

Una aproximación a la World Wide Web, desde el concepto de Internet y un poco de historia, hasta el funcionamiento básico de la Web, antes de adentrarnos en el desarrollo y puesta en producción de sitios web para la adquisición y la presentación de datos.

Introducción a la Web

Bases de Datos 2023/2024

Luis Valencia Cabrera, David Orellana Martín

Contenido

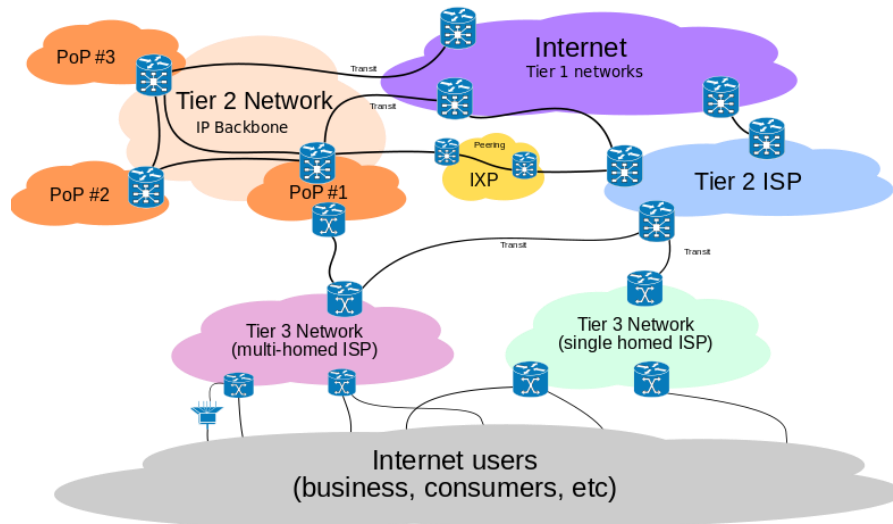
Introducción a la Web.....	2
Internet	2
Dirección IP	4
Dominio de Internet.....	4
Servicios y aplicaciones en Internet.....	5
World Wide Web (WWW).....	5
Funcionamiento de la Web.....	6
Ejemplo.....	8
Servidor Web	9
Páginas Web.....	10
El lenguaje HTML	10

Introducción a la Web

Internet

Internet es un conjunto descentralizado de [redes de comunicación](#) interconectadas que utilizan la familia de [protocolos TCP/IP](#), garantizando que las redes físicas [heterogéneas](#) que la componen funcionen como una red lógica única, de alcance mundial. Sus orígenes se remontan a [1969](#), cuando se estableció la primera conexión de computadoras, conocida como [ARPANET](#), entre tres universidades en [California](#) y una en [Utah, Estados Unidos](#).

Internet se estructura en capas de comunicaciones, con servicios proporcionados por los proveedores de servicios de Internet (ISP) que operan en cada capa.

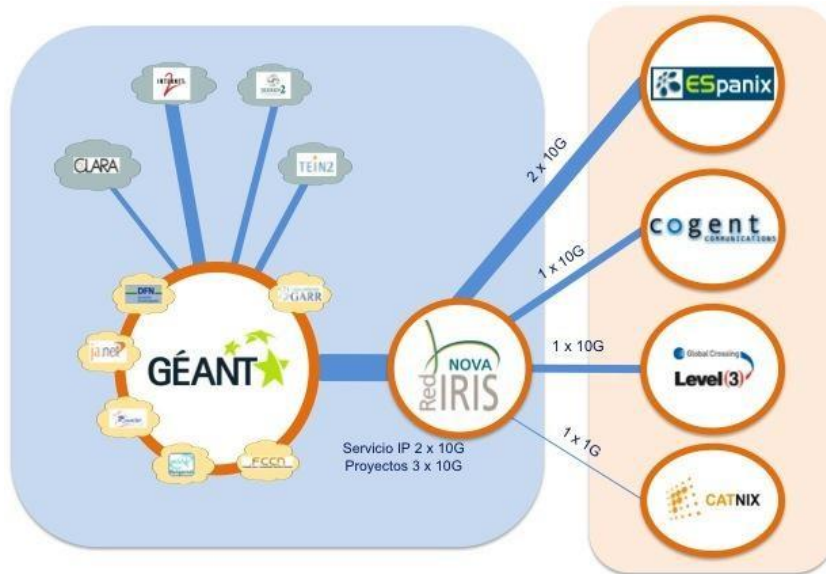


La [capa 1](#) está formada por un grupo muy reducido de ISPs que tienen una cobertura internacional.

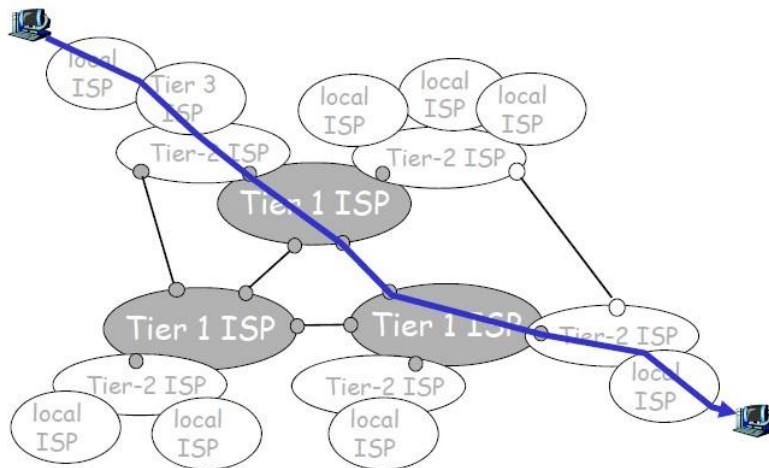
Por ejemplo, la red de la Universidad de Sevilla está integrada en RedIris:



A su vez, RedIris se encuentra conectado con otras redes, dentro de Géant, entre ellas la red de la capa 1 Cogent:



¿Qué nos proporciona esto? La interconexión entre las redes y, por ende, los equipos, de manera que podamos conectarnos desde nuestro equipo de casa a otro determinado equipo o servidor de la red:



Esta red de interconexiones se sustenta sobre el [modelo OSI](#), que incluye una serie de niveles entre los que se encuentra el **nivel de red (IP)**, encargado de hacer que los datos lleguen desde el origen al destino, aun cuando ambos no estén conectados directamente, realizando el *direccionamiento lógico* y la *determinación de la ruta* de los datos hasta su receptor final.

Niveles superiores a IP garantizan el correcto transporte de la información, el mantenimiento de una sesión entre los equipos, la presentación de la información en los formatos adecuados y el nivel superior, de aplicación, que nos permiten acceder a servicios como SMTP, FTP, etc. El usuario interactúa con aplicaciones amigables como el **correo** (hace uso de SMTP) o el **navegador web** y de archivos (hace uso de FTP, HTTP...), que lo abstraen de los detalles de los niveles comentados.

Dirección IP

Para poder llevar a cabo la comunicación de datos entre origen y destino dentro de la red anterior, es necesario que cada elemento de la red disponga de una dirección determinada, como ocurre cuando un cartero ha de hacer llegar un determinado paquete a nuestra casa.

Así, por ejemplo, un equipo de un despacho en la Universidad de Sevilla dispondrá de una conexión a Internet a través de un cable. De este modo, si nos preguntan por la dirección del profesor podremos decir que es:

- ETSII. Av. Reina Mercedes s/n, 41012 Sevilla, módulo H, despacho H1.46, o bien
- 150.214.186.135 (*inventada la terminación de la dirección IP, es aproximado*).

Es decir, la dirección IP que llega a mi despacho, donde conecto mi equipo, sería 150.214.186.135. Esta dirección (si fuera la correcta) identificaría de forma única a mi equipo dentro de Internet. Igualmente, la dirección 216.58.201.163 identifica (en el momento de realizar la comprobación – 07/01/2020, desde el despacho del profesor) a Google España.

Para que podamos acceder a esa dirección para solicitar algún servicio (SMTP, HTTP, FTP...), debe tratarse de una **IP pública**. Se pueden configurar redes privadas no accesibles desde Internet sino únicamente desde una Intranet, y para distinguir cada equipo accederíamos a su **IP privada**.

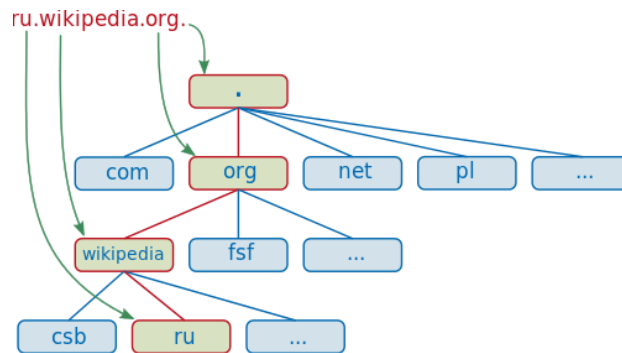
Por otro lado, las direcciones IP proporcionadas por la Universidad de Sevilla no cambian generalmente, están asignadas de forma fija, de modo que se denominan **IP estática** o **fija**. Sin embargo, cuando nos conectamos desde casa, generalmente nuestro proveedor de Internet (ISP, como Movistar, Jazztel, Vodafone, etc.) nos proporcionará una **IP dinámica**, es decir, irá cambiando a lo largo del tiempo. Las empresas tienen la posibilidad de contratar una IP estática pública, de modo que puedan ofrecer servicios a Internet desde su empresa. Si un particular desea ofrecer esos servicios, generalmente hará uso de algún hosting, formando parte del conjunto de direcciones de una empresa, alojando los contenidos en la infraestructura de ésta.

Ahora bien, para acceder a Google España, ¿necesitamos conocer la dirección IP en cuestión, 216.58.201.163? No parece muy práctico...

Dominio de Internet

Un dominio de Internet es una red de **identificación** asociada a un grupo de dispositivos o equipos conectados a la red Internet.

El propósito principal de los nombres de dominio en Internet y del **sistema de nombres de dominio (DNS)**, es traducir las direcciones IP de cada nodo activo en la red, a términos memorizables y fáciles de encontrar. Esta abstracción hace posible que cualquier servicio (de red) pueda moverse de un lugar geográfico a otro en la red Internet, aun cuando el cambio implique que tendrá una dirección IP diferente.



Como vemos, la URL (dirección) *ru.wikipedia.org* formará parte del dominio **wikipedia.org**, que a su vez forma parte del dominio de nivel superior **org**. De la coordinación de estos nombres se encarga la ICANN:

ICANN es una organización que opera a nivel internacional y es la responsable de asignar las direcciones del protocolo IP, de los identificadores de protocolo, de las funciones de gestión del sistema de dominio y de la administración del sistema de servidores raíz.

Existen una serie de dominios de nivel superior genéricos (org, com, net, ...), y otros geográficos (es, uk, eu, ...). Dentro de cada uno de ellos se van asignando dominios, como google.es.

Como comentaba, estos nombres nos sirven para identificar de forma más sencilla un equipo, y de forma que no cambie a lo largo del tiempo en caso de cambio de IP. Si queremos ver la dirección IP asociada a un determinado nombre (por ejemplo, google.es), podemos ir al símbolo del sistema (Inicio > Ejecutar > cmd, o bien Inicio > Todos los programas > Accesorios > Símbolo del sistema), y desde la consola lanzar **ping www.google.es**. Esta instrucción intentará conectarse a la dirección IP correspondiente a la dirección por nombre de dominio que estamos indicando, y nos mostrará entre otra información la IP, 216.58.201.163.

Servicios y aplicaciones en Internet

Como hemos comentado, estamos familiarizados con programas como los clientes de correo (sobre el protocolo SMTP), de transferencia de archivos (sobre FTP), navegadores (para transferencia de hipertexto, a través de HTTP), y otros muchos.

De entre todos ellos, destaca el uso de los navegadores, que nos permiten el acceso a una cantidad ingente de información, haciendo uso fundamentalmente del protocolo HTTP. Este sistema de distribución basado en navegadores y HTTP se conoce como World Wide Web (www), y se puede considerar el buque insignia de Internet.

World Wide Web (WWW)

Se trata de un sistema, un medio de comunicación/transmisión de texto, gráficos y otros objetos multimedia a través de Internet, incluyendo enlaces o hipervínculos entre los grupos de información (se conoce de forma general como *hipertexto*). La www no es Internet, sino un conjunto de ella. Muchos juegos, el correo, el FTP, etc. son parte de Internet, la red que veníamos comentando, pero no forman parte de la www, que podemos identificar mejor con aquello que vamos solicitando desde nuestros navegadores (Chrome, Firefox, Edge, Explorer, Opera, Safari...) y que nos muestra como respuesta a nuestras peticiones desde los mismos.

La www fue creada en 1989 por Tim Berners-Lee con Robert Cailliau mientras estaban en el CERN en Suiza, el entonces nodo más importante de Internet en Europa. Él y su grupo crearon el lenguaje [HTML](#) (HyperText Markup Language), el protocolo [HTTP](#) (HyperText Transfer Protocol) y el sistema de localización de objetos en la web [URL](#) (Uniform Resource Locator). Todo ello para poder intercambiar y distribuir sus investigaciones de forma más efectiva, y como antesala a su creación de la WWW, comenzando con un servidor (httpd) y un navegador (WorldWideWeb). En 1994 se trasladó al MIT, y pasó a encabezar el organismo de coordinación y estandarización de la web, el W3C (World Wide Web Consortium).

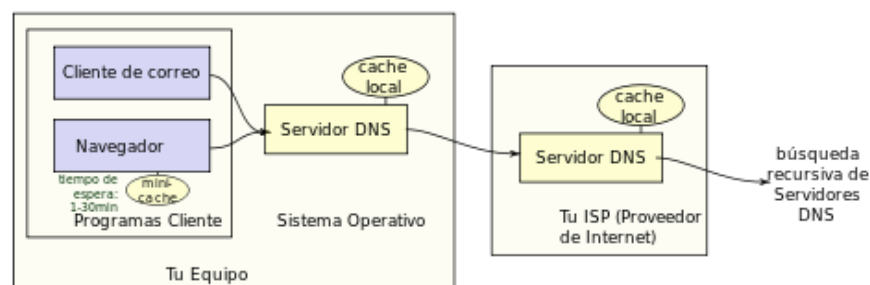
Funcionamiento de la Web

El funcionamiento general de la web consiste en una serie de pasos:

1. Se introduce en nuestro navegador Web una cadena de texto que hace referencia a una determinada dirección web accesible a través de Internet. Esta cadena incluye una parte que se refiere al nombre del servidor, dentro de un dominio de Internet.
2. La parte de la URL referida al nombre del servidor es traducida en una dirección IP usando el sistema de nombres de dominio o DNS. Se trata de una gran base de datos distribuida y jerárquica a lo largo de Internet, conteniendo las asignaciones de nombres de dominio a direcciones IP. El servidor DNS utiliza esta base de datos para devolvernos la dirección IP correspondiente al nombre introducido.

Los usuarios generalmente no se comunican directamente con el servidor DNS: la resolución de nombres se hace de forma transparente por las aplicaciones del cliente como los navegadores. Al realizar una petición que requiere una búsqueda de DNS, la petición se envía al servidor DNS local del sistema operativo. El sistema operativo, antes de establecer alguna comunicación, comprueba si la respuesta se encuentra en la memoria caché. En el caso de que no se encuentre, la petición se enviará a uno o más servidores DNS.

La mayoría de usuarios domésticos utilizan como **servidor DNS** el proporcionado por el proveedor de servicios de Internet (**ISP**). La dirección de estos servidores puede ser configurada de forma manual o automática mediante DHCP. En otros casos, los administradores de red tienen configurados sus propios servidores DNS.



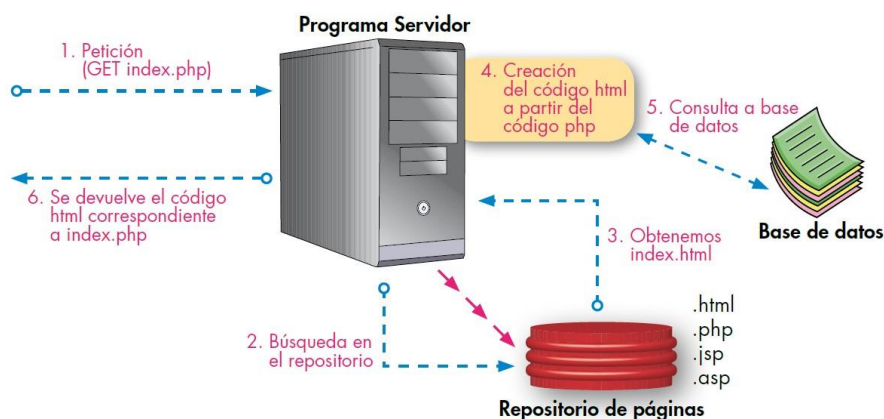
En definitiva, como resultado de este paso se obtiene la dirección IP del servidor al que hicimos referencia. **Esta dirección IP es necesaria para contactar con el servidor web y poder enviarle paquetes de datos.**

3. A continuación, se envía una petición **HTTP** al servidor web solicitando el recurso (página web, por ejemplo).
4. El servidor web devuelve la respuesta HTTP, junto con el contenido de la página solicitada; por ejemplo, la página **HTML** de respuesta.
5. El navegador recibe la página HTML. En el caso de una página web típica, el código HTML es inmediatamente **analizado por el navegador**, que hace peticiones adicionales si es necesario traer nuevos archivos al cliente para completar la página (por ejemplo, imágenes).
6. Con la página HTML junto con los ficheros necesarios (en su caso), **el navegador renderiza la página** tal y como se describe en el código **HTML**, el **CSS** y **otros lenguajes web**. Al final se incorporan las imágenes y otros recursos para producir la página que ve el usuario en su pantalla.

De forma muy simplificada, podemos ver el proceso como la resolución de la dirección, más la petición HTTP más la respuesta con la página:



Si pasamos a las peticiones HTTP de páginas web dinámicas (en lugar de un HTML, solicitamos algo terminado en .php, .jsp, .do, etc), antes de obtener la respuesta del servidor se tendrá que realizar cierto procesamiento en el mismo antes de generar y devolver la respuesta HTML. El proceso sería a grandes rasgos del siguiente tipo (obviando en este caso los pasos iniciales para resolver el nombre del dominio):



En resumen:

- Las páginas Web están almacenadas en un servidor web.
- Se utiliza un navegador web para solicitar y acceder a dicha información.

- El servidor proporciona al navegador las páginas Web tal cual sin modificar (en el caso de páginas estáticas, se generan y envían tal cuál) o bien se procesan mediante un lenguaje del lado del servidor (PHP, Python, JS mediante NodeJS, Ruby, etc.) en el caso de páginas dinámicas. *Como veremos más adelante, se puede solicitar y enviar información que no sea propiamente de páginas web sino con otro tipo de contenidos y formatos, como JSON.*
- Las páginas Web (estáticas o ya devueltas tras generarlas con el lenguaje de servidor) son interpretadas en el navegador. Si hacen referencia a archivos CSS, JS o imágenes, esta información se solicita por parte del navegador a las rutas adecuadas, y el servidor envía los archivos en cuestión.

Ejemplo

1. Desde el navegador tecleamos en la barra de direcciones una dirección del dominio example.com.
2. En las herramientas del desarrollador de mi navegador se me muestra información sobre la petición HTTP, la respuesta, la dirección IP remota, etc.
3. Por ejemplo, se envía una petición HTTP desde el navegador:

```
GET /wiki/World_Wide_Web HTTP/1.1
Host: example.org
```

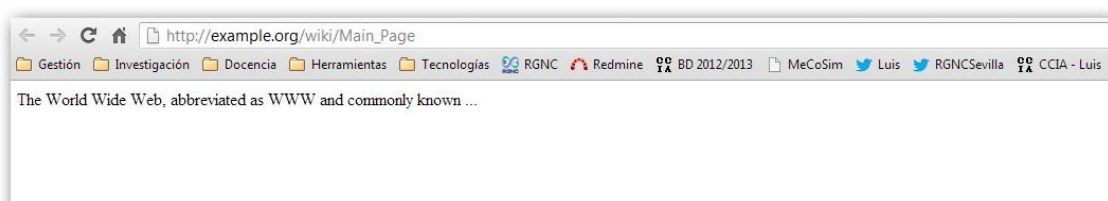
4. Se recibe la respuesta HTTP.

```
HTTP/1.0 200 OK
Content-Type: text/html; charset=UTF-8
```

Seguida de la página HTML:

```
<html>
<head>
<title>Example.org - The World Wide Web</title>
</head>
<body>
<p>The World Wide Web, abbreviated as WWW and commonly known ...</p>
</body>
</html>
```

5. El navegador analiza el código HTML. En este caso, no hace referencia a ningún archivo adicional, tal como imágenes, etc, luego no requiere peticiones HTTP adicionales al servidor.
6. Finalmente, la página es renderizada en el navegador:



Servidor Web

Como hemos visto en los apartados anteriores, hay dos actores cruciales en la web: el **cliente**, que trabaja a través del navegador realizando peticiones de páginas web y otros archivos, y el **servidor**, que procesa las peticiones y las respuestas.

En el ejemplo anterior, hemos actuado como clientes, realizando una petición HTTP (de forma transparente a nosotros, al ser desencadenada por nuestro navegador) y obteniendo una respuesta HTML, visualizada en el navegador.

Nuestro trabajo en este cuatrimestre será situarnos del lado del servidor, proporcionando las páginas a través de las que los clientes podrán realizar peticiones a nuestro servidor, tanto para servirle páginas HTML estáticas, como para procesar peticiones que generen páginas dinámicas.

Para poder poner a disposición las páginas de forma que los clientes puedan realizar peticiones, será necesario:

- Disponer de un **equipo** que actúe como **servidor**. Por ejemplo, un equipo habilitado por la Universidad de Sevilla (pendiente de verificar los términos y condiciones), un servidor de Internet que nos dé el servicio de hosting gratuito (en ambos casos proporcionando direcciones IP públicas), y posiblemente dominio gratuito (es decir, nombre asociado a la dirección IP); o bien nuestro propio equipo, que aceptará peticiones de sí mismo (reconocido como localhost).
- Disponer de un servicio/programa que actúe como servidor web. Por ejemplo, el servidor web **Apache** que viene con **XAMPP** y que hemos instalado en nuestro equipo, o los equivalentes en los servicios proporcionados por terceros. Este servidor deberá poder alojar, al menos: páginas HTML (junto con CSS, JS, imágenes, etc.); y páginas dinámicas de servidor (PHP, NodeJS, Python, ASP.NET, JSP, etc.) Para los fines que nos ocupan y para el uso más general, podrá incluir en el mismo equipo o en otro diferente un servidor de bases de datos, tal como MySQL/MariaDB, o como MongoDB, y la capacidad de ejecutar código de consulta sobre las bases de datos desde las páginas dinámicas. Al margen de los servidores locales que usaremos, se contemplará el uso de otras opciones gratuitas online como **heroku**.
- Crear las páginas web a alojar. Para ello, podremos usar editores tales como **VisualStudioCode** o **Notepad++**. También se podrán usar editores visuales que nos proporcionen gran parte del código, de forma que nos liberen de parte del trabajo y nos centremos en aquello que es propio de nuestra web. No obstante, es conveniente dominar los entresijos de las distintas tecnologías a emplear, y si una vez dominamos los distintos lenguajes decidimos usar herramientas que nos ahorran parte del trabajo, al menos entenderemos bien lo que hacen o cómo modificar el resultado que proporcionan.
- **Alojar las páginas** deseadas **en el servidor**. Junto a ellas, cualesquiera otros archivos necesarios, como imágenes que deban aparecer en las páginas, etc. En el caso de que el servidor esté en nuestro equipo, bastará con copiar el contenido correspondiente en la carpeta destinada por el servidor web a tal efecto. En el caso de que el servidor web y el hosting correspondiente no sea propio sino proporcionado por terceros, podremos acceder a través de protocolos tales como SFTP, SSH..., facilitados por programas tales como **WinSCP**.

En adelante, iremos trabajando en nuestro equipo con **XAMPP**. Nuestro dominio será por tanto **localhost**, con su correspondiente dirección IP privada fija **127.0.0.1**. Podemos usar cualquier editor de textos para el código HTML, JS, PHP, etc, si bien recomendamos **VisualStudioCode**. Una vez comencemos a conectarnos con la base de datos MySQL/MariaDB usaremos la aplicación web preinstalada en XAMPP para acceso a la BD, **phpMyAdmin**, o bien otras herramientas instaladas como MySQL Workbench.

En el tema siguiente veremos una introducción al lenguaje HTML, tratando de comprender los fundamentos básicos del lenguaje y comenzando a emplearlo para desarrollar nuestras primeras páginas web básicas.

Páginas Web

Las páginas Web se identifican unívocamente mediante su dirección URL:

- <http://www.cs.us.es/cursos/bd-2023>
- <http://www.us.es>

En la dirección URL de una página Web se distinguen:

- El protocolo de comunicación, como **http://**
- El nombre del servidor que contiene la página: www.cs.us.es
- La localización de la página en el servidor: `/cursos/bd-2023`

El lenguaje HTML

HTML: Lenguaje de Marcado de Hipertexto.

Es un lenguaje descriptivo diseñado para escribir páginas Web.

Las marcas indican cómo se debe presentar el contenido de una página en el navegador, pero el diseño final depende de este último. Debería ser estándar, independiente del navegador empleado, pero esto no siempre está garantizado.

Se puede escribir código HTML en cualquier ordenador o sistema operativo, mediante editores de texto, editores visuales y otros. Descartamos los editores visuales para comprender mejor los lenguajes a estudiar y su funcionamiento. En el futuro podremos optar por probar algunas opciones visuales, una vez adquirido el conocimiento fundamental.