

# 1

---

## CONCEPTOS PREVIOS

### 1.1 HERRAMIENTAS Y SOFTWARE A UTILIZAR EN EL CURSO

---

A continuación, se expone un resumen más o menos detallado sobre las herramientas y servicios que resultan beneficiosas para el aprendizaje y creación de sitios web. En muchos casos, se utilizarán o podrán utilizar a lo largo de este libro.

#### VISUAL STUDIO CODE

Visual Studio Code es un IDE (Entorno de Desarrollo Integrado) desarrollado por Microsoft que permite editar los archivos en modo texto. Esto es necesario para modificar los códigos fuente de las páginas que vamos a crear. Además, presenta multitud de extensiones entre las que se incluyen soporte para la depuración, control integrado de Git y el resaltado de sintaxis.

Es descargable desde:

<https://code.visualstudio.com/download>

#### FUENTES VECTORIALES

Una fuente vectorial es una biblioteca de elementos que representa letras, símbolos y caracteres creados a partir de un conjunto de vectores que responden a unos valores numéricos. La principal ventaja que tiene el formato vectorial es que, los elementos, pueden ser aumentados o reducidos sin pérdida alguna de calidad.

Entre los repositorios de fuentes vectoriales más conocidos podemos encontrar **GOOGLE FONTS** y **ADOBE FONTS**. Ambas pueden ser una buena opción, sin embargo, nosotros utilizaremos las posibilidades de Google Fonts, primero porque es gratuita y, segundo, porque está más extendida a nivel mundial.

### URL de consulta y ayuda:

<https://fonts.google.com/>

Un detalle importante sobre Google Fonts es que contiene gran cantidad de fuentes vectoriales que nos ayudarán a crear diseños de gran calidad y con formatos legibles compatibles para todos los dispositivos, sean móviles o no.

Otro tipo de fuentes vectoriales que se suelen utilizar en casi todos los diseños son las fuentes vectoriales icónicas. A diferencia de las anteriores, las fuentes vectoriales icónicas, presentan imágenes a modo de icono que pueden ser manipuladas como si de texto se tratase a través de CSS.

Las fuentes vectoriales icónicas más conocidas quizás sean FontAwesome y Material Icons. Ambas nos proporcionan la posibilidad de aportar un modo gráfico moderno sin repercutir mucho en el rendimiento y tamaño de las páginas.

### URLS de consulta y descarga:

<https://material.io/resources/icons/?style=baseline>  
<https://fontawesome.com/>

## IMÁGENES E ICONOS

Aunque las imágenes con fines decorativos son un recurso que, afortunadamente, está cada vez más en desuso, siempre serán necesarias en otras circunstancias. Buen ejemplo de ello podría ser la implementación de un banner publicitario o la presentación de un gráfico específico.

Sin embargo, el proceso de creación de imágenes puede llevar mucho tiempo y esfuerzo, por lo que disponer de bancos de imágenes e iconos puede ser la mejor forma de aumentar la productividad y un buen recurso a la hora de diseñar nuestras páginas web. En este sentido, dos de los bancos más conocidos y utilizados en diseño web quizás sean PIXABAY y UNSPLASH.

En estos bancos de imágenes podemos encontrar infinidad de imágenes sin derechos de autor que pueden ser adheridas a nuestras páginas.

### URLS de consulta y descarga:

<https://pixabay.com/es/>  
<https://unsplash.com/>

En lo referente a iconos, los repositorios más conocidos son ICONFINDER y ICON8. En estos repositorios podemos encontrar iconos gratuitos para uso comercial fácilmente implementables en nuestros diseños. Además, pueden ser descargados tanto en formato PNG, como en formato SVG.

## URLS de consulta y descarga:

```
https://www.iconfinder.com/  
https://icons8.com/
```

## COMPRESORES Y MINIFICADORES

Como veremos, algo imprescindible a la hora de diseñar de sitios web es la necesidad de servir los contenidos comprimidos y optimizados, tanto a nivel de imágenes, como a nivel de archivos.

Si lo que deseamos es optimizar la compresión de una imagen, lo habitual es recurrir a algún software específico como Photoshop, Lightroom o Gimp. No obstante, si no disponemos de ningún editor de imágenes, se puede recurrir a alguna herramienta online como es TINYPNG o IMAGEOPTIM. Ambas utilidades son una forma rápida de realizar compresiones con pérdida en archivos PNG y JPEG y que pueden disminuir el tamaño de las mismas de forma drástica.

### URL de acceso y uso:

```
https://tinypng.com/  
https://imageoptim.com/online
```

Ahora, si lo que deseamos es comprimir o reducir el tamaño de nuestros archivos de texto, la opción es utilizar un minificador. Este tipo de utilidades son muy interesantes y permiten reducir el tamaño de los archivos mediante la eliminación de caracteres innecesarios y, como consecuencia, mejoran el SEO.

Entre los más conocidos podemos encontrar MINIFIER.ORG, un minificador de CSS y JS construido en PHP y fácil de integrar. Entre otras cosas, elimina los espacios en blanco, saltos de línea y signos de puntuación innecesarios, borra los comentarios, combina múltiples archivos y optimiza algunos patrones de programación comunes.

### URL de ayuda y uso:

```
https://www.minifier.org/
```

No obstante, según fuentes oficiales de Google, los más recomendados son HTMLMINIFIER para minificar HTML, CSSNANO o CSSO para minimizar CSS y, UGLIFYJS o CLOSURE COMPILER para minificar JavaScript.

### URLS de ayuda y uso:

```
# Minificadores HTML  
https://github.com/kangax/html-minifier  
  
# Minificadores CSS  
https://github.com/cssnano/cssnano  
https://github.com/css/csso
```

**# Minificadores JS**<https://github.com/mishoo/UglifyJS><https://developers.google.com/closure/compiler?hl=es>

## COMUNIDADES DE CÓDIGO

Otro recurso que nos puede venir bien a la hora de crear nuevos desarrollos es la utilización de fragmentos de código ya probados. Esto, y mucho más, es lo que ofrece la web de CODEPEN, una comunidad en línea que permite probar fragmentos de código en JS, CSS y HTML, incluyendo varios motores de plantillas, lenguajes compilados de marcado de hojas de estilo y transcompiladores de JavaScript.

Además, funciona como un editor online de código abierto donde los desarrolladores exponen sus trabajos a modo de presentación y pueden ser compartidos a través de varios métodos.

No obstante, aunque son menos “conocidas” o están menos extendidas para este contexto, existen otras comunidades de códigos en línea como son JSFIDDLE, CODESANDBOX o JSBIN.

**URL de acceso y uso:**<https://codepen.io/><https://jsfiddle.net/><https://codesandbox.io/><https://jsbin.com/>

## MANIPULACIÓN DE COLORES

Para la manipulación de colores lo que podemos usar es COLORZILLA, una extensión disponible para Chrome y Firefox y descargable e instalable desde sus respectivos portales. Incluye un selector de color con conversión a varios formatos, cuentagotas y un generador de degradados, entre otras herramientas.

**URLS para descarga de extensiones:**<https://chrome.google.com/webstore><https://addons.mozilla.org/es/firefox/>

Cabe mencionar que, aunque COLORZILLA incluye un generador de degradados, si no lo tenemos instalado y necesitamos crear un degradado de manera rápida, se puede recurrir a la utilidad online de ULTIMATE CSS GRADIENT GENERATOR, también suministrada y mantenida por Colorzilla.

**URL de ayuda y uso:**<https://www.colorzilla.com/gradient-editor/>

---

## OTROS RECURSOS DE DISEÑO

Aunque en este curso no se utilizarán, siempre es bueno tener una referencia a ciertos recursos por si hiciesen falta.

Uno de estos recursos son las plantillas porque, además de ser una buena herramienta para obtener una primera aproximación de manera rápida, pueden ser una buena fuente de inspiración para nuestros diseños.

En este sentido, los repositorios de plantillas más extendidos quizás sean HTML5UP, TEMPLATED, STYLESHOUT, THEMEWAGON y FREEHTML5, los cuales ofrecen plantillas muy funcionales de código abierto, con y sin derechos de autor.

### URLS de consulta y descarga:

```
https://html5up.net/  
https://templated.co/  
https://www.styleshout.com/free-templates/  
https://freehtml5.co/  
https://themewagon.com/theme_tag/free/
```

Otro de estos recursos son los frameworks de CSS, que ofrecen un set de utilidades que hacen posible que los diseños se realicen mucho más ágilmente. En este sentido, los frameworks de CSS más extendidos quizás sean BOOTSTRAP, MATERIALIZE, PURECSS, BULMA y FOUNDATION.

### URLS de consulta y descarga:

```
https://getbootstrap.com/  
https://materializecss.com/  
https://purecss.io/  
https://bulma.io/  
https://get.foundation/
```

## 1.2 HISTORIA DE LA WEB

---

En los años 60 se produjo una nueva forma de compartir información con otros usuarios. Se trataba de un servicio de comunicación que sólo permitía la inclusión de textos y eran manipulables a través de navegadores de sólo texto. Sin embargo, no fue hasta principios de los noventa cuando se creó HTML, lo que provocó que la web empezara a tener una aceptación suficiente y extenderse como la pólvora.

En aquel entonces, la web era un sistema unidireccional de publicación estático de sólo texto que no presentaba gráficos o imágenes, no ofrecía opciones de personalización, no permitía la actualización y, mucho menos, la posibilidad

de realizar intercambio de datos, por lo que los usuarios no podían interactuar con el contenido y, únicamente, se limitaban a consultar o leer la información que el administrador de la página web hubiese subido a la red. A esto, se denominó la Web 1.0.



La web 2.0, término que fue bautizado por O'Reilly en el año 2004, supuso la segunda generación de sitios y páginas web. Este tipo de webs ya no eran estáticas, ni de sólo lectura y permitían, entre otras cosas, compartir e interactuar con la información de una manera sencilla. Como consecuencia de ello, se produjo un desarrollo de la inteligencia colectiva que fomentaba la colaboración y el intercambio de información a través de comunidades y redes sociales.

Es, por esta época, cuando se crean y extienden sistemas tan conocidos como son los blogs, chats, wikis o foros. Sistemas bidireccionales, los cuales, permitían manipular y gestionar la información de forma sencilla, además de permitir la adición de comentarios y opiniones o interactuar con otros usuarios que presentaban las mismas inquietudes, pero que no requerían tener el mismo nivel técnico o cultural.

Sin embargo, no fue hasta la web 3.0 dónde se produjo un salto cuantioso en lo referente a los sistemas en red. La web 3.0, la tercera generación de sitios y páginas web, ya no sólo era una forma de interactuar y compartir la información de manera sencilla, ahora, su objetivo darle significado y enriquecer la experiencia del usuario.

Es aquí, como alguno ya habrá pensado, cuando nace la Semántica web y las páginas web empiezan a estructurarse a través de un lenguaje natural que puede ser interpretado por el software definiendo qué parte tiene qué función. De esta forma, acceder a la información resulta más sencillo y rápido de procesar, porque hasta las máquinas son capaces de “entender” los contenidos y su objetivo.

Se dice que la web 3.0 también tiene bastante que ver con la inteligencia artificial puesto que las páginas y aplicaciones web ya poseen la capacidad de conectarse entre sí para ofrecer un mejor servicio a los intereses de cada usuario. No obstante, es en la web 4.0 dónde esta premisa está más presente puesto que es quién obtiene un comportamiento más inteligente, predictivo y simple que implica menos movimientos y más acciones con mejores resultados.

Con la web 4.0, nacen el aprendizaje profundo (Deep Learning) y el aprendizaje automático (Machine Learning), tecnologías que forman parte de una familia de métodos de aprendizaje automático basados en redes neuronales con aprendizaje de representación. En otras palabras, tecnologías basadas en sistemas capaces de aprender a realizar tareas tras analizar diferentes patrones y muestras mediante técnicas aprendizaje que permiten descubrir de manera automática las características de una entidad a partir de datos sin procesar.

El ejemplo más conocido o extendido de todo esto quizás sea Watson de IBM, un software capaz de responder preguntas realizadas en lenguaje natural y de realizar tareas como Speech To Text, el cual permite hablarle a una máquina y convertir lo dicho en texto escrito.

Pero esto no es todo, la web 4.0 es la responsable de que los usuarios sean advertidos por sus dispositivos móviles antes de que ellos mismos se den cuenta. Por ejemplo, ¿quién no ha recibido notificaciones con la ruta más corta al trabajo, avisos por atascos en la carretera, alertas por fuertes tormentas o lluvias o mensajes de advertencia sobre tu elevado ritmo cardiaco?

### **1.3 CÓMO FUNCIONA LA WEB**

---

De forma básica, cuando un usuario se conecta a Internet con un dispositivo cualquiera, se le asigna un identificador único mediante los protocolos TCP/IP (Protocolo de Control de Transmisión / Protocolo de Internet). El protocolo TCP proporciona el medio para crear las conexiones y el protocolo IP proporciona el mejor “camino” para alcanzar su destino.

Este identificador único, más conocido como dirección IP, suele estar compuesto por cuatro códigos de 8 bits y vinculado a un nombre, también único, el cual utilizamos para acceder a un sitio web (véase, por ejemplo, <https://google.es>).

¿Qué es lo que sucede entre medias? Como hemos dicho, Internet se mueve a través de direcciones IP, por lo que, para conseguir la dirección IP asignada a ese nombre que hemos introducido, primero se debe acceder a un sistema intermedio que almacena dicha relación.

Ese sistema intermedio se conoce como DNS (Sistema de Nombres de Dominio) y, fundamentalmente, lo que hace es recopilar un catálogo de correspondencias de nombres e IPs y devolver un valor concreto como, por ejemplo, 216.58.211.35.

Una vez que se tiene el objetivo al que dirigirse, el navegador, también llamado Cliente en términos de comunicaciones, abre una instancia de comunicación con el Servidor mediante el protocolo HTTP (Protocolo de Transferencia de Hipertexto). Este protocolo es quién dicta las normas para que el Cliente se comunique con el Servidor Web asignado a la IP anteriormente adquirida y es, además, quién define la sintaxis y semántica que se debe utilizar en cada conexión.

No obstante, si accedemos a la consola del navegador (pulsando F12) y recuperamos la información de la pestaña NETWORK, al recargar la página veremos que la mayoría de estas conexiones entre el Cliente y el Servidor se realizan a través de HTTPS, o lo que es lo mismo, la versión segura del protocolo HTTP.

En este tipo de comunicación, el servidor establece un cifrado basado en la seguridad de textos mediante los protocolos criptográficos SSL/TLS, los cuales, permiten crear una capa codificada intermedia entre los protocolos HTTP y TCP/IP por el que envía el código HTML que el navegador muestra al usuario.

A continuación, se muestra un gráfico que representa todo el proceso:

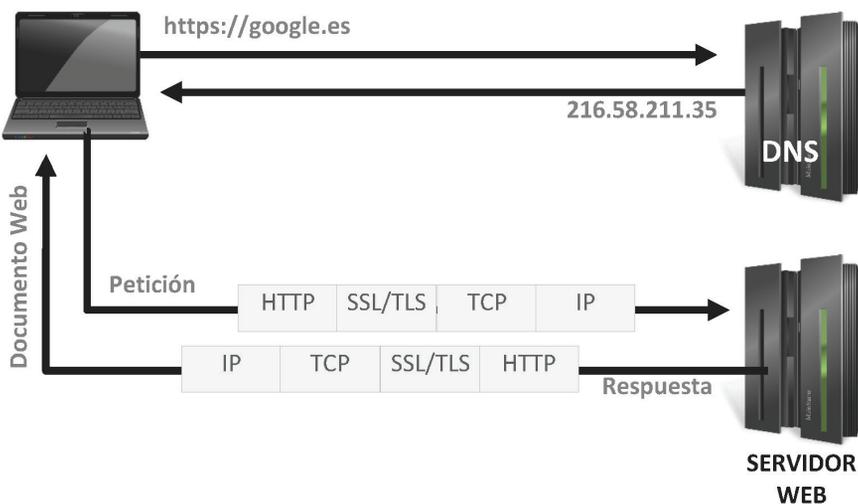


Figura 1.1. Proceso de una petición web.

---

## **1.4 DISEÑO GRÁFICO Y EL PAPEL DEL DISEÑADOR**

---

El diseño gráfico podría definirse como una disciplina que consiste en presentar información visual y cuyo objetivo es que, los usuarios, capten mensajes específicos sobre un tema o materia determinada.

También podríamos definir el diseño gráfico como el arte de crear una composición de imágenes y texto de forma que se transmitan mensajes de forma efectiva.

Cuando hablamos de diseñadores gráficos, no estamos hablando de comerciales ni de artistas, no obstante, es un perfil que tiene algo de ambos. El objetivo de un diseñador gráfico no es vender nada, aunque pueda hacerlo. Ni tampoco es ser un experto comunicador, aunque puede y debe tener una cierta maestranza para poder abstraer los mensajes que se quieren divulgar y transmitirlos de forma eficiente.

Por tanto, lo primero que necesita un diseñador gráfico para empezar a trabajar es un documento o reunión informativa en la que se le proporcione todos los datos necesarios para poder afrontar de manera ordenada, estratégica y creativa el objetivo a cumplir.

Más tarde, cuando ya disponga de los requerimientos y necesidades, y tenga una idea más o menos clara de lo que se debe crear, el diseñador podrá ayudarse, y en general lo hará, de herramientas específicas como son PhotoShop, Illustrator, Gimp o FireWorks, y de lenguajes de marcado como HTML5, CSS3 o SVG, los cuales facilitan el trabajo en gran medida.

---

## **1.5 ARQUITECTURA DE LA INFORMACIÓN Y EL PAPEL DEL ARQUITECTO**

---

La Arquitectura de la Información (AI) es el arte, disciplina o metodología, según quien lo defina, encargada de estudiar, analizar, organizar, y estructurar la información.

Cuando se habla de Arquitectura de la Información, a lo que se hace referencia es a unos principios de diseño y construcción que ayudan a seleccionar y presentar la información en sistemas interactivos y pasivos.

En lo referente a lo que busca la Arquitectura de la Información podríamos decir que, su objetivo, es:

- Procesar y dosificar la ingente cantidad de información que se produce cada día en las redes.
- Desarrollar y verificar los procesos de producción de forma que los usuarios puedan acceder con los mínimos problemas posibles.

- Utilizar la información de manera clara y precisa.
- Organizar, estructurar, marcar y distribuir los sistemas de información para que la experiencia de los usuarios se vuelva más sencilla y menos frustrante.

Si pensamos en la Arquitectura de la Información como proceso, lo que encontramos es una especie de ecosistema que tiene un ciclo de vida completo y que va definiendo los objetivos para que la producción y desarrollo se realice de forma efectiva y eficiente.

De hecho, para que los contenidos puedan ser accedidos, utilizados y asimilados de manera eficiente y efectiva, la Arquitectura de la Información define una serie de técnicas que, dependiendo de cuáles sean y de cuándo se produzcan, ayudan al desarrollo y creación de sistemas de información.

Estas técnicas pueden definir:

- El objeto o propósito del producto.
- El público objetivo.
- Los estudios de audiencia y análisis de competencia.
- La planificación, gestión y desarrollo de contenidos.
- El diseño de la interacción, la Usabilidad y la Accesibilidad.
- El diseño de la navegación, organización y apariencia de los contenidos.
- El marcado de los datos para acceder a los contenidos y facilitar su búsqueda.
- Los procesos de reingeniería del sistema.

Por tanto, un arquitecto de la Información es una persona que lleva a cabo la ejecución y verificación del diseño del sistema o interfaz, además de estar en constante cooperación con los diseñadores gráficos y responsables de la parte de procesamiento y lógica de negocio para definirla.

Sin embargo, no me sentiría cómodo al cerrar esta parte, sin antes, realizar unos últimos comentarios. Cuando se habla de Experiencia de Usuario (UX) se puede estar haciendo referencia a la Arquitectura de la Información, porque es una parte o componente de misma. No obstante, mientras que la Arquitectura de la Información se centra en la información, la Experiencia de Usuario se centra en los usuarios y en cómo pueden actuar y pensar.

Piénsese que el concepto de UX es un conglomerado de técnicas y elementos que define o ayuda a definir la estructura y organización de la información, el diseño de interacción, la usabilidad y accesibilidad de los sistemas y que, para ello, se apoya en el diseño gráfico y procesos cognitivos y de percepción.

---

## 1.6 ELEMENTOS DE BÁSICOS DE UNA PÁGINA WEB

---

Antes de entrar en materia, veamos que es una página web. Desde un punto de vista conceptual, una página web se podría definir como un documento que contiene información electrónica que se presenta de forma ordenada, estructurada y/o jerarquizada.

Desde un punto de vista más formal, una página web podría definirse como un conjunto de contenidos textuales y no textuales (como puedan ser sonidos, imágenes, vídeos, scripts, etcétera) que se presentan de manera secuencial, ordenada y estructurada, que son accesibles mediante una red informática externa o interna y que pueden ser presentados por un agente de usuario (como pueda ser un navegador, lector de pantalla, magnificador u otra herramienta de asistencia).

Dicho de un modo más sencillo. Si se observa una página web desde un navegador, lo que se percibe es un conjunto de entidades a las que se les dota de un significado determinado. Este conjunto de entidades tiene, además, una serie de características concretas como puedan ser la forma, el color o el tamaño que hacen que los elementos se perciban aisladamente o en conjunción con otros.

Si la entidad es una unidad léxica o palabra, su objetivo suele conllevar una interpretación cognitiva y un proceso de memorización, pero si la entidad es otra cosa como, por ejemplo, una imagen o un video, el objetivo suele ser el impacto sensorial, la comprensión y/o la satisfacción.

No obstante, una página o documento web no sólo es un conjunto de elementos relacionados entre sí, también puede contener personalizaciones, animaciones y transiciones, o llevar asociadas acciones y reacciones como el envío y/o consulta de información o el intercambio de datos.

Esto es posible gracias a que, una página web, no sólo se compone de un etiquetado en HTML, que es el nivel más bajo de marcado, sino que, por lo general, tiene partes definidas en otras tecnologías como puedan ser CSS, SVG o JavaScript.

La combinación de HTML con estas otras tecnologías es la máxima responsable de que las páginas web tengan la interacción que se produce entre el usuario y la máquina y, posiblemente, también es la máxima responsable de que el uso de Internet tenga éxito que tiene.

### 1.6.1 Configuración y estructura básica de una página web

Todo documento web presenta siempre una misma estructura básica. Una cabecera que prepara el documento, un contenido central o principal que habitualmente se denomina cuerpo del documento y un pie de página que suele proporcionar algunos contenidos adicionales.

La cabecera de la página es la parte en donde se proporcionan todos los datos de configuración del documento y todos los datos que describen el contenido del documento que se va a presentar. Estos datos que describen el contenido del documento se denominan formalmente metadatos y, en pocas palabras, se definen como datos acerca de los datos.

Estos metadatos pueden ser de muy diferente índole, desde datos puramente informativos, como quién creó el documento o el título del mismo, hasta datos que configuran la presentación del contenido. Estos últimos, pueden preparar la codificación de caracteres a usar en el documento, establecer las palabras clave que permiten la indexación en los motores de búsqueda como Google o Yahoo!, permitir o no que la página sea rastreada por los robots de los motores de búsqueda, definir el periodo de validez de los datos en caché, etcétera, etcétera, etcétera.

El cuerpo de la página, por el contrario, no describe nada referente al propio documento, sino que proporciona todo el contenido perceptible, es decir, es la parte del documento donde se establecen los textos, imágenes, vídeos, gráficos estadísticos e, incluso, los sonidos o música de fondo.



**Figura 1.2.** Ejemplo de estructura básica de una página web.

Como se puede observar en la ilustración anterior, cada elemento de una página web suele estar vinculado a una sección determinada. Por ejemplo, el logo y menú principal suelen estar ubicados en la zona de la cabecera y, en el pie de página, se suele mostrar información sobre el copyright.

No obstante, las secciones, casi nunca suelen tener tan pocos elementos o ser tan triviales. De hecho, los pies de página, habitualmente, suelen llevar otros contenidos adicionales como puedan ser los enlaces de interés, los créditos, acceso a las redes sociales y, en algunos casos proporcionan un formulario para apuntarse a la Newsletter o para realizar un contacto.

---

## **1.7 LOS CONTENIDOS**

---

El contenido web podría definirse como un conjunto de datos expresados en lenguaje visual y que tienen en un formato específico. Así, un contenido web podría ser un contenido textual, que hace referencia a un contenido de sólo texto, o un contenido multimedia, que hace referencia a un contenido visual y/o auditivo.

A menudo, todos los contenidos de una web están ubicados en el cuerpo de la página, sin embargo, no todo lo que conforma el cuerpo de una página es contenido web. De hecho, un enlace no se considera contenido web debido, principalmente, a que representa un vínculo hacia otro contenido.

Evidentemente, esto tiene una excepción, porque, si un enlace forma parte de un listado temático si puede representar un contenido web, debido a que se considera un directorio o catálogo que da una visión global sobre una materia determinada.

Dicho esto, ahora viene la pregunta. ¿Qué tipos de contenidos puede haber?

### **1.7.1 Tipos de contenido**

#### **1.7.1.1 TEXTUAL**

El contenido textual es aquel en el que requiere de la lectura y escribe sobre una materia concreta, independientemente de la forma que se utilice para contarla. No obstante, dependiendo del tono, objetivo, formato o forma, los contenidos pueden ser interpretados de manera diferente.

Así, por ejemplo, si el tono es más riguroso y está escrito en segunda o tercera persona, podríamos pensar en blogs o artículos de investigación, pero si el tono es más visceral y está escrito en primera persona, podríamos estar hablando de artículos de opinión.

Si pensamos en el objetivo, la cosa también puede cambiar mucho. Por ejemplo, si el contenido explica un tema o materia de forma detallada y específica a través de un archivo PDF, o a través de una presentación interactiva que el usuario puede ir probando, podríamos estar hablando de una guía o manual de ayuda.

Ahora bien, si el contenido es principalmente visual y está diseñado o construido a través de alguna herramienta como Prezi o InVision, podríamos estar hablando de una presentación para una ponencia o videoconferencia.

#### **1.7.1.2 AUDIOVISUAL**

El contenido audiovisual es aquel que viene representado por una sucesión de imágenes y/o secuencias de audio susceptibles de ser emitidos y/o transmitidos.

Este tipo de contenido es uno de los preferidos por los usuarios porque es el más fácil de entender, además de ser el más impactante. Además, no suele estar en la misma infraestructura que la página o documento, primero, porque requieren de unas características específicas diferentes al contenido textual y segundo, pueden provocar “cuellos de botella” en la transmisión y ocupar gran espacio en el servidor.

En general, este tipo de contenido viene codificado en formato MP4, OGG o AVI, aunque el más aceptado y extendido es el MP4. La opción de utilizar uno u otro, en general, suele determinarse en función de los navegadores a los que se les quiere dar soporte, como ya veremos.

### 1.7.1.3 GRÁFICO

Una imagen es una representación visual de una figura u objeto real o imaginario. Las imágenes son otro de los contenidos más apreciados por los usuarios porque provocan sensación de moderno y atractivo, sin embargo, hay que utilizarlas con precaución porque pueden provocar problemas de rendimiento, memoria, accesibilidad y usabilidad.

Uno de los contenidos visuales más apreciados, además de las imágenes, son las infografías, las cuales permiten explicar, a través de imágenes, procesos complejos.

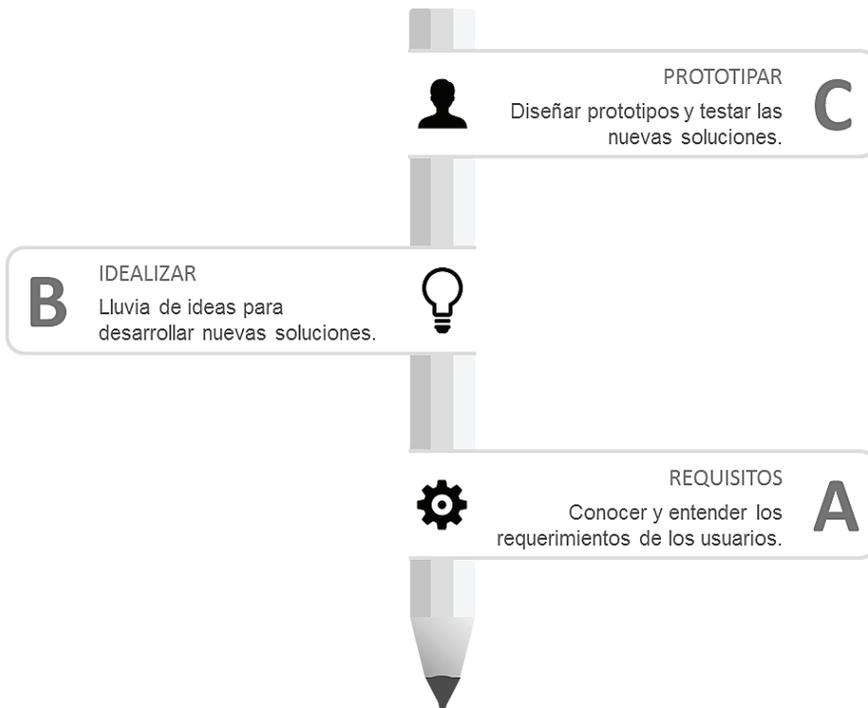


Figura 1.3. Ejemplo de infografía

En esta categoría, podríamos meter las newsletter, que son un tipo de contenido que se suele enviar por correo electrónico y suele estar formado por una estructura tipo tabla que intenta provocar la atención de los usuarios a través de imágenes y con expresiones o frases cortas. El principal objetivo de estos contenidos es dar acceso a los contenidos o productos que ofrecen las empresas.

También, en esta categoría, podríamos introducir los memes, que son una mezcla de imágenes con expresiones o frases cortas, en las que figura un primer plano de un personaje real o de ficción, y una frase que podría decir el personaje, pero que no tiene por qué ser verídica, cierta o real.

#### **1.7.1.4 AUDITIVO**

El contenido auditivo podríamos decir que viene representado por una sucesión de sonidos bajo un cierto orden legible.

En esta categoría, podríamos meter los efectos sonoros o la música de fondo, lo cuales suelen estar codificados en formato MP3, aunque no es el único posible.

También, en esta categoría, podríamos meter a los podcats, que son un formato de audio que representa una emisión de radio o televisión que ha sido digitalizada para que los usuarios puedan descargarla y escucharla cuando deseen. Este tipo de contenido suele estar codificado en formato MP3 o MP4.

Al igual que sucede con el contenido audiovisual, la opción de utilizar uno u otro, en general, suele determinarse en función de los navegadores a los que se les quiere dar soporte, sin embargo, el más recurrente es el MP3.

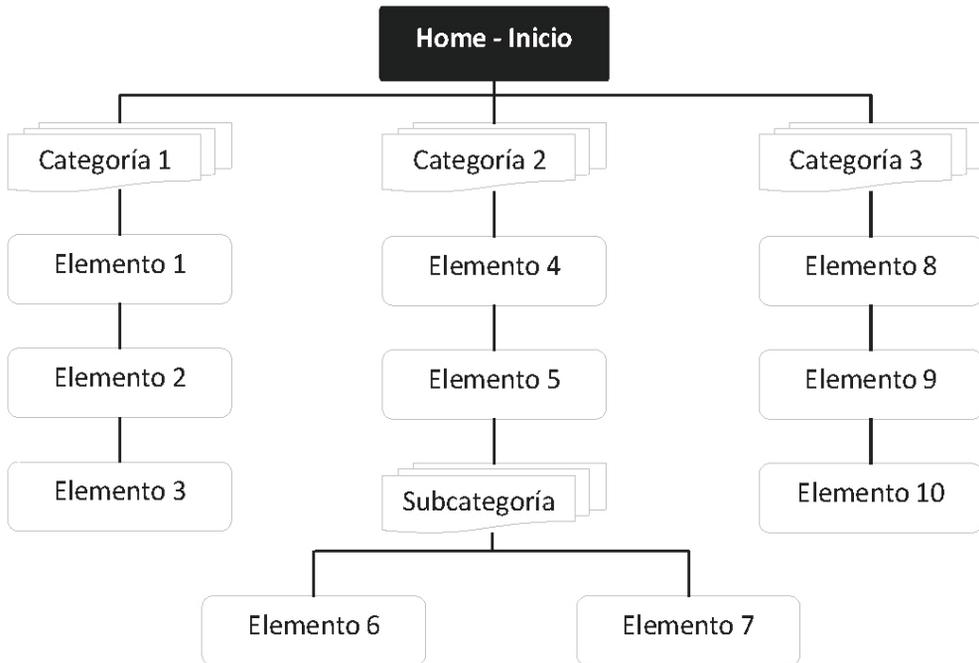
#### **1.7.1.5 E-BOOK**

Este tipo de contenido suele considerarse especial porque hace referencia a grandes cantidades de información sobre temas específicos descargables y utilizables a través de dispositivos especiales, comúnmente denominados libros electrónicos.

Aunque, estos dispositivos, a menudo, suelen admitir la codificación en PDF, entre otros, las codificaciones más extendidas suelen ser en formato ePUB o Mobi.

### **1.7.2 El árbol de contenidos**

Un árbol de contenidos se puede definir como un esquema que representa, de forma visual, la composición o configuración integral de un documento, página o sistema de información. En otras palabras, un árbol de contenidos es una estructura jerarquizada que determina cómo se estructura la navegación dentro de un sistema o interfaz.



**Figura 1.4.** Ejemplo de árbol de contenidos sencillo.

Como se puede apreciar en la ilustración, a través de la representación visual se puede establecer una jerarquía para cada nodo o elemento que permite pronosticar cómo deberán aparecer los contenidos en el sistema o interfaz y cuantos pasos tendrán que hacerse para llegar al contenido deseado.

No obstante, un árbol de contenidos no sólo es útil para distribuir el contenido y definir las rutas de navegación, también sirve para ver si la estructura de contenidos es o no compleja y si, los usuarios, van a ser capaces o no de llegar a encontrar la información deseada de forma sencilla y natural.

### 1.7.2.1 OPTIMIZACIÓN DE UN ÁRBOL DE CONTENIDOS

Si se analizasen unos cuantos sitios web, se podría comprobar que, en general, la mayoría de ellos suelen responder al nivel de complejidad propuesto en la figura 1.4, es decir, el nivel más profundo de contenidos está a dos o tres niveles por debajo de la página de inicio. Además, el número de categorías iniciales suele establecerse entre cinco y siete elementos, como máximo.

Esto no quiere decir que no se puedan integrar sistemas web con una mayor complejidad de niveles o categorización. Aunque la recomendación es esa, un árbol

de contenidos puede requerir más elementos y más categorías. Cuando esto pasa, una de las formas para conseguir simplificarlo es tener claro cuáles son los objetivos del sitio web y, basándose en la competencia, crearlo de tal manera que una persona sin conocimientos adquiridos previos sea capaz de manejarlo.

Otra opción, si la anterior no es posible, es priorizar la información según unos criterios beneficiosos para el negocio bajo una distribución lógica e intuitiva. No obstante, también deberá estar diseñado para conseguir que los usuarios puedan percibirlo y comprenderlo de forma rápida y adecuada, con palabras claras y representativas.

Por ejemplo, imaginemos un árbol de contenidos de un comercio electrónico cualquiera en donde las zapatillas de andar por casa han sido clasificadas dentro de la sección de hogar. Aunque, en un momento dado, podría tener sentido que ese artículo pudiera ser clasificado así, lo más probable es que, cuando los usuarios vayan a buscarlo, lo hagan dentro de la sección de calzado y, de no estar ahí, lo más seguro es que se frustren y se vayan a otro comercio.

### **1.7.2.2 NIVELES Y CAPAS DE NAVEGACIÓN**

Los niveles y capas de navegación deben estar diseñados para proporcionar accesibilidad a los contenidos de los sitios web y aumentar el número de conversiones.

El concepto de conversión es una métrica que suele indicar el porcentaje de usuarios que realiza una acción buscada. Uno de los ejemplos más claros de esto, bien podría ser la reserva de hotel. Si una web que ofrece reservas de hotel tiene 100 visitas al día, pero sólo reservan 18, su número de conversiones (tasa de conversión) será del 18%.

Para intentar conseguir esto se puede recurrir a varias fórmulas o métodos, los cuales se comentan a continuación.

#### **1.7.2.2.1 Estructura lineal**

La estructura lineal es aquella que presenta un único nivel de navegación, a partir del cual, los demás contenidos pueden ser accedidos haciendo un único clic en una dirección determinada. En otras palabras, es como cuando estamos haciendo un curso en Internet. Cuando terminamos de leer una sección o capítulo, el sistema siempre nos ofrece la posibilidad de acceder al anterior o al siguiente contenido.

Este tipo de estructura es especialmente útil cuando lo que se desea es que el usuario haga un itinerario o camino específico sin que pueda saltar a otros contenidos, es decir, es útil cuando lo que se desea es diseñar una guía de ayuda o tutorial.

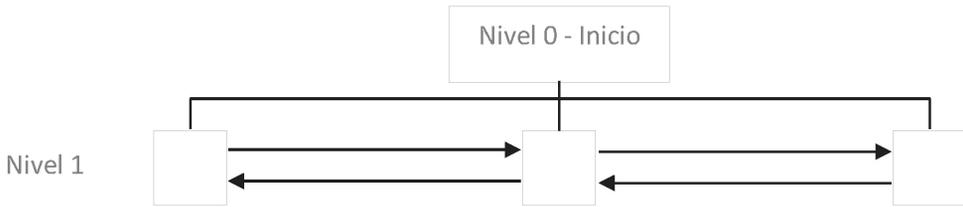


Figura 1.5. Estructuración lineal.

Cabe destacar que, no es recomendable que el número de contenidos encadenados sea muy elevado porque puede provocar sensación de fatiga y frustración y no permite regresar con facilidad al punto dónde se abandonó la última sesión. Además, suele producir poca relevancia en los contenidos porque los motores de búsqueda como Google, dan mayor importancia a la página de inicio.

#### 1.7.2.2 Estructura jerárquica

La estructura jerárquica define un nivel de navegación que representa los grandes grupos o categorías del negocio y, a partir de ahí, la navegación se realiza en cascada. Es decir, un nodo puede representar un contenido final o una categoría que lleva a una subcategoría que, a su vez, puede llevar a un contenido o subcategoría, y así sucesivamente.



Figura 1.6. Estructuración jerárquica.

Este tipo de estructura es especialmente útil cuando lo que se desea es organizar contenidos que, desde un punto de vista estricto, no tienen nada que ver unos con otros. Es decir, la sección de servicios no tiene nada que ver el con blog o con la sección contactar. Por ello, es la más utilizada por las empresas y organizaciones, porque facilita la organización de grandes cantidades de información y ayuda a estructurar de forma lógica, es decir, de lo general a lo específico.

Cabe destacar que, esta estructura es una buena opción, siempre y cuando, no se tengan más de 3 o 4 niveles ya que, si tienen más niveles los usuarios pueden frustrarse y abandonar la página por tardar en encontrar lo que buscan o por no comprender bien la navegación.

Sin embargo, la relevancia de los contenidos se genera según su nivel de jerarquía porque los motores de búsqueda como Google, indexan los contenidos en función del nivel de profundidad con respecto a la página de inicio.

### 1.7.2.2.3 Estructura en red

La estructura en red define un nivel de navegación que permite acceder a los contenidos de forma libre y flexible, es decir, no existen niveles como tal y desde cualquier contenido se puede ir a otros contenidos que llevan a unos contenidos diferentes.

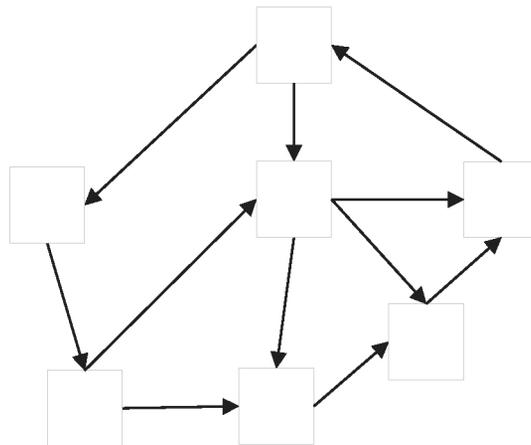


Figura 1.7. Estructuración en red.

Este tipo de estructura puede llegar a ser útil cuando el contenido no es muy grande, fundamentalmente, porque la relación existente entre sus elementos puede generar confusión y desorientación a los usuarios. Esto es porque, al no haber ninguna jerarquía establecida, los usuarios pueden no saber de dónde han venido y puede que, ni siquiera, a dónde van, por lo que se frustran y abandonan el sitio.

No obstante, como las páginas se relacionan entre sí mediante enlaces internos, se favorece la indexación de las páginas del sitio y eso puede ayudar a que la indexación de los motores de búsqueda como Google posicionen mejor los contenidos.

## 1.8 LA INTERACCIÓN

---

Cuando hablamos de interacción, en general nos referimos a una relación, influencia o acción - reacción entre dos seres humanos, animales o cosas, pero, en el contexto online o web, la interacción se refiere únicamente a una acción - reacción que se produce entre una persona y un sistema o aplicación web.

Dicha interacción, puede producirse de varias maneras y, habitualmente, puede clasificarse en tres grandes grupos o categorías.

### 1.8.1 Mecanismos lingüísticos

Los mecanismos lingüísticos son aquellos en los que el intercambio de información se produce a través de dispositivos de tipo palanca o interruptor, como pueda ser un teclado.

Dónde más se suele ver este tipo de interacción es cuando se trabaja con procesos que requieren de un dispositivo como el teclado y/o el ratón. Por ejemplo, es habitual utilizar este tipo de interacción cuando se está desarrollando un sistema o interfaz, cuando se manipula un control de versiones para llevar un histórico de los cambios o cuando se pulsa una combinación de teclas en una página web determinada que hace que se ejecute una acción asociada.

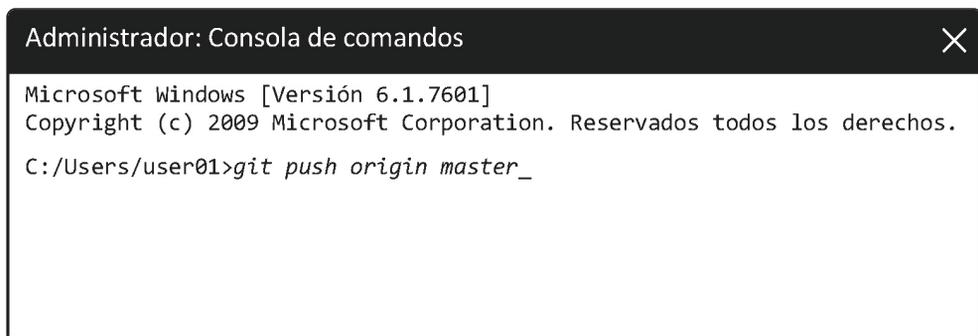


Figura 1.8. Símbolo de sistema de Windows

### 1.8.2 Mecanismos contextuales

Los mecanismos contextuales son aquellos en los que el intercambio de información se produce alrededor de un hecho, evento o situación. Este hecho, evento o situación, puede producirse por una única razón, no obstante, lo normal es que se produzca por conjunto de circunstancias que les rodean o condicionan.

### 1.8.2.1 DIÁLOGOS EMERGENTES

Uno de los mecanismos más recurrentes hoy día son los diálogos emergentes o ventanas modales, los cuales, pueden tener varias funcionalidades. Así, por ejemplo, no es lo mismo un diálogo emergente que solicita la intervención del usuario que un diálogo emergente que informa de una situación problemática, sea o no error.

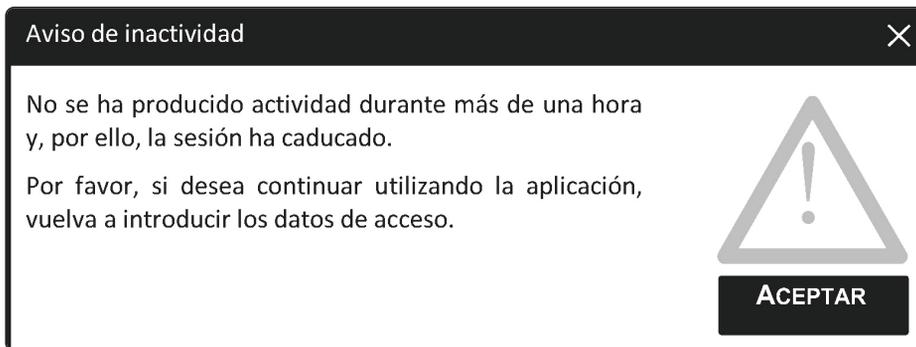


Figura 1.9. Ejemplo de diálogo o popup

En este contexto, también hay que matizar dos situaciones que pueden darse, cuando el diálogo modal interrumpe la ejecución del sistema o interfaz y cuando no la interrumpe. En el contexto web, lo normal es que todos los diálogos emergentes obliguen al usuario a realizar una acción, la cual, si no se ejecuta, no permita continuar, pero habrá casos en que no.

Por ejemplo, cuando una página web nos realiza una pregunta a través de un cuadro de diálogo emergente o se nos muestra una alerta típica de JavaScript, la ejecución de todo lo que esté debajo de dicha acción quedará a la espera de confirmación o cancelación.

Los diálogos emergentes pueden ser una buena opción cuando se trata de ofrecer distintos grados de experiencia, debido a que son autoexplicativas y suelen ser fáciles de comprender, pero el uso abusivo de las mismas puede desorientar o abrumar a los usuarios. Además, dependiendo de cómo se gestionen, pueden sobrecargar las páginas y empeorar el rendimiento global del sistema.

### 1.8.2.2 FORMULARIOS

Otro de los mecanismos que más se suelen ver en sistemas y aplicaciones web son los formularios. Este tipo de interacción permite la introducción de datos de manera estructurada.

Los formularios son un mecanismo de interacción que solicita la intervención del usuario para recuperar algún tipo de información. Entre los diferentes elementos

que puede componer un formulario tenemos cajas de texto, casillas de verificación, botones de radio, desplegados y botones de acción.

NOMBRE <b>Pedro Fernández Rodríguez</b>	FUNCIÓN	Profesión o fun-
NÚMERO DE TELÉFONO 999-999-999	EMAIL Introduzca su email de contacto	
EXPERIENCIA Y HABILIDADES Principales experiencias, habilidades y conocimientos		
ESTÁ ACTUALMENTE TRABAJANDO Sí <input type="checkbox"/>	DISPONIBILIDAD EN DÍAS <b>15</b>	<b>ENVIAR SOLICITUD</b>

**Figura 1.10.** Ejemplo de formulario

Los formularios puede que sean uno de los recursos más utilizados puesto que se pueden utilizar en multitud de situaciones, desde la realización de encuestas, hasta la creación y edición de productos y servicios, sin embargo, antes de crear un formulario, es importante tener en cuenta las restricciones que se puedan dar para no incumplir el RGPD (Reglamento General de Protección de Datos).

### 1.8.2.3 MENÚS

Otro mecanismo de interacción contextual que no suele faltar, son los menús de navegación. Este tipo de interacción permite presentar una lista de opciones seleccionables que el usuario puede ejecutar.

Dependiendo de cuál sea su función, los menús podrán ser de tipo comando, atributo o estado.

- Los de **tipo comando** son aquellos es los que se realiza una acción específica, como pueda ser copiar, cortar o pegar.
- Los de **tipo atributo** son aquellos en los que se manipula una característica sobre uno o varios elementos, como pueda ser cambiar el tipo de letra o el color del texto.
- Los de **tipo estado** son aquellos en los que se habilita o deshabilita una opción de configuración, como pueda ser habilitar un documento para sólo lectura o desactivar un botón de “enviar” hasta que no se hayan rellenado todos los campos requeridos.



**Figura 1.11.** Ejemplo Menú de navegación

Cabe destacar que, un desplegable de un formulario puede ser considerado un menú, puesto que responde a la definición de lista con opciones seleccionable, no obstante, en sistemas de información web, lo habitual es diferenciarlos en función de si llevan o no una acción asociada.



**Figura 1.12.** Ejemplo Menú desplegable

### 1.8.3 Mecanismos por manipulación directa

Los mecanismos por manipulación directa son aquellos en los que el intercambio de información se produce cuando se tienen que realizar o ejecutar tareas complejas, pero que no requieren una inversión de tiempo elevada en lo que aprendizaje se refiere.

La manipulación directa se basa en el reconocimiento visual y a la simplicidad por lo que, a menudo, los usuarios recuerdan mejor los conceptos principales sin tener que recurrir a la memorización de opciones y/o conceptos.

Además, la manipulación directa tiene una cosa bastante buena y es que, en general, es reversible. Es decir, si el usuario comete un error, siempre dispone de algún método para poder subsanarlo, lo que aumenta la seguridad y disminuye la ansiedad y frustración.



Figura 1.13. Ejemplo manipulación directa

Si nos fijamos en la ilustración anterior, uno de los tipos de manipulación directa más extendidos y frecuentemente utilizados es el icono. Sólo debemos fijar la mirada en la parte derecha de cualquiera de los recuadros y ver que hay uno o dos triángulos.

Cada una de estas figuras representa un icono, entendiendo que, un icono, es una entidad visual que representa un concepto. Por ejemplo, en el recuadro del margen de arriba, los triángulos nos indican que puede aumentarse o disminuirse su valor.

### 1.8.4 Mecanismos de interacción social

Los mecanismos de interacción social son aquellos en los que el intercambio de información se produce entre dos o más personas bajo un mismo contexto o sistema.

La interacción social se basa en el intercambio de mensajes y opiniones sobre un tema o materia en particular y, habitualmente se utiliza para conseguir una mayor cercanía entre los usuarios y las empresas y organizaciones, aunque, también se suele utilizar para crear audiencia, notoriedad e influencia sobre productos y servicios.

Por su definición, este tipo de interacción se da de manera más notoria en las redes sociales a través de comunicaciones directas que incluyen el identificador de usuario, pero también suelen darse muchas comunicaciones indirectas que incluyen un identificador de hashtag.

## 1.9 LAS BASES DE DATOS

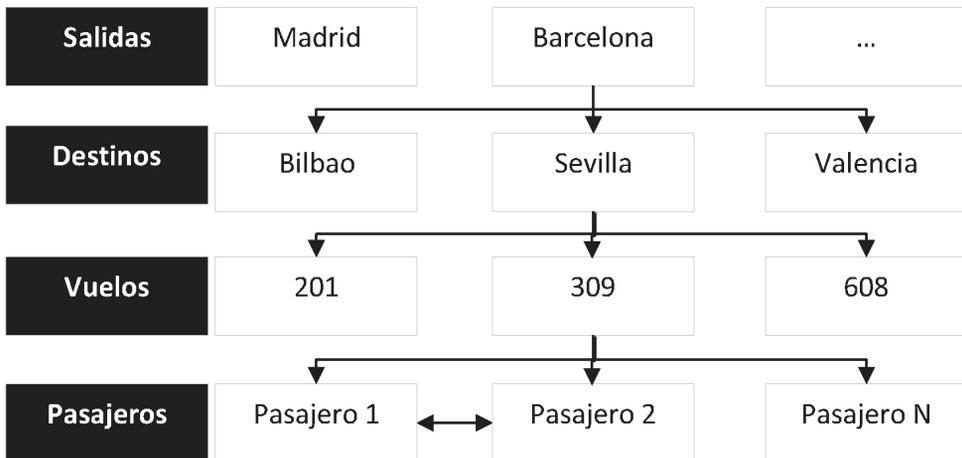
Una base de datos es una herramienta que permite la manipulación, organización y administración de información relacionada y estructurada de forma sencilla y sistemática a través de una interfaz gráfica o de línea de comandos.

A los programas o aplicaciones que gestionan estas herramientas se las suele denominar Sistemas Gestores de Bases de Datos (SGBD) y son consideradas una de las partes fundamentales dentro de un sistema o aplicación web.

Las bases de datos pueden ser de múltiples tipos, aunque las más extendidas son las que responden al modelo relacional y las que responden al modelo documental.

### 1.9.1 Bases de datos jerárquicas

Una base de datos jerárquica es un modelo de gestión de bases de datos en dónde la información se estructura y almacena en forma de árbol.



**Figura 1.14.** Ejemplo de base de datos jerárquica

Se denominan jerárquicas porque poseen un nodo raíz que no participa como registro hijo y porque todos los nodos, a excepción del nodo raíz, admiten un único padre.

No obstante, la característica de jerarquía no tiene por qué estar de manera obligatoria, es decir, puede suceder que se establezca una relación entre elementos del mismo nivel, como es el caso entre los elementos hoja Pasajero 1 y Pasajero 2.

Entre las principales limitaciones que presenta este modelo podemos encontrar que:

- No se garantiza la unicidad de datos porque pueden aparecer registros duplicados en el mismo nivel.
- No se garantiza la integridad referencial porque permite realizar dejar elementos inconexos o ejecutar acciones como borrar un elemento que contiene otros elementos.
- No se garantiza la normalización porque, por ejemplo, no existen los campos de identificador único y se permiten la combinación de relaciones o la creación de grupos repetidos.

Como nota adicional, sólo diremos que este modelo puede ser considerado como un caso especial o particular de base de datos en red.

### 1.9.2 Base de datos de red

Una base de datos en red es un modelo de gestión de bases de datos en dónde la información está formada por un conjunto de registros que se conectan entre sí a través de vínculos de red.

Se denominan en red porque cada registro propietario (nodo padre), tiene uno o más registros miembros (nodos hijo). Además, pueden existir conjuntos singulares en los que, el propietario, puede ser el sistema.

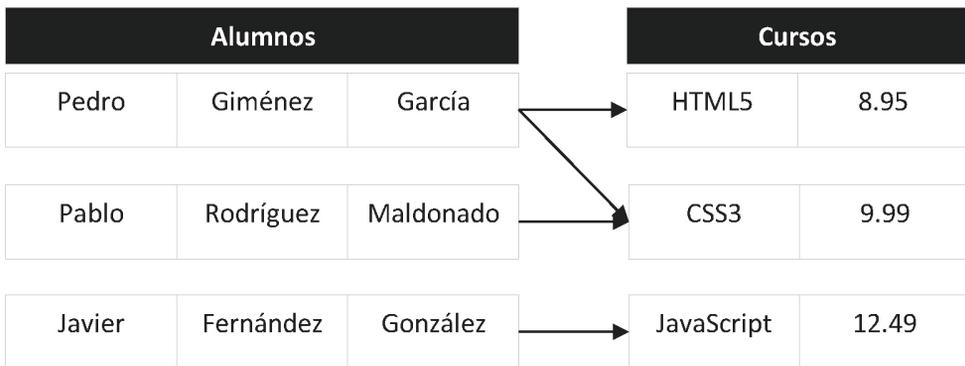


Figura 1.15. Ejemplo de base de datos en red

En lo referente a su estructura, una base de datos en red es bastante más compleja que la jerárquica, entre otras cosas, porque cualquier registro, sea propietario o miembro, puede pertenecer a uno o varios conjuntos y porque permite la declaración de un registro como propietario en un conjunto, pero como miembro en otro.

Cabe destacar que, los registros de este modelo son objetos similares a los utilizados o descritos en el modelo relacional, pero con la diferencia de que únicamente pueden almacenar un dato. No obstante, un registro miembro puede tener más de un registro propietario, a diferencia del modelo jerárquico que sólo admite un nodo hijo por padre.

Entre las principales limitaciones que presenta este modelo podríamos destacar las mismas que en el modelo jerárquico, a excepción de que, la adquisición de una buena seguridad de la información es más compleja.

### 1.9.3 Bases de datos transaccionales

Una base de datos transaccional es un modelo de gestión de bases de datos en dónde la información se recolecta, almacena y manipula a través de transacciones, es decir, mediante eventos que crean o actualizan información a partir de datos que se encuentran almacenados en otros sistemas de información.



Figura 1.16. Ejemplo de base de datos transaccional

Las bases de datos transaccionales están pensadas para el almacenamiento de grandes cantidades de información y son, a menudo, utilizadas por empresas y organizaciones que pertenecen a las áreas de ventas, finanzas, marketing o recursos humanos. La razón es, esencialmente, que requieren una fuerte consistencia entre sistemas de información diferentes.

Pensemos, por ejemplo, en las transferencias bancarias. Cuando se realiza una transferencia bancaria de una cuenta a otra, independientemente de si es o no

el mismo banco, la cantidad a restar de la cuenta emisora debe ser exactamente la misma que la cantidad a sumar en la cuenta receptora, de lo contrario se produciría una incoherencia que podría acabar en problemas mayores.

Entre las principales limitaciones que presenta este modelo podríamos destacar que presenta poca flexibilidad en las consultas, requieren un tiempo de respuesta entre peticiones muy pequeño (cuanto más pequeño mejor) y que requiere una alta fiabilidad de la información, con posibilidad de recuperación y respaldo de manera rápida y fácil.

### 1.9.4 Bases de datos relacionales

Una base de datos relacional es un modelo de gestión de bases de datos en dónde la información presenta relaciones predefinidas y se organiza como un conjunto de filas con una o varias columnas.

Las bases de datos relacionales se denominan así porque cumplen el modelo relacional, el cual se basa en la idea de que cada relación (o tabla) está compuesta por unos registros o filas denominadas tuplas.

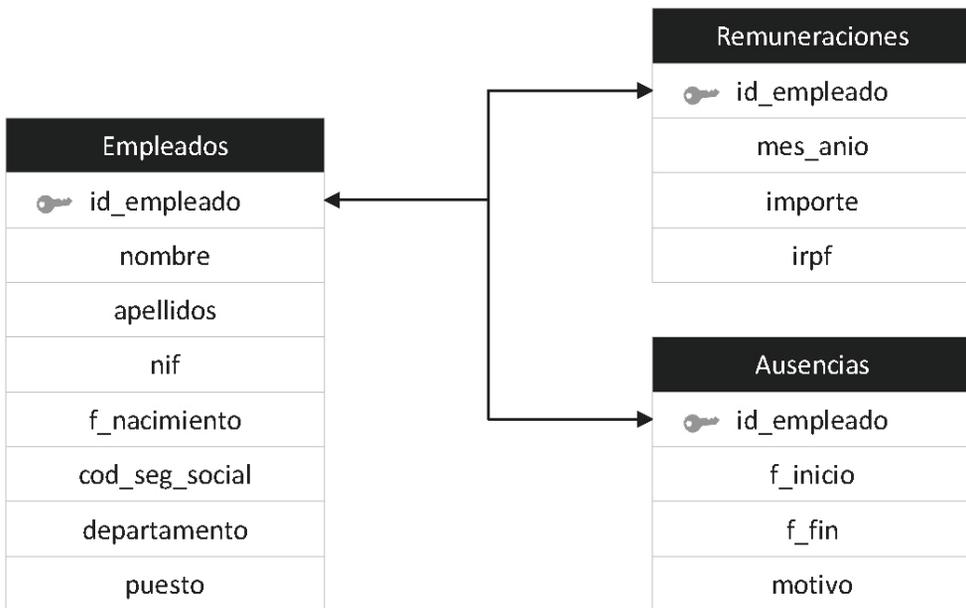


Figura 1.17. Ejemplo de base de datos relacional

Entre sus principales características, podemos destacar que poseen un modelo de datos único, que obliga a que la relación entre tablas se lleve a cabo a

través de unos identificadores únicos bidireccionales (clave primaria y clave ajena) y que garantiza que, cada relación o tabla, se realice de forma única, es decir, no permite la existencia de dos tablas con el mismo nombre.

Si el identificador único se declara como clave primaria, actuará como clave principal de forma que ese valor no pueda repetirse. Además, los registros serán indexados a través de estos valores.

Si el identificador único se declara como clave foránea o ajena, deberá tener el mismo valor que su correspondiente clave primaria y actuará como clave de indexación de forma que ese valor quede vinculado a la clave primaria, independientemente de su relación (1:1 o 1:N).

Aunque las relaciones se establecen a través del par clave primaria – foránea, en general, todas las bases de datos relacionales permiten indexación con valores repetidos a partir de uno o varios campos. No obstante, este tipo de índices no es posible utilizarlos para establecer relaciones.

Además, el modelo de base de datos relacional utiliza el lenguaje de comunicación estándar SQL y garantiza el principio designado como ACID:

- **Atomicidad:** Es la propiedad que asegura si se ha realizado la transacción o no. Es decir, garantiza si todas las partes intermedias que conforman la transacción se realizaron con éxito o no.
- **Consistencia:** Es la propiedad que asegura que los datos involucrados en las transacciones sólo cambien la información afectada cuando se cumplan las reglas y directrices de integridad referencial. De este modo, se podrá garantizar que la información que se devuelve siempre sea la misma.
- **Aislamiento:** Es la propiedad que asegura que cada operación pueda ejecutarse de forma independiente sin generar errores ni afectar a otras operaciones anteriores o posteriores.
- **Durabilidad:** Es la propiedad que asegura que, una vez hechas las operaciones, los datos almacenados persistirán en el tiempo, aunque el sistema falle.

Sin ningún lugar a dudas, este es el modelo de base de datos más utilizado y extendido en Internet para gestionar y administrar datos. Ciertamente es que el modelo de base de datos documental está tomando auge, no obstante, nunca podrá superar ni reemplazar al modelo de base de datos relacional.

Entre los sistemas gestores de bases de datos que se basan en el modelo relacional, los más conocidos son MySQL, Oracle, PostgreSQL y SQLServer.

La utilización de uno u otro SGBD suele depender de la arquitectura y tecnologías que se deseen utilizar.

- **MySQL:** Está pensada para trabajar con, casi, cualquier arquitectura, aunque su mayor éxito se da en arquitecturas PHP y Python bajo sistemas Linux o Windows. Es, quizás, el sistema gestor de base de datos más extendido, primero, porque permite ahorrar costes iniciales, segundo, porque es compatible con todos los servicios de hosting, sean o no compartidos y, tercero, porque es uno de los lenguajes más sencillos de aprender.
- **Oracle:** Está pensada para trabajar con, casi, cualquier arquitectura, aunque su mayor éxito se da en arquitecturas Java y PHP bajo sistemas Linux o Windows.
- **PostgreSQL:** Está pensada para trabajar con arquitecturas que permiten su integración a modo de objeto, es decir, permite que los elementos puedan tratarse de forma similar a como lo hace un lenguaje de programación orientado a objetos.
- **SQLServer:** Está pensada para trabajar con, casi, cualquier arquitectura, aunque su mayor éxito se da en arquitecturas Java y .NET bajo sistemas Linux o Windows.

### 1.9.5 Bases de datos orientadas a objetos

Una base de datos orientada a objetos es un modelo de gestión de bases de datos en dónde la información es representada a través de objetos similares a los que se utilizan en los lenguajes de Programación Orientada a Objetos (POO).

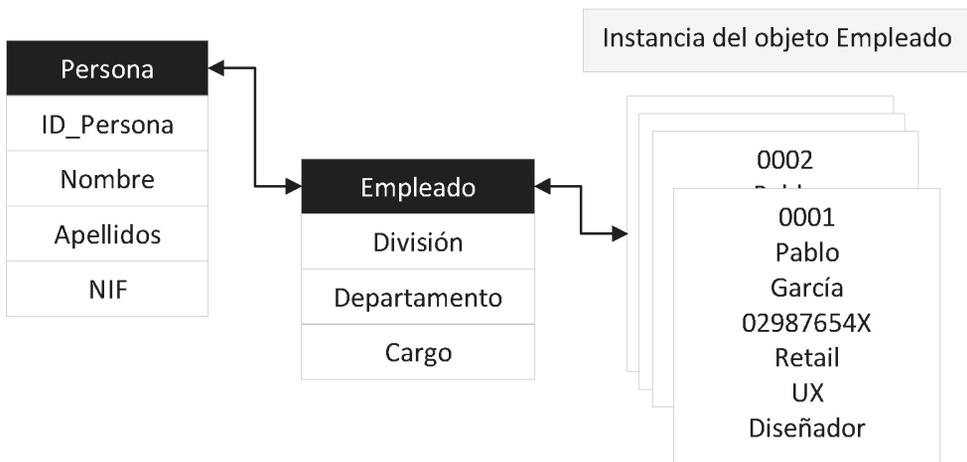


Figura 1.18. Ejemplo de base de datos orientada a objetos

Mientras que el modelo de base de datos relacional requiere de un lenguaje SQL externo para poder acceder a la información, el modelo de base de datos orientado a objetos permite integrarse con los lenguajes de programación de su misma tipología o paradigma, obteniendo así un Sistema Gestor de Base de Datos orientado a Objetos (SGBDO o ODBMS en inglés), el cual permite que los objetos de base de datos puedan ser manipulados de forma transparente y como si de un objeto del propio lenguaje se tratase.

Entre las principales características que posee, podemos destacar que:

- Permiten la encapsulación, es decir, la ocultación de información al resto de entidades u objetos.
- Permiten el polimorfismo, es decir, que una propiedad pueda aplicarse a objetos de distinta tipología, que presentan una fuerte herencia, lo que permite que se hereden comportamientos.
- Permiten su extensión o ampliación de forma sencilla.
- Permiten la declaración de operaciones sobre los datos como parte de la definición de la base de datos.

Entre los sistemas gestores de bases de datos que se basan en el modelo orientado a objetos, los más conocidos son Object Database++, ObjectStore, GemStone/S, Wakanda y ObjectDB.

La utilización de uno u otro SGBD suele depender del objetivo y tecnologías utilizadas.

- **Object Database++:** Está más pensada para el alojamiento de aplicaciones en servidores remotos y suele utilizarse cuando los sistemas requieren muchas búsquedas con un fuerte control sobre las mismas.
- **ObjectStore:** Está más pensada para sistemas que requieren almacenamiento en caché y suele utilizarse cuando se requiere una migración y sincronización de datos entre sistemas diferentes.
- **GemStone/S:** Está más pensada para sistemas que requieren alta disponibilidad y suele utilizarse cuando se requiere una integración de aplicaciones de diferente índole de forma escalable y segura.
- **Wakanda:** Está más pensada para el desarrollo web y móvil con aplicaciones en JavaScript y suele utilizarse cuando se requiere un control más exhaustivo sobre el código.
- **ObjectDB:** Está más pensada para sistemas que trabajan en Java y suele utilizarse cuando se trabaja con máquinas virtuales y se requiere el uso de JPA o JDO.

## 1.9.6 Bases de datos documentales

Una base de datos documental es un modelo de gestión de bases de datos en donde la información está formada con un conjunto de programas o aplicaciones que almacenan, recuperan y manipulan datos, de algún modo, estructurados.

Las bases de datos documentales son una de las principales tipologías que pertenecen al grupo de bases de datos NoSQL y se denominan así porque están diseñadas alrededor de lo que se define como noción abstracta de “Documento”, aunque también es porque esa información estructurada de algún modo, a menudo, se refiere a datos de documentos.

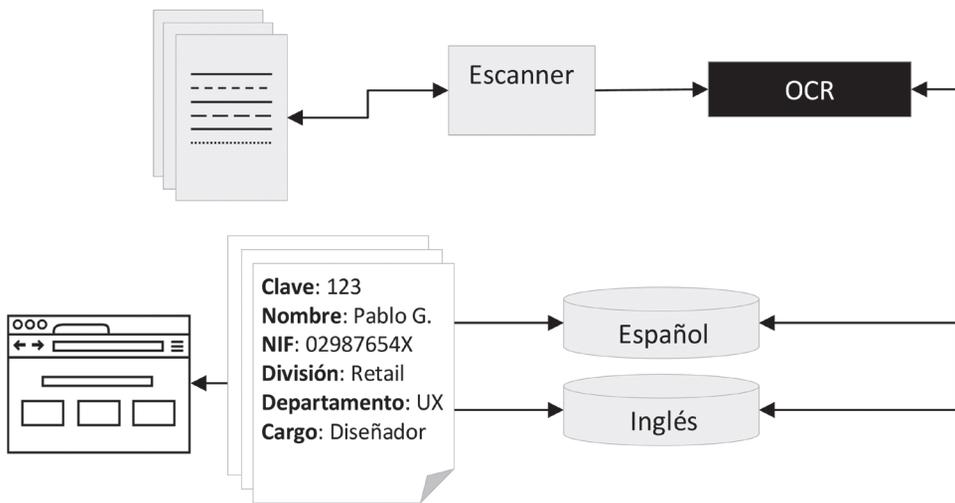


Figura 1.19. Ejemplo de base de datos documental

Como se puede observar en la ilustración anterior, los registros de datos dentro del documento son similares a las tuplas o registros utilizados en el modelo relacional, no obstante, son menos restrictivos porque no requieren de un formato estándar, ni que tengan los mismos atributos o campos.

Por ejemplo, mientras que un documento puede tener una lista sobre los hobbies y aficiones de un empleado, otro puede no disponer de esa información (ni siquiera los atributos) y contener una lista de sus experiencias laborales.

Entre los sistemas gestores de bases de datos que se basan en el modelo documental, los más conocidos son MongoDB, DynamoDB, CouchBase, RavenDB, Cassandra e IndexedDB.

La utilización de uno u otro SGBD suele depender del objetivo y tecnologías utilizadas.

- **MongoDB:** Es un SGBD estructurado bajo el esquema NoSQL que utiliza un formato de almacenamiento basado en BSON (similar al JSON). Posee un alto rendimiento en la realización de consultas y actualizaciones y suele utilizarse cuando se requiere un registro de datos diversos con altos volúmenes de información.
- **DynamoDB:** Es una base de datos NoSQL creada por Amazon que utiliza un modelo de datos basado en clave-valor. Cumple el principio ACID y puede llegar a manejar hasta 20 millones de solicitudes por segundo y suele utilizarse cuando las aplicaciones no disponen de servidor propio.
- **CouchBase:** Es una base de datos NoSQL que utiliza un formato de almacenamiento basado en JSON organizado a modo de pares de clave-valor. Posee una baja latencia con un rendimiento sostenido y suele utilizarse cuando se requiere una buena accesibilidad y compatibilidad entre dispositivos diferentes.
- **RavenDB:** Es una base de datos NoSQL que utiliza un formato de almacenamiento basado en JSON y permite la actualización a través de colas LINQ y métodos API. Está escrita en .NE, por lo que suele utilizarse en estos entornos ya que facilita su manipulación.
- **Cassandra:** Es una base de datos NoSQL distribuida que utiliza el modelo de datos basado en clave-valor. Está escrita en Java y permite la adición y eliminación sin que afecte a la consulta o actualización y suele utilizarse cuando se requieren grandes volúmenes de datos en forma distribuida.
- **IndexedDB:** Es una base de datos NoSQL que utiliza un formato de almacenamiento basado en JSON. Posee está creada en JavaScript y suele utilizarse cuando se requiere un acceso a datos en modo de fuera de línea (offline) con almacenamiento en caché.

Cabe destacar que, aunque las bases de datos documentales pueden ser muy diferentes en lo referente a los detalles, en general, todas suelen cumplir con el principio de encapsulación de documentos y con alguna codificación estándar JSON o BSON, no obstante, también pueden llegar a utilizar los estándares de XML o YAML.

### 1.9.7 Bases de datos multidimensionales

Una base de datos multidimensional es un modelo de gestión de bases de datos que está pensado para data warehouse y aplicaciones de procesamiento analítico en línea (OLAP). Aunque se suelen crear a partir de una base de datos relacional, la forma de gestionar y almacenar la información es muy distinta.

Mientras que las bases de datos relacionales almacenan los datos en objetos tipo tabla con filas y columnas, las bases de datos multidimensionales almacenan la información en un vector multidimensional donde, cada dimensión, representa un conjunto o rango de valores determinado.

<i>ID</i>	<i>Curso</i>	<i>Alumno</i>	<i>Fecha Inicio</i>	<i>Nota</i>
1	HTML5	Pablo García	1575244800	8.5
2	CSS3	Pablo García	1575244800	6.6
3	HTML5	Jorge González	1575244800	7.2
4	CSS3	Elena Fernández	1575244800	4.3
5	CSS3	Elena Fernández	1573171200	8.7
6	HTML5	Jorge González	1573171200	9.2
7	HTML5	Elena Fernández	1573171200	5.8



**Figura 1.20.** Ejemplo de base de datos multidimensional

Las bases de datos multidimensionales se basan en la idea de que los datos se pueden representar en tres dimensiones, como si de un cubo se tratase. Cada cubo representa una matriz que contiene un conjunto de dimensiones (categorías descriptivas) y un conjunto de medidas (valores cuantitativos) que coexisten con las dimensiones de análisis y proporcionan un contexto y descripción a través de atributos. Al conjunto de todas esas dimensiones es lo que se denomina jerarquías.

Dependiendo del número de dimensiones de este vector, el nombre puede diferir. Así, si el número de dimensiones del vector es de tres, se le suele denominar cubo, mientras que, si es igual o superior a cuatro, se le suele denominar hipercubo.

La forma de almacenar los datos es como si fuese una matriz en donde las celdas son las medidas y las posiciones de cada celda son las jerarquías. Por decirlo de algún modo, cada celda de esta matriz representa un qué, y el valor de las dimensiones representa un cómo y cuándo.

La principal razón para utilizar una base de datos multidimensional es porque permiten analizar gran cantidad de datos muy diferentes con una gran agilidad y velocidad por lo que, el tiempo y los recursos empleados durante el análisis se reduce de manera bastante notoria.

Entre los sistemas gestores de bases de datos multidimensionales más populares podemos encontrar Microsoft Analysis Services, Hyperion Essbase y Cognos Power Cube. La utilización de uno u otro SGBD suele depender del gusto y criterio de cada uno.

### **1.9.8 Bases de datos deductivas**

Una base de datos deductiva, también conocida como base de datos lógica, es un modelo de gestión de bases de datos en donde la información es representada a través de hechos, los cuales pueden ser deducidos o inferidos a partir de unas reglas.

Por decirlo así, los hechos son las tuplas o registros de relaciones que representan el conocimiento explícito y, las reglas, son los criterios deductivos que representan el conocimiento implícito. Tanto los hechos, como las reglas, están explícitamente almacenados en la base de datos.

Las bases de datos deductivas se denominan así porque funcionan en base a una lógica matemática preprogramada. Es decir, la base de datos representa un conjunto de fórmulas lógicas que permiten definir otras nuevas.

No obstante, no dejan de ser un sistema de base de datos relacional que está supervisado por un sistema de inferencia. El usuario manipula este sistema dual y recupera la información de la base de datos que contiene los hechos y reglas.

El lenguaje de definición que utiliza este sistema de inferencia suele ser Datalog (un subconjunto de Prolog) y, entre sus cualidades, podemos destacar que permite realizar consultas recursivas, utilizar algoritmos de evaluación eficientes y soporta objetos y conjuntos complejos.

---

## 1.10 PRINCIPALES TIPOLOGÍAS DE WEB

---

A continuación, se muestran las principales topologías web que, actualmente, se encuentran en el mercado.

### 1.10.1 Según su diseño o estructura

#### 1.10.1.1 PÁGINAS WEB ESTÁTICAS O INFORMATIVAS

Las páginas web estáticas, también conocidas como informativas, son aquellas en las que el usuario no puede interactuar con ellas, es decir, lo más que pueden hacer es acceder y leer el contenido propuesto. No utilizan sistemas gestores de bases de datos de ningún tipo, por lo que si hay que actualizar el contenido hay que cambiar los archivos desde el servidor de forma directa.

Este tipo de páginas suelen ser recomendables cuando sólo se desea tener presencia en Internet, rara vez se van a realizar cambios en el contenido y se dispone de pocos contenidos.

#### 1.10.1.2 PÁGINAS WEB DINÁMICAS

Las páginas web dinámicas son aquellas en las que el usuario puede interactuar con ellas de alguna forma, es decir, son webs que disponen de aplicaciones internas que permiten hacer diferentes funcionalidades. En términos generales, suelen requerir algún lenguaje de programación web (entre los más conocidos están Angular, Python, PHP, .Net y JSP) y mucho de su contenido se adquiere a través de sistemas gestores de bases de datos o a través de APIs externas.

Este tipo de páginas suelen ser recomendables cuando:

- Se espera tener una gran cantidad de contenido que puede o debe ser actualizado con cierta frecuencia de forma sencilla y rápida.
- Se desea que los usuarios puedan alterar el diseño o los contenidos de la web.
- Se desean implementar funcionalidades complejas como un buscador de contenidos dentro de la web.

### 1.10.2 Según su funcionalidad u objetivo

Cabe destacar que casi todas las tipologías descritas a continuación son, según su diseño o estructura, dinámicas puesto que, en general, están pensadas para cumplir una, si no varias, de las necesidades descritas anteriormente.

### **1.10.2.1 CMS**

Un CMS Content Management System (sistema de gestión de contenido) es una herramienta web que permite, prácticamente, a cualquier usuario publicar, modificar y administrar todo el contenido de un sitio.

Los CMS se caracterizan porque suelen permitir una fácil instalación y actualización a través de asistentes, el uso de plantillas con diferentes tecnologías de Frontend, la gestión de perfiles de usuario y grupos, la optimización de direccionamiento web para ayudar al SEO y la manipulación de los contenidos a través de algún sistema gestor de base de datos. Estos contenidos pueden ser desde los propios artículos o anuncios, hasta el diseño del sitio o complementos externos.

Dentro de los CMS podemos encontrar varios tipos según su intencionalidad. Así, por ejemplo, los blogs están pensados para que, prácticamente, cualquier persona pueda publicar contenidos más o menos cortos de una forma sencilla y la posibilidad de que puedan comentar esos u otros contenidos.

Entre los diferentes sistemas que existen para crear blogs, los más conocidos posiblemente sean WordPress, Joomla, Drupal o Blogger, todos ellos basados en PHP y MySQL.

### **1.10.2.2 FOROS**

Un foro podría considerarse un CMS, no obstante, la principal diferencia es que, los contenidos, son provistos por los usuarios que visitan el sitio, no por quién lo ha instalado o mantiene. Además, esta herramienta está más pensada para que una gran cantidad de personas participen de forma activa entre los diferentes contenidos con discusiones entrelazadas.

Entre los diferentes sistemas que existen para crear foros, los más conocidos posiblemente sean phpBB, BBPress y MyBB, todos ellos basados en PHP y MySQL.

### **1.10.2.3 WIKIS**

Una Wiki, también conocida como una comunidad virtual, es un término hawaiano que significa rápido y que hace referencia a un tipo de sistema web que permite a los usuarios realizar todo tipo de operaciones de gestión sobre los contenidos con el objetivo de compartirlo y proporcionar un acceso libre. En pocas palabras, es un sistema que permite que los contenidos puedan ser creados y/o editados por múltiples usuarios a través de un navegador.

Entre los diferentes sistemas que existen para crear wikis, los más conocidos posiblemente sean mediaWiki, DokuWiki y TikiWiki, todos ellos basados en PHP y MySQL.

#### 1.10.2.4 PORTALES WEB

Un portal web es un tipo de sistema web que está pensado para crear comunidades. En este tipo de sistemas, se provee el acceso, de forma sencilla e integrada, a diversos recursos y servicios externos en los que se tratan temas relacionados o no de forma que puedan generar o suscitar el interés de los usuarios.

Entre los diferentes sistemas que existen para crear portales web, los más conocidos posiblemente sean Drupal, LifeRay, Microsoft Sharepoint y DotNetNuke, todos ellos basados en diferentes tecnologías como PHP, Java o .NET.

#### 1.10.2.5 COMERCIOS ELECTRÓNICOS

Un comercio electrónico, también conocido como e-commerce o tienda online, es un término que, originalmente, se utilizaba para hacer referencia a sistemas en línea que realizaban transacciones mediante medios electrónicos, no obstante, ahora se identifica más con un tipo de sistema web cuyo objetivo es comprar y vender a través de medios electrónicos.

Entre los diferentes sistemas que existen para crear comercios electrónicos, los más conocidos posiblemente sean PrestaShop, Magento, WooCommerce (que es un plugin de Wordpress) y openCart, todos ellos basados en PHP y MySQL.

#### 1.10.2.6 WEBS CORPORATIVAS

Una web corporativa, también conocida como web institucional, es un sistema web que está creado con el único objetivo de dar a conocer a una empresa. En este tipo de páginas, lo que se suele mostrar son los productos y servicios que ofrece e información sobre su creación e historia facilitando, además, la posibilidad de contactar con la empresa.

Las webs corporativas suelen estar creadas a medida a partir de las tecnologías que estén en auge en ese momento. Por dar algunos ejemplos, las tecnologías más utilizadas para la creación de webs corporativas son:

- PHP, Java, .Net y Python para el backend.
- MySQL, Oracle y SQL Server como posibles bases de datos.
- HTML5, CSS3 y JavaScript para el frontend.

No obstante, también es bastante frecuente que las webs corporativas se realicen bajo algún CMS o sistema de creación de portales web.

#### 1.10.2.7 PORTFOLIOS

Un portfolio es un tipo de sistema web en dónde los usuarios muestran sus habilidades, en especial, las que son susceptibles de ser cobradas o contratadas.

Ejemplos de este tipo de páginas podrían ser un portfolio de fotografías, donde un fotógrafo muestra sus trabajos a modo de publicidad o, un currículum vitae, donde los trabajadores muestran sus conocimientos a modo de presentación.

Al igual que sucede con las webs corporativas, los portfolios suelen estar creadas a medida y a partir de las tecnologías que estén en auge en ese momento, aunque, tampoco es raro verlos a modo de CMS. Esto es:

- PHP, Java, .Net y Python para el backend.
- MySQL, Oracle y SQL Server como posibles bases de datos.
- HTML5, CSS3 y JavaScript para el frontend.

### **1.10.2.8 LANDING PAGES**

Una landing page, también conocida como página de aterrizaje, es una página web que tiene, como única finalidad, que los usuarios realicen algún tipo de acción determinada como pulsar en un botón concreto, suscribirse a una newsletter, hacer clic en un banner o anuncio, etcétera.

A diferencia de los tipos anteriores de web, las landing pages suelen ser creadas de forma estática y con diseño impactante para intentar aumentar la tasa de conversión o el porcentaje de éxito.

### **1.10.2.9 WEBS PERSONALES**

Una web personal es toda página que presenta, como su propio nombre indica, un carácter o enfoque personal. Es el tipo de web al que uno se refiere cuando el objetivo es presentar contenido de naturaleza personal como puedan una web sobre un grupo de música rock, un blog de moda o cocina, un diario o una página sobre los templarios.

Uno podría pensar que un portfolio de fotografía o un currículum vitae son páginas personales y, sí, tienen un carácter personal, no obstante, existe una gran diferencia, el objetivo. Las páginas personales no están pensadas para ofrecer servicios, más bien están pensadas para informar, entretener o dar a conocer.

El uso de las tecnologías en este tipo de páginas es muy variable, desde sistemas más o menos complejos como un CMS, hasta páginas creadas artesanalmente a través de las tecnologías que estén en ese momento en auge.

## **1.11 LENGUAJES DE MARCADO**

---

Los lenguajes de marcado, también se les suele llamar lenguajes de marcas, aunque este tipo de referenciación no es nada habitual.

Históricamente, los lenguajes de marcado eran utilizados por las editoriales y medios de comunicación para imprimir instrucciones concretas en los márgenes. Para ello, utilizaban unos “marcadores” que indicaban el tipo de fuente, estilo, tamaño de letra y la corrección de errores. Una cosa llevó a la otra y, con el tiempo, se estandarizaron como un grupo de marcas que, más tarde, fueron reinterpretadas en lo que hoy conocemos como lenguajes de marcado.

El lenguaje de marcado es un término que hace referencia a una manera de codificar la información. Esta codificación se caracteriza porque los contenidos están envueltos y declarados a través de unas entidades que llamaremos etiquetas o marcas.

En este contexto, cuando se habla de etiquetas o marcas, además se les suele dar un significado que puede indicar qué es o qué debe contener, dónde utilizarlo o cómo presentarlo. No obstante, ninguna de estas marcas tiene la capacidad de ejecutar acciones ni de realizar operaciones de ninguna clase.

Esto último debe tenerse claro porque, en ocasiones, los lenguajes de marcado se confunden con los lenguajes de programación, y nada más lejos. Un lenguaje de programación es un idioma de computación que presenta varias peculiaridades como son la posibilidad de declarar variables, realizar operaciones matemáticas o crear fragmentos de código que suelen ser agrupados en procedimientos o funciones. Un lenguaje de marcado no hace nada de eso, sólo etiqueta la información aportándole un valor semántico.

### 1.11.1 Características de los lenguajes de marcado

Un lenguaje de marcado se caracteriza porque tiene marcas o etiquetas que son declaradas a modo de elemento o entidad. Cada uno de estos elementos o entidades tiene un contenido y uno o varios atributos que definen su comportamiento, su función y su significado.

Un ejemplo de todo esto podría ser la declaración de un producto.

```
<producto>
  <cabecera>
    <titulo>Nombre producto</titulo>
    <subtitulo>Categoría</subtitulo>
  </cabecera>

  <descripcion>
    <texto>Descripción del producto</texto>
  </descripcion>

</pie>
```

```
<negrita>Precio</negrita>  
<accion>Comprar</accion>  
</pie>  
</producto>
```

**Figura 1.21.** Ejemplo de lenguaje de marcado

Como se puede observar en la ilustración anterior, se ha definido una entidad concreta a través de unos elementos que definen su significado y comportamiento y que, en ocasiones, tienen propiedades intrínsecas que aportan un valor a la presentación.

Un ejemplo de ello podría ser la etiqueta `SUBTITULO`, que indica que, su contenido textual, es un texto de cabecera o a modo de título y que está en el segundo nivel de la jerarquía en lo referente a títulos.

Si, además, este código fuese interpretado por una herramienta o sistema específico, podría llevar una interpretación específica que provocase que fuese presentado de forma concreta, con un tamaño, margen y estilo concretos, entre otras cosas.

## 1.11.2 Clasificación de los lenguajes de marcado

Los lenguajes de marcado suelen ser clasificados en tres grupos, no obstante, pueden llegar a mezclarse propiedades de varios en uno, formando nuevas especificaciones.

### 1.11.2.1 MARCADO DE PRESENTACIÓN

El marcado de presentación es aquel que está enfocado hacia la presentación de la información del documento, es decir, es aquel que sirve para dar formato al texto. Este tipo de marcado es suficiente para el proceso de lectura, pero insuficiente para el procesamiento automático de la información. Además, las etiquetas de marcado están ocultas al usuario, lo que hace que su contenido esté ofuscado y sea complicado de extraer si no es con la herramienta adecuada.

El marcado de presentación puede ser sencillo de elaborar, sin embargo, su mantenimiento y/o modificación pueden volverse complicados, por lo que su uso no está demasiado extendido.

Un ejemplo de marcado de presentación es RTF, en donde la composición de las etiquetas suele proporcionar un comportamiento y/o un significado distinto. Es decir, el título del documento podría venir predefinido por una combinación de varios saltos de línea en cualquier posición del documento y terminar con un carácter especial justo detrás del contenido textual.

```
{\rtf1\ansi\ansicpg1252\deff0\deflang3082
{\fonttbl{\f0\fswiss\fcharset0 Arial;}{\f1\fmodern\frpq1\fcharset0 Arial;}}
{\colortbl ;\red255\green0\blue0;\red0\green0\blue128;}
\viewkind4\uc1\pard\b\f0\fs20 RTF\b0 es un \cf1\b\fs24 lenguaje de marcado de
presentaci'f3 \f0\fs20 .\par}
```

**Figura 1.22.** Fragmento de código de documento RTF

### 1.11.2.2 MARCADO DE PROCEDIMIENTOS

El marcado de procedimientos es aquel que está enfocado hacia la presentación del texto, al igual que sucede con el marcado de presentación. No obstante, las marcas o etiquetas que componen el documento no están ocultas, es decir, son visibles en todo momento para el usuario.

El marcado de procedimientos es sencillo de elaborar y de modificar. Además, es interpretado por orden de aparición, por lo que ayuda a predecir cómo será su resultado durante el proceso de lectura.

Un ejemplo de marcado de procedimientos es LaTeX y HTML. Ambos requieren de una marca o etiqueta que funciona a modo de identificador especial que le indica al sistema cómo se debe renderizar el contenido textual. Este renderizado suele venir definido a través de una serie de instrucciones que establecen el texto con un tipo de fuente, tamaño, alineación y estilos concretos.

```
<article>
  <header>
    <h2>Nombre producto</h2>
  </header>

  <div class="descripcion">
    <p>Descripción del producto</p>
  </div>

  <footer>
    <b>Precio</b>
    <button>Comprar</button>
  </footer>
</article>
```

**Figura 1.23.** Fragmento de código de documento HTML5

### 1.11.2.3 MARCADO DESCRIPTIVO

El marcado descriptivo, también conocido como semántico, es aquel que utiliza las etiquetas para describir fragmentos de texto, pero sin especificar en qué orden o cómo deben representarse.

El marcado descriptivo suele ser sencillo de elaborar y modificar y, a menudo, sus etiquetas pueden ir acompañadas de atributos que definen su significado o comportamiento, pero su interpretación está separada de la presentación, es decir, sólo especifica la descripción del tipo y contenido de los documentos.

Al marcado descriptivo también se le atribuye la cualidad de simplificar el proceso de reformateado del texto, principalmente, porque la información del formato está separada del propio contenido. Por esta razón, un fragmento indicado como cursiva a través de la etiqueta `i` (`italic`), podría emplearse para marcar énfasis o para indicar un contenido gráfico.

Cabe destacar que, si quisiéramos sortear esta última situación en el marcado de presentación y en el marcado de procedimientos, el proceso probablemente sería bastante tedioso, sin embargo, si los contenidos se hubiesen diferenciado descriptivamente mediante etiquetas distintas, podrían representarse de manera diferente sin esfuerzo.

Un ejemplo de marcado descriptivo es SGML y el XML. Ambos son muy flexibles ya que los fragmentos se etiquetan como son y no como deberían mostrarse.

```
<!doctype email system "email.dtd">
<email>
  <to>info@ejemplo.com
  <from>alumno@gmail.com
  <date>mon, 29 jan 2020 12:00:02 - 0100 (est)
  <subject>error de acceso
  <contents>
    Hola, buenas tardes. ¿Mi cuenta está bloqueada o deshabilitada?.
    <url>http://ejemplo.com</url>
    saludos
  </contents>
</email>
```

Figura 1.24. Fragmento de código de documento SGML