

Descifrando la Web: Un Viaje de la Superficie al Servidor

Explorando las capas de tecnología
que impulsan el World Wide Web.



El Origen: Una Red para Compartir Conocimiento



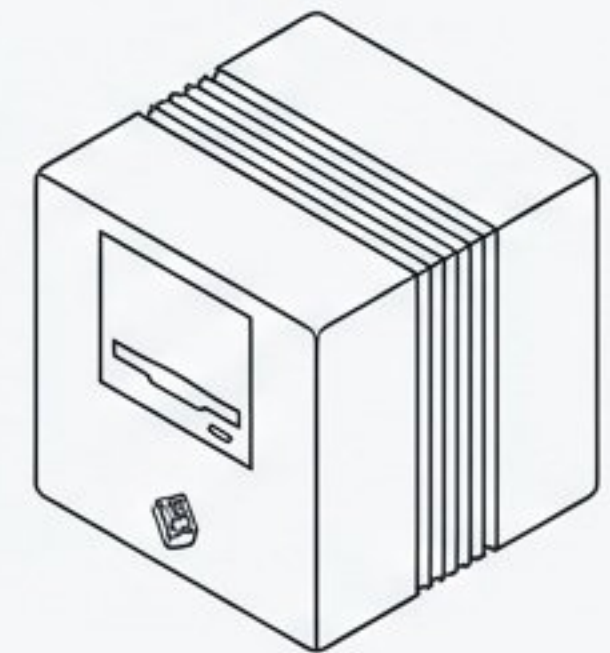
El Problema Original

Facilitar el intercambio de textos de investigación entre científicos, permitiendo saltar directamente a las referencias de un artículo.

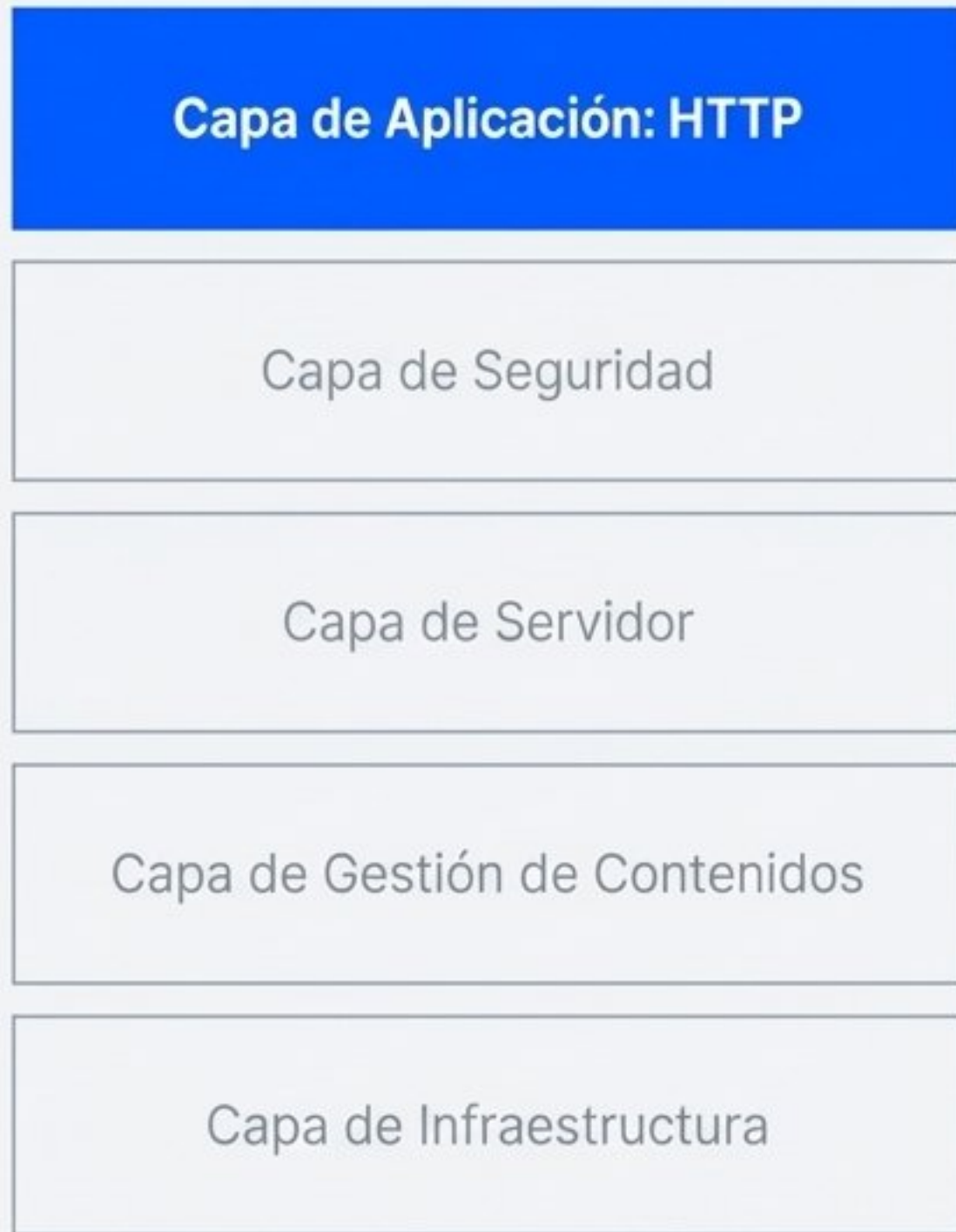
La Solución Revolucionaria

El Hipertexto. Un sistema que enlaza documentos, permitiendo una lectura no secuencial.

La Solución Revolucionaria: El Hipertexto



Capa 1: HTTP, El Lenguaje de la Web



Definición: HTTP (Hypertext Transfer Protocol) es el protocolo de comunicación que permite las transferencias de información en la World Wide Web.

Arquitectura: Funciona bajo un modelo cliente-servidor. El cliente (navegador) solicita y el servidor responde.

Un Protocolo sin Estado: HTTP no guarda información sobre conexiones anteriores. Para mantener el estado se utilizan *cookies* (cliente) y *sesiones* (servidor).



Puerto 80: HTTP



Puerto 443: HTTPS

La Anatomía de una Conversación Digital



Solicitud del Usuario

El usuario introduce una URI en el navegador.



Resolución DNS

El sistema de nombres de dominio (DNS) traduce el dominio a una dirección IP.



Conexión y Petición

El navegador abre una conexión TCP (puerto 80) y envía la solicitud HTTP al servidor.



Procesamiento del Servidor

El servidor busca el recurso. Si lo encuentra, lo prepara. Si no, genera un error.



Respuesta y Renderizado

El servidor envía la respuesta HTTP. El navegador interpreta el código y muestra la página. La conexión se libera.

La Petición del Cliente: ¿Qué Se Pide y Cómo?

1. Línea de Solicitud

- **Método:** `GET` (La acción)
- **URI:** `/articles/news/today.asp` (El recurso)
- **Versión:** `HTTP/1.1` (El protocolo)

2. Encabezados (Headers)

Metadatos sobre la solicitud y el cliente.

```
GET /articles/news/today.asp HTTP/1.1
Host: localhost
User-Agent: Mozilla/4.0 (...)
Accept-Language: en-us
Accept-Language: en-us
```

3. Cuerpo de la Solicitud (Opcional)

Datos enviados al servidor, comúnmente con el método `POST`.

La Respuesta del Servidor: El Veredicto y el Contenido

1. Línea de Estado

- **Versión:** `HTTP/1.1`
- **Código de Estado:** `200`
- **Frase Explicativa:** `OK`

2. Encabezados de Respuesta

Metadatos sobre la respuesta y el servidor.

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0
Content-Type: text/html
Content-Length: 2291

<HTML><BODY>...
```

3. Cuerpo de la Respuesta

El recurso solicitado (ej. el código HTML).

El Léxico de HTTP: Métodos y Códigos de Estado

Métodos Principales (Los "Verbos")



GET

Solicita una representación del recurso.



POST

Envía datos para ser procesados (ej. un formulario). Los datos van en el cuerpo.



PUT

Sube o crea un recurso en el servidor.



DELETE

Borra el recurso especificado.



HEAD

Pide la misma respuesta que un GET, pero sin el cuerpo.

Códigos de Estado Comunes



1xx (Informativos)

Petición recibida, continuando proceso.



2xx (Éxito)

`200 OK` - La solicitud tuvo éxito.



3xx (Redirección)

`303 See Other` - El recurso se ha movido.



4xx (Error del Cliente)

`404 Not Found` - El recurso no existe.



5xx (Error del Servidor)

`500 Internal Server Error` - Error inesperado en el servidor.

Capa 2: HTTPS, El Transporte Blindado de la Web

HTTP

Capa de Seguridad: SSL/TLS

El Problema

HTTP viaja sin cifrar y puede ser interceptado. No es seguro para información sensible como contraseñas o datos financieros.

La Solución

HTTPS (HyperText Transfer Protocol Secure). Utiliza una capa de seguridad (SSL/TLS) para cifrar la comunicación.

Los Cuatro Pilares de la Seguridad



Secreto (Confidencialidad): Mantiene la información privada.



Validación (Autenticación): Determina la identidad de las partes.



Integridad: Asegura que el mensaje no ha sido alterado.



No Repudio: Garantiza que el emisor no puede negar el envío.

Las Llaves de la Confianza: Criptografía y Certificados Digitales

Criptografía Asimétrica (Clave Pública/Privada)



Cada parte tiene un par de claves: una **pública** (se comparte) y una **privada** (secreta).

Certificado Digital (Estándar X.509)



Es un documento digital que vincula una identidad con una clave pública.

Emitido por una **Autoridad de Certificación (CA)**, un tercero de confianza.

Funciona como un “pasaporte de Internet”, verificando que el servidor es quien dice ser.

Capa 3: El Servidor Web, La Sala de Máquinas



¿Qué es?

Un programa que escucha permanentemente peticiones HTTP/HTTPS y proporciona los recursos solicitados.

Jugadores Clave



Apache HTTP Server



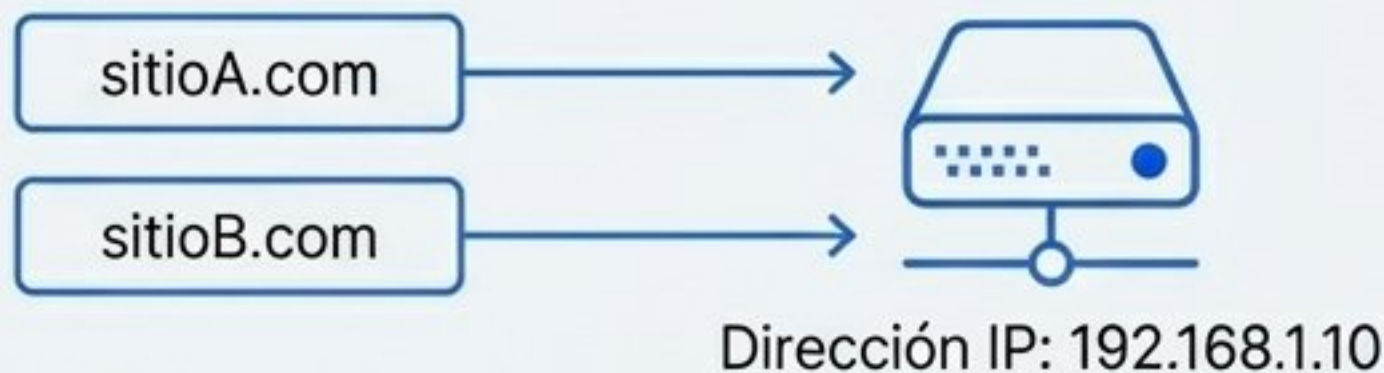
Microsoft IIS



Nginx

Funcionalidades Esenciales

- Servir páginas estáticas y dinámicas.
- **Hosts Virtuales:** Alojar múltiples sitios web en una sola máquina.
- Gestión de seguridad y autenticación.



Estático vs. Dinámico: Las Dos Caras del Contenido Web

Páginas Estáticas

Contenido fijo que no admite interacción del usuario.

El servidor simplemente envía el fichero (HTML, CSS, imagen) tal como está en el disco.

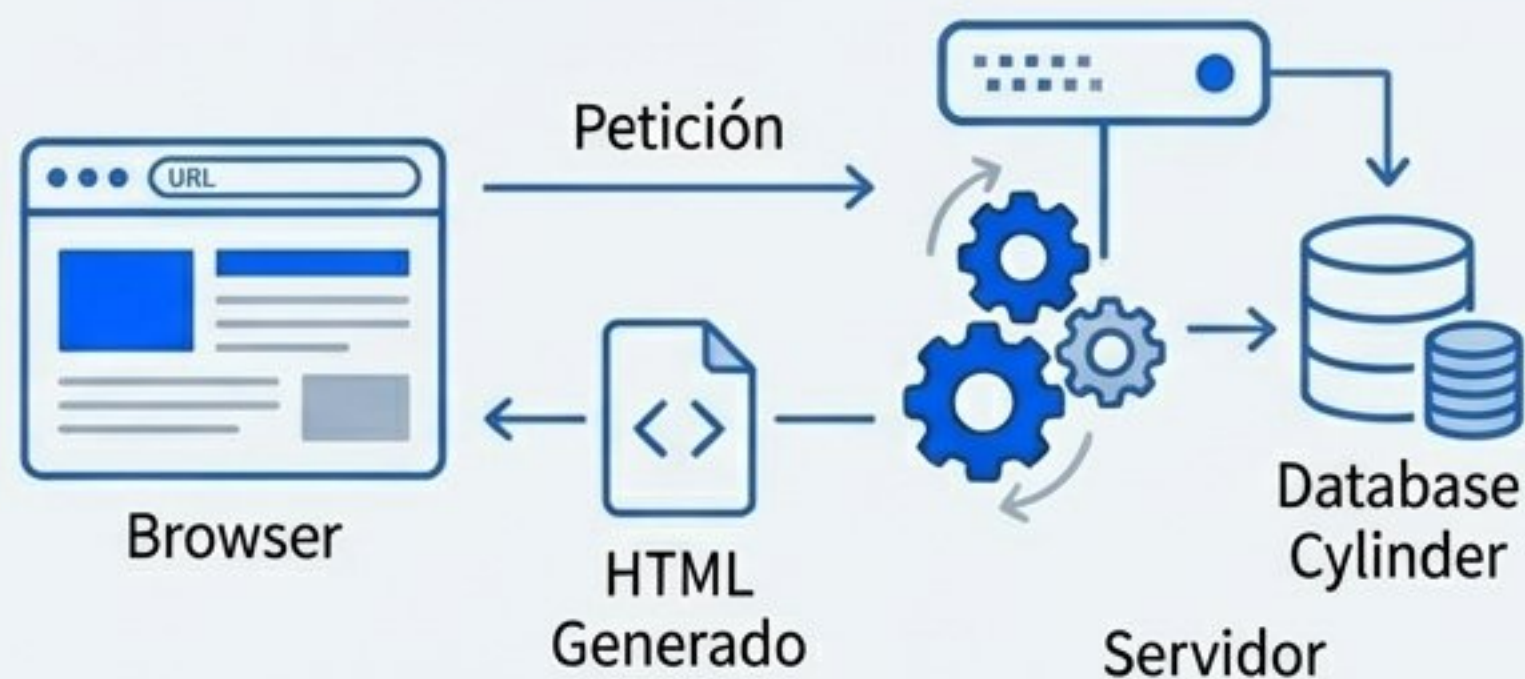


Páginas Dinámicas

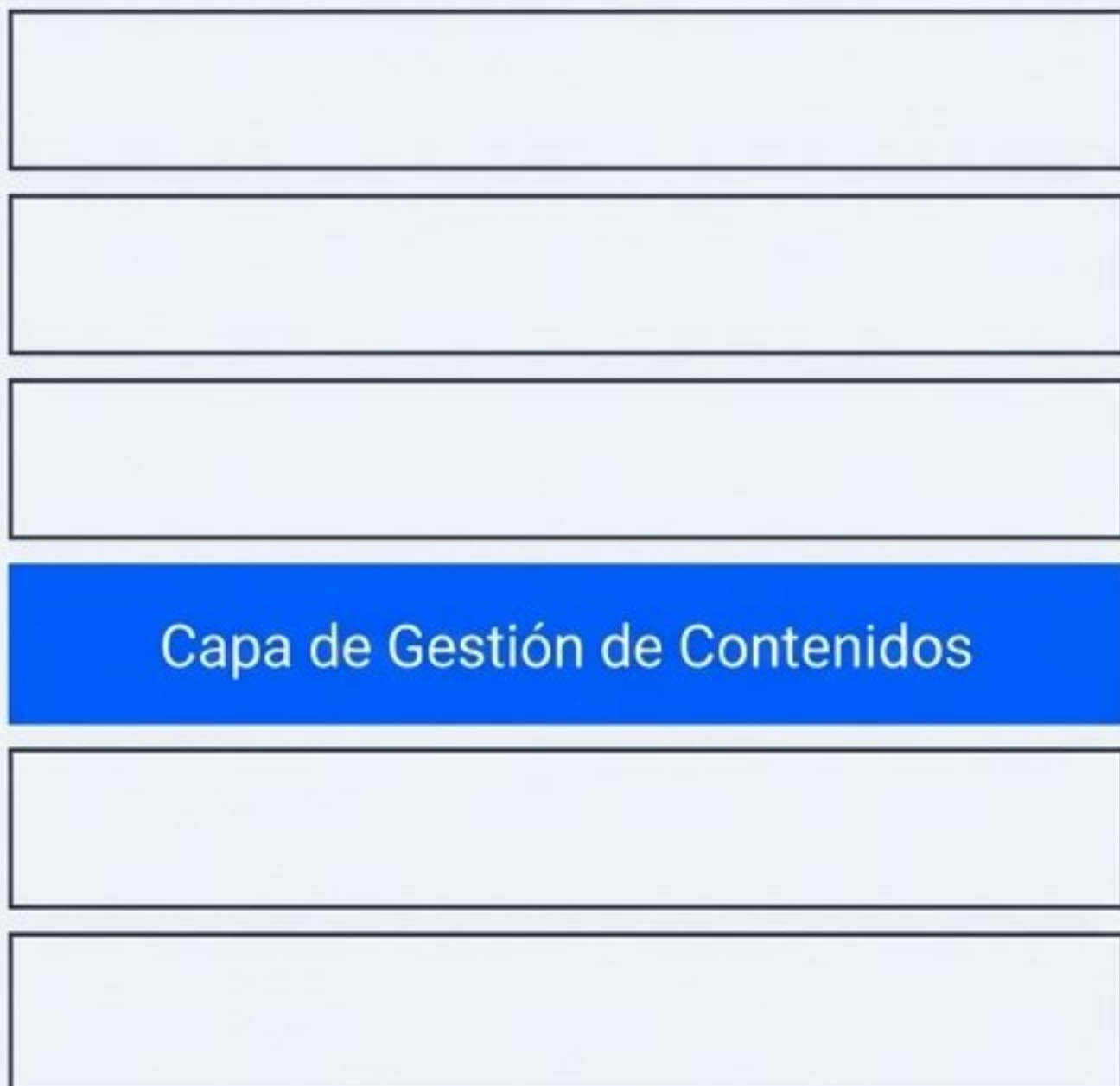
El contenido se genera en el momento de la solicitud.

Del Lado del Cliente: Procesamiento en el navegador (ej. JavaScript).

Del Lado del Servidor: El servidor ejecuta un script para generar el HTML. Lenguajes: **PHP, ASP.NET, JSP, Python, Ruby.**



Capa 4: El CMS, La Fábrica de Contenidos



¿Qué es un CMS (Content Management System)?

Un sistema de software que permite a los usuarios crear, gestionar y modificar contenido en un sitio web sin necesidad de conocimientos técnicos.

Ventajas Clave



Facilidad de Creación: Editores visuales (WYSIWYG).



Control de Usuarios y Roles: Permisos para administradores, editores, etc.



Separación de Contenido y Presentación.

Ecosistemas Populares

Portales/Blogs:



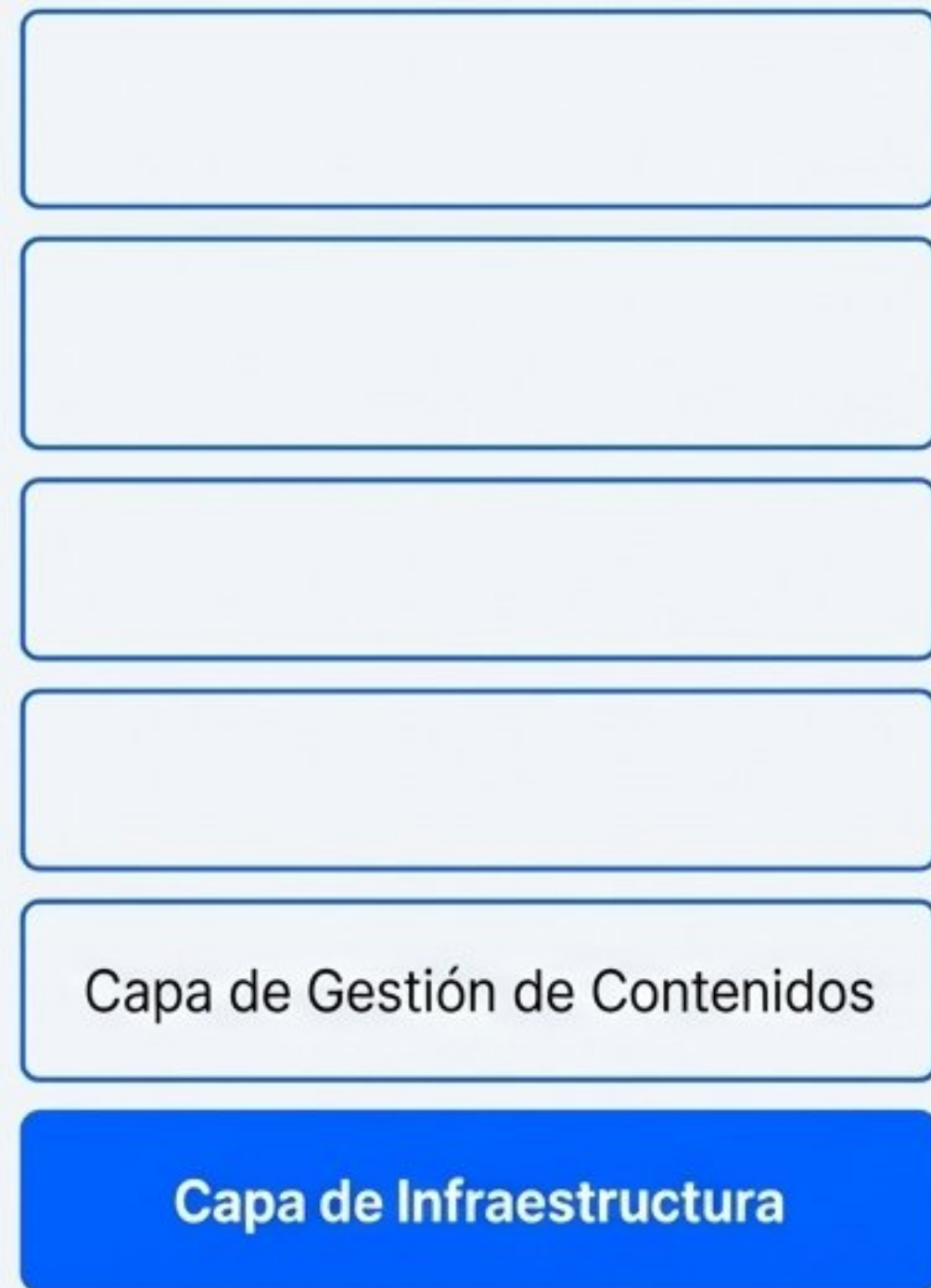
E-Learning:



Wikis:



Capa 5: El Taller del Desarrollador, Los Stacks de Infraestructura

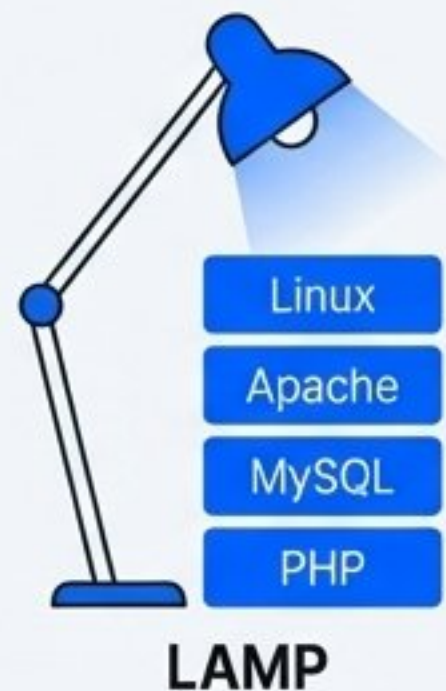


El Desafío

Instalar y configurar por separado un servidor web, una base de datos y un lenguaje de programación es complejo.

La Solución: Stacks de Software

Paquetes que contienen todo lo necesario para un entorno de desarrollo local.

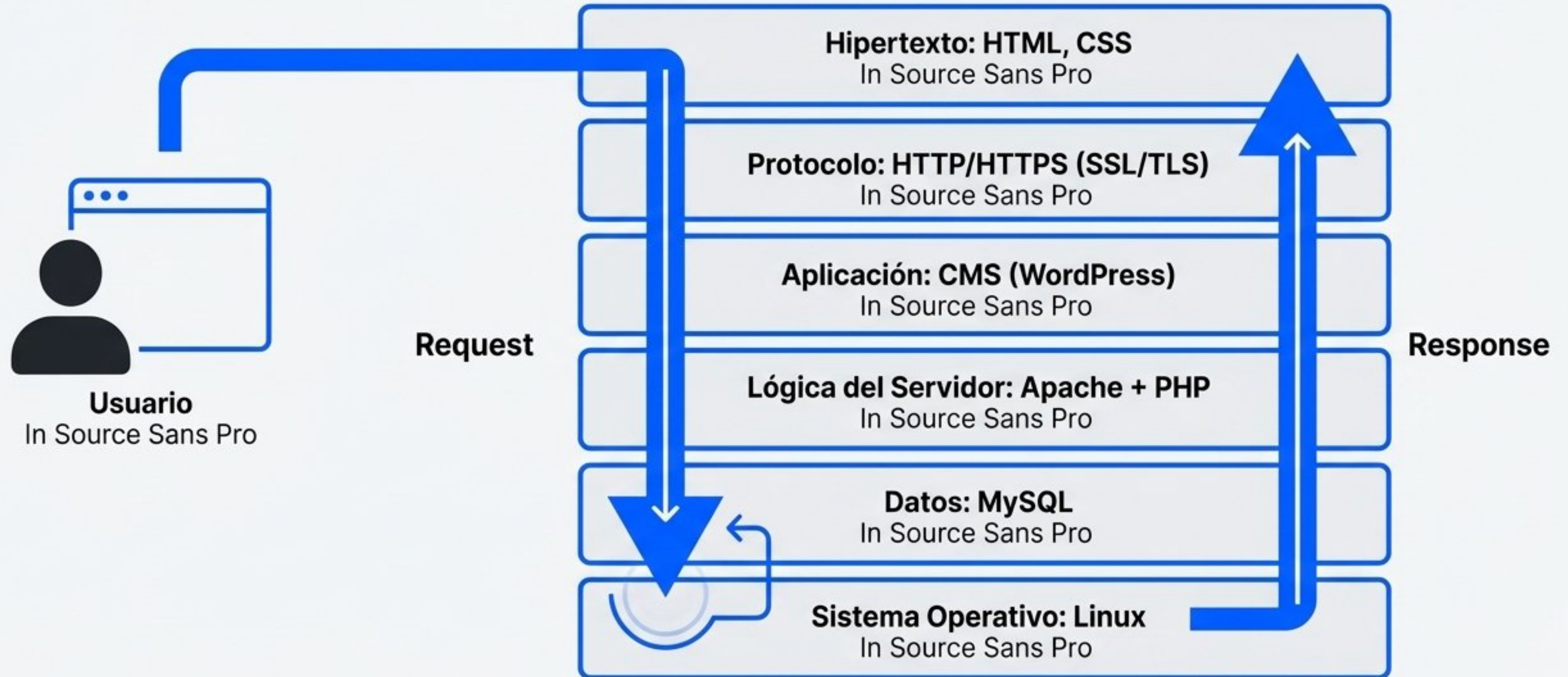


XAMPP
Una solución multiplataforma (X) que incluye Apache, MariaDB, PHP y Perl.



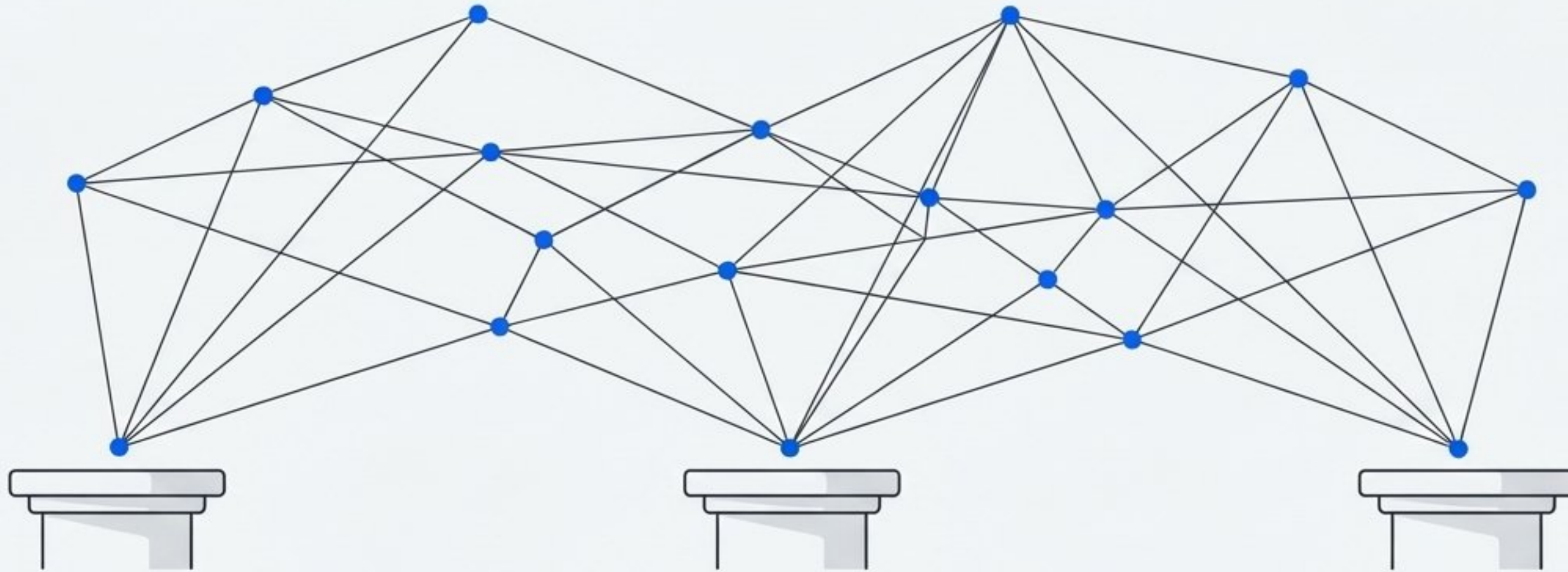
MEAN
MongoDB, ExpressJS, AngularJS, NodeJS.
Un stack basado completamente en JavaScript.

La Pila Completa: Una Síntesis de la Arquitectura Web



La Arquitectura Invisible

Más allá de los protocolos y el software, la Web se fundamenta en tres ideas revolucionarias que persisten hoy:



Descentralización

No existe una autoridad central. Cualquiera puede crear y enlazar documentos.

Localización Universal

Un mecanismo único (URI) para localizar cualquier recurso, en cualquier lugar.

Interfaz Uniforme

El navegador oculta la complejidad, ofreciendo un acceso simplificado a la información.