

# Selección de arquitecturas y herramientas de programación.



---

## Unidad Trabajo nº 1

David García Fernández



Este obra está bajo una [licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-nc-sa/4.0/).



## Objetivos de la Unidad Trabajo

---

- Conocer la arquitectura de referencia del SW cliente.
- Conocer las características de navegadores.
- Instalar un entorno de desarrollo web.
- Conocer las diferentes formas de integrar código JS en páginas web.



# Evolución y características navegadores web.

---

- **www (world wide web):** representa un compendio de informa accesible a través de Internet.
- El funcionamiento de la web es posible gracias a componentes SW y HW: físicos, protocolos, DNS, etc.
- El desarrollo web se realiza bajo un modelo denominado cliente/servidor (Arquitectura).
  - Cliente realiza peticiones al servidor que permanece a la “escucha” para ofrecer el servicio solicitado.
  - La comunicación cliente/servidor se fija en base a unos protocolos determinados en función del servicio (http, ftp, etc)
  - El servidor debe resolver concurrencia de peticiones.
- Navegador: cliente SW con funcionalidad compleja que hace aumentar la capacidad de interacción del usuario.

# Navegadores o browser.

- SW permite navegar por internet a través de direcciones URL a recursos hipertexto (HTML).
- Tabla características:

Navegadores actuales y su soporte para distintas tecnologías									
Browser	<a href="#">JavaScript</a>	<a href="#">ECMAScript 3</a>	<a href="#">DOM 1</a>	<a href="#">DOM 2</a>	<a href="#">DOM 3</a>	<a href="#">XPath</a>	<a href="#">DHTML</a>	<a href="#">XMLHttpRequest</a>	<a href="#">Rich editing</a>
<a href="#">Amaya</a>	No <a href="#">[note 1]</a>	No <a href="#">[note 1]</a>	No <a href="#">[note 1]</a>	No	No	No	No	No	No
<a href="#">AOL Explorer</a>	Yes	Yes	Partial	Yes	No	No	Yes	Yes	Yes
<a href="#">Avant</a>	Yes	Yes	Partial	No	No	No	Yes	Yes	Yes
<a href="#">Camino</a>	Yes	Yes	Yes	Yes	No <a href="#">[note 2]</a>	Yes	Yes	Yes	Yes
<a href="#">Dillo</a>	No	No	No	No	No	No	No	No	No
<a href="#">DocZilla</a>	Yes	No	Yes	Yes	Partial	Yes	Yes	Yes	No
<a href="#">ELinks</a>	Partial	Partial	No	No	No	No	No	No	No
<a href="#">Web</a>	Yes	Yes	Yes	Yes	No <a href="#">[note 2]</a>	Yes	Yes	Yes	Yes
<a href="#">Flock</a>	Yes	Yes	Yes	Yes	No <a href="#">[note 2]</a>	Yes	Yes	Yes	Yes
<a href="#">Galeon</a>	Yes	Yes	Yes	Yes	No <a href="#">[note 2]</a>	Yes	Yes	Yes	Yes
<a href="#">Google Chrome</a>	Yes	Yes	Yes	Yes	Partial	Yes	Yes	Yes	Yes
<a href="#">iCab</a>	Yes	Yes	Partial	Partial	No	No	Yes	Yes	No
<a href="#">Internet Explorer</a>	Yes	Yes	Yes	Yes <a href="#">[note 3]</a>	Partial	Yes	Yes	Yes	Yes
<a href="#">Internet Explorer for Mac</a>	Yes	Yes	Partial	No	No	No	Yes	No	No
<a href="#">K-Meleon</a>	Yes	Yes	Yes	Yes	No <a href="#">[note 2]</a>	Yes	Yes	Yes	Yes
<a href="#">Konqueror</a>	Yes	Yes	Yes	Yes	Partial	No	Yes	Yes	No
<a href="#">Links</a>	No <a href="#">[note 4]</a>	No	No	No	No	No	No	No	No
<a href="#">Lynx</a>	No	No	No	No	No	No	No	No	No
<a href="#">Maxthon</a>	Yes	Yes	Partial	No	No	Yes	Yes	Yes	Yes
<a href="#">Midori</a>	Yes	Yes	Yes	Yes	Partial <a href="#">[note 5]</a>	Yes <a href="#">[note 5]</a>	Yes	Yes	Yes



# Navegadores o browser.

---

- Características:
  - Plataforma de ejecución: dependencia del SO (Safari).
  - Funcionalidades adicionales: marcadores, descargas, almacén contraseñas, datos formularios, búsquedas.
  - Personalización de interfaz.
  - Soporte de tecnologías web: Estandarización. CSS. RSS, XHTML, etc.
  - Licencia de SW: mayoría SW libre.



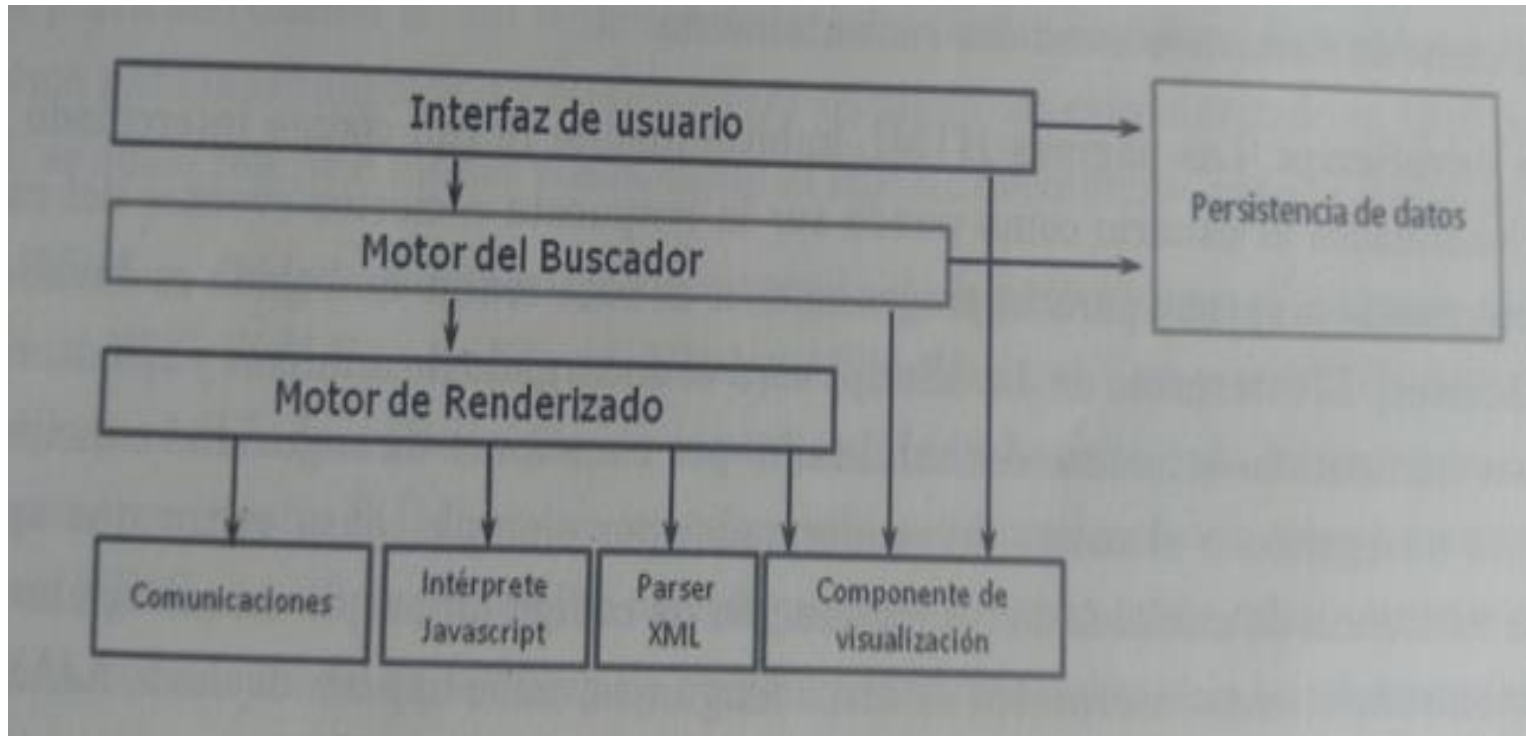
## Arquitectura de ejecución.

---

- Cada navegador tiene su propia forma de interpretar la interacción con el usuario.
- El conjunto de módulos de ejecución del SW y su forma de interactuar entre ellos (comunicación) definen la arquitectura.
- Arquitectura de referencia: constituyen los módulos comunes que utilizan los navegadores.

# Arquitectura de ejecución.

- Arquitectura de referencia:





# Arquitectura de ejecución.

---

- Arquitectura de referencia:
- Subsistema de interfaz de usuario
  - Capa SW actúa interfaz entre usuario y navegador.
  - Funcionalidades:
    - Visualizar barra de herramientas.
    - Proceso de carga de páginas.
    - Gestión inteligente de descargas.
    - Preferencias de configuración de usuario e impresión.
    - Comunicación con SO para el manejo de sesiones o almacenamiento de preferencias de visualización o configuración.





# Arquitectura de ejecución.

---

- Arquitectura de referencia:
- Subsistema del motor del navegador.
  - Ofrece una interfaz de alto nivel al motor de renderizado.
  - Funcionalidad:
    - Cargar un URL.
    - mecanismos de recarga de página, ir a siguiente o anterior.
    - Gestión de alertas de JS.
    - Consuta y administra preferencias del motor de renderizado.



# Arquitectura de ejecución.

---

- Arquitectura de referencia:
- Subsistema de renderización.
  - Ofrece una representación visual del recurso accedido por URL.
  - Funcionalidades:
    - Interpreta el código de la página web.
    - Muestra documentos HTML,XML, hojas CSS, etc.



# Arquitectura de ejecución.

---

- Arquitectura de referencia:
- Subsistema de comunicación.
  - Implementa los protocolos de transferencia de ficheros y documentos (HTML,FTP).
  - Funcionalidades:
    - Identifica la codificación de ficheros en base a su tipo (MIME-Multipurpose Internet Mail Extensions).
    - En algunos navegadores puede almacenar caché de elementos.



# Arquitectura de ejecución.

---

- Arquitectura de referencia:
- Interprete Javascript.
  - Funcionalidades:
    - Analiza y ejecuta el código incrustado en páginas HTML.
- Parser XML.
  - Funcionalidades:
    - Permite cargar en memoria una representación del DOM (Document Object Model). Facilitando una mayor velocidad de acceso a los elementos.



# Arquitectura de ejecución.

---

- Arquitectura de referencia:
- Componentes de visualización:
  - Funcionalidades:
    - Ofrece funciones de dibujo y posicionamiento de ventanas, componentes predefinidos (widgets), fuentes tipográficas.
- Subsistema persistencia de datos.
  - Funcionalidades:
    - Almacén de distintos tipos de datos utilizados para el resto de subsistemas. (historial navegación, sesiones usuarios, barra de herramientas, marcadores, certificados y cookies.



# Lenguajes y tecnologías de programación en entorno cliente.

---

- HTML y derivados.
- CSS
- JS
- Applets Java
- AJAX
- Adobe Flash y ActionScript



## Integración del código en HTML.

---

- JS en el mismo html
- JS en un archivo externo.
- JS en elementos HTML `<p onclick=...>`



# Integración del código en HTML.

- JS en el mismo html

```
<!DOCTYPE html PUBLIC "-//w3c//dtd XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
    <title> Ejemplo 1 </title>
    <script type="text/javascript">
      alert("Prueba de JS");
    </script>
  </head>
  <body>
    <h1>Ejemplo 1: código embebido</h1>
  </body>
</html>
```





# Integración del código en HTML.

- JS en un archivo externo.

```
<!DOCTYPE html PUBLIC "-//w3c//dtd XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="
      text/html; charset=iso-8859-1" />
    <title> Ejemplo 2</title>
    <script type="text/javascript"
      src="/inc/mensaje.js">
    </script>
  </head>
  <body>
    <h1>Ejemplo 2: Fichero externo</h1>
  </body>
</html>
```



# Integración del código en HTML.

- JS en elementos HTML <p onclick=...>

```
<!DOCTYPE html PUBLIC "-//w3c//dtd XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="
      text/html; charset=iso-8859-1" />
    <title> Ejemplo 3</title>
  </head>
  <body>
    <p onclick="alert('Prueba JS');">
      Ejemplo 2: Fichero externo
    </p>
  </body>
</html>
```



# Ejercicios Práctica.

---

1. Realizar la instalación y configuración de un entorno de trabajo para programar con JavaScript.
  - Para ello tendrás que buscar e instalar en tu sistema operativo 2 IDE. Todas las aplicaciones que instales tendrán que ser gratuitas y tendrás que explicar las razones por las que has elegido tu editor web entre muchos otros.
  - Usando la dirección de validación del W3C indicada en los apuntes realiza la validación de una página web sencilla e indica los tipos de errores encontrados.
2. Prueba las tres formas de integrar código javascript en una página básica HTML. Ejemplo una alerta o mensaje.



## Bibliografía.

---

- DWEC. Editorial Rama.