

---

# UD1 WEB y entornos de desarrollo

**2º CFGS DAW**

**Desarrollo Web en Entorno Cliente**

**2019-20**



# Internet

Redes de comunicaciones conectadas para formar una red global.

Todas las redes tienen un objetivo: intercambiar información.

Utiliza la pila de protocolos TCP/IP

El servicio más usado en internet es: **World Wide Web**

# 1.- World Wide Web

Otros nombres:

- WWW
- WEB
- Internet

# 1.- World Wide Web

Muchos servicios se han adaptado para ofrecerse por WEB.

- Correo electrónico

- Streaming

- FTP

- Mensajería instantánea

Algo similar ha ocurrido en los últimos años con las aplicaciones para móviles.

# 1.- World Wide Web

La WEB se basa en el **hipertexto**.

El primer hipertexto se componía de archivos de texto que enlazaban a otros archivos de texto a través de **hipervínculos**.

Hoy en día el hipertexto permite enlazar cualquier tipo de medio:

Texto      Imágenes      Vídeo      ...

# 1.- World Wide Web

## HTTP - Hypertext Transfer Protocol

Es el protocolo, conjunto de reglas, que han de seguir las aplicaciones que se conectan a internet para poder hacer uso de hipertexto.

Existen diferentes tipos de peticiones:

GET      HEAD      PUT      POST      DELETE      ...

# 1.- World Wide Web

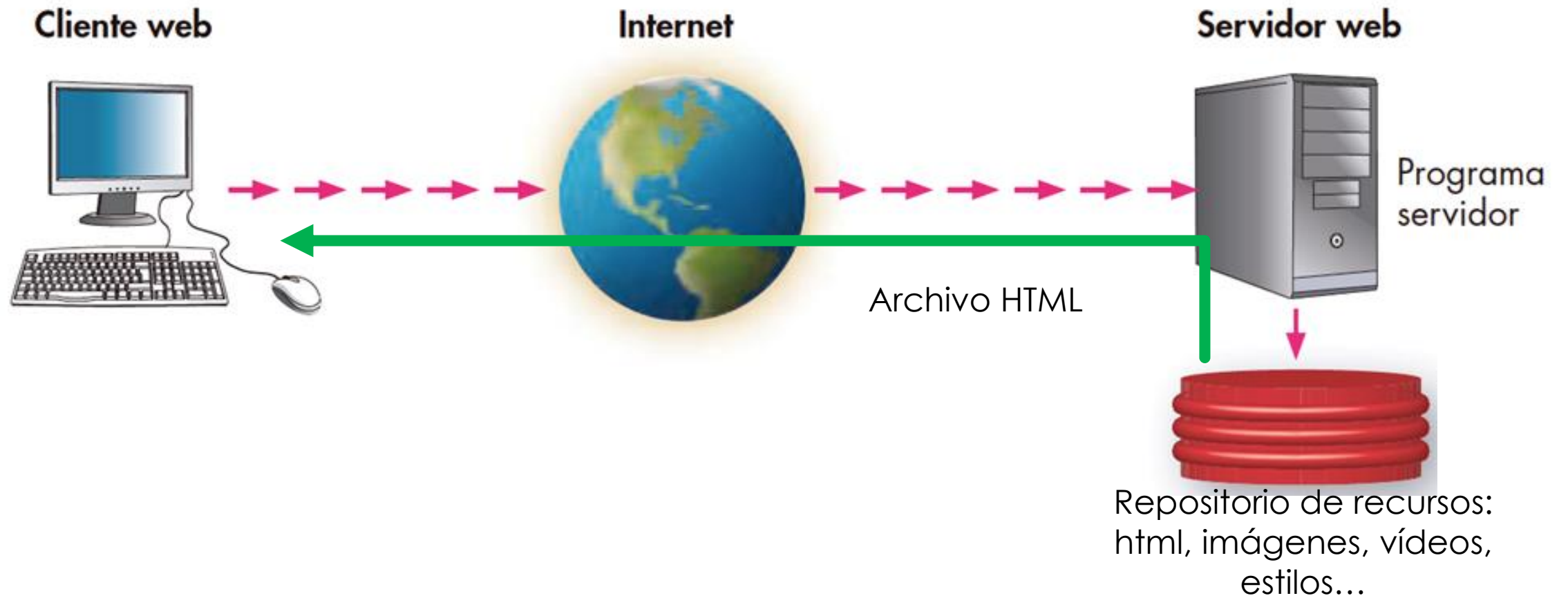
## HTML - HyperText Markup Language

Es el lenguaje en el que se tiene que programar las aplicaciones web.

**Página web vs Aplicación web**

# 1.- World Wide Web

## Funcionamiento de la WEB



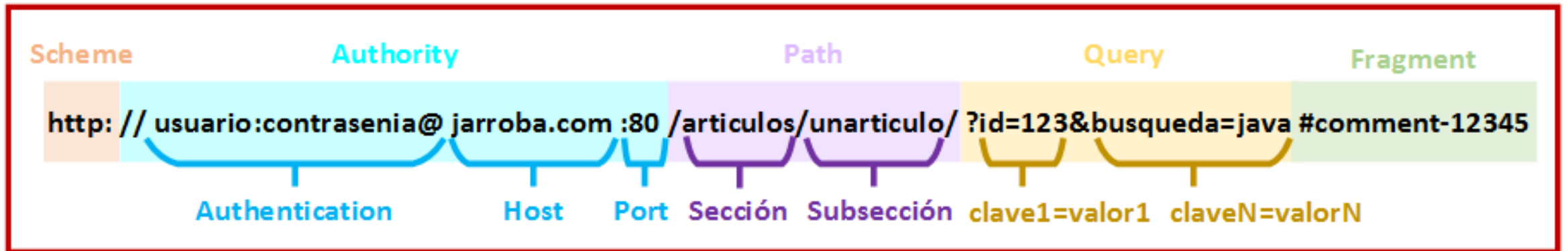


# 1.- World Wide Web

## Dirección de los recursos WEB

**URL: Uniform Resource Locator** (localizador uniforme de recursos).

### URI / URL

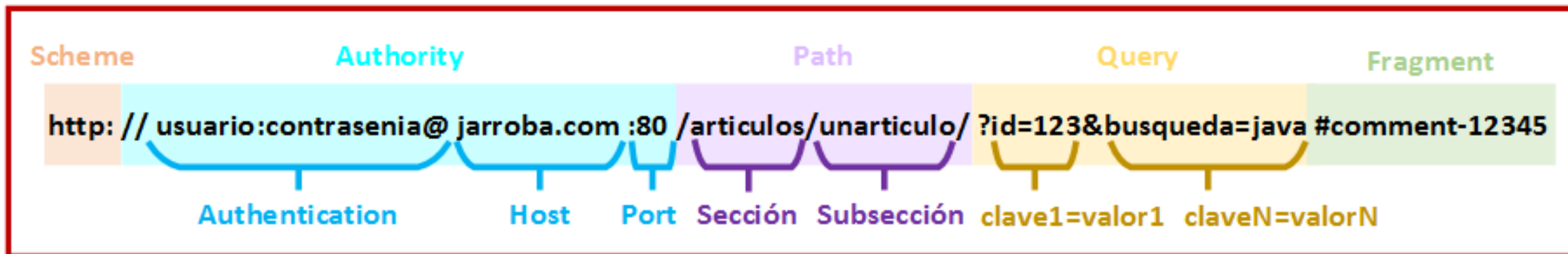


La parte **Authority** no se suele utilizar (DAW).

# 1.- World Wide Web

## Dirección de los recursos WEB

### URI / URL







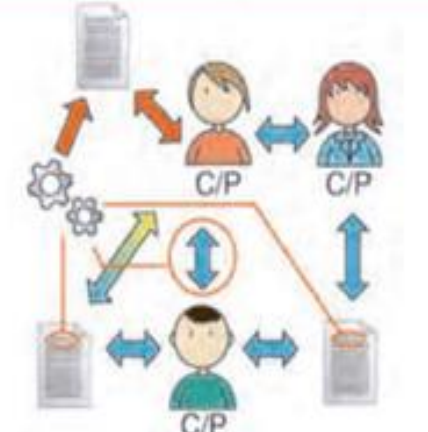
`http://my-server.com/unit1/exercices/1.html`

`http://my-server.com/unit1/notes.html#p3`

`http://my-server.com/search?value=manises`

# 1.- World Wide Web

## Aplicaciones web estáticas vs dinámicas

Web 1.0	Web 1.5	Web 2.0	Social media	Web semántica
		 <p>El C/P es el Prosumidor</p>		
estática	dinámica	colaborativa	social	inteligente
1989 – 1997	1997 – 2003	2003 – 2008	2008 – Actual	En un futuro cercano

# 1.- World Wide Web

## Aplicaciones web

Estáticas	Dinámicas
<p>Almacenadas en su forma definitiva.</p> <p>Solo varían si el desarrollador altera el contenido.</p> <p>Su utilidad se basa en mostrar información concreta.</p> <p>Consumen menos recursos.</p> <p>La extensión de los archivos es .html</p>	<p>Su contenido cambia dependiendo de diferentes factores: día y hora, si se accede con usuario, acciones realizadas previamente...</p> <p>El cliente recibe un archivo cuyo contenido es html, pero dicho contenido no se encuentra dentro de un archivo inalterable.</p> <p>La extensión del archivo será .php, .asp, .cgi... (según el lenguaje de programación de páginas web dinámicas que entienda el servidor)</p>

---

## 1.- World Wide Web

Aplicaciones web

# ¿Estáticas o Dinámicas?

# 1.- World Wide Web

## Aplicaciones web

### Estáticas:

No es necesario saber programar. 👍

Su contenido nunca varía, los enlaces siempre mostrarán lo mismo. 👍

Mejor posicionamiento SEO al tener siempre el mismo contenido. 👍

Actualización de manera manual por el desarrollador web. 👎

### Dinámicas:

Más flexibilidad. 👍

Mayor dificultad en el desarrollo. 👎

Mayor consumo de recursos. 👎

Hay que ser cuidadoso para el posicionamiento SEO. 👎

Menor velocidad. 👎

Mayor coste de mantenimiento de recursos. 👎

---

# 1.- World Wide Web

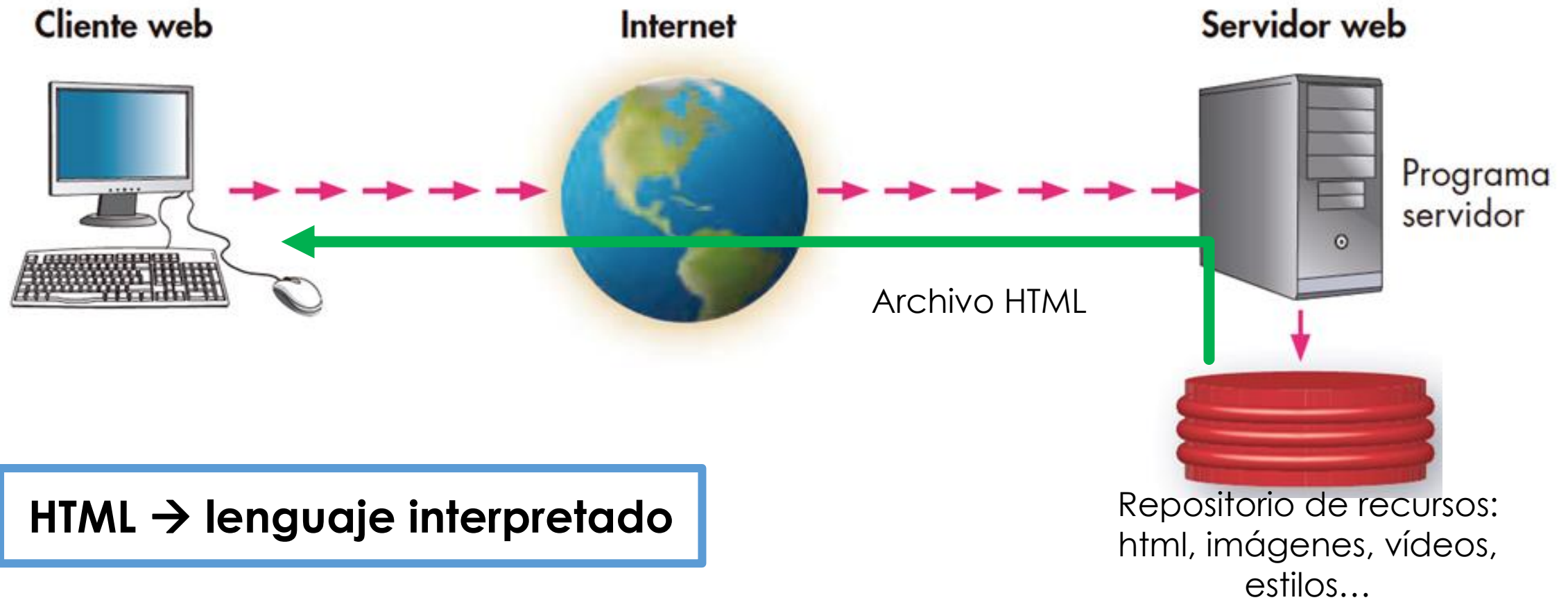
## Aplicaciones web

# Estáticas y Dinámicas

Las aplicaciones web actuales tienen diferentes partes cada una puede ser estática o dinámica aprovechando lo mejor de cada una.

# 1.- World Wide Web

## Aplicaciones web



**HTML → lenguaje interpretado**



# 1.- World Wide Web

## Aplicaciones web

**HTML → Lenguaje interpretado**

¿Dónde se ejecuta?

¿Quién lo ejecuta?

¿Si el código HTML necesita algún archivo (imagen o css) qué pasa?

---

## 1.- World Wide Web

**Programación en el  
lado del cliente**

**y**

**Programación en el  
lado del servidor**

---

# 1.- World Wide Web

Programación en el lado del cliente

**HTML**



**JS**



**CSS**



# 1.- World Wide Web

## Programación en el lado del cliente

Las nuevas metodologías aconsejan separar de manera muy diferenciada los elementos que conforman una página web:

- Contenido → HTML
- Estilo → CSS
- Interacción → JavaScript

Se obtendrán aplicaciones web más modulares y reutilizables.

Se debe utilizar una estructura de directorios adecuada.

# 1.- World Wide Web

## Programación web

1. Se diseñará y se creará con HTML la estructura de la aplicación web.
2. Se introducirá el contenido dinámico (php, Java...).
3. Se aplicará estilo mediante CSS.
4. Se podrá añadir interacción mediante JavaScript y sus librerías (jQuery...).

# 1.- World Wide Web

HTML → Lenguaje interpretado

¿Dónde se ejecuta?

**¿Quién lo ejecuta?**

¿Si el código HTML necesita algún archivo (imagen o css) qué pasa?

## 2.- Navegador web

Software que permite el acceso a archivos de hipertexto.

Principalmente su funcionalidad es la de interpretar texto HTML.

Los recursos a los que se acceden principalmente se encuentran en internet.

También se puede acceder a recursos propios del ordenador o de equipos vecinos de la red sin necesidad de que estos tengan instalado un servidor web.

## 2.- Navegador web

Gracias a los hipervínculos incluidos en los archivos HTML el usuario puede ir enlazando los contenidos uno tras otro.

Esta acción se conoce como **navegar**, de aquí el nombre de los navegadores.

Termino de los 90: surfear





## 2.- Navegador web

Los primeros navegadores solo eran capaces de interpretar una versión muy simple del lenguaje HTML.

Las empresas desarrollaron navegadores que añadían funcionalidades generalmente no incluidas en los estándares.

Esto derivó en que los programadores creaban aplicaciones web que solo se “veían bien” en un navegador determinado (Internet Explorer).

## 2.- Navegador web

Actualmente los navegadores web cumplen los estándares en un alto porcentaje.

Los programadores deben de cumplir también los estándares.

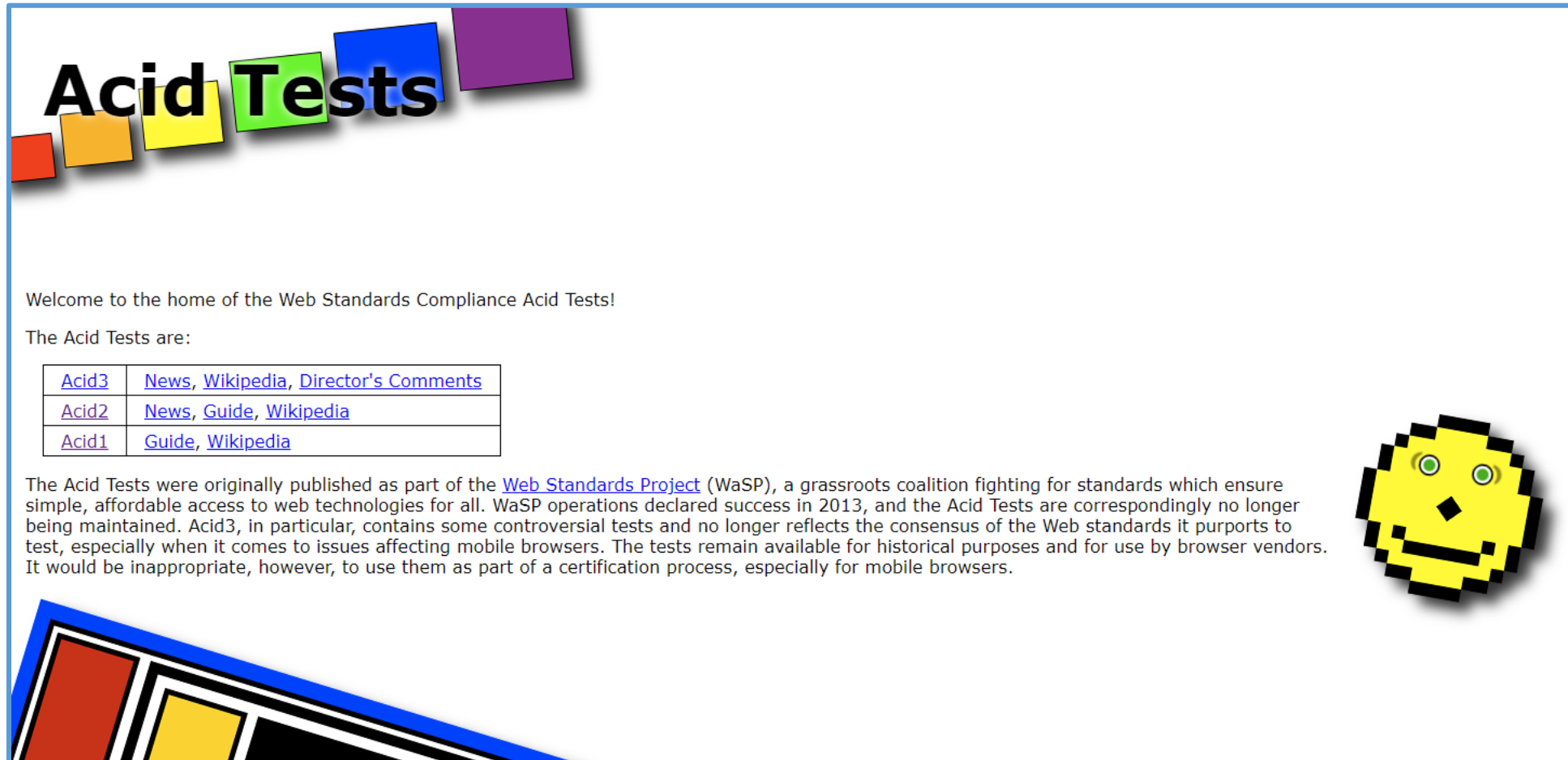
Si se cumplen las dos condiciones anteriores una aplicación web funcionará igual en cualquier navegador actual.

HTML5

CSS3

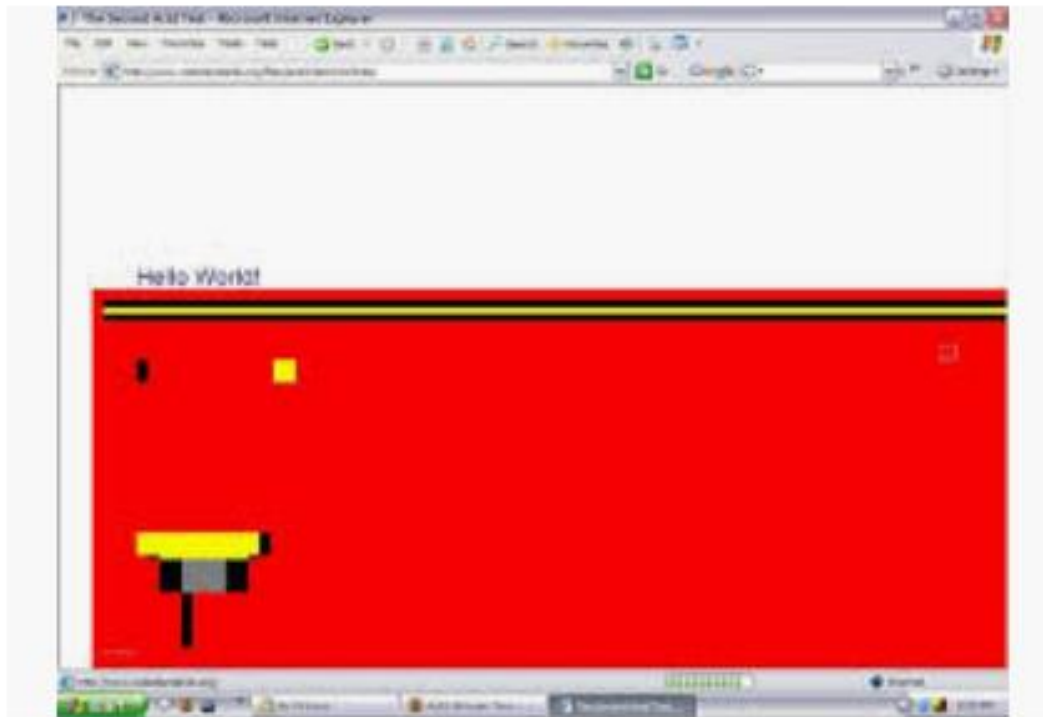
## 2.- Navegador web

Testea tu navegador: <https://www.acidtests.org>



## 2.- Navegador web

Ejemplo de AcidTests1 en Internet Explorer 6 (año 2001)



## 2.- Navegador web

Como programador existen herramientas para saber si se está cumpliendo con los estándares o no: WWW Consortium (W3C)

HTML: <http://validator.w3.org/>

CSS: <https://jigsaw.w3.org/css-validator/>

## 2.- Navegador web

Una buena manera de saber cómo usar las etiquetas HTML y los selectores CSS es consultar la documentación oficial.

También se puede usar la aplicación web <https://www.w3schools.com/>

## 2.- Navegador web

Protocolo de funcionamiento: HTTP

También soportan otros protocolos: FTP, HTTPS...

Incluso son capaces de mostrar archivos directamente: imágenes, pdf...

## 2.- Navegador web

Los usuarios utilizan los navegadores para acceder a las diferentes aplicaciones web disponibles en internet.

Existen muchos navegadores en el mercado.



Cualquier programador web que se precie además de usar los validadores deberá probar sus aplicaciones web en varios navegadores.



### 3.- Principales navegadores web

**Microsoft Edge** (sucesor de Internet Explorer)

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

Está diseñado para ser un navegador web ligero con un motor de renderizado de código abierto construido en torno a los estándares web.

## 3.- Principales navegadores web

### **Mozilla Firefox**

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

Es un navegador web libre y de código abierto desarrollado por la Corporación Mozilla y la Fundación Mozilla.

Usa el motor Gecko para renderizar páginas webs, el cual implementa actuales y futuros estándares web.

Versión para desarrolladores: “Firefox Developer Edition”

## 3.- Principales navegadores web

### Google Chrome

URL Oficial: <https://www.google.com/chrome/>

Es un navegador web desarrollado por Google y compilado con base en varios componentes e frameworks de código abierto, como el motor de renderizado Blink (bifurcación o fork de WebKit).

Está disponible gratuitamente bajo condiciones específicas del software privativo o cerrado.

### 3.- Principales navegadores web

#### Opera

URL oficial: <http://www.opera.com/es>

Es un navegador web creado por la empresa noruega Opera Software.

Usa el motor de renderizado Blink.

Tiene versiones para escritorio, teléfonos móviles y tablets.

## 3.- Principales navegadores web

### Google Chrome

URL Oficial: <https://www.google.com/chrome/>

Es un navegador web desarrollado por Google y compilado con base en varios componentes e frameworks de código abierto, como el motor de renderizado Blink (bifurcación o fork de WebKit).

Está disponible gratuitamente bajo condiciones específicas del software privativo o cerrado.

### 3.- Principales navegadores web

Los navegadores anteriores son los que más cuota de mercado tienen.

¿Se te ocurre algún otro navegador web?

¿Qué navegador web es mejor? ¿Por qué?

¿Cuál usarías durante el curso? ¿Por qué?

### 3.- Principales navegadores web

Firefox y Chrome

Para el curso se recomienda usar uno de estos dos debido a las posibilidades de depuración que ofrecen.

Recuerda que el navegador web es el que interpreta el código por lo que es el lugar idóneo para depurarlo.

Además de poder añadir diferentes plugins.



# Actividades

## Actividad 1

Navegadores web



## 4.- Herramientas del navegador para el desarrollo WEB

**Inspeccionar elemento.**

**Consola WEB.**

Red.

Recursos.

Rendimiento.

# Actividades

## Actividad 2

Inspeccionando la aplicación web

## 5.- Entorno de desarrollo

Como con cualquier lenguaje, no es necesario ningún software específico.

El único programa necesario para desarrollar aplicaciones web en el cliente es el navegador web.

Usar un entorno de desarrollo ofrece unas características muy importantes que ya se estudiaron el curso pasado. ¿Recuerdas alguna?

## 5.- Entorno de desarrollo

- **Brackets**
- Notepad++
- **SublimeText**
- **Visual Studio Code**
- Dreamweaver
- Eclipse
- Netbeans
- ...

Si no se van a desarrollar aplicaciones web dinámicas mediante Java es recomendable no usar las suite específicas de Java como Eclipse o NetBeans

## 5.- Entorno de desarrollo

Se recomienda el uso de **Visual Studio Code** por su potencia o **SublimeText**.

Página oficial: <https://code.visualstudio.com/>

Algunos manuales libres de uso de Visual Studio Code en castellano:

<http://www.mclibre.org/consultar/informatica/lecciones/vsc-instalacion.html>

<http://www.mclibre.org/consultar/informatica/lecciones/vsc-personalizacion.html>

Página oficial: <https://www.sublimetext.com/>

## 5.- Entorno de desarrollo

Existe una herramienta muy útil en el desarrollo web:

<https://emmet.io/>

Esta herramienta permite aumentar la productividad mediante creación de código de manera rápida: <https://youtu.be/rP-K62mbUm0?t=224>



# Actividades

## Actividad 3

El entorno de desarrollo – IDE

## Actividad 4

Cumpliendo los estándares.

## 6.- Control de versiones

El curso pasado se aprendió para que sirven los controles de versiones.

Son una herramienta que facilitan mucho el trabajo, tanto por la posibilidad de crear ramas como cuando se trabaja en diferentes ordenadores.

Se recomienda usar algún control de versiones como <https://bitbucket.org/>.

Si el IDE lo permite se puede integrar en él, o si no se puede usar un cliente como <https://www.gitkraken.com/>.



## 6.- Control de versiones

- Para usar Git en Visual Studio Code

<https://code.visualstudio.com/docs/editor/versioncontrol>

<http://www.mclibre.org/consultar/informatica/lecciones/vsc-git-repositorio.html>

- Aquí un ejemplo del uso de Git en Visual Studio Code.

<https://code.visualstudio.com/docs/introvideos/versioncontrol>

# Actividades

## Actividad 5

El portafolio.