



Módulo 2: Fundamentos de Desarrollo Front-End

**Desarrollador de aplicaciones
Full Stack**

Python Trainee



MÓDULO 2 – FUNDAMENTOS DE DESARROLLO FRONT-END

2.1.- Aspectos principales del desarrollo en web

2.1.1.- El desarrollo Web

2.1.1.1.- Qué se entiende por desarrollo web

Se entiende al desarrollo web como el conjunto de estándares, técnicas y herramientas que permiten construir y mantener sitios web. Corresponde al trabajo que tiene lugar en un segundo plano y que permite que una web tenga una apariencia impecable, un funcionamiento eficiente y con un desempeño tal que permita la mejor experiencia de usuario. Los desarrolladores web son parte fundamental de un proyecto, ya que permiten que todo esté bien y funcione de manera rápida y eficiente.

Los conocimientos y habilidades vinculados al desarrollo web son muy demandados en la actualidad, ya que todo proyecto existente en la web, independiente del código que pueda tener, hace uso de los conceptos que son parte de esta disciplina.

Un desarrollador web realiza sus labores diarias a través de diversos lenguajes de programación. El lenguaje que usa en cada momento depende del tipo de tarea que está haciendo. El desarrollo web se divide, de forma general, en Frontend (la parte cliente) y Backend (la parte servidor).

2.1.1.2.- Diferencias entre Front-End, Back-End y Fullstack

- **Desarrollador Frontend:** en un proyecto de desarrollo web, este rol se encarga de la composición, el diseño y de la interactividad de un sitio usando el lenguaje HTML, CSS y JavaScript. El desarrollador Frontend toma una idea y la convierte en realidad. Dicho de otro modo, su foco está en “lo que se ve y lo que se usa”, como por ejemplo el aspecto visual del sitio web, los menús desplegables y el texto, los que son creados por el desarrollador Frontend, que escribe una serie de scripts y rutinas para dar estructura, forma e interactividad a estos elementos. Estas rutinas se ejecutan posteriormente a través de un navegador.
- **Desarrollador Backend:** este rol se encarga de lo que no se ve, es decir, la parte de un sistema en dónde se administran los datos. Sin datos no hay Frontend. El Backend consiste en el servidor que acoge la web, una



aplicación para ejecutarlo y una base de datos. El desarrollador Backend utiliza programas de computación para asegurar que el servidor, la aplicación y la base de datos tengan un desempeño regular conjunto. Además, analiza qué necesita la empresa y proporciona soluciones de programación eficientes. Para hacer este trabajo se utiliza una serie de lenguajes del lado del servidor, como PHP, Ruby, Python y Java, entre otros.

- **Desarrollador Full-stack:** es rol aplica para personas que les apasiona tanto el Frontend como el Backend. El desarrollador Full-stack está a cargo tanto del Frontend como del Backend, y necesita saber cómo funciona la web a todos los niveles para determinar cómo se van a coordinar la parte cliente y la parte servidor.

2.1.1.3.- Qué es el lenguaje de marcación de hipertexto (HTML)

La sigla HTML es una abreviatura que viene del inglés “HyperText Markup Language”, y es un lenguaje compuesto por un conjunto de etiquetas definidas con un nombre rodeado de paréntesis “<” y “>”. Los paréntesis angulares delimitan la etiqueta, y el nombre define el tipo de contenido que representa. Por ejemplo, la etiqueta <html> indica que el contenido es código HTML. Algunas de estas etiquetas son declaradas individualmente (por ejemplo,
) y otras son declaradas en pares, considerando una de apertura y otra de cierre, como <html></html> (en la etiqueta de cierre el nombre va precedido por una barra invertida).

Las etiquetas individuales y las de apertura pueden incluir atributos para ofrecer información adicional acerca de sus contenidos (por ejemplo, <html lang="es">). Las etiquetas individuales y la combinación de etiquetas de apertura y cierre se llaman elementos. Los elementos compuestos por una sola etiqueta se usan para modificar el contenido que los rodea o incluir recursos externos, mientras que los elementos que incluyen etiquetas de apertura y cierre se utilizan para delimitar el contenido del documento.

Es necesario incluir varios elementos para definir un documento. Los elementos son listados en secuencia descendente, y pueden contener otros elementos en su interior. Por ejemplo, se dijo anteriormente que la etiqueta <html></html> indica el inicio y cierre de cualquier sitio; por lo tanto, el resto de los elementos que describen el contenido de ese documento se deben declarar entre las etiquetas <html> y </html>. A su vez, las etiquetas dentro del elemento <html> pueden incluir otras más.



El ejemplo indicado a continuación muestra un documento HTML sencillo que incluye todos los elementos necesarios para definir una estructura básica y mostrar el mensaje “HOLA MUNDO!” en un navegador.

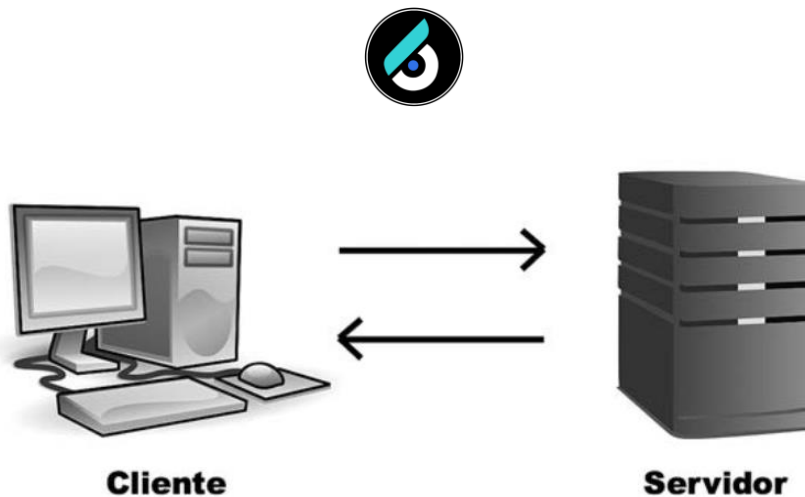
```
1  <!DOCTYPE html>
2  <html lang="es">
3    <head>
4      <title>Mi primer documento HTML</title>
5    </head>
6    <body>
7      <p>HOLA MUNDO!</p>
8    </body>
9  </html>
```

Ilustración 1: Estructura de una página HTML

En el ejemplo anterior se muestra un código que podría resultar sencillo en principio, pero que posee una estructura compleja. En la primera línea se encuentra una etiqueta que señala el tipo de documento existente en las líneas posteriores (<!DOCTYPE html>), seguida por una etiqueta de apertura <html lang="es">. Entre las etiquetas <html> y </html> se incluyen otros elementos que representan la cabecera y el cuerpo del documento (<head> y <body>), los cuales a su vez encierran más elementos con sus respectivos contenidos (<title> y <p>), demostrando cómo se compone un documento HTML. Los elementos se listan uno a continuación de otro y también dentro de otros elementos, de modo que se construye una estructura de tipo árbol con el elemento <html> como raíz.

2.1.1.4.- El rol del navegador

Los sitios web son archivos que los usuarios descargan a través de sus navegadores web desde equipos remotos. Cuando un usuario accede a una página, le indica al navegador la dirección de la misma; el navegador descarga los archivos, procesa el contenido y lo despliega. Dado que los archivos deben estar disponibles todo el tiempo, no pueden ser almacenados en equipos convencionales sino que deben estar dispuestos en equipos especializados. El equipo que hace la consulta se denomina “cliente”, mientras que el que la procesa y envía una respuesta es el “servidor” [1].



Un sitio web es la suma de documentos y recursos multimedia que el navegador descarga cuando el usuario los solicita. Los documentos que forman un sitio web se llaman “páginas” y el proceso de abrir nuevas páginas se conoce como “navegar”.

Si se desea crear un sitio web, es necesario crear un archivo por cada página que se incluirá. Junto con estos archivos, se deben incluir los archivos con las imágenes y cualquier otro recurso que se desplegará en estas páginas; considera que las imágenes y otros medios gráficos se almacenan en archivos aparte.

Los archivos de un sitio web son similares a los documentos que se pueden encontrar en un computador común. Cada uno de ellos tiene un nombre asignado por el desarrollador y una extensión que determina el lenguaje utilizado para crear su contenido. Aunque es posible asignar cualquier nombre a estos archivos, el documento que genera la página inicial presenta algunos requisitos, como por ejemplo que el sitio inicial tenga como nombre “index.html”; sin embargo, esta última condición es configurable de acuerdo a las opciones existentes en el software que actúa como servidor web.

Cuando un sitio ha sido creado, todos los archivos que lo componen son almacenados en un servidor web. Cada servidor se identifica con un valor llamado IP (esta sigla significa “Internet Protocol”). Esta IP es única para cada equipo y, por lo tanto, trabaja como una dirección que permite ubicarlo dentro de una red. Cuando el browser (navegador) web tiene que acceder al servidor para descargar el documento solicitado por el usuario, primero busca el servidor a través de esta dirección IP y luego le pide que le envíe el documento.

En teoría, se puede acceder a cualquier servidor utilizando su dirección IP, pero estos valores son difíciles de recordar. Por lo mismo, Internet usa un sistema que identifica a cada servidor con un nombre específico. Estos nombres personalizados, llamados dominios, son identificadores sencillos que cualquier persona puede recordar, como google o yahoo, con una extensión que determina



el propósito del sitio web al que hacen referencia, como .com (comercial) o .org (organización).

2.1.1.5.- Qué es la W3C

El Consorcio World Wide Web (W3C) [2] es una comunidad internacional donde las organizaciones Miembro, personal a tiempo completo y el público en general trabajan conjuntamente para desarrollar estándares Web. El objetivo del W3C es guiar la Web hacia su máximo potencial a través del desarrollo de protocolos y pautas que aseguren el crecimiento futuro de la Web.

Los principios que guían el trabajo de la W3C son los siguientes:

- **Web para todo el mundo:** El valor social que aporta la Web es que ésta hace posible la comunicación humana, el comercio y las oportunidades para compartir conocimiento. Uno de los objetivos principales del W3C es hacer que estos beneficios estén disponibles para todo el mundo, independientemente del hardware, software, infraestructura de red, idioma, cultura, localización geográfica, o habilidad física o mental.
- **Acceso a la Web desde cualquier dispositivo:** La cantidad de dispositivos diferentes para acceder a la Web ha crecido exponencialmente en los últimos años. Actualmente, los teléfonos móviles, teléfonos inteligentes, PDAs, sistemas de televisión interactiva, sistemas de respuesta de voz, puntos de información e incluso algunos pequeños electrodomésticos pueden acceder a la Web.

La visión del W3C para la Web incluye la participación, el compartir conocimiento y, de esta forma, construir confianza a gran escala. En base a esto se destacan los siguientes puntos:

- **Web de los Autores y Consumidores:** La Web fue creada como una herramienta de comunicación para permitir el intercambio de información entre todo el mundo y desde cualquier lugar. Durante muchos años, para muchas personas la Web fue una herramienta de "solo lectura". Los blogs y wikis trajeron más autores a la Web y las redes sociales emergieron del próspero mercado para crear contenido y personalizar las experiencias en la Web. Los estándares del W3C han apoyado esta evolución gracias a la robusta arquitectura y a los principios de diseño.
- **Web de los Datos y Servicios:** Algunas personas ven la Web como un repositorio gigante de datos enlazados mientras otros la ven como un conjunto enorme de servicios que intercambian mensajes. Ambas vistas son complementarias y los requisitos de cada aplicación pueden ser los



mejores determinantes para decidir qué aproximación elegir para solucionar progresivamente los problemas complejos mediante tecnología Web.

- **Web de Confianza:** La Web ha cambiado la forma en la que se comunican las personas. Al ocurrir esto, la naturaleza de las relaciones sociales ha cambiado también. En la actualidad, las personas se "conocen en Internet", y llevan a cabo relaciones personales y comerciales sin haberse visto en persona anteriormente. El W3C reconoce que la confianza es un fenómeno social, pero el diseño de las tecnologías puede fomentar la confianza y la responsabilidad. A medida que cualquier actividad se hace a través de la Web, cada vez es más importante apoyar las interacciones complejas entre distintas partes alrededor del mundo.

2.1.1.6.- Evolución del HTML hacia el HTML5

A principio de la década de los 90 apareció el lenguaje HTML, y a lo largo de más de 20 años ha ido evolucionando llegando a día de hoy a la versión 5. A continuación se indicarán las diferencias entre la versión de HTML inicial y la versión HTML5, a fin que se pueda analizar la evolución que ha sufrido el lenguaje HTML desde su primera versión hasta la más reciente.

HTML 1.0

HTML es una tecnología ya antigua, cuya primera versión apareció en 1991, y es una especificación que fue creada por Tim Berners-Lee. Surgió en el CERN, el cual es el Laboratorio de Física Nuclear de Ginebra, y cuyo objetivo principal es sus inicios, única y exclusivamente, era compartir información científica en la comunidad académica.

Está basado y deriva de SGML, un sistema general de marcado, por lo que surgió como un lenguaje de marcas inicialmente muy pequeño, ya que solamente tenía 20 etiquetas, de las cuales 13 aún perduran.

HTML sirve para hacer páginas web, por lo que junto con la primera especificación se creó el primer navegador, que se llamó Nexus.

Durante todos los años posteriores han ido apareciendo distintas versiones de HTML, que han ido añadiendo nuevas etiquetas, nuevas propiedades, permitiendo añadir estilos, etc., hasta llegar a la versión de HTML que actualmente está vigente, que es la versión HTML5.

HTML 5.0



HTML5 sigue siendo recomendación, ya que aún no es un estándar al 100%. Esta versión fue publicada en 2014, pero es una especificación que llevaba haciéndose más de 15 años, por lo que ha ido cambiando mucho.

Se hizo de esta manera para solucionar los problemas que tenía HTML en aplicaciones complejas, por lo que introduce muchísimas novedades, de las cuales las más importantes son las siguientes:

- Introduce etiquetas semánticas para facilitar la vida a los desarrolladores, a los buscadores y a todo tipo de dispositivos para personas con discapacidad. Al utilizar etiquetas semánticas se están dando pistas del contenido que va en cada parte de la página web.
- Añade APIs, que no son APIs en HTML, sino desarrolladas en un lenguaje de programación como JavaScript, que permiten, basadas en etiquetas HTML, poder desarrollar juegos completos, geolocalizar páginas, trabajar con bases de datos ligadas a etiquetas HTML de una página web, hacer cosas como arrastrar y soltar en la página web de manera muy fácil, etc.
- Simplifica el uso de las etiquetas.
- Mejora el uso de formularios y, sobre todo, de su validación, ya que atributos de las etiquetas van a permitir que se valide un formulario, dependiendo de la información que se haya introducido, antes de poder enviarla al servidor.

Al ser una recomendación, si se desea saber si el navegador soporta HTML5 completamente o no, es posible usar una herramienta muy recomendable como es la que ofrece la web <http://html5test.com> [3]. Esta herramienta indica hasta qué punto el navegador está soportando esta nueva especificación, que todavía no es un estándar y tiene una cantidad importante de opciones.

2.1.1.7.- La triada HTML, CSS y JavaScript: contenido, presentación y comportamiento

Tal como se ha visto hasta ahora, HTML es un lenguaje de maquetación, vale decir, se utiliza para definir la estructura base del sitio. Esto se logra por medio de las etiquetas que el lenguaje posee. Además, se debe considerar que se basa en la construcción de sitios estáticos, ya que la misma información que está almacenada en el servidor es la que recibirá el navegador, sin cambio alguno en la estructura. Esto es básico a fin que el resultado que se obtenga al visualizar un sitio web en un navegador sea exactamente el mismo que en otro navegador.



Una vez que se posee la estructura del sitio, se debe asignar colores, formas, tamaños y posición sobre los elementos que componen dicha estructura. En los inicios del lenguaje HTML, este tipo de características se aplicaba en el mismo código, pero con el tiempo esto fue mutando hasta que fue parte de manera íntegra por otro lenguaje: las hojas de estilo en cascada, conocidas también como CSS. Esto permite al desarrollador separar la estructura del sitio de sus atributos de diseño, generando orden y escalabilidad, entre otras cosas.

Ya establecida la estructura del sitio y el diseño que se desea aplicar, en determinadas situaciones se necesita generar interacción entre el usuario y los elementos de la web, ya sea validación de formularios, desplazamiento de elementos del sitio, cambio de formas o efectos atractivos, solo por mencionar algunas posibilidades. Todo esto y mucho más se puede lograr con JavaScript, el cual es un lenguaje de programación que se ejecuta a nivel del navegador, y permite lograr estos efectos.

Como se puede apreciar, todos estos elementos trabajan juntos en el logro de un objetivo superior. Ninguno de ellos es más importante que el otro, y si bien su estudio y avance puedan ir por caminos separados en las aplicaciones adicionales que existen, todas las mejoras que se puedan implementar sobre los estándares que las rigen toman sentido en el trabajo conjunto.

2.1.2.- El Entorno de Desarrollo

2.1.2.1.- Descarga del editor Visual Studio Code e instalación

Visual Studio Code es un editor de código fuente creado por la empresa Microsoft, desde una perspectiva multiplataforma. Actualmente se encuentra disponible para los sistemas operativos Windows, Linux y macOS [4].

Esta aplicación además permite ejecutar operaciones de desarrollo, tales como la depuración de código, la ejecución de tareas y el control de versiones. Su objetivo es proporcionar solo aquellas herramientas que un desarrollador necesita para un ciclo eficiente de creación de código y depuración. Los flujos más complejos, en tanto, se dejan en manos de entornos de desarrollo integrados (IDE) más complejos.

Una característica interesante de este software es que su uso es liberado, tanto para proyectos personales o comerciales. Para descargarlo, solo se debe acceder al enlace incluido en las referencias.



Para descargar este programa, se puede hacer ingresando a la dirección <http://code.visualstudio.com>. Una vez dentro del sitio, se presiona el botón “Download” ubicado en el extremo superior derecho de la página, con lo cual se ingresará a una página en la que se debe seleccionar el sistema operativo correspondiente. Descargado el ejecutable al computador, se debe hacer doble clic sobre éste, con lo cual se desplegará el asistente de instalación.

2.1.2.2.- Utilizar el potencial de un editor de texto para el desarrollo

En términos generales, un editor de texto es un tipo de software diseñado para crear, modificar y almacenar archivos de texto. Se diferencia de los procesadores de texto tradicionales por la ausencia de opciones de formato avanzadas y de la posibilidad de mostrar elementos gráficos o multimedia. Casi todos los sistemas operativos masivos cuentan con un editor de texto, además de los editores de texto creados por terceros.

Entre las ventajas que considera una herramienta de esta naturaleza, se destacan las siguientes:

- **Multiplataforma:** cada sistema operativo cuenta con un conjunto de programas de escritorio que permiten editar archivos de texto, algunos incorporados y otros creados por entidades externas. Independiente de esta diversidad, situaciones tales como el manejo de saltos de líneas se encuentra estandarizado, mejorando la visualización y manipulación.
- **Integridad de los datos:** un editor de texto por lo general no agrega caracteres extraños provenientes de formatos o elementos gráficos, permitiendo generar archivos de texto plano. Para codificar sitios web, es fundamental que las etiquetas incorporadas sean incluidas como texto plano, ya que cualquier formato adicional puede alterar la integridad del archivo, generando errores en su visualización en un navegador.
- **Tamaño de archivos:** otra ventaja de los editores de texto es la posibilidad de manipular archivos de gran tamaño, situación que no se aplica a procesadores de texto habituales. Esto incluye tanto la edición de los documentos como la búsqueda de términos.

2.1.2.3.- Conociendo el inspector de elementos en un navegador

Un navegador web es una aplicación de software que comunica un dispositivo cliente (PC, notebook, celular, Tablet, etc.) con un servidor web, al cual se le solicita parte de la información que tiene almacenada. Si lo analizamos en términos simples, una página web no es más que un conjunto de archivos de texto

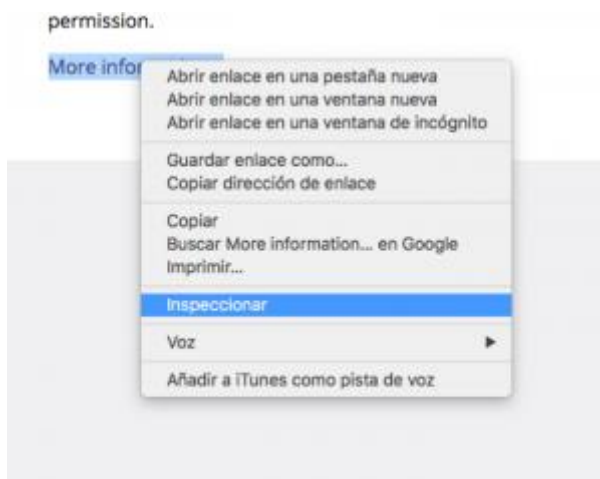


plano y documentos multimedia que son interpretados por un navegador, brindándole forma.

Los navegadores más usados en la actualidad son Microsoft Edge, Google Chrome, Mozilla Firefox y Opera [2]. Si bien todos ellos cumplen un objetivo específico y transversal, presentan características que los diferencian de sus pares, y que traen como consecuencia que los usuarios tengan un navegador de preferencia.

Una herramienta que es transversal en los navegadores conocidos es el “inspector de elementos”, la cual es permite analizar los contenidos de cualquier sitio web. A través de esta aplicación se puede ver el código fuente de la página, así como analizar cada uno de los componentes que la conforman. Es de mucha utilidad cuando se quiere ver cómo está hecho algo en un sitio, o bien para afinar estilos y depurar scripts que se ejecuten.

Cada inspector de los navegadores tiene diferencias menores, pero las acciones básicas a desarrollar son prácticamente las mismas. En las imágenes siguientes se usará como referencia el navegador Google Chrome. Para acceder al inspector de elementos debe seleccionar el objeto a analizar, y con el botón derecho del ratón seleccionar la opción “Inspeccionar”.



El inspector se compone de varias pestañas, donde cada una determina distintas funcionalidades y características:





- **Elements:** se puede revisar el código fuente del sitio, junto a los estilos que se aplican sobre ella. Se pueden modificar localmente y añadir nuevas reglas.
- **Console:** en esta sección se muestran los mensajes de error y de advertencias que se producen en la página (imágenes que no cargan correctamente, errores de Javascript, entre otros).
- **Sources:** recursos de la página, expresados en forma de árbol. Se puede analizar su origen y modificarlos de manera local.
- **Network:** esta ventana permite analizar cada una de las peticiones que se hacen a un servidor web, considerando en ello el contenido de la petición, la respuesta que se obtuvo y los tiempos asociados.
- **Performance:** herramientas que permiten medir el desempeño de un sitio.
- **Memory:** mide la memoria que se consume durante el proceso de carga del sitio y durante la ejecución de las tareas respectivas.
- **Application:** datos de utilidad sobre los recursos que emplea una aplicación web.
- **Security:** información sobre los sitios que están enlazados al portal analizado, y el estado de su certificado de seguridad respectivo.
- **Audits:** contiene una serie de controles predeterminados que permiten validar errores típicos.



Anexo: Referencias

[1] J.D Gauchat “El Gran libro de HTML5, CCS3 y JavaScript”
3ª edición

[2] Sobre el W3C
Referencia: <https://www.w3c.es/Consortio/>

[3] HTML5 Test
Referencia: <http://html5test.com/>

[4] Sitio oficial de Visual Studio Code
Referencia: <https://code.visualstudio.com/>