La Sénia
Institut d'Educació Secundària
Paiporta

UD1. Desarrollo Web en Entorno Servidor

Desarrollo Web en Entorno Servidor

Profesora: Silvia Vilar Pérez

Curso 2024-2025

Contenidos

- Arquitecturas cliente/servidor.
- Aplicaciones Web.
- Lenguajes de programación en entorno servidor.
- Herramientas de programación. IDE
- Tecnologías de Servidor

Modelos de programación en entornos cliente/servidor

- URL (Uniform Resource Locator)

Protocolo://nombrededominio/recurso

Protocolo://maquina[:puerto]/recurso

Ejemplo:

https://www.microsoft.com/es-es/windows/

https://23.214.202.161/windows

Actividad:

Acceder a la carpeta docs de la página de PHP con cada uno de los formatos de arriba

Arquitectura Cliente-Servidor

La arquitectura cliente/servidor está basada en la idea de **servicios:**

- El cliente solicita servicios (**REQUEST**)
- El servidor es un proceso proveedor de dichos servicios (**RESPONSE**)

La comunicación entre ambos se realiza mediante el intercambio de mensajes.

El cliente, a través de un navegador,
inicia el intercambio de información, solicitando datos al servidor.

El servidor responde enviando uno o más flujos de datos al cliente.

Arquitectura Cliente-Servidor

La arquitectura Cliente-Servidor contempla varias capas:

- **Presentación:** Esta capa se comunica únicamente con la capa de negocio
- **Lógica o de Negocio:** Es la capa intermedia, Esta capa se comunica con la capa de presentación para recibir las solicitudes y presentar los resultados, y con la capa de persistencia, para solicitar al gestor de base de datos almacenar o recuperar datos de él.
- **Persistencia:** Es la capa donde se encuentran los datos almacenados para generar información

Capa Presentación

- Es la capa que interactúa con el usuario
- Conocida como interfaz gráfica de usuario (IGU)
- Debe ser «amigable» para el usuario
- Esta capa se comunica únicamente con la capa de negocio

Capa Lógica o Negocio

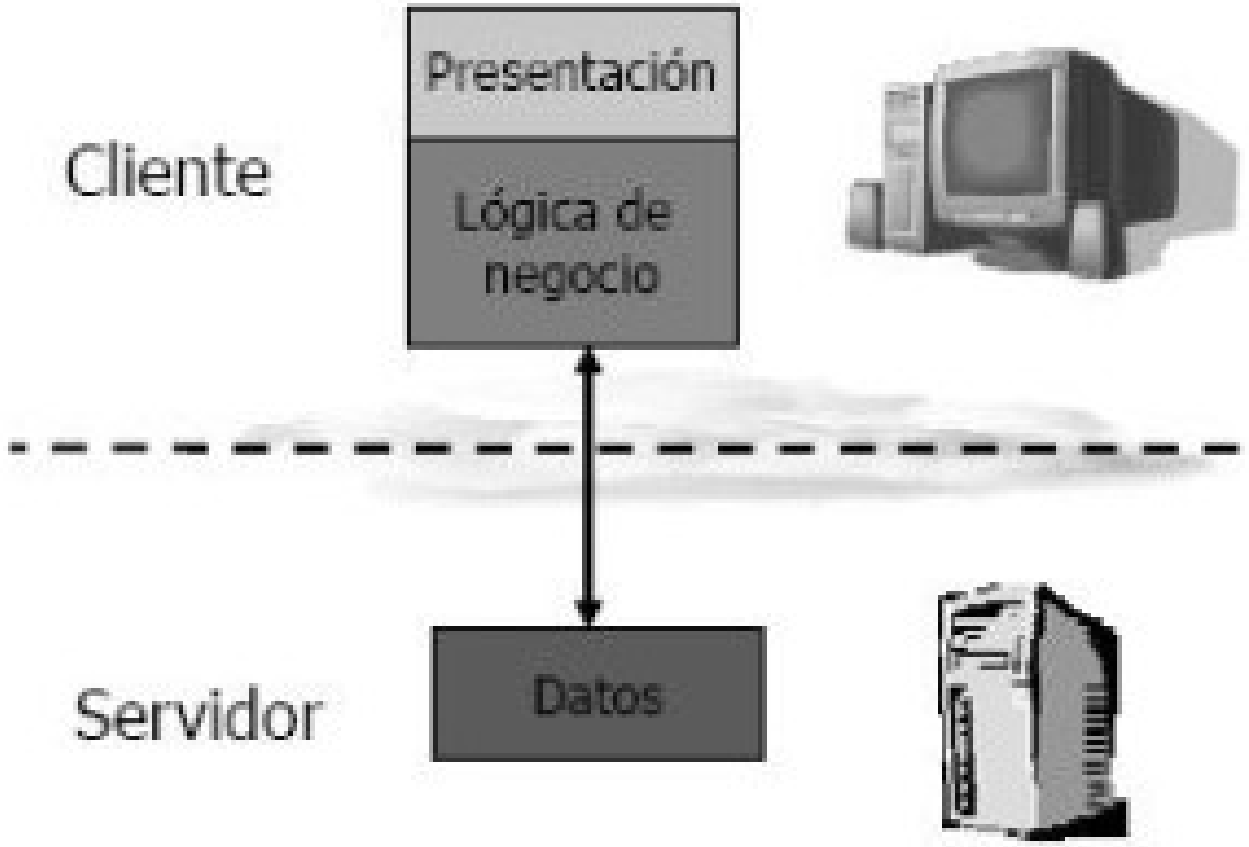
- Es donde residen los programas que se ejecutan.
- Recibe las peticiones del usuario y se envían las respuestas tras el proceso
- Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todas las reglas que deben cumplirse.

Capa Persistencia o Datos

- Donde residen los datos.
- Encargada de acceder a los mismos.
- Formada por uno o más SGBD que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

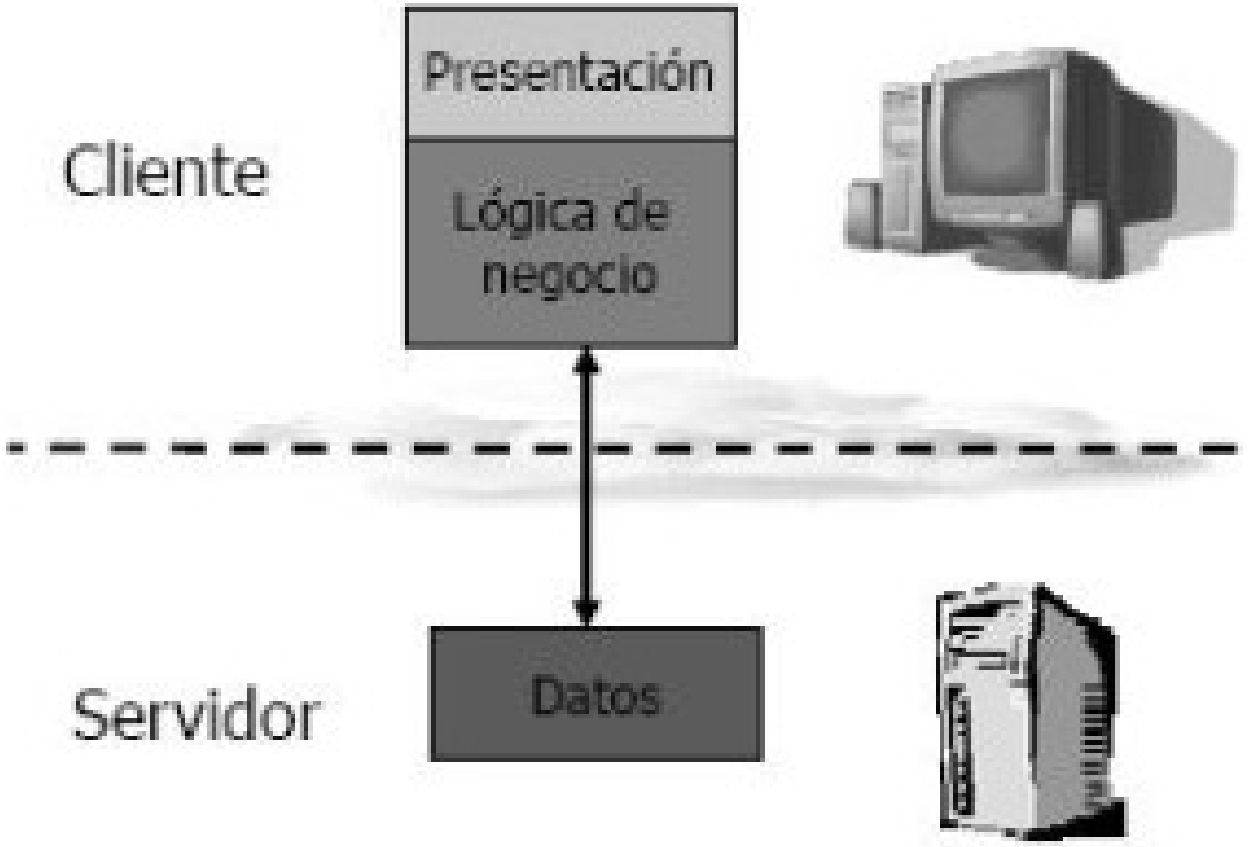
Cliente-Servidor de 2 capas

El servidor responde con sus propios recursos, no requiere otra aplicación/servidor para proporcionar parte del servicio



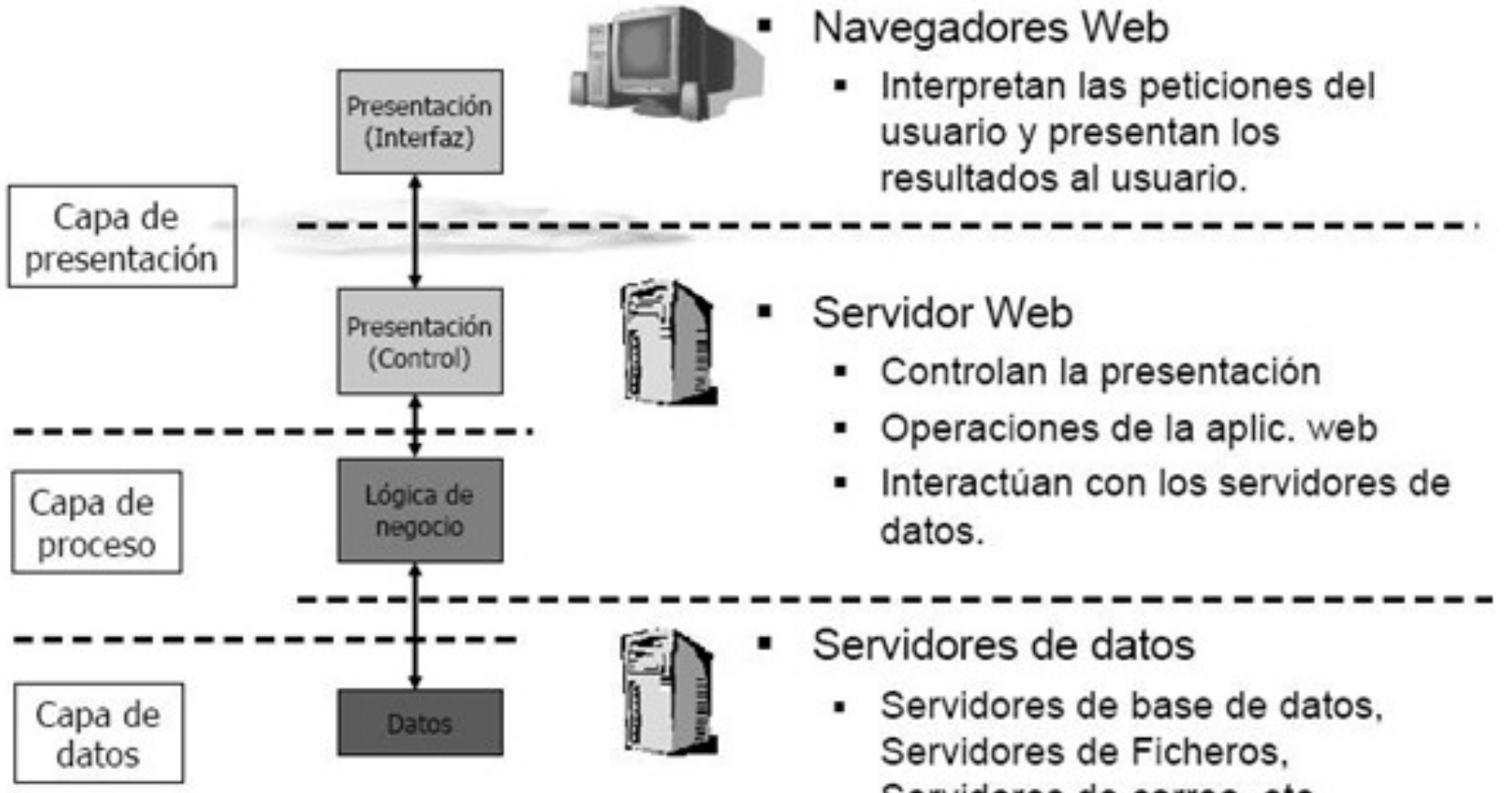
Cliente-Servidor de 2 capas

Actividad: Piensa un ejemplo de esta arquitectura. Comparte con el grupo de forma razonada tu ejemplo



Cliente-Servidor de 3 capas

El servidor requiere otra aplicación/servidor para proporcionar parte del servicio

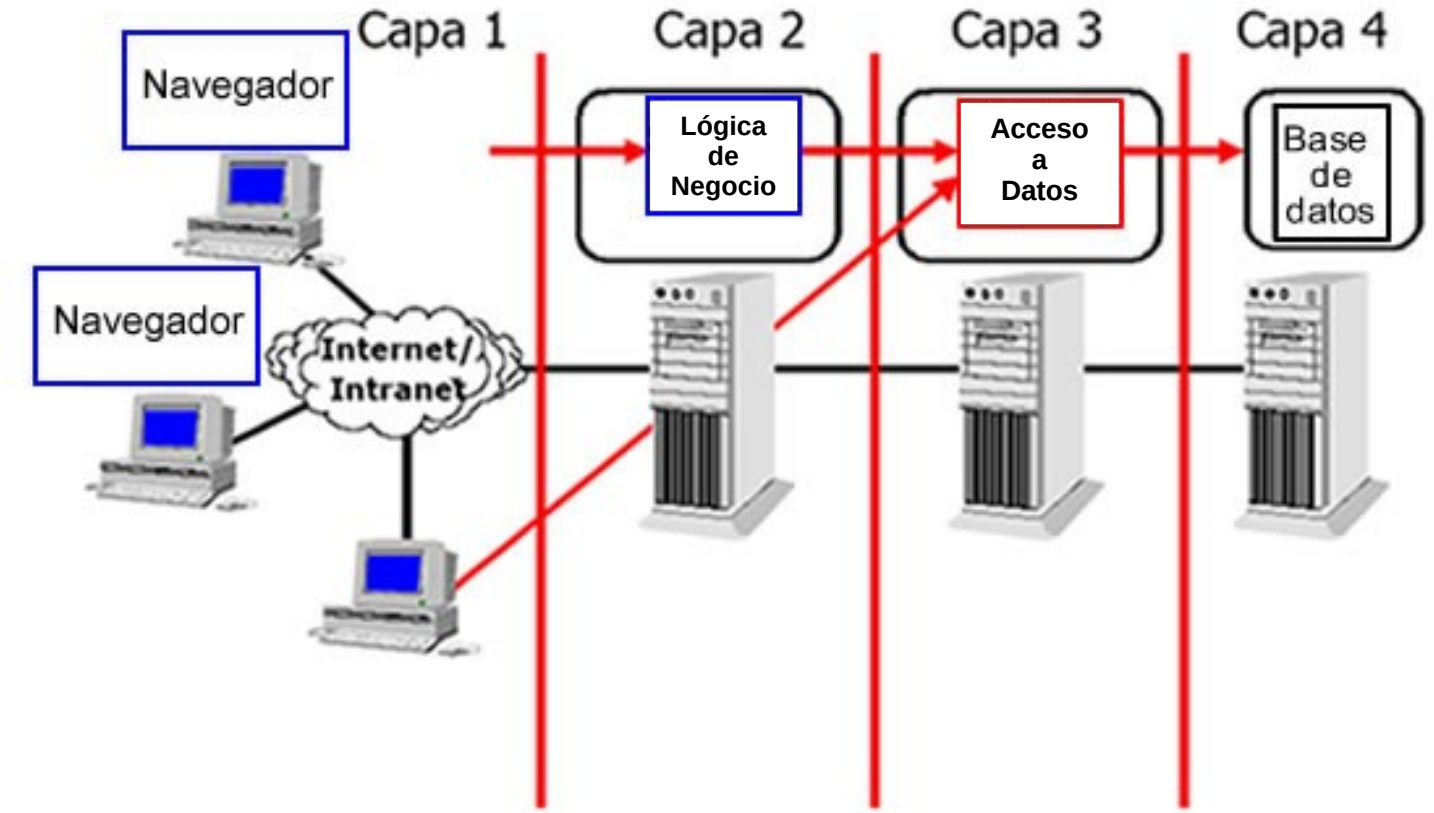


Cliente-Servidor de 3 capas

Actividad: Piensa un ejemplo de esta arquitectura. Comparte tu ejemplo razonadamente

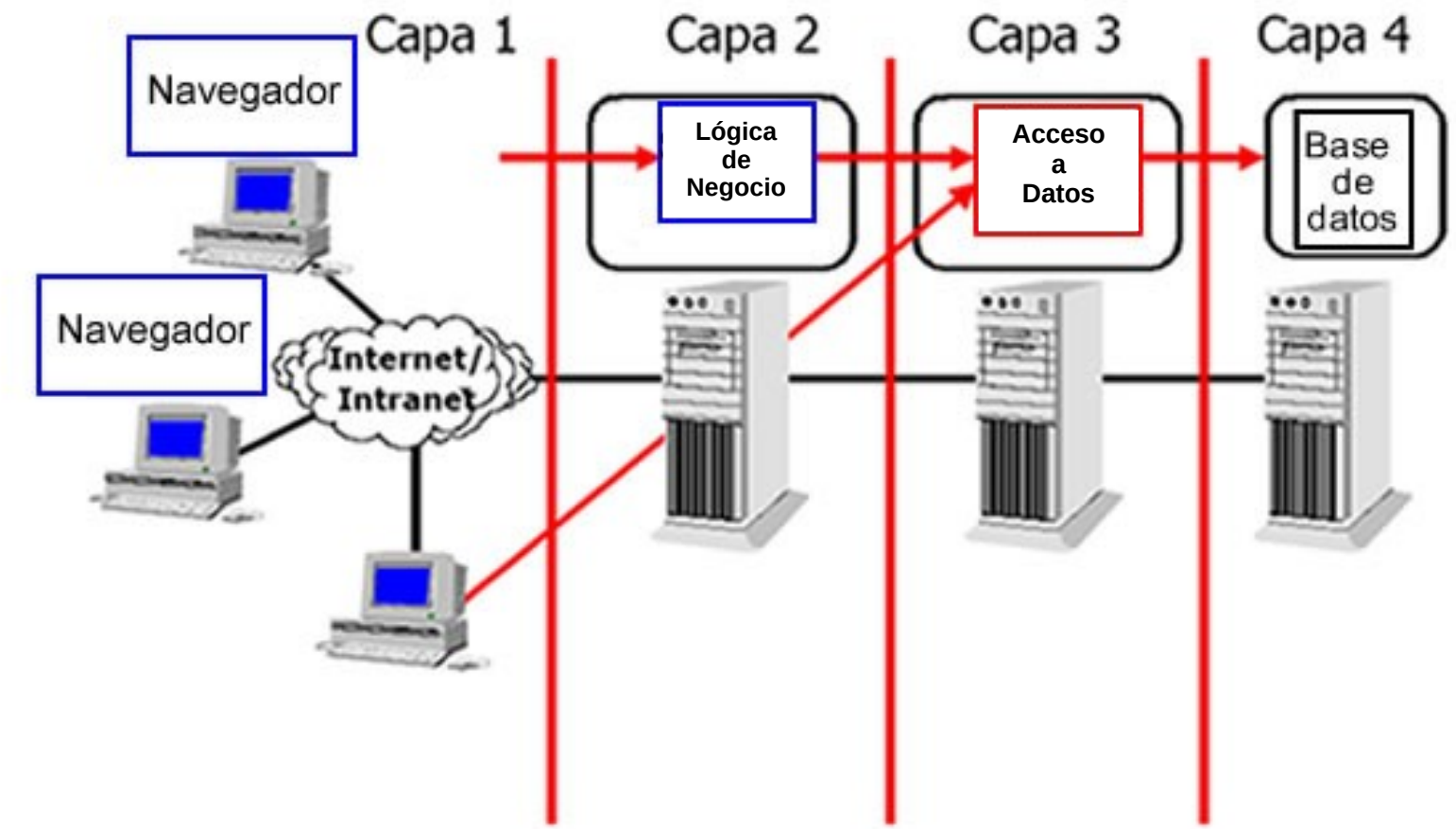
Cliente-Servidor de N capas

Los diferentes procesos están distribuidos en diversas capas, lógicas y/o físicas, ejecutándose en diferentes equipos



Cliente-Servidor de N capas

Actividad: Piensa un ejemplo de esta arquitectura. Comparte tu ejemplo razonadamente



Aplicaciones Web

- Proporcionada por un servidor Web y utilizada por usuarios que se conectan desde cualquier punto vía clientes web (navegadores)
- La colección de páginas son en una buena parte dinámicas (ASP, PHP, etc.), y están agrupadas lógicamente para dar un servicio al usuario.
- El acceso a las páginas está agrupado también en el tiempo (sesión).
- Ejemplos: venta de libros, reserva de billetes, etc.

Aplicaciones Web

Al observar la capacidad de las aplicaciones web de comunicarse con los usuarios, las podemos clasificar en:

- Aplicaciones Web Estáticas
- Aplicaciones Web Dinámicas
- Aplicaciones Web Interactivas

Aplicaciones Web Estáticas

El cliente recibe una página web desde el servidor, que no conlleva ningún tipo de acción, ni en la propia página, ni genera ninguna respuesta por parte del servidor. Utilizan HTML para la organización visual de la información.

```
001  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
002  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="es-ES" lang="es-ES">
003     <head>
004        <title>T&iacute;tulo de la p&aacute;gina</title>
005        <meta http-equiv="content-type" content="text/html; charset=utf-8">
006     </head>
007     <body>
008        <p>Ha accedido a las 9:20</p>
009     </body>
010  </html>
```

Aplicaciones Web Dinámicas

La interacción del cliente con la página web recibida desde el servidor produce algún cambio en la visualización de la misma (cambio de formato, ocultación de partes de la página, comienzo de animaciones, aparición de elementos nuevos, etc.). Incluyen DHTML, Flash, CSS, JavaScript, etc.

```
001  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
002  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="es-ES" lang="es-ES">
003          <head>
004                  <title>T&iacute;tulo de la p&aacute;gina</title>
005                  <meta http-equiv="content-type" content="text/html; charset=utf-8">
006          </head>
007          <body>
008                  <p>Ha accedido a las <script type='text/javascript'>var hora=new
     Date();document.write(hora.getHours()+':'+hora.getMinutes());</script></p>
009          </body>
010  </html>
```

Aplicaciones Web Interactivas

- La interacción del cliente con la página web recibida desde el servidor hace que se genere un diálogo entre ambos.
- Son las aplicaciones que más se utilizan en Internet actualmente.
- En el lado cliente encontramos HTML, controles ActiveX, Flash, applets, AJAX, etc.
- En el lado servidor se utilizan lenguajes embebidos en código HTML como PHP, ASP, JSP, enlaces a ejecutables CGI, servlets, ASP.net.

Ejecución de código en un servidor Web

Para ejecutar código en un servidor Web se requieren los siguientes componentes:

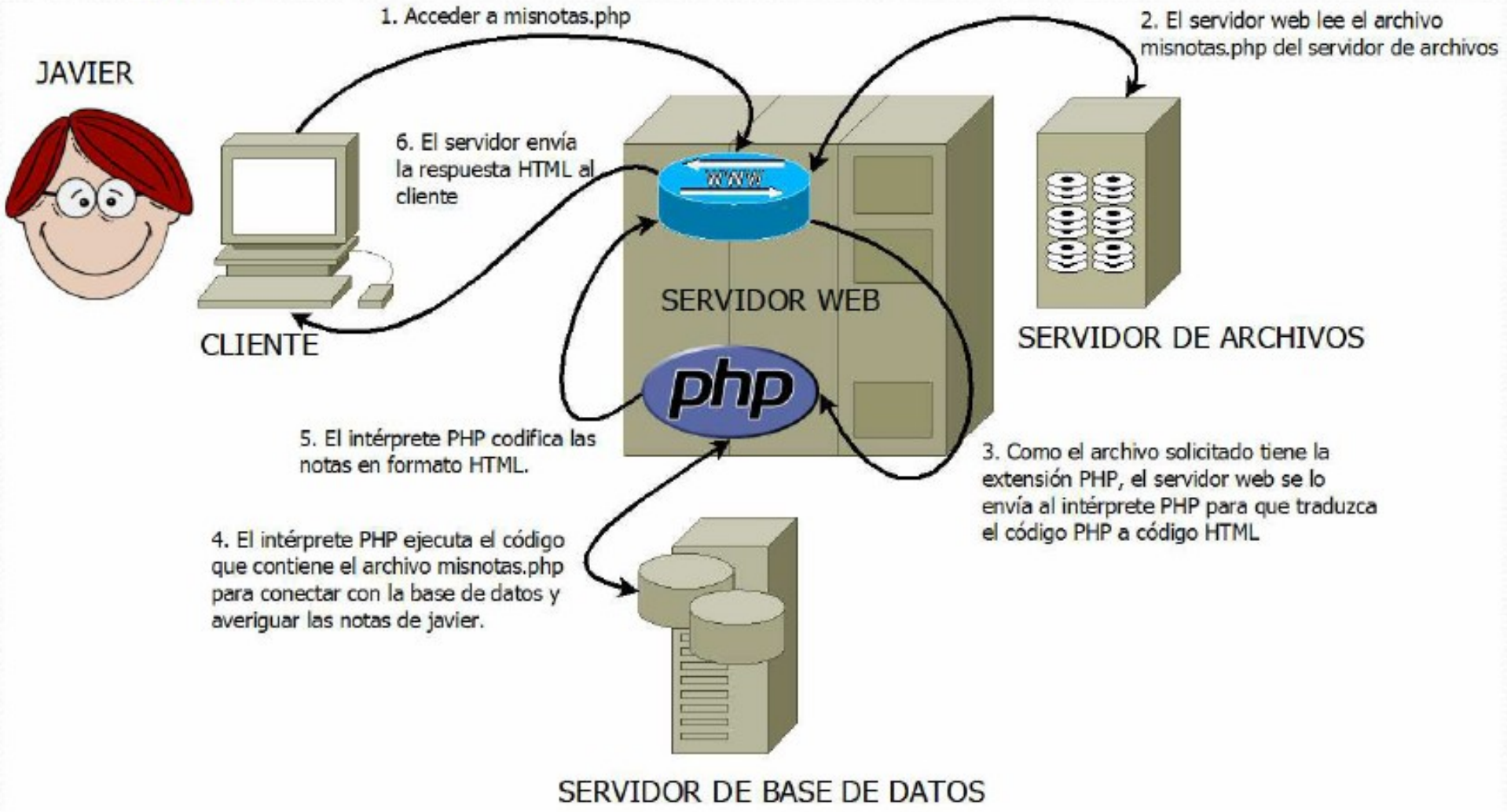
- **Servidor Web**: recibe las peticiones, elige el módulo que se encarga de la ejecución del código y se comunica con él
- **Módulo encargado de ejecutar el código**: para poder generar la página web resultante debe integrarse con el servidor web y depende del lenguaje y tecnología en que se programa la aplicación web
- **Aplicación de BD**: normalmente será un SGBD que almacene los datos
- **Lenguaje de programación** en el que se desarrollan las aplicaciones

Servidores de Aplicaciones

- Los **servidores de aplicaciones** ejecutan aplicaciones web o aplicaciones de escritorio.
- Constan lenguajes de programación, conectores de BD y el código necesario para implementar, configurar, administrar y conectar estos componentes en un servidor Web
- Típicamente incluyen también middleware (software de conectividad) que les permite intercomunicarse con varios servicios, para confiabilidad, seguridad, no-repudio, etc.

Proceso consulta Aplicación Web

El proceso que se realiza a la hora de visitar una página PHP sería:



Lenguajes de programación

Los lenguajes de entorno servidor pueden ser:

- Lenguajes de scripting o interpretados
- Lenguajes compilados a código nativo
- Lenguajes compilados a código intermedio

Lenguajes de Scripting o interpretados

Los programas se ejecutan directamente a partir de su código fuente mediante un intérprete que procesa las líneas del programa y genera como resultado el código HTML

No se compilan, se interpretan las instrucciones de script directamente por lo que sólo requieren el intérprete

A este grupo pertenecen los lenguajes Perl, Python, PHP, Ruby y ASP

Lenguajes compilados a código nativo

El código fuente se traduce a código binario (dependiente del procesador) antes de ser ejecutado. De este modo, el programa en binario se ejecuta directamente al invocarlo desde el servidor Web

A este grupo pertenece los CGI (Common Gateway Interface) que se programan en lenguajes que produzca un ejecutable como C, Perl, C++, Java, etc.

Lenguajes compilados a código intermedio

El código fuente original se traduce a un código intermedio (bytecode, independiente del procesador) antes de ser ejecutado en varias plataformas distintas

A este grupo pertenecen los lenguajes Java Server Pages o JSP que se ejecuta en servidores que permiten Servlets (código embebido que es compilado y almacenado en el contenedor de servlets del servidor y, tras la primera ejecución, se carga en memoria para agilizar posteriores invocaciones. El servlet queda activo escuchando peticiones para servirlas)

Patrón de diseño MVC

En el desarrollo Web se usa el patón de diseño **MVC** (Model, View, Controller) para separar las capas de persistencia o datos (modelo), presentación o vista y lógica o de negocio (controlador). Así pues:

- El **Modelo** contiene una representación de los datos que maneja el sistema y sus mecanismos de persistencia.
- La **Vista**, o interfaz de usuario, compone la información que se envía al cliente y los mecanismos interacción con éste.
- El **Controlador** actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de la lógica o modelo de negocio de cada aplicación.

Herramientas de programación

Los instrumentos involucrados en el desarrollo web se pueden clasificar en:

- Navegadores: Internet Explorer, Google Chrome, Mozilla firefox, etc.
- Editores de documentos
- Entornos de programación (IDE): Eclipse, NetBeans, Visual Studio Code, Dreamweaver, FrontPage, Sublime Text, etc.
- Herramientas para la creación y administración de bases de datos
- Herramientas para el tratamiento de imágenes.

Entornos de Desarrollo Integrado

Los Entornos de Desarrollo Integrados (IDE) aportan los siguientes elementos:

- Resaltado de texto: Distinto color para los diferentes elementos del lenguaje: sentencias, variables, comentarios, etc.
- Indentado automático para diferenciar de forma clara los distintos bloques de un programa.
- Completado automático.
- Navegación en el código. Permite buscar de forma sencilla elementos dentro del texto

Entornos de Desarrollo Integrado II

- Generación automática de código creando la estructura básica (esqueleto o stub), para que sólo haya que rellenarla.
- Ejecución y depuración.
- Gestión de versiones. Combinado con un sistema de control de versiones, el IDE te puede ayudar a guardar copias del estado del proyecto a lo largo del tiempo, para poder revertir los cambios realizados.

Frameworks

Un Framework es una estructura o esqueleto software que permite aplicar unos patrones de diseño (en nuestro caso MVC) en la construcción de nuestra aplicación Web.

Entre muchas ventajas, permite la modularidad y la reusabilidad de código permitiendo aplicar múltiples interfaces para distintas vistas (web, móvil, servicio web, etc.)

Algunas funcionalidades que aporta son:

- Sistema de plantillas Web
- Mecanismos de seguridad y autenticación
- Acceso, mapeo y configuración de BD
- Mapeo de “URL amigable” (Ej: "/product.php?cat=home&topic=chair" puede accederse desde "/product/home/chair")

Tecnologías de Servidor

Según el modo de procesar las peticiones del cliente, podemos clasificar los servidores web en:

- Basados en Procesos
- Basados en Hilos
- Dirigidos por Eventos
- Implementados en el Kernel

Servidor basado en Procesos

1. Un proceso principal escucha posibles peticiones de clientes
2. Cuando llega una petición, el proceso se duplica creando una copia exacta (*fork*)
3. La copia atiende la petición mientras el proceso principal sigue escuchando nuevas peticiones

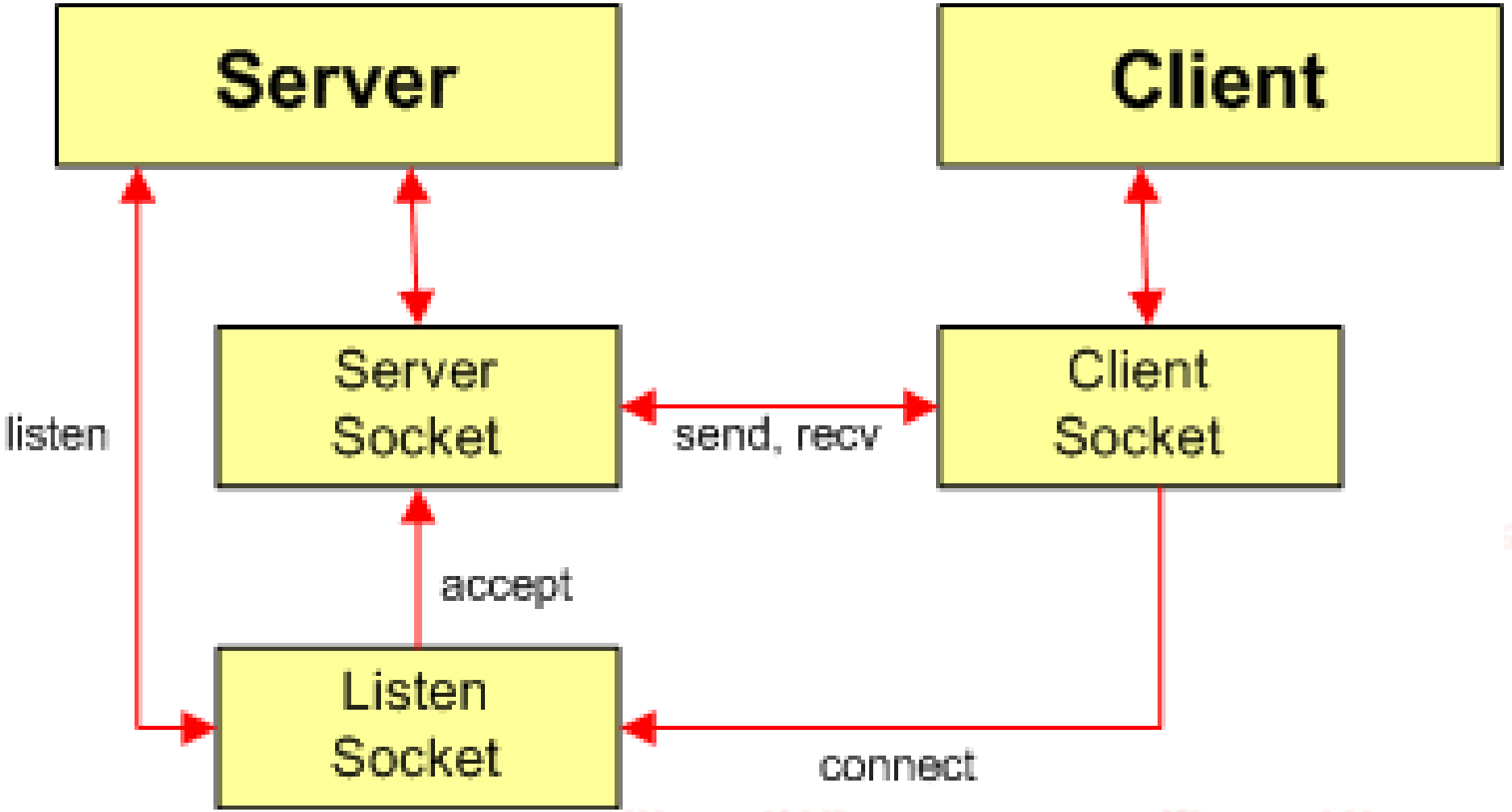
Servidor basado en Hilos

1. Un proceso principal escucha posibles peticiones de clientes
2. Cuando llega una petición, el proceso crea un Hilo de ejecución (*thread*) que comparten el mismo espacio de memoria (interbloqueo)
3. El hilo atiende la petición mientras el proceso principal sigue escuchando nuevas peticiones

Servidor dirigido por eventos

- Se basa en *sockets* no bloqueantes.
- Socket: espacio de memoria para compartir entre procesos de dos aplicaciones en dos máquinas distintas (cliente/servidor)
- Las lecturas/escrituras entre sockets son asíncronas y bidireccionales
- Se identifican mediante IP:puerto
- La concurrencia de procesamiento es simulada: hay un único proceso y un sólo hilo atendiendo las conexiones gestionadas por sockets

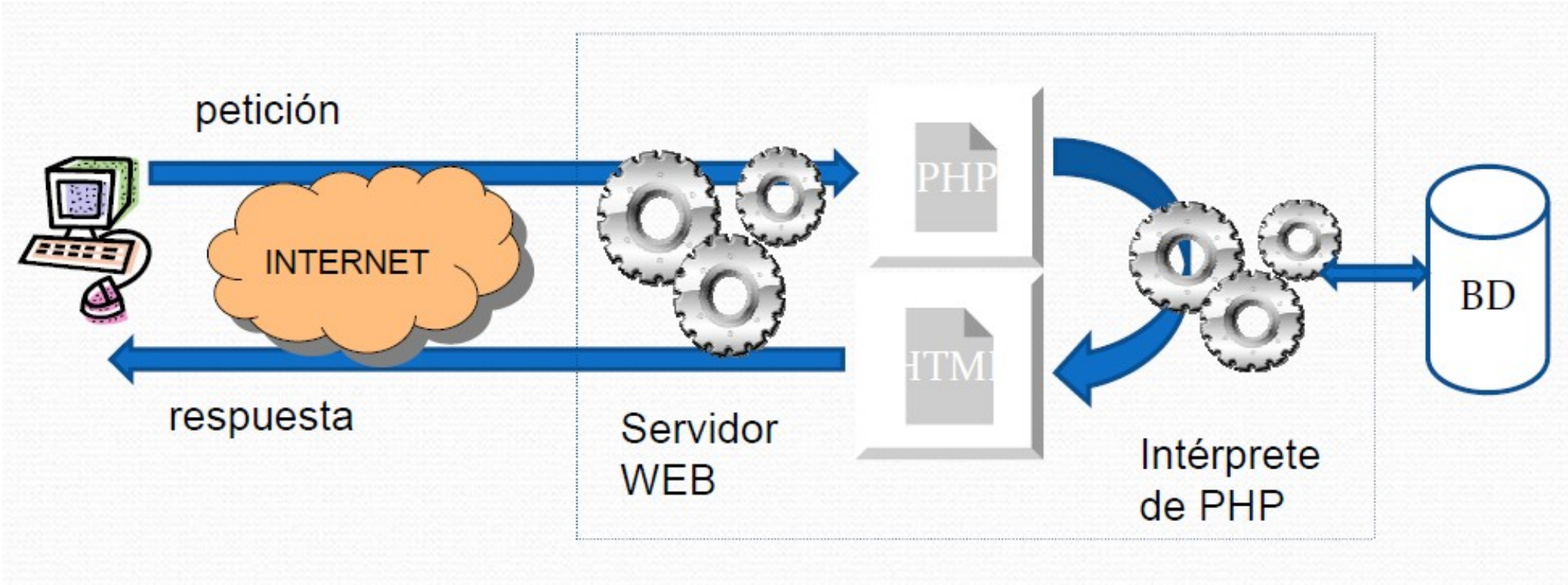
Servidor dirigido por eventos



Servidor implementado en el Kernel

- Se usa un espacio de trabajo perteneciente al SO y no en el área de usuario
- En la práctica o mundo real, tiene muchos problemas e inconvenientes
- Cualquier problema que se produce a nivel de Kernel puede inutilizar el SO

Obtención del lenguaje de marcas para mostrar en el cliente



El documento PHP, una vez interpretado correctamente en el servidor, produce una página HTML que será enviada al cliente.

Tarea Servidores

De los servidores Apache, Internet Information Services, Nginx, Lighttpd y Sun Java System Web Server deberéis investigar:

- Plataforma en la que se ejecutan
- Lenguajes de programación que interpretan
- Propietario
- Cuota de Mercado de 2023 (o 2024 si está disponible)

Realizad una tabla comparativa tras el estudio con una conclusión personal y razonada eligiendo uno de ellos.

Subid el documento con el nombre **Alumno_Servidores.odt** como por ejemplo SilviaVilar_Servidores.odt