

#

# DESARROLLO WEB FULLSTACK

con Python y JavaScript

polotic  
misiones



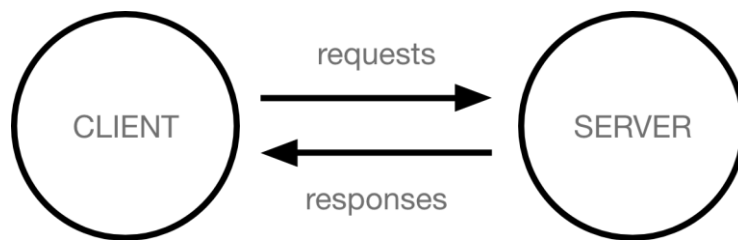
# FRONTEND

## HTML



# Introducción

Los equipos conectados a la web se denominan clientes y servidores. Un diagrama simplificado de cómo interactúan podría verse así:



Los **clientes** son los dispositivos conectados a Internet del usuario web típico (por ejemplo, su computadora conectada a su Wi-Fi o su teléfono conectado a su red móvil) y el software de acceso a la web disponible en esos dispositivos (generalmente un navegador web como Firefox o Chrome ).

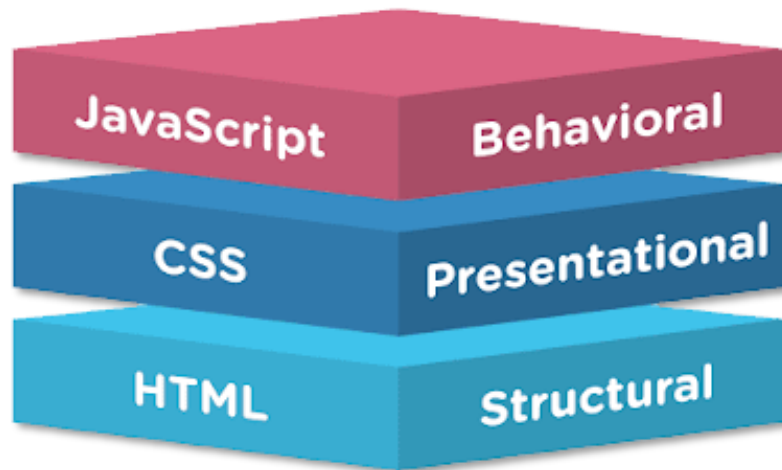
Los **servidores** son computadoras que almacenan páginas web, sitios o aplicaciones. Cuando un dispositivo cliente desea acceder a una página web, se descarga una copia de la página web desde el servidor a la máquina cliente para que se muestre en el navegador web del usuario.



# El Funcionamiento de la Web

1. El navegador accede al servidor DNS y encuentra la dirección real del servidor en el que se encuentra el sitio web.
2. El navegador envía un mensaje de solicitud HTTP (request) al servidor, pidiéndole que envíe una copia del sitio web al cliente. Este mensaje, y todos los demás datos enviados entre el cliente y el servidor, se envían a través de su conexión a Internet mediante el protocolo TCP / IP.
3. Si el servidor aprueba la solicitud del cliente, el servidor envía al cliente un mensaje "200 OK", que significa "¡Todo bien, aquí está su sitio web", y luego comienza a enviar los archivos del sitio web al navegador como una serie de pequeños trozos llamados paquetes de datos.
4. El navegador reúne los pequeños fragmentos en una página web completa y se la muestra.

# La Estructura de un Sitio o Aplicación Web





# Objetivos de HTML 5



1. Entregar **contenido enriquecido**, como gráficos y videos, al cliente sin la necesidad de complementos adicionales (es decir, Flash y Silverlight)
2. Proporcionar un mejor **soporte semántico** para la estructura de la página web mediante la introducción de nuevas etiquetas de elementos estructurales
3. Proveer un estándar de análisis más estricto para simplificar el manejo de errores y para simplificar la compatibilidad con documentos escritos con estándares anteriores.

# HTML

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>¡Hola!</title>
  </head>
  <body>
    ¡Hola Mundo!
  </body>
</html>
```

# HTML

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>¡Hola!</title>
  </head>
  <body>
    ¡Hola Mundo!
  </body>
</html>
```



# HTML

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>¡Hola!</title>
  </head>
  <body>
    ¡Hola Mundo!
  </body>
</html>
```

# HTML

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>¡Hola!</title>
  </head>
  <body>
    ¡Hola Mundo!
  </body>
</html>
```

# HTML

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>¡Hola!</title>
  </head>
  <body>
    ¡Hola Mundo!
  </body>
</html>
```

# HTML

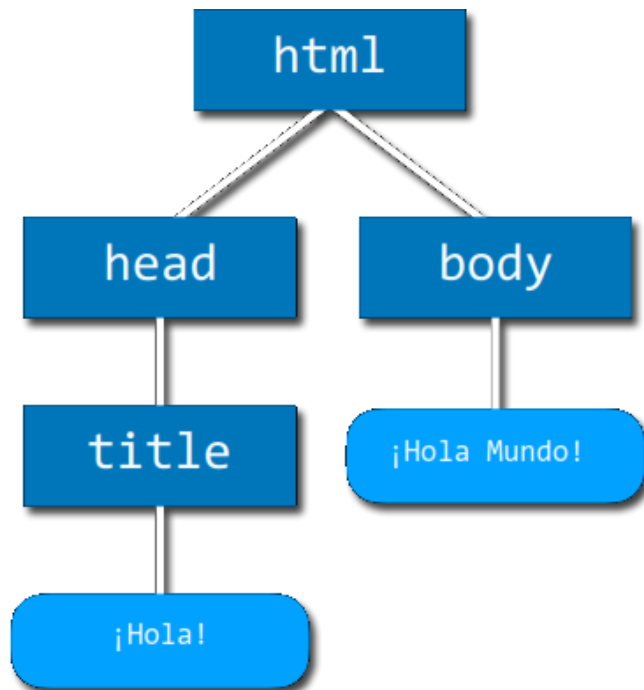
```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>¡Hola!</title>
  </head>
  <body>
    ¡Hola Mundo!
  </body>
</html>
```

# HTML

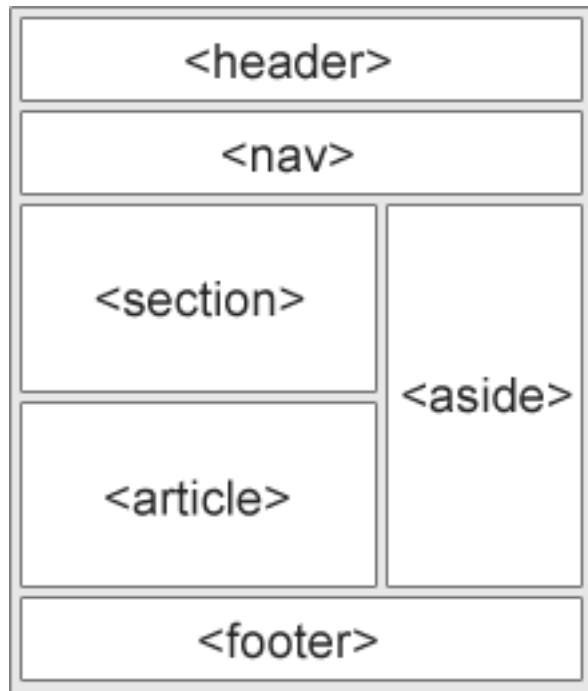
```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>¡Hola!</title>
  </head>
  <body>
    ¡Hola Mundo!
  </body>
</html>
```

# DOM – Document Object Model

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>¡Hola!</title>
  </head>
  <body>
    ¡Hola Mundo!
  </body>
</html>
```



# Layout de un Sitio Web



- `<header>`: define un encabezado para un documento o una sección
- `<nav>`: define un conjunto de enlaces de navegación
- `<section>`: define una sección en un documento
- `<article>`: define un contenido autónomo e independiente
- `<aside>`: define el contenido al costado del contenido principal (como una barra lateral o *sidebar*)
- `<footer>`: define un pie de página para un documento o una sección
- `<details>`: define detalles adicionales que el usuario puede abrir y cerrar a pedido.
- `<summary>`: define un encabezado para el elemento `<details>`



# Etiquetas comunes HTML

- `<h1>`, `<h2>`, ..., `<h6>`
- `<ol>`, `<ul>`
- `<img>`
- `<a>`
- `<table>`
- `<form>`
- ...





# Agregados de HTML5



- Soporte mejorado para **incrustar** contenido de **gráficos**, **audio** y **video** usando las nuevas etiquetas `<canvas>`, `<audio>` y `<video>`
- subprocessos múltiples a través de Web Workers
- Nuevas extensiones para la API de JavaScript, como la geolocalización
- Nueva funcionalidad de arrastrar y soltar elementos
- Funciones de almacenamiento y almacenamiento en caché
- Muchas etiquetas semánticas nuevas y nuevos controles de formulario para complementar la lógica estructural de las aplicaciones web modernas

# Ejemplo de un formulario simple

```
<form>
```

```
<label>Nombre Completo:</label>
```

```
<input type="text" id="nombre_completo" name="nombre_completo" placeholder="Aqui va su nombre" required>
```

```
<label>Dirección de Email:</label>
```

```
<input type="email" id="dir_email" name="dir_email" required>
```

```
<label>Fecha de arribo:</label>
```

```
<input type="date" id="fecha_arribo" name="fecha_arribo" required>
```

```
<label>Numero de noches:</label>
```

```
<input type="number" id="noches" name="noches" value="1" min="1" max="30" required>
```

```
<label>Cantidad de Visitantes:</label>
```

```
<input type="number" id="visitantes" name="visitantes" value="1" min="1" max="4" required>
```

```
<label>Codigo promocional:</label>
```

```
<input type="text" id="promo" name="promo" pattern="[A-Za-z0-9]{6}" title="Solo 6 caracteres alfanumericos.">
```

```
<input type="submit" value= "Solicitar Reserva" />
```

```
</form>
```



# Trabajemos Juntos



Llegó la hora del desafío:

- Intenten replicar (usando solo HTML) la estructura de la portada de Wikipedia en español (<https://es.wikipedia.org/wiki/Wikipedia:Portada>)



#

¡HASTA LA  
*próxima!*

[www.polotic.misiones.gob.ar](http://www.polotic.misiones.gob.ar)

[f](#) [@](#) [v](#) [d](#) /poloticmisiones

polotic  
misiones



Gobierno  
de Misiones

