a formatear la página web.

Saltos de línea

Para añadir saltos de línea en un documento HTML, debe insertar la etiqueta
. En caso de que añada los saltos de línea dentro del propio código, el navegador los ignorará. Vea un ejemplo donde se comprende mejor este hecho:

```
<html><head>
```

```
<head>
```

```
<body>
```

Esto es

Un ejemplo

```
</body>
```

```
</html>
```

Si intenta ejecutar este código en un navegador, verá que el salto de línea que ha escrito de manera

Esto es un ejemplo

natural no es lo que mostrará el navegador. Por lo que el navegador mostrará lo siguiente:

En cambio, si escribe la misma frase incluyendo la etiqueta de salto de página, el navegador mostraría la frase de forma correcta.

Vea el ejemplo de código:

```
<html>
```

```
<head>
```

```
<head>
```

```
<body>
```

Esto es
 un ejemplo

```
</body>
```

```
</html>
```

Vea lo que mostraría el navegador:

Esto es
un ejemplo

Espacios en blanco

Con los espacios en blanco, ocurre lo mismo que con los saltos de línea. Si decide insertar espacios en blanco en un documento sin su etiqueta correspondiente, ocurrirá que el navegador ignorará ese espacio. Para insertar espacios en blanco, debe utilizar la cadena s;. Vea un ejemplo a continuación:

```
<html>
```

```
<head>
```

```
</head>
```

```
<body>
```

Esto s; s; s; s;es un ejemplo.

```
</body>
```

```
</html>
```

Lo que muestra el navegador es lo siguiente.

Esto es un ejemplo.

Párrafos

Los párrafos son estructuras donde se agrupan los textos del documento. En un documento HTML para indicar las limitaciones de esas estructuras, se utiliza las etiquetas <p> y </p>. Vea un ejemplo de párrafo:

```
<html>
```

```
<head>
```

```
</head>
```

```
<body>
```

```
<p>Esto es un ejemplo de párrafo. Dentro de esta estructura puede escribir todo el texto que quiera. En caso de que quiera crear más estructuras como estas, tan solo, debe añadir más etiquetas de párrafo</p>
```

```
</body>
```

```
</head>
```

Al escribir un párrafo en su documento, el navegador será el encargado de establecer una separación mínima entre este elemento y el elemento anterior o posterior que haya en el documento. Por ejemplo, en el caso en que tuviéra más de un párrafo podría observar este comportamiento. Vea un ejemplo:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
</head>
```

```
<body>
```

```
<p>Este es el primer párrafo</p><p>Este es el segundo párrafo</p><p>Este es el tercer párrafo</p>
```

```
</body>
```

```
</html>
```

Lo que mostrará el navegador será lo siguiente:

Este es el primer párrafo

Este es el segundo párrafo

Este es el tercer párrafo

1.4. 1. *Marcas de inicio y final*

En HTML5 existen diferentes etiquetas que permiten, de una manera más precisa, estructurar el contenido de la página web. Para ello, HTML5 cuenta con nuevas etiquetas para poder diferenciar, delimitar e identificar de manera más exacta los contenidos, cosa que no pasaba en las versiones antiguas de HTML. Las etiquetas que permiten realizar esta ordenación del contenido son: <nav>, <header>, <footer>, <section>, <article>, <address> y <aside>.

Como se mencionó anteriormente, lo que se propone HTML5 con estas nuevas etiquetas es separar de una manera limpia y eficiente la estructura del documento de los estilos del mismo. Esta es la razón por la que en esta versión de HTML puede encontrar estas nuevas etiquetas. A continuación, se aprenderá para qué puede utilizar estas etiquetas:

- **<header>**

Esta etiqueta se utiliza para delimitar el contenido que va a ser parte de la cabecera de un documento web. La cabecera es donde el documento HTML va a contener la versión del documento, los títulos, los subtítulos, la imagen de título y los elementos de navegación como <nav>. A continuación, vea la sintaxis de este elemento:

<header>Aquí establece el contenido de la cabecera de la página</header>

No podrá insertar esta etiqueta en otra etiqueta <header>, ni en el <footer>, ni tampoco en <address>.

- **<nav>**

Se utiliza esta etiqueta para establecer los elementos de navegación de su documento HTML. Por ejemplo, dentro de esta etiqueta, se establece el menú principal de su página, el menú lateral, etc. Para utilizar esta etiqueta debe utilizar la siguiente sintaxis:

<nav>Aquí establece los elementos de navegación de su web</nav>

- **<article>**

La utilidad de esta etiqueta es delimitar un contenido que es externo a su página web. Por ejemplo, un artículo de prensa, un texto que haya utilizado de otra web, etc. Para utilizar esta etiqueta debe usar la siguiente sintaxis:

<article>Aquí añadirá el contenido externo que quiera utilizar en su web</article>

- **<section>**

Al usar esta etiqueta, puede delimitar las áreas del documento HTML. Esta delimitación se utiliza porque, por alguna razón, quiera separar esa zona de las demás zonas que contenga el documento. Para utilizar esta etiqueta se emplea la siguiente sintaxis:

<section>Esta zona está separada de las demás</section>



TOME NOTA

Si quiere dar estilo a las áreas separadas por la etiqueta <section>, debe emplear para ello código CSS.

<aside>

Se utilizará esta etiqueta con la finalidad de delimitar una zona del documento en la que se añade un elemento que complemente o aclare los demás contenidos de la página principal. Para utilizar esta etiqueta, se emplea la siguiente sintaxis:

<aside>Este elemento complementa al resto</aside>

- <footer>

Esta etiqueta sirve para establecer los elementos del pie de página en su interior. La sintaxis de la misma es:

<footer>Pie de página</footer>

- <address>

Como ya sabe, las páginas web que están publicadas en Internet, necesitan contener información de contacto, ya sea, del autor, de la empresa, etc. Aquí es donde la etiqueta <address> encuentra su utilidad, ya que, dentro de ella es donde se añade esta información de contacto.

Debe tener en cuenta que, cuando añada este tipo de etiqueta a su documento, es importante saber si la información de contacto va a ser relativo a una de las etiquetas <article>, la cual ya se vio anteriormente, o va a ser relativa a los propios contenidos de la página, los cuales se almacenan en el <body>. Debe recordar que la etiqueta <article> se utiliza para delimitar un contenido que pertenezca a un sitio externo de su web, es decir, que el autor de ese contenido no es usted.

En caso de que emplee la etiqueta <address> al contenido del <body>, la información de contacto se aplicará a todo el contenido del documento web.

Por otro lado, si quiere añadir información de contacto sobre el autor de la web, se añade la etiqueta <address> en el pie de página o <footer>. Para utilizar esta etiqueta se sigue la siguiente sintaxis:

<address>Información de contacto</address>

Para finalizar, vea un ejemplo de cómo quedaría un documento HTML que utilice todas las etiquetas que ha aprendido en este epígrafe. Vea el ejemplo:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Documento Web HTML5</title>
```

```
</head>

<body>

<header>

    <nav></nav>

</header>

    <section>

        <article></article>

        <address></address>

    </section>

    <aside></aside>

    <footer></footer>

</body>

</html>
```



TOME NOTA

Es muy importante tener en cuenta que para poder visualizar un documento que utilice las nuevas etiquetas de HTML5, el navegador donde vaya a visualizarlo, sea compatible con la versión HTML5.

1.4.2. Marcas de aspecto

En este epígrafe, se verán algunas de las etiquetas que se utilizan para diferenciar el tipo de texto que ellas contienen:

<abbr>

Utilice esta etiqueta para indicar que el texto que está estableciendo es una abreviatura. Para indicar el significado de la abreviatura, se emplea *title* para que cuando ponga el

puntero del ratón encima del mismo, pueda ver el significado. Vea un ejemplo de uso de esta etiqueta:

```
<html>
<head>
</head>

<body>
    <p>El código <abbr title="HyperText Markup Language">HTML</abbr>es el que se
utiliza para dar estructura a un documento web.</p>
</body>
</html>
```

Ejemplo de lo que visualizaría en el navegador:

El código HTML es el que se utiliza para dar estructura a un documento web.

HyperText Markup Language

<mark>

Esta etiqueta se utiliza para marcar texto, de manera que, el texto se mostrará destacado con un color de fondo.

<kbd>

Esta etiqueta se utiliza cuando quiera que un usuario introduzca un texto por teclado. Este texto que usuario introduzca, se albergará dentro de esta etiqueta, la cual mostrará ese texto con una fuente diferente a la del resto del documento.

<ins>

Se utiliza esta etiqueta para subrayar un texto.

<i>

Se utiliza esta etiqueta para mostrar un texto en cursiva.

Esta etiqueta se utiliza para enfatizar el texto. El texto contenido en esta etiqueta se mostrará en cursiva. La diferencia que tiene esta etiqueta con la anterior es que los navegadores por voz son capaces de interpretar que ese texto va con énfasis, en cambio, si se utiliza solo la etiqueta *<i>*, los navegadores con voz no serán capaces de interpretar el énfasis.

<dfn>

Se utiliza esta etiqueta para mostrar definiciones. El texto que haya dentro de esta etiqueta se mostrará en cursiva.

Esta etiqueta se utiliza para tachar el texto. Por lo que, el texto contenido en ella se visualizará tachado.

<code>

Esta etiqueta se utiliza para añadir código y que se muerte con un tipo de letra fijo.

<cite>

Esta etiqueta se utiliza para escribir títulos de películas, libros, trabajos, canción, etc.

Esta etiqueta se utiliza para poner el texto en negrita.

<var>

Esta etiqueta indica que los datos que contiene son una variable o instancia de un programa. Estos datos se mostrarán en cursiva.

<time>

Se utiliza esta etiqueta para establecer fechas y horas en un formato que pueda ser interpretado por los navegadores. Estas fechas podrán incluirse en el atributo datetime.

<sup>

Esta etiqueta se utiliza para añadir superíndices.

<sub>

Esta etiqueta se utiliza para añadir subíndices.

<small>

El texto que contenga dentro de esta etiqueta, se mostrará con un tamaño de letra más pequeño que el resto. Esta etiqueta se suele utilizar, por ejemplo, para añadir copyright.

<samp>

Esta etiqueta se utiliza para mostrar textos que provienen de programas, *scripts*, etc.

<q>

Esta etiqueta se utiliza cuando se quiere mostrar un texto citado, por lo que el texto que haya dentro de esta etiqueta se mostrara entre comillas.

1.4.3. *Marcas de párrafo*

Las principales marcas de párrafo son las cabeceras. Existe un grupo de etiquetas que se utilizan para establecer el tamaño y, por consiguiente, nivel de importancia del texto. Las etiquetas son: `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` y `<h6>`. El nivel de importancia va desde `<h1>` siendo el más importante, hasta `<h6>` siendo el menos importante.

Estas etiquetas se comportan de la misma forma que la etiqueta `<p>` de los párrafos, puesto que, cada vez que se abre o se cierra una etiqueta `<hx>`, automáticamente se añadirá un salto de página. La diferencia más sustancial entre las etiquetas de cabecera y la de párrafo, es que las etiquetas de cabecera establecen un estilo por defecto en el texto que estas contiene. De este modo es como se establece la importancia de las diferentes cabeceras.



Aunque las etiquetas `<hx>` establezcan un estilo predefinido, se pueden cambiar utilizando estilos CSS.

TOME NOTA

Vea a continuación un ejemplo donde se emplean todas estas etiquetas de cabecera:

```
<!DOCTYPE html>
```

```
<html>  
<head>  
</head>  
<body>  
<h1>Título H1</h1>  
<h2>Título H2</h2>  
<h3>Título H3</h3>  
<h4>Título H4</h4>  
<h5>Título H5</h5>  
<h6>Título H6</h6>  
</body>  
</html>
```

Vea cómo se muestra en el navegador:

Título H1

Título H2

Título H3

Título H4

Título H5

Título H6

1.4.4. *Marcas de fuentes y colores*

Anteriormente a la existencia de HTML5, debe utilizar la etiqueta `` si quería establecer un tipo de fuente y color para el texto. La etiqueta `` crea un bloque de texto en el que poder aplicar el

tipo de letra y el color de la misma. Para ello, debe utilizar sus dos atributos *face* y *color*, que va a ver a continuación:

- **Face:** Este atributo permite establecer el tipo de fuente con el que se visualizará el texto. Al establecer este atributo, es muy frecuente indicar más de una fuente separada por una coma. La razón de realizarlo así es para que el navegador escoja la primera fuente disponible que tenga el usuario en su ordenador, ya que, no todos los usuarios tendrán las mismas fuentes instaladas.
- **Color:** Este atributo permite aplicarle un color determinado a un texto. Para aplicarle color a un texto pueda hacerlo de dos maneras.
 - La primera es utilizar el nombre del color en inglés, por ejemplo, white, orange, purple.
 - La segunda, es establecer el valor del color en hexadecimal, por ejemplo, #FFF, #000.



TOME NOTA

La fuente que se establece en los documentos HTML por defecto, dependerá de la configuración que tenga el usuario en su navegador.

Vea a continuación un ejemplo:

```
<!DOCTYPE html>

<html>

<head>
</head>

<body>
<p>Esto es un párrafo con la letra por defecto del navegador</p>

<p><font face="Arial, Helvetica" color="purple">Esto es un párrafo con la letra Arial en color morado.</font></p>

</body>

</html>
```

Vea cómo lo muestra el navegador:

Esto es un párrafo con la letra por defecto del navegador

Esto es un párrafo con la letra Arial en color morado.

Lo siguiente que puede ver es una tabla donde se reúnen los colores más básicos de HTML los cuales puede utilizar por su nombre en inglés o por su código hexadecimal:

Nombre	Color	Código
black		#000000
teal		#008080
blue		#0000FF
navy		#000080
lime		#00FF00
white		#FFFFFF
purple		#800080
yellow		#FFFF00
olive		#808000
red		#FF0000
maroon		#800000
gray		#808080
fuchsia		#FF00FF
green		#008000
silver		#C0C0C0
aqua		#00FFFF

Como se decía anteriormente, la etiqueta `` era la encargada de establecer el tamaño y el color del texto. Sin embargo, con la nueva versión de HTML5 se usan las hojas de estilo para este cometido.

Aunque, puede seguir utilizando la etiqueta ``, no es conveniente hacerlo, puesto que, se encuentra como etiqueta obsoleta y podría dar problemas en muchos navegadores. Vea a continuación cómo conseguiría el resultado del ejemplo anterior, pero utilizando el nuevo método de tamaño y color con CSS:

```
<!DOCTYPE html>

<html>

<head>
```

```
</head>

<body>

<p>Esto es un párrafo con la letra por defecto del navegador</p>

<p style="font-family:Arial, Helvetica;color:purple">Esto es un párrafo con la letra Arial en color morado.</p>

</body>

</html>
```

Lo que mostraría el navegador sería lo siguiente:

Esto es un párrafo con la letra por defecto del navegador

Esto es un párrafo con la letra Arial en color morado.



TOME NOTA

Se puede seguir utilizando la etiqueta aunque haya desaparecido en la versión HTML5. Sin embargo, no se aconseja su uso porque en un futuro puede dar problemas debido a su obsolescencia.

1.4.4. 1. Listas ordenadas

En este epígrafe va a aprender qué son las listas y cómo usarlas. En HTML las listas son un recurso muy útil e interesante, ya que, permiten clasificar y organizar el texto. Existen dos tipos de listas más importantes, las listas ordenadas y las no ordenadas. Vea a continuación cómo utilizar ambos recursos en documentos web.

- **Listas no ordenadas**

El conjunto de listas no ordenadas son las que aparecen precedidas con una viñeta de tipo cuadrado, guion, punto, flecha, etc. Para implementar este tipo de lista en su documento web, se utiliza la etiqueta ``, para abrir, y `` para cerrar. Dentro de estas etiquetas será donde se establezca cada uno de los elementos de la lista. Para ello, se aplica la etiqueta ``, para abrir y `` para cerrar. Vea un sencillo ejemplo de cómo implementar una lista no ordenada en su página web:

```
<!DOCTYPE html>
```

```
<html>

<head>

</head>

<body>

<p>Lista no ordenada</p>

<ul>

<li>Elemento 1</li>

<li>Elemento 2</li>

<li>Elemento 3</li>

<li>Elemento 4</li>

</ul>

</body>

</html>
```

Lo que mostraría el navegador sería lo siguiente:

Lista no ordenada

- Elemento 1
- Elemento 2
- Elemento 3
- Elemento 4

Como puede ver el tipo de viñeta predeterminado es el círculo. Sin embargo, puede cambiar el tipo de viñeta usando código CSS. Además, también existe un atributo que ya está obsoleto según la versión HTML5, que es el atributo *type*, el cual puede contener los siguientes valores: “square” para cuadrado, “circle” para círculo y, por último, “disc” para disco.

Por otro lado, si quiere anidar listas, es decir, meter una lista dentro de otra lista, tan solo, debe saber que un elemento dentro de una lista, puede ser otra lista. Vea a continuación, un ejemplo de listas anidadas:

```
<!DOCTYPE html>
```

```
<html>

<head>

</head>

<body>

<p>Lista no ordenada</p>

<ul>

<li>Lista 1 dentro de lista</li>

<ul>

<li>Elemento 1</li>

<li>Elemento 2</li>

</ul>

<li>Lista 2 dentro de lista</li>

<ul>

<li>Elemento 1</li>

<li>Elemento 2</li>

</ul>

</ul>

</body>

</html>
```

Esto es lo que mostraría el navegador:

Lista no ordenada

- Lista 1 dentro de lista
 - Elemento 1
 - Elemento 2
 - Lista 2 dentro de lista
 - Elemento 1
 - Elemento 2
-
- **Listas ordenadas**

La diferencia que se encuentra al usar listas ordenadas es que cada elemento de la lista está precedido por un número, en vez de por viñetas. Para establecer listas ordenadas en su documento, se utiliza la etiqueta ``, mientras los elementos de la lista irán encerrados entre etiquetas ``, al igual que pasaba en las listas anteriores. Vea a continuación un ejemplo de cómo utilizar estas listas en su documento HTML:

```
<!DOCTYPE html>

<html>

<head>

</head>

<body>

<p>Lista ordenada</p>

<ol>

<li>Elemento 1</li>

<li>Elemento 2</li>

<li>Elemento 3</li>

</ol>

</body>

</html>
```

En el navegador se mostrará de la siguiente manera:

Lista ordenada

1. Elemento 1
2. Elemento 2
3. Elemento 3

Como se comentaba anteriormente, con código CSS puede modificar el tipo de viñeta o de numeración que va a utilizar su lista. Además, también puede utilizar el atributo `type` para el mismo cometido. Este atributo puede adquirir los siguientes valores:

- Numeración estándar como, por ejemplo: 1,2,3,4...
- Numeración en números romanos en minúsculas como, por ejemplo: i, ii, iii, iv, ...
- Numeración en números romanos en mayúsculas como, por ejemplo: I, II, III, IV, ...
- Numeración alfabética como, por ejemplo: a, b, c, d, ...
- Numeración alfabética en mayúsculas como, por ejemplo: A, B, C, D, ...

Además, al igual que las listas no ordenadas, estas también pueden anidarse entre ellas o con otras listas no ordenadas.

1.5. Enlaces y direccionamientos

Los enlaces son los elementos más importantes que componen la *World Wide Web*, son los que conectan una información con otra hasta conformar la vasta red de Internet.

En el caso de los sitios web, los enlaces y direccionamientos componen y enriquecen la complejidad del contenido, haciendo que el usuario pueda saltar de un contenido a otro tanto dentro del propio sitio web como con contenidos externos al sitio web visitado.

El funcionamiento de los enlaces y direccionamientos es algo sencillo: un elemento se programa para que tenga la funcionalidad de redirigir al interactuar con él a través de los diferentes eventos que ya ha visto como, por ejemplo, los de ratón (`onclick`, `onmouseover`, etc.). Lo más común es que un botón, una palabra o una imagen al ser pulsada con el cursor del ratón dirija a otro contenido, ya sea interno en el mismo sitio web o externo.

1.5.1. Creación de un enlace

La etiqueta que se utiliza para crear un enlace en HTML es `<a>`. Dentro de esta etiqueta puede poner múltiples atributos para indicar qué elemento quiere convertir en enlace, ya sea texto, imágenes, botones, etc.

Los atributos más utilizados son:

- `href`: Con este atributo se indica la dirección asociada al enlace.
- `title`: En este atributo puede definir el texto que desea que aparezca junto al enlace cuando el ratón se coloque encima.

- `target`: Este atributo refiere al lugar donde se abrirá el contenido del enlace. Este atributo, a su vez, puede contener otros valores para hacer más específica la interactividad del evento.
- `_blank`: Indica que el enlace abrirá el documento en una ventana o pestaña nueva.
- `_parent`: Indica que el documento se abrirá en la misma ventana o pestaña que está visualizando.
- `_self`: El documento se mostrará en la ventana donde se encuentra el enlace.
- `_top`: Se usará todo el espacio que ocupa la ventana del navegador para mostrar la información del enlace.

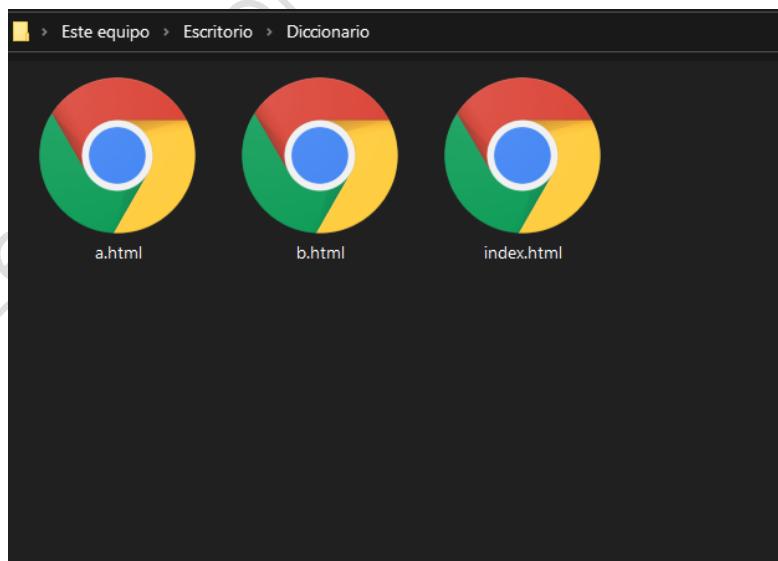
Este es un ejemplo de cómo escribir en HTML un enlace para dirigirse a la página de Google en una nueva pestaña:

```
<a href="http://www.google.es" title="Enlace a Google" target="_blank">Enlace a la página principal de Google</a>
```

1.5.2. Tipos de enlace

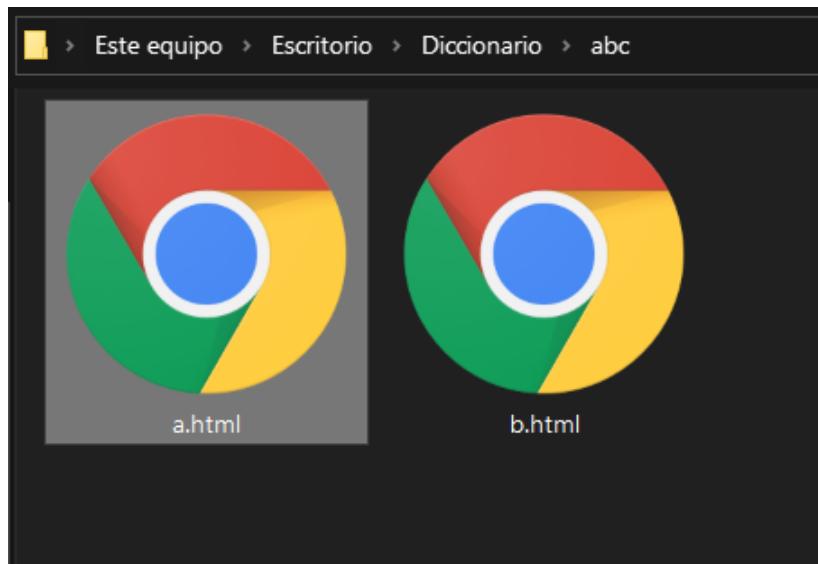
Existen varios tipos de enlaces en HTML para enlazar múltiples tipos de archivos o para definir el tipo de acción que sucede al pulsarlos. Estos pueden ser para enlaces locales, enlaces internos (anclas), enlaces externos, enlaces a direcciones de correo y enlaces a archivos.

En la estructura básica para llamar a un archivo dentro del mismo directorio, página o externamente siempre se usa la etiqueta `<a>` y el atributo `href`. También debe tener en cuenta dónde se encuentran los archivos que llama, si en la misma carpeta donde está el `index.html` o en subcarpetas. Imagine que está haciendo un diccionario y tiene divididos los archivos por cada letra del abecedario.



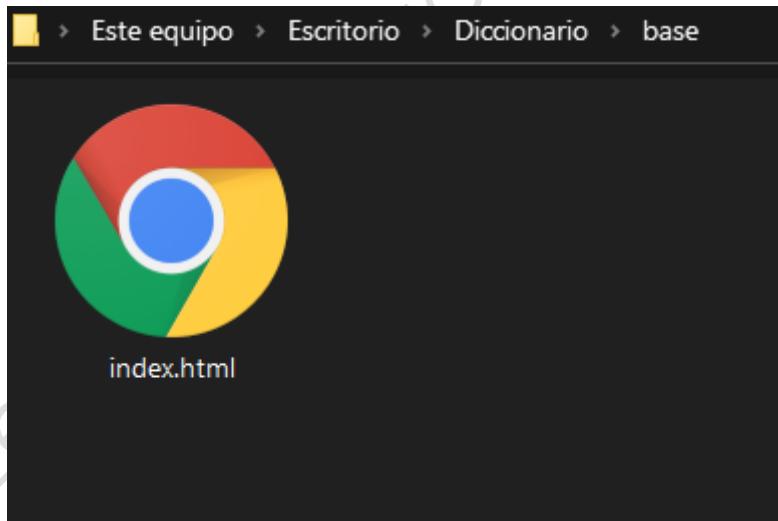
En este caso, si el `index.html` se encuentra en el mismo nivel que un archivo llamado “`a.html`”, la ruta sería:

```
<a href="a.html">Vínculo a A</a>
```



Sin embargo, si mete todos los archivos en una carpeta llamada “abc”, quedando el index.html fuera, la ruta sería:

```
<a href="abc/a.html">Vínculo a A</a>
```



Para dirigir a un documento que se encuentre en un nivel superior al index.html, debe indicarlo con “..” por cada nivel que tenga que subir para llegar al archivo que interesa.

```
<a href="../a.html">Vínculo a A</a>
```

Ancillas

Los enlaces internos son un recurso muy útil para crear una página dinámica. Los enlaces, o anclas, pueden dirigir al usuario de una sección a otra dentro del mismo documento. Este tipo de ancla puede verlo en multitud de páginas web que, cuando llegue al final de su contenido, aprecia un ícono

de flecha apuntando hacia arriba que, al pulsarlo, lleva de nuevo a la parte superior de la página. Además de esta utilidad, puede usarlo para vincular con secciones específicas de la página, por ejemplo, menús concretos, artículos, noticias, etc.

Para crear estas anclas necesita usar el atributo *name* en la etiqueta que indica el destino, y el atributo *href* de otra etiqueta en el que debe escribir el nombre del destino anteponiéndole “#”.

```
<!DOCTYPE html>

<html>

<head>
    <title>Este es un ejemplo de ancla</title>

</head>

<body>
    <a name="inicio">Principio de página</a>
    <p> | |<br></p>
    <p> V <br></p>

    <a href="#inicio">Subir</a>
</body>

</html>
```

Ejemplo de ancla. En este caso se asigna “inicio” al *name* del Principio de página e “#inicio” al *href* de la segunda etiqueta del final del documento.

Vínculos

Dispone de varios tipos de vínculos. Los vínculos externos son aquellos que direccionan a documentos externos a la página que está visualizando. Por ejemplo, si quiere que la página contenga enlaces a otras páginas web, debe añadir el protocolo http en el atributo href.

[Ir a Google](https://www.google.es)

Otro tipo de vínculo muy usado y útil es el vínculo a direcciones de correos. En la sección “Contacto” de multitud de páginas web puede ver este tipo de enlace que, al pulsar, puede seleccionar el asistente de correo electrónico que tenga instalado en su ordenador para enviar un correo electrónico. Los asistentes de correo electrónico pueden ser Mozilla Thunderbird, IncrediMail, Outlook, Gmail, etc.

jContáctanos!

Además de los atributos base para definir el vínculo, puede añadir otros como, por ejemplo, el asunto del mensaje cuya sintaxis se escribe con "?". También es posible indicar el otro correo electrónico en copia, para ello es necesario incluir "&".

Contáctanos

El vínculo a archivos se hace a través del mismo atributo *href* para indicar la ruta donde se encuentra. Recuerde que para indicar directorios superiores se escriben tantos "../" como directorios tenga que subir.

Descargar

A continuación, podrá ver en este vídeo, de unos ocho minutos de duración, el procedimiento de creación y navegación a través de las anclas:



1.5.3. Los enlaces y la navegación

El tema de los enlaces en los documentos web es algo fundamental para desarrollar una página web, ya que, estos elementos son cruciales para cualquier tipo de página. Por ejemplo, para crear un menú necesita enlaces para que cuando el usuario pulse la opción deseada, sea redirigido hacia la página correcta, para una tienda online también necesita enlaces, e incluso para un buscador como Google. Por lo tanto, sin estos elementos sería imposible crear cualquier tipo de página web. Puesto que, en vez de una página web lo que obtendría es una imagen con la que no podría interactuar.

Un enlace es cualquier elemento de un documento web en el que si pulsa sobre él, va a llevar hacia otra dirección web o recurso. Esto significa que cuando pulse el enlace, el navegador va a redirigir a otra URL que, previamente, estará asociada a ese enlace.

Por ejemplo, un enlace puede ser un botón, puede ser una palabra que esté enlazada a otra url, también puede incluir imágenes como enlace, etc. En definitiva, tiene múltiples opciones de enlace.

1.6. Marcos y capas

Los marcos y las capas son elementos para diseñar la estructura del sitio web. Los marcos, más conocidos por *frames*, han sido elementos básicos desde los orígenes del desarrollo web.

En el siguiente vídeo, de tres minutos de duración, se describe de forma rápida y sencilla qué son las capas:



1.6. 1. Marcos

Los marcos crean áreas independientes que subdividen el documento HTML. En estas áreas se pueden cargar otras páginas web, es decir, en la misma página web que está visualizando puede cargar otras páginas dentro de la misma composición de la página web origen.

Cabe mencionar que este tipo de etiquetas han quedado obsoletas en nuevo HTML5, aunque muchos de los navegadores modernos siguen soportando e interpretando correctamente estas directivas.

En el siguiente vídeo, de diez minutos de duración, conocerá la forma de utilizar los marcos en HTML:



1.6. 1. 1. Creación de marcos

Existen dos tipos principales de marcos, se utilizarán las etiquetas `<frame>` e `<iframe>` para crearlos. Vea a continuación cómo se crean y para qué se utilizan cada uno de ellos:

- **Frames**

Este tipo de marcos se utilizan para dividir la página en varias zonas de navegación donde puede cargar en ellas diferentes documentos HTML. Para añadir estos elementos a su documento HTML debe utilizar las etiquetas `<frameset>` y, dentro de estas etiquetas es donde se establecen las etiquetas `<frame>`.

Debe tener en cuenta que cuando usa los *frames*, no incluirá la etiqueta `<body>`, sino `<frameset>`.

El siguiente ejemplo utiliza el documento main.html, el cual divide en tres partes que contendrán el doc1.html, doc2.html y doc3.html:

```
<!DOCTYPE html>

<html>

<head>

<title>Ejemplo frames</title>

</head>

<frameset cols="30%, 70%">

<frameset rows="50%, 50%">

<frame src="doc1.html"/>

<frame src="doc2.html"/>

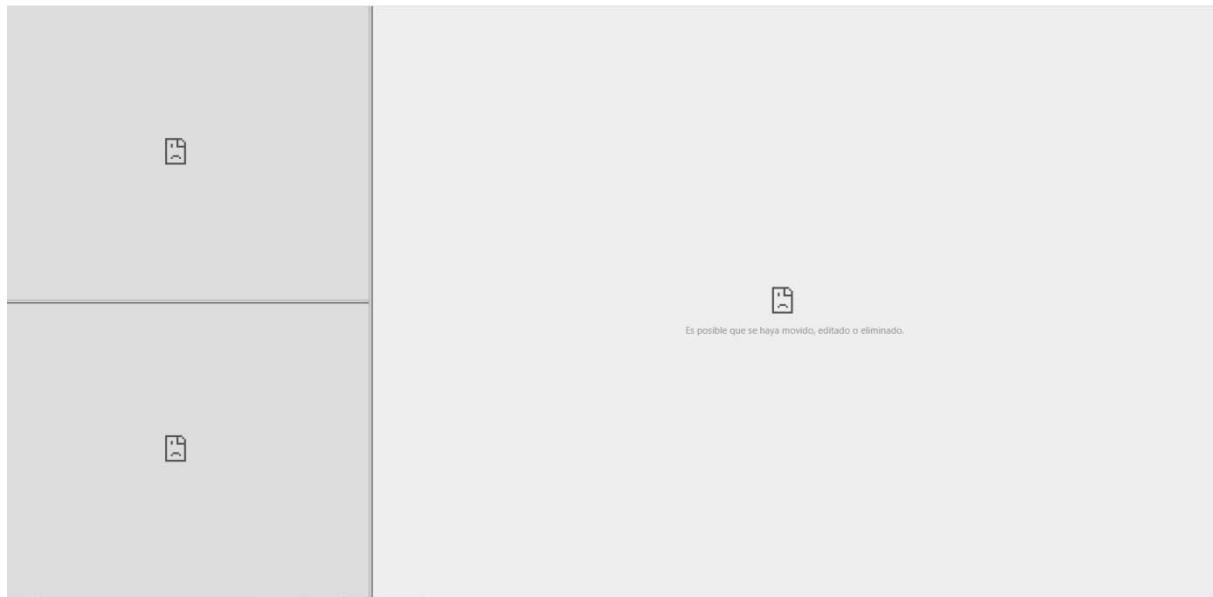
</frameset>

<frame src="doc3.html"/>

</frameset>

</html>
```

Como los archivos que ha enlazado no son archivos reales, el navegador mostrará las diferentes partes como si el archivo estuviera roto. Sin embargo, es importante que vea el ejemplo de cómo muestra el navegador la división de este documento:



Como puede observar en el ejemplo anterior, en la etiqueta <frameset> se han establecido dos columnas de 30% y 70% que sirven para dividir el área del documento main.html en dos partes una del 70% y la otra del 30%. También podría haber dividido ambas partes en píxeles en vez de en porcentajes, sin embargo, hubiera tenido problemas a la hora de redimensionar la ventana del navegador, ya que los píxeles toman un tamaño fijo.

Por otro lado, el área más grande que es la que se ha dividido en el 70% del ancho total, se define con la etiqueta <frame src="doc3.html"/>. Como puede observar, cada archivo se carga en base al atributo src de cada etiqueta <frame>.

En cuanto a la columna más pequeña, la cual es del 30% del ancho total, queda definida con el siguiente trozo de código:<frameset rows="50%, 50%">

```
<frame src="doc1.html"/>  
  
<frame src="doc2.html"/>  
  
</frameset>
```

Como puede observar, esta columna se ha dividido, a su vez, en dos áreas más pequeñas que toman cada una el 50% del ancho total de la columna y contendrán los documentos doc1.html y doc2.html.

- **Iframes**

Por otro lado, tiene los *iframes* que se utilizan para crear áreas flotantes donde puede albergar diferentes documentos HTML, al igual que pasaba con los *frames*. Para establecer este elemento en su documento utiliza la etiqueta <iframe> la cual puede tener los siguientes atributos:

- **Name:** Este atributo establece el nombre del iframe.

- **Scrolling:** Este atributo establece si el iframe posee o no barra de scroll. Las dos opciones que puede elegir son yes o no.
- **Src:** Este atributo, como ha visto anteriormente, indica una ruta donde se encuentra un elemento a cargar.
- **Sandbox:** Este atributo permite restringir el acceso a determinados contenidos web que contiene el iframe.
- **Width:** Este atributo establece el ancho del iframe.
- **Height:** Este atributo establece la altura del iframe.



TOME NOTA

La única etiqueta que soporta HTML5 para establecer marcos es la etiqueta <iframe>. Sin embargo, los navegadores más actuales y conocidos sí son capaces de soportar las demás etiquetas.

1.6.1.2 Tipos de marcos

Existen dos tipos de marcos, los frames e iframes.

Los frames son marcos que dividen la página web en zonas donde se cargan otros documentos HTML. Para ello, se utiliza la etiqueta `<frameset>` como etiqueta madre y sustituye a la etiqueta `body`, es decir, en lugar de usar la estructura `<body></body>`, se usa `<frameset></frameset>` como base. Dentro de estas etiquetas frameset se usan las etiquetas `<frame>` para crear esas zonas de carga. Las etiquetas `frame` tienen autocierre, es decir, no tienen otra etiqueta de cierre, por lo tanto se escribe así: `<frame src="documento.html" />`.

Ejemplo:

```
<frameset cols="30%, 70%">  
  <frameset rows="50%, 50%">  
    <frame src="a.html" />  
  
    <frame src="b.html" />  
  
  </frameset>  
  
<frame src="c.html" />  
  
</frameset>
```

En el ejemplo se puede apreciar que para enlazar a otro documento exterior se usa el atributo src y para definir la estructura de los frames usa cols y rows, es decir, columnas y filas.

Por otro lado, los *iframes* son áreas flotantes en las que carga otros documentos. La etiqueta iframe puede contener otros atributos a su vez como:

- name: Le asigna nombre al iframe.
- src: Indica la ruta donde está el documento local o la URL de la web que se cargará en el marco.
- width: Indica el ancho en píxeles del iframe.
- height: Indica el alto en píxeles del iframe.
- sandbox: Con este atributo se puede especificar restricciones de seguridad en el acceso a los contenidos de la web que se cargará en el iframe.
- scrolling: Este atributo indica si el nuevo marco tendrá scroll o no.

```
<iframe src="https://www.google.es" width="400" height="300" scrolling="Yes">
```

1.6.1.3. Situación de los marcos

Actualmente, el uso de marcos se encuentra obsoleto. Esto se debe a múltiples razones como que los motores de búsqueda no son capaces de indexar de manera correcta una página que contenga marcos, ocupan demasiado espacio en pantalla, no se pueden utilizar las opciones de ir hacia delante o hacia atrás que ofrece el navegador y presentan problemas de accesibilidad y usabilidad.

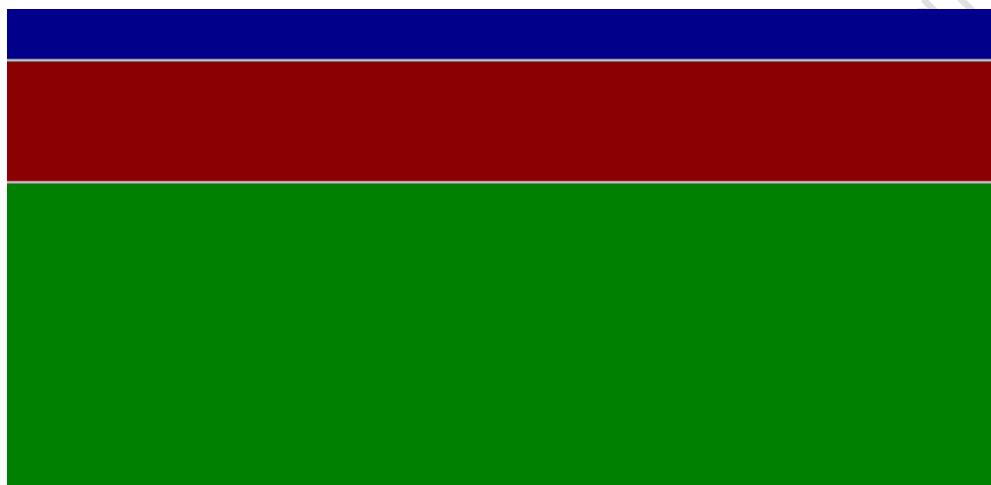
Además, de que el uso de marcos esté obsoleto, también se desaconseja su uso, aunque deba conocer su uso, puesto que, aún existen muchos sitios webs que utilizan estos elementos.

Para configurar los marcos, puede hacerlo dándoles un valor fijo o dinámico. Esto se establecerá si el tamaño está en píxeles o en porcentaje, respectivamente. Cuando se divide la página con marcos, debe establecer el tamaño de las columnas (cols), y las filas (rows). Además, debe conocer el significado del asterisco (*) cuando se establecen los tamaños de los marcos. El asterisco significa que el marco cogerá el resto del espacio que le dejen los demás marcos establecidos. El siguiente ejemplo establece un marco de 100 píxeles de altura, otro que tome el 25% de altura de la ventana y otro que tome el resto del espacio: <!DOCTYPE html>

```
<html>
  <head>
    </head>
    <frameset rows= 100,25%,*>
      <frame style="background-color: darkblue;">
```

```
<frame style="background-color: darkred;">  
  
<frame style="background-color: green;">  
  
</frameset>  
  
</html>
```

Lo que mostrará el navegador será lo siguiente, el marco con altura de 100 píxeles se verá en azul, el marco que toma el 25% de la altura de la pantalla se verá en rojo, y por último, el marco que toma la altura restante se verá en verde:



1.6.1.4. *Configuración de los marcos*

Al usar los marcos, puede dividir una página web tanto de manera vertical, como horizontal. Sin embargo, no puede dividirla de ambas maneras a la vez.

En cambio, existe la posibilidad de anidar marcos dentro de otros marcos. Esto posibilita cambiar la disposición de los mismos.

1.6.2. *Capas*

Las capas se pueden definir como un concepto de “contenedor” el cual sirve para albergar y componer otros elementos dentro de una “caja” mayor.

Definición de capas

Las capas se utilizan para funcionar como contenedores de otros elementos. Por ejemplo, las capas pueden contener imágenes, botones, textos, incluso otras capas. Es por ello, que puede definir una capa como un elemento que divide el documento en varias partes y que puede contener elementos que se comporten de manera diferente al resto de los elementos de la página. Si, además, utiliza

código CSS, puede modificar las capas a su antojo y de manera individual o colectiva. Además, podrá posicionarla donde quiera y como quiera. Vea a continuación un documento con capas:



Como puede observar, este documento tiene varias capas y en cada una de ellas contiene diferentes elementos como un vídeo, texto, títulos y enlaces.

Para utilizar capas en sus documentos HTML, tendrá que utilizar la siguiente sintaxis:

```
<!DOCTYPE html>

<html>

<head>

</head>

<div>Esto es una capa</div>

<span>Esto es otra capa, pero para agrupar contenidos pequeños</span>

</html>
```

Creación de las capas

Para la creación de capas en HTML se pueden usar dos etiquetas `<div>` y ``. La diferencia entre ambas es que `<div>` sirve para agrupar grandes cantidades de contenido como, por ejemplo, un título, dos párrafos y una imagen. Y `` se usa para agrupar contenidos pequeños como varias palabras de un párrafo.



TOME NOTA

Debe recordar que la versión HTML5 ofrece un conjunto de nuevas etiquetas que permite dividir el documento de manera más específica facilitando la identificación de las partes. Las etiquetas son: <nav>, <section>, <header>, <article>, <address>, <footer> y <aside>.

Utilización de las capas

En este apartado verá unos ejemplos de cómo usar <div> y y la razón de porqué utilizarlos. Además, verá cómo añadir un poco de código CSS a estas dos etiquetas.

Como ya se vió anteriormente, la etiqueta se utiliza para albergar poco contenido como, por ejemplo, una palabra o un conjunto de palabras de un párrafo para destacarlas. Para ello, vea un ejemplo de cómo hacerlo:

```
<!DOCTYPE html>

<html>

<head>

</head>

<p><span style="color:blueviolet; font-weight: bold;">Bienvenidos</span> a mi web</p>

</html>
```

Si utiliza este código, el navegador mostrará lo siguiente:

Bienvenidos a mi web

Ahora, se pasa a la etiqueta <div> que se usa para crear secciones independientes dentro de su página. Vea continuación un ejemplo donde crear una sección que va a contener dos párrafos y un título:

```
<!DOCTYPE html>

<html>

<head>

</head>
```

```
<body>

    <div style="background-color: cadetblue; width:400px; height: 400px; border-style: dotted; border-width: 1px; padding: 10px; text-align: justify;">

        <h2>Este es el título de la capa</h2>

        <p>

            Lorem ipsum dolor sit amet consectetur adipiscing elit et, rhoncus pretium class sem arcu habitasse suspendisse elementum fermentum, lacus tortor fames fusce dapibus hac quis. Eros platea velit vestibulum congue iaculis in praesent quisque imperdiet nisl, cum ante convallis laoreet facilisis est blandit viverra libero, at inceptos nam cursus tempus suspendisse sollicitudin eget curae. Augue neque molestie luctus taciti urna lacinia commodo ut fringilla convallis, magna pharetra porta leo hendrerit risus sed mattis purus.

            Orci consequat fermentum platea sociis cras porttitor magnis dui, mi tristique bibendum pretium sodales erat tortor, turpis ad fames in elementum rutrum integer. Maecenas donec cursus ultricies vel dignissim ante, suscipit interdum cras luctus neque massa, curae consequat convallis venenatis orci. Et nascetur suspendisse eget purus congue leo magna aliquet hac feugiat, himenaeos nec sagittis maecenas sollicitudin sodales sociis placerat tincidunt, posuere sed natoque volutpat montes molestie praesent elementum felis.

        </p>

    </div>

</body>

</html>
```

Lo que muestra el navegador es lo siguiente:

Este es el título de la capa

Lorem ipsum dolor sit amet consectetur adipiscing elit et, rhoncus pretium class sem arcu habitasse suspendisse elementum fermentum, lacinia tortor fames fusce dapibus hac quis. Eros platea velit vestibulum congue iaculis in praesent quisque imperdiet nisl, cum ante convallis laoreet facilisis est blandit viverra libero, at inceptos nam cursus tempus suspendisse sollicitudin eget curae. Augue neque molestie luctus taciti urna lacus commodo ut fringilla convallis, magna pharetra porta leo hendrerit risus sed mattis purus. Orci consequat fermentum platea sociis nascetur suspendisse sollicitudin sodales erat tortor, turpis ad fames in elementum rutrum integer. Maecenas donec cursus ultricies vel dignissim ante, suscipit interdum nascetur suspendisse sollicitudin sodales nisi placerat tincidunt, posuere sed natoque volutpat montes molestie praesent elementum felis.

Como puede haber experimentado, el uso de <div> y es muy útil y eficiente para delimitar las zonas del documento con distintos contenidos. Además, es algo que puede ver de manera habitual en todas las páginas web. Sin embargo, es imprescindible saber manejar un poco de código CSS para poder obtener todas las ventajas que ofrece el uso de estos elementos.

2. IMÁGENES Y ELEMENTOS MULTIMEDIA

En HTML puede insertar elementos multimedia para que sean visualizados en su página web. El uso de estos elementos aporta claridad al mensaje que quiera transmitir. Por ejemplo, imagine que quiera crear una página de fisioterapia. Aunque se explique en modo texto cómo realizar determinados ejercicios, siempre va a ser más explicativo un vídeo mostrando cómo hacer ese ejercicio. Pues esto, sería un elemento multimedia.

Por otro lado, los elementos multimedia también están pensados para captar la atención de los usuarios, esto a veces puede llegar a ser un verdadero reto para los desarrolladores web. Sin embargo, al usar los elementos multimedia es mucho más fácil captar la atención de los usuarios, ya que, las imágenes, audios y vídeos siempre son elementos más amenos para el usuario que montones de líneas de texto. Esto sucede porque la información que se da mediante elementos multimedia, es una información rápida y sencilla, es decir, el usuario no necesita hacer mucho esfuerzo para consumir esa información. Es por ello que las plataformas como Instagram o Facebook son tan populares, porque la información se consume a través de elementos multimedia.

En definitiva, en este tema aprenderá cómo insertar estos elementos multimedia en su web y cómo configurar el contenido multimedia como imágenes, vídeos, audios, etc. En sus documentos HTML.

Se comienza por las imágenes, ya que, son el elemento multimedia más básico y utilizado. Además, también se verá cómo modificar las imágenes manipulando sus atributos en HTML.

Después se verán los vídeos y los audios donde también se aprenderá a manipular sus atributos para adaptarlos mejor a su web. Y, por último, verá cómo utilizar una especie de pequeños programas codificados en el lenguaje de programación Java y se insertará en un documento HTML.

2.1. Inserción de imágenes: formatos y atributos

Para poder insertar una imagen en su web, antes hay que realizarla, ya sea, dibujando, fotografiando, escaneando, etc. Además, habrá que adecuarla para utilizarla en el espacio en que la precisa, es decir, no es lo mismo poner una imagen de fondo, que ponerla como ícono. Para preparar las imágenes debe utilizar aplicaciones externas como Photoshop o Gimp, aunque hay muchas más. Algunas de estas aplicaciones son más potentes que otras, por lo que, brindarán mejores resultados como, por ejemplo, a la hora de corregir la iluminación de una foto, los colores, eliminar fallos como, por ejemplo, los ojos rojos y aplicar filtros visuales.

Puede utilizar las imágenes de maneras muy diversas:

- **Imagen simple:** Se pueden utilizar las imágenes de manera que aporte algún tipo de información, por ejemplo, el logo de una empresa, los banners, fotos de productos, etc.
- **Imagen como vínculo:** Se pueden utilizar imágenes que, al pinchar en ellas, transporten a otro documento, este ejemplo se ve mucho en los blogs donde se pone una imagen llamativa sobre un post y al pinchar en ella, lleva al artículo.

- **Imagen como mapa de imagen:** Los mapas de imagen permiten tener varias zonas interactivas que vinculan a otros documentos. De este ejemplo, puede ver muchos en las tiendas online donde puede ver imágenes de productos que tienen una zona donde hay un botón integrado en la imagen y al pinchar ahí lleva al producto.

Como ya sabe, insertar elementos multimedia es una tarea muy sencilla y va a ayudar de manera significativa a crear páginas atractivas y profesionales.

Antes de insertar imágenes en su web, debe aprender qué formatos existen y para qué se utiliza cada uno. Los archivos de imagen se diferencian por su formato, los formatos más utilizados son JPG, GIF y PNG. Aunque existen otros muchos formatos de imagen.

Es importante que las imágenes que utilice tengan un tamaño adecuado y esto se debe a tres factores principales:

La navegación móvil. Como ya sabe el uso del móvil es cada vez mayor y la velocidad que proporcionan las redes 4G y 5G no puede asemejarse con la banda ancha que aporta la fibra óptica. Por lo tanto, los usuarios de dispositivos móviles agradecerán que su web no tarde en cargar mucho rato.

- **Pérdida de usuarios.** Se sabe que una web lenta es sinónimo de perder usuarios, lo que se traduce como un porcentaje de rebote muy alto. Para que esto no suceda, es importante que optimice su web y este proceso empieza por optimizar el peso de sus imágenes.
- **Posicionamiento SEO.** Los distintos buscadores disponen de un tiempo limitado para rastrear las webs, es decir, que cuanto menos pese su web, más páginas podrá rastrear y tendrá más posibilidades de posicionar mejor su web. Además, una de las cosas en las que Google hace mucho hincapié actualmente es la velocidad de carga. Por lo que, si mejora la velocidad de carga ganará puntos y su posicionamiento mejorará.

A continuación, va a ir desgranando cada uno de los formatos más conocidos y utilizados de imagen. Esto es importante para saber qué tipo de imagen debe utilizar para cada finalidad.

GIF

Este formato se utiliza, sobre todo, para dibujos y no en imágenes reales, ya que, este formato utiliza una paleta de 256 colores y es capaz de contemplar que uno de esos colores sea el color transparente. Este formato, también es utilizado para imágenes animadas. Antiguamente, era el formato más utilizado para las imágenes web, pero al existir un problema con el algoritmo de compresión que utilizaba, se dejó de usar debido a la incompatibilidad. Este tipo de imágenes se diferencian sobre todo en dos cosas fundamentales:

- **Animaciones.** Esta es la característica más fundamental de este formato de imagen. Con este tipo de imágenes puede crear, en un mismo archivo, una secuencia de imágenes que se muestra de manera cíclica creando así una animación.

- **Transparencias.** Además, este tipo de imágenes pueden contener fondos transparentes, por lo tanto, si superpone esta imagen con otra detrás, la de detrás también se mostrará.

JPEG o JPG

Este formato se utiliza para imágenes reales como, por ejemplo, fotografías. Esto se debe a que este formato brinda la posibilidad de elegir el nivel de compresión de la imagen para poder obtener una mejor relación entre peso y calidad de la imagen.

Este tipo de formato lo que aporta es una gran compresión del archivo, por lo tanto, es ideal para usarlo en las imágenes, ya que, así la web requerirá un menor tiempo de carga. En cambio, este formato se desaconseja si quiere obtener fotografías o imágenes con mucha calidad.

PNG

Este formato fue diseñado para sustituir el formato GIF debido al problema que se ha comentado anteriormente que tuvo el formato GIF. El formato PNG es un formato que sirve tanto para imágenes reales como para dibujos, ya que, ofrece la posibilidad de controlar la transparencia o la corrección de gamma y, además, ofrece un tipo de compresión de imágenes donde no se pierde calidad.

Este formato está aceptado por los navegadores más conocidos, pero puede encontrar aún navegadores donde no es compatible este tipo de formato. Las imágenes que utilizan este formato pueden tener un grado de mayor compresión si éstas utilizan un número reducido de colores. Este formato, también, permite crear imágenes que aparecen una tras otra progresivamente e imágenes donde existe el color transparente. Este formato de imagen se conoce por la característica de que puede aplicar transparencia a las imágenes. Además de esta característica tan fundamental, este formato presenta otras características que lo diferencia de los demás formatos:

- **Velocidad.** Este tipo de imágenes se pueden visualizar antes de haberse cargado por completo.
- **Compresión.** La compresión de este tipo de imágenes es un 30% más reducida que la de las imágenes GIF.
- **Fidelidad.** Si utiliza este formato de imagen, puede estar seguro de que los colores de la imagen se verán de la misma manera sin importar la plataforma en la que se esté visualizando.

A continuación un ejemplo de uso de imágenes PNG donde puede apreciar la transparencia de la imagen:



Como puede observar la imagen es la luna y los murciélagos. Sin embargo, el color de fondo de la página es el azul oscuro y ese fondo se muestra aun teniendo la imagen superpuesta.

Como ya ha visto anteriormente, las imágenes para una web deben estar en formato GIF, en PNG o en JPG y los archivos deberán tener sus respectivas extensiones. Este punto es importante porque de este modo es cómo el navegador es capaz de reconocer el tipo de archivo que está procesando.

- **Imagen .gif.** Hoy por hoy este tipo de imágenes están obsoletas. Solo se utilizan cuando se quiere añadir una imagen animada.
- **Imagen .png.** Como se vio anteriormente, estas imágenes sustituyen a las imágenes gif, puesto que también permiten utilizar el color transparente. Este formato es el mejor a utilizar para imágenes planas o con mucho espacio en blanco como, por ejemplo, logos, capturas de pantalla, productos para tienda online, etc. Es cierto que el formato png no comprime igual de bien como jpg, en cambio aporta imágenes con mayor calidad.
- **Imagen .jpg.** Este formato es muy útil para utilizarlo con fotografías con muchos colores y detalles, ya que, es un formato capaz de comprimir al máximo las imágenes, aunque la desventaja que tiene es que la imagen también perderá calidad y nitidez.

Uno de los mejores programas para optimizar imágenes es Photoshop. Si no tiene Photoshop, también puede utilizar diferentes herramientas online que permiten optimizar sus imágenes sin necesidad de instalar una aplicación.

2.1.1. Incluir imágenes en las páginas

Para añadir imágenes a su documento HTML, debe utilizar la etiqueta que es la que enlazará el archivo de imagen con el documento. Para ello, debe utilizar la siguiente sintaxis:

```

```

Como puede observar en esta sintaxis, el atributo *src* sirve para insertar la ruta donde se encuentra el archivo de la imagen. Si la imagen está en la misma carpeta que el documento HTML, tan solo, debe poner el nombre de la imagen y su formato. En cambio, si la imagen se encuentra en alguna otra carpeta, debe especificar su ruta de la siguiente manera *src="..../imágenes/nombrelImagen.jpg"*.

Por otra parte, el atributo *alt* se utiliza para insertar un texto alternativo que describa la imagen. La razón de este texto alternativo es que si el navegador web, por alguna razón, no puede cargar la imagen, se mostrará este texto en su lugar.

Para finalizar, el atributo *title* permite establecer un título descriptivo que se mostrará cuando deje el puntero del ratón posicionado encima de la imagen.

En caso de que necesite aplicar unas dimensiones a la imagen que quiera añadir a su documento, lo más recomendable es hacerlo con CSS. Sin embargo, puede también utilizar los atributos *width* y *height* que ofrece HTML. El atributo *width* se utiliza para modificar la anchura de la imagen. En cambio, el atributo *height* se utiliza para modificar la altura. Puede establecer estas dimensiones en píxeles o en porcentajes dependiendo del resultado que deseé obtener.

A continuación, se va a realizar un ejemplo de cómo insertar una imagen en un documento cuando esa imagen está almacenada en otra carpeta:

	Imagenes	09/08/2021 15:26	Carpeta de archivos
	main.html	09/08/2021 15:27	Chrome HTML Do... 1 KB

La imagen que inserte tendrá las siguientes características:

- Anchura: 50%
- Al posicionar el cursor sobre la imagen deberá aparecer el mensaje “Feliz Halloween”
- El documento deberá tener el color blue de fondo.
- En caso de que la imagen no cargue correctamente, deberá aparecer un texto alternativo diciendo “Texto de felicitación de Halloween”
- El título del documento HTML se establecerá como “Insertando imagen en HTML”
- Por último, el documento deberá ser compatible con los caracteres de Europa Occidental

Vea el código resultante:

```
<!DOCTYPE html>

<html>

    <head>

        <title>Insertando imagen en HTML</title>

        <meta charset="UTF-8"/>

    </head>

    <body bgcolor="blue">

    </body>

</html>
```

Vea lo que muestra el navegador:



2.1.2. Atributos de las imágenes

Como ya ha visto, la etiqueta `` se utiliza para configurar todo tipo de elementos gráficos como, por ejemplo, imágenes, logos, botones, enlaces, entre otros. Sin embargo, este elemento en sí es un

elemento vacío, ya que, para que este elemento se muestre necesita enlazar algo a él. Por lo tanto, si quiere enlazar un logo, por ejemplo, necesitará aplicar el siguiente código:

```
<!DOCTYPE html>

<html>

    <head>

        <title>Ejemplo</title>

        <meta charset="UTF-8">

    </head>

    <body>

    </body>

</html>
```

Por lo tanto, en el ejemplo de este código se encuentra el atributo `src` que lo utiliza, precisamente, para enlazar el archivo a mostrar. Y, por otra parte, se encuentra el atributo `alt`, que se utiliza para añadir una pequeña descripción de la imagen que quiera mostrar.

Además, al utilizar los atributos `height` y `width`, el navegador reservará un espacio específico que será justamente el necesario para almacenar la imagen que ha enlazado con `src`.

Debe tener en cuenta que a la hora de utilizar los atributos `height` y `width`, debe tener especial cuidado en conservar la relación de aspecto de la imagen, si lo se obtiene es una imagen desproporcionada y para nada profesional, ni estética. Esto ocurre porque si no le da las proporciones adecuadas, el navegador va a distorsionar o sesgar esa imagen que le ha dado.

A continuación, va a ver los atributos de la etiqueta ``, sus valores y cómo utilizarlos:

- **Align.** Este atributo se utiliza para alinear la imagen con el texto adyacente. Sus valores pueden ser: `right`, `left`, `top`, `middle` y `bottom`. Sin embargo, este atributo se encuentra obsoleto y se recomienda utilizar, en su lugar, la propiedad `float` de CSS.
- **Alt.** En este atributo se establece una pequeña descripción que describa la imagen a mostrar.
- **Border.** Este atributo se utiliza para establecer el borde de la imagen en píxeles. Si se establece el valor a cero, se evita que se muestren los bordes de hipervínculo en una imagen. Sin embargo, este atributo está obsoleto y, en su lugar, se recomienda utilizar la propiedad `border` de CSS.

- **Height.** Este atributo permite establecer la altura de la imagen en píxeles.
- **Hspace.** Este atributo permite establecer la cantidad de espacio, en píxeles, que queda a la izquierda y a la derecha de la imagen. Este atributo, actualmente, está obsoleto y, en su lugar, se utiliza la propiedad padding de CSS.
- **Id.** Este atributo se utiliza para establecer un nombre identificador. Para ello, se establecerán nombres alfanuméricos que comiencen con una letra y que no tengan espacios. Este valor identificador debe ser único y, por lo tanto, no se utiliza para otros identificadores en el mismo documento.
- **Longdesc.** Este atributo se utiliza para añadir la url de un recurso que contiene la descripción de una imagen.
- **Name.** Este atributo se utiliza para nombrar una imagen para que, posteriormente, se pueda acceder fácilmente a ella desde un lenguaje de script del lado del cliente como, por ejemplo, JavaScript. El nombre de texto deberá cumplir los siguientes requisitos: deberá contener caracteres alfanuméricos, comenzar por una letra y no tener espacios. Sin embargo, este atributo está obsoleto y, en su lugar, se utiliza el atributo id.
- **Src.** Este atributo se utiliza para indicar la url del archivo donde está contenida la imagen.
- **Srcset.** Este es un nuevo atributo que ofrece HTML 5.1. Se utiliza para que el navegador admita la visualización de imágenes con contenido sensible.
- **Title.** Se utiliza este atributo para añadir un pequeño texto que contenga una pequeña descripción de la imagen. Normalmente, esta descripción es más descriptiva que la que se añade en el atributo alt.
- **Vspace.** Este atributo se utiliza para establecer la cantidad de espacio, en píxeles que debe haber encima y debajo de la imagen. Sin embargo, actualmente, este atributo está obsoleto y, en su lugar, se utiliza la propiedad padding que ofrece CSS.
- **Width.** Este atributo se utilizará para indicar el ancho de la imagen en píxeles.

Al observar los atributos vistos hasta ahora, puede ver que algunos de estos atributos están, actualmente, obsoletos. Sin embargo, aunque estén obsoletos en HTML5, aún puede utilizarlos en XHTML. Por lo tanto, podrá encontrar páginas codificadas con estos atributos y, por ello, es importante que los aprenda a utilizar. Además de aprender a utilizar estos atributos obsoletos, también aprende algo de código CSS para poder implementarlo en sus webs y, de ese modo, crear webs actualizadas con los nuevos estándares.

Por otro lado, tiene que tener muy en cuenta el uso del atributo alt, ya que, este atributo proporciona accesibilidad en su sitio web, y esto es algo muy importante para el SEO. El atributo alt es el que utiliza para establecer una descripción como texto alternativo de la imagen. El navegador utilizará este texto de dos maneras:

- La primera será que el navegador mostrará este texto alternativo en el área de la imagen, mientras esta se descarga y se muestra.
- La segunda, es que algunos navegadores pueden mostrar este texto alternativo como información sobre herramientas cada vez que ponga el cursor del ratón sobre el área de la imagen. Lo importante es saber que los textos alternativos pueden ser leídos por lectores de

pantalla en voz alta. Por otro lado, un navegador móvil puede mostrar el texto alternativo en vez de la propia imagen.

Además, los navegadores tipo estándar como, por ejemplo, Safari, no son los únicos tipos de aplicaciones que pueden acceder a un sitio web. Existen los denominados robots de Internet que son utilizados por los navegadores de búsqueda para inspeccionar las páginas web. Lo que hacen estos pequeños programas es indexar y categorizar los diferentes sitios web. En cambio, no pueden interpretar el texto que está dentro de las imágenes. Sin embargo, algunos de estos robots sí que pueden procesar el texto que está establecido en el atributo *alt* de las imágenes.

Para finalizar, el W3C recomienda que el texto que escriba dentro del atributo *alt*, no supere los cien caracteres. Además, debe evitar utilizar el nombre del archivo o palabras como imagen y gráfico. Por lo tanto, debe utilizar una frase corta que describa la imagen en sí.

Como ha mencionado anteriormente, puede crear vínculos con una imagen. Para ello, debe insertar un pequeño código para que funcione esa imagen como un hipervínculo. Para poder crear un enlace con una imagen, debe rodear la etiqueta ** con etiquetas de anclaje. Por ejemplo, se coloca un enlace alrededor de una imagen llamada animación.gif.

Vea e ejemplo:

```
<!DOCTYPE html>

<html>

  <head>
    <title>Ejemplo</title>
    <meta charset="UTF-8">
  </head>

  <body>
    <a href="index.html"></a>
  </body>

</html>
```

Por otro lado, también tiene los enlaces de miniaturas que se realizan con una pequeña imagen configurada como un enlace de imagen. Para ello, se utiliza el atributo *href* el cual apuntará hacia otro archivo de imagen, en vez de apuntar a otra página web. Vea un ejemplo:

```
<!DOCTYPE html>

<html>

    <head>

        <title>Ejemplo</title>

        <meta charset="UTF-8">

    </head>

    <body>

        <a href="anoecer.jpg"></a>

    </body>

</html>
```

2.1.3. Propiedades de ubicación de las imágenes

Antiguamente, antes de la versión de HTML5, se disponía de montones de atributos para controlar el posicionamiento de las imágenes y la manera en la que estas se visualizarían en el navegador. Sin embargo, actualmente es imprescindible utilizar código CSS para poder realizar esta tarea.

Usar imágenes en las páginas web es un recurso fundamental, ya que, además de añadir información gráfica adicional que hace que el usuario pueda consumirla rápida y eficazmente, también mejora el aspecto gráfico de la página y de los elementos que la constituyen. Vea algunos ejemplos de lo que puede conseguir al usar imágenes en su web:

- Puede utilizar imágenes para crear degradados o texturas en el fondo de la web.
- Puede crear sombras y bordes en los elementos, de manera que parezca que estos tienen volumen.
- Además, también puede utilizar imágenes en proporción de ¼ y posicionarlas en las esquinas de los elementos para que parezca que estos elementos tienen esquinas redondeadas. Vea a continuación una imagen con las esquinas redondeadas con el uso de código:



El próximo ejemplo que se verá es tomando el ejemplo anterior del texto de felicitación de Halloween, va a hacer que la imagen se ubique en el centro del documento. Para ello, se utiliza código CSS:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Insertando imagen en HTML</title>

    <meta charset="UTF-8"/>

  </head>

  <body bgcolor="blue">

  </body>

</html>
```

El resultado que mostrará el navegador es el siguiente:



TOME NOTA

Existen múltiples propiedades de CSS que puede utilizar para ubicar como quiera las imágenes y elementos en su documento.

2.2. Mapas de imágenes

Los mapas de imágenes se utilizan cuando quiera establecer un enlace en alguna parte de la imagen. Por ejemplo, las típicas imágenes de tienda online donde dice “Pinche aquí” y lleva a la página para comprar el artículo. Ese “Pinche aquí” es un área de la imagen que está enlazada con la página a la que tiene que dirigir.

En este epígrafe va a aprender qué son estos elementos y cómo implementarlos en su documento web.

2.2.1. Definición de mapa

Un mapa de imagen es una imagen la cual está dividida por zonas en las que se le ha añadido un enlace a conveniencia. Cuando pulsa en una de las zonas, serán redirigidos hacia la url que se habrá especificado en el momento de su codificación, es decir, estas zonas actuarán como enlaces convencionales. La única diferencia es que se trata de una imagen.

Las zonas con las que se divide la imagen, pueden tener diferentes formas. Por ejemplo, pueden ser circulares, rectangulares o, incluso, pueden formar complejas formas poligonales.

Para crear mapas de imágenes con HTML, puede hacerlo de dos maneras. La primera es crearlas con mapas de servidor y, la segunda con mapas de cliente. Realmente, la primera opción casi nunca se utiliza, por lo tanto, aprenderá a realizar esta tarea con la segunda opción. Y, es en esta opción donde va a profundizar en este epígrafe.

2.2.2. Creación de un mapa con una imagen

Para crear los mapas de imágenes en HTML, se utilizan la etiqueta <map>. Dentro de esta etiqueta es donde se irá definiendo las distintas áreas de la imagen, donde se añaden los enlaces, estableciéndolas con la etiqueta <área>. Esta etiqueta puede contener los siguientes atributos:

- **Href.** Este atributo es el que se utiliza para indicar la url a la que se redirige al pulsar sobre el área especificada en la imagen.
- **Shape.** En este atributo se especifica la forma que tendrá el área la cual dividirá la imagen. Las formas que pueden adoptar son: rectangular (rect), circular (circle) y poligonal (poly).
- **Title.** En este atributo se establece el texto que quiera que aparezca cuando sitúe el cursor del ratón encima del área.
- **Coords.** En este atributo se establecen las coordenadas que delimitarán el área interactiva de la imagen. Las coordenadas van a depender del tipo de forma de área que haya establecido.

2.2.3. Establecer diferentes partes en la imagen

Cuando se establecen las diferentes áreas para un mapa de imagen, debe tener en cuenta que las coordenadas se establecerán tomando como punto de origen la esquina superior izquierda de la imagen en la que está estableciendo el área interactiva. Por ejemplo, si una imagen mide 250 píxeles de ancho y 450 píxeles de alto, la esquina superior izquierda corresponderá con la coordenada (0,0), mientras que la esquina inferior derecha corresponderá con la coordenada (250, 450).

A continuación, va a ver los diferentes tipos de áreas que puede establecer en su mapa de imagen:

Rectangular

Para crear un área rectangular debe elegir el valor (rect) del atributo *shape*. Para definir este tipo de área se indican las coordenadas de los puntos de la esquina superior izquierda y de la esquina inferior derecha del rectángulo que quiera crear.

La sintaxis para establecer este tipo de área en su documento HTML es la siguiente:

```
<!DOCTYPE html>  
  
<html>  
  <head>  
    <title>Ejemplo</title>  
  
    <meta charset="UTF-8">  
  
  </head>  
  
<body>
```

```
<area shape="rect" coords="x1,y1,x2,y2" href="">  
</body>  
</html>
```

Vea a continuación un ejemplo:



Circular

Para crear un área circular debe tomar el valor (circle) del atributo *shape*. Además, debe escoger como coordenada de origen el centro del círculo y añadir su radio.

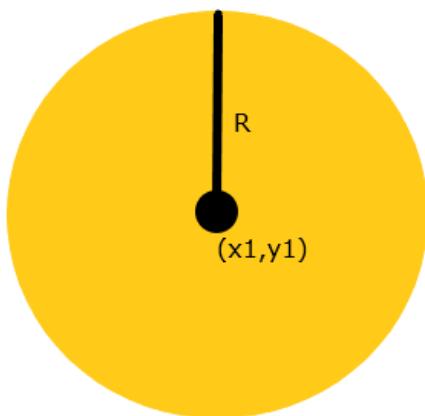
La sintaxis para establecer un área circular en un mapa de imagen es la siguiente:

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title>Ejemplo</title>  
    <meta charset="UTF-8">  
  </head>  
  <body>  
    <area shape="circle" coords="x1,y1,r" href="">
```

```
</body>
```

```
</html>
```

Vea a continuación una imagen de ejemplo de cómo establecer las coordenadas para este tipo de área:



Poligonal

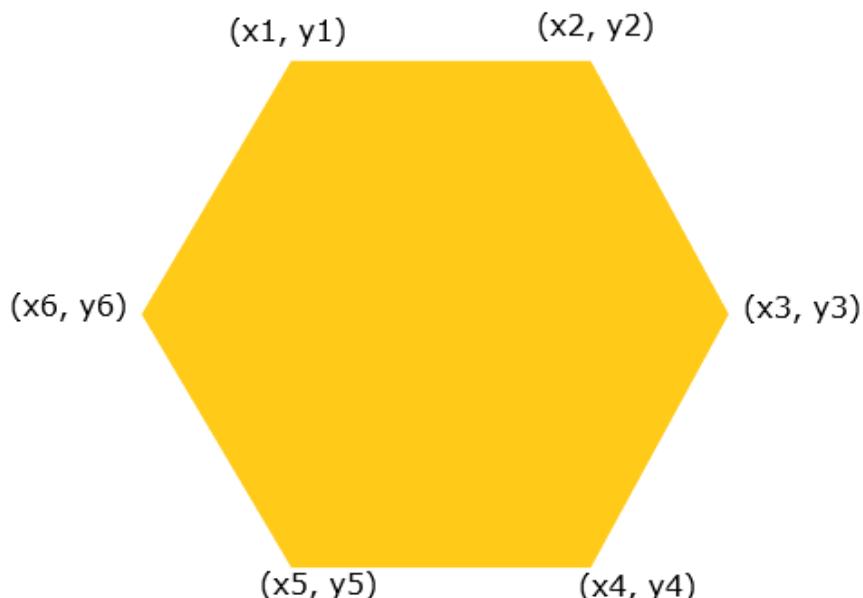
Para establecer un área poligonal, tendrá que elegir el valor (poly) para el atributo *shape*. Cuando quiera establecer un área de este tipo tendrá que establecer todos los puntos de coordenada de manera ordenada, de esta forma se establece el perímetro a seguir del polígono para delimitar la zona. Esto se debe a que, al establecer este tipo de formas, puede establecer cualquier forma. Por ejemplo, forma de flecha, de estrella, de diamante, etc.

La sintaxis para establecer formas poligonales es la siguiente:

```
<!DOCTYPE html>  
  
<html>  
  <head>  
    <title>Ejemplo</title>  
    <meta charset="UTF-8">  
  </head>  
  <body>
```

```
<área shape="poly" coords="x1,y1,x2,y2,x3,y3,x4,y4" href="">  
</body>  
</html>
```

Vea a continuación un ejemplo de cómo establecer las coordenadas:



2.2.4. Vincular las diferentes partes de la imagen

Previamente a crear el mapa de imagen, debe establecer la imagen fuera de la etiqueta `<map>`. Se establecerá la imagen con su correspondiente etiqueta `` como ya se vio anteriormente. Posteriormente a establecer la imagen, debe establecer las áreas interactivas de la imagen en la etiqueta `<map>`. Además, debe añadir en la etiqueta `` el atributo `usemap` el cual tendrá como valor el nombre del atributo `name`, precedido con una almohadilla, que haya establecido para la etiqueta `<map>`. Vea a continuación un ejemplo:

```
<!DOCTYPE html>
```

```
<html>
```

```
 <head>
```

```
 </head>
```

```
<body bgcolor="blue">

    <map name="mapaDelImagen">

        <area/>

    </map>

</body>

</html>
```

Debe recordar que para establecer una imagen se utiliza la etiqueta ``. Además, para especificar la ruta con la que se enlazará la imagen, se utiliza el atributo `src`. Por otro lado, el atributo `title` funcionará como texto explicativo de la imagen cuando se pare el cursor encima de la misma. Y, por último, el atributo `alt` servirá para que el navegador muestre ese texto alternativo en caso de que la imagen, por alguna circunstancia, no pueda ser visualizada.

Lo siguiente que va a ver es un ejemplo de todo lo que ha aprendido anteriormente. Va a crear un documento HTMML que va a contener un mapa de imagen para la siguiente imagen:



Va a definir en esta imagen cinco áreas, una para cada ícono, las cuales van a redirigir cada una al respectivo navegador al que pertenezca cada ícono que pulse. Además, todas las áreas para cada ícono serán circulares, menos la de Edge que será rectangular.

Vea el resultado en código:

```
<!DOCTYPE html>
```

```
<html>

<head>

<title>Mapa de imagen</title>

<meta charset="UTF-8">

</head>

<body>



<map name="mapaDelImagen">

<area target="_blank" alt="navegador Opera" title="navegador Opera" href="https://www.opera.com/es" coords="230,92,73" shape="circle">

<area target="_blank" alt="navegador Chrome" title="navegador Chrome" href="https://www.google.com/intl/es_es" coords="442,91,79" shape="circle">

<area target="_blank" alt="navegador Edge" title="navegador Edge" href="https://windows.microsoft.com/es-es" coords="176,340,32,192" shape="rect">

<area target="_blank" alt="navegador Mozilla Firefox" title="navegador Mozilla Firefox" href="https://www.mozilla.org/es-ES/firefox" coords="321,265,85" shape="circle">

<area target="_blank" alt="navegador Safari" title="navegador Safari" href="https://support.apple.com/kb/DL1531?" coords="540,273,95" shape="circle">

</map>

</body>

</html>
```

Las áreas marcadas serán las siguientes:



Vea a continuación un caso práctico donde se utiliza lo aprendido hasta ahora sobre los mapas de imágenes. En este caso práctico va a insertar una F en la que dependiendo en que segmento de la letra pulse será enviado a un sitio diferente. Vea el código a continuación:

```
<!DOCTYPE html>

<html>

<head>

    <meta charset="UTF-8">

    <title>Caso práctico mapa de imagenes</title>

</head>

<body>

            <area shape="rect" coords="105,95,125,207" href="https://www.google.es"
            alt="tutorial html" target="_blank">

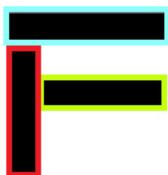
            <area shape="rect" coords="126,95,215,117" href="https://www.youtube.com alt=""'
            target="_blank">

            <area shape="rect" coords="128,140,215,160" href="https://www.gmail.com" alt=""'
            target="_blank">

        </map>
    </img>
</body>
```

```
</map>  
  
</body>  
  
</html>
```

Vea, a continuación, lo que muestra el navegador, como puede ver cada segmento de esta letra llevará a las url establecidas en el código HTML. Al igual que puede hacer esto con esta letra, podría hacerlo con cualquier elemento. Debe recordar que, tan solo, hace falta establecer las áreas donde tendrá efecto el enlace:



2.3. Inserción de elementos multimedia: audio, vídeo y programas

Los elementos multimedia no solo son las imágenes, sino que también están compuestos por elementos de vídeo, audio y programas. En este epígrafe se verá el uso de estos elementos para aprender a manipularlos de manera que pueda insertarlos en su web. Y de ese modo hacerla más atractiva para los usuarios. Se realizará esta tarea utilizando HTML.

2.3.1. Características y propiedades de los elementos multimedia

A continuación, aprenderá las características que comparten este tipo de elementos entre sí. Esto es importante que lo sepa antes, incluso, de aprender cómo insertar un elemento de este tipo.

Estos elementos permiten que el usuario consuma la información de manera rápida, ya que, para el usuario esto supone un estímulo auditivo y visual.

Además, estos elementos ofrecen una mejora para las presentaciones, puesto que, se basan en texto e imágenes e incluir audio y vídeo hará que los oyentes se sientan más interesados en su contenido. Por ello, estos elementos también son conocidos por captar fácilmente la atención del usuario.

Todos estos elementos harán que el usuario retenga más y de mejor manera la información que le proporciona. Además, el conjunto de todos estos elementos aporta un contenido entretenido y ameno para el público.



TOME NOTA

Para poder visualizar este tipo de elementos, a veces, necesita utilizar un plug-in que permita visualizarlos en el navegador.

Sin embargo, los navegadores que son compatibles con HTML5 no suelen necesitar la instalación de *plug-ins* para visualizar los contenidos multimedia.

2.3.2. Recursos necesarios para el funcionamiento de los elementos multimedia

Para que la mayoría de los recursos multimedia puedan ser reproducidos o visualizados en las páginas web, estos deben cumplir con una serie de requisitos que se verá a continuación:

Antes de poder reproducir archivos multimedia que estén ubicados en una web, debe tener un dispositivo con un navegador instalado que sea capaz de acceder a Internet. Esto puede sonar a algo obvio, pero, por ejemplo, con los dispositivos móviles antiguos no podía acceder a archivos multimedia ubicados en la web. Sin embargo, con los dispositivos móviles actuales sí es viable.

Por otra parte, el navegador que utilice para acceder a ese recurso multimedia, debe ser compatible con el formato de compresión del archivo que intenta reproducir. En caso de que el navegador no sea compatible con el formato de compresión del archivo a reproducir, tiene dos posibilidades. La primera es instalar un *plug-in* en el navegador para solucionar la incompatibilidad de este con el archivo. La segunda es utilizar otro navegador distinto que sea compatible de manera nativa.

Además, por la parte que le toca al desarrollar debería tener siempre a disposición herramientas específicas de edición y conversión de archivos multimedia como imágenes, audio y vídeo. Estas herramientas ayudarían mucho, por ejemplo, a cortar partes de una imagen cuando la imagen contiene textos o alguna información que no quiera añadir en la web. También, estas herramientas permiten cambiar los colores, dimensiones y diversas características de la imagen para adecuarla al contenido de su web. Por otra parte, si, por algún motivo, necesita cambiar los formatos de los archivos de imágenes también puede echar mano de este tipo de herramientas.

En el caso de los archivos de audio y vídeo, puede extraer los fragmentos que interese en caso de que quiera añadir todo el vídeo a su web. Además, podrá convertir los formatos de vídeo y audio según convenga. Por último, puede modificar las dimensiones, el volumen o los códecs de los archivos de audio y vídeo.

Actualmente, existe una gran diversidad de aplicaciones que permiten realizar estas tareas de editar archivos multimedia. Aunque, algunas de estas aplicaciones pueden ser de pago, como es el caso de Photoshop. Sin embargo, puede encontrar otras muchas aplicaciones que permitirán realizar las mismas tareas y que, además, son gratuitas como pueden ser Gimp para edición de imágenes, Picasa

también para edición de imágenes, Ocen Audio para edición de audio, Audacity para edición de audio, Lightworks para edición de vídeo, VirtualDub para el mismo cometido que la anterior, entre muchas otras.

2.3.3. Etiquetas y propiedades para la inserción de audio

El nuevo estándar HTML5 ofrece nuevas etiquetas para manipular contenidos de audio y vídeo. En el caso de los contenidos de audio, tiene la etiqueta `<audio>` la cual va a permitir insertar archivos de audio en su documento HTML. Vea a continuación su sintaxis:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Ejemplo</title>

    <meta charset="UTF-8">

  </head>

  <body>

    <audio src=""></audio>

  </body>

</html>
```

Como puede observar la etiqueta `<audio>` contiene el atributo `src` que será con el cual enlaza el archivo de audio a su documento HTML. Vea un ejemplo donde va a incluir el archivo de audio `miAudio.mp3` dentro de su documento HTML. Debe tener en cuenta que, en este ejemplo, el archivo de audio se encuentra en la misma carpeta que su archivo HTML. Por lo tanto, su ejemplo quedaría de la siguiente manera:

```
<audio src="miAudio.mp3"></audio>
```

A continuación, va a ver los atributos que puede utilizar en la etiqueta `<audio>`:

- **Src.** Este atributo es donde establece la url que enlazará con el archivo a reproducir.
- **Autoplay.** Este atributo se utiliza para que el audio que está incluyendo en su web, se reproduzca de manera inmediata sin necesidad de que el usuario pulse nada.
- **Controls.** Este atributo establece una interfaz de controles para poder controlar la reproducción del audio, tales como el sonido, parar o continuar el audio, etc. Vea un ejemplo de controles de audio:

▶ 0:00 / 0:00



TOME NOTA

Cuando se usa el atributo controls, debe tener en cuenta que en cada navegador se va a mostrar la barra de controles de una manera diferente

- **Loop.** Se utiliza este atributo cuando quiere que el audio se vuelva a reproducir una vez haya terminado.
- **Preload.** Al utilizar este atributo puede darle tres valores:
 - **Auto.** Este valor le indicará al navegador que debe descargar el archivo de audio lo más inmediatamente que pueda.
 - **Metadata.** Este atributo hace que el navegador pueda obtener información sobre el audio. Por ejemplo, cuánto dura la pista.
 - **None.** Este valor le indica al navegador que el archivo de audio no debe ser cachead. Esto se utiliza cuando quiera minimizar el tráfico de datos.



TOME NOTA

Debe tener en cuenta que no solo existe una sola manera de insertar audio en un documento HTML. Antes de existir la etiqueta <audio>, se insertaba el audio con las etiquetas <embed> y <object>.

Todos los navegadores no son capaces de reproducir todos los tipos de formatos de audio y vídeo que existen, por lo tanto, si inserta un audio con el siguiente código `<audio src="miAudio.mp3"></audio>`, no todos los navegadores lo soportarían, por lo que, no en todos podría reproducir ese audio.

Para darle solución a este problema, puede echar mano de una etiqueta que se añade dentro de las etiquetas `<audio>` y `</audio>`. En esta etiqueta es donde puede definir los archivos de audio que va a utilizar con diferentes formatos. La etiqueta de la que se está hablando es `<source>` y lo que hará es añadir una etiqueta de este tipo por archivo insertado. Al establecer la etiqueta `<source>` el navegador primero comprobará el audio que tenga definido en la primera etiqueta `<source>`, si el navegador no puede reproducir ese audio debido a su incompatibilidad, es cuando va a comprobar los archivos que están establecidos en las siguientes etiquetas `<source>` hasta que encuentre uno

con el que sí tenga compatibilidad y, entonces, reproducirá el archivo. A continuación, va a ver la sintaxis que debe utilizar para establecer estas etiquetas:

```
<!DOCTYPE html>

<html>

<head>

    <title>Ejemplo</title>

    <meta charset="UTF-8">

</head>

<body>

    <audio>

        <source/>

        <source/>

    </audio>

</body>

</html>
```



TOME NOTA

En caso de que utilice la etiqueta <source> el atributo src de las etiquetas <audio> y <video> será reemplazado por la misma. De esta manera el navegador puede elegir qué tipo de formato reproducir.

A continuación, va a ver los atributos que puede utilizar con la etiqueta <source>:

- **Src.** Este atributo, como pasaba anteriormente con las etiquetas <audio> y <video>, se utiliza para establecer la url donde se encuentra el archivo a reproducir.
- **Type.** Se utiliza este atributo para establecer el tipo MIME del archivo de audio, su valor va a depender del tipo de archivo. Vea la siguiente tabla donde puede ver algunos tipos MIME más utilizados al definir los archivos de audio en la etiqueta <source>:

Formato	Tipo MIME
MP3	AUDIO/MPEG
OGG	AUDIO/OGG
WEBM	AUDIO/WEBM

- **MIME** es una norma de Internet que funciona de manera que hace que los navegadores sean capaces de identificar los tipos de archivo antes de ejecutarlos.

En el momento en que un navegador recibe un archivo, además, recibe metadatos que son la información sobre el propio archivo. Dentro de esa información también recibe el tipo MIME. De esta manera se facilita el proceso para tratar los archivos.

Si, por ejemplo, un navegador no reconoce el tipo MIME, ofrecerá la opción de escoger de manera manual un programa que sea capaz de interpretar ese archivo. Un pequeño ejemplo de tipo MIME son los siguientes: audio/mpeg, text/css, image/gif, etc.

Vea a continuación un ejemplo donde puede ver código HTML utilizando la etiqueta <audio> y <source> para establecer dos tipos de formatos de archivo de audio:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Ejemplo</title>

    <meta charset="UTF-8">

  </head>

  <body>

    <audio>

      <source src="miAudio.mp3" type="audio/mpeg"/>

      <source src="miAudio.webm" type="audio/webm"/>

    </audio>

  </body>

</html>
```

Ahora, va a ver un ejemplo real de cómo utilizar todos los elementos que ha aprendido en este epígrafe. Para ello, va a crear un documento que contendrá una etiqueta <audio>. Esta etiqueta va a incluir un archivo en dos formatos diferentes, mp3 y ogg. Además, se deberán mostrar en el navegador la barra de controles de reproducción. El archivo debe reproducirse de nuevo una vez haya terminado. Para finalizar, almacena los archivos de audio en una carpeta llamada audios.

En lo relativo al documento, deberá ser compatible con los caracteres de Europa Occidental y, además, tendrá como título “Etiqueta audio”. Vea a continuación el ejemplo de código:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Etiqueta audio</title>

    <meta charset="UTF-8">

  </head>

  <body>

    <audio controls loop>

      <source src="Audios/miAudio.mp3" type="audio/mpeg"/>

      <source src="Audios/miAudio.ogg" type="audio/ogg"/>

    </audio>

  </body>

</html>
```

A continuación, verá cómo se mostrará en el navegador:



2.3.4. Etiquetas y propiedades para inserción de vídeo

En el nuevo estándar HTML5 también puede encontrar una etiqueta específica para insertar archivos de vídeo en páginas web. La etiqueta es <video> y se verá su sintaxis a continuación:

<video src=""></video>

Puede utilizar el siguiente conjunto de atributos para esta etiqueta:

- **Poster.** Se utiliza este atributo para que el navegador muestre una imagen mientras se carga el vídeo. Para ello, se estable la url de la imagen a mostrar.
- **Controls.** Al igual que pasaba con la etiqueta <audio>, este atributo se utiliza, en este caso, para mostrar la barra de controles de vídeo.
- **Loops.** Este atributo también se encontraba en la etiqueta <video>. Se utiliza para volver a reproducir el vídeo una vez que este haya terminado.
- **Autoplay.** Este atributo se establece cuando quiera que el navegador inicie el vídeo en cuanto sea posible, sin necesidad de que el usuario lo inicie.
- **Preload.** Este atributo puede contener tres valores diferentes:
- **Auto.** Este es el valor que toma este atributo por defecto. Su utilidad es indicar al navegador que descargue el vídeo tan rápido como sea posible.
- **Metadata.** La utilidad de este valor es indicar al navegador que tome toda la información relacionada con el archivo que le sea posible.
- **None.** Por último, este valor indica que el vídeo no debe ser cacheado. Esto permite minimizar el tráfico de datos.



TOME NOTA

Se pueden utilizar también los atributos width y height con la etiqueta <video> para establecer el ancho y el alto del vídeo como si de una imagen se tratara.

Además, debe destacar que la etiqueta <video>, al igual que la de <audio>, también soporta el uso de las etiquetas <source>. Esto se debe a que, de la misma manera que ocurría con el audio, los navegadores no son capaces de soportar todos los formatos de vídeo existentes. Por lo tanto, la mejor manera de solucionar este asunto es utilizar la etiqueta <source> la cual tiene la siguiente sintaxis:

<!DOCTYPE html>

<html>

<head>

<title>Ejemplo</title>

```
<meta charset="UTF-8">  
</head>  
  
<body>  
  
    <video>  
  
        <source/>  
  
        <source/>  
  
    </video>  
  
</body>  
  
</html>
```

Por otra parte, puede utilizar en la etiqueta `<source>` los siguientes atributos:

- **Src.** Este atributo se utilizará para indicar la url en la que está ubicado el archivo a reproducir.
- **Type.** Se utiliza este atributo para establecer el tipo MIME del archivo. A continuación, vea una tabla en la que pueda observar los tipos MIME más utilizados para los archivos de vídeo:

Formato	Tipo MIME
MP4	Video/mp4
OGG	Video/ogg
WebM	Video/webm

Lo que va a ver a continuación es un ejemplo de uso de todos estos elementos. Se insertará un archivo de vídeo utilizando su etiqueta respectiva. Se insertará el vídeo en dos formatos diferentes mp4 y avi. Se deberá establecer la barra de controles de vídeo. Además, se indica que el vídeo se reproduzca de nuevo una vez haya terminado.

Por otra parte, el vídeo comenzará a reproducirse de manera automática cuando el navegador lo haya cargado. Mientras el vídeo carga, establece una imagen que se visualizará en su lugar. El nombre de la imagen será portada.jpg y se localizará en la misma carpeta donde se localiza el archivo HTML.

Además, los archivos de vídeo se localizarán en una carpeta denominada Vídeos, que estará, a su vez, en la misma carpeta del archivo HTML.

Para finalizar, el documento será compatible con los caracteres de Europa Occidental y el título será “Etiqueta video”. Vea el código resultante a continuación:

```
<!DOCTYPE html>

<html>

    <head>

        <title>Etiqueta video</title>

        <meta charset="UTF-8">

    </head>

    <body>

        <video controls loop autoplay poster="portada.jpg">

            <source src="Videos/miVideo.mp4" type="audio/mp4"/>

            <source src="Videos/miVideo.avi" type="audio/avi"/>

        </video>

    </body>

</html>
```

Vea a continuación cómo se muestra en el navegador:



Debe tener en cuenta que además de poder enlazar contenido local en las etiquetas <video> y <audio>, también puede hacerlo con archivos que se encuentren alojados en servidores de Internet. Para realizar esta tarea, tan solo, debe indicar en las etiquetas <source> la url donde se encuentran estos archivos. Vea un ejemplo:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Ejemplo</title>

    <meta charset="UTF-8">

  </head>

  <body>

    <video controls>

      <source src="https://urldelapagina.com/miVideo.mp4">

      <source src="https://urldelapagina.com/miVideo.avi">

    </video>

  </body>

</html>
```

Otra cosa que también debe tener en cuenta es que en las etiquetas <audio> y <video> puede incluir además de las etiquetas <source>, contenido adicional que será visualizado en caso de que el navegador sea incapaz de reproducir el contenido multimedia que le está indicando. Vea un ejemplo donde aparecería un texto indicando qué es el contenido que no ha podido ser visualizado en un navegador que no soporta HTML5:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Ejemplo</title>

    <meta charset="UTF-8">
```

```
</head>

<body>

<audio autoplay>

    <audio src="miAudio.ogg" type="audio/ogg"/>

    <audio src="miAudio.mp3" type="audio/mp3"/>

    Esto es un archivo de audio en HTML5

</audio>

</body>

</html>
```

Etiquetas <embed> y <object> para insertar elementos multimedia

Como ya se mencionó anteriormente, además de existir las etiquetas <audio> y <video> para insertar elementos multimedia en una web, también existen las etiquetas <embed> y <object> que permiten realizar esta tarea.

Etiqueta <embed>

Esta etiqueta se utiliza para insertar tanto archivos de audio, como de vídeo en un documento HTML. Los elementos que inserte con esta etiqueta serán reproducidos si el navegador tiene los plug-ins correspondiente para ello. La sintaxis que se utiliza para usar la etiqueta <embed> será la siguiente:

```
<embed src=""></embed>
```

Además, esta etiqueta tiene varios atributos dependiendo del tipo de archivo a mostrar, es decir, si el archivo es de audio o de vídeo. Vea a continuación algunos de los atributos de esta etiqueta:

- **Src.** Se utiliza este atributo para indicar la ruta donde se encuentra el archivo a reproducir.
- **Name.** Este atributo establecerá un nombre al objeto que está introduciendo.
- **Height y width.** Estos dos atributos se utilizarán para establecer la altura y la anchura de la ventana donde se mostrará el elemento, si el archivo a reproducir es un vídeo. Los valores de estos atributos los establece en píxeles.
- **Type.** Como en las etiquetas anteriores. Este atributo es el encargado de establecer el tipo MIME correspondiente. En caso de que no especifique el tipo MIME, el navegador tomará la extensión del archivo indicada en el atributo src.

Debe tener en cuenta que esta etiqueta tiene una particularidad y es que, puede utilizar atributos específicos de los diferentes *plug-ins*, que son los encargados de recibir esa información, validarla y utilizarla para modificar los parámetros de visualización correspondientes del elemento multimedia.

A continuación, verá un ejemplo práctico con todos estos elementos. Para ello, se crea un archivo HTML que contenga un archivo de vídeo. Debe utilizar la etiqueta `<embed>`, el archivo estará ubicado en la carpeta Videos y establece un tamaño de altura de 400 píxeles y de anchura de 600 píxeles. Por último, el documento deberá ser compatibles con los caracteres de Europa Occidental y tendrá el título de “Etiqueta Embed”. Vea el código:

```
<!DOCTYPE html>

<html>

    <head>

        <title>Etiqueta embed</title>

        <meta charset="UTF-8">

    </head>

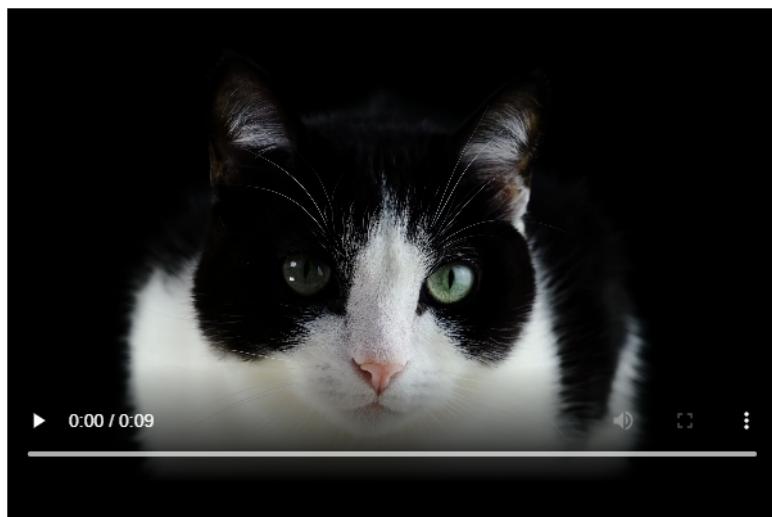
    <body>

        <embed src="Videos/miVideo.mp4" height="400" width="600" type="video/mp4"></embed>

    </body>

</html>
```

Ahora, verá lo que muestra el navegador:



Etiqueta <object>

Originalmente, esta etiqueta se creó, como la manera más recomendada, para insertar elementos multimedia en una página web como, por ejemplo, imágenes, audios, videos, programas, etc. A continuación, aprenderá a utilizar la sintaxis de esta etiqueta:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Ejemplo</title>

    <meta charset="UTF-8">

  </head>

  <body>

    <object>

      <param/>

      <param/>

    </object>

  </body>

</html>
```

Como puede observar, en la sintaxis de la etiqueta <object> aparecen las etiquetas <param>. Estas etiquetas se utilizan para añadir ciertos parámetros de visualización del elemento multimedia.

Además, debe saber que esta etiqueta permite el anidamiento de otras etiquetas para poder definir elementos alternativos en caso de que el navegador sea incapaz de reproducir el elemento multimedia.

A continuación, va a ver un ejemplo en el que puede observar un documento HTML donde se utiliza la etiqueta <object>, dentro de la cual se ha añadido otro elemento alternativo, en este caso una imagen, que se visualizará cuando el navegador no pueda reproducir el contenido original:

```
<!DOCTYPE html>

<html>
```

```
<head>

    <title>Ejemplo</title>

    <meta charset="UTF-8">

</head>

<body>

    <object type="application/x-shockwave-flash" data="mymovie.swf" width="600" height="400">

        <object data="imagen.jpg" width="600" height="400">

            </object>

        </object>

    </body>

</html>
```

Vea los atributos más fundamentales de la etiqueta `<object>` a continuación:

- **Data.** Este atributo es el encargado de indicar la ruta donde se encuentra el elemento multimedia que quiera insertar. Debe tener en cuenta que hay algunos *plug-ins* que necesitan que la información contenida en este atributo se especifique en la etiqueta `<param>`.
- **Form.** Este atributo permite asociar un elemento a un formulario.
- **Name.** Este atributo se utiliza para establecer un nombre para el elemento.
- **Height y width.** Estos dos atributos se utilizan para establecer la altura y anchura de la interfaz donde se va a visualizar el elemento. Puede establecer estas dimensiones tanto en píxeles como en porcentajes.
- **Usemap.** Se utiliza este atributo cuando quiera insertar una imagen que tome el comportamiento de un mapa interactivo, es por ello que este atributo contiene el identificador de un elemento `<map>`.
- **Type.** Este atributo es el que establece el tipo MIME del elemento.

Vea a continuación un ejemplo de cómo insertar mapa de imágenes utilizando la etiqueta `<object>`:

```
<!DOCTYPE html>

<html>

    <head>
```

```
<title>Ejemplo</title>

<meta charset="UTF-8">

</head>

<body>

<object data="mapa.jpg" usemap="#mapa" type="image/jpg" width="500" height="500">

<map name="mapa">

<a href="cocina.html" coords="20,30,40,60">Cocina</a>

<a href="dormitorio.html" coords="80,80,120,120">Dormitorio</a>

</map>

</object>

</body>

</html>
```

Implementar mapas de imágenes con la etiqueta `<object>` ofrece la ventaja de que, si el navegador no es capaz de cargar la imagen que le indica, en su lugar, podrá visualizar el texto de los enlaces que se corresponden a las áreas interactivas establecidas en el mapa de imagen. Por ejemplo, si toma el ejemplo anterior, podría observar los enlaces de Cocina y Dormitorio.

Vea todo lo aprendido con un ejemplo práctico. Creará un documento HTML donde va a insertar un vídeo utilizando la etiqueta `<object>`. Para el tamaño, va a utilizar de alto 400px y de ancho 600px. Debe establecer el tipo MIME correspondiente con el formato mp4.

Dentro de la etiqueta `<object>`, va a establecer una etiqueta `<param>` que se utiliza para indicar al *plug-in* correspondiente que el vídeo se va a reproducir en loop. Vea el código de ejemplo.

```
<!DOCTYPE html>

<html>

<head>

<title>Etiqueta object</title>

<meta charset="UTF-8">
```

```
</head>

<body>

<object data="Videos/miVideo.mp4" height="400" width="600" type="video/mp4">

    <param name="loop" value="true"/>

</object>

</body>

</html>
```

Vea lo que muestra el navegador:



2.3.5. Etiquetas y propiedades para la inserción de programas

En este epígrafe va a aprender a utilizar la etiqueta `<applet>` que permite añadir programas Java dentro de una página web. Comienza aprendiendo su sintaxis:

```
<!DOCTYPE html>
```

```
<html>
```

```
    <head>
```

```
<title>Ejemplo</title>

<meta charset="UTF-8">

</head>

<body>

<applet></applet>

</body>

</html>
```

En la nueva versión de HTML5 no está incluida esta etiqueta, en cambio, se ha sustituido por la etiqueta <object>. La razón por la cual ha sucedido esto es porque la etiqueta <object> es una etiqueta genérica para añadir elementos multimedia a las páginas web. Sin embargo, la etiqueta <applet> aún es reconocida por la gran mayoría de los navegadores.



Si al utilizar la etiqueta <applet> añade un texto entre la etiqueta de apertura y cierre, este texto se mostrará cuando el navegador sea incapaz de reproducir el contenido..

TOME NOTA

A continuación, vea los atributos más utilizados que puede usar con la etiqueta <applet>:

- **Name.** Este atributo se utiliza para establecer el nombre del applet, el cual podrá ser usado por otros applets para comunicarse entre los mismos.
- **Code.** Se utiliza este atributo para albergar el nombre de la clase Java donde está contenido el applet. Otra opción es establecer la ruta completa añadiendo el nombre de la clase Java. Se interpretará este valor conforme a lo que haya especificado en el atributo *codebase*.
- **Codebase.** En este atributo es donde se indica la ruta donde se localiza el *applet* y los recursos pertinentes para su ejecución. Si no indica ningún valor en este atributo, por defecto se tomará la ruta del archivo presente.
- **Mayscript.** Se utiliza este atributo para permitir que el *applet* acceda a las funcionalidades del código JavaScript que haya en el propio documento.
- **Align.** Este atributo puede tomar cinco valores: right, left, middle, top y bottom. Y se utiliza para alinear el *applet*.
- **Hspace y vspace.** Estos dos atributos se utilizan para indicar el espacio horizontal y vertical que debe haber entre el *applet* y el texto que haya a su lado. Los valores de estos atributos se expresarán en píxeles.

- **Width y height.** Se utiliza estos dos atributos para establecer la anchura y la altura del área donde se muestra el *applet*.
- **Alt.** Este atributo se utiliza para introducir un texto, por si acaso el usuario desactiva la función de applets en el navegador, este texto se mostrará en su lugar.
- **Archive.** Este atributo se utiliza para establecer varios archivos que el navegador precargará para poder ejecutar el applet correctamente. Este tipo de archivos serán clases y otros archivos necesarios que utilizará el applet.
- **Object.** Se utiliza este atributo para especificar un archivo de recursos donde se encontrará la imagen serializada con el estado del *applet*. Esta imagen se utilizará para cargar la implementación del *applet* para su ejecución.



TOME NOTA

El estándar de HTML no recomienda utilizar los atributos hspace, vspace y align, en su lugar utiliza las hojas de estilo CSS.

A continuación, verá un ejemplo de cómo insertar un *applet* en un documento HTML, en este caso se insertará un reloj:

```
<!DOCTYPE html>

<html>

<head>
    <title>Etiqueta object</title>
    <meta charset="UTF-8">
</head>

<body>
    <applet code="clock.class" width="200" height="450">
    </applet>
</body>

</html>
```

2.4. Formatos de audio y vídeo

Puede definir los formatos de audio y vídeo como maneras en las que guardar una misma información para que, posteriormente, pueda ser reproducida por diversos dispositivos que sean capaces de visualizar la información que hay almacenada en estos archivos.

En este epígrafe se estudiarán los tipos de formatos de audio y vídeo más utilizados y conocidos. Además, verá sus características especificando sus diferencias entre cada uno de ellos. También, verá cómo utilizar estos tipos de archivo para su correcta reproducción o visualización en los diferentes navegadores web que existen actualmente.

2.4.1. Descripción de los formatos de audio

Existen múltiples formatos de audio, sin embargo, los formatos más conocidos y que más se utilizan son: MP3, WAV, MIDI y OGG.

WAV

Las siglas WAV significan *WaveForm Audio File*. Este tipo de formato fue desarrollado en un principio por Microsoft para almacenar la información en archivos de audio.

Los archivos de audio que están comprimidos en este formato tienen las siguientes características:

- Este tipo de compresión de audio es ideal para comprimir audios originales mediante los cuales puede comprimir, a su vez, a otros tamaños y formatos para que puedan ser publicados en diferentes medios. Además, este tipo de formato de compresión aporta una gran calidad de audio.
- Una de las desventajas de este tipo de formato es que los archivos pesan bastante si los compara con otros tipos de compresión de audio. Por ejemplo, una canción en formato wav que tenga una calidad media puede ocupar unos 30Mb o 40Mb.
- Por otra parte, puede guardar los archivos wav con diferentes tipos de compresión. Las dos compresiones que se suelen utilizar con mayor frecuencia son la compresión PCM y la ADPCM. Sin embargo, aunque utilice estos tipos de compresión, los archivos wav seguirán siendo bastante pesados.
- Debido al gran peso de este tipo de archivos, si alguna vez se utiliza en una web solo será para clips de audio que duren unos tres o cuatro segundos.
- Por último, debe saber que los archivos en formato wav tienen una extensión .wav.



En la imagen se muestra el ícono de un archivo wav en Windows.

MP3

Este formato es uno de los más conocidos y utilizados en el mundo y fue desarrollado por el Instituto Fraunhofer. Las características de este formato son las siguientes:

- Este formato tiene un alto grado de compresión y aun así es capaz de mantener una buena calidad de audio. Por esta razón es el formato más utilizado a nivel mundial.
- Además, este es el formato de audio que se recomienda para implementar elementos de audio en las páginas web, puesto que, pesa poco y puede reproducirse en la mayoría de dispositivos de audio.
- Otra de las características de este formato es que resulta realmente sencillo convertir un archivo en formato wav a uno en formato mp3. Esta tarea puede realizarla en diversas herramientas gratuitas de escritorio y herramientas online.
- Por otro lado, la pérdida de calidad de audio al convertir a formato mp3 es mínima y no será perceptible en los dispositivos más comunes de reproducción de audio.
- Por último, los archivos con este formato tendrán una extensión .mp3.



Un archivo mp3 se muestra de la siguiente manera en Windows.



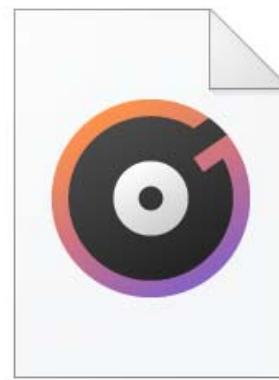
TOME NOTA

Un archivo mp3 ocupa unas diez veces menos que el mismo archivo en formato wav.

OGG Vorbis

Este formato es uno de los formatos más recientemente desarrollados. Su desarrollador es la Fundación Xiph.org. Este formato se conoce por las siguientes características:

- Este formato es de código abierto, característica que lo diferencia del formato mp3. Además, este formato aporta un grado de compresión similar al formato mp3, aunque hay que destacar que la calidad de sonido es mejor que la que ofrece el formato mp3.
- El inconveniente de este formato es que no todos los reproductores de audio son capaces de reproducirlo. La única solución a este problema es instalar en su dispositivo una serie de codecs necesarios para reproducir los archivos en este tipo de formato de audio.



- Por último, este formato puede utilizarse tanto para contener información de audio como de vídeo.

Este tipo de archivo se muestra en Windows de la siguiente manera:

MIDI

Las siglas de este formato significan *Musical Instrument Digital Interface*, o lo que es lo mismo, Interfaz digital para Instrumentos Digitales. Los archivos que están comprimidos en este tipo de formato llevan almacenadas las secuencias de los sintetizadores que aportan información sobre el instrumento que interviene en el audio y, además, cómo interviene y en qué momento lo hace. A continuación, va a ver las principales características de estos archivos:

Estos archivos pueden reproducirse en la mayoría de los dispositivos de reproducción de audio. Además, este tipo de archivo puede ser editado en cualquier programa de audio especializado, por ejemplo, Premier.

Los archivos MIDI normalmente tienen un tamaño pequeño. Esto se debe a que este tipo de archivos no almacena el sonido en sí, sino que dispone de la información necesaria para que el ordenador sea capaz de componer e interpretar los sonidos mediante una tarjeta de sonido.

Este tipo de archivo de audio se utilizaba antiguamente para agregar sonidos de fondo para páginas web y, también, para escuchar composiciones únicamente instrumentales, ya que, este tipo de formato no brinda una gran calidad de sonido como otros que ha visto anteriormente.

Para finalizar, los archivos MIDI tienen la extensión .mid.

2.4.2. Descripción de los formatos de vídeo

Ahora, va a aprender un poco más sobre los distintos formatos de vídeo que puede encontrar más comúnmente. Los archivos de vídeo más utilizados y, por lo tanto, más conocidos son: FLV, MPEG-4, H.264, WebM y Ogg Theodora.

MPEG-4

El formato *Moving Pictures Expert Group* o MPEG, como se conoce por sus siglas, es un formato estándar que se utiliza para la compresión de vídeo digital. Vea a continuación algunas de sus características:

- Este formato fue desarrollado por un grupo de trabajo llamado MPEG que pertenecía a un subcomité de ISO/IEC o International Organization for Standardization / International Electrotechnical Commission. Este grupo era el encargado de desarrollar los estándares internacionales para la compresión, procesado, descompresión y representación codificada de audio, vídeo y la combinación de ambas al mismo tiempo.

- Además, este tipo de formato es capaz de admitir diferentes tipos de códecs para su compresión. Vea a continuación cuáles son esos códecs: MPEG-1 comprime en calidad de CD, MPEG-2 comprime en calidad DVD, MPEG-3 es una compresión que está orientada al audio MP3, y, por último, MPEG-4 que es la compresión más orientada para usar en una web.
- Por otro lado, el formato MPEG-4 es un estándar para la codificación de audio y vídeo donde puede encontrar los códecs DivX y Xdiv para vídeo y AAC para audio. Estos códecs son muy comunes en los dispositivos, es por ello que es tan fácil encontrar dispositivos capaces de visualizar y reproducir los archivos en este formato.
- El formato MPEG-4 tiene características comunes con los formatos MPEG-1 y MPEG-2. En MPEG-4 una de las características más novedosas, en comparación con los formatos citados anteriormente, es que es capaz de soportar audio, vídeo, contenido 3D, codificación de baja velocidad binaria, entre otros. Además, también posee soporte para la gestión de derecho de autor o copyright.
- Actualmente, este formato es el que se suele utilizar para las televisiones HDTV desbancando así al formato MPEG-2 que era el que se utilizaba anteriormente a este.

H.264

A este formato también se le conoce como *MPEG-4 Advanced Video Coding*, por lo tanto, este formato es la evolución del formato MPEG-4 clásico. Vea las características de este formato a continuación:

- Una de las características fundamentales de este formato es que ofrece una extraordinaria calidad de vídeo aun teniendo una compresión muy elevada. Esta característica hace que este formato sea el predilecto para ser utilizado en los elementos de vídeo de una página web.
- Además, este formato no es libre, por lo que está sujeto a derecho de autor. Esto quiere decir que si un navegador reproduce un vídeo en este formato se podría pedir que pagara por hacerlo.
- También, hay que destacar que este formato de vídeo es muy utilizado para la codificación de vídeos en HD.
- Por último, los archivos que están comprimidos en este formato pueden tener la extensión .mp4 o .avi.



TOME NOTA

Los videos codificados en formato H.264 pueden ser distribuidos con el tamaño de sus fotogramas hasta cuatro veces mayor al tamaño original y, sin embargo, podrá reducir el ancho de banda, necesario para reproducirlo, en una tercera parte.

OGG Theodora

Este formato ya lo ha visto anteriormente cuando se describían los formatos de audio. Y, como ya se mencionó antes este tipo de formato, puede almacenar información tanto de audio, como de vídeo. Esta es la gran ventaja de este tipo de formato, puesto que, puede tener archivos de audio en formato OGG donde el formato de audio pertenezca al formato OGG Vorbis y el formato de vídeo al formato OGG Theodora. De esta manera, puede tener archivos audiovisuales sin tener la necesidad de utilizar otros formatos de pago.

WebM

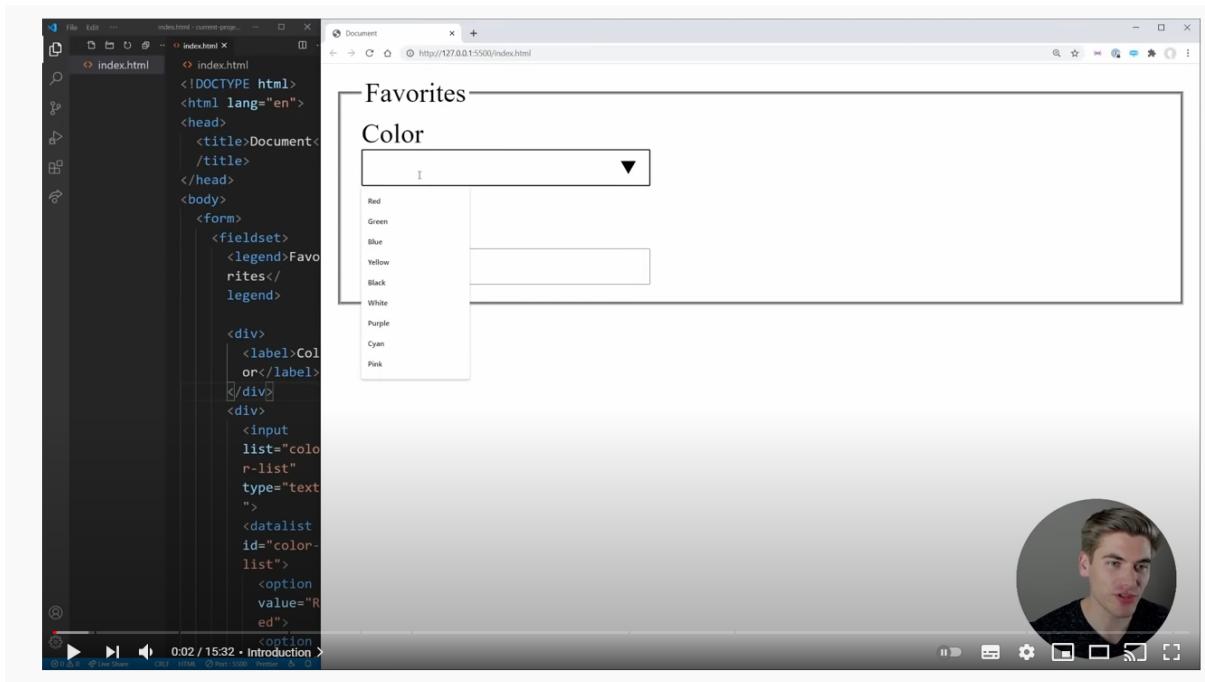
Este formato está desarrollado por Google y tiene las siguientes características:

- Es un formato de código abierto.
- Está basado en Matroska. Matroska es el famoso formato MKV que la ventaja que ofrece es que es capaz de adaptarse a cualquier tipo de formato.
- Además, el formato WebM utiliza el códec de vídeo VP8 y el códec de audio Vorbis.
- La desventaja que puede encontrarle a este tipo de formato es que, al ser un formato tan reciente, puede tener problemas para encontrar soporte. En cambio, la ventaja es que en un archivo con formato WebM puede almacenar información tanto de audio, como de vídeo.

FLV

Este formato está desarrollado por Adobe y sus principales características son:

- Uno de los inconvenientes de utilizar este formato es que, debe tener instalado previamente el reproductor o *plug-in* específico de Adobe Flash Player, si no, no podrá visualizar ni reproducir este tipo de archivos.
- Por otro lado, este formato utiliza los códecs On2 VP6 y Sorenson Spark. Estos dos códecs permiten una calidad visual muy buena.
- Los archivos de este tipo tendrán una extensión .flv.
- Otras de las ventajas es que puede encontrar múltiples reproductores que son capaces de reproducir vídeos en este formato. Algunos de los reproductores son: Xine, VLC, Riva, entre otros.
- Este es el formato que utilizan muchas páginas como Vimeo o Youtube. Esto se debe a que los archivos tienen una buena relación calidad/peso. Y, además, permiten hacer *streaming*. Vea, a continuación, un ejemplo de *streaming* en Youtube:



2.4.3. Configuración de los recursos para audio

Antes que nada, debe saber que el nuevo estándar de HTML5 ofrece nuevas etiquetas como `<audio>` y `<video>`. Estas nuevas etiquetas permiten que los navegadores sean capaces de reproducir audio y vídeo sin necesidad de instalar *plug-ins*. Sin embargo, ya debe saber que no todos los navegadores pueden producir todos los formatos de audio o vídeo. Por esta razón debe tener en cuenta que, al utilizar estas nuevas etiquetas para añadir audio o vídeo en una web, puesto que, si incluye ese archivo en un único formato, se arriesga a que algunos navegadores no pudieran reproducirlos.

Para poder solucionar este problema de incompatibilidad, las nuevas etiquetas `<audio>` y `<video>` permiten archivos que estén comprimidos en diferentes tipos de formato. Por ello, se recomienda añadir el mismo archivo con diferentes formatos, de esta manera se asegura de que todos los navegadores puedan reproducir el contenido.



TOME NOTA

El problema de que no todos los navegadores soporten los mismos formatos de compresión de archivos, se debe a las patentes.

A continuación, verá una tabla donde puede observar la compatibilidad que tienen algunos de los navegadores más conocidos y utilizados con los diferentes formatos de audio que se vieron anteriormente.

Vea la tabla a continuación:

Nombre del navegador	WAV	MP3	OGG Vorbis	WebM
Internet Explorer 8	No	No	No	No
Internet Explorer 9	No	Sí	No	No
Internet Explorer 10	No	Sí	No	No
Safari	No	Sí	No	No
Mozilla Firefox	Sí	No	Sí	Sí
Opera	Sí	No	Sí	Sí
Google Chrome	Sí	Sí	Sí	Sí

2.4.4. Configuración de los recursos para vídeo

Antes de proceder con los formatos de vídeo, debe saber que un formato contenedor es un tipo de formato de archivo que se utiliza para contener información tales como audio, vídeo, metadatos, etc. Por ejemplo, algunos formatos contenedores pueden ser AVI, MPG, MOV, ASF, MP4, etc.

A continuación, puede observar una tabla donde ver la compatibilidad que tienen ciertos formatos de vídeo con los navegadores más conocidos por el momento:

Nombre del navegador	MPEG-4	H.264	OGG Theodora	WebM
Internet Explorer 8	No	No	No	No
Internet Explorer 9	No	Sí	No	No
Internet Explorer 10	No	Sí	No	No
Safari	Sí	Sí	No	No
Mozilla Firefox	No	No	Sí	Sí
Opera	No	No	Sí	Sí
Google Chrome	No	Sí	Sí	Sí

2.5. Marquesinas

A continuación, puede ver un elemento denominado marquesina. Las marquesinas son ventanas donde puede ver un texto en desplazamiento. La etiqueta para utilizar marquesinas es <marquee>Texto</marquee>. Por defecto, al utilizar estas etiquetas, aparecerá el texto por el margen derecho y desaparecerá por el margen izquierdo.

Esta etiqueta no pertenece al estándar de HTML5, por lo que el uso de esta etiqueta está ya obsoleto. A continuación, va a ver una tabla donde puede observar la compatibilidad que tiene con los navegadores de escritorio:

Característica	Chrome	Firefox	Opera	Safari
Soporte básico	1.0	1.0 o anterior a 1.7	7.2	1.2
Atributo truespeed	Sin soporte	3.0	Sin soporte	Sin soporte
Hspace y vspace	Sin datos	3.0	Sin datos	Sin datos
Loop	Sin datos	3.0	Sin datos	Sin datos

Por otro lado, la compatibilidad que tiene esta etiqueta con dispositivos móviles es con el navegador Firefox mobile.

2.5.1. Los textos con movimiento

Como ya ha visto anteriormente, las marquesinas son una especie de ventanas en las que se muestra un texto en movimiento.

Vea a continuación un ejemplo de este tipo:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Marquesinas</title>

    <meta charset="UTF-8">

  </head>

  <body>

    <marquee>Esto es un texto en movimiento</marquee>

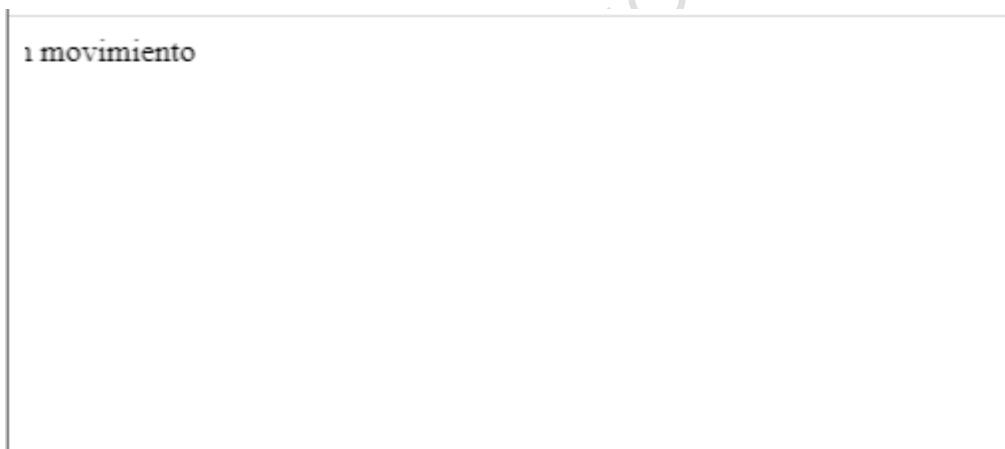
  </body>

</html>
```

Esto es un texto en movimiento

El texto de una marquesina se desplaza de manera que cuando desaparece por el lado izquierdo, vuelve a aparecer por el derecho. Vea el ejemplo:

Aquí, puede observar cómo desaparece el texto por la izquierda



En esta otra imagen, vea cómo aparece por la derecha:



En el epígrafe siguiente, se detalla cómo configurar la velocidad con la que aparece o desaparece el texto en las marquesinas.

2.5.2. Utilización de etiquetas para incluir marquesinas

Para crear una marquesina más personalizada, puede añadir más parámetro como, por ejemplo, anchura y altura utilizando las etiquetas *width* y *height* respectivamente. Los valores que les puede dar a estas etiquetas pueden ser un valor fijo en píxeles o un porcentaje de la pantalla. Actualmente, lo normal es trabajar con porcentajes, así, es como creará un diseño responsive que permita adaptarse a cualquier tipo de dispositivo. Vea un ejemplo:

```
<marquee width="40%" height=70>Soy una marquesina con el 40% de la ventana de ancho y 70 píxeles de alto</marquee>
```

También, puede añadir un alineamiento de texto en la marquesina utilizando la etiqueta *align*. Las opciones de alineamiento de texto en esta etiqueta son: *top*, si quiere que se alinee hacia arriba; *middle*, si quiere que se alinee a la mitad y *bottom*, si quiere que se alinee hacia abajo. Por lo que quedaría un ejemplo como el siguiente:

```
<marquee width=40% height=70 align=middle>Soy una marquesina con el texto alineado en el medio</marquee>
```

A la marquesina, también puede añadirle un color de fondo con la etiqueta *bgcolor*. Vea un ejemplo:

```
<marquee bgcolor="green">Soy una marquesina con el color de fondo verde</marquee>
```

Para cambiar el comportamiento de la marquesina, se utiliza la etiqueta *behavior*. Si utiliza la opción *alternate* en esta etiqueta, el texto se moverá de un lado a otro sin desaparecer. Vea el ejemplo:

```
<marquee behavior="alternate">Texto que no desaparece</marquee>
```

También, puede utilizar la opción *scroll* de la etiqueta *behavior* que permite que el texto aparezca de la misma manera que por defecto. Vea el ejemplo:

```
<marquee behavior="scroll">
```

Además, puede cambiar la dirección del texto con la etiqueta *direction*, donde puede elegir las opciones *right*, *left*, *up* y *bottom* en las que el texto se moverá hacia la derecha, hacia la izquierda, hacia arriba o hacia abajo, respectivamente. Vea un ejemplo.

```
<marquee direction="up">Soy una marquesina que se mueve hacia arriba</marquee>
```

Otra de las propiedades que puede utilizar es la propiedad *scrollamount* que indica cuántos píxeles avanzaría el texto. Por ejemplo, si quiere que el texto avance de 100 píxeles en 100 píxeles, se configura de la siguiente manera:

```
<marquee scrollamount=100>El texto avanza de 100 en 100 píxeles</marquee>
```

De esta manera, el texto avanzará muy rápido.

En cambio, si quiere que el texto avance más lento, se utiliza la etiqueta *scrolldelay* donde debe indicar cada cuántos milisegundos va a avanzar el texto, es decir, cuanto mayor sea el número indicado, más lento irá el texto. Va a poner un ejemplo donde el texto avance cada 1000 milisegundos, lo cual avanzará extremadamente lento. Vea el ejemplo.

```
<marquee scrolldelay=1000>Este texto avanzará muy lento</marquee>
```

Además, también puede establecer el número de veces que quiera que aparezca el texto. Debe saber que por defecto es infinito. Para ello, se utiliza la etiqueta *loop* e indicará un valor que serán las veces que quiera que aparezca el texto. Vea el ejemplo:

```
<marquee loop="2">Este texto solo aparecerá dos veces</marquee>
```

Puede establecer una separación entre el texto de la marquesina y el borde de la propia ventana de la marquesina. Para ello, se utilizan las etiquetas *hspace* y *vspace* y se establecen los píxeles de separación que quiera. *Hspace* sirve para la separación del margen izquierdo y derecho y *vspace* para la separación de los márgenes de arriba y debajo de la ventana. Vea un ejemplo:

```
<marquee vspace=20; hspace=15>Texto separa de los bordes de la ventana</marquee>
```

Por último, también puede cambiar la tipografía del texto de la marquesina aplicando las etiquetas *Font* con el atributo *fase*, todo ello fuera de la etiqueta *marquee*. Vea un ejemplo:

```
<Font fase="Verdana"><marquee>Texto con tipografía Verdana</marquee></Font>
```

Por último, el atributo *truespeed* se puede especificar con los valores: *true*, *false*, *on* y *truespeed*. Este atributo se utiliza en conjunto con el atributo *scrolldelay*, ya que este atributo ignorará los valores por debajo de 60. Sin embargo, al utilizar el atributo *truespeed*, los valores por debajo de 60 serán aceptados. Vea un ejemplo:

```
<marquee scrollamount="3" scrolldelay="6" truespeed="truespeed">¡Hola mundo! <marquee>
```

2.5.3. Las marquesinas y los distintos navegadores

Como ya ha visto, el uso de la etiqueta *<marquee>* está obsoleto. Esto ocurre porque esta etiqueta solo era soportada en el navegador Internet Explorer. Por lo tanto, si hacía una web con este elemento, gran parte de ese contenido no se podía mostrar a no ser que el usuario utilizara el navegador Internet Explorer.

Sin embargo, actualmente la mayoría de los navegadores soportan esta etiqueta, pudiendo mostrar así su contenido de manera correcta. De hecho, las imágenes que ha visto anteriormente sobre el movimiento de las marquesinas están tomadas en el navegador Google Chrome.

Aunque actualmente esta etiqueta es soportada por la mayoría de los navegadores, no debe olvidar que según el estándar de HTML5 está obsoleta y, por tanto, debe intentar no hacer uso de la misma.

En su lugar, para crear un texto en movimiento, puede utilizar JavaScript o incluir imágenes GIF en su defecto.



TOME NOTA

Como ya sabe, la característica más conocida de las imágenes GIF es que se puede crear una secuencia de una imagen para que aparente que esta se mueve.

A continuación, se verá un ejemplo utilizando algunos de los elementos que ha aprendido en este tema. Lo que se va a hacer es un documento que contendrá un juego que utiliza los elementos de marquesina. Este juego se basa en un conjunto de gif donde tendrá que pulsar la imagen seleccionada para poder ganar. Esta imagen estará rodeada por otras imágenes e irán a gran velocidad. Vea, a continuación, el código:

```
<!DOCTYPE html>

<html>

<head>

<title>Ejercicio marquesina multimedia</title>

<meta charset="utf-8"/>

<link rel="icon" type="image/jpg" href="imagenes/textura.jpg"/>

</head>

<body>

<center>

<h1>Juego de imágenes</h1>

<p>¡Clica la imagen para ganar!</p>



<br/>

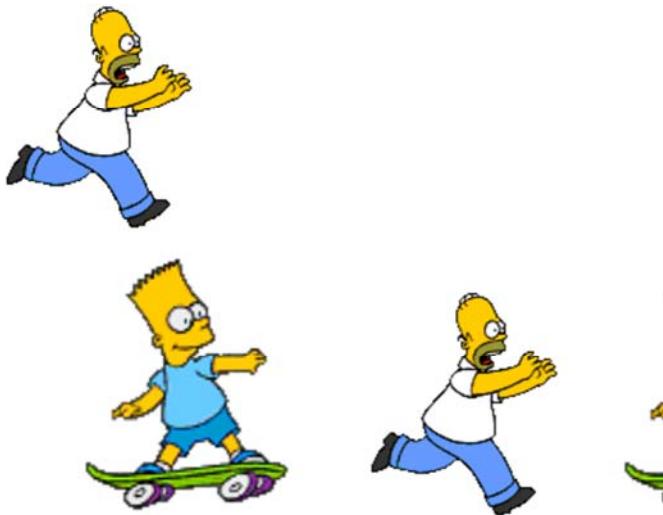
<br/>
```

```
<marquee scrollamount="40" direction="right" width="75%">  
      
  
    <a href="ganador.html">  
          
    </a>  
  
      
  
</marquee>  
  
</center>  
  
</body>  
  
</html>
```

Si ejecuta este código en un navegador, mostrará lo siguiente.

Juego de imágenes

¡Clica la imagen para ganar!



3. TÉCNICAS DE ACCESIBILIDAD Y USABILIDAD

Las técnicas de accesibilidad y usabilidad son conceptos muy utilizados en el mundo del diseño y desarrollo web, pero no debe confundir ambos ya que, en realidad, son conceptos diferentes.

La accesibilidad se refiere a las normas que rigen el diseño de un sitio web. Gracias a ellas, se crean mecanismos que facilitan el acceso al contenido de la página web para usuarios con discapacidades, ya sean auditivas, visuales, entre otras. En la accesibilidad también se engloba cómo diseña el acceso a la web desde diferentes dispositivos donde poder visualizarla.

La usabilidad se define como el conjunto de recomendaciones a tener en cuenta para diseñar el sitio web, siendo conscientes de las desventajas que puede suponer un diseño ineficiente o que no esté pensado para todo tipo de usuario.



3.1. Accesibilidad web, ventajas de la accesibilidad

La accesibilidad web se ha convertido en un aspecto muy importante para todo tipo de usuario que tenga acceso a contenido en la web. Además, es un concepto que implica igualdad de condiciones para todos los usuarios ya que, ayuda a la participación activa en la sociedad de personas con discapacidad.

Se estima que un 20% de la población con acceso a Internet tiene algún tipo de discapacidad. Para empresas es un porcentaje que cuenta como potenciales clientes. Para instituciones públicas, no tener en cuenta estas normas, supone una infracción de la ley.

En el siguiente vídeo, de siete minutos de duración, se describen los tributos de accesibilidad:

RECURSO MULTIMEDIA

3.1.1. Definición de Accesibilidad

El concepto de accesibilidad no solo implica el estudio de un diseño que facilite el uso de la web a personas con discapacidad. La accesibilidad, además de este cometido, se ha convertido en un hito

dentro del desarrollo web cuya finalidad es diseñar páginas web accesibles desde diferentes tipos de dispositivos con los que puede conectarse a Internet.

Vea a continuación el tipo de dificultades de acceso a la información de la WWW que puede tener una persona con discapacidad:

- **Deficiencias visuales:** La ceguera, la visión baja y la ceguera al color.
- **Deficiencias auditivas:** Aunque es menos utilizada que la visión, es un aspecto muy a tener en cuenta en el desarrollo de la accesibilidad web.
- **Deficiencias motrices:** Es la incapacidad para utilizar el ratón o el teclado, o tener un tiempo de respuesta lento, dificulta el uso adecuado de los periféricos. En este caso se usan otros periféricos alternativos basados en la voz o incluso en la vista.
- **Deficiencias cognitivas y de lenguaje:** Dificultad para el aprendizaje, la distracción o la dificultad para recordar o concentrarse.

Todas estas categorías tienen, además, multitud de variantes y niveles de gravedad.

Además, hay que tener en cuenta que un usuario puede tener una o varias discapacidades, permanentes o transitorias. También hay que tener en cuenta la multitud de dispositivos que hay y que cada usuario puede usar particularmente, es decir, una misma página web la puede ser visualizada en una pantalla grande o en un móvil con pantalla pequeña por un usuario con dificultades visuales. Esta página debería ser capaz de adaptarse a ambos dispositivos y, en este caso, con un diseño en el que se puedan agrandar el texto para facilitar la lectura o visualización de imágenes, o que pueda activarse lectura por voz. También existen dificultades relacionadas con el dispositivo que usa y el ambiente en el que se encuentra el usuario como, por ejemplo, un lugar ruidoso en el que no sea audible el ayudante de voz para leer las páginas.

3.1.2. Aplicabilidad de la Accesibilidad

Para asegurarse de que una página web se desarrolla con garantías de ser realmente accesible, es necesario entender el concepto y comprometerse a cumplirlos, se deben aprender los conocimientos técnicos aplicarla y, sobre todo, conocer y aplicar la normativa vigente sobre accesibilidad web.

En general, el concepto de accesibilidad está muy infravalorado por desarrolladores, ya sea por desconocimiento o por desinterés, forjando así una creencia sobre que la accesibilidad supone un esfuerzo poco rentable en el momento de crear el sitio web, por lo que, queda en el olvido.



Aunque el problema ya es un viejo conocido, muchas organizaciones no se comprometen a garantizar una accesibilidad en los sitios web que desarrollan. Para garantizarla, deben existir políticas internas que regulen los procedimientos y normas para comprometerse, realmente, en el desarrollo de un sitio web accesible.

Respecto a los conocimientos técnicos, para un desarrollador web experimentado, no supondría un gran esfuerzo aprender los conceptos y técnicas para una eficiente accesibilidad web. Lo más importante es estar en continuo aprendizaje sobre las nuevas técnicas y las nuevas normativas.

La accesibilidad es una problemática considerada en ámbitos nacionales e internacionales, por ello, se acuerdan una serie de recursos para especificar las características que deben tener los contenidos en la web para que puedan ser usados por el mayor número de personas. En la web oficial de la Administración Electrónica del Gobierno de España se puede encontrar las directivas, regulaciones y leyes vigentes sobre las normas de accesibilidad que han de cumplirse por parte de las instituciones y demás profesionales del ámbito del desarrollo web.

En el siguiente enlace puede aprender sobre la cantidad de normativas que rigen las [Normas de accesibilidad](#).

Observe además algunas de las principales leyes y decretos españolas vigentes sobre accesibilidad web:

- LEY 34/2002, de 11 de julio, de servicios de la sociedad de la información y de comercio electrónico.
- LEY 51/2003, de 2 de diciembre, de igualdad de oportunidades, no discriminación y accesibilidad universal de las personas con discapacidad.
- REAL DECRETO 1414/2006, de 1 de diciembre, por el que se determina la consideración de persona con discapacidad a los efectos de la Ley 51/2003, de 2 de diciembre, de igualdad de oportunidades, no discriminación y accesibilidad universal de las personas con discapacidad.
- REAL DECRETO 366/2007, de 16 de marzo, por el que se establecen las condiciones de accesibilidad y no discriminación de las personas con discapacidad en sus relaciones con la Administración General del Estado.
- LEY 11/2007, de 22 de junio, de acceso electrónico de los ciudadanos a los Servicios Públicos.
- REAL DECRETO 1494/2007, de 12 de noviembre, por el que se aprueba el Reglamento sobre las condiciones básicas para el acceso de las personas con discapacidad a las tecnologías, productos y servicios relacionados con la sociedad de la información y medios de comunicación social.
- LEY 27/2007, de 23 de octubre, por la que se reconocen las lenguas de signos españolas y se regulan los medios de apoyo a la comunicación oral de las personas sordas, con discapacidad auditiva y sordociegas.
- LEY 49/2007, de 26 de diciembre, por la que se establece el régimen de infracciones y sanciones en materia de igualdad de oportunidades, no discriminación y accesibilidad universal de las personas con discapacidad.
- LEY 56/2007, de 28 de diciembre, de Medidas de Impulso de la Sociedad de la Información.

- LEY 7/2010, de 31 de marzo, General de la Comunicación Audiovisual.
- LEY 26/2011, de 1 de agosto, de adaptación normativa a la Convención Internacional sobre los Derechos de las Personas con Discapacidad.
- Norma UNE 139803:2012: Requisitos de accesibilidad para contenidos en la web.
- Norma UNE 139802:2009: Requisitos de accesibilidad del software.



Más Info

RECURSO MULTIMEDIA



3.1.3. Descripción de las ventajas de la Accesibilidad

La accesibilidad conlleva una infinidad de ventajas para todos los usuarios del sitio web, además de para los desarrolladores que la diseñan. En este epígrafe va a ver algunas de las más importantes.

a) Facilidad de acceso

Debe entender que una web accesible es una web que cumple con los estándares y, por lo tanto, un mayor número de usuarios puede acceder a ella. Esto sucede porque hay usuarios que no pueden acceder a Internet de manera convencional, ya sea porque son invidentes, sordos, etc. Y necesitan software de terceros que interpreten la información de la página web.

Además, en relación al SEO, una página que conste de accesibilidad será una página mejor posicionada que una que no cuente con ello. Por otra parte, debe saber que una web que cumpla con los estándares de accesibilidad, es una web con mayor usabilidad, cosa con la que debe contar a la hora de diseñar un sitio web. Porque lo que más interesa es que la página sea visitada por el mayor número de usuarios posible.

b) Mejoras en la navegación

Debe tener en cuenta que las páginas que cuentan con la accesibilidad recomendada, también cuentan con ventajas relacionadas con la facilidad del usuario para navegar a través de los contenidos de la misma. Vea a continuación, algunas de estas ventajas sobre la mejora de la navegación en un sitio web:

- **Texto alternativo en cada imagen de la web.** Al añadir un texto alternativo en cada una de las imágenes de su web, permitirá a las personas invidentes que no puedan ver con sus

propios ojos estas imágenes o simplemente porque el navegador no cargue las imágenes, puedan tener una idea de las imágenes que hay en su web. Esto es importante porque las imágenes también forman parte de la información que aporta su web.

Para añadir un texto alternativo a las imágenes de su web, se utiliza el atributo alt que, a su vez, irá dentro de la etiqueta que contendrá la imagen. Vea un ejemplo de código a continuación:

```
<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Ejemplo de imagen</title>

</head>

<body>

<img src="" alt="Esto es una imagen">

</body>

</html>
```

Ahora, va a ver cómo se mostraría en el navegador este texto al no poder cargar la imagen:

Esto es una imagen

- **Controlar el uso de JavaScript.** Recuerde que JavaScript es un lenguaje de script que puede combinar con HTML para crear efectos y hacer que las páginas web sean más dinámicas.
-

Además, este lenguaje es un lenguaje interpretado como HTML, esto quiere decir que no necesita ser compilado y que será interpretado por un navegador. Por lo tanto, si los usuarios desactivan JavaScript en su navegador, todas las características de su web que estén diseñadas con este lenguaje no serán mostradas para este usuario. Es por ello que debe diseñar su web de manera que, aunque un usuario desactive JavaScript, este sea capaz de navegar por su web sin problema alguno.

- **Descripción para el destino de enlaces.** Como atributo de la etiqueta `<a>`, tiene el atributo `title` que se utiliza para que aparezca un texto al dejar el puntero del ratón sobre un enlace. Este atributo debe acostumbrarse a usarlo para añadir de manera clara hacia donde van a redirigir los enlaces de la web. En caso de que los enlaces se utilicen para descargas de archivos, también debe informar de ello. Esto les aporta a los usuarios seguridad y confianza en su página. Vea a continuación un ejemplo de código:

```
<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Ejemplo de enlace</title>

</head>

<body>

<a href="https://www.google.es" title="Será redirigido a Google">Soy un enlace</a>

</body>

</html>
```

Lo siguiente que verá es cómo lo mostraría el navegador:



- **Destacar elementos importantes de la web.** Recuerde las etiquetas <h1>, <h2>, <h3>, <h4>, <h5> y <h6>. Estas etiquetas las utilizará para destacar mejor la estructura semántica de su web, ya que, es importante darle a la web una estructura coherente para, de ese modo, facilitar la navegación al usuario. Así, de esta manera destacará qué información es primordial y cuál secundaria.
- **Organización de menús de navegación.** Si es capaz de estructurar los menús de navegación de manera intuitiva para el usuario, se creará una web donde el usuario va a tener una buena experiencia de usuario y, por lo tanto, volverá a su web. Por el contrario, si a los usuarios les parece que su web es liosa, lo normal, es que no duren mucho navegando por ella. Si esto ocurre, nunca va a salir de los primeros en ningún buscador y su web caerá en el olvido porque estará enterrada bajo millones de webs.

c) Independencia de los navegadores

Para que su web sea accesible, debe tener en cuenta que pueda ser visualizada, de manera independiente, desde cualquier dispositivo y navegador. Esto es muy importante porque no se sabe qué tipo de navegador o dispositivo va a utilizar un usuario para acceder a su web. Sin embargo, debe asegurarse de que pueda acceder completamente desde cualquier navegador.

Además, debe recordar que actualmente, el uso de dispositivos móviles está por encima del uso del ordenador convencional. Por ello, es muy importante que haga webs de tipo responsive para que puedan ser visualizadas desde los dispositivos móviles correctamente.

Además, de todas las ventajas que ha visto hasta ahora, puede enumerar unas cuantas más a continuación:

- **Aportar accesibilidad** a su web hace que cumpla con los derechos de los ciudadanos en cuanto a que les da la opción de poder participar y no ser discriminados debido a su discapacidad.
- Cumplir con la **normativa** que exige la Administración Pública sobre la accesibilidad.
- **Maximizar el número de usuarios** que pueden acceder a su web, y con ello, lograr mejores resultados de tráfico en Internet y mejorar el posicionamiento de su web.
- Todo ello mejorará la **indexación** de su página en los buscadores. Esto se debe a que, si cumple con los estándares, con la semántica y estructura de su web, los buscadores serán capaces de identificar e interpretar mejor los contenidos de su web. Por lo tanto, mejorará su posicionamiento en Internet.

	<p><i>Según las estadísticas un 9% de la sociedad en España presenta algún tipo de discapacidad. Es por esto que siempre que desarrolle una web debe pensar en su accesibilidad.</i></p>
TOME NOTA	

3.1.4. Usabilidad web, importancia de la usabilidad

La accesibilidad es uno de los puntos clave del desarrollo y el diseño web. Si no ofrece esa accesibilidad, su web nunca va a aparecer una de las primeras, ya que, actualmente los buscadores ponen mucho hincapié en esta práctica.

Por otro lado, cuando se habla de usabilidad web, se hace referencia a la experiencia de usuario cuando interactúa con una página web. En este epígrafe, se hablará sobre varios aspectos que están relacionados con la usabilidad de una web. También tiene que tener muy presente que el diseño y el desarrollo web sean profesionales y eficientes.

3.1.4.1. Definición de usabilidad

Cuando se habla de un sitio web con alta usabilidad, se está hablando de una web en la que su contenido se muestra de manera clara y concisa. Dos elementos que hacen que, para el usuario, consumir esa información, sea fácilmente accesible.

	<p>Actualmente, es realmente imposible crear una web que resulte clara y eficiente para todos los usuarios del planeta. Sin embargo, un desarrollador debe esforzarse por mostrar el contenido de la web de la manera más clara posible, de forma que el contenido confuso sea mínimo.</p>
TOME NOTA	

Al comienzo del desarrollo web, la usabilidad siempre se había ignorado por parte de los propios desarrolladores web. Sin embargo, a partir de que WWW empezó a popularizarse, es cuando se le dio la importancia que tiene hoy y se comenzó a añadir como parte del diseño y desarrollo web.

Como ya se mencionó al principio de este tema, la usabilidad web se puede entender como un conjunto de recomendaciones que debe tener en consideración al desarrollar y diseñar una web. Puesto que, estas recomendaciones están relacionadas con que los usuarios puedan acceder a la información de su web de manera sencilla.

Si ha comprendido todo lo que significa la accesibilidad, se habrá dado cuenta de que este es un aspecto que puede determinar el éxito o fracaso de una página web. La razón de esto es que, si un sitio web no está estructurado de la manera en la que el usuario desea utilizarlo, todos los aspectos positivos pasarán a un segundo plano y el usuario dejará de utilizar la web.

Por otro lado, puede pensar que los aspectos relacionados con la accesibilidad web, son aspectos que pueden ser subjetivos, puesto que, depende del individuo que la utilice. Sin embargo, existen ciertos parámetros para aportarle al usuario eficiencia, eficacia y satisfacción. Y esos parámetros son totalmente medibles para poder evaluar el grado de usabilidad de su sitio web.



TOME NOTA

En Internet se puede topar con montones de páginas web que tienen muy buenos contenidos. Sin embargo, debido a los problemas de accesibilidad que presentan, no tienen muchas visitas de usuarios.

3.1.4.2. Interacción web-individuo

Este aspecto es otro que debe tener en cuenta en el desarrollo de una web, ya que, es muy importante a la hora de desarrollar un sitio web usable. Cuando se diseña la interactividad en un sitio web, debe adelantarse a cómo va a actuar el usuario a las diferentes acciones que le ofrecen en la web. De este modo, puede desarrollar la interactividad de la web de manera que los elementos funcionen fluidamente.

Ahora, va a ver las consideraciones que debe tener en cuenta a la hora de diseñar la interactividad de una web:

- En una web, todos los botones, enlaces y elementos con los que el usuario pueda interactuar, deben diferenciarse completamente del resto de elementos y de contenidos.
- Debe hacer que la interacción con los elementos de la web sea rápida y sencilla. Además, debe minimizar los clics que el usuario debe realizar para navegar por la web.
- No debe utilizar en la web elementos interactivos que tengan una interacción que pueda confundir al usuario cuando los utilicen, es decir, que un elemento se comporte una forma que no debería comportarse. Por ejemplo, a la hora de crear formularios, es muy común que los desarrolladores cometan este error y utilicen elementos para realizar acciones que no tienen nada que ver con la naturaleza del elemento en sí.

Vea a continuación un ejemplo de código de formulario en HTML5:

```
<!DOCTYPE html><html>

<head>

<title>Formulario de prueba de HTML5</title>

</head>

<body>

<form action="" method="get">

<p>
```

Nombre: <input type="text" name="name_control" autofocus required />

</p>

<p>

Correo electrónico: <input type="email" name="email_control" required />

</p>

<p>

URL: <input type="url" name="url_control" placeholder="Escribe la URL de tu página web personal" />

</p>

<p>

Fecha: <input type="date" name="date_control" />

</p>

<p>

Tiempo: <input type="time" name="time_control" />

</p>

<p>

Fecha y hora: <input type="datetime" name="datetime_control" />

</p>

<p>

Mes: <input type="month" name="month_control" />

</p>

<p>

Semana: <input type="week" name="week_control" />

</p>

```
<p>Número (min -10, max 10): <input type="number" name="number_control" min="-10" max="10" value="0" />
```

```
</p>
```

```
<p>
```

```
Teléfono: <input type="tel" name="tel_control" />
```

```
</p>
```

```
<p>
```

```
Término de búsqueda: <input type="search" name="search_control" />
```

```
</p>
```

```
<p>
```

```
Color favorito: <input type="color" name="color_control" />
```

```
</p>
```

```
<p>
```

```
<input type="submit" value="Enviar" />
```

```
</p>
```

```
</form>
```

```
</body>
```

```
</html>
```

Ahora, vea cómo se muestra el navegador los elementos y cómo se diferencian los elementos interactivos de los que no lo son:

Nombre:

Correo electrónico:

URL: Escribe la URL de tu página

Fecha: dd / mm / aaaa

Tiempo: -- : --

Fecha y hora:

Mes: ----- de -----

Semana: Semana -- , -----

Número (min -10, max 10): 0

Teléfono:

Término de búsqueda:

Color favorito:

3.1.4.3. Aplicabilidad de la usabilidad

Cuando un desarrollador web quiere crear una web accesible o modificar una web ya construida para darle accesibilidad, el primer pilar y más fundamental es que sea consciente de que para alcanzar ese nivel de accesibilidad, debe ofrecer la mejor experiencia de usuario que sea capaz de desarrollar en la web. Cuando se llega a este nivel de entendimiento y conciencia, es cuando se empezará a desarrollar el sitio web y a pensar las estrategias para lograrlo.



Cuando se habla de usabilidad, no existe un nivel mínimo o máximo que pueda aplicar en su web, sino que es un concepto que se mueve por el gradiente de si la web es usable o no.

TOME NOTA

Para llevar a cabo una planificación para establecer las medidas de usabilidad en su web, debe hacer un planteamiento donde responder las siguientes preguntas:

- ¿Cómo se van a ejecutar las medidas?
- ¿Por dónde debe comenzar?
- ¿Qué beneficios van a proporcionar los cambios?
- ¿Qué va a suponer la efectuación de esos cambios?
- ¿La planificación es igual para todos?
- ¿Cuál será el desembolso económico para la implementación de estas medidas?
- ¿Cuántos profesionales voy a necesitar para ello?

Para poder tener una idea de cómo se ejecutarán las medidas, debe realizar un proceso donde se lleven a cabo las siguientes acciones:

- Analizar las necesidades que se van a cubrir.
- Definir un plan de acción que seguir para llevar a cabo este propósito.
- Estudiar diversos métodos a ejecutar.
- Por último, estudiar un grupo de recomendaciones que están relacionadas con el plan que habrá definido.



TOME NOTA

A la hora de implementar cualquier medida de usabilidad en una web, debe encontrar los puntos en los que se encuentran el equilibrio entre el objetivo que quiera para el sitio y lo que los usuarios esperan del mismo.

3.1.4.4. Recursos sobre usabilidad

Para que comprenda el concepto de usabilidad, puede encontrar en Internet diversos recursos que van a ayudar a ello, ya que, van a facilitar la implementación de las recomendaciones en sus proyectos de desarrollo web. Vea a continuación algunos de estos recursos:

- [**WAI**](#). Este sitio se llama Web Accessibility Initiative o, lo que es lo mismo en español, Iniciativa para la Accesibilidad Web. Esto es una rama del W3C que es la encargada de velar por la accesibilidad de la WWW. Vea a continuación el portal del W3C:

News

RTC Accessibility User Requirements (RAUR) Note Published

(2021-05-25)

[RTC Accessibility User Requirements \(RAUR\)](#) is published as a Working Group Note. Real-time communication (RTC) provides real-time peer to peer audio, video, and data exchange directly between supported user agents. This enables instantaneous applications for video and audio calls, text chat, file exchange, screen sharing, and gaming. RAUR describes various accessibility related user needs, requirements, and scenarios for real-time communication (RTC) applications. These user needs should drive accessibility requirements in various related specifications and the overall architecture that enables RTC.

Accessibility Translations: Getting the 'Global' in GAAD

(2021-05-20)

All WAI Translations now lists translations in 35 languages. Thanks to the translators. (If you know anyone who might be interested in contributing to more accessibility translations, please point them

See what we have for you:

Get Resources for...

- Content Writers
- Designers
- Developers
- Evaluators, Testers
- Managers
- Policy Makers
- Trainers, Educators
- Web Users, People with Disabilities, Advocates
- Other Languages

Training Course:

Digital Accessibility Foundations

- [**Usability.gov**](#). Este es el sitio web que está desarrollado por el departamento de salud y servicios Humanos de Estados Unidos. Vea a continuación el portal:

- [**UPA**](#). En esta web puede encontrar servicios sobre experiencia de usuario y accesibilidad. Vea su portal:



- **Test de Usabilidad**. En este sitio web puede encontrar de manera detallada cómo hacer un test de usabilidad a su web. Vea su portal:



- **Usabilidad.tv**. En esta web, puede encontrar mucha información sobre la usabilidad y accesibilidad web. Vea su portal:

The screenshot shows the homepage of Usabilidad.TV. At the top, there's a banner with a globe graphic and the text "usabilidad.tv ENTRA EN LA RED". Below the banner is a navigation menu titled "USABILIDAD WEB" with links like "Usabilidad - Página principal", "Participa con nosotros", "Usabilidad_web", "Buscadores", "Posicionamiento_en buscadores", "Tutorial_HTML", and "Tutorial_CSS". To the right of the menu, there's a main content area with a heading "Usabilidad .TV" and a sub-section "Te damos la bienvenida a usabilidad .TV". It includes a brief description and a bulleted list under "A lo largo de estas páginas encontrarás información sobre:". The list includes: "• USABILIDAD WEB: Llamamos usabilidad a la experiencia que tiene un usuario cuando interactúa con páginas de un web. Te presentamos algunas consideraciones de como diseñar tus páginas web para que sean accesibles para todo tipo de personas con discapacidad o accesibles simplemente para un usuario sin dificultad alguna.", "• BUSCADORES: Conocé todo sobre los buscadores la Red, te presentamos noticias y artículos sobre los buscadores.", "• POSICIONAMIENTO EN BUSCADORES: El posicionamiento consiste en mejorar sus páginas web para conseguir los primeros puestos en los buscadores.", and "• TUTORIALES: tutoriales de HTML, CSS". Below this is a section titled "VÍDEOS DESTACADOS DE USABILIDAD WEB" which currently displays a placeholder message: "Adobe Flash Player ya no está disponible".

- **Nosolousabilidad**. Esto es una revista digital en la que puede encontrar artículos muy interesantes que informan sobre la usabilidad en la web. Vea su portal:

The screenshot shows the homepage of the nro magazine website. The header features the nro logo and the tagline "no solo usabilidad: revista sobre personas, diseño y tecnología". Below the header, there are navigation links for "PORTADA", "PUBLICAR", "ARCHIVO", "AUTORES", "LIBROS", "ACERCA DE", "Twitter", and "Search". The main content area features a large image of a wall covered in sticky notes during a user-centered design workshop. Overlaid on the image is the title "Modelo de Integración Genérica para Diseño Centrado en el Usuario MIG-DCU". Below the title, it says "22 de Abril de 2019, n. 18 (preprint)." and "Marmolejo, Melba Cristina".

TOME NOTA

Se puede pensar que conceptos que están relacionados con la usabilidad y accesibilidad web, son conceptos totalmente subjetivos. Sin embargo, existen ciertos aspectos que prueban la eficiencia, satisfacción y eficacia de un sitio web, lo que ayuda a evaluar el grado de usabilidad de la misma.

3.1.5. Aplicaciones para verificar la accesibilidad de sitios web (estándares)

En Internet puede encontrar múltiples aplicaciones que van a permitir evaluar y verificar la accesibilidad de un sitio web. En este epígrafe, va a aprender sobre el funcionamiento y las características de las herramientas más importantes que puede encontrar en Internet y que van a permitir comprobar el grado de accesibilidad de su página web.

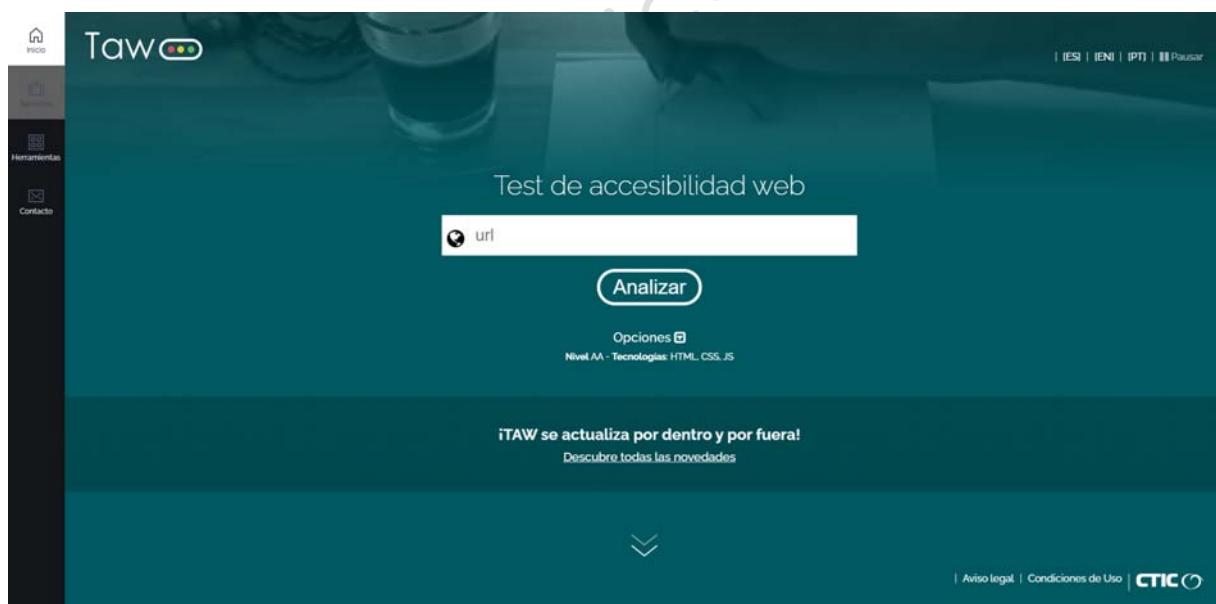
Recuerde que darle accesibilidad a su web va a ofrecerle, a usted y a los usuarios, un conjunto muy amplio de ventajas.

3.1.5.1. Recursos web estándares

En este epígrafe, se verán algunos de los recursos web más valiosos que permitirán evaluar el grado de accesibilidad de los sitios web.

TAW

Esta herramienta es un analizador automático de los elementos que componen una web. Esta herramienta, la ventaja que tiene es que está en castellano y ofrece una versión ejecutable en la dirección web siguiente: <https://www.tawdis.net>. Vea una imagen donde se muestra esta herramienta:



Para que esta herramienta de por buena la accesibilidad de la que dispone su web, su web no deberá contener ningún problema de prioridad tipo 1, ya sea de tipo manual o automático.

Si quiere utilizar esta herramienta para analizar una web, lo primero que debe hacer es seleccionar la normativa con la que la herramienta va a analizar la web. Estas normativas pueden ser: MobileOK, WCAG 1.0 o WCAG 2.0. Además, tiene que seleccionar el nivel de análisis que tenga intención de validar. Cuando haya finalizado, se insertará la url de la web que quiera analizar y pulsar el botón analizar. Después, la aplicación mostrará los resultados.



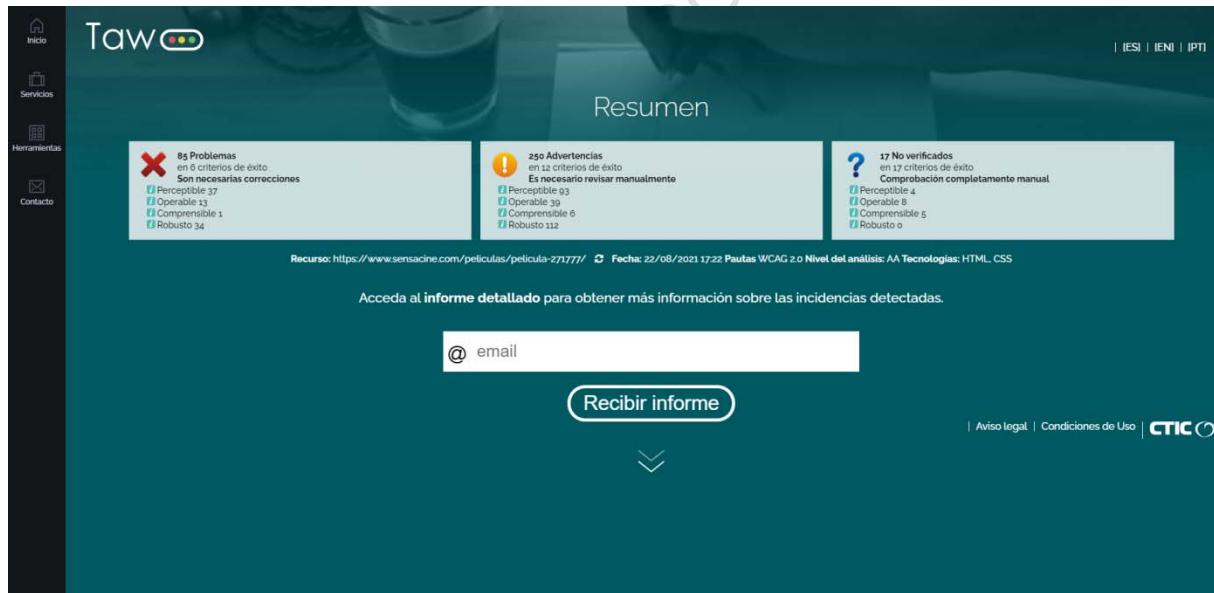
La herramienta TAW, además de poder analizar páginas web de manera individual, ya sea en remoto o en local, también puede ser configurada para analizar un sitio web completo.

TOME NOTA

Esta herramienta es capaz de distinguir entre dos tipos de problemas de accesibilidad:

- **Problema de tipo manual:** Este tipo de problemas requieren que un usuario real los verifique, puesto que estos problemas están relacionados con aspectos que la aplicación es incapaz de verificar. Estos problemas serán visualizados como advertencias.
- **Problema de tipo automático:** Estos problemas son los que la propia aplicación es capaz de verificar y solucionarlos.

A continuación, se hará una prueba de uso de esta herramienta. Para ello, se va a tomar como ejemplo una página de estrenos de cine y va a analizarla con la aplicación. Estos son los resultados que muestra:



Como puede ver, esta página que se ha cogido como ejemplo no pasaría la prueba de accesibilidad por todos los errores que contiene. Más abajo, puede ver cada uno de los errores que contiene y de qué tipo son. Por ejemplo, a continuación, puede ver los errores perceptibles, que son los errores que se pueden ver a simple vista por cualquier usuario:

Perceivable

La información y los componentes de la interfaz de usuario deben ser presentados a los usuarios de modo que puedan percibirlos.

Pauta	Nivel	Resultado	Problemas	Advertencias	No verificados
1.1-Textos alternativos			34	35	0
1.1.1 - Contenido no textual	A	X	34	35	0
1.1.2 - Medios basados en el tiempo	A	na	0	0	0
1.1.3 - Solo audio y solo vídeo (grabaciones)	A	na			
1.1.4 - Subtítulos (pregrabados)	A	na			
1.1.5 - Audioldescripción o Medio Alternativo (Pregrabado)	A	na			
1.1.6 - Subtítulos en directo	AA	na			
1.1.7 - Descripción auditiva (Pregrabada)	AA	na			
1.1.8 - Adaptable			3	40	1
1.1.9 - Información y relaciones	A	X	3	44	
1.1.10 - Secuencia con significado	A	!		4	
1.1.11 - Características sensoriales	A	?			1
1.4-Distinguible			0	9	3
1.4.1 - Uso del color	A	?			1
1.4.2 - Control del audio	A	na			
1.4.3 - Contraste (Mínimo)	A	?			1
1.4.4 - Redimensionamiento del texto	AA	!	9		
1.4.5 - Imágenes de texto	AA	?			1

Más adelante, se ven los errores operables, es decir, todos aquellos que tienen que ver con el funcionamiento de la interfaz de usuario:

Operable

Los componentes de la interfaz de usuario y la navegación deben ser operables.

Pauta	Nivel	Resultado	Problemas	Advertencias	No verificados
2.1-Accesible mediante el teclado			0	0	1
2.1.1 - Teclado	A	?			1
2.1.2 - Sin bloques de teclado	A	?			1
2.2-Tiempo suficiente			0	0	1
2.2.1 - Tiempo ajustable	A	?			1
2.2.2 - Pausar, detener, ocultar	A	?			1
2.3-Provocar ataques			0	0	1
2.3.1 - Umbral de tres destellos o menos	A	?			1
2.4-Navegable			13	39	4
2.4.1 - Evitar bloques	A	?			1
2.4.2 - Páginas tituladas	A	!		1	
2.4.3 - Orden del foco	A	!		3	1
2.4.4 - Propósito de los enlaces (en contexto)	A	X	13	23	
2.4.5 - Múltiples vías	AA	?			1
2.4.6 - Encabezados y etiquetas	AA	!		12	
2.4.7 - Foco visible	AA	?			1

Lo siguiente que verá, son los errores que tienen que ver con cómo está mostrada la información y si es comprensible para el usuario:

Comprensible

La información y el manejo de la interfaz de usuario debe ser comprensible.

Pauta	Nivel	Resultado	Problemas	Advertencias	No verificados
3.1-Legible			0	0	3
3.1.1 - Idioma de la página	A	✓			
3.1.2 - Idioma de las partes	AA	?			1
3.2-Predicible			0	0	1
3.2.1 - Al recibir el foco	A	?			1
3.2.2 - Al introducir datos	A	?			1
3.2.3 - Navegación consistente	AA	?			1
3.2.4 - Identificación consistente	AA	?			1
3.3-Introducción de datos asistida			1	6	0
3.3.1 - Identificación de errores	A	!		2	
3.3.2 - Etiquetas o instrucciones	A	X	1		
3.3.3 - Superencias ante errores	AA	!		1	
3.3.4 - Prevención de errores legales, financieros, datos	AA	!		3	

Por último, tendrá los errores robustos que son aquellos que pueden interferir con cómo se comportan los navegadores con la web:

Robusto						
Peñís	Nivel	Resultado	Problemas	Advertencias	No verificados	
4.1-Compatibile			34	112	1	
4.1.1 - Procesamiento	A	X	29			
4.1.2 - Nombre, función, valor	A	X	5			
No se han encontrado problemas		Existen problemas	Requiere revisión manual	Imposible realizar comprobación automática	novo aplicable	

Tomando todo lo que ha visto, vea a continuación un ejemplo práctico de ello. Si tiene la siguiente página que quiere analizar con la herramienta TAW. Vea el código de la página:

```
<!DOCTYPE html>

<html>

<head>

<title></title>

</head>

<body>

<h1>Título del contenido</h1>

<ol>

<li>Elemento 1</li>

<li>Elemento 2</li>

<li>Elemento 3</li>

</ol>



<p>Párrafo de prueba</p>

</body>

</html>
```

Al realizar el análisis con la herramienta, muestra que tiene dos problemas de accesibilidad. El primero es que en la web hay un título vacío y el segundo es que tiene una imagen que no ofrecer un texto en *alt*.

Por lo tanto, para resolver ambos problemas, debe escribir algo en la etiqueta <title> y añadir el atributo *alt* con su pequeño texto en la etiqueta .

Barra AIS

Esta barra de herramienta de Accesibilidad Web fue desarrollada por el equipo AIS, o lo que es lo mismo *Accessible Information Solutions* que formaba parte del NILS o *National Information and Library Service*. La barra AIS está traducida completamente al español por la fundación ONCE y puede encontrar información en la siguiente página: <https://tecnoadecuado.net/directorio/technosite>. Vea a continuación, cómo es esta barra:



Esta barra de herramientas de accesibilidad web ha sido desarrollada para facilitar el análisis manual de muchos elementos relativos a la accesibilidad en páginas web. Vea a continuación de qué funcionalidades dispone esta barra:

- Dispone con la información sobre la versión de la barra de herramientas y su desarrollo.
- Acceso a la web de información sobre cómo funciona la propia barra.
- Además, proporciona enlaces a referencias y otros recursos de información.

Es capaz de simular la experiencia de diferentes tipos de usuarios, debido a las siguientes características:

- **Daltonismo:** Al activar esta función, aparecerá una capa sobre la página que está visualizando, que modificará la paleta de colores que utiliza la web por defecto. Gracias a esta modificación, podrá visualizar cómo ven los usuarios que sufren de daltonismo su web. De esta manera, puede corregir los colores para aportar accesibilidad a la web.
- **Escala de grises:** Con esta función puede comprobar cómo se vería la web en caso de que un usuario utilizara una pantalla o navegador que solo pueda interpretar el blanco y negro.
- **Contraste reducido:** Esta función permite establecer una capa en la que podrá modificar la opacidad en un 30%, 60% y 90%. De esta manera puede comprobar cómo verían las personas mayores su web, ya que, suelen sufrir una reducción en la sensibilidad al contraste.
- Puede facilitar el uso de aplicaciones webs de terceros:
 - Verifica el código CSS de la web que esté manipulando. Para ello, usará el validador CSS que ofrece el W3C.
 - También, analiza el código HTML de la web que esté manipulando. Para ello, utilizará el validador HTML que ofrece el W3C.
 - Por último, envía la dirección url de la página que esté manipulando y la insertará en la herramienta de verificación de accesibilidad TAW, que ya ha visto previamente
- También, sirve para identificar los elementos que componen una página web:

- Muestra elementos de tabla como las etiquetas <table>, <th> y <td> que haya en la página web.
- También, muestra los elementos las listas tanto ordenados como las no ordenadas, las cuales equivalen a las etiquetas y respectivamente.
- Por último, muestra los elementos de encabezados de la etiqueta <h1>, hasta la <h6> que pudiera haber en la web.

Validador de W3C

El W3C proporciona a los usuarios en su web unas herramientas que permiten validar código CSS y HTML que compone una página web, de manera que cumpla con los estándares. Realmente, este tipo de validación que ofrece esta herramienta, no tiene nada que ver con el tema de la accesibilidad. Sin embargo, para que una web sea accesible se exige que su código HTML y CSS sea correcto y, por lo tanto, válido. Para poder utilizar estas herramientas, debe dirigirse a estas dos páginas web:

- Si quiere utilizar el validador de HTML, se dirigirá a <https://validator.w3.org>.
- En caso de que quiera utilizar el validador de CSS, se dirigirá a <https://jigsaw.w3.org/css-validator>.

	<p><i>La página del W3C también ofrece una herramienta para validar la accesibilidad de sus páginas web. Para utilizar esta herramienta, debe dirigirse a la página siguiente: https://www.w3.org/WAI/ER/tools/complete.</i></p>
<p>TOME NOTA</p>	

3.1.5.2. Utilización de los recursos en las páginas web

La accesibilidad web es la característica que hace que el mayor número de usuarios posible, indiferentemente si tienen discapacidad o no, puedan acceder a un servicio o producto web. Debe tener en cuenta que hay un conjunto de limitaciones a la hora de afrontar el diseño de una web para facilitar el acceso a la mayor cantidad posible de usuario.

Las limitaciones de las que se está hablando pueden ser debidas a las discapacidades propias del usuario o a las circunstancias con las que se encuentra el acceso a la web, que son las que pueden causar que un usuario sin discapacidad se encuentre con problemas de acceso a diversos contenidos, de la misma manera que podría encontrarse un usuario discapacitado.

Vea a continuación las principales discapacidades que debe tener en cuenta en la accesibilidad web:

- **Discapacidades cognitivas y del lenguaje:** Problemas en la lectura, problemas de percepción y en la memoria de corto alcance, entre otros.
- **Discapacidades auditivas:** sordera parcial o total.

- **Discapacidades motrices:** Son las discapacidades que impiden que el usuario pueda interactuar con el sistema utilizando los dispositivos más comunes como teclado y ratón.
- **Discapacidades visuales:** Ceguera, problemas con la visualización de ciertos colores, visión reducida.

Si tiene en cuenta este tipo de discapacidades se va a diseñar una mejor accesibilidad a su web para aquellos usuarios que no se encuentran en la situación ideal.

Como ya se visto anteriormente, una de las instituciones que trabajan por la accesibilidad de los usuarios en la red es el W3C o *World Wide Web Consortium*. Esta organización es internacional y orienta y estructura el desarrollo global de Internet o *World Wide Web*. El objetivo de esta organización es lograr todo el potencial de Internet mediante el desarrollo de protocolos comunes que promuevan su evolución. A esto es lo que se denomina **estándares**.

Esta organización es la que creó en 1998 la Iniciativa de Accesibilidad Web o WAI, que ya ha visto anteriormente. Esta iniciativa es la que promueve la accesibilidad de la web junto con otras instituciones.

Sin embargo, la institución que se dedica a esto mismo, pero en español se denomina Seminario de Iniciativas en Discapacidades y Accesibilidad a la Red o SIDAR. Vea aquí su [web](#):

The screenshot shows the homepage of the Fundación SIDAR - Acceso Universal website. The header includes the SIDAR logo and navigation links for 'Principal', 'Presentación', 'Fundación Sidar - Acceso Universal', and 'Seminario SIDAR'. The main content area has a sidebar with links for 'El Seminario', 'La Fundación', 'Las Jornadas', 'Elegir estilo', and 'IDonar ahora!'. The main content includes sections for 'Accesibilidad, adaptabilidad, usabilidad = Fundación Sidar - Accesibilidad Universal', 'Últimas noticias sobre accesibilidad, usabilidad y web semántica' (with a link to 'Accesibilidad Hoy'), and 'iRevise la accesibilidad y usabilidad de sus páginas Web!' (with a note about WCAG 2.0). A sidebar on the right is titled 'Suscríbete a ACCESOWEB' and provides information about the Yahoo! Groups mailing list.

3.1.5.3. Comprobar la accesibilidad en las páginas web

A continuación, va a conocer cuáles son las herramientas de accesibilidad más importantes que puede encontrar en Internet. Sin embargo, antes es importante que aprenda a diferenciar los aspectos de accesibilidad que se revisan de manera manual y lo que se revisan de manera automática.

- **Revisión manual.** Este tipo de revisión trata de verificar el funcionamiento de una página web cuando ocurren unas determinadas circunstancias. Por ejemplo, visualización y manejo de páginas web con diferentes navegadores o el uso de ayudas técnicas que sean específicas para ciertas discapacidades, entre otras.
- **Revisión automática.** Esta revisión trata sobre analizar el código que compone una página web, de manera que devuelva un informe sobre todos los errores de usabilidad que haya encontrado la herramienta en la página web. Por ejemplo, una herramienta para realizar esta tarea puede ser TAW, que según sus siglas es un Test de Accesibilidad Web.

Para realizar bien un test de accesibilidad, además de contar con estos dos tipos de revisiones y con las herramientas especializadas para ello, sería ideal contar con usuarios como, por ejemplo, discapacitados o ancianos, que participaran en esta revisión. De esta manera, sería mucho más conscientes de las dificultades con las que se encontrarían los usuarios reales y, así, podría mejorar la accesibilidad de su web de manera mucho más eficaz.

	<p>Cuando se realiza una revisión automática, debe analizar tanto el código HTML, como el CSS, de esta manera consigue solucionar y evitar muchos errores de accesibilidad.</p>
TOME NOTA	

3.1.6. Diseño de sitios web usables

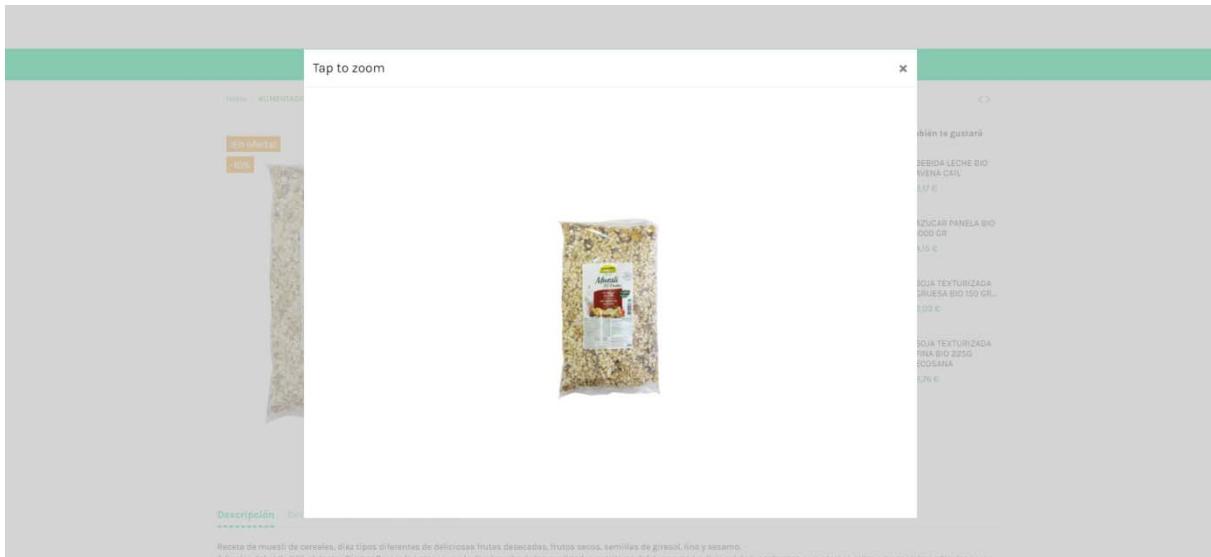
Cuando se desarrolle una página web, debe ponerle especial atención al diseño gráfico de la misma, ya que, este aspecto es muy valorado por los usuarios, aunque estos no sepan sobre diseño gráfico una web bien diseñada va a entrar mejor por los ojos de los usuarios y, por lo tanto, la valorarán más y mejor. Al desarrollar el diseño gráfico de una web, debe tener en cuenta no abusar de los efectos visuales, ya que, aunque quiera llamar la atención de los usuarios, si añade muchos efectos visuales conseguirá lo contrario y es cansar al usuario perdiendo, de este modo, su atención. Vea a continuación algunas recomendaciones para ayudarse con la inserción de imágenes y demás elementos multimedia en su página web:

Debe evitar incluir demasiados elementos gráficos en una web, ya que, podrían ralentizarla y la carga podría volverse excesivamente lenta.

Además, debes utilizar las imágenes únicamente en los casos en los que aporten alguna información importante relacionada con el contenido textual. De modo que estas imágenes estén presentadas correctamente para que sean fáciles de visualizar junto con ese contenido.

En caso de que inserte imágenes con una gran resolución, debe añadir también una miniatura de la misma y establecer, por ejemplo, un enlace que permita visualizarla en su tamaño real. Además, debe establecer un texto alternativo tipo *alt* y descriptivo tipo *title* que describan tanto la imagen

como la función que ha establecido en cada una de las imágenes que contenga la web. Vea un ejemplo de una imagen que se comporte de la manera descrita anteriormente:



Otra de las recomendaciones es que evite en todo lo posible el uso de elementos móviles que pueda provocar alguna distracción al usuario o que puedan confundir al usuario con los elementos del fondo de la web.

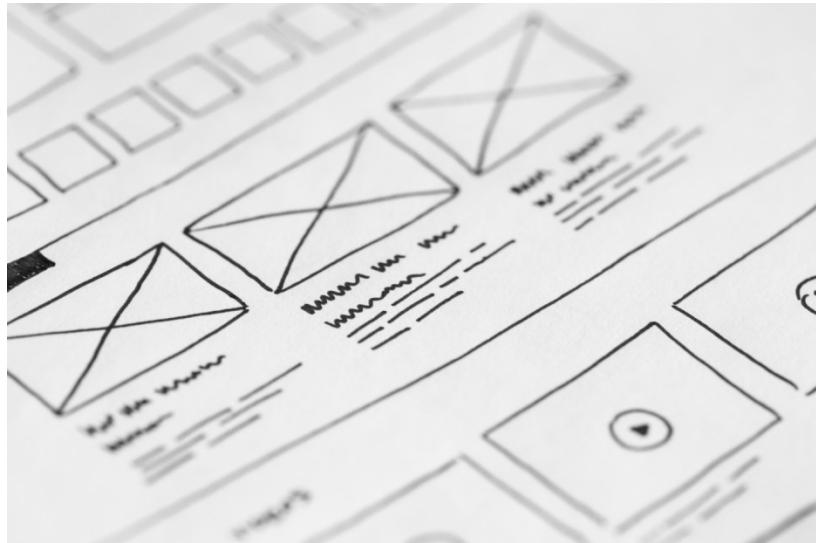
Otra de las cosas que debe evitar es establecer elementos multimedia que solo sirvan para mejorar el sitio web de manera tecnológica, puesto que, como se mencionó anteriormente, los elementos multimedia siempre se deben utilizar con algún propósito o finalidad específica. Además, en el caso en que se inserte este tipo de elementos debe asegurarse de que el usuario puede controlar completamente la reproducción de los elementos, es decir, que pueda activarlos, desactivarlos, retroceder, avanzar, etc.

Si en una página web se añade recursos multimedia que sean audios, debe asegurarse de que los usuarios son capaces de controlar el volumen y la evolución en el tiempo del propio audio, es decir, debe asegurarse de establecer una interfaz para el audio. Además, debe incluir un texto o subtítulos para cumplir con las recomendaciones de accesibilidad.

Recuerde que gracias al atributo *title*, puede añadir un texto explicativo que se visualizará cuando posicione el cursor del ratón encima del elemento al que añade este atributo. Además, también tiene el atributo *alt* que utiliza para que el navegador muestre un pequeño texto descriptivo en caso de que no se pueda cargar la imagen.

3.1.6.1. Descripción de sitios web usables

Para que una página web presente un grado considerable de usabilidad es importante fijarse en cómo están escritos los contenidos, y si tienen suficiente legibilidad. Todo esto debe estar presente junto con la estructura y la arquitectura de la información en un sitio web.



Escritura del sitio web

La lectura de información mediante una pantalla es mucho más incómoda y afecta más a la visión que si lee la misma información en papel. Es por ello que los usuarios no suelen leer la totalidad del contenido expuesto en una página web, sino que solo leen lo que estaban buscando.

Este es el motivo por el que debe adaptar el contenido de una web al medio en el que va a ser leído. La razón es que su objetivo siempre sea captar la atención de los usuarios de su web.

Vea a continuación un conjunto de consejos que están relacionados con la correcta escritura de los contenidos de la web:

- Los contenidos que constituyen la información de su web deben ser concisos, breves y sencillos de comprender. En caso de que necesite añadir textos extensos, debe dividirlos de la manera más adecuada posible y, además, deberán tener la estructura lógica más consistente y estructurada posible. También, es importante que añada enlaces de tipo Leer más donde permita al usuario ser redirigido a una versión más extensa del contenido.
- Por otro lado, es importante que evite añadir párrafos textuales demasiado extensos, ya que, puede afectar de manera negativa a la manera en la que se visualiza el documento. Otro punto a tener en cuenta es que, para muchos usuarios de Internet, el contenido que realmente les interesa es el que se muestra una vez la página haya sido cargada. Es por ello que este tipo de usuarios ignoran todo el contenido que hay debajo, es decir, el contenido que se muestra al desplazar la barra de desplazamiento hacia abajo. Vea una imagen donde aparece una barra de scroll:

The screenshot shows the homepage of the W3C España website. At the top, there's a header with the W3C logo and a search bar. Below the header, there are several navigation menus: 'ESTÁNDARES', 'PARTICIPAR', 'UNIRSE', and 'SOBRE EL W3C'. The main content area features a news article about a new online course on web accessibility. To the right, there's a sidebar with sections for 'VALIDADORES Y SOFTWARE' and 'TESTIMONIOS MIEMBROS W3C', which includes a testimonial from Fundación ONCE.

Otro consejo es ubicar la información más importante de la página al principio de la misma. De esta manera el usuario puede acceder a dicha información de manera rápida, y así ver si el contenido es o no de su interés. Para ello, utilizará títulos descriptivos que señalen de manera explícita la información que va a ser mostrada a continuación. Además, debe tener en mente cuáles de los títulos que añada son más importantes en comparación con los otros, ya que, a mayor tamaño más importancia tendrán.

Se recomienda destacar los textos y palabras que sean realmente importantes, pero teniendo en mente que no debe abusar del uso de esta recomendación. En definitiva, debería destacar solamente las palabras clave que pertenecen al contenido que va a mostrar en cuestión. De esta manera se enfoca de manera directa la atención del usuario.

Además, otra de las recomendaciones es que los títulos y textos de los encabezados deber ser directos y concisos, puesto que, para la mayoría de usuarios estos pequeños textos son concluyentes a la hora de que los usuarios determinen si les interesa el contenido que verán a continuación.

Conforme a los títulos de las páginas, se recomienda que sean los más descriptivos posibles, por lo que, lo ideal es que comiencen con el nombre de la empresa, marca, institución o algo similar, según a quién pertenezca la página.

Es de mera importancia centrarse mucho en la buena ortografía de la información que está exponiendo, ya que, si los usuarios se encuentran con errores ortográficos o gramáticos se sentirán inseguros con el propio sitio web y lo primero que pensarán es que se trata de un sitio web scam.

Por último, es muy importante constatar la auditoría del sitio web y de los propios contenidos del mismo, ya se trate de un sitio personal, empresarial o institucional. Lo importante es indicar lo más claramente posible quiénes son los autores del sitio y cómo se puede contactar con ellos.



TOME NOTA

Recuerde que el título o etiqueta <title> se insertará dentro de la etiqueta <head>, es en esta etiqueta donde se indica el título del documento web.

Lectura del sitio web

Es importante que mantenga una buena legibilidad de los contenidos del sitio web que esté diseñando. De esta manera se permite que el usuario pueda leer los contenidos de manera cómoda en su pantalla. Vea a continuación una lista con un conjunto de recomendaciones que ayudarán a realizar esta tarea para mantener la legibilidad en una página web:

- Uno de los principales consejos es que el usuario pueda cambiar el tamaño de letra en la web. Para ello se le proporcionará esta opción en la cabecera de la misma.
- También, debe cerciorarse de que el contraste del color de la letra con el color de fondo sea suficiente como para leer el contenido de manera cómoda. Nunca debe poner la letra negra con un fondo oscuro o un fondo con imágenes.
- Otra cosa que debe hacer es utilizar una tipografía adecuada para que el texto sea totalmente legible por pantalla. Por ejemplo, escoja una tipografía sans-serif. Vea una imagen donde aparecen los botones para cambiar el tamaño de letra en la propia web:

Cambiar el tamaño de letra dinámicamente

By Luis in Accesibilidad, css, Javascript, Tipografía

April 25, 2006



Leyendo esta entrada sobre el uso de botones para cambiar el tipo de letra de forma dinámica, se nos ha ocurrido contar cómo poder hacerlo.

Se puede hacer de varias formas: incrementando o decrementando el tamaño de letra según se pulse en los botones de aumentar o reducir el tipo de letra, o de forma fija, aumentando el tamaño de la letra y disminuyéndolo, pero solo a unos tamaños prefijados.

Para hacerlo de forma incremental, debemos creamos un elemento HTML con un *id*, para poder acceder a él y cambiarte los estilos directamente.

```
<p class="texto" id="contenido">...</p>
```

En el estilo creado deberemos indicar el tamaño de la letra usando porcentaje o medidas de tamaño de fuente (*em*), sobre todo por comodidad, en este caso usaremos porcentaje.

```
p.texto {  
    font-size: 100%;  
}
```

El código necesario para modificar el estilo sería algo así:

```
function tamano(mas) {  
    var signo = (mas) ? 1 : -1; // Para sumar o restar el porcentaje  
    // Obtenemos el objeto que contiene el texto
```

- Otra de las recomendaciones es que no utilice colores oscuros e imágenes como fondo de pantalla, puesto que, afecta muy negativamente a la legibilidad de los contenidos.

Además, no debe cambiar el estilo de las **familias tipográficas** que tiene ya en la web si, al cambiarlas, se visualizar peor el texto. Debe tener en cuenta las siguientes recomendaciones:

- No debe tener una separación entre letras demasiado excesiva, es por ello que es preferible utilizar la alineación de texto hacia la izquierda que de manera justificada.
- No debe abusar de los textos en negrita o cursiva.
- Evitar el uso de marquesinas o de textos parpadeantes.
- No debe escribir párrafos enteros en mayúsculas o versalitas.
- Lo aconsejable es insertar interlineados que ofrezcan ligereza a los párrafos. Además, debe poner especial atención en que los párrafos no sean demasiado extensos para que los usuarios no vean que la información mostrada en la página está sobrecargada.
- Debe tener presente que el contenido que ofrezca en su web sea claro y estructurado, además de ser sencillo para que todos los usuarios puedan comprender lo que están leyendo. Por esta razón, no debe utilizar tecnicismos ni palabras que puedan resultar complejas para la mayoría de usuarios. En definitiva, debe utilizar un lenguaje directo, claro y que sea de fácil comprensión.
- Por último, los contenidos relacionados o que redirijan al usuario hacia otra página web, solo debe utilizarlos cuando sea estrictamente necesario. Debe evitar añadir contenidos irrelevantes que lo único que aporten sea recargar la web sin aportar nada útil a esta.

Vea a continuación un ejemplo de web mal estructurada. Esta web se supone que es una tienda online, pero como podrá observar no cumple ninguna de las recomendaciones que ha aprendido en este epígrafe y, como es normal, cualquier usuario que entre en esta web se marchará a los pocos segundos:



Como puede observar, los contenidos de esta web no están estructurados, el menú casi ni puede diferenciarse del resto. No puede ver el logo de la web de manera sencilla, tampoco puede ver cómo

contactar con el propietario de la misma, ya que, como verá a continuación no hay ningún tipo de información en el pie de página, sino que la información de contacto está totalmente rodeada por los demás elementos de manera caótica:



Este es un ejemplo de página que no debe realizar. Los contenidos son confusos y cualquier usuario de Internet no sabría ni por dónde empezar a visualizar esta web.

Viendo este ejemplo puede saber lo importante que es seguir las recomendaciones para diseñar una página web profesional.

3.1.6.2. Estudio de la estructura y diseño de los sitios web usables

Como ya sabe, cuando diseña una página web usable debe crear un diseño dirigido a la sencillez, de esta manera evitará las confusiones y distracciones que les pueda surgir a los usuarios. Para realizar esta tarea, debe contar con una organización de las estructuras de la información que tengan las siguientes características:

- Diseñar la navegabilidad de la web de manera que ofrezca contenido que no den lugar a confusión, desde su inicio hasta su final. No debe utilizar tecnicismos ni palabras complejas cuando quiera ayudar al usuario a navegar por el sitio web.
- Diseñar la estructura de la página de manera que la estructura concuerde con los elementos y contenidos de la propia página.
- Ubicar las imágenes, textos y otros elementos, de manera adecuada para que el entorno que ofrece en su web sea lo más homogéneo posible. Todo ello hará posible que el mensaje se comunique de manera efectiva y correcta y con la finalidad de ayudar al usuario a que se haga una idea de la estructura general del propio sitio web.

Vea a continuación un conjunto de consejos sobre la usabilidad que están relacionados con el diseño conceptual de las páginas web:

- Una página web debe estar optimizada tanto de manera técnica, como desde el punto de vista de la interactividad, puesto que, el usuario debe tardar lo mínimo posible en encontrar

lo que busca. De esta manera, el usuario será capaz de reconocer de manera inmediata los elementos pertinentes para poder navegar por la web sin problema.

- Además, debe mantener el mismo diseño en todas las páginas que sean parte del mismo sitio web, el diseño de todas estas páginas deberá ser homogéneo y cumplir los mismos requisitos durante toda la navegación del mismo sitio web. Sin embargo, la página de inicio del sitio web es la única que sí puede tener un diseño un poco diferente al de las demás, puesto que, es la página de entrada principal del sitio web y se debe mostrar una visión general del propio sitio al usuario. De esta manera el usuario puede hacer una idea de qué es lo que puede encontrar en el sitio web.
- Debe tener presente que, en una web, además de organizar la información de manera estructurada para que los usuarios puedan encontrar lo que buscan en el mínimo tiempo posible, debe añadir un sistema de búsqueda alternativo para que el usuario pueda encontrar lo que busca sin necesidad de varios clics en la web. Es decir, debe añadir un motor de búsqueda capaz de mostrar resultados relacionados con las palabras que el usuario introduce en la barra de búsqueda.

Lo más común es encontrar en la web un campo de búsqueda que se comportará de la misma manera que los buscadores de los navegadores. En este campo, se introducirá el o los términos a buscar y llevará a los resultados. Este campo de búsqueda se suele encontrar en la cabecera de las páginas web, que es donde se encontrará también el menú de navegación.

En esta sección encontrará los enlaces de los menús que se redirigirán a una página del propio sitio web o a una página externa. Vea un ejemplo del cuadro de búsqueda en una web:

The screenshot shows the homepage of naturaleeasy.es. At the top, there is a green header bar with social media icons (Facebook, Twitter) and a 'Lista de deseos (0)' button. Below the header is the website's logo, which features a stylized green leaf or brain-like shape above the text 'naturaleeasy.es'. To the right of the logo is a search bar containing the placeholder 'Busca entre más de 2000 productos' and a magnifying glass icon. Further to the right are a 'Carrito / Vacío' button and an 'Iniciar sesión' button. At the bottom of the page is a green footer navigation bar with categories: Alimentación, Complementos, Infusiones, Cosmética, Salud e higiene, and Hogar.

- Otra de las recomendaciones es que la estructura visual de la página, también, debe ser homogénea, es decir, debe ser constante. Esto quiere decir que la localización de los elementos tales como menús, barras, etc. Deben estar en la misma posición en todas las páginas. Además, es importante que la estructura sea consistente e inmóvil para que el usuario no se desoriente y se confunda con los elementos de la página. Otra de las cosas a las que debe prestarle atención es al uso de iconos y estilos textuales, los cuales deben ser iguales cuando quiera representar lo mismo dentro de la propia web. Por ejemplo, si el ícono que debe pulsar para que redirija a la página de inicio es una casita, se deberá mantener este mismo ícono para todas las páginas que tenga el sitio web.

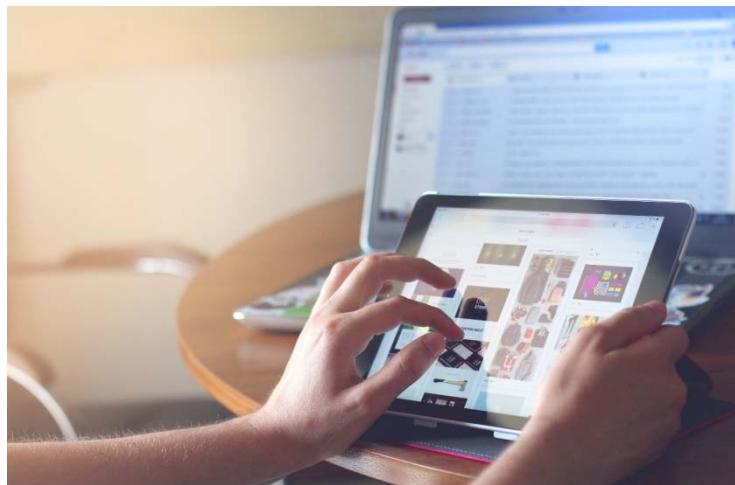
- Por otro lado, es importante que limite la cantidad de clics que debe hacer el usuario para llegar a la información que quiere visualizar. Este tema es muy importante porque cuantos más clics haga el usuario querrá decir que más compleja es la estructura de su web y, por lo tanto, más difícil será que el usuario llegue a su destino.
- Algo fundamental del diseño web es que debe diseñar una página web teniendo en mente que la web sea capaz de adaptarse a diferentes resoluciones y tamaños de pantalla. Para ello, lo que se recomienda es hacer diseños líquidos para todos los elementos que van a ser parte de la página web. Esto hará que los elementos se adapten al área de visualización de la web. A este tipo de diseño también se le llama **diseño responsive**.
- En cuanto al logotipo de una web, debe destacar bastante para que el usuario lo identifique como lo que es. Si el logotipo está acompañado de algún texto, también deberá leerse cómoda y fácilmente. La ubicación correcta de un logotipo en una web es la parte superior de la página y, si puede ser, alineado a la izquierda. Lo recomendable es que el logotipo aparezca en todas las páginas del sitio web.
- Por último, debe comprender que la jerarquía visual del sitio web debe ser clara e idéntica en todas las páginas que componen el sitio web. Para ello, se dividirán las diferentes secciones de una manera constante y definida, como ya se comentó anteriormente. Además, se evitará la superposición de los elementos que se muestren en un mismo nivel. Es importante que utilice mecanismos que sean capaces de identificar los diversos elementos de la web, de manera que le quede claro al usuario con qué elementos puede interactuar.

3.1.7. Adaptación de sitios web usables

En este epígrafe se dan consejos básicos a seguir cuando se esté diseñando y desarrollando una página web. Como ya se ha visto, la usabilidad web puede definirla como un conjunto de recomendaciones que debe seguir a la hora de desarrollar un sitio web. Y que estas recomendaciones están basadas en la facilidad de acceso a los contenidos por parte de los usuarios.

Utilización de los sitios web usables

La manera más típica en la que los usuarios interactúan con una página web es mediante los enlaces de esta. Es por ello que los enlaces que inserta en una web deben estar insertados en los elementos que sean más importantes y debe hacerlo de una manera lo más claramente posible, ya que, si no indica que un elemento contiene un enlace podría provocar errores de interpretación por parte del usuario. Además, es importante señalar de manera explícita cuál es el destino de ese enlace donde va a ser redirigido el usuario, de esta manera el usuario no sentirá inseguridad a la hora de pulsar en el enlace. De otro modo, podría crearle inseguridad al usuario y podría pensar que se trata de un contenido scam.



Vea a continuación algunas de las recomendaciones que puede llevar a cabo para insertar enlaces en su web:

- En caso de que coloque enlaces en un párrafo, debería posicionarlos al final del mismo. De esta manera se evitará que se interrumpe el flujo de lectura del propio texto. Además, es importante que no incluya demasiados enlaces para evitar que los usuarios se confundan y desorienten con otros contenidos sin que hayan terminado de visualizar el contenido principal.
- Otra recomendación es que utilice enlaces que redireccionan a otras páginas en las que haya información que sea relativa a la información que está dando en su web. De esta manera si el usuario no encuentra lo que busca en su web, puede encontrarlo en alguna otra que haya enlazado.
- Otra cosa que debe evitar es el uso de enlaces “Haz clic aquí”, ya que, no todos los usuarios navegan con ratón, por lo que, no podrían clicar en ese enlace y, además, no aporta ninguna semántica a la web.
- Además, debe asegurarse de que los enlaces tengan asociada una pequeña descripción sobre el lugar al que va a redirigir y la acción que va a desencadenar el mismo. Por ejemplo, si al pulsar el enlace se va a abrir la nueva página en una nueva pestaña. Para ello, se utiliza el atributo *title* donde escribir un texto emergente que aparecerá cada vez que se posicione el puntero del ratón encima del elemento.
- En caso de que un enlace esté apuntando hacia un archivo que no sea una página web, es muy importante que le indique al usuario qué tipo de archivo va a abrir o va a descargar. Para ello, lo mejor es que utilice el atributo *title*. Además, también se recomienda indicar al usuario información sobre el tamaño del archivo, y en caso de que sean archivos de audio o vídeo, sería aconsejable indicar la duración de los mismos.

Vea a continuación el código de cómo insertaría un archivo descargable:

```
<!DOCTYPE html>

<html>

<head>

<title></title>

</head>

<body>

<h1>Descargar Archivo</h1>

<a href="https://rafarjonilla.com/contenido/swiper-no-
robes.png" download="Swiper, el zorro ladrón" title="Descargar una imagen .png">Haz clic aquí para
descargar el archivo</a>
```

</body>

</html>

Y ahora vea el resultado:

Descargar Archivo

[Haz clic aquí para descargar el archivo](#)

- Otra de las recomendaciones que puede llevar a cabo es evitar el uso de enlaces que estén creados mediante JavaScript, puesto que, no todos los usuarios tienen activado JavaScript y por lo tanto no les funcionaría el enlace. En caso de que no pueda obviar los enlaces creados con JavaScript, debe cerciorarse de que estos enlaces podrán funcionar en los navegadores en los que JavaScript esté desactivado. Debe recordar que una de las formas para ejecutar funciones en JavaScript es incluyendo el nombre de dicha función en el atributo *href* del enlace pertinente, escribiendo con anterioridad a la función JavaScript: dentro del valor del atributo.
- Al principio de la aparición de los primeros navegadores de tipo gráfico, los enlaces siempre han sido inmutables y han sido los típicos enlaces de color azul y subrayados para los enlaces no visitados, y para los enlaces que ya han sido visitado se ha utilizado siempre el color morado. Es por ello que actualmente cualquier usuario que vea un elemento de este tipo lo entiende como un enlace de manera inconsciente. Sin embargo, hoy por hoy puede crear los enlaces con los colores que quiera. En cambio, sí debe subrayarlos y destacarlos del resto de elementos de la página con la finalidad de que el usuario lo entienda como tal. Además, es importante que establezca un color para los enlaces que ya han sido visitados y los que no hayan sido visitados.
- Por último, al haber múltiples páginas que cambian de manera continua de dirección url, es importante que verifique de manera periódica que los enlaces siguen apuntando a los documentos originales.

4. HERRAMIENTAS DE EDICIÓN WEB

En este tema va a aprender sobre las herramientas de edición web. Aprenderá cómo descargarlas e instalarlas paso a paso y cómo configurarlas. Algunas de las herramientas con las que va a trabajar son Atom, Brackets, Sublime, entre otras.

Como ya verá más adelante, este tipo de herramientas va a ayudar muchísimo a la hora de desarrollar una web; ya que son herramientas especializadas en el desarrollo web y tienen configuraciones que ayudan a ahorrar tiempo y evitar errores de código.

Además, va a ver de cada herramienta sus funciones y características más importantes, para saber qué herramienta se va a amoldar más a su forma de trabajar. Cada diseñador web tiene sus preferencias en cuanto a herramientas de desarrollo.

Por otra parte, va a aprender a utilizar y encontrar los elementos que ofrecen estas herramientas. Aprenderá todas las propiedades de los elementos que la conforman y cómo utilizarlo para realizar las tareas más sencilla y rápidamente posibles.

Por último, se verá cómo codificar lenguaje HTML con este tipo de herramientas, de modo que rentabilice el tiempo de trabajo en cuanto al manejo de errores y la rapidez del mismo.

4.1. Instalación y configuración de herramientas de edición web

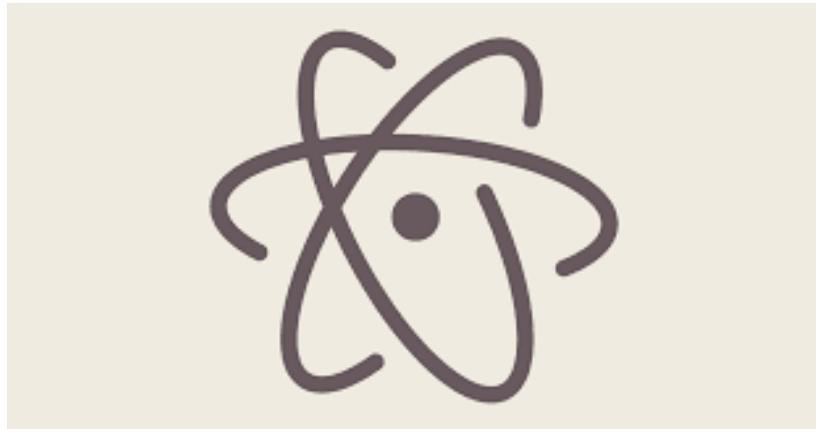
El uso de editores de texto plano para la creación de páginas web puede llegar a ser algo complejo. Las páginas web no solo utilizan código HTML, sino que, también, requieren de la integración de otros lenguajes como JavaScript para aportar funcionalidad, además de, hojas de estilo CSS para darle formato, y otros lenguajes de servidor como PHP, ASP.net o Perl para que la página web llegue al usuario a través de Internet.

Para este diseño y desarrollo de la página web se utilizan los, anteriormente denominados, programas de edición web. En estos programas se aúnan todas las herramientas para el desarrollo e integración de la web pudiendo ser utilizados los diferentes lenguajes de programación y gracias a su interfaz GUI (interfaz gráfica de usuario) el trabajo es mucho más sencillo e interactivo que en un editor de texto convencional.

Existen multitud de aplicaciones, unas más sencillas u otras más complejas, pudiendo ser gratuitas o de pago y otras aplicaciones enfocadas a un tipo de proyecto u otro. Por ello, puede elegir la que más se ajuste a sus propósitos. Estas son las características de algunos de los editores más conocidos.

4.1.1. Atom

Es un editor de código abierto, es decir, que su arquitectura es modificada y evoluciona gracias a la colaboración de programadores y usuarios de la comunidad de GitHub. Este editor puede soportar múltiples códigos de programación como Node.js, C++, JSON, SQL, XML, Less, JAVA, JavaScript, etc.



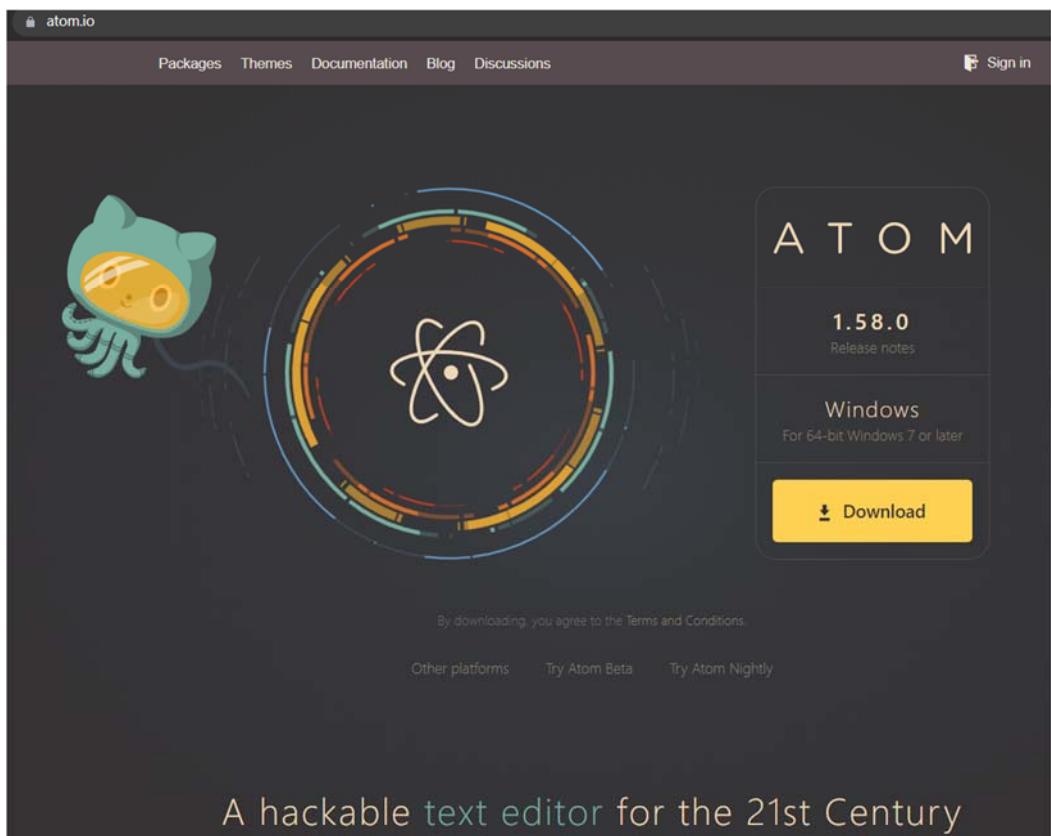
Icono de Atom

Atom es una potente herramienta. Además de ser un editor de código HTML, Atom es uno de los denominados Frameworks. Los frameworks son “espacios de trabajo” en los que se puede trabajar con diferentes lenguajes de programación a la vez sin requerir de otros softwares. Entre sus muchas ventajas, este editor también se puede usar como entorno de desarrollo integrado (IDE), esto quiere decir que aúna las herramientas necesarias para escribir, generar, probar y depurar un programa.

Es, además, un editor gratuito, por lo que no requiere licencias de pago. Vea a continuación cómo instalar este potente editor.

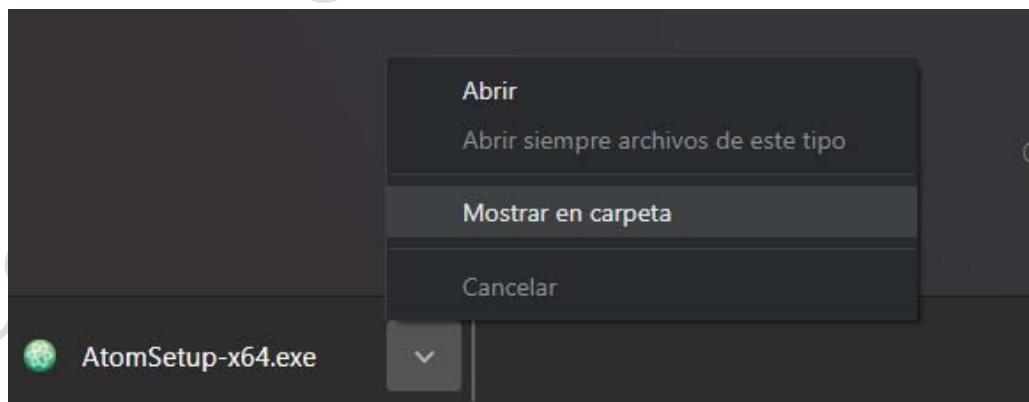
Paso 1: Descargar Atom.

Primero debe descargar el editor desde su navegador a través de su página oficial [Atom](#).



Página oficial de Atom

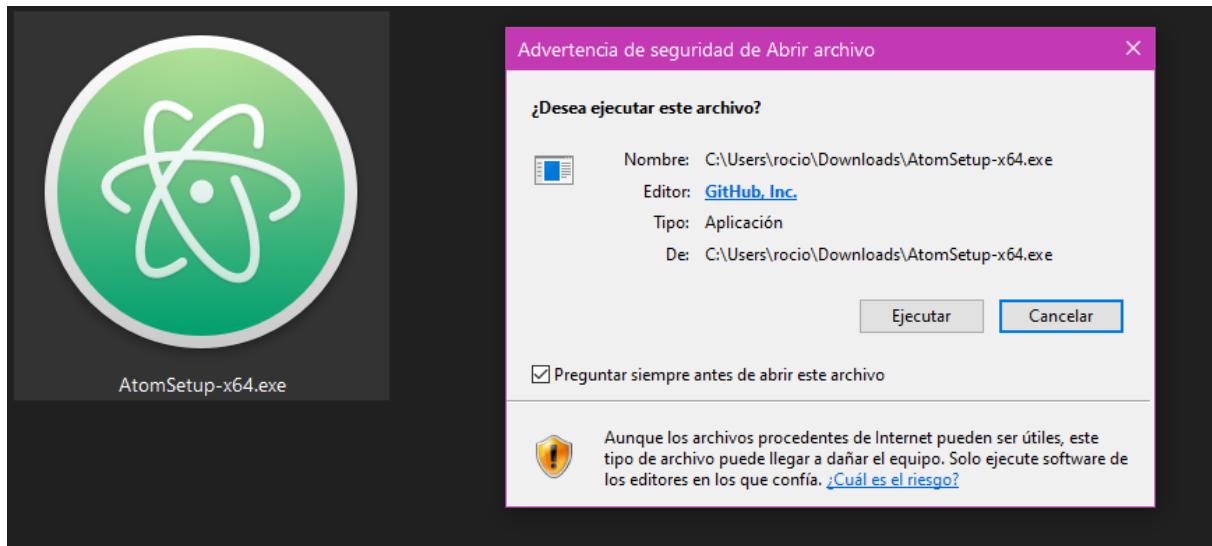
Al comenzar la descarga, el archivo descargado se guardará donde haya indicado, según el navegador desde el que está trabajando (en este caso Google Chrome).



Abrir el archivo en Descargas

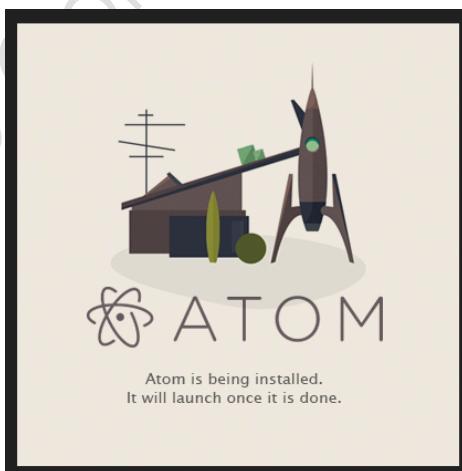
Paso 2: Instalar Atom.

Abra la ubicación del archivo descargado y haga doble clic sobre él. Al ser un archivo ejecutable notificará con una advertencia de los posibles daños al equipo, pero lo ha descargado desde su sitio oficial, por lo que es un archivo seguro.



Abrir archivo ejecutable

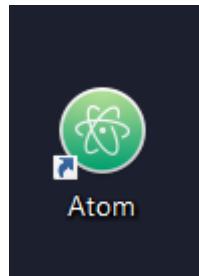
Continúe con el proceso siguiendo los pasos que indica el programa.



Instalador de Atom

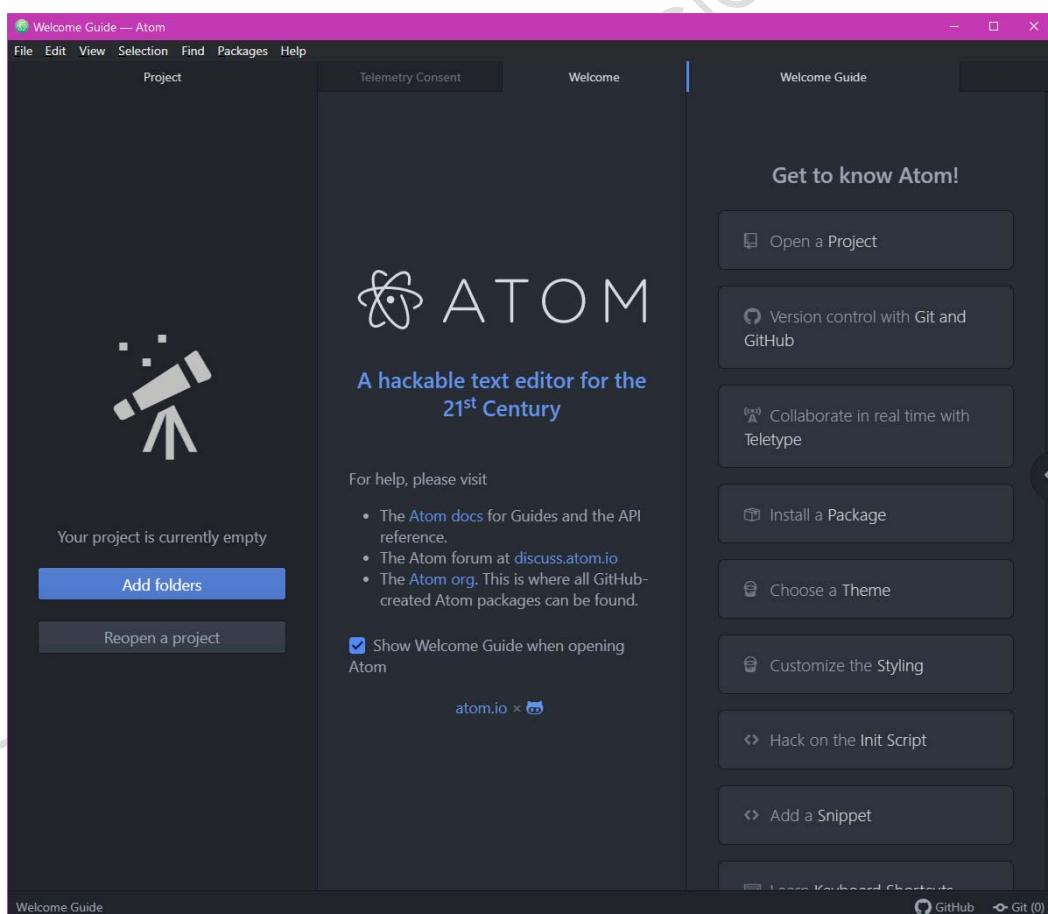
Paso 3: Ejecutar Atom.

Una vez instalado Atom en su equipo, aparecerá un ícono de acceso directo en su escritorio.



Icono acceso directo a Atom

Ya puede ejecutarlo y entrar en el editor. En la pantalla de Bienvenida puede encontrar los primeros consejos para comenzar a usar Atom, desde abrir un proyecto, conectar con GitHub y colaborar en proyectos, instalar paquetes, customizar la interfaz, añadir *snippets* y aprender los atajos de teclado. Con todo ello podrá comenzar a usar Atom.



Entorno de Atom



TOME NOTA

GitHub es una plataforma donde se desarrollan proyectos de herramientas y aplicaciones de código abierto. Cualquier usuario puede colaborar aportando su conocimiento para mejorar el código de los proyectos.

4.1.2. Brackets

Brackets fue diseñador por Adobe y es uno de los editores preferidos por desarrolladores web y programadores. Es un editor gratuito de código abierto, lo que hace que sea cada vez más potente gracias a sus continuas mejoras por parte de la comunidad de GitHub. Entre sus muchas características se puede destacar su bajo consumo de memoria RAM, lo que lo convierte en un editor ligero y agiliza la carga de la CPU. Otra de sus características es que soporta archivos .psd, es decir, de Adobe Photoshop, lo que lo convierte en una herramienta muy útil para crear sitios web rápidamente.



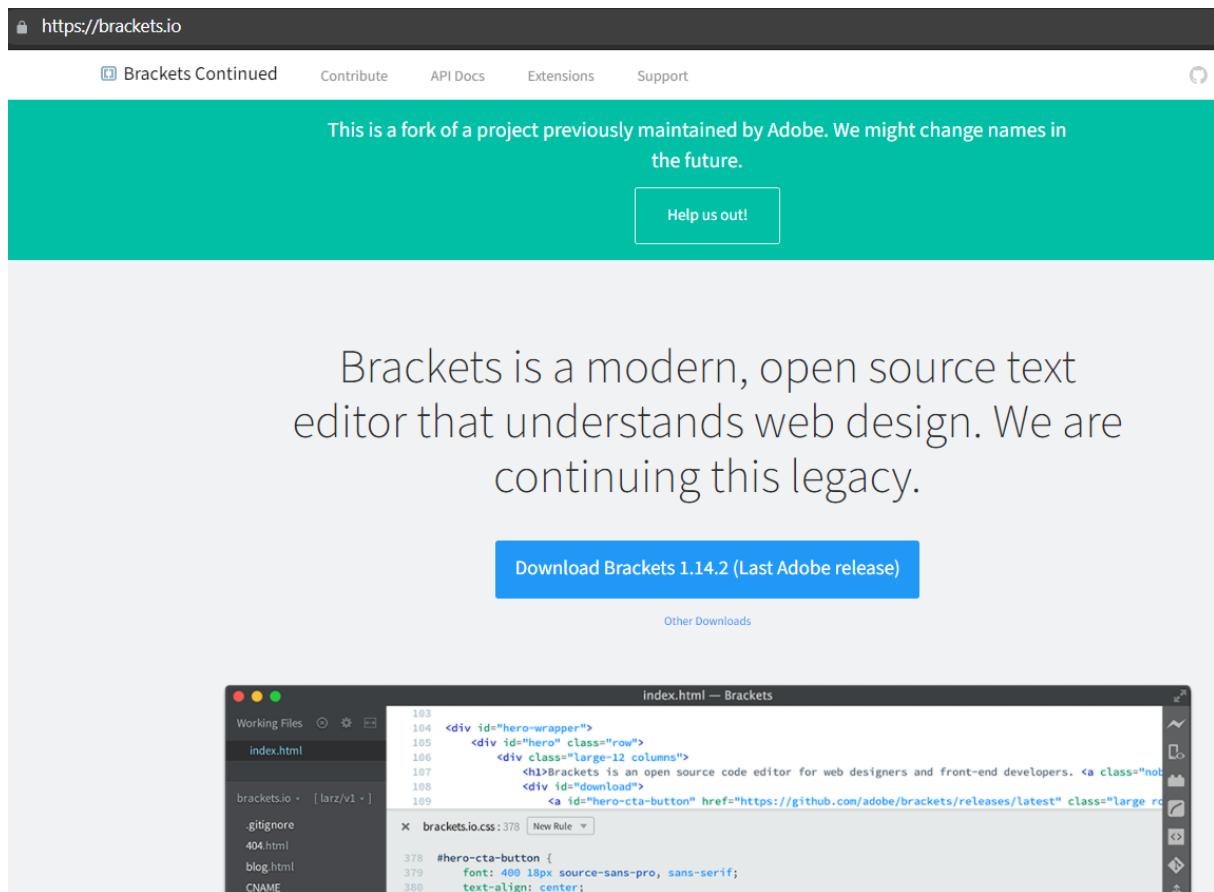
Icono de Brackets

Una de sus funcionalidades más interesantes es su “modo vista previa dinámica” la que permite ver los cambios realizados en el código a tiempo real en el navegador. También permite editar código CSS desde el código HTML, es decir, Brackets combina todo el trabajo en una sola ventana en vez de saltar de archivo en archivo. Esto confiere agilidad a la tarea y ahorra mucho tiempo de trabajo. Otra característica que permite un trabajo más eficiente es la visualización de imágenes y colores directamente desde el código.

La interfaz de este editor es moderna y minimalista, lo que permite enfocarse en el trabajo sin distracciones y con acceso rápido a las herramientas que necesite. Aun así, este editor permite una completa personalización gracias a sus extensiones lo que agilizará su trabajo. Vea a continuación cómo instalar este editor.

Paso 1: Descargar Brackets.

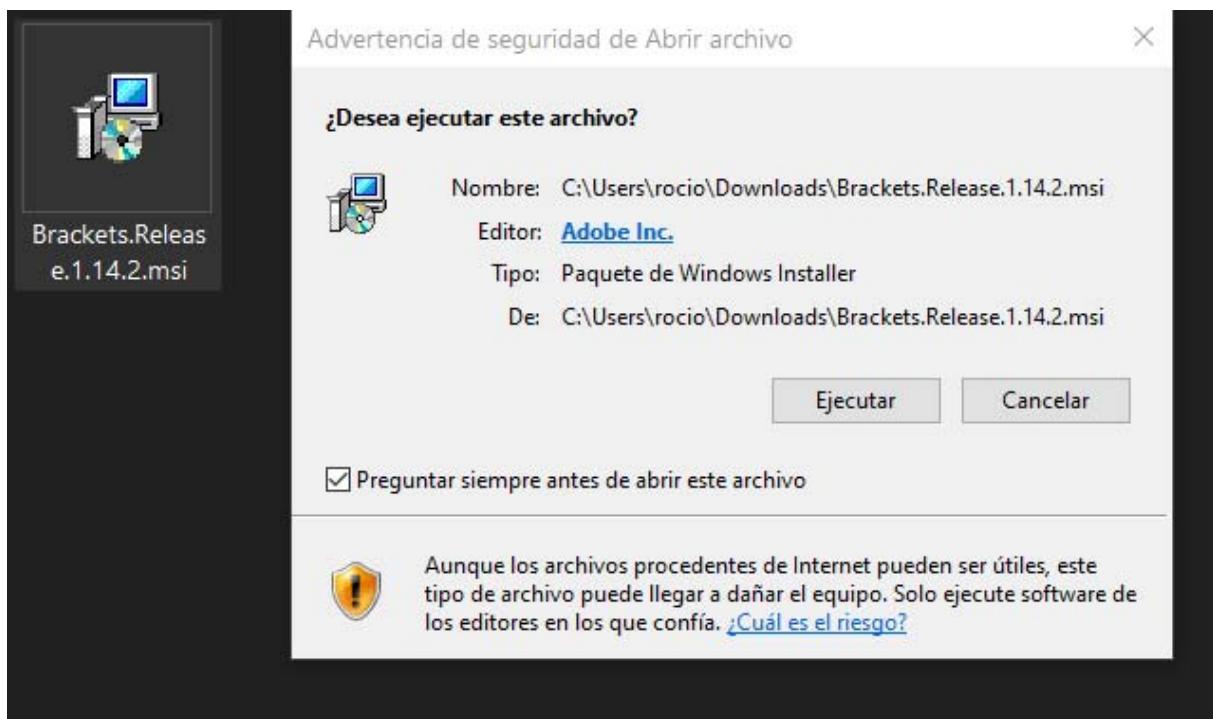
Primero debe descargar el editor desde su navegador a través de su página oficial [Brackets](#).



Página oficial de Brackets

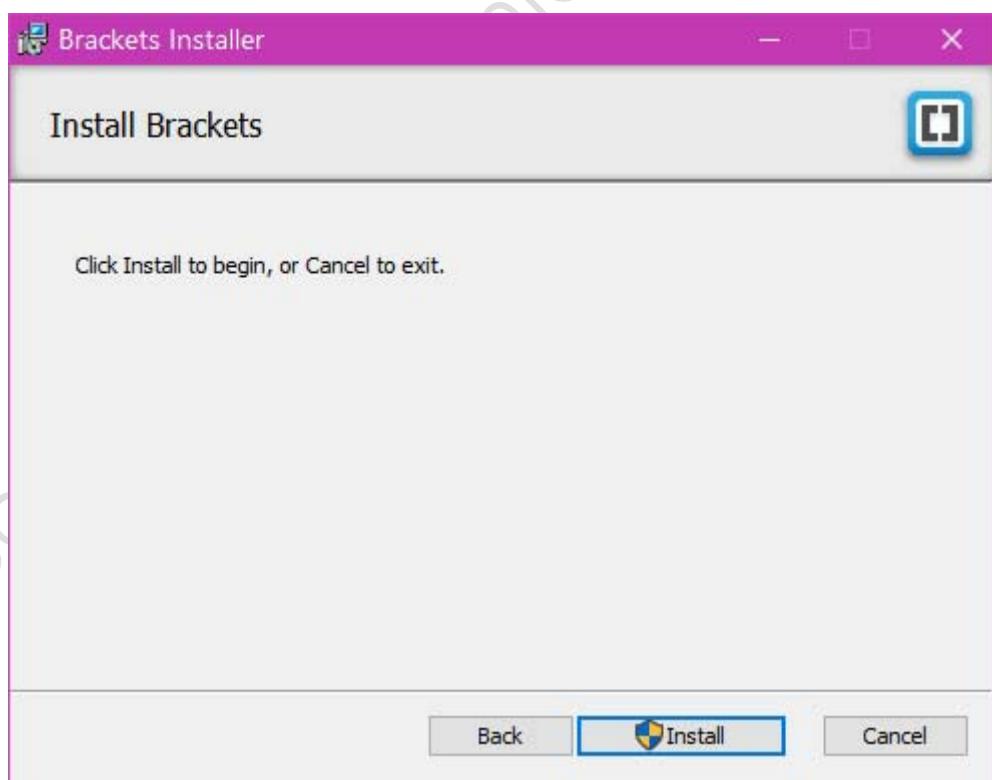
Paso 2: Instalar Brackets.

Abra la ubicación del archivo descargado y haga doble clic sobre él. Al ser un archivo ejecutable notificará con una advertencia de los posibles daños al equipo, pero lo ha descargado desde su sitio oficial, por lo que es un archivo seguro.



Abrir archivo ejecutable

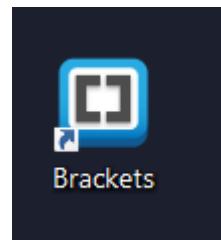
Continúe con el proceso siguiendo los pasos que indica el programa.



Instalador

Paso 3: Ejecutar Brackets.

Una vez instalado en su equipo, aparecerá un ícono de acceso directo en su escritorio.



Icono acceso directo a Brackets

Ya puede ejecutarlo y entrar en el editor. En la pantalla de Bienvenida puede encontrar un archivo HTML donde están escritos los primeros consejos para comenzar a usar Brackets.

A screenshot of the Brackets code editor interface. The title bar says "index.html (Primeros Pasos) - Brackets". The menu bar includes Archivo, Edición, Buscar, Ver, Navegación, Desarrollo, Ayuda. The left sidebar shows a file tree with "Primeros Pasos" expanded, showing "screenshots", "index.html", and "main.css". The main workspace displays the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <meta http-equiv="X-UA-Compatible" content="IE=edge">
7   <title>PRIMEROS PASOS CON BRACKETS</title>
8   <meta name="description" content="Una guía interactiva de primeros pasos para Brackets.">
9   <link rel="stylesheet" href="main.css">
10 </head>
11 <body>
12
13   <h1>PRIMEROS PASOS CON BRACKETS</h1>
14   <h2>¡Esta es tu guía!</h2>
15
16 <!--
17   HECHO CON <3 Y JAVASCRIPT
18 -->
19
20 <p>
21   Bienvenido a Brackets, un nuevo editor de código abierto que entiende el diseño web. Es un editor de
22   código liviano y potente al mismo tiempo que incluye herramientas visuales dentro del mismo para
23   que puedas obtener la ayuda que necesites cuando la necesites.
24 </p>
25
26 <!--
27   ¿QUÉ ES BRACKETS?
28 -->
29 <p>
30   <em>Brackets es un editor diferente.</em>
31   Brackets tiene varias características únicas como la Edición rápida y la Vista previa dinámica y muchas
32   más que no vas a encontrar en otros editores. Además, Brackets está escrito en JavaScript, HTML y CSS.
33   Esto significa que la mayoría de quienes usan Brackets tienen las habilidades necesarias para modificar y
34   extender el editor. De hecho, nosotros usamos Brackets todos los días para desarrollar Brackets. Para
35   saber más sobre cómo utilizar estas características únicas, continúa leyendo.
36 </p>
37
38 <!--
39   EMPIEZA CON TUS PROPIOS ARCHIVOS
40 -->
41
42 <h3>Proyectos en Brackets</h3>
43 <p>
44   Para poder editar tu propio código en Brackets, puedes simplemente abrir la carpeta que contiene los
45   archivos. Brackets considera a la carpeta abierta como el "proyecto"; características como las Sugerencias
46   de código, la Vista previa dinámica y la Edición rápida solo utilizan los archivos contenidos dentro de
47   la carpeta actualmente abierta.
48 </p>
49
50 <samp>
51   Una vez que estés listo para salir del proyecto de ejemplo y editar tu propio código, puedes usar el menú
52   despegable en la barra de la izquierda para cambiar de carpeta. En estos momentos, el menú despegable dice
53   "Primeros Pasos" - la cual es la carpeta que contiene el archivo que estás viendo en estos momentos. Haz
54   clic en el menú despegable y selecciona "Abrir carpeta..." para abrir tu carpeta.
55   También puedes usar el menú despegable para abrir las carpetas que abriste recientemente, incluyendo este
56   proyecto de ejemplo.
57 </samp>
```

The status bar at the bottom shows "Línea 1, Columna 1 — 213 líneas" and icons for INS, UTF-8, HTML, and Espacio.

Entorno de Brackets

4.1.3. Sublime Text

Este editor tomó importancia desde su origen como extensión de Vim hasta convertirse en un editor independiente. Aunque se puede descargar de forma gratuita no es un software libre y de código abierto. Será necesario adquirir una licencia de pago para acceder a todas las funcionalidades y actualizaciones, aun así, la versión de prueba no tiene caducidad.



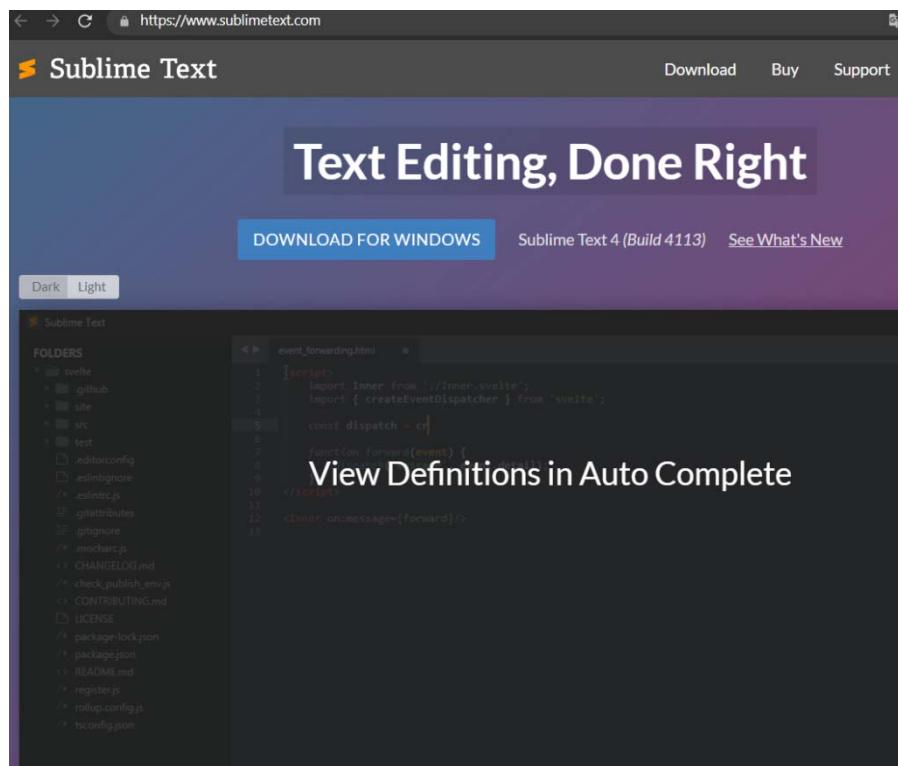
Icono de Sublime Text

Son algunas de las características de este editor, muy alabado entre desarrolladores:

- **Minimap:** es la previsualización en miniatura de todo el código sobre el que se está trabajando. Es muy útil para desplazarse rápidamente por la estructura del código.
- **Pestañas:** permite abrir diferentes archivos en diferentes pestañas para organizarlos y mover rápidamente de uno a otro.
- **Syntax Highlight configurable:** se puede configurar el remarcado de sintaxis para acceder rápidamente a los trozos de código que interesa.
- **Auto completado y marcado de llaves:** puede dirigir a la apertura y cierre de cada llave para identificar rápidamente el código que quiera trabajar.
- **Multi Layout:** se puede configurar diferentes divisiones dentro de una sola ventana para visualizar diferentes archivos a la vez.
- **Búsqueda dinámica:** se puede buscar expresiones, archivos o proyectos de forma simultánea.

Paso 1: Descargar Sublime Text.

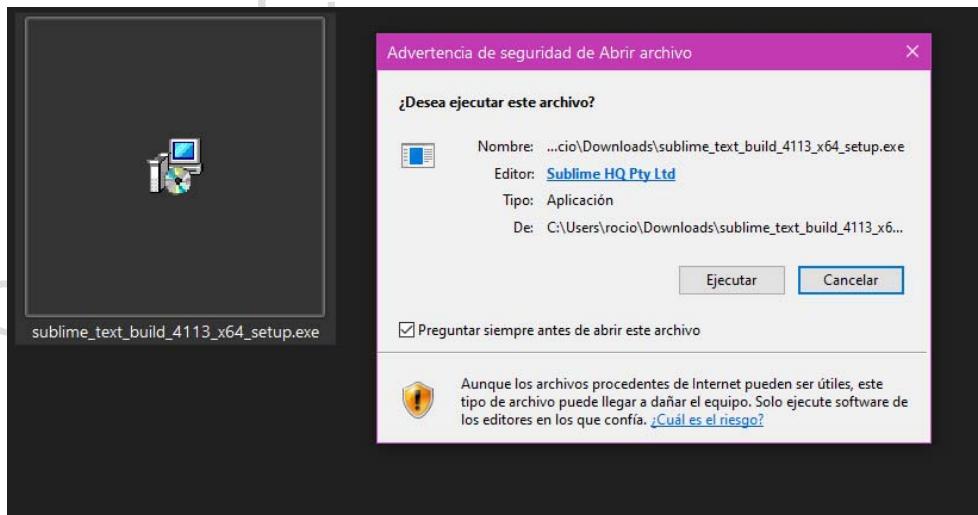
Primero debe descargarse el editor desde el navegador a través de su página oficial [Sublime Text](#).



Página web oficial de Sublime Text

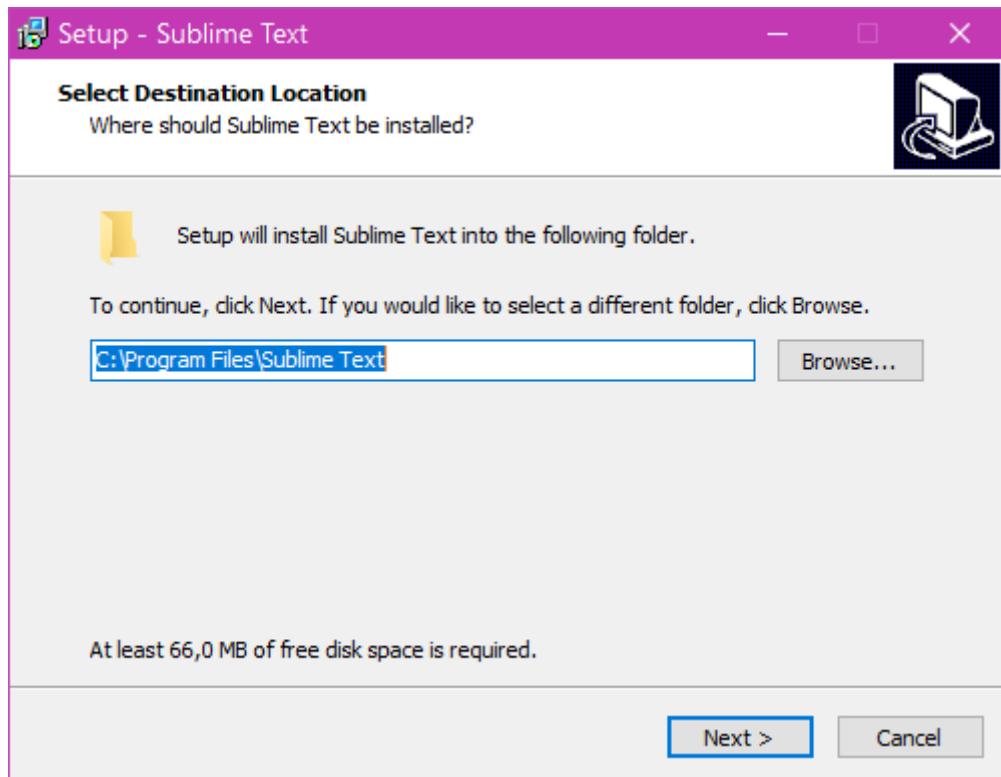
Paso 2: Instalar Sublime Text.

Abra la ubicación del archivo descargado y haga doble clic sobre él. Al ser un archivo ejecutable notificará con una advertencia de los posibles daños al equipo, pero lo ha descargado desde su sitio oficial, por lo que es un archivo seguro.



Abrir archivo ejecutable

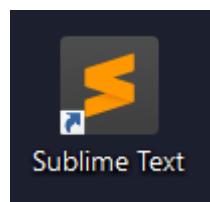
Continúe con el proceso siguiendo los pasos que indica el programa.



Instalador

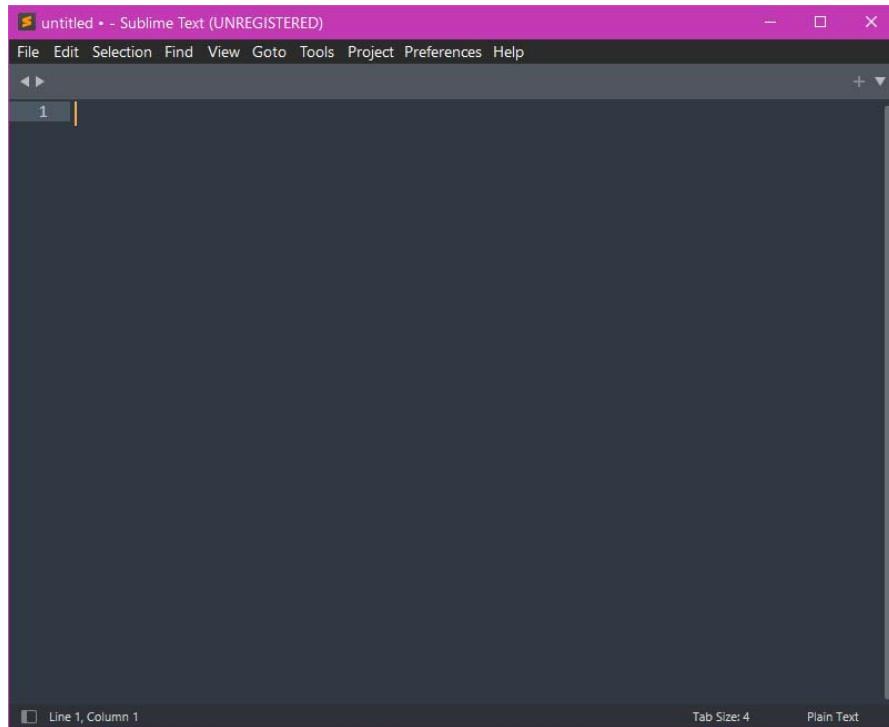
Paso 3: Ejecutar Sublime Text.

Una vez instalado en su equipo, aparecerá un ícono de acceso directo en el escritorio.



Icono acceso directo a Sublime Text

Ya puede ejecutarlo y entrar en el editor, aparece directamente un archivo para comenzar a editar. En la página web oficial puede encontrar una gran cantidad de [documentación](#) para explorar este gran editor que tiene infinitud de posibilidades.



Entorno de Sublime Text

4.1.4. Notepad++

Este es uno de los editores favoritos para los nostálgicos de la programación. Es un software de código abierto por lo que los usuarios lo desarrollan para diferentes plataformas como Linux, Unix y MacOS X. Su diseño es muy similar al Bloc de notas que viene por defecto en Windows. Es un editor de texto de código abierto que soporta varios lenguajes de programación. Su desarrollo en C++ hace que sea un editor rápido y veloz.

Estas son algunas de sus características:

- **Coloreado de código:** Esto ayuda a una lectura más rápida del código, ya que resalta la sintaxis y los cierres de las mismas. Además, reconoce más de 40 lenguajes de programación y sus diferentes sintaxis.
- **Pestañas:** Permite abrir varios documentos a la vez para moverse entre ellos.
- **Interfaz personalizable:** Este editor viene con 20 temas ya preconfigurados para elegir el que más guste.



Icono de Notepad++

Paso 1: Descargar Notepad++.

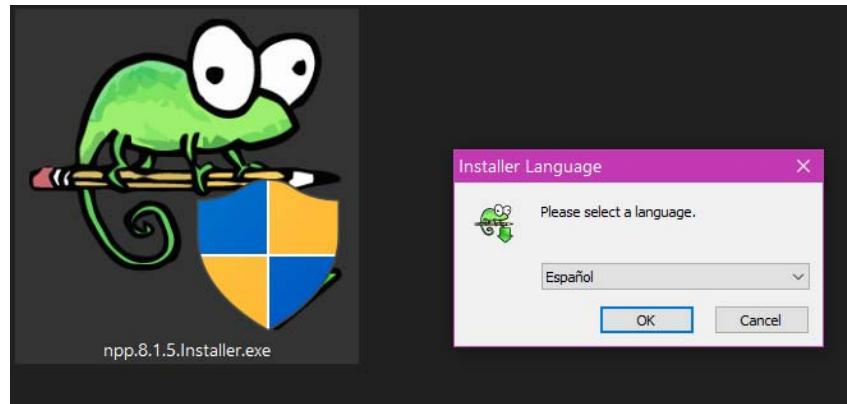
Primero debe descargar el editor desde su navegador a través de su página oficial [Notepad++](#). Hay multitud de descargables, pero siempre será más recomendable descargar la última, ya que tendrá muchos errores corregidos respecto a su versión predecesora.

The image consists of two parts. On the left is a screenshot of the Notepad++ official website. It shows the logo, a navigation menu with options like Home, Download, News, Online Help, Resources, RSS, Donate, and Author. Below the menu is a section for 'Ahal Develop' with a purple button labeled 'Start your free 30-day trial today'. On the right is a screenshot of the Notepad++ application window. The title bar says '*D:\source\notepad4ever.cpp - Notepad++'. The code editor contains the following C++ code:

```
1 #include <GPL.h>
2 #include <free_software.h>
3
4 void notepad4ever()
5 {
6     while (true)
7     {
8         NotePad++;
9     }
10 }
```

Sitio oficial de Notepad++

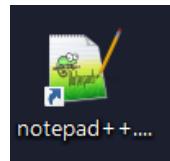
Abra la ubicación del archivo descargado y haga doble clic sobre él. Al ser un archivo ejecutable notificará con una advertencia de los posibles daños al equipo, pero lo ha descargado desde su sitio oficial, por lo que es un archivo seguro.



Abrir archivo ejecutable

Paso 3: Ejecutar Notepad++.

Una vez instalado en su equipo, aparecerá un ícono de acceso directo en su escritorio.



Ícono acceso directo a Notepad++

Ya puede ejecutarlo y entrar en el editor. Aparece un archivo con las últimas actualizaciones del editor. En la página web oficial puede encontrar una gran cantidad de [documentación](#) para explorar este gran editor que tiene infinitud de posibilidades.

The screenshot shows the Notepad++ application window with the title bar "C:\Program Files (x86)\Notepad++\change.log - Notepad++". The menu bar includes Archivo, Editar, Buscar, Vista, Codificación, Lenguaje, Configuración, Herramientas, Macro, Ejecutar, Plugins, Ventana, and ?.

The main pane displays the content of the "change.log" file:

```
1 Notepad++ v8.1.5 new features, enhancements & bug-fixes:
2
3 1. Make Insert date time customizable.
4 2. Make date time order reversible for short & long format.
5 3. Add Insert Date/Time commands shortcut & macro capacities.
6 4. Fix regression: no error message when locked file could not be saved - Show file locked warning now.
7 5. Make "Confirm Save All" dialog more clear: use "Always Yes" button instead of "Cancel".
8 6. Improve JSON, PHP, PYTHON, CSS, BANNC and VHDL syntax highlighting in dark mode.
9 7. Fix inconsistency of Document List context menu.
10 8. Enable/disable ext column via Document list context menu directly.
11 9. Add path column in Document list panel.
12 10. Allow custom color for Document Map.
13 11. Fix docked/float panels in RTL layout issue.
14 12. Fix CSS pseudo-elements not recognized issue. Update CSS keywords and add new styles.
15 13. Fix issue that current directory not been applied after changing current directory option in preferences.
16 14. Sort Default language combobox of "New Document" section in Preferences.
17
18
19
20 More fixes & implementations detail:
21 https://notepad-plus-plus.org/downloads/v8.1.5/
22
23
24 Included plugins:
25
26 1. NppExport v0.3
27 2. Converter v4.3
28 3. Mime Tool v2.6
29
30
31 Updater (Installer only):
32
33 * WinGup (for Notepad++) v5.2
34
```

At the bottom, status bar information includes: Normal text file, length : 1.298, lines : 34, Ln : 1, Col : 1, Pos : 1, Windows (CR LF), UTF-8, and INS.

Entorno de Notepad++

4.1.5. Adobe Dreamweaver

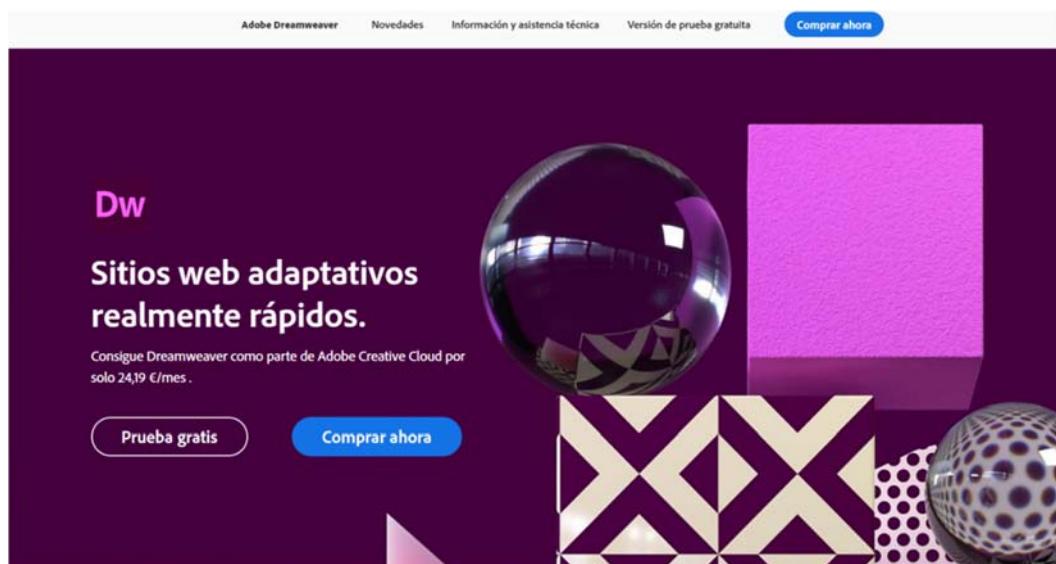
Adobe Dreamweaver es un editor HTML para desarrollar sitios, páginas y aplicaciones web desarrolladas por Adobe. Es un editor de pago bajo licencia de Adobe Creative Cloud CC. Este editor goza de éxito desde los años 90 y aún mantiene una buena posición en el mercado frente a sus principales competidores. Aunque ha sufrido de muchas fallas en su programación, ha ido depurándose versión tras versión hasta convertirse en un IDE fiable que ha sabido integrar todas las herramientas necesarias para trabajar las diferentes aplicaciones web en un mismo framework. Dreamweaver puede ser usado en los diferentes sistemas operativos: Windows, Mac y Linux.

Sus características más destacables son:

- **Live View**, una característica que permite trabajar en tiempo real, es decir, los cambios aplicados pueden verse al instante en el navegador.
- **Rapidez y flexibilidad**. Dreamweaver sugiere la mejor codificación de código para simplificar y reconocer los diferentes lenguajes para ayudar a reducir errores y tiempo de trabajo.
- Por defecto, dispone de **plantillas** ya configuradas y ajustadas según qué tipo de proyecto. Las puede elegir desde el principio, antes de iniciar el proyecto.

Paso 1: descargar Dreamweaver.

Dreamweaver es un editor de pago bajo licencia de Adobe Creative Cloud, pero siempre puede optar antes por la versión de prueba gratis y luego decidir si suscribirse a la versión de pago. Puede descargar la versión de prueba gratuita desde su web oficial de Adobe.



Página web de Adobe de donde descargar Dreamweaver

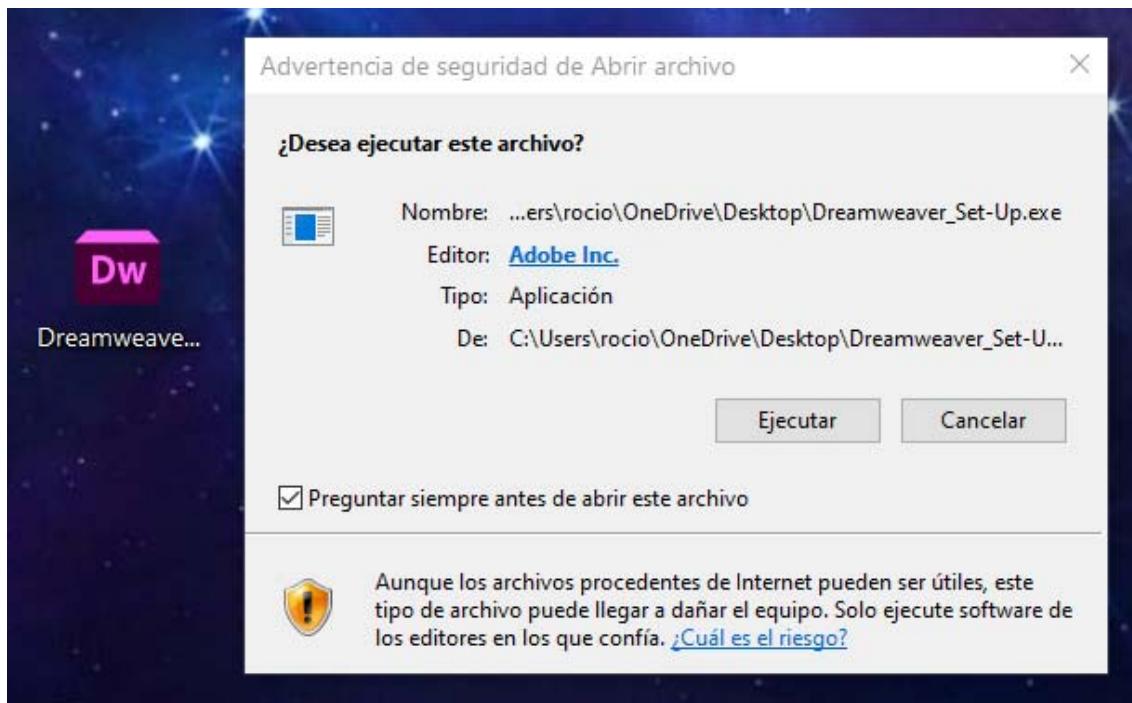
Al pinchar sobre “Prueba gratis”, aparecerá una pantalla como la siguiente y se descargará automáticamente el instalador de Dreamweaver en su carpeta de descargas de su ordenador.

A screenshot of a web browser showing the download page for the Dreamweaver installer. The address bar shows the URL "creativecloud.adobe.com/apps/download/dreamweaver". The main content area has a large instruction: "Sigue las instrucciones que vienen a continuación para instalar Dreamweaver". Below this, a note says: "Haz clic en el instalador y luego sigue las instrucciones. (Para tu información: la aplicación de escritorio de Creative Cloud viene con tu descarga)". In the center is a large icon of a box containing the Adobe logo. At the bottom, there's a message: "¿No funciona la descarga? Reiniciar descarga | Ayuda" and "Volver a Adobe.com". A blue callout box contains the text: "Usa el siguiente instalador para obtener la última versión de Dreamweaver compatible con tu sistema operativo." At the very bottom, there's a file download bar showing "Dreamweaver_Set-....exe".

Página de descarga automática del instalador de Dreamweaver

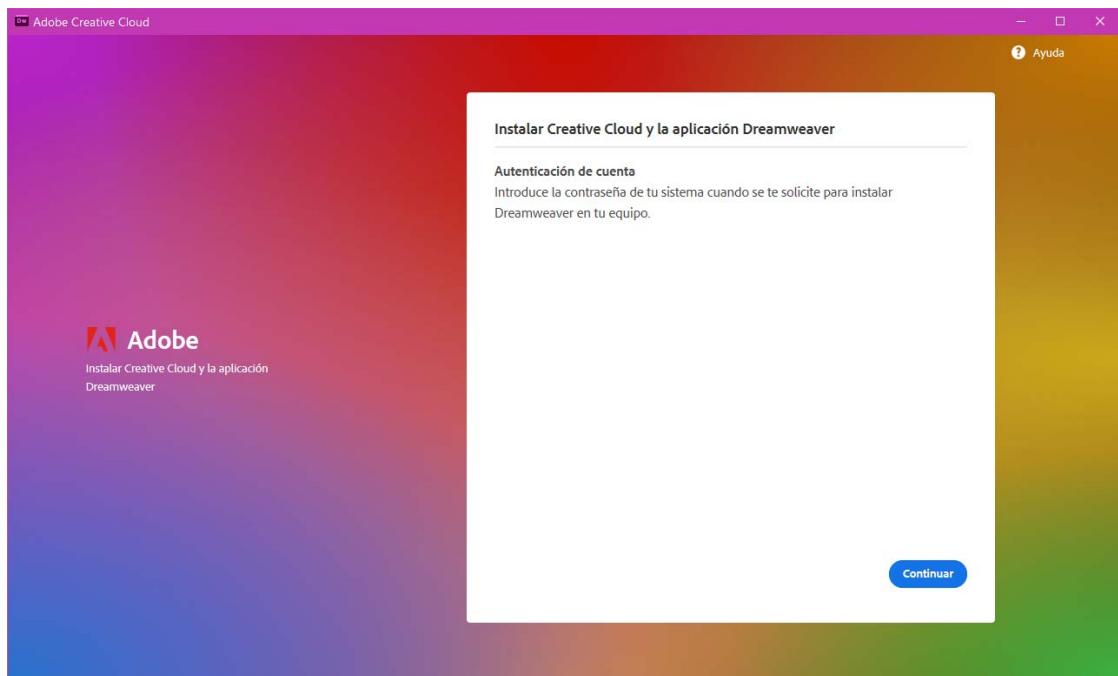
Paso 2: Instalar Dreamweaver.

Se abre la ubicación del archivo descargado y se hace doble clic sobre él. Al ser un archivo ejecutable notificará con una advertencia de los posibles daños al equipo, pero lo ha descargado desde su sitio oficial, por lo que es un archivo seguro.



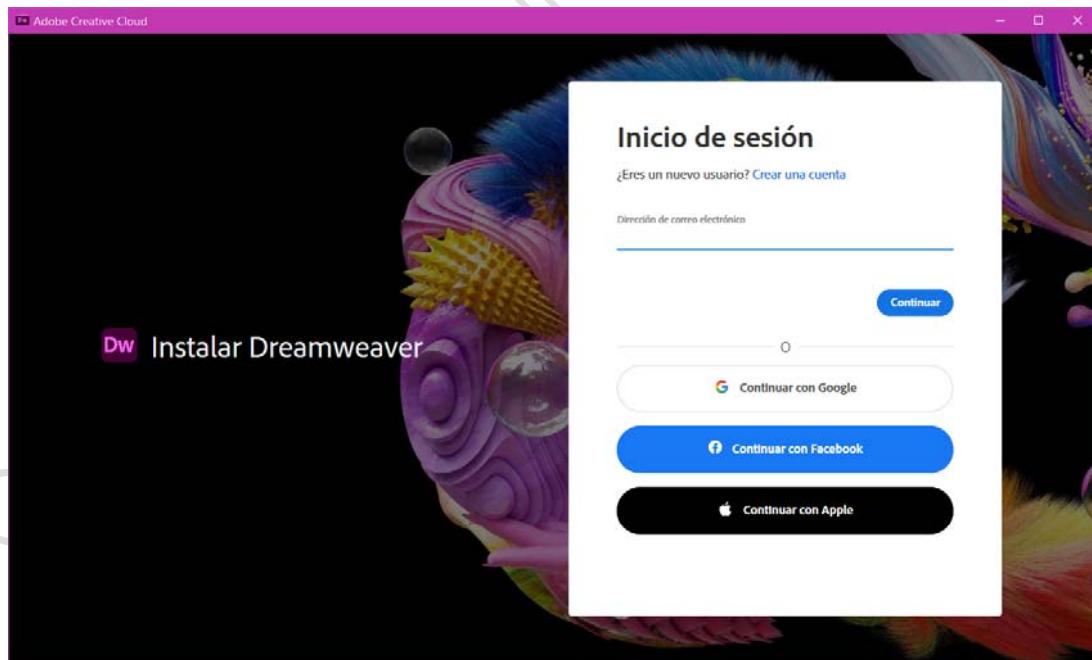
Abrir archivo ejecutable de Dreamweaver

Los programas de Adobe Creative Cloud tienen la particularidad de que siempre debe acceder a ellos mediante credenciales. Cuando pulsa en el botón “Continuar” para proceder a la instalación, llevará a una pantalla donde solicitará introducir un email y una contraseña.



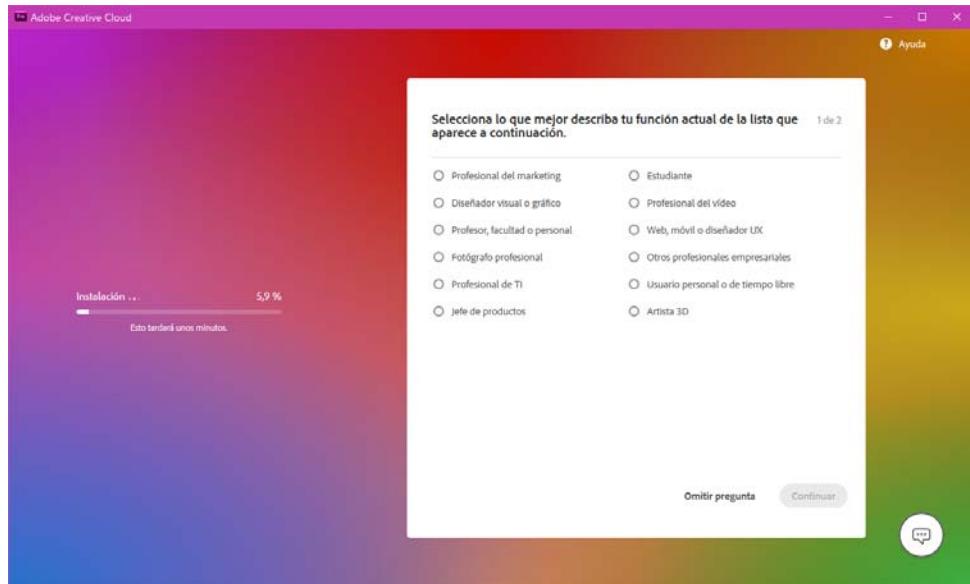
Pantalla de petición de credenciales de Adobe

Debe loguearse con su usuario y contraseña si previamente tiene registro en Adobe, o bien crear una nueva cuenta para proceder a la instalación y uso de Dreamweaver. En ambos casos, seguirá los pasos hasta completar una de las dos opciones.



Pantalla de credenciales

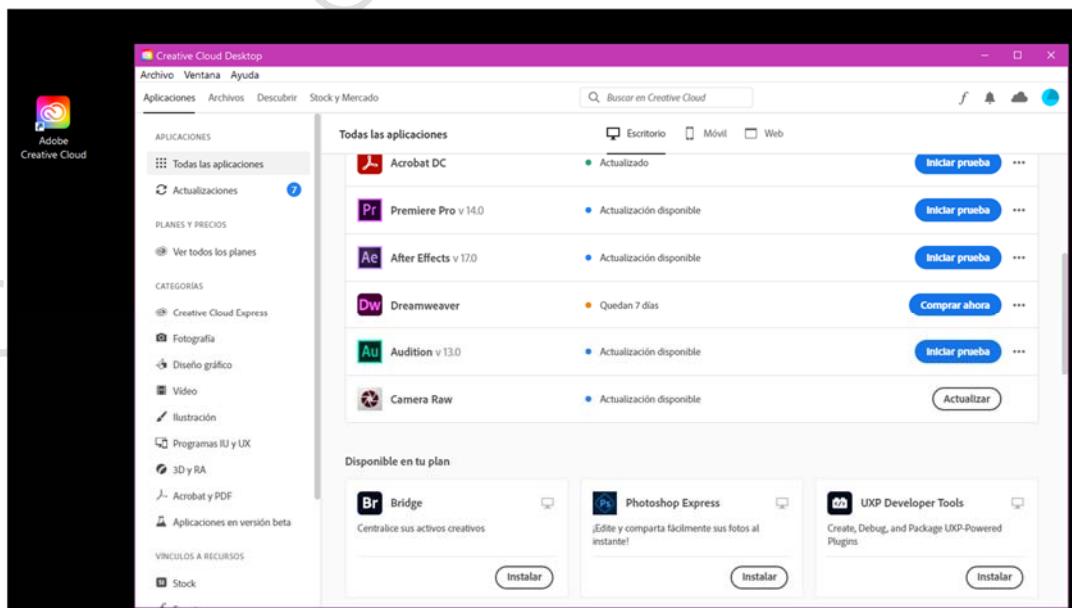
Cuando complete los pasos para crear una nueva cuenta o acceder con sus credenciales, en caso de tener cuenta previamente, el instalador automáticamente comenzará la instalación. Tiene la opción de contestar al cuestionario simplemente para tener un perfil más completo y ofrecer más datos para comunicaciones comerciales y demás recursos de Adobe.



Pantalla de instalación

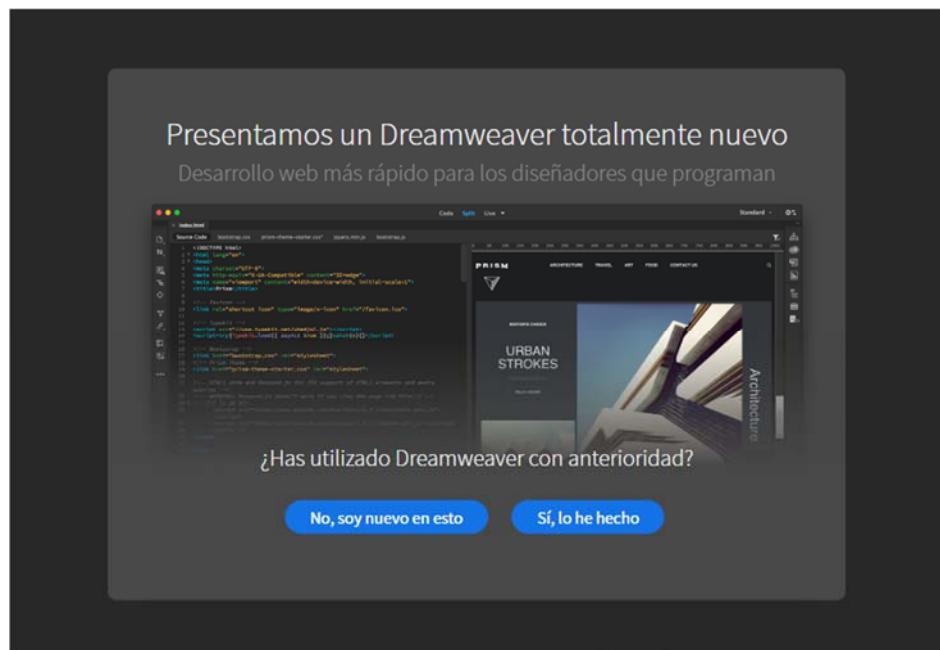
Paso 3: Ejecutar Dreamweaver.

Durante el proceso de instalación, en su escritorio se creará un ícono de acceso directo a la aplicación de Creative Cloud Desktop donde tendrá acceso a todos los recursos de Adobe y desde donde podrá gestionar todas las aplicaciones de Adobe que tenga instaladas.



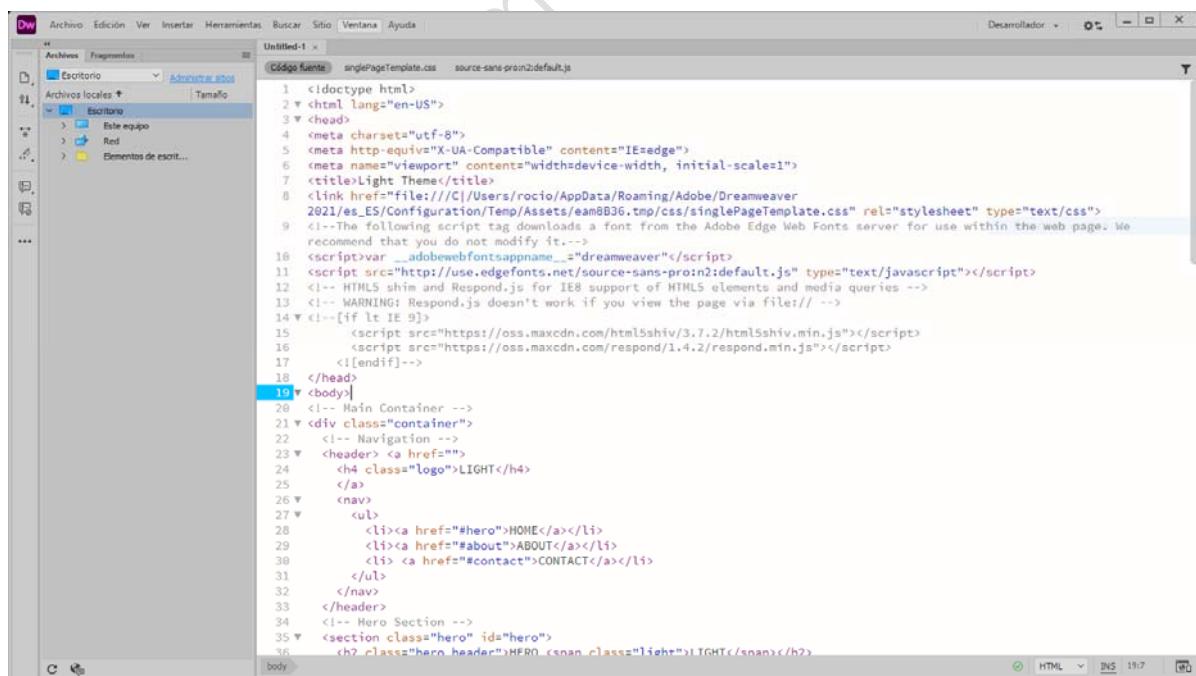
Aplicación de Creative Cloud

Al finalizar el proceso de instalación, la aplicación de Dreamweaver se abrirá automáticamente y encontrará una serie de preguntas muy útiles que facilitará la introducción a este editor web.



Pantalla de inicio de Dreamweaver

Aparecerá el entorno de trabajo de Dreamweaver en base a las preferencias que haya seleccionado previamente en el cuestionario de inicio.



Entorno de Dreamweaver

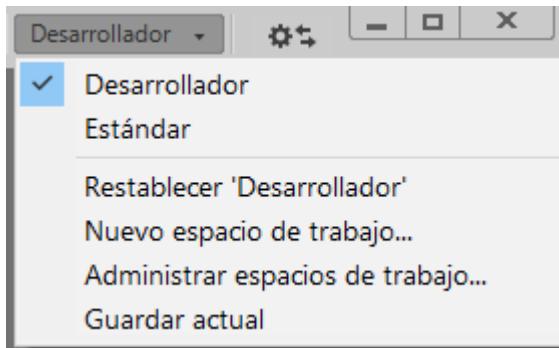
Su interfaz es sumamente modificable según su nivel y necesidades. Dispone de diferentes espacios de trabajo predefinidos como “desarrollador” y “estándar” y cada uno dispone las herramientas y opciones adaptadas al tipo de desarrollo de trabajo. También se puede crear un espacio de trabajo personalizado

ajustado

a

sus

preferencias.



Espacios de trabajo de Dreamweaver

Dreamweaver es un editor que se ajusta al nivel del usuario con, por ejemplo, la opción de arrastrar y soltar elementos para componer visualmente la página y trabajar desde plantillas predefinidas gracias a Bootstrap incorporado hasta opciones más complejas que se ajustan a niveles más expertos para editar el código bruto de forma directa.



Bootstrap es un potente framework que combina CSS y JavaScript para dar estilo y funcionalidad a los elementos de una página HTML. Hace que la usabilidad de un sitio web se potencie y además convierte a las páginas web en sitios responsive adaptativos a cualquier dispositivo móvil.

TOME NOTA

Para conocer más sobre este framework y cómo se incluye en multitud de editores visite la web oficial de [Bootstrap](#).

4.1.6. Visual Studio Code

Visual Studio Code es un programa de código abierto desarrollado por Microsoft en 2015 para las plataformas Windows, Linux y macOs. Visual Studio se compone de diferentes versiones, desde el más básico para escribir código HTML para crear páginas web hasta un completo IDE para desarrolladores avanzados que trabajen con proyectos exigentes.

Visual Studio Code, en su versión más básica, no deja indiferente al usuario ya que contiene todo lo necesario para comenzar un proyecto desde cero y, lo más importante, permite escalar el proyecto todo lo que el usuario exija. Cuenta con una enorme red de recursos en continuo crecimiento gracias

a la aportación de los usuarios de la comunidad. Dispone de una gran cantidad de herramientas, recursos y extensiones necesarios para afrontar cualquier tipo de proyecto.

Aunque dispone de una versión completamente gratuita con todas las funcionalidades totalmente operativas, también existen otras licencias para sectores profesionales y de empresas.

Visual Studio Community 2022	Visual Studio Professional 2022	Visual Studio Enterprise 2022
IDE gratuito con todas las características para alumnos, desarrolladores de código abierto y desarrolladores individuales	Herramientas de desarrollo profesionales, servicios y ventajas para suscripción para equipos pequeños	Solución completa para satisfacer las exigentes necesidades de calidad y escala de equipos de todos los tamaños
Descarga gratuita	Prueba gratuita	Prueba gratuita

Comparativa de licencias de Visual Studio

Algunas de las características más reseñables de Visual Studio que mejoran la eficiencia y productividad son las siguientes:

- **Subrayados ondulados y Acciones rápidas:** Los subrayados ondulados de color rojo debajo de las palabras alertan de errores que debe corregir para que el código corra correctamente.

`public int Horario() => * Math Log8`

Las Acciones rápidas se muestran mediante un icono de bombilla que aparece junto a la palabra y, si pulse sobre el mismo, se despliegan una serie de opciones rápidas para corregir el error.

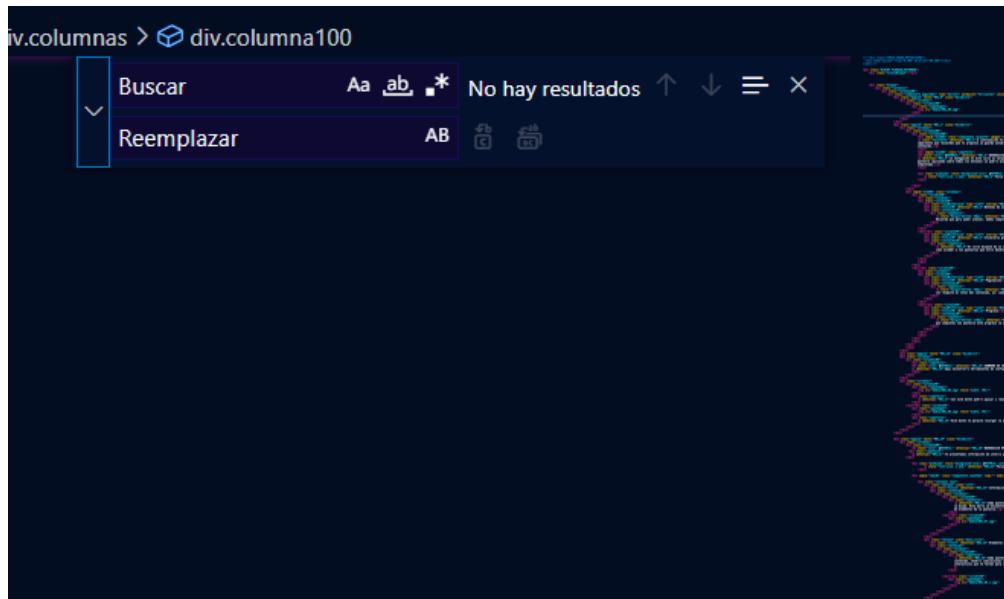


Ejemplo del icono Acción rápida

La ventaja de estas opciones es que indican posibles errores al tiempo que se escribe el texto, lo que significa que ahorra tiempo en el momento de compilar y correr el código ya que, si este contiene errores, ralentizará el proceso y obligará a revisar el código para subsanar el error.

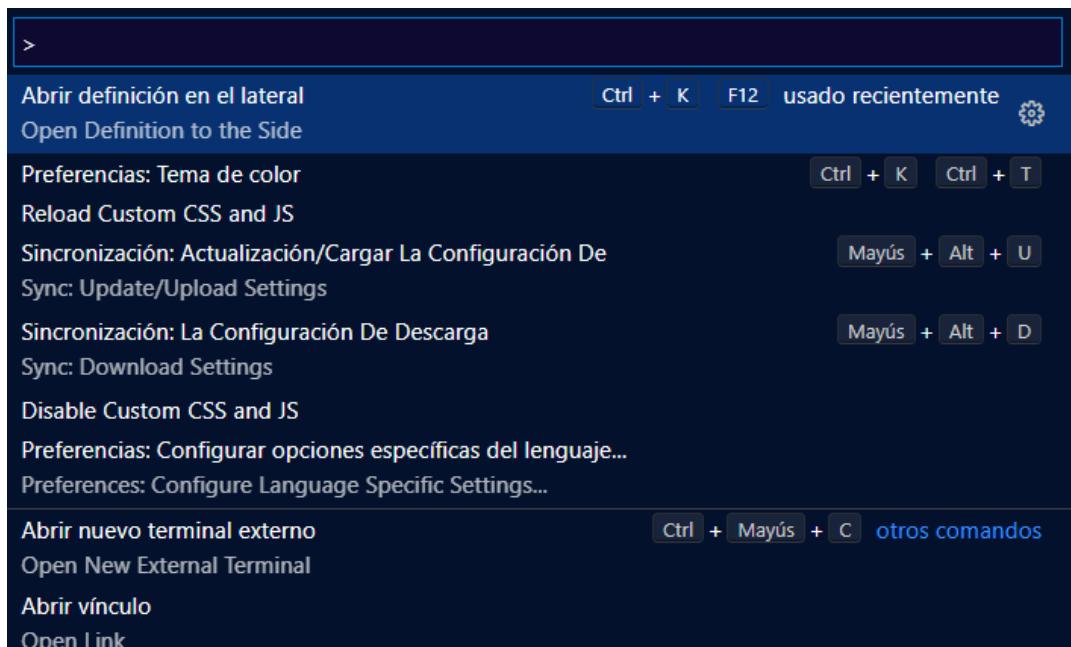
- **Limpieza de código:** Aunque actualmente solo está disponible para el código C#, esta opción permite solucionar problemas de escritura de código antes de su posterior revisión y testeo.

- **Búsqueda de Visual Studio:** Dispone de múltiples opciones de búsqueda en Visual Studio para encontrar rápidamente las características, funcionalidades y código que necesite buscar y reemplazar. Si pulsa el comando Ctrl+F puede acceder al buscador de reemplazo rápido de código.



Buscador de reemplazo de código Ctrl+F

También existe otro buscador, llamado Paleta de comandos, con el que se accede a múltiples funciones de configuración dentro de Visual Code. Desde aquí, puede cambiar, por ejemplo, la configuración visual del editor para ajustar a sus gustos y necesidades desde: cambiar el color, cambiar el tamaño de la letra, cambiar la disposición de los elementos de trabajo, el tamaño y forma del puntero. También se puede configurar cada lenguaje, la sintaxis que estos presentan, si desea activar o desactivar ciertos lenguajes de programación, etc. En definitiva, puede personalizar Visual Code todo lo que necesite.



Buscador paleta de comandos Ctrl+Mayus+P

- **CodeLens:** Esta opción que ofrece Visual Code es una herramienta muy útil para buscar referencias en una parte del código, cambios, errores y hacer pruebas unitarias.
- **IntelliSense:** Esta es una característica muy interesante que ayudará a tener una escritura más eficiente sobre el código. Mientras escribe, se mostrará información sobre el código directamente en el editor e, incluso, permitirá insertar pequeños fragmentos de código. Es similar a tener pequeños fragmentos de documentación automatizada para insertar directamente. Además, IntelliSense, reconoce el lenguaje en el que escribe y ofrecerá información acorde al mismo.

The screenshot shows the Visual Studio Code editor with the following code snippet:

```
arrayElementosSueltos = ['botonAyuda'];
arra|
```

An IntelliSense dropdown menu is open, listing several suggestions:

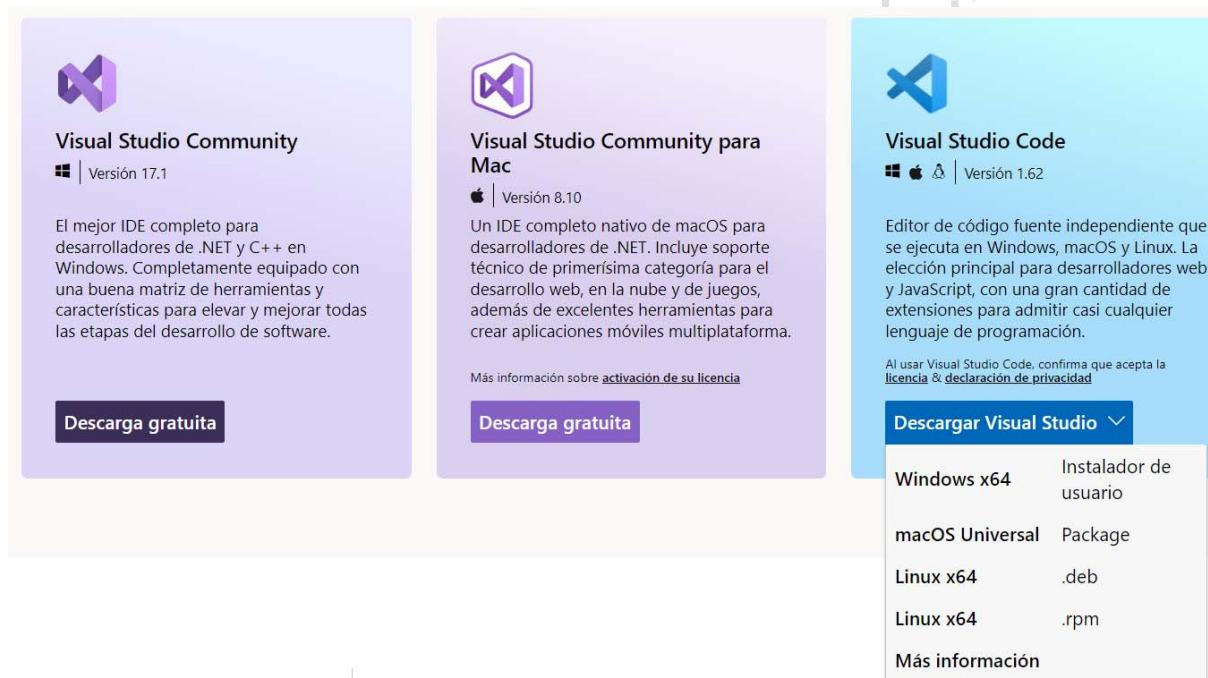
- abc arrayElementosSueltos
- abc arrayMenuConfiguracion
- /** abc arrayMenuSecundario
- * A abc arrayMenuUtiles
- * { [o] Array
- * D [o] ArrayBuffer
- * e [o] Int8Array
- * i [o] SafeArray
- *** [o] Int16Array
- [o] Int32Array
- /* [o] Uint8Array
- vars [o] PluginArray

Ejemplo de IntelliSense

- **Refactorización:** Refactorizar significa reestructurar el código fuente internamente sin que afecte a su comportamiento. Es una forma de “limpiar” el código, ya sea eliminando líneas duplicadas, usando clases, usando patrones para problemas específicos, encontrando bugs, etc. Refactorizar permite llevar el código al siguiente nivel de calidad sin alterar su funcionalidad, conseguirá un código mucho más claro y fácil de leer y de mantener en el tiempo. Visual Code, gracias a sus opciones inteligentes, permite realizar operaciones como el cambio de nombre de variables, extracción de líneas de código o el cambio de orden en los parámetros, todo ello para pulir su código lo máximo posible.

Paso 1: descargar Visual Studio Code.

La descarga de Visual Code es muy sencilla e intuitiva. Desde la [página web oficial de Visual Studio](#), ofrecen múltiples opciones para descargar la versión que mejor se adapte a sus necesidades y a las condiciones de hardware y software que presente su equipo.

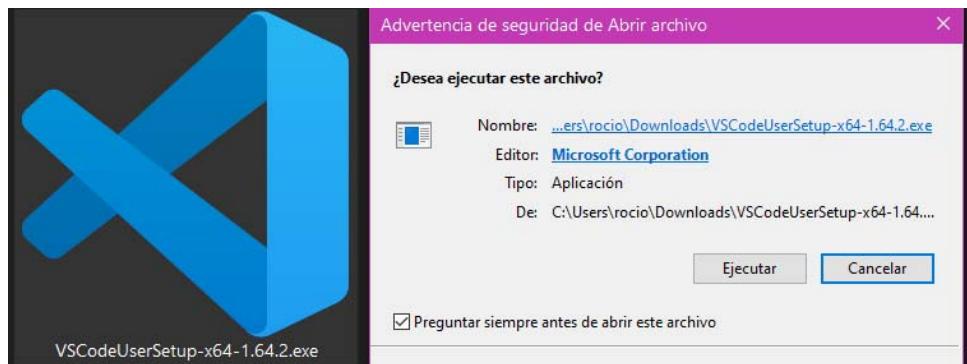


Múltiples opciones para descargar Visual Studio

Al seleccionar la mejor opción que se adapte a los requisitos de su sistema, la descarga comenzará automáticamente.

Paso 2: Instalar Visual Studio Code.

Una vez la descarga ha finalizado puede ejecutar, con total seguridad, el instalador.



Descarga segura de Visual Code desde su página web

El proceso de instalación es muy sencillo, ya que, está guiado por un asistente virtual durante todo el proceso de instalación del programa. Puede encontrar la información relativa a su instalación en la [página web oficial de Visual Studio Code](#).

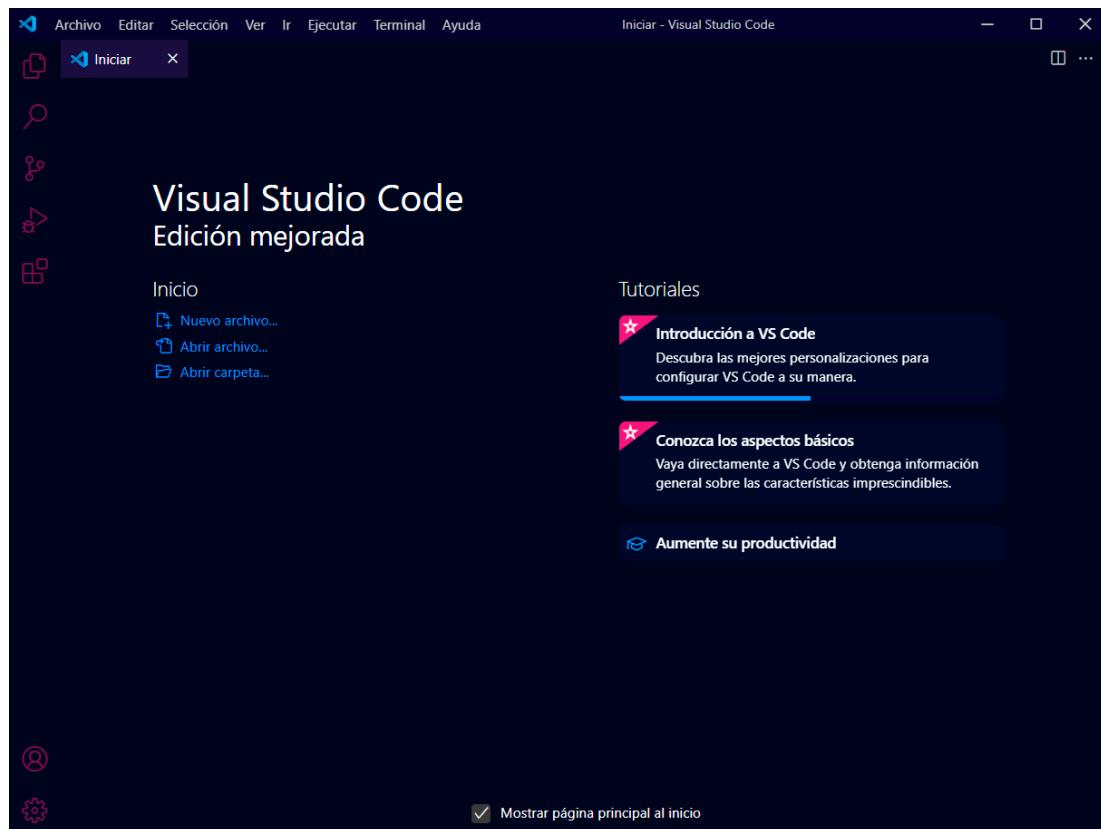
Paso 3: Ejecutar Visual Studio Code.

El proceso de instalación es rápido, además, Visual Code es un programa que consume muy pocos recursos. Una vez instalado, se creará un ícono de acceso directo en su escritorio.

Ya puede ejecutar el programa. Su apariencia preliminar será algo parecido a esta, aunque, como se ha mencionado antes, puede personalizar casi hasta el más mínimo detalle de la interfaz. Además, en la pantalla de inicio puede encontrar tutoriales muy prácticos para aprender los aspectos básicos de Visual Code, desde personalizar su apariencia hasta un repaso por sus características imprescindibles para aprender a manejar esta potente herramienta.



Icono de acceso directo de Visual Code



Interfaz de Visual Code



4.2. Funciones y características

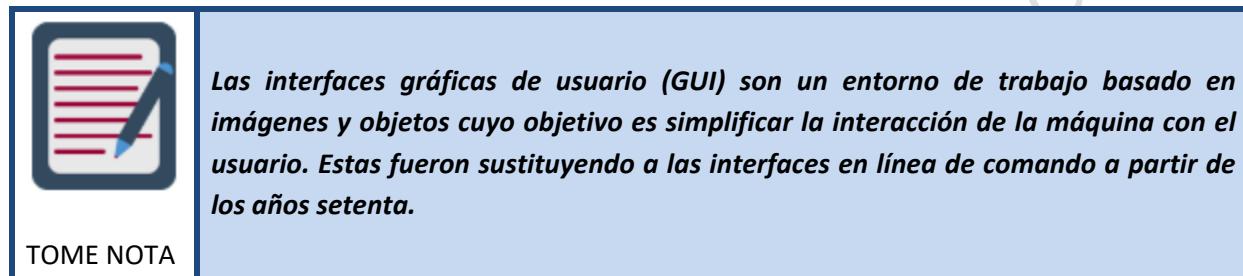
Los programas de edición web ofrecen unas características y funcionalidades muy útiles para el desarrollo de páginas web. En este caso se verán todas las opciones que Visual Studio ofrece al usuario. Cabe destacar que Visual Studio está en constante actualización. La gran comunidad de

programadores que respaldan su desarrollo hace que este programa esté a la última en cuanto elementos y funcionalidades de programación. A menudo hay actualizaciones disponibles que se descargarán e instalarán de forma automática implementando mejoras para su trabajo diario.

4.2.1. Descripción de los elementos de las herramientas web

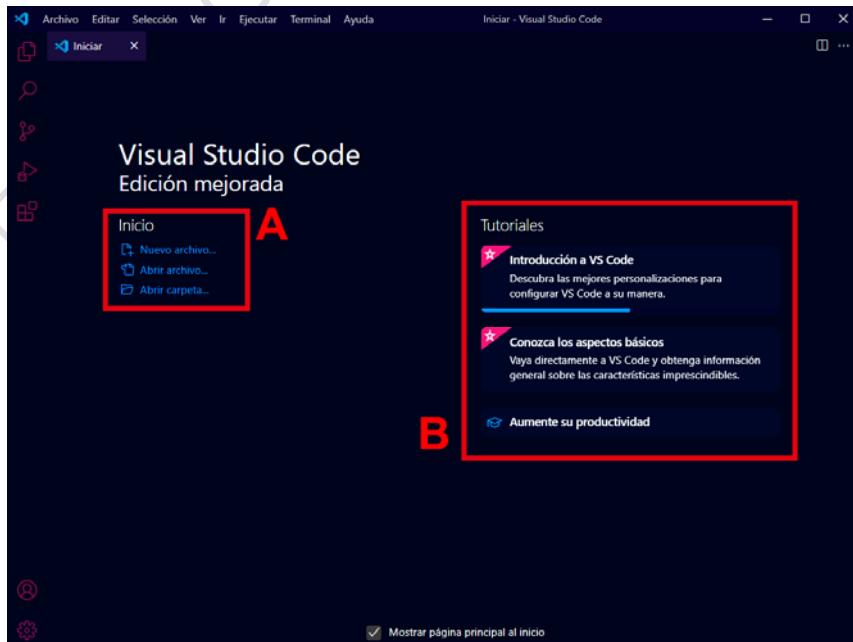
Los programas de edición disponen de una interfaz gráfica, denominada GUI, en la cual se pueden desarrollar todas las operativas para desarrollar un sitio web minimizando la construcción con código.

Es un entorno mucho más visual donde puede encontrar los elementos necesarios como herramientas y opciones en el menor tiempo posible.



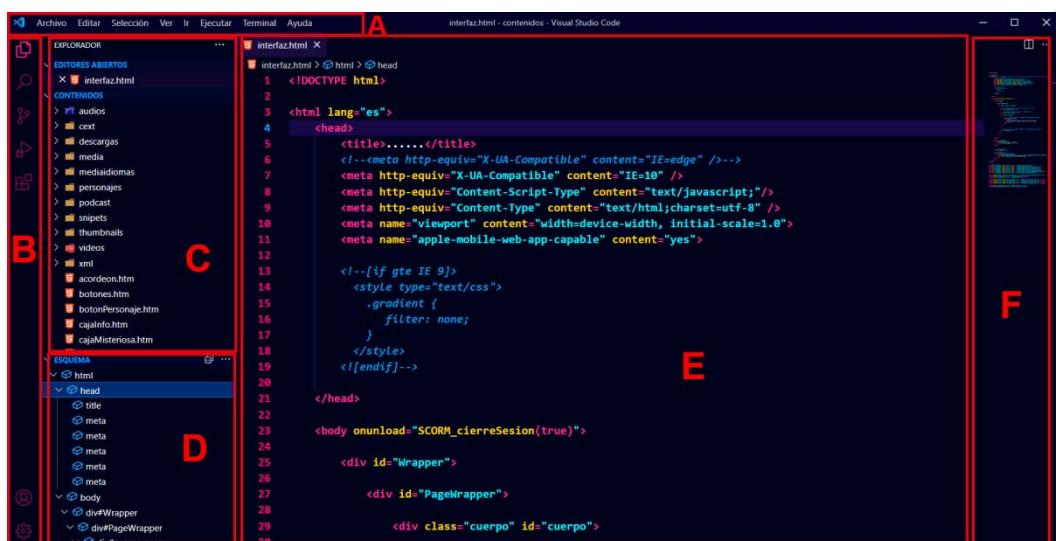
4.2.2. Ubicación de los elementos de las herramientas web

A continuación, se verá dónde se ubican las principales herramientas de Visual Studio Code dentro de su interfaz. La primera pantalla que aparece al iniciar Visual Code es la de Inicio y en ella tiene varias opciones para comenzar a usar este editor.



Pantalla de Inicio Visual Code

- A (Inicio):** En esta sección se encuentran opciones para crear un nuevo archivo, abrir archivos ya existentes o abrir una carpeta que contenga varios archivos y que Visual Code permite visualizar a la vez para moverse entre ellos sin salir del editor.
- B (Tutoriales):** Un apartado muy práctico para los primeros pasos. Dispone de un Tutorial de introducción a Visual Code y otro sobre los aspectos básicos que ayudarán a tener una visión global de las principales características de la herramienta.



Interfaz de trabajo de Visual Code

El aspecto de la interfaz de trabajo en pleno desarrollo con herramientas básicas tendría un aspecto similar.

A (Menús): Esta barra de menús permite acceder a la mayoría de opciones de Visual Code.

- En **Archivo** encuentra opciones como crear nuevo archivo, nueva ventana, abrir o cerrar archivos y carpetas, abrir archivos recientes, guardar área de trabajo, duplicar área de trabajo, guardar y guardar como, autoguardado, preferencias, revertir archivo, cerrar editor, ventana o carpeta y salir del programa.
- En **Editar** puede deshacer o rehacer acciones, cortar, copiar y pegar fragmentos de código; buscar y reemplazar; buscar y reemplazar en archivos. Estas tareas son muy recomendables automatizarlas mediante comandos de teclado para agilizar y economizar en tiempo de trabajo. Los comandos básicos más usados son Crtl+C (copiar), Crtl+V (pegar) y Crtl+X (cortar). Además, en Visual Code puede configurar atajos de teclado (*shortcuts*) para personalizarlos. También dispone de [documentación oficial](#) específica sobre estos atajos para Visual Code.
- En **Selección** dispone de múltiples opciones para trabajar línea a línea a través de todo el archivo de código. La opción de cursor múltiple es una herramienta muy interesante que

permite cambiar código en varias líneas al mismo tiempo situando el cursor en varias líneas a la vez.

- En **Ver** puede seleccionar habilitar o deshabilitar visualizar herramientas adicionales para añadir a la interfaz.
- La pestaña **Ir** es muy útil para moverse dentro del código de forma rápida y directa. Puede optar por múltiples opciones para dirigirse a una línea concreta a través de definición, implementaciones, referencias, problemas, cambios, etc.
- En **Ejecutar** dispone de las opciones necesarias para depurar el código.
- Dentro de **Terminal** encuentra opciones relativas a aspectos más complejos de la programación. También encuentra opciones para gestionar tareas.
- Y finalmente, en **Ayuda**, dispone de todos los recursos necesarios de ayuda, actualizaciones, documentación e información adicional sobre Visual Code.

B (Barra de herramientas): En esta parte de la interfaz se aúnan unas opciones muy prácticas para desarrollar su proyecto.

- **Explorador** que contendrá una visual directa sobre los archivos que está trabajando.
- **Buscar y reemplazar** palabras en su archivo o carpetas. Puede buscar por palabras completas, hacer coincidir mayúsculas y/o minúsculas, buscar partes de palabras o expresiones regulares.
- **Control de código de fuente** que sirven para conectar a una carpeta compartida en [Git](#) para desarrollar código junto a otros programadores. Es una especie de carpeta compartida en tiempo real.
- **Ejecutar y depurar** código.
- **Y Extensiones**, una de las opciones más interesantes para buscar *plug-ins* para añadir a Visual Code y conseguir un flujo de trabajo más interesante y adaptado a sus necesidades.

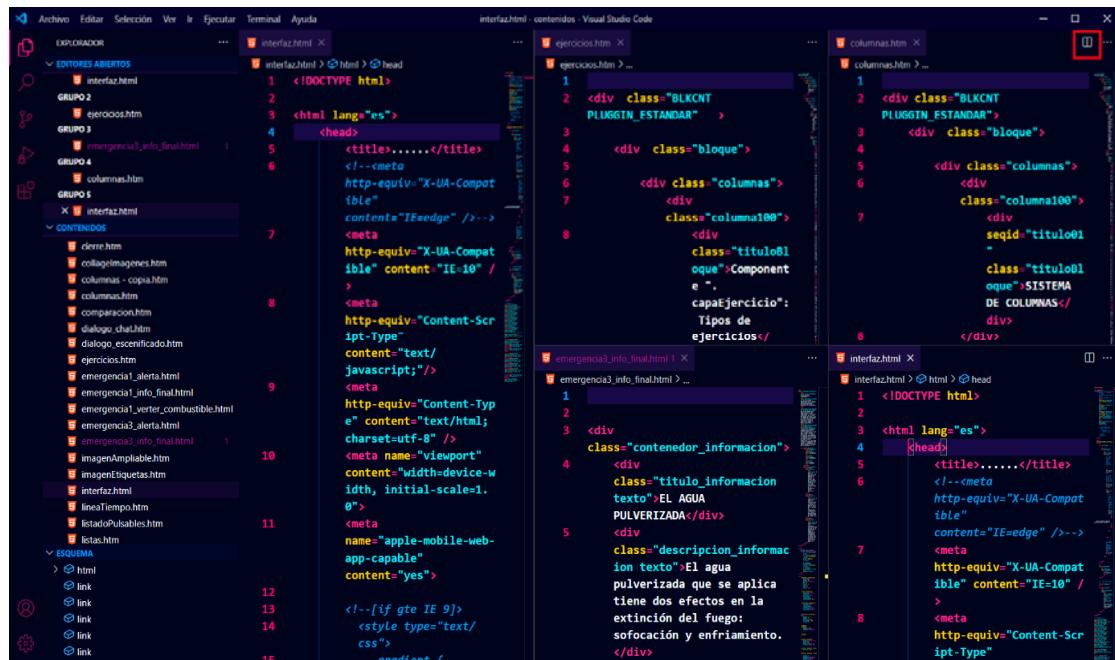
C (Explorador): En esta sección aparece la jerarquía de carpetas y archivos con los que esté trabajando. Permite ir de un archivo a otro sin necesidad de abrir otras ventanas del editor y sin tener que usar carpetas externas al editor. Puede trabajar en todos los archivos que necesite sin salir del editor lo que ayuda mucho a la concentración en el trabajo, permitiendo un flujo constante de trabajo sin distracciones y permitiendo ver todos los procesos de su proyecto de principio a fin.

D (Esquema): Este apartado representa la composición del archivo con el que está trabajando. Dispondrá en forma de esquema las principales partes del archivo como cabecera, body, divs, links, scripts, etc. Pulsando sobre cada uno de ellos conducirá a la parte que quiera visualizar en el archivo.

E (Archivo): En esta parte de la interfaz es donde se visualizará el archivo con el que esté trabajando.

F (Mapa): Esta sección dentro de la interfaz, que puede habilitarse y deshabilitarse, es un mapa en miniatura del archivo que esté abierto y seleccionado en ese momento. Es muy práctico para desplazarse dentro del documento y ahorrar tiempo haciendo scroll. Si sabe aproximadamente en qué parte del archivo buscar, solo tendrá que pulsar sobre la parte correspondiente al mini mapa y, automáticamente, el archivo se colocará en la posición homóloga a la del mapa.

La interfaz de Visual Code permite abrir tantos archivos como necesite gracias a esta opción llamada **Dividir Editor Derecho** (o abajo) que puede encontrar al lado derecho del editor. Permite crear ventanas para abrir varios archivos a la vez para implementar eficacia a su flujo de trabajo.

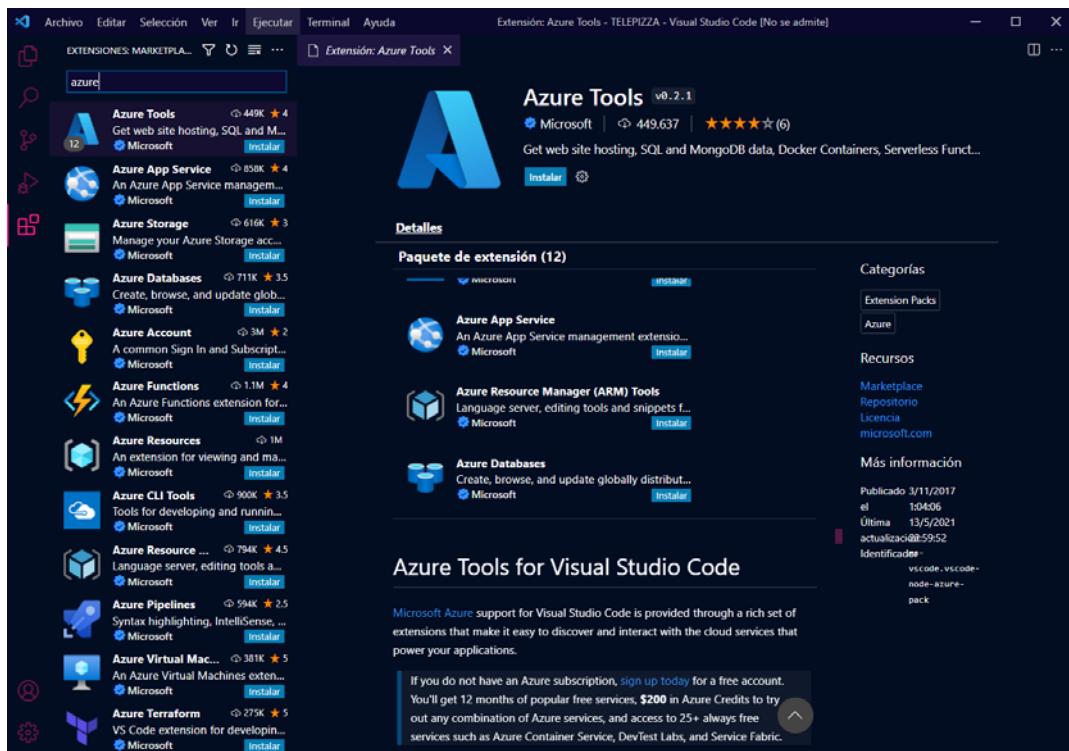


Múltiples archivos abiertos al mismo tiempo en Visual Code

4.2.3. Propiedades de los elementos de las herramientas web

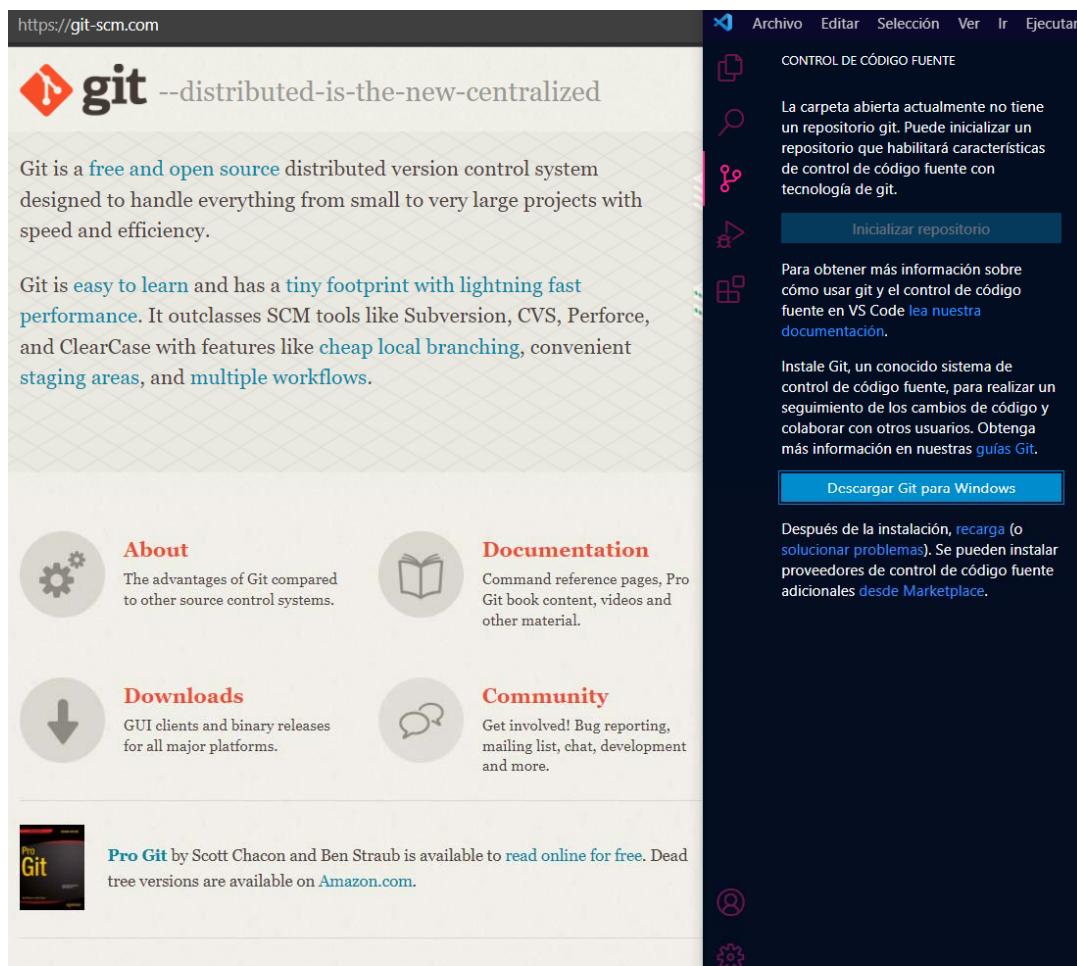
Visual Studio Code presume de ser un editor modular, esto significa que, gracias a todas las extensiones que se pueden añadir, se adapta a cualquier lenguaje y brinda muchas facilidades para desarrollar sus proyectos.

- **Microsoft Azure:** En este editor puede integrar todas las herramientas necesarias para conectar proyectos en la nube de Microsoft Azure. Esto mejora el flujo de trabajo en un equipo, donde los diferentes integrantes pueden trabajar remotamente mientras el proyecto compartido estará actualizado y disponible a través de la nube. Al ser una extensión creada y validada por Microsoft, puede estar seguro de que es una extensión fiable, estable y totalmente compatible.



Infinidad de extensiones para implementar Azure a sus proyectos

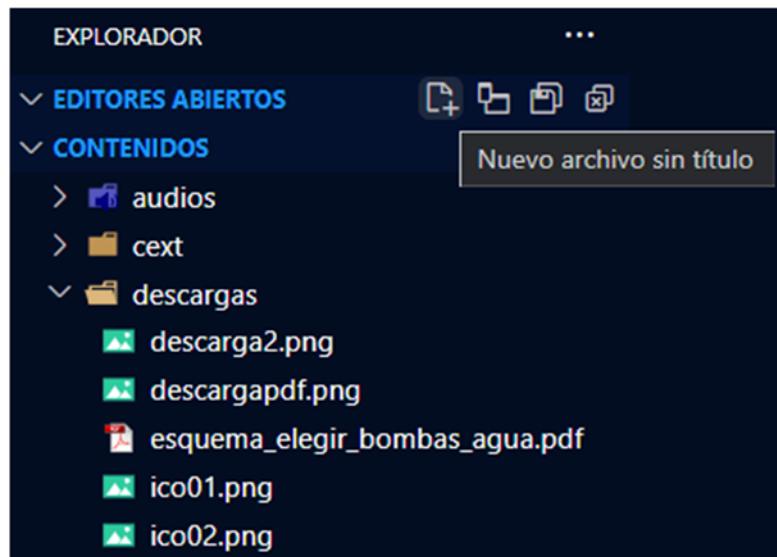
Este no es el único recurso que se puede usar en Visual Studio para compartir proyectos. Visual ya trae incorporado una opción para conectar con **Git** que es un servicio de carpetas compartidas, un servicio como Azure, pero ya integrado en la propia interfaz.



En la interfaz de Visual Studio debe seguir los pasos para instalar Git

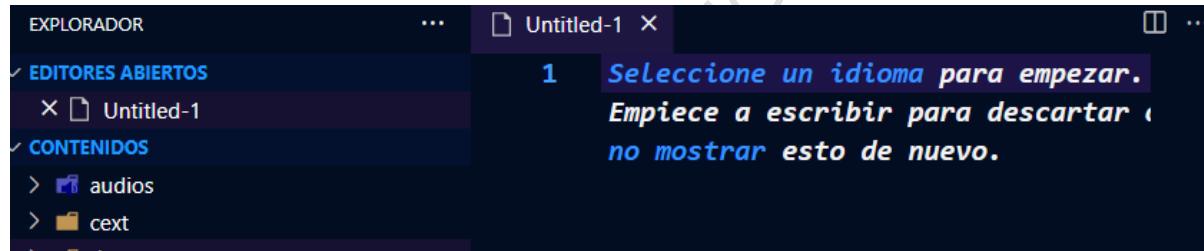
Además de estos servicios de carpetas compartidas, en Visual Studio también puede encontrar extensiones y herramientas para crear un entorno de desarrollo para bases de datos SQL, entre otras.

En Visual Studio puede crear archivos de todo tipo de lenguajes de manera muy sencilla. Creando un nuevo archivo colocando el cursor sobre la barra de **Editores abiertos**. Al hacer esto aparecerán los diferentes iconos, entre ellos, el de **crear un nuevo archivo**.



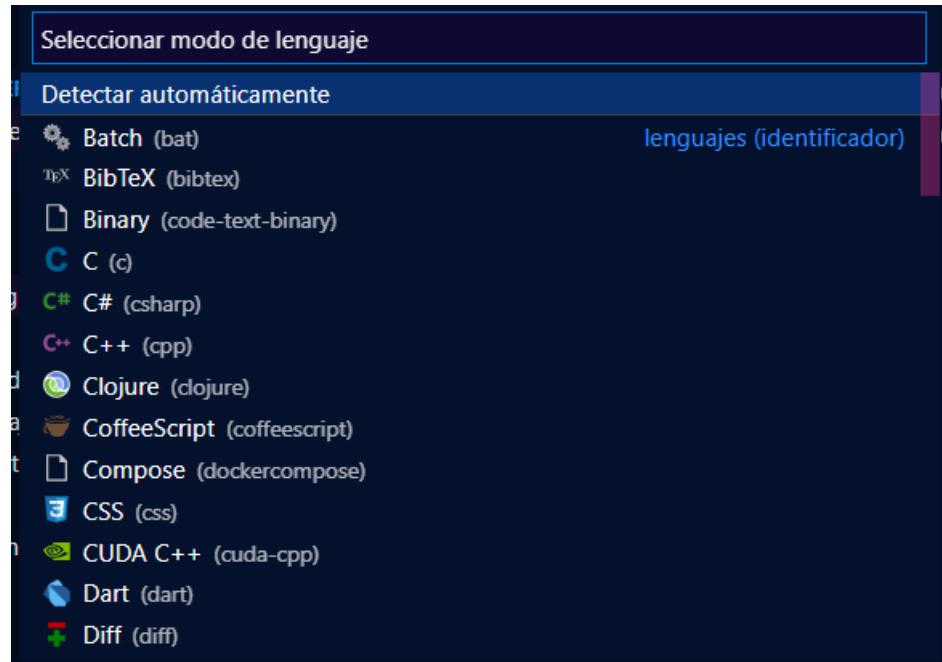
Paso 1: crear nuevo archivo

Al crear este nuevo archivo, las primeras líneas permitirán especificar qué tipo de archivo necesita que sea.



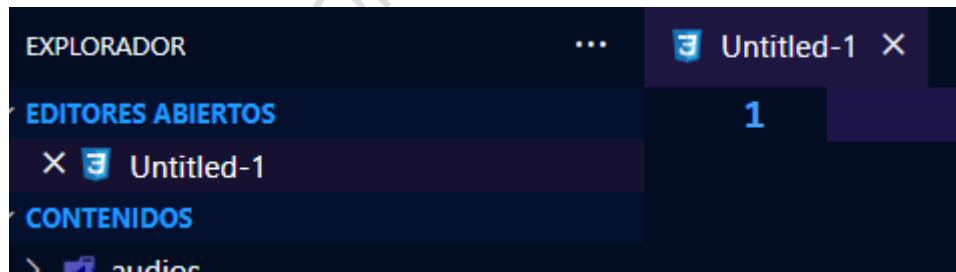
Paso 2: Pulsar sobre Seleccione un idioma

Al pulsar sobre **Seleccione un idioma** se desplegarán las opciones sobre las que tendrá que elegir el lenguaje.



Paso 3: Seleccionar un lenguaje

Una vez seleccionado el lenguaje de programación en el que escribirá su archivo, el ícono del mismo cambiará para que pueda identificarlo inmediatamente. Ya podrá guardarlo en la ubicación deseada con el nombre apropiado. En el caso de este ejemplo, el archivo se guardará con extensión .css automáticamente.



Paso 4: Guardar archivo

4.2.4. Tareas a realizar con una herramienta de edición web

En editores de código como Visual Code se puede trabajar con diferentes lenguajes de programación desde los más básicos como HTML y CSS hasta los más complejos como JavaScript, Java, JQuery, entre otros muchos, para desarrollar todo tipo de proyectos exponenciales. Esto quiere decir que puede desarrollar sus proyectos dentro del mismo editor sin necesitar aplicaciones externas y adicionales que hagan engorroso el trabajo de tener código suelto en diferentes ubicaciones. Este es

uno de los objetivos básicos sobre los que se han venido desarrollando las herramientas de edición web durante estos años. Resulta increíble comprobar la evolución de estos cuando, años atrás, el usuario medio solo disponía de aplicaciones como Notepad o Wordpad en Windows.

Actualmente, estas herramientas de edición web han implementado y desarrollado tecnologías adaptativas dentro de su propio código de desarrollo que les permite adaptarse a las nuevas tendencias y moldearse a los nuevos códigos.

Por todo ello, hoy día, no solo puede *picar* código en estas herramientas de edición web, puede hacer mucho más:

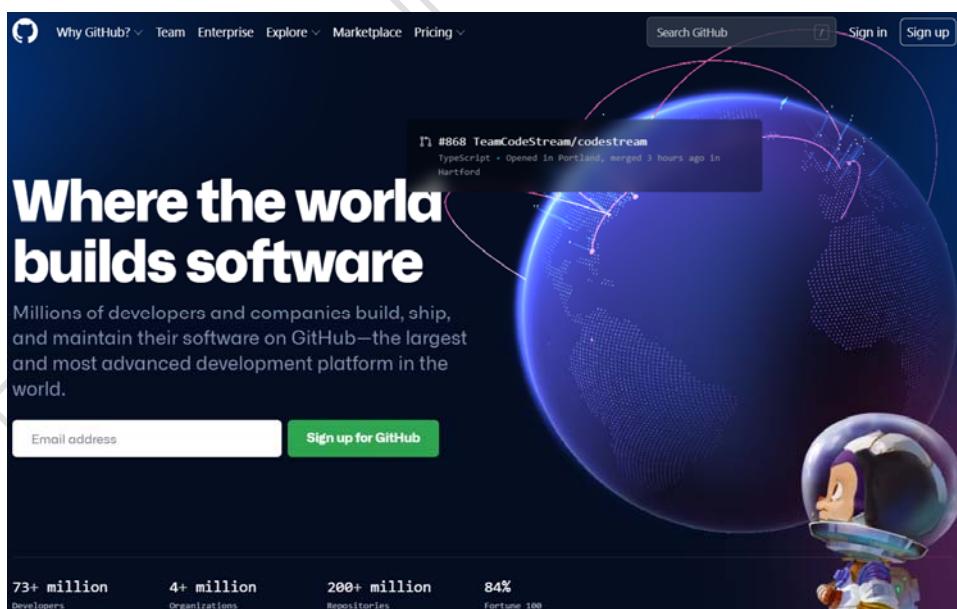
- **Desarrollar:** Estas herramientas de edición web permiten desarrollar proyectos en diferentes lenguajes de programación incluso trabajar diferentes lenguajes dentro del mismo proyecto sin que haya conflictos.
- **Compilación:** Esto significa que lo que escriba en un lenguaje de programación, la herramienta de edición es capaz de aunarla, procesarla y prepararla para convertir ese código en un código que entiendan las máquinas, que es el código binario. Por ejemplo, cuando escribe <h1>Hola mundo</h1> y es lo que aparece en el navegador. Pues ese proceso ya está automatizado en las nuevas herramientas de edición web.
- **Depurar (Debug):** Esta es una opción que salva de muchísimos errores y pérdidas de tiempo buscando un error entre miles de líneas. Depurar el código significa buscar y corregir errores. Las herramientas de edición han integrado y mejorado esta herramienta dentro de sus plataformas lo que permite mejorar el código sin salir de la aplicación o usar aplicaciones externas.
- **Prueba:** Cabe mencionar la importantísima herramienta integrada en la mayoría de editores webs. Se trata la opción de comprobar y probar el código en tiempo real en un navegador web. Algunas extensiones permiten ver los cambios incluso mientras se escribe o modifica el código.
- **Implementar:** Una parte importante a la hora de desarrollar un proyecto es la planificación. Las herramientas de edición web ponen a disposición de los usuarios herramientas muy prácticas para organizar los proyectos de principio a fin y ofrece opciones para controlar el flujo de trabajo, ahorrando así mucho tiempo de gestión y ejecución. Por ejemplo, los servicios de nube Azure o Git, los cuales permiten organizar los grupos de trabajo de manera remota y en tiempo real. A su vez ofrece un control de cambios muy exhaustivo, lo que evita errores que consuman un tiempo valioso.
- **Mejorar rendimiento:** Gracias a que muchas empresas dejan el código de sus programas liberado, los usuarios pueden mejorar el código base. Esto se traduce a que, continuamente, se están implementando mejoras para facilitar el trabajo de programadores de cualquier índole. Se multiplica la potencia de las extensiones y se reduce el consumo de recursos que necesita la herramienta de edición web, lo que significa que aumenta la eficiencia de trabajo sobre el código.

- **Diseñar interfaces de usuario:** Las modernas herramientas de edición web permiten diseñar y programar infinitud de aplicaciones para los usuarios. No solo pueda diseñar y desarrollar páginas web, puede también desarrollar aplicaciones móviles o cualquier otro tipo de aplicación gracias a que puede implementar el uso de cualquier lenguaje de programación. Además, los nuevos editores web permiten crear aplicaciones para los diferentes tipos de plataformas: Windows, Linux e iOS.



Equipo de trabajo

- **Control de versiones:** Este sistema es muy importante para cualquier tipo de proyecto, pero, sobre todo, para grandes proyectos en los que trabajen muchas personas. Este software ayuda a realizar un seguimiento de todas las modificaciones que se realizan sobre un proyecto. Esto es un seguro contra pérdidas que pueden causar mucho daño, tanto económico como de recursos. Por ello, Visual Studio Code trae incorporado Git como parte esencial del trabajo con herramientas de edición web. Merece la pena mencionar otro gran repositorio de control de versiones de gran importancia: [GitHub](#).



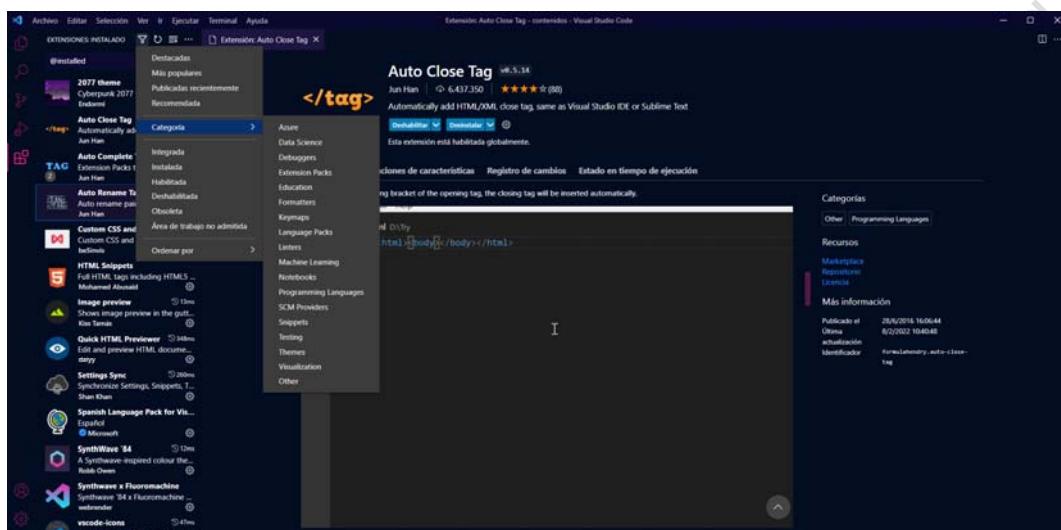
Página web oficial de GitHub

- **DevOps:** Son prácticas que tratan de optimizar y automatizar las diferentes etapas en el flujo de trabajo de un proyecto. Aúna diferentes etapas como la planificación, la codificación, compilación y prueba, puesta en marcha, funcionamiento y supervisión. En todos estos procesos ayudan las herramientas y opciones que incorporan las nuevas herramientas de

edición web. Optimizan el control de los cambios, la organización de las tareas, ayudan a corregir errores en el código, etc. Cada vez se trata más de automatizar procesos para obtener mayor rentabilidad de recursos para conseguir mejores resultados.

4.2.5. El lenguaje de marcas y las herramientas de edición web

Gracias a que Visual Code es un programa de código abierto, programadores de todo el mundo contribuyen a mejorar la funcionalidad de sus herramientas y sus opciones creando multitud de extensiones para agilizar el trabajo. Este aspecto de Visual Code es una de las características que marca la diferencia frente a otros editores, ya que permite una personalización completa. En el margen izquierdo de la interfaz puede encontrar la opción de extensiones.



Diversidad de extensiones para Visual Code

Aquí puede encontrar multitud de extensiones. Se pueden seleccionar por destacadas, recomendadas, populares, publicadas recientemente, por categorías o directamente por el nombre, si lo sabe.

Algunas de las extensiones más comunes y prácticas que ayudarán a agilizar el trabajo con su código son las siguientes:

- **Material Icon Theme:** Gracias a esta extensión se reconoce más rápidamente los archivos pertenecientes a cada extensión. Esto no evitará leer las extensiones de cada archivo. Además, le aportará color y dinamismo a su interfaz.

Icon	Name	Icon	Name	Icon	Name	Icon	Name	Icon	Name
3d	Cypress	Ionic	Nunjucks	Siyuan					
Abc	D	Istanbul	Nuxt	Sketch					
Actionscript	Dart	Jar	Ocaml	Slim					
Ada	Database	Java	Odin	Slug					
Adonis	Denizenscript	Javaclass	Opa	Smarty					
Advpl_include	Dependabot	Javascript	Opam	Sml					
Advpl_prw	Dhall	Javascript-map	Pascal	Snowpack					
Advpl_ptm	Diff	Jenkins	Pawn	Snyk					
Advpl_tpp	Dinophp	Jest	Pdf	Solidity					
Android	Disc	Jinja	Percy	Stencil					
Angular	Django	Jsconfig	Perl	Stitches					
Angular-component	Docker	Json	Php	Storybook					
Angular-directive	Document	Julia	Php_elephant	Stryker					
Angular-guard	Dotjs	Jupyter	Php_elephant_pink	Stylelint					

Iconos de Material Icon Theme

- **Auto Close Tag:** Esta herramienta ayuda a cerrar etiquetas conforme termina de escribir la apertura de la misma. Por ejemplo, cuando termine de escribir `<div>` automáticamente aparecerá el cierre de la misma `</div>`. Con esta extensión se asegura de que toda etiqueta está correctamente cerrada y permitirá arrastrar errores.
- **Auto rename tag:** Esta extensión junto con Auto close tag se convierten en el combo perfecto para evitar errores de escritura. Esta herramienta permite modificar la etiqueta de apertura y cierre al mismo tiempo.

```

<body>
  <div>
    <testdiv>
      <div>Hello!</div>
    </testdiv>
  </div>

  <a href="https://www.microsoft.com">MSFT</a>
</body>

```

Funcionalidad de Auto rename tag

- **Beautify:** Existen infinidad de extensiones como esta cuyo único propósito es conseguir una lectura más clara del código. Si trabaja con gran cantidad de líneas y archivos, con el tiempo, la vista se cansa y puede cometer errores por no hallar fácilmente las líneas que esté trabajando. Por esto, se usan mucho este tipo de extensiones. Su configuración suele distinguir entre etiquetas, clases, nombres, texto simple... a cada elemento se le asigna un

color. Así, con el tiempo, distinguirá rápidamente cada elemento por su color y ayudará a visualizar más rápido la estructura del código.

```
<div class="columnas">
  <div class="columna40">
    <div class="caja">
      <div class="tituloCaja">
        <p>Descripción</p>
      </div>
      <div class="capaTexto">
        <p>Componente que muestra los diferentes tipos de ejercicios:</p>
      </div>
      <div class="capaLista">
        <ul class="noordenada small">
          <li><span><strong>Atributo "xml"</strong>: Define el xml que con<br>estar en el directorio "xml".</span></li>
          <li><span><strong>Clase "IGNOREELEMENT"</strong>: Incluir esta c<br>lase para ignorar el bloqueo de avance.</span></li>
          <li><span><strong>Nota</strong>: Toda la configuración del ejerc<br>icio se hace desde el XML del mismo.</span></li>
        </ul>
      </div>
    </div>
  </div>
```

Colores de Beatify

- **HTML snippets:** esta extensión ayuda al autocompletado del código HTML. Proporciona a su vez fragmentos de código HTML para inserción automática.
- **Image preview:** A la hora de trabajar con imágenes puede llegar a ser muy engorroso recordar los nombres de los archivos o tener que salir a carpetas externas para recuperar algún tipo de imagen. Por ello, esta extensión brinda una imagen en miniatura donde se encuentra el enlace a la imagen dentro del código.

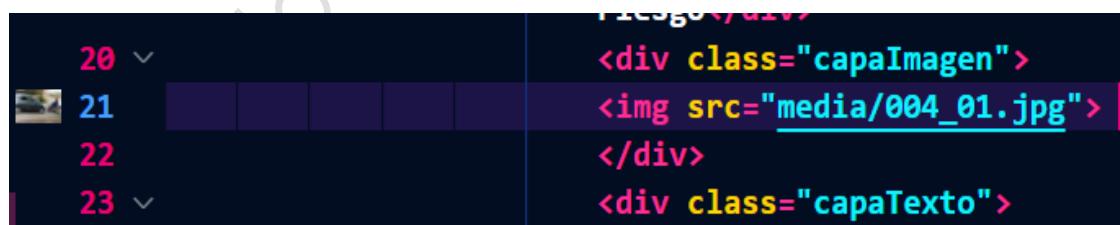


Image preview

- **Color picker:** Esta extensión es muy útil para seleccionar color dentro del mismo código que escriba. Pulsando sobre el cuadro de color puede elegir el color que necesite sin necesidad de saber el código hexadecimal o el código RGB del color. Es una extensión muy recomendable para escribir código CSS.

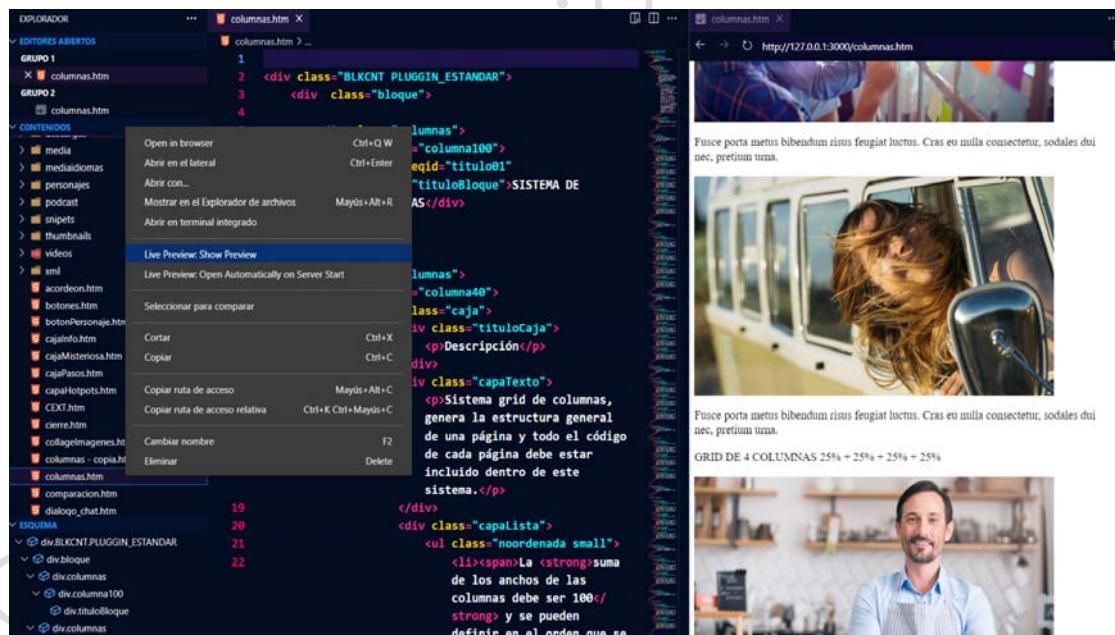
```
.desktop #cuerpo .capaHotspots .hotspotCnt .contenido_hotspot {
    background: white;
}

.caja.destacamos {
    background-color: #fff !important;
    text-align: left;
}

.destacadoRojo{
    border:solid red;
    padding: 2%;
}
```

Color picker

- Live Preview:** Esta extensión de Microsoft permite ver en una ventana dentro de la interfaz y en tiempo real los cambios que hace sobre el código.



Live Preview

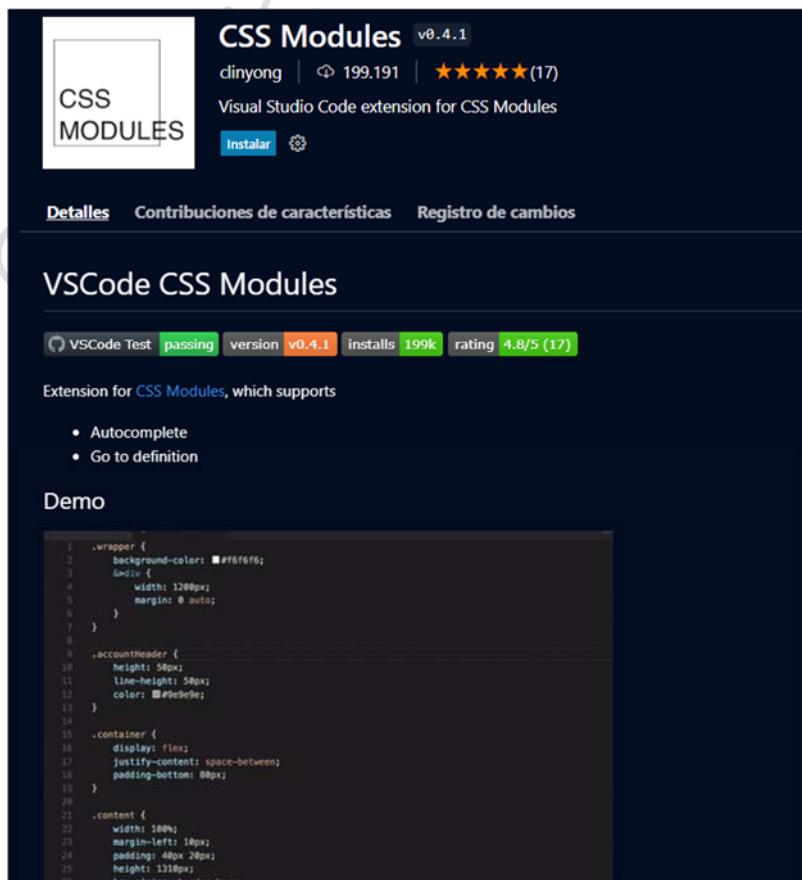
- Paquete de idioma español para VS Code:** Visual Code se puede configurar en múltiples idiomas. Existen extensiones para configurar la interfaz en casi cualquier idioma. En caso de que descargue Visual Studio en otro idioma que no sea español, siempre podrá descargar este sencillo y eficaz *plug-in* donde viene explicado paso a paso cómo configurar el paquete para que traduzca toda la interfaz.



Paquete de idioma español para VS Code

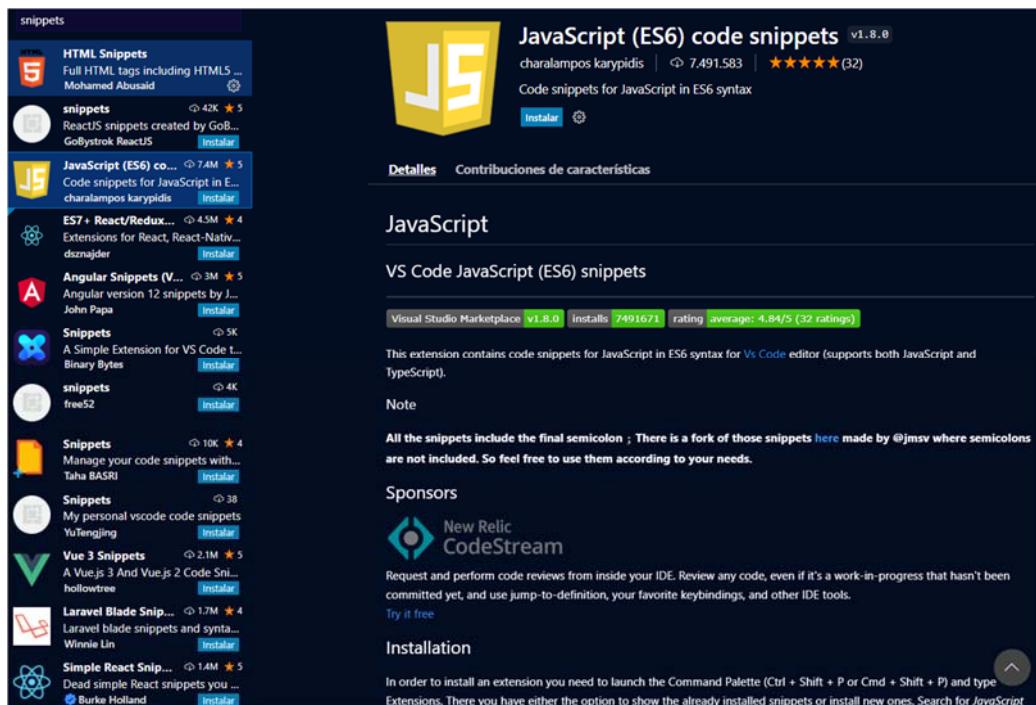
También dispone de una gran cantidad de extensiones para mejorar su código HTML y CSS.

- **CSS Modules:** Puede usar la potencia de Intellisense para mejorar la eficacia de su trabajo en código CSS. Esta extensión ofrece una lista de sugerencias de la etiqueta que quiera escribir y la completa automáticamente. Lo interesante de esta extensión es que diferencia camelCase, esto son las letras mayúsculas que encuentra dentro del código css. La ayuda extra de diferenciar y corregir camelCase ahorrará mucho tiempo en búsqueda de errores por una equivocación puntual al escribir una etiqueta o definición.



Intellisense para código CSS

- **Snippets:** Los snippets son códigos de fragmentos ya configurados listos para insertar en el código. Es una opción muy interesante para agilizar tareas y automatizar escrituras de código que se usa con asiduidad. Puede encontrar infinidad de snippets ya configurados por otros usuarios sobre multitud de funciones para todo tipo de lenguajes: JavaScript, HTML, CSS, Angular, Laravel, React... También puede personalizar los suyos, según sus necesidades. Para ello existen muchísimas extensiones donde encontrar la más adecuada a su proyecto.



Snippets para todo tipo de lenguajes

Más Info

RECURSO MULTIMEDIA

RESUMEN

En este tema ha aprendido a instalar y configurar las diferentes herramientas de edición como son:

- Atom
- Brackets
- Sublime
- NotePad++
- Dreamweaver
- Visual Code

Además, ha visto cada una de las funciones y características de cada una de estas herramientas.

Para ello, ha visto detenidamente cada uno de los elementos que componen estas herramientas.

Además, ha aprendido dónde se ubica cada uno de estos elementos dentro de la propia herramienta.

Para saber cómo funcionan y cómo utilizar estos elementos, ha aprendido todas las propiedades de los mismos.

Asimismo, ha aprendido a realizar las tareas con cada una de las herramientas de edición web.

GLOSARIO

Applet: Un applet es un pequeño programa codificado en lenguaje Java que se ha programado específicamente para ser ejecutado en un navegador.

Códecs: Los códecs son unos programas pequeños que se instalan en un dispositivo para que sea capaz de reconocer un formato codificado de audio o vídeo.

Código abierto: Es un código de una aplicación que está abierto a todo el mundo, es decir, que cualquiera puede modificarlo de manera colaborativa para mejorarlo.

Enlace o vínculo: Los enlaces pueden ser de texto o de imagen y son elementos que hacen referencia a otro recurso del mismo o de otro documento.

Flash: Es una tecnología desarrollada por Adobe que permite crear animaciones gráficas vectoriales que no dependen del navegador. Además, estas animaciones no requieren un ancho de banda elevado para poder ser visualizadas.

Herramienta de desarrollo: Son todas aquellas herramientas que facilitan el trabajo del desarrollo web y ayudan a crear su proyecto de manera especializada.

Hojas de estilo CSS: Una hoja de estilo es aquella que contiene lenguaje CSS y se utiliza para darle una estética concreta a los elementos de un archivo HTML.

JavaScript: Es un lenguaje de programación interpretado que se define como lenguaje de programación orientado a objetos y es, junto con el HTML y CSS, uno de los principales lenguajes que se utiliza en el desarrollo web.

Metainformación: Es una palabra que significa datos acerca de los datos y se utiliza para ofrecer información sobre los propios datos producidos. Por ejemplo, la metainformación de una foto puede ser la fecha en la que ésta se tomó.

MP4: Este formato forma parte del estándar MPEG-4. Es un tipo de formato muy utilizado para vídeo y audio, puesto que, es ligero y tiene calidad de audio.

Scam: Es un término anglosajón que se utiliza para denominar las estafas por medios electrónicos.

SEO: Son un conjunto de acciones y técnicas que se encargan de mejorar el posicionamiento de una web en los buscadores.

Streaming: Esta tecnología se conoce por permitir recibir vídeo y sonido de Internet sin tener que esperar a que el recurso se descargue por completo.

Tarjeta de sonido: Es un dispositivo hardware que puede encontrar en la mayoría de ordenadores. En algunos este dispositivo está integrado en la placa. Sin embargo, puede añadirle una tarjeta de audio de más calidad a su ordenador para poder obtener mejor calidad de audio.

Tipo MIME: MIME significa Multipurpose Internet Mail Extension. Es un mecanismo que permite a los navegadores identificar el formato de un archivo antes de ejecutarlo.

Tipografía: Son los diferentes tipos de letras que puede utilizar en un diseño o escribir.

URL: Es la dirección única y específica que se le asignan a los recursos disponibles que existen en Internet.

Validador: Es un programa que se utiliza para comprobar la validez o corrección sintáctica de un trozo de código o un documento entero.

Versalitas: Este término se utiliza cuando la letra tiene una tipografía donde las mayúsculas y minúsculas no se diferencian.

W3C: Es el World Wide Consortium y es un comité que se dedica a implementar tecnologías uniformes en el uso y desarrollo de Internet.