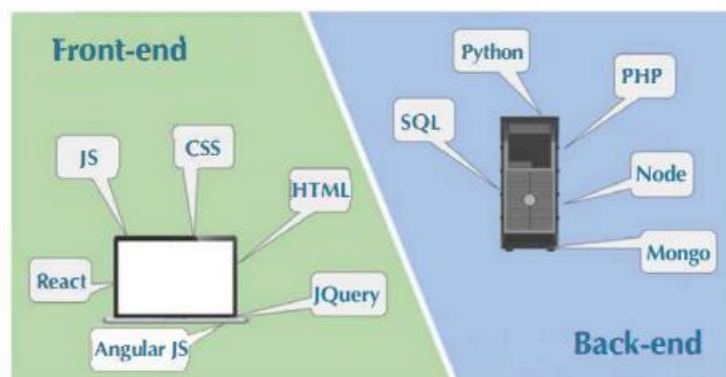


ANEXO 2

Front-end y Back-end

Actualmente, se suele catalogar el desarrollo web en dos partes **back-end** (la parte no visible de la web, como las bases de datos o los scripts que se ejecutan en el servidor) y **front-end** (la parte visible de una web, como las hojas de estilo, el código HTML, los scripts que se ejecutan en el lado del cliente, etc.). A continuación, se describirá con más profundidad este modelo cliente servidor.



- La parte del **back-end** se encarga de todo el proceso en el **lado del servidor**, como el acceso a la base de datos (MySQL, MaríaDB, PostgreSQL, MongoDB, Oracle, etc.), creación de servicios, etc. Estos técnicos programarán en lenguajes como **PHP, Ruby on Rails, Django, Node.js, .NET, etc.**
- En la parte del **front-end**, prima la parte creativa y la originalidad, puesto que el perfil es más cercano al diseñador, aunque también se trabaja en el código. La programación de la interfaz se llevará a cabo en lenguajes como **HTML, CSS, JavaScript, etc.**
- Actualmente, con **frameworks** como **AngularJS, ReactJS y otros**, la parte cliente o front-end está ganando terreno a la parte servidora. En muchas ocasiones, estos frameworks están diseñados para almacenar estructuras de datos relegando al back-end o simplemente para asegurar la persistencia de estos. Entre las **ventajas** que aportan los frameworks, están:
 - **El coste.** Muchos de estos frameworks **son de código abierto**, con lo cual no hay que realizar ninguna inversión.
 - Están **suficientemente probados** y su código **suele carecer de errores**, puesto que muchos programadores lo utilizan. Además, suelen tener un alto nivel de seguridad y rendimiento.
 - **Permite desarrollar mucho más rápido**, puesto que muchas de las estructuras, **clases, patrones de diseño, etc.**, vienen ya incorporadas en el framework.
 - Cualquier persona que maneja un framework determinado puede entender e incorporarse de forma eficiente y efectiva a un equipo de desarrollo que lo esté utilizando en un proyecto determinado.

Lenguajes de programación en entorno cliente

La programación web en el lado del cliente se basa en **tres pilares fundamentales** que se citan a continuación:

1. **El lenguaje HTML.** No es un lenguaje de programación, sino un **lenguaje de marcado**. El HTML **define el contenido que va a tener el documento**. La función del **navegador web** será la de **leer e interpretar** todo este contenido, junto con las etiquetas, **y visualizarlo al cliente**. La ventaja del código HTML es que cualquier navegador debería visualizar el contenido de la página web de la misma forma y con el mismo aspecto.
2. **El lenguaje CSS.** Define la **presentación del documento**. CSS es un **lenguaje de diseño gráfico** y su objetivo es que la página web sea atractiva al usuario. No modifica el comportamiento ni el contenido, sino el aspecto de la página web.
3. **El lenguaje JavaScript.** El código o lenguaje JavaScript **agrega el contenido dinámico a las páginas web**. JavaScript es un verdadero lenguaje de programación, a diferencia de los lenguajes HTML y CSS.

Actualmente, las empresas no programan directamente sobre JavaScript, sino basándose en un framework de este. A continuación, se citarán algunos de los frameworks más utilizados para JavaScript.

- A. **ReactJS** (<https://reactjs.org/>) es un **framework** creado por **Facebook** que permite a los programadores **realizar aplicaciones web de una forma rápida y eficiente renderizando (dibujando) los componentes del front-end** de una manera **sencilla y eficaz**. React utiliza **programación orientada a componentes** (que no objetos). Los componentes gestionan su propio estado y, cuando se agrupa una serie de componentes, los programadores son capaces de ir creando las interfaces de usuario. Una de las características de React es que **utiliza un DOM virtual** que mapea los objetos desde este hasta el DOM del navegador.
- B. **Angular** (<https://angular.io>) fue **creado** y es mantenido **por Google**. La primera versión de Angular se denominó AngularJS y todavía hay una comunidad utilizando este framework por su **fácil integración con JavaScript**. Las versiones sucesivas de Angular se denominan Angular a secas y ya a dchejado de ser una simple librería para pasar a ser una plataforma de desarrollo. El problema con este y otros frameworks es que su **curva de aprendizaje** es bastante **pronunciada**, dado que no son fáciles de aprender. Actualmente, en **Angular, se programa en TypeScript**, que es un **superconjunto de JavaScript** desarrollado por **Microsoft** y utiliza el patrón reactivo RxJS.
- C. **Vue.js**. Una de las características de este **framework** (<https://vuejs.org>) frente a otros es la **ligereza** y la **velocidad de ejecución**. El objetivo que se plantearon sus desarrolladores y diseñadores fue el crear un framework con las mejores ventajas de los existentes. A diferencia de Angular, su curva de aprendizaje no es tan pronunciada y los desarrolladores de **Laravel** (un framework de back-end) lo utilizan **para** usarlo en el

Unidad 1: Selección de la arquitectura y herramientas de programación

front-end de sus aplicaciones. Al igual que React, utiliza un DOM virtual, dada las ventajas que ofrece este tipo de implementaciones.

Características de los lenguajes de script

Hubo un tiempo en el que los lenguajes de programación se utilizaban para crear programas de nóminas, contabilidad, edición de texto, hojas de cálculo, gestión de almacenes, etc., en los que se realizaba un análisis, un diseño de la aplicación y luego se pasaba a codificar todo lo desarrollado en fases anteriores. Los programas se ejecutaban en un equipo standalone o en entornos cliente servidor.

Desde hace mucho tiempo, ese tipo de necesidades han cambiado, puesto que la conectividad total y las nuevas necesidades de las empresas y clientes son distintas. Ahora se necesitan aplicaciones que se puedan ejecutar sobre un navegador web o aplicaciones para dispositivos móviles como smartphones o tabletas.

Ya no se desarrolla todo desde cero, sino que, en muchos casos, se utiliza un sistema host -como puede ser el navegador- para integrar pequeños fragmentos de código que aporten el aspecto dinámico a las páginas web visualizadas.

Los scripts nacieron como fragmentos de código que realizaban ciertas tareas o rutinas concretas. En los sistemas operativos, los scripts se usan para automatizar tareas y siempre van a ser ejecutadas por un intérprete de comandos (por lo tanto, los scripts siempre van a ser interpretados).

Actualmente, los scripts no son pequeños fragmentos de código, sino que pueden considerarse auténticos programas. Véanse algunas de las diferencias entre los lenguajes de script y los lenguajes de programación:

- Los lenguajes de script son interpretados, mientras que, muchas veces, los lenguajes de programación se compilan.
- Los lenguajes de script utilizan componentes ya preexistentes, mientras que los lenguajes de programación, en ocasiones, empiezan a desarrollarse desde cero.
- A veces, los scripts se incrustan dentro de otros programas (como el JavaScript se integra con el HTML).
- Los programas de lenguajes de programación se pueden ejecutar de forma independiente, mientras que un lenguaje de script se ejecuta muchas veces dentro de otro programa.
- Los scripts se ejecutan línea a línea, con lo cual pueden producirse muchos errores en ejecución.
- Los lenguajes de script no generan un fichero ejecutable.
- Los lenguajes de script no necesitan ser compilados.
- Los lenguajes de script han sido diseñados para que sean fáciles de utilizar y programar. Algunos lenguajes de programación son: Java, C, C++, Swift, Pascal,

Unidad 1: Selección de la arquitectura y herramientas de programación

etc. Algunos lenguajes de scripting son: JavaScript, Shell, Perl, PHP, Python, Ruby, etc.

Integración de JavaScript dentro de HTML

Como ya se ha explicado, JavaScript se combina o complementa al código HTML de una página web. Existen dos opciones:

1. Integrar el código JavaScript dentro de los archivos HTML.
2. Tener separado del HTML el código JavaScript en archivos con extensión js.

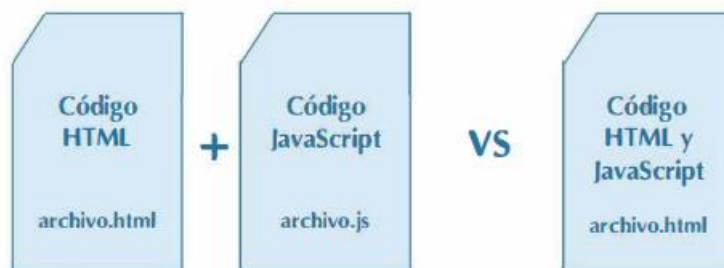


Figura 1.2
Código JavaScript dentro y fuera de un archivo HTML.

Lo más limpio y eficaz es tener el código JavaScript fuera de los archivos HTML por las ventajas que ofrece.

EJEMPLOS

- Ejemplo de JavaScript con código dentro del HTML

```
<!DOCTYPE html>
<html>
<title>Myfpschool</title>
<head>
<script>
  function diAlgo()
  {
    alert("Hola");
  }
</script>
</head>
<body>

<script>
diAlgo();
</script>

</body>
</html>
```

Unidad 1: Selección de la arquitectura y herramientas de programación

- Ejemplo de JavaScript en ficheros js separados

```
<!DOCTYPE html>
<html>
<title>Myfpschool</title>
<head>
<script src="script.js"></script>
</head>
<body>
</body>
</html>
```

A continuación, se muestra el contenido del archivo *script.js*:

```
function diAlgo()
{
    alert("hola");
}
diAlgo();
```

Herramientas de programación en JavaScript

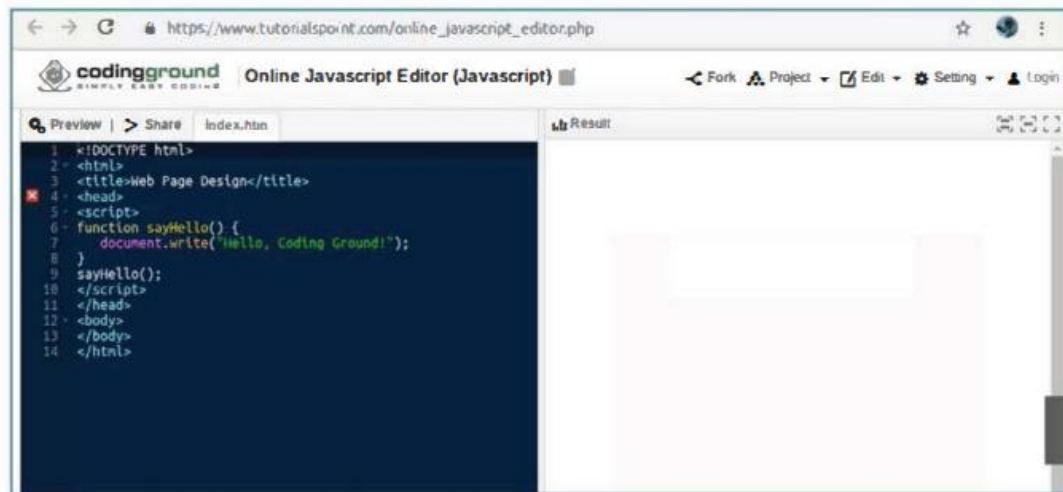
Existen múltiples alternativas a la hora de elegir una herramienta de programación. Una de las ventajas de programar en JavaScript es que puede servir un simple editor de texto como Bloc de Notas. Esto no es lo deseable cuando se desarrolla un proyecto y, por lo tanto, las empresas y profesionales suelen utilizar otras alternativas como los **entornos de programación online** o herramientas de programación con **sistemas de control de versiones**. A continuación, se mostrarán dos alternativas de herramientas de programación.

1. Herramientas online

Las **ventajas** que ofrece un **IDE (entorno integrado de desarrollo)** online son evidentes, se puede **ejecutar y probar código desde cualquier dispositivo** solamente teniendo **acceso a internet**. Existen muchas herramientas en el mercado y una de ellas es:

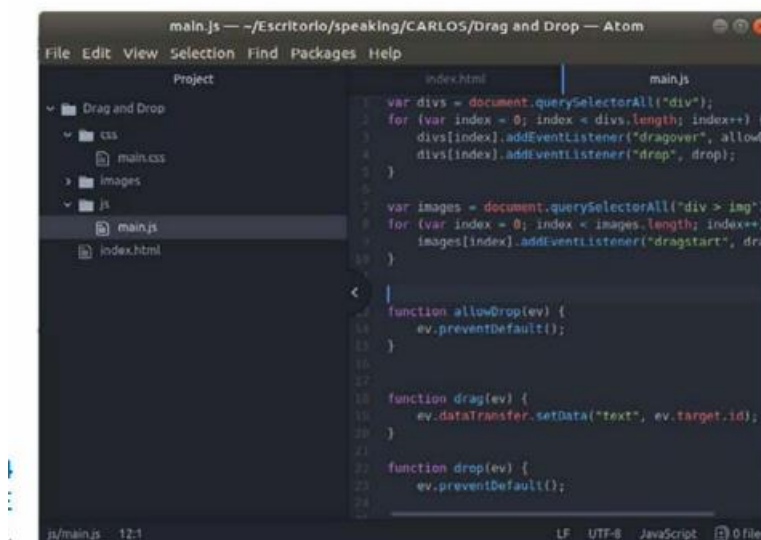
- Coding Ground de Tutorialspoint (<https://www.tutorialspoint.com/online-javascript-editor.php>). Esta herramienta no solo permite gestionar los distintos ficheros de un proyecto, sino que se puede descargar o agregar cualquier fichero que haga falta.

Unidad 1: Selección de la arquitectura y herramientas de programación



2. Utilización de IDE y sistemas de control de versiones

Otra de las opciones es utilizar 11n IDE complementado con un sistema de control de versiones. Una alternativa puede ser utilizar Ato1n (<https://atom.io>). Ato111 está disponible para plataformas Lin11x, Windows y Mac OS X y es 11na herramienta libre, gratuita, ligera y potente.



A continuación, se citan algunas ventajas de este IDE:

- Es open-source, lo que quiere decir que, si se desea hacer modificaciones o arreglar el código fuente, está disponible para todo el público. Muchas veces los programadores no van a modificar el código, pero al ser software libre se

Unidad 1: Selección de la arquitectura y herramientas de programación

aseguran de que **una comunidad** de usuarios se encargará de ello, haciendo que los **errores y mejoras se corrijan** mucho antes que con un software propietario.

- **Es gratuito.**
- Tiene una gran comunidad detrás de la herramienta, con lo cual cualquier problema o inquietud puede ser resuelta en Slack o incluso en Twitter.
- **Es modular.** Se pueden deshabilitar ciertas funciones y reemplazar por otras del gusto del programador.
- **Tiene integrado un package manager.** Se pueden instalar paquetes y temas con un comando apm desde el terminal.
- **Está integrado con GitHub.** Atom está pensado para utilizarlo conjuntamente con este sistema tan famoso de control de versiones.
- El sistema de paneles de Atom es muy apreciado por los programadores.
- **Tiene un sistema muy bueno de autocompletado.**

NOTA - Como alternativa a Atom, se puede utilizar:

- **Sublime text** (<https://www.sublimetext.com/>)
- **VisualStudio** (<https://code.visualstudio.com/>).

Posibilidades que ofrece JavaScript

A continuación, se mostrarán varios ejemplos básicos de qué se puede hacer con JavaScript.

1. Modificación del contenido de una página web

En el siguiente código, se muestra cómo se puede modificar el contenido HTML de forma dinámica:

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

<h1>Modificando el código HTML</h1>

<p id="prueba">Modificando el contenido.</p>

<button type="button"
onclick="document.getElementById('prueba').innerHTML = 'CAMBIANDO el
contenido!'">
¡Dale!</button>

</body>
</html>
```




2. Cambiar atributos de objetos HTML

En el siguiente ejemplo, se va a modificar de forma dinámica el atributo src de un objeto tipo img en un código HTML. Tras copiar el siguiente código en un archivo de texto, puede observarse cómo funciona:

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

<h1>Cambio de imágenes con JavaScript</h1>



<p>Haz click sobre las letras para cambiarlas.</p>

<script>
function cambiaPic() {
    var image = document.getElementById('myFPImage');
    if (image.src.match("green")) {
        image.src = "http://myfpschool.com/wp-content/uploads/2016/06/myblack.jpeg";
    } else {
        image.src = "http://myfpschool.com/wp-content/uploads/2016/06/mygreen.jpeg";
    }
}
</script>

</body>
</html>
```

El aspecto de la página web es el que se muestra en la figura 1.6.



Unidad 1: Selección de la arquitectura y herramientas de programación

Se puede observar que, cuando se hace clic sobre las letras MY, estas cambian de negro a verde, y viceversa, porque, en el script anterior, comprueba qué imagen se está visualizando

ACTIVIDAD:

Copia el código del ejemplo de este apartado y comprueba si funciona en un navegador. En esta ocasión, utiliza tus propias imágenes.

3. Cambiar el estilo CSS

Con JavaScript no solamente se puede cambiar el contenido, sino que también se puede cambiar el estilo (CSS) de cualquier elemento de la página HTML. Para ver cómo se puede cambiar el aspecto, se va a generar una página cuyo aspecto sea:



y, una vez que se pulse un botón, cambie el aspecto del párrafo para parecerse:



```
<!DOCTYPE html>
<html>
<body>
<p id="mytxt">Aprende JavaScript con MyFPSchool!</p>
<button type="button" onclick="myFunction()">¡Dale!</button>
<script>
function myFunction() {
    var x = document.getElementById("mytxt");
    x.style.fontSize = "25px";
    x.style.color = "red";
}
</script>
</body>
</html>
```

Como se puede observar, en la función myFunction, se selecciona el párrafo que se quiere modificar y se cambia su color a rojo y su tamaño a 25 píxeles

4. Comunicación de JavaScript con el exterior.

Existen varias **opciones** para que el código JavaScript se comuniquen con el usuario. A continuación, se detallan cada una de ellas.

a. Escribir en la consola del navegador utilizando console.log()

- Esta opción **la utilizan** solamente por **los desarrolladores**. Ningún usuario suele acceder a la consola para ver el contenido escrito en ella.
- Para acceder a la consola, basta con pulsar F12 y elegir Consola en su navegador
- Ejemplo:

```
<script>
    console.log("Síntesis, ejemplo de consola");
</script>
```

b. Escribir en cualquier elemento HTML utilizando el atributo innerHTML

1. A continuación, se muestra un ejemplo de un script que utiliza el atributo **innerHTML** para **cambiar** el contenido de **un elemento HTML**:

```
<p id="parrafito"></p>
<script>
    document.getElementById("parrafito").innerHTML = 5 + 6;
</script>
```

c. Generar directamente HTML utilizando el método document.write()

2. Con el método write se puede generar código HTML directamente. Se puede también añadir etiquetas en la llamada al método si se desea.

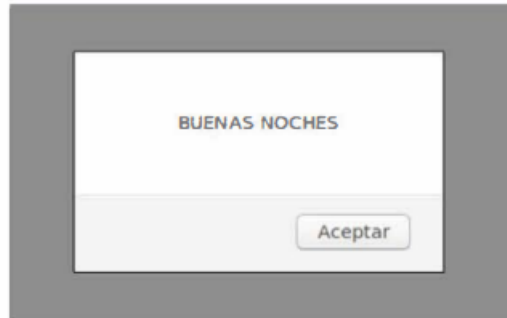
```
<script>
    document.write("<h2>Buenos días</h2>");
</script>
```

d. Generar un mensaje de alerta utilizando el método window.alert()

3. Con el método window.alert () o simplemente alert (), se puede mostrar un diálogo en1er gente en el navegador. A continuación se presenta un ejen1plo de utilización del método window. alert () dentro de un código JavaScript:

Unidad 1: Selección de la arquitectura y herramientas de programación

```
<script>
    window.alert("BUENAS NOCHES");
</script>
```



Resumen

- W3C es el que se encarga de desarrollar los estándares para que el desarrollo de internet funcione a largo plazo.
- Los frameworks nacieron como librerías, más o menos completas, que tenían una serie de estructuras que permitían al programador disponer de una base para la creación y el desarrollo de sus proyectos.
- Los técnicos de back-end se encargan de todo el proceso en el lado del servidor, como el acceso a la base de datos.
- En la parte del front-end, prima la parte creativa y la originalidad, puesto que el perfil es más cercano al diseñador, aunque también se trabaja en el código.
- ReactJS es un framework creado por Facebook.
- Angular fue creado y es mantenido por Google.
- En Vue.js, prima la ligereza y la velocidad de ejecución.
- Los scripts nacieron como fragmentos de código que realizaban ciertas tareas o rutinas concretas.
- En JavaScript, se puede colocar el código dentro y fuera del HTML, aunque se aconseja colocarlo fuera.
- Entre las posibilidades que ofrece JavaScript, están:
 - Modificar el contenido de una página web.
 - Cambiar atributos de objetos HTML.
 - Cambiar el estilo CSS.
- JavaScript se puede comunicar con el exterior mediante:

Unidad 1: Selección de la arquitectura y herramientas de programación

- `Console. log()`.
- Mediante el atributo `innerHTML`.
- `document. write()`.
- `window.alert()`.

Actividades

1. Elabora una secuencia de imágenes con al menos 4 fotogramas. Según el usuario vaya haciendo clic sobre las imágenes, la secuencia se irá reproduciendo.
2. Realiza un test de 7 preguntas donde cada pregunta tenga dos botones (verdadero y falso). Dependiendo de si el usuario hace clic en verdadero o en falso la frase se pone verde (acierto) o roja (error).
3. Desarrolla una página web en la que al pulsar un botón se genere un mensaje por consola.
4. Diseña una web con tres botones para que al pulsarlos se genere un mensaje de bienvenida en ruso, español e inglés en la misma página. El mensaje irá en un párrafo (etiqueta). Se pide que los mensajes tengan un color diferente dependiendo del idioma.
5. Modifica el código anterior para que los mensajes de bienvenida se generen con alertas.
6. Cambia el código anterior para que los mensajes de bienvenida se generen en consola.
7. Transforma el código anterior para que los mensajes de bienvenida se generen con el método `write()`.
8. ¿En qué se diferencia JavaScript de Java?
9. Describe las ventajas más importantes de usar JavaScript.