

# ÍNDICE

<b>Prefacio</b>	<b>15</b>
<b>1 Hardware</b>	<b>19</b>
1.1 Objetivos . . . . .	19
1.2 Introducción . . . . .	19
1.3 Arquitectura y diseño lógico . . . . .	20
1.3.1 Funciones del computador . . . . .	21
1.3.2 Diseño lógico . . . . .	22
1.3.3 Codificación de la información . . . . .	23
1.4 La CPU . . . . .	28
1.4.1 Diseño de instrucciones CISC y RISC . . . . .	31
1.4.2 Computación en paralelo . . . . .	31
1.5 La memoria . . . . .	33
1.5.1 Registros . . . . .	36
1.5.2 Memoria caché . . . . .	36
1.5.3 Memoria principal . . . . .	36
1.5.4 Memoria secundaria . . . . .	37
1.5.5 Formato de almacenamiento de datos . . . . .	37
1.5.6 Memoria virtual . . . . .	39
1.6 Características de la memoria . . . . .	39
1.7 Duración de la información de las memorias y soporte físico . . . . .	42

1.7.1	Memoria de semiconductor . . . . .	43
1.7.2	Memoria magnética . . . . .	45
1.7.3	Memoria óptica . . . . .	48
1.7.4	La BIOS . . . . .	49
1.8	El bus . . . . .	50
1.8.1	Buses internos . . . . .	52
1.8.2	Buses de expansión . . . . .	53
1.9	Periféricos y conexiones . . . . .	53
1.9.1	Métodos de control de dispositivos de E/S . . . . .	55
1.9.2	Conexiones serie y paralelo . . . . .	56
1.10	Estado actual del tema . . . . .	56
1.10.1	La CPU . . . . .	56
1.10.2	Las memorias . . . . .	57
1.10.3	Los buses . . . . .	57
1.10.4	Conectividad . . . . .	58
1.11	Lecturas recomendadas . . . . .	58
1.12	Ejercicios de autocomprobación . . . . .	58
1.13	Actividades propuestas . . . . .	63
<b>2</b>	<b>Puertos de conexión y periféricos</b>	<b>65</b>
2.1	Objetivos . . . . .	65
2.2	Introducción . . . . .	65
2.3	Interacción con el ordenador . . . . .	67
2.3.1	Uso del teclado y el ratón . . . . .	70
2.3.2	La conexión plug-and-play . . . . .	73
2.3.3	Pantallas de ordenador . . . . .	75
2.3.4	Impresión de archivos . . . . .	79
2.4	Elementos interactivos . . . . .	81

2.4.1	Pantallas táctiles . . . . .	82
2.4.2	Tabletas digitalizadoras . . . . .	83
2.4.3	Pizarras interactivas . . . . .	84
2.4.4	Mandos para juegos . . . . .	85
2.5	Dispositivos de entrada ópticos . . . . .	87
2.5.1	Escáner y reconocimiento óptico de caracteres . . . . .	87
2.5.2	Lector de código de barras . . . . .	87
2.5.3	Lector de huella digital . . . . .	88
2.6	Dispositivos audiovisuales . . . . .	88
2.6.1	Cámaras digitales y tarjetas de captura de vídeo . . . . .	88
2.6.2	Micrófonos y altavoces . . . . .	89
2.6.3	Reconocimiento de voz . . . . .	89
2.6.4	Proyectores y dispositivos de proyección . . . . .	90
2.7	Puertos de conexión . . . . .	91
2.7.1	Conexión serie por cable . . . . .	92
2.7.2	Interconexión de periféricos de audio y vídeo . . . . .	96
2.7.3	Conexión paralela por cable . . . . .	98
2.7.4	Puertos inalámbricos . . . . .	101
2.8	Sensores . . . . .	103
2.8.1	Etiquetas RFID . . . . .	104
2.8.2	Internet de las cosas . . . . .	105
2.8.3	Wearables . . . . .	106
2.8.4	Gafas 3D . . . . .	107
2.8.5	Plataformas de hardware libre con código abierto . . . . .	108
2.9	Aspectos de ergonomía . . . . .	110
2.10	Estado actual del tema . . . . .	112
2.11	Lecturas recomendadas . . . . .	114

2.12	Ejercicios de autocomprobación . . . . .	114
2.13	Actividades propuestas . . . . .	119
<b>3</b>	<b>Sistemas Operativos</b>	<b>121</b>
3.1	Objetivos . . . . .	121
3.2	Introducción . . . . .	122
3.2.1	Vistiendo la máquina . . . . .	123
3.2.2	Programas y procesos . . . . .	125
3.3	Interacción con el hardware . . . . .	129
3.3.1	Modos de ejecución . . . . .	129
3.3.2	Interrupciones y excepciones . . . . .	130
3.3.3	Llamadas al sistema . . . . .	132
3.3.4	Memoria virtual . . . . .	133
3.3.5	Controladores software o <i>drivers</i> de dispositivo . . . . .	133
3.4	Clasificación de los sistemas operativos . . . . .	134
3.4.1	Clasificación por el número de tareas . . . . .	134
3.4.2	Clasificación según el número de usuarios . . . . .	136
3.4.3	Clasificación por el tipo de aplicación . . . . .	138
3.5	Servicios básicos de un SO . . . . .	142
3.5.1	Sistemas de archivos . . . . .	142
3.5.2	Gestión de memoria . . . . .	147
3.5.3	Control de procesos . . . . .	147
3.5.4	Comunicación y sincronización entre procesos . . . . .	149
3.6	Interacción con el usuario . . . . .	151
3.6.1	Interfaz de línea de órdenes . . . . .	151
3.6.2	Interfaz gráfica de usuario . . . . .	152
3.7	Otros servicios del Sistema Operativo . . . . .	157
3.7.1	Seguridad del Sistema Operativo . . . . .	158

3.7.2	Mantenimiento . . . . .	161
3.8	Evolución de los Sistemas Operativos . . . . .	163
3.8.1	Familia UNIX . . . . .	164
3.8.2	Sistemas Operativos Windows . . . . .	165
3.8.3	Apple . . . . .	171
3.8.4	Linux . . . . .	174
3.8.5	SO de dispositivos móviles . . . . .	175
3.9	Estado actual del tema . . . . .	177
3.9.1	Windows 10 . . . . .	177
3.9.2	Mac OS 10.12 Sierra . . . . .	177
3.9.3	Linux . . . . .	177
3.10	Lecturas recomendadas . . . . .	179
3.11	Ejercicios de autocomprobación . . . . .	180
3.12	Actividades propuestas . . . . .	185
<b>4</b>	<b>Programación y desarrollo de Software</b>	<b>189</b>
4.1	Objetivos . . . . .	189
4.2	Introducción . . . . .	189
4.3	Máquinas programables . . . . .	190
4.3.1	Computadores . . . . .	191
4.4	Programas de computadora . . . . .	193
4.5	Lenguajes de programación . . . . .	195
4.5.1	Compiladores e intérpretes . . . . .	197
4.5.2	Evolución de los lenguajes de programación . . . . .	198
4.5.3	Entornos de desarrollo . . . . .	201
4.6	Programación estructurada . . . . .	206
4.6.1	Elementos de un programa estructurado . . . . .	207
4.7	Construcción de datos . . . . .	225

4.8	Programación orientada a objetos . . . . .	227
4.8.1	Clases y objetos . . . . .	228
4.8.2	Atributos y métodos de una clase . . . . .	229
4.8.3	Herencia y polimorfismo . . . . .	232
4.9	Desarrollo de software. El ciclo de vida . . . . .	233
4.9.1	¿Qué es el software? . . . . .	234
4.9.2	El ciclo de vida del software . . . . .	234
4.9.3	Fases del ciclo de vida del software . . . . .	235
4.9.4	Documentos que se generan en el ciclo de vida . . . . .	237
4.10	Estado actual del tema . . . . .	239
4.11	Lecturas recomendadas . . . . .	241
4.12	Ejercicios de autocomprobación . . . . .	241
4.13	Actividades propuestas . . . . .	246
<b>5</b>	<b>Bases de datos y sistemas de información</b>	<b>247</b>
5.1	Objetivos . . . . .	247
5.2	Introducción . . . . .	247
5.3	¿Qué es una base de datos? . . . . .	248
5.3.1	Elementos básicos de una base de datos . . . . .	249
5.3.2	Operaciones básicas en una base de datos . . . . .	250
5.3.3	Bases de datos relacionales . . . . .	253
5.3.4	BBDD de archivos múltiples y relacionados . . . . .	255
5.3.5	¿Cómo se construye una base de datos? . . . . .	258
5.3.6	Construcción de una base de datos con MS-Access . . . . .	259
5.3.7	Bases de datos multiusuario . . . . .	262
5.3.8	SQL . . . . .	265
5.4	Sistemas de información . . . . .	266
5.4.1	Tipos de Sistemas de información . . . . .	267

5.4.2	Hardware de Sistemas de información . . . . .	270
5.4.3	Sistemas de información en Internet . . . . .	271
5.4.4	Desarrollo de sistemas de información. Ciclo de vida . . . . .	272
5.5	Estado actual del tema . . . . .	274
5.5.1	Microsoft Access . . . . .	274
5.5.2	MySQL . . . . .	274
5.5.3	Oracle . . . . .	275
5.6	Lecturas recomendadas . . . . .	275
5.7	Ejercicios de autocomprobación . . . . .	275
5.8	Actividades propuestas . . . . .	280
<b>6</b>	<b>Software de usuario</b>	<b>281</b>
6.1	Objetivos . . . . .	281
6.2	Introducción . . . . .	281
6.3	Programas de procesamiento de textos . . . . .	285
6.3.1	Conceptos básicos . . . . .	287
6.3.2	Otros programas de texto . . . . .	293
6.4	Programas de hojas de cálculo . . . . .	296
6.4.1	Conceptos básicos . . . . .	297
6.4.2	Otros programas de cálculo . . . . .	304
6.5	Programas de presentaciones . . . . .	305
6.6	Programas de imágenes . . . . .	307
6.6.1	Modo de color . . . . .	309
6.6.2	Profundidad de color . . . . .	310
6.6.3	Compresión de imágenes . . . . .	311
6.7	Programas de audio y vídeo . . . . .	311
6.7.1	Compresión de audio y vídeo . . . . .	313
6.8	Estado actual del tema . . . . .	315

6.8.1	Programas de procesamiento de textos . . . . .	315
6.8.2	Programas de hojas de cálculo . . . . .	316
6.8.3	Programas de presentaciones . . . . .	316
6.8.4	Programas de imágenes . . . . .	316
6.8.5	Programas de audio y vídeo . . . . .	318
6.9	Lecturas recomendadas . . . . .	319
6.10	Ejercicios de autocomprobación . . . . .	319
6.11	Actividades propuestas . . . . .	324
<b>7</b>	<b>Redes y conectividad</b>	<b>327</b>
7.1	Objetivos . . . . .	327
7.2	Introducción . . . . .	327
7.3	Redes de ordenadores . . . . .	329
7.4	Para qué sirve una red . . . . .	329
7.4.1	Compartir archivos . . . . .	330
7.4.2	Compartir impresoras . . . . .	331
7.4.3	Compartir un fax . . . . .	332
7.4.4	Copias de seguridad . . . . .	333
7.4.5	Colaboración en red . . . . .	335
7.5	Transmisión de datos . . . . .	335
7.5.1	Tipos de señales . . . . .	336
7.5.2	Componentes de una red . . . . .	337
7.6	Características de las redes . . . . .	338
7.6.1	Cobertura geográfica . . . . .	338
7.6.2	Medios de transmisión . . . . .	339
7.6.3	Topologías de red . . . . .	346
7.7	Tecnología y Cableado . . . . .	349
7.7.1	Red Local Ethernet . . . . .	349

7.7.2	Redes locales de alta velocidad . . . . .	351
7.7.3	Tecnologías inalámbricas . . . . .	353
7.8	Red Internet . . . . .	355
7.8.1	Conexión a Internet . . . . .	355
7.8.2	Direccionamiento IP . . . . .	356
7.8.3	Encaminamiento de mensajes . . . . .	358
7.8.4	Servicio de nombre de dominio DNS . . . . .	359
7.8.5	Telnet . . . . .	360
7.8.6	Aspectos básicos de seguridad . . . . .	360
7.9	El papel de la regulación y los estándares . . . . .	362
7.9.1	Protocolo OSI . . . . .	363
7.9.2	Protocolos Internet TCP/IP . . . . .	366
7.10	Dispositivos de interconexión . . . . .	367
7.10.1	Concentradores . . . . .	368
7.10.2	Conmutadores . . . . .	369
7.10.3	Repetidores . . . . .	369
7.10.4	Puentes . . . . .	369
7.10.5	Encaminadores . . . . .	370
7.10.6	Pasarelas . . . . .	370
7.10.7	Modems y tarjetas de red . . . . .	371
7.11	Tecnologías avanzadas de telecomunicación . . . . .	372
7.11.1	Voz sobre IP . . . . .	373
7.11.2	Computación en la nube . . . . .	373
7.11.3	Tecnologías de Big Data . . . . .	377
7.12	Estado actual del tema . . . . .	378
7.13	Lecturas recomendadas . . . . .	380
7.14	Ejercicios de autocomprobación . . . . .	380

7.15	Actividades propuestas . . . . .	385
<b>8</b>	<b>Internet y seguridad informática</b>	<b>387</b>
8.1	Objetivos . . . . .	387
8.2	Introducción . . . . .	387
8.3	Servicios básicos de Internet . . . . .	388
8.3.1	Navegación . . . . .	388
8.3.2	Correo electrónico . . . . .	393
8.3.3	Transferencia de ficheros . . . . .	395
8.4	Internet en la sociedad . . . . .	396
8.4.1	Educación . . . . .	397
8.4.2	Comercio electrónico . . . . .	397
8.4.3	Noticias . . . . .	398
8.5	Internet interactivo . . . . .	398
8.5.1	Comunicación instantánea . . . . .	399
8.5.2	Blogs . . . . .	400
8.5.3	Redes sociales . . . . .	400
8.6	Lenguajes de la web . . . . .	401
8.6.1	HTML . . . . .	402
8.6.2	JavaScript . . . . .	404
8.6.3	CSS . . . . .	405
8.7	Seguridad . . . . .	407
8.7.1	Amenazas a usuarios . . . . .	408
8.7.2	Amenazas al hardware . . . . .	411
8.7.3	Amenazas a los datos . . . . .	413
8.8	Estado actual del tema . . . . .	414
8.8.1	Servicios básicos de Internet . . . . .	414
8.8.2	Internet en la sociedad . . . . .	414

8.8.3	Internet interactivo . . . . .	415
8.8.4	Lenguajes de la web . . . . .	418
8.8.5	Seguridad . . . . .	418
8.9	Lecturas recomendadas . . . . .	418
8.10	Ejercicios de autocomprobación . . . . .	419
8.11	Actividades propuestas . . . . .	423
<b>Índice alfabético</b>		<b>425</b>

## TEMA 8

# INTERNET Y SEGURIDAD INFORMÁTICA

*Autor: Emilio Letón Molina*

### 8.1. OBJETIVOS

En este capítulo se pretenden conseguir los siguientes objetivos:

- Conocer los servicios básicos de Internet de navegación, correo electrónico y transferencia de ficheros.
- Ser conscientes de la importancia de Internet en la sociedad a través de sus aplicaciones a la educación y al comercio electrónico.
- Asumir el papel fundamental en la interacción social entre personas a través de sistemas de comunicación instantánea, blogs y redes sociales.
- Aprender una sintaxis mínima de ciertos lenguajes de la Web como HTML, JavaScript y CSS.
- Conocer los conceptos básicos relativos a seguridad informática en relación a los usuarios, al hardware y a los datos.

### 8.2. INTRODUCCIÓN

La importancia social de Internet en la actualidad es abrumadora: Internet está presente en prácticamente todos los ámbitos de nuestra vida. En los orígenes de Internet, como se introdujo en la sección 7.8, sólo existían unos pocos servicios disponibles, entre los que destacaban la navegación, el correo electrónico y la transferencia de ficheros. Hay que destacar que al principio, el uso de estos servicios no era fácil ya que no había software amigable para hacer dichas tareas. Sin embargo, hoy en día, el uso de los servicios de Internet es fácil e intuitivo por lo que todo el mundo puede utilizarlo. Además, como se verá en este capítulo, los servicios de Internet se han extendido y hoy se utilizan tanto en el ámbito educativo como en

el de comercio electrónico o simplemente para comunicarnos con personas de todo el mundo. También se mostrarán algunos aspectos técnicos relativos a los conceptos básicos de algunos de los lenguajes que se utilizan en Internet, que aunque no son necesarios aprender para utilizar Internet, sí lo son para poder conocer mejor sus posibilidades. Por último, se tratarán algunos aspectos en relación a que existen posibles problemas de seguridad asociados al uso de Internet que hay que tener en cuenta y que conviene conocer para evitarlos.

### 8.3. SERVICIOS BÁSICOS DE INTERNET

En esta sección se verán los principales servicios básicos con los que contaba Internet y que todavía se usan (aunque de forma muy distinta): la navegación, el correo electrónico y la transferencia de ficheros.

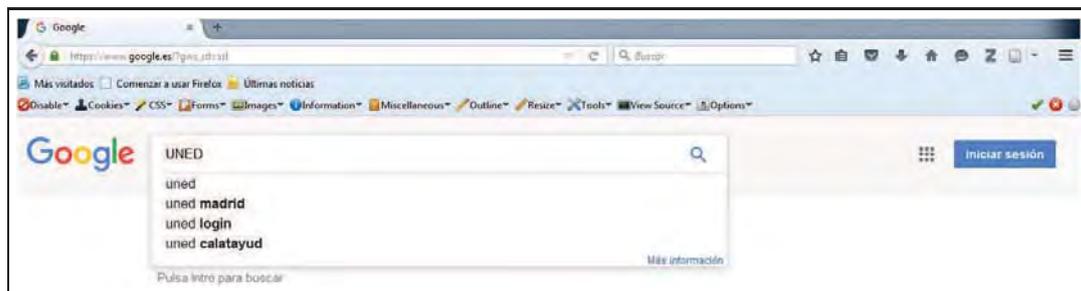
#### 8.3.1. Navegación

El servicio navegación web, llamado también simplemente web o www (World Wide Web), es el servicio más usado de Internet. En este servicio el elemento básico son las páginas web que permiten ver información en forma de texto, tablas, imágenes, sonido y/o vídeos. Además las páginas web permiten acceder a otras páginas web, lo que se conoce con el término de navegar. El software que permite ver las páginas web y navegar por ellas se llama browser o navegador (ver la figura 8.1 que muestra la página principal de la UNED con el navegador Firefox).



Figura 8.1: Página principal de la UNED con el navegador Firefox

A menudo es necesario buscar información en Internet, esto se hace con la ayuda de un software que se denomina buscador. En los orígenes de Internet buscar información era muy difícil ya que la única forma con la que se podía buscar era consultando directorios que eran páginas web donde se recopilaba información de forma manual. Hoy en día los buscadores, como Google o Bing, son auténticos robots (“bots”) que permiten búsquedas fáciles y rápidas. En la figura 8.2 se muestra una búsqueda con Google para encontrar la página principal de la UNED. Estas consultas se pueden hacer por búsqueda de palabras de aparición desordenada o por búsqueda exacta de aparición de palabras, para lo cual hay que ponerlas entre comillas.



**Figura 8.2:** Ejemplo de búsqueda con Google para encontrar la página principal de la UNED

Las páginas web se construyen, principalmente, con el lenguaje HTML (que será veré más adelante en este mismo capítulo en la sección 8.6) y se alojan en servidores (ver capítulo 7). La web está supervisada por The World Wide Web Consortium (W3C), entidad sin ánimo de lucro financiada por un número importante de miembros corporativos y que tiene por objetivo principal desarrollar estándares tecnológicos disponibles para todos, que garanticen el crecimiento homogéneo de la Web. El hecho de trabajar con estándares facilita a programadores y diseñadores la compatibilidad necesaria para crear sitios Web que puedan ser descargables desde cualquier dispositivo o plataforma.

**Ejemplo 1** Buscar información sobre el Consorcio Web. Para buscar esta información, se utiliza un buscador y se hace la búsqueda de forma análoga a como se hace en la figura 8.2 pero utilizando “Consorcio Web” que obliga a que ésta sea la forma exacta de aparición.

## **URL**

Las URL (Uniform Resource Locator) son localizadores uniformes de recursos. En Internet, un recurso es cualquier cosa que se puede encontrar, por ejemplo una página web, una imagen o un programa a descargar, el resultado de ejecutar un

programa o la imagen de una webcam. No obstante, la mayoría de los recursos que se pueden encontrar en Internet son páginas web. La URL a la que se desee visitar, debe ser tecleada en la caja de texto ubicada en la barra de navegación del navegador. El formato general de una URL es:

esquema://maquina.dominio/ruta/archivo

Los esquemas más comunes de URL (ver capítulo 7 para más información) son,

- http: para recursos Hypertext Transfer Protocol.
- https: es http secure, http seguro, sobre SSL (Secure Sockets Layer).
- ftp: File Transfer Protocol.
- mailto: para direcciones de correo electrónico o e-mail.
- file: para recursos disponibles locales.
- news: grupo de noticias Usenet.
- Telnet: para acceder a otras máquinas.

La parte de la URL desde “//” hasta la primera barra “/”, es decir maquina.dominio, constituye la dirección IP, que ya se introdujo en la sección 7.8.2 (y se puede escribir en mayúsculas o en minúsculas indistintamente) y la parte de la URL desde la primera barra “/” hasta el final, marca la ruta del recurso en dicha IP (aquí la ruta es sensible a las mayúsculas y minúsculas porque ya depende del sistema operativo de la IP). La parte de la IP conviene leerla de derecha a izquierda. Así, la parte más próxima a la primera “/” corresponde al dominio de nivel superior o primer nivel.

Existen distintos tipos de dominios de nivel superior. Los más utilizados son los dominios de nivel superior geográficos y los genéricos. La máquina o host suele tener el nombre “www”, ya que por razones históricas a los servidores que servían páginas web se les denotaba por “www”, aunque no siempre es así, ya que dentro de un dominio puede haber distintos servidores.

La dirección IP, que ya se introdujo en la sección 7.8.2, es realmente la dirección que utilizan los ordenadores. Lo que sucede es que como son difíciles de recordar se le asocian los caracteres de máquina y dominio que constituyen el DNS (“Domain Name System”) o sistema de nombres de dominio.

**Ejemplo 2** Dominios geográficos y genéricos. Algunos ejemplos son,

- Dominios geográficos o ccTLD (Country code top-level domain): “.es” (España), “.ar” (Argentina), “.at” (Austria), “.cx” (Isla de Navidad), “.tv” (Islas Tuvalu), “.eu” (Europa) o “.us” (EEUU).
- Dominios genéricos o gTLD (Generic top-level domain): “.com” (Comercial), “.info” (Información), “.net” (redes y proveedores de Internet), “.org” (organizaciones y asociaciones), “.cat” (cultura y lengua catalanas), “.edu” (Educación), “.gov” (Gobierno de EEUU) o “.mil” (Fuerzas armadas de EEUU).
- Dominios compuestos: en algunas ocasiones el dominio de primer nivel es compuesto, por ejemplo, “gob.es” (Gobierno de España), “ac.uk” (instituciones académicas del Reino Unido) o “edu.co” (instituciones académicas de Colombia).

**Ejemplo 3** Localizadores uniformes de recursos. Los siguientes ejemplos muestran distintas URL,

- <http://www.uned.es>  
En este caso el dominio es uned.es y el servidor es www. Dado que hay un index.html, en este ejemplo se muestra el contenido de dicha página que es el dado por la página web principal de la UNED. Sería equivalente a escribir <http://62.204.192.16>, por lo tanto su IP es 62.204.192.16.
- <http://www.uned.es/index.html>  
En este caso se va explícitamente a la página web index.html que está en www.uned.es.
- <http://www.ia.uned.es/minivideos/publicaciones.html#h-premios>  
En este ejemplo ia es un subdominio del dominio principal uned.es y #h-premios es un marcador (indicador o ancla # que permite ir directamente a un lugar de una página html, siempre y cuando se haya definido dicho marcador en la página web correspondiente) dentro de la página minivideos/publicaciones.html. Naturalmente, otra forma de llegar a esta información es tecleando la URL <http://www.ia.uned.es/minivideos> e ir haciendo clic en los enlaces de la página que se muestran en ella, es decir navegando.
- <https://es.wikipedia.org/wiki/Especial:Buscar?search=url&go=Go>

En este caso se tiene que `wiki/Especial:Buscar` es la ruta del recurso y que `?search=url&go=Go` es la cadena de búsqueda. Esta URL es un recurso que busca información sobre las cadenas de carácter “url” (por eso ? de “query” (consulta) igual a url). En este ejemplo la URL, el recurso, no es propiamente una página web, sino que es el resultado de la ejecución de un programa al que se le pasan una serie de parámetros. En este caso se puede observar la presencia de un candado cerrado junto a la URL que indica que es una página web segura.

**Ejemplo 4** Registrar un dominio. En este ejemplo se van a ver los pasos principales para crear y comprar un dominio en Internet, que será `emilioleton.com`, para contribuir a crear una marca o presencia digital.

En primer lugar hay que comprobar que el dominio está libre en Internet. Una forma sencilla de realizar esta comprobación es tecleando la URL “`http://www.emilioleton.com`” en un navegador. Si dicha URL no existe, el navegador devolverá un mensaje del estilo: “Se encontró el siguiente error al intentar recuperar la URL “`http://www.emilioleton.com/`”, incapaz de determinar la dirección IP a partir del nombre de la máquina `www.emilioleton.com`, el servidor DNS devolvió: Name Error: The domain name does not exist”. Otra forma de comprobar la existencia del dominio es ir a un proveedor de Internet y hacer la consulta como ilustra la figura 8.3.

Una vez que se está seguro de que el dominio está libre, se puede comprar utilizando algunas de las formas de pago que se ilustran en la sección 8.4.2 sobre Comercio electrónico. Además del dominio hace falta contratar como mínimo un hospedaje o hosting, es decir un sitio físico en un servidor donde poder almacenar ficheros de cualquier tipo, ya sean html o ficheros de tipo imagen o vídeo. El último paso es crear la página `index.html` para que cuando alguien teclee la URL “`http://www.emilioleton.com`” se muestre la página principal. Mientras no se cree dicha página se mostrará por defecto una página similar a la que aparece en la figura 8.4 que además informa de que el dominio ya no está disponible. Al registrar un dominio hay que facilitar una serie de datos personales que justifiquen una relación con el nombre del dominio. Antiguamente cualquiera podría registrar un dominio sin acreditar una relación con él (es famoso el caso de Coca-Cola, donde tuvo que mediar el Centro de Mediación y Arbitraje de Dominios de la Organización Mundial de

la Propiedad Intelectual (OMPI)), lo cual ahora ya no es posible. Sin embargo, todavía se encuentran dominios que intentan dar la impresión de que son oficiales cuando en realidad no lo son.

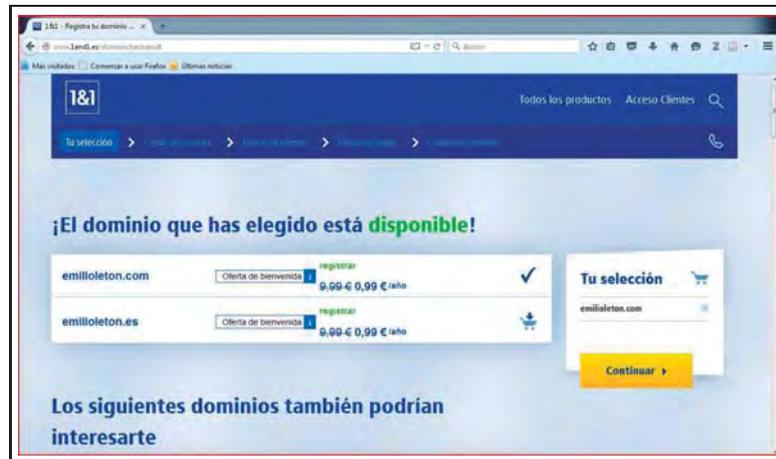


Figura 8.3: Comprobación de existencia de dominio con la Empresa 1&1



Figura 8.4: Ejemplo de dominio registrado

### 8.3.2. Correo electrónico

El servicio de correo electrónico (correo-e o e-mail) fue uno de los primeros servicios que tuvo Internet. En sus orígenes era complicado su manejo hasta el punto que no era fácil, por ejemplo, incluir un fichero junto a un mensaje. Con el tiempo su uso se fue extendiendo y su manejo haciéndose más sencillo.

Mediante el correo electrónico los usuarios envían y reciben mensajes que se componen de texto y que pueden incluir ficheros de cualquier tipo (imágenes, vídeo, audio o incluso software) como información adjunta. Para utilizar este servicio los usuarios tienen que tener una cuenta de correo electrónico. Hoy en día es habitual que todo el mundo tenga una cuenta de correo electrónica e incluso varias.

Una cuenta de correo electrónico tiene el esquema nombre@dominio. El símbolo @ se lee en castellano como arroba y en inglés como “at”, precisamente porque el usuario nombre está en (“at”) dominio (proveedor). Por ejemplo nombre.apellido@gmail.com o nombreapellido@hotmail.com pueden ser cuentas de correo personal, mientras que nombre.apellido@empresa.es puede ser una cuenta de trabajo profesional. Sin embargo, en la actualidad, los usuarios (sobre todo los más jóvenes) prefieren cada vez más comunicarse a través de opciones más interactivas (ver, más adelante, la sección 8.5) y están dejando de utilizar el servicio de correo electrónico.

Es importante tener presente los siguientes aspectos:

- Las cuentas de correo electrónico tienen que tener configurado un servidor de entrada (por ejemplo correo.uned.es) y uno de salida (por ejemplo smtp.uned.es o smtp.gmail.com). El servidor de salida gestiona el correo saliente y utiliza el protocolo Simple Mail Transfer Protocol (SMTP) y el servidor de entrada gestiona el correo entrante y puede utilizar el protocolo POP3 o el Internet Message Access Protocol (IMAP).
- El protocolo POP3 descarga los mensajes del servidor al cliente eliminándolos del servidor, por lo que hoy en día es preferible el IMAP que permite acceder a los mensajes del servidor desde varios dispositivos con conexión a Internet ya que los mensajes consultados siguen estando almacenados en el servidor.
- Conviene señalar que no hace falta que el receptor esté conectado para que el correo se reciba, ya que éste se almacena en el servidor hasta que el receptor se conecte.
- El servicio de correo electrónico se utiliza mediante un software cliente de correo electrónico, que es un programa de ordenador que se instala en el ordenador del usuario (cliente) para leer y enviar mensajes de correo electrónico. Otra posibilidad es utilizar un software que no se instale en el cliente sino que esté en la web (en ese caso dicho software recibe el nombre de webmail o correo web).

En la escritura del mensaje hay que rellenar los siguientes campos,

- Destinatarios: pueden ser una o varias direcciones de correo electrónico. Se puede incluso decidir quiénes son los destinatarios principales en el campo “Para”, los que conviene que estén informados de forma pública en el campo “cc” o “Con copia” (“Carbon Copy”) y los que estén informados de manera no pública en el campo “cco” o “Con copia oculta” (“bcc” o blind carbon copy).

- **Asunto:** es el nombre que se le da al mensaje. Conviene rellenarlo siempre y que sea corto y autoexplicativo. Rellenando este campo se facilita el almacenamiento de correos y también ayuda a evitar SPAM (ver más adelante, la sección 8.7.1).
- **Cuerpo:** es donde se escribe el mensaje propiamente dicho.

A la hora de recepcionar un mensaje se puede responder al que lo ha enviado, responder a todos los que aparecen como destinatarios visibles, archivarlo en un sistema de carpetas o de etiquetas, o borrarlo.

La forma mediante la cual los correos electrónicos viajan por Internet es utilizando el Registro Mail eXchange record o registro de intercambio de correo (MX) que consulta al DNS para saber en qué servidor está el dominio de los destinatarios y cuál es el mejor camino para llegar a ellos.

### 8.3.3. Transferencia de ficheros

A menudo es útil poder mandar y recibir archivos entre ordenadores. Para realizar esta tarea se puede utilizar el protocolo FTP (File Transfer Protocol; ver capítulo 7 para más información). La utilización de FTP requiere de un software cliente FTP y de un software servidor FTP. El programa cliente FTP es el encargado de establecer la conexión con el servidor. Cuando se establece la conexión FTP, el software servidor FTP solicita un nombre de usuario o user login y una contraseña o password, según se ve por ejemplo en la figura 8.5.

El protocolo FTP no es seguro ya que no utiliza información cifrada. Para realizar una transferencia segura se puede utilizar el protocolo SCP (Secure Copy Protocol) y el SFTP (Secure File Transfer Protocol) que utiliza SSH (Secure SHell). Un software cliente SFTP y un software cliente FTP son parecidos en el aspecto pero emplean protocolos distintos, por lo que no es posible conectar un software cliente FTP a un servidor SFTP.

En la actualidad, la mayoría de los clientes SFTP son gráficos, eso significa que su apariencia es la dada en la figura 8.6 donde se observa que en la parte de la izquierda están los archivos en local y en la parte de la derecha los archivos en remoto en el servidor. Con los clientes gráficos SFTP se pueden copiar archivos fácilmente de local a remoto simplemente seleccionando con el ratón los archivos y arrastrándolos a donde se deseen transferir.

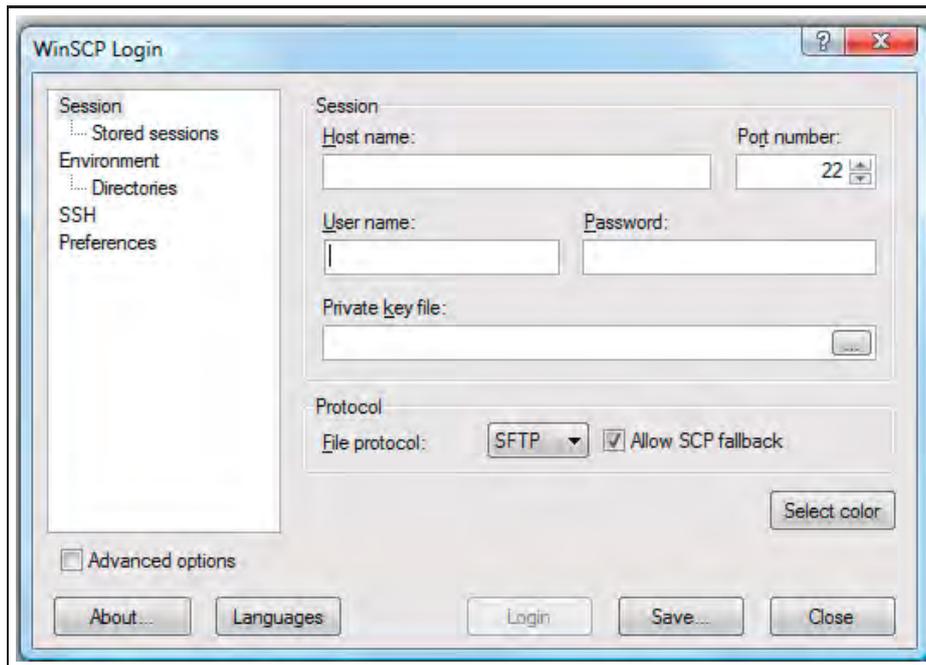


Figura 8.5: Ejemplo de login a WinSCP

Un ejemplo muy importante de transferencia de ficheros es la “subida” de los archivos de una página web a un servidor en Internet.

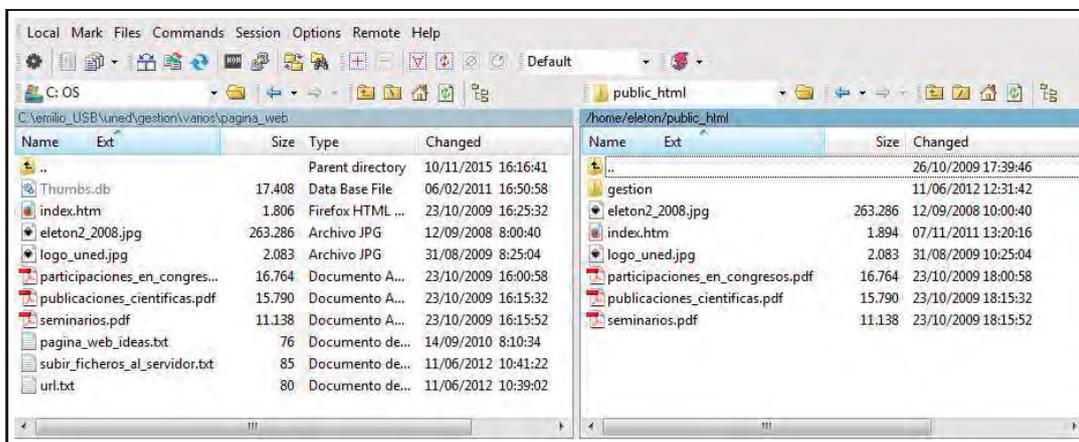


Figura 8.6: Ejemplo de transferencia de ficheros con WinSCP

#### 8.4. INTERNET EN LA SOCIEDAD

En esta sección se mostrarán algunos ejemplos donde Internet ha irrumpido socialmente: Educación, comercio electrónico y noticias.

### 8.4.1. Educación

Internet está suponiendo una auténtica revolución en el mundo educativo. Hoy en día es habitual realizar formación a distancia con la ayuda de Internet, con las ventajas que ello conlleva en términos de libertad física y de horarios. En la mayoría de estos cursos hay una plataforma educativa LMS (Learning Management System) donde están ubicados los contenidos, ya sea en formato texto, audio o vídeo, y el foro que es el lugar donde los estudiantes y profesores pueden intercambiar mensajes de una forma ordenada en hilos, que son formas de estructurar los mensajes sobre un mismo tema para que se puedan leer éstos de forma consecutiva de principio a fin.

Actualmente algunos de los cursos a distancia se encuentran también en formato abierto y gratuito OCW (Open Course Ware) o MOOC (Massive Open Online Course).

### 8.4.2. Comercio electrónico

En Internet es fácil realizar transacciones comerciales de empresa a consumidor o B2C (Business to consumer) y de empresa a empresa o B2B (Business to business). Estos tipos de transacciones constituyen lo que se conoce como e-commerce o comercio electrónico. En este libro se tratarán las transacciones B2C, que son la que más interesan desde un punto de vista de usuario.

**Ejemplo 5** Transacciones comerciales B2C. Ejemplos de B2C son la compra de un artículo u operaciones bancarias. Para comprar un artículo por Internet o comprar en línea, primero hay que encontrarlo en una página o sitio web en Internet, lo que sería algo análogo a lo que se hace en una compra no virtual, donde lo primero es encontrar lo que se quiere comprar. Estas páginas web donde se puede comprar reciben el nombre de tiendas en línea, en contraposición a las tiendas con local. En algunos casos existen también tiendas híbridas que mezclan parte de tienda con local y tienda en línea. Una vez encontrado el producto se selecciona y pasa a formar parte de un carrito de la compra virtual, que sería algo análogo al carrito de la compra real, y que constituye un área de almacenamiento virtual de artículos seleccionados que se pagan todos juntos al final mediante una tarjeta de crédito, tarjeta e-cash o por el sistema de pago PayPal (ver figura 8.7). Conviene señalar que un sentido

amplio comprar un artículo puede ser comprar un billete de avión o tren o incluso reservar una habitación en un hotel.

Otra forma de B2C es realizar operaciones bancarias. Las operaciones típicas bancarias en línea son consultar movimientos y saldos de cuenta, realizar transferencias y pagos de recibos o incluso crear una cuenta. En este último caso, es necesario enviar por correo normal o de forma presencial la documentación que solicite el banco.

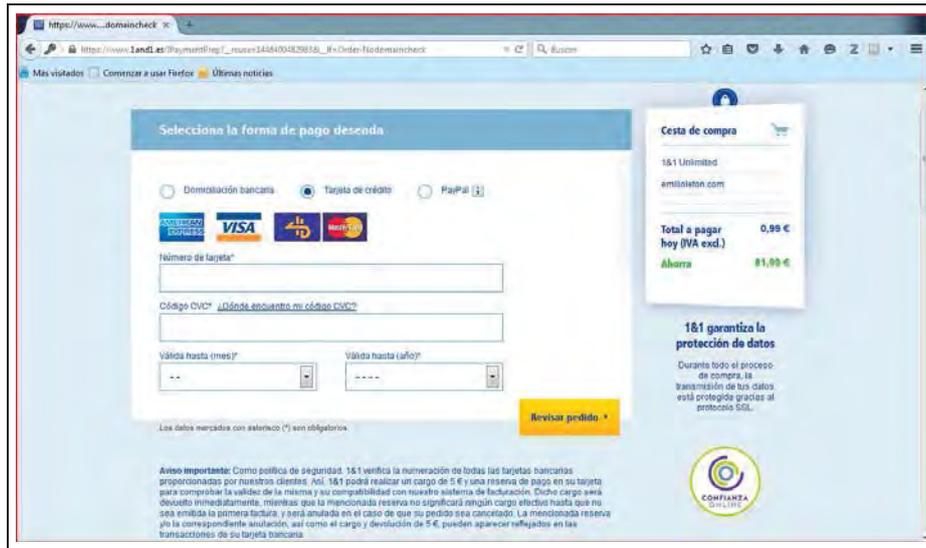


Figura 8.7: Formas de pago para comprar un dominio

### 8.4.3. Noticias

En la actualidad es muy común informarse de las noticias a través de Internet ya que existen, no sólo webs que dan noticias, sino incluso periódicos digitales. De hecho, algunos expertos vaticinan el fin próximo de la prensa escrita. Lo normal de los servicios de noticias es que sean gratuitos, aunque hay otros que permiten hacerse socio por una pequeña cuota anual que les permite servicios más exclusivos, como, por ejemplo disponer antes de los artículos de opinión.

## 8.5. INTERNET INTERACTIVO

En esta sección de Internet interactivo se verá, a modo de ejemplo entre las muchas posibilidades que Internet ofrece, la comunicación instantánea, los blogs y las redes sociales.

### 8.5.1. Comunicación instantánea

La comunicación instantánea es un tipo de servicio en Internet que permite la comunicación entre usuarios en tiempo real. Los mensajes que se transmiten pueden ser de tipo texto, imagen, sonido o incluso vídeo. Evidentemente, estos servicios requieren que las personas que deseen establecer una comunicación, utilicen el mismo software para intercambiar mensajes. Algunos ejemplos son,

- Chat: el término “Chat” es un término en inglés que significa charlar. Un chat o ciberchat, por tanto, es una forma de comunicación por Internet que emula una conversación espontánea entre 2, 3 ó más personas que puede ser público o privado. A veces los usuarios no muestran su nombre verdadero utilizando un apodo (pseudónimo) o “nick”. Una diferencia fundamental frente al correo electrónico es que en un chat los mensajes son muy cortos y se redactan con gran rapidez, lo que a menudo conlleva a economizar los caracteres y a cometer faltas de ortografía. Aunque el chat sea una forma informal de comunicación, conviene seguir unas normas de buen compartamiento o convivencia que se conocen como “Netetiqueta”.
- Voz sobre IP: VoIP (Voice over IP) es una tecnología que permite digitalizar la voz y transmitirla a través del protocolo de Internet IP, que ya se introdujo en la sección 7.8. La voz digitalizada se transmite en pequeños fragmentos o paquetes, que se dispersan en Internet eligiendo el camino más corto y que se recomponen en el destino. Un servicio análogo a VoIP es el ToIP (Telephony over IP) que es la telefonía por IP que conlleva grandes ventajas económicas ya que su coste es el coste de la conexión a Internet frente a la conexión telefónica convencional entre dos puntos distantes.
- Videoconferencia: la videoconferencia permite la comunicación entre usuarios en tiempo real a través de vídeo utilizando Internet. Su uso puede ser para comunicarse de manera informal o incluso profesional. El software de videoconferencia permite también compartir la pantalla de un ordenador, con lo que hace posible trabajar a distancia como si se estuviera de forma presencial. Un primer paso para establecer la comunicación es, como ya se ha dicho anteriormente, ser usuarios del mismo software (por ejemplo Skype o Hangouts). Un segundo paso es saber el nombre del contacto en dicho software. Para facilitar comunicaciones posteriores, es habitual guardar los contactos en una libreta de contactos que proporciona el software. Entre las opciones que tienen estos programas, cabe destacar una línea de chat para intercambiar mensajes de texto.

### 8.5.2. Blogs

Un blog es un sitio web en el que un usuario publica, a modo de diario personal o boletín electrónico, sus opiniones o contenidos que para él son interesantes. Estos contenidos, artículos o “posts”, suelen ser cortos y de carácter actual y novedoso. Los blogs son muy dinámicos, en el sentido de que se actualizan muy a menudo y permite además que otras personas publiquen en él.

El origen de la palabra blog proviene del inglés “Web log” (registro web o cuaderno de bitácora web) que posteriormente se escribió como “We blog” y que luego se acertó a “blog”. En la actualidad es un término muy utilizado, así como el de “blogger” para referirse a las personas que tienen y mantienen un blog.

### 8.5.3. Redes sociales

Una red social es una estructura, de ahí el nombre de red, que reúne a individuos que interactúan entre sí formando comunidades de personas con intereses o actividades en común, de ahí el nombre de social. Su origen se remonta al ámbito de las listas de distribución, los grupos de noticias y los foros web.

Con la ayuda de Internet, las redes sociales se han potenciado de forma que las comunidades son virtuales por lo que son de fácil creación y de fácil expansión. Para poder utilizar una red social hay que darse de alta en la red social, es decir hay que crearse una cuenta en dicha red social. Una vez que se ha creado una cuenta de una red social es posible interactuar con otros usuarios. En la mayoría de ellas para poder interactuar hay que mandar una petición a un usuario que ya esté dentro de la red social, aunque no siempre es ese el caso, ya que hay redes que permiten seguir a individuos sin que éstos lo autoricen. Hay que señalar que es muy normal que los usuarios de una red social no se conozcan entre sí y que sea gracias a dicha red social cuando se empiecen a conocer. También suele haber en las redes sociales unas normas de convivencia, existiendo la posibilidad de poder expulsar de una red a un usuario que no las cumpla.

En la figura 8.8 se muestra la interacción de un usuario en la red social Twitter.

Las redes sociales se utilizan con distintos propósitos, entre los que destacan,

- Intercambiar fotos, vídeos o cualquier tipo de material multimedia.
- Compartir opiniones.



Figura 8.8: Usuario @emilioleton en la red social Twitter

- Buscar trabajo.
- Buscar nuevas amistades.
- Estar en contacto con familiares o amigos.

En la actualidad las empresas y universidades también tienen presencia en las principales redes sociales y se ha creado la figura de Community Manager (CM) que sería un gestor de la comunidad generada por una red social. El CM gestiona las redes sociales para que sea una forma útil para la empresa o universidad de promocionarse y de darse a conocer. Asimismo, intenta proteger a la empresa o universidad de daños corporativos, tiene una escucha activa de la competencia y analiza el resultado de las estrategias realizadas para ver si están sirviendo para conseguir un posicionamiento eficaz en la web. Es un puesto de trabajo emergente dentro de la mercadotecnia y la publicidad en Internet.

El estudio de las interacciones entre usuarios en una red social ha dado lugar a una nueva disciplina conocida como Análisis de redes sociales y hay software específico para ello, ya que las cifras de usuarios en una red social alcanza fácilmente varios millones de usuarios al día.

## 8.6. LENGUAJES DE LA WEB

El lenguaje usual para la creación de páginas web se basa en HTML (HyperText Markup Language) o lenguaje de marcado o etiquetado hipertextual. Existen, no

obstante, otros lenguajes de marcas que combinados con HTML potencian a este, como por ejemplo MathML o MathJax para poder escribir expresiones matemáticas, JavaScript para dotar de interactividad o CSS para dar estilo. En los ejemplos de esta sección se anima al lector a que escriba con la ayuda de un editor de texto (ver el apartado 6.3.2) los comandos de cada ejemplo en un archivo de texto plano con extensión “htm” o “html” y que lo abra con un navegador para ver el resultado final.

### 8.6.1. HTML

El lenguaje HTML se escribe usando etiquetas o tags que vienen delimitadas por corchetes angulares. Existen etiquetas de inicio con sus respectivas etiquetas de fin que tienen la siguiente estructura:

- `<nombredeetiqueta>`
- `</nombredeetiqueta>`

Cuando un cliente hace una petición al servidor web utilizando un navegador, el servidor web responde al cliente enviando el código HTML de la página. El navegador interpreta el código HTML y lo muestra de forma legible para que lo visualice ya interpretado. Por eso una misma página web se puede ver de forma distinta por distintos navegadores, incluso puede verse de forma correcta en algunos navegadores e incorrecta en otros, y más aún, depender de la versión del navegador. Por eso es conveniente tener varios navegadores instalados y con sus versiones más actuales.

La primera descripción de HTML es debida a Tim Berners-Lee en 1991. A grandes rasgos las características del lenguaje HTML son:

- Es de fácil uso, ya que al estar basado en hipertexto se navega de manera intuitiva por los documentos que genera.
- Es dinámico y fácil de actualizar.
- Es compatible con todo tipo de ordenadores, dispositivos y sistemas operativos.
- Es descriptivo, ya que se define los elementos que tiene la página web, pero es el navegador en última instancia quien decide cómo mostrar dichos elementos. Es por ello que una misma página web no se verá igual en todos los navegadores.
- Permite incorporar la accesibilidad, haciendo que personas con discapacidad puedan utilizar también las páginas web.

Para elaborar un documento en HTML hay que realizar dos pasos.

- Editar: en este paso se crea un fichero de texto plano con extensión “htm” o “html” que tenga los comandos adecuados de HTML y el texto de la página web que se desee. Este fichero se puede hacer con un editor básico o con uno especializado en HTML.
- Compilar/Visualizar: este paso se hace de forma conjunta con un navegador que interpreta los comandos HTML y muestra lo que se ha entendido con dicho código. En el caso de que no se haya escrito correctamente el código HTML no se mostrará nada o se mostrará de forma parcial el contenido interpretado del fichero “htm” o “html”.

**Ejemplo 6** Texto normal y texto con superíndices en HTML. En este ejemplo se trata de generar en HTML una página web que tenga el aspecto de la figura 8.9. Para ello habrá que escribir en un archivo, con la ayuda de un editor de texto, el código HTML dado en la figura 8.10 y abrir dicho archivo con un navegador.

Los comandos de la figura 8.10 se describen a continuación:

- `<html>` y `</html>` representan la “etiqueta de inicio html” y la “etiqueta de fin html” respectivamente. Estas etiquetas indican que empieza y acaba el código HTML. Son la primera y la última línea de un código HTML. Dentro de una página HTML hay dos zonas: la cabecera (head) y el cuerpo (body).
- `<head>` y `</head>` delimitan el bloque de la cabecera de la página web.
- `<body>` y `</body>` delimitan el bloque del cuerpo de la página web propiamente dicha.
- `<title>` y `</title>` delimitan el bloque del título de la página web. Este título está dentro del bloque de la cabecera y será el título de la pestaña.
- `<p>` y `</p>` delimitan un párrafo.
- `<sup>` y `</sup>` delimitan un superíndice.

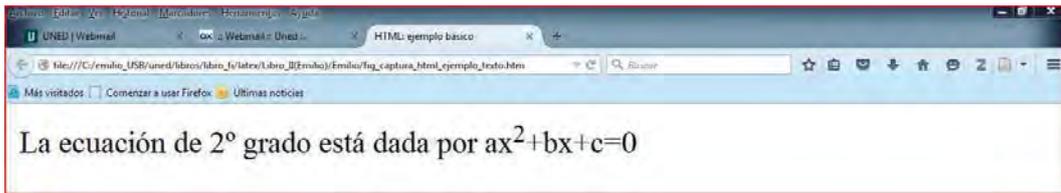


Figura 8.9: Ejemplo de web con texto normal y texto con superíndices

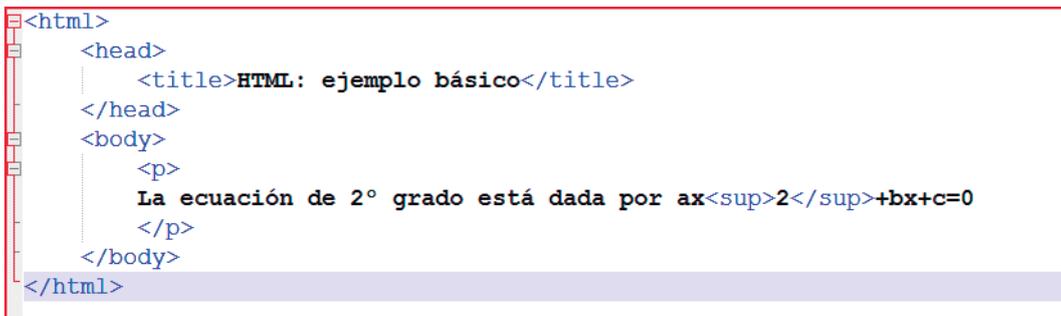


Figura 8.10: Ejemplo de código HTML para web con texto normal y con superíndices

### 8.6.2. JavaScript

JavaScript es un lenguaje de programación interpretado orientado a objetos que se utiliza para dotar de interactividad a una página HTML. Se escribe embebido, incrustado, en una página HTML y permite realizar programas que potencian el lenguaje HTML. Hay varias formas de escribir código JavaScript en una página HTML.

- En línea: utilizando pequeños fragmentos de código asociados a una etiqueta.
- En la cabecera dentro de la etiqueta head: para ello hay que incluir las etiquetas `<script type="text/javascript">` y `</script>` y dentro de estas el código JavaScript.
- En un fichero externo: con lo que se permite su reutilización desde otras páginas HTML.

**Ejemplo 7** JavaScript para resolver una ecuación de segundo grado. Para fijar ideas se supondrá que la ecuación de segundo grado es la dada por  $a = 1$ ,  $b = -1$  y  $c = -2$ , es decir  $x^2 - x - 2 = 0$ . La página web tiene el aspecto de la figura 8.11. Para ello habrá que escribir en un archivo, con la ayuda de un editor de texto, el código HTML con JavaScript dado en la figura 8.12 y abrir dicho archivo con un navegador que soporte JavaScript.

Los comandos de la figura 8.12 se describen a continuación:

- `<script type="text/javascript">` y `</script>` indican que empieza y acaba el código JavaScript. Este código conviene ponerlo en la zona de la cabecera.
- `var a=1; var b=-1; var c=-2;` crean las variables  $a$ ,  $b$  y  $c$  que son los coeficientes de la ecuación de segundo grado y almacenan en ellas los coeficientes deseados por el usuario que en este caso son  $a = 1$ ,  $b = -1$  y  $c = -2$ , pues se quiere resolver  $x^2 - x - 2 = 0$ . Si se quisiera resolver otra ecuación de segundo grado hay que cambiar los coeficientes almacenados en las variables  $a$ ,  $b$  y  $c$ .
- `d=Math.pow(b,2)-(4*a*c)` indica que  $d = b^2 - 4ac$ .
- `x1=(-b+Math.sqrt(d))/(2*a)` indica que  $x_1 = \frac{-b+\sqrt{b^2-4ac}}{2a}$ .
- `x2=(-b-Math.sqrt(d))/(2*a)` indica que  $x_2 = \frac{-b-\sqrt{b^2-4ac}}{2a}$ .
- `if (d>=0) { }` evalúa la condición entre paréntesis y si es cierta se ejecuta lo de dentro de las llaves.
- `document.write('Solución:&nbsp;'+x1+'&nbsp;+&nbsp;'+x2)` indica que se escribe en la propia página HTML las soluciones  $x_1$  y  $x_2$ . Para ello se escribe la cadena de caracteres dada por Solución: y un espacio en blanco, indicado por `&nbsp;`, luego la solución primera de la ecuación, a continuación la cadena de caracteres dada por espacio en blanco, la letra  $y$ , seguida de un espacio en blanco, y por último el valor de la solución segunda de la ecuación. Conviene señalar que el signo más, no indica suma, sino concatenación de caracteres.

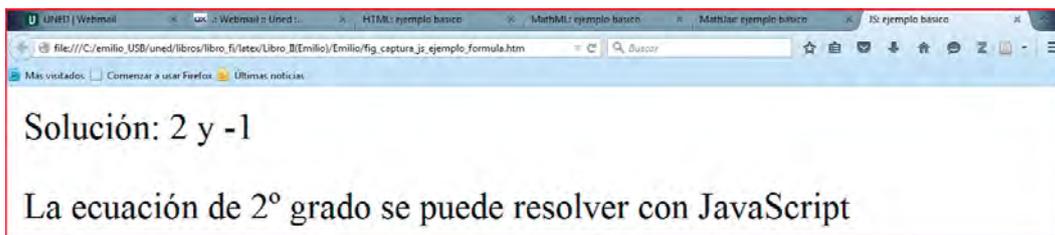


Figura 8.11: Ejemplo de web para resolver una ecuación de 2º grado

### 8.6.3. CSS

La tecnología CSS (Cascading Style Sheets) u hojas de estilo en cascada permite dar estilo a un documento HTML, separando su contenido de la presentación que

```

<html>
  <head>
    <title>JS: ejemplo básico</title>
    <script type="text/javascript">
      var a=1; var b=-1; var c=-2;
      var d=Math.pow(b,2)-4*a*c
      if (d>=0) {
        var x1=(-b+Math.sqrt(d))/(2*a)
        var x2=(-b-Math.sqrt(d))/(2*a)
        document.write('Solución:&nbsp;&nbsp;'+x1+'&nbsp;&nbsp;y&nbsp;&nbsp;'+x2)
      }
    </script>
  </head>
  <body>
    <p>La ecuación de 2º grado se puede resolver con JavaScript</p>
  </body>
</html>

```

**Figura 8.12:** Ejemplo de código HTML con JS para web que resuelve ecuación 2º grado

es la forma en qué va a ser visualizado dicho documento. El CSS se define mediante pares propiedad-valor, por lo que para cada propiedad CSS que se desee modificar hay que decir el nuevo valor asignado. Hay varias formas de definir el CSS.

- En línea: se añade un atributo de estilo a una etiqueta, de forma que dicha propiedad de estilo sólo es válida para esa etiqueta, con lo que la separación entre contenido y presentación es muy pequeña.
- En la cabecera de la página HTML: con lo que las propiedades de estilo son válidas en toda la página pero no fuera de ella; ahora la separación entre contenido y presentación es mayor. Para ello en la cabecera dentro de la etiqueta head, hay que incluir las etiquetas `<style type="text/css">` y `</style>` y dentro de estas el código CSS.
- En un archivo aparte: en este caso, la separación entre contenido y presentación es máxima.

**Ejemplo 8** Estilos básicos con CSS. En este ejemplo se trata de generar en HTML con CSS una página web que tenga el aspecto de la figura 8.13. Para ello habrá que escribir en un archivo, con la ayuda de un editor de texto, el código HTML con CSS dado en la figura 8.14 y abrir dicho archivo con un navegador que soporte CSS.

Los nuevos comandos de la figura 8.14 se describen a continuación,

- `<style type="text/css">` y `</style>` indica que empieza y acaba el código CSS. Este código conviene ponerlo en la zona de la cabecera.