

DWEC

TEMA 1



INTRODUCCIÓN



INTRODUCCIÓN

Evolución de la web

-Web 1.0: en los inicios, los internautas sólo tienen un rol: consumir la información que se aloja en servidores informáticos. La navegación era netamente textual y las consultas son muy limitadas. Cuando aparece el lenguaje de programación web HTML (Hyper TextMarkup Language), se hace más atractiva la organización de los elementos que se visualizan en la pantalla.

INTRODUCCIÓN

-Web 2.0: Alrededor del año 2000 se empieza a hablar de la Web 2.0, al fomentarse la interacción entre páginas web y usuarios. Internet se entiende como una plataforma colaborativa, en la que todos los usuarios participan. Ahora, además de leer, los internautas también pueden generar informaciones y publicirlas en sitios web, en foros de usuarios, blogs, redes sociales y wikis (páginas editables por cualquier usuario).

INTRODUCCIÓN

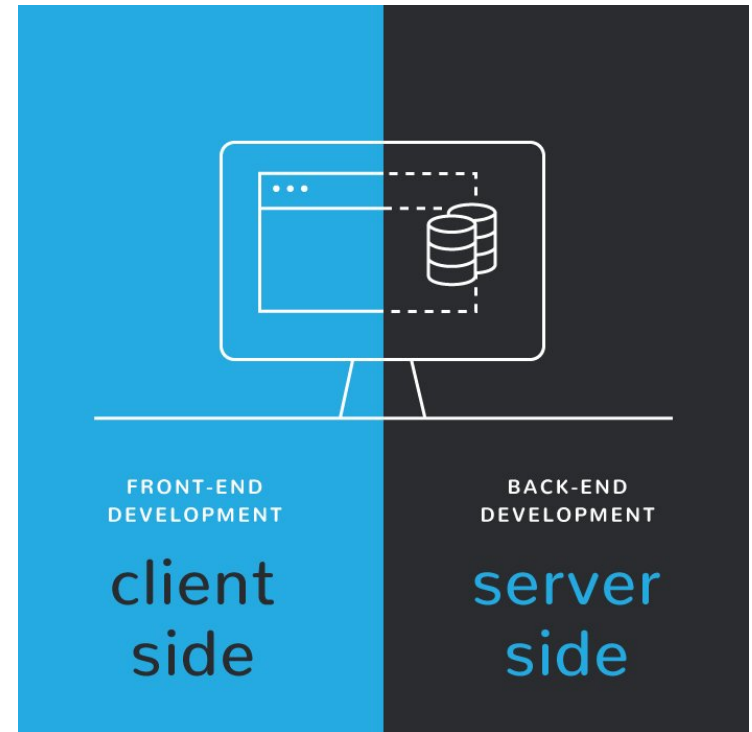
-Web 3.0: Se refiere a que las páginas web se pueden relacionar de manera semántica, añadiendo metadatos que aportan valor a la información y mejora la búsqueda y la posibilidad de encontrar información en la web al comprenderse el significado de las palabras.

INTRODUCCIÓN

-Web 4.0: Internet puede funcionar de manera predictiva, ya no solo recibiendo órdenes de parte del usuario, sino que mediante el empleo de la inteligencia artificial se puede anticipar a sus solicitudes e incluso predecir comportamientos en el futuro. Se generalizan también las interacciones mediante la voz, con el uso de dispositivos inteligentes que permiten realizar búsquedas habladas.

FRONT-END Y BACK-END

Actualmente se suelen catalogar el desarrollo web en dos partes (la parte no visible de la web, como las bases de datos o los scripts que se ejecutan en el servidor) y el 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).



FRONT-END Y BACK-END

BACK-END:

En las empresas, los técnicos especialistas en back-end se encargan de todo el proceso en el lado del servidor, como el acceso a las bases de datos:

MySQL, MariaDB, PostgreSQL, MongoDB, etc.

Y programan en lenguajes como PHP, Django, Node.js, .NET, etc.

Frontend



FRONT-END Y BACK-END

FRONT-END:

En la parte del front-end, prima la parte creativa y la originalidad, puesto que el perfil es mucho más cercano al diseñador, aunque también se trabaja en el código. La programación del interfaz se llevará a cabo en lenguajes como HTML, CSS, JavaScript, etc. Actualmente los Frameworks como AngularJS, ReactJS y VUE están ganando terreno a la parte servidor.

Frontend



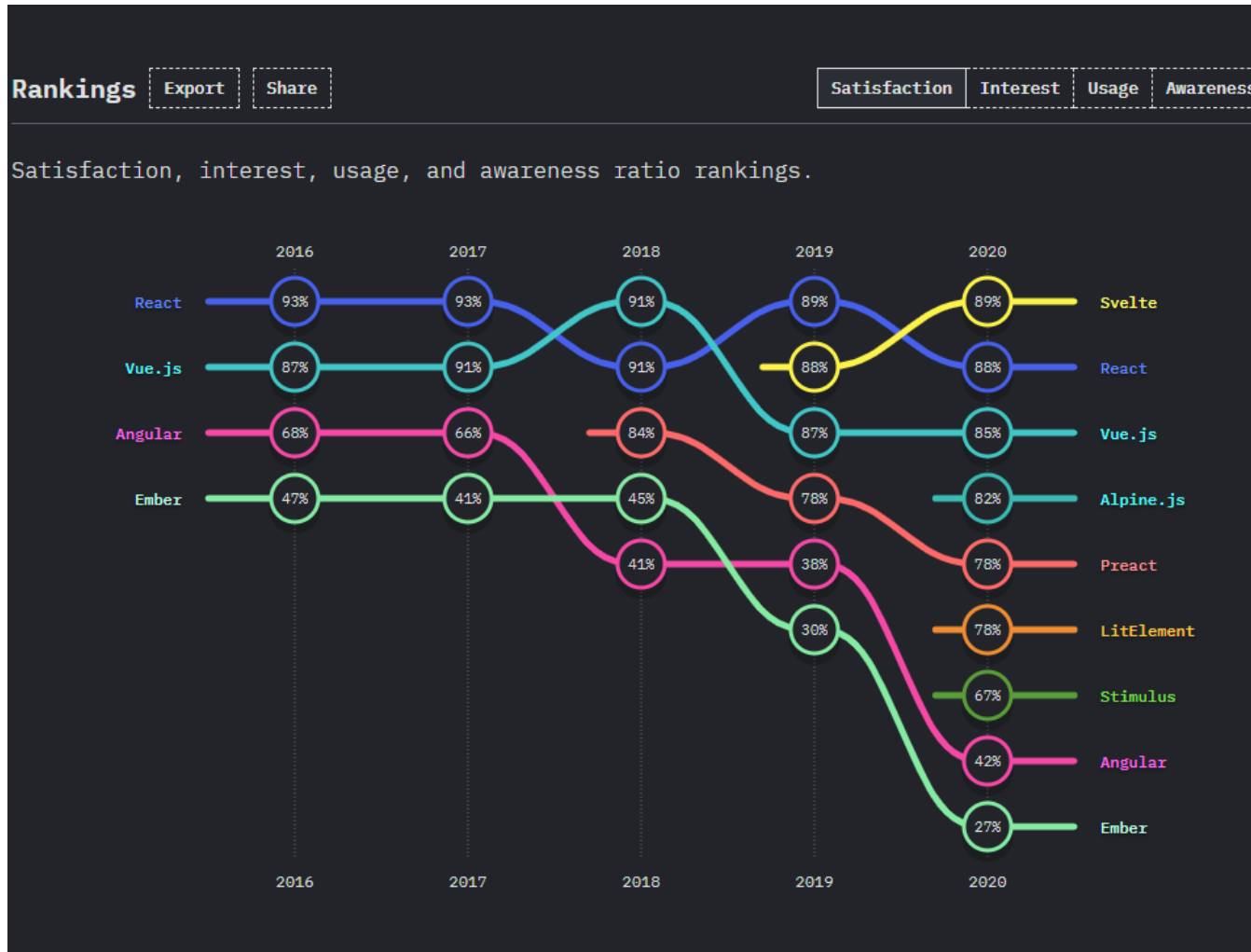
QUÉ ES UN FRAMEWORK?

Los frameworks nacieron como librerías, más o menos completas, que tenían una serie de estructuras que permitían al programador tener una base para la creación y desarrollo de sus proyectos. Actualmente, son mucho más que eso, puesto que pueden utilizar lenguajes como TypeScript o JSX, los cuales luego se compilan a JavaScript. Sus ventajas son:

- Su coste: La mayoría son de código abierto.
- Están muy probados y su código carece de errores. Suelen tener alto nivel de seguridad y rendimiento.
- Permite desarrollar mucho más rápido ya que tiene muchas estructuras y clases ya creadas.
- Si una persona ya maneja un framework puede entender y incorporarse a un equipo de desarrollo.



TOP 10 FRAMEWORKS



FRAMEWORK ANGULARJS

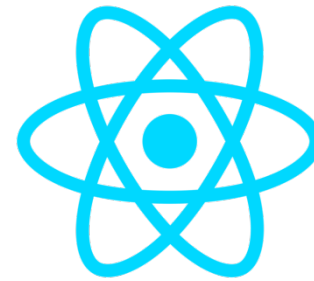


Angular fue creado y mantenido por Google. La primera versión se denominó AngularJS y todavía hay una comunidad utilizando ese framework por su fácil integración con JavaScript. Las versiones sucesivas de Angular se denominan Angular a secas y ya han dejado de ser simples librerías para pasar a ser una 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.

Actualment Angular se programa en TypeScript, que es un superconjunto de JavaScript desarrollado por Microsoft y utiliza el patrón reactivo RxJS

FRAMEWORK REACTJS



ReactJS es un framework creado por facebook que permite a los programadores realizar aplicaciones web de 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 son objetos). Los componentes gestionan sus propios estados y, cuando se agrupan varios componentes, los programadores son capaces de ir creando las interface de usuarios.

Una de las características de React es que usa el DOM virtual que mapea los objetos desde este hasta el DOM del navegador.



FRAMEWORK VUE.JS

Una de las características de este framework frente a otros es la ligereza y la velocidad de ejecución. El objetivo que se plantearon los 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 Larabel lo utilizan para usarlo en el front-end de sus aplicaciones.

Al igual que React, utiliza un DOM virtual, dada las ventajas que ofrece este tipo de implantaciones.

LENGUAJES DE PROGRAMACIÓN EN ENTORNO CLIENTE

La programación web en el lado cliente se basa en tres pilares fundamentales:

1. **HTML:** No es un lenguaje de programación, sino un lenguaje de marcado. Define el contenido que va a tener el documento. La Función del navegador web será el de leer y interpretar todo el contenido para ser visualizado.
2. **EL Lenguaje CSS** define la presentación del documento. CSS es un lenguaje de diseño gráfico y su objetivo es hacer la página atractiva al usuario.
3. **JavaScript.** El código o lenguaje JS agrega el contenido dinámico a las paginas web. JS si que es un lenguaje de programación.



CARACTERÍSTICAS DE LOS LENGUAJES SCRIPT

Los scripts nacieron como fragmentos de código que realizaban ciertas tareas o rutinas concretas. En los sistemas operativos los scripts se utilizan para automatizar tareas y siempre van a ser ejecutados por un interprete de comandos.

Actualmente los scripts ya no son fragmentos de código, sinó que pueden considerarse en auténticos programas.

Diferencias entre lenguajes de script y programación:

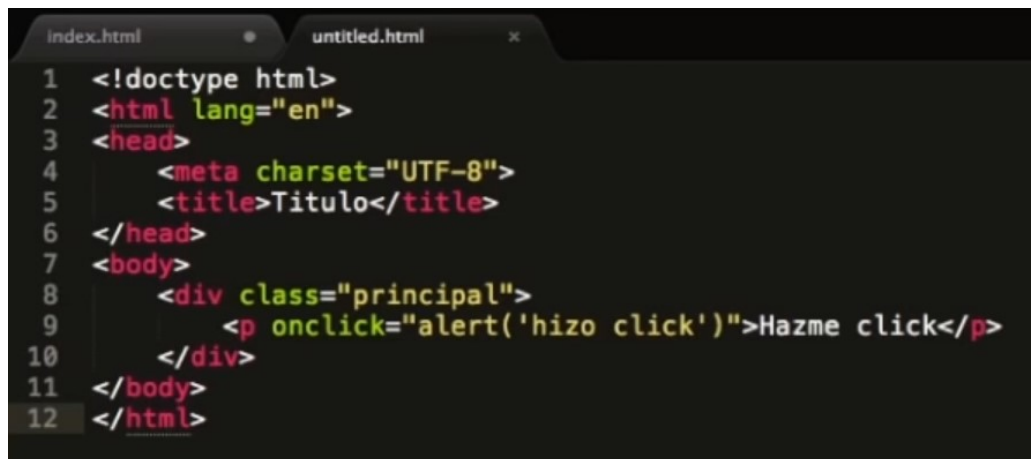
- **Script son interpretados, los de programación compilados.**
- **Scripts utilizan componentes ya preexistentes, mientras que los de programación empiezan desde cero.**
- **Los scripts se puede incrustar dentro de otros programas (HTML).**
- **Los lenguajes de script no generan fichero ejecutable.**
- **Los scripts no necesitan ser compilados.**
- **Los scripts han sido diseñados para ser faciles de utilizar y programar.**

INTEGRACIÓN DE JS DENTRO DE HTML

JS se combina o complementa al código HTML de una página web. Así existen tres opciones:

- En línea
- Embebido
- Archivo externo (recomendado)

EXEMPLO EN LÍNEA:

A screenshot of a code editor with two tabs: 'index.html' and 'untitled.html'. The 'untitled.html' tab is active, showing a basic HTML document structure. The code is as follows:

```
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Titulo</title>
6 </head>
7 <body>
8   <div class="principal">
9     <p onclick="alert('hizo click')">Hazme click</p>
10  </div>
11 </body>
12 </html>
```


INTEGRACIÓN DE JS DENTRO DE HTML

EXEMPLO EMBEBIDO:
(ojo poner al final del script)

```
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Embebido</title>
6 </head>
7 <body>
8   <p>Bienvenidos a DevCode.la</p>
9
10  <p>Saludos a todos!</p>
11
12
13  <script>
14    alert('Hola DevCode.la')
15  </script>
16 </body>
17 </html>
```

EXEMPLO EXTERNO:

```
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Externo</title>
6 </head>
7 <body>
8   <p>Saludos!</p>
9
10
11  <script src="codigo.js"></script>
12 </body>
13 </html>
```

INTEGRACIÓN DE JS DENTRO DE HTML

Recuerda:

El código JavaScript si està dentro de un fichero HTML debe de utilizarse las etiquetas `<script></script>`

El código JS se puede integrar tanto en la etiqueta `<HEAD>` como en la `<BODY>`. O esta al principio o esta al final . Y ten en cuenta que si esta al principio y hace referencia a algún elemento que este detrás no lo reconocerá el navegador.

Si queremos insertar el código JS en un fichero externo (práctica recomendada) debemos de utilizar la etiqueta `<script src='script.js'></script>`

INTEGRACIÓN DE JS DENTRO DE HTML

EJERCICIO 1:

Realiza los un programa que nos salude a iniciar la aplicación con un alert. Debes de hacerlo con las 3 opciones de integración:

- **En Línea**
- **Embebido.**
- **Archivos externo.**

HERRAMIENTAS DE PROGRAMACIÓN EN JS

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 en notepad o nano. Esto no es lo deseable cuando se desarrolla un proyecto ya existen varias alternativas mucho más potentes como los editores online o los IDE. Los IDE nos van a orientar a corregir errores y nos hacen el trabajo mucho más cómodo.

- Herramientas online

Las ventajas de ofrecer un IDE online son evidentes, se puede ejecutar código y probarlo desde cualquier dispositivo solamente teniendo acceso a internet.

Ejemplo:

https://www.tutorialspoint.com/online_javascript_editor.php

HERRAMIENTAS DE PROGRAMACIÓN EN JS

Otra opción es utilizar un IDE complementado con un sistema de control de versiones. Existen muchas alternativas como los que pasamos a detallar:

- Atom <https://atom.io>
- PhpStorm <https://www.jetbrains.com/phpstorm/>
- VisualStudioCode <https://code.visualstudio.com/>
- Brackets <https://brackets.io/>

En este curso vamos a utilizar VisualStudioCode ya que es la herramienta que mas esta utilizando la comunidad en JavaScript.

POSIBILIDADES QUE OFRECE JS

A continuación, se mostrarán varios ejemplos básicos de qué se puede hacer con JS:

- **Modificar el contenido de una página web:**

```
<!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>
```

POSIBILIDADES QUE OFRECE JS

- **Cambiar Atributos de objetos HTML.**

```
<!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/my-
black.jpeg";
    } else {
        image.src = "http://myfpschool.com/wp-content/uploads/2016/06/my-
green.jpeg";
    }
}
</script>

</body>
</html>
```

POSIBILIDADES QUE OFRECE JS

- **Cambiar el estilo CSS**

```
<!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>
```


POSIBILIDADES QUE OFRECE JS

EJERCICIO 2:

Realiza las tres posibilidades que ofrece JS personalizando tú el resultado.

COMUNICACIÓN DE JS CON EL EXTERIOR

Existen varias opciones para que el código JS se comuniqué con el usuario. A continuación, se detallan cada una de ellas:

1. Escribir en la consola del navegador (Console.log)

Con el siguiente código podemos registrar datos de la consola del navegador para comprobar si el funcionamiento de programa es el correcto .

```
✓ <script>  
  |   console.log ("Hola Mundo");  
  </script>
```

Mediante la tecla F12 podemos ir a las opciones del desarrollador del navegador y en la pestaña de consola comprobar el resultado.

COMUNICACIÓN DE JS CON EL EXTERIOR

2. Escribir en cualquier elemento HTML utilizando el atributo innerHTML

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

3. Generar directament HTML utilizando document.write()

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

COMUNICACIÓN DE JS CON EL EXTERIOR

4. Generar un mensaje de alerta utilizando el método `windows.alert()`

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

