



Aplicaciones Interactivas

1er Cuatrimestre 2025 – Ing. Santos Sanchez – Ing. Carlos
Santiago Yanzon

TABLE OF CONTENTS

01

Objetivos de la
materia

02

HTML | CSS

03

Mecanismos de
Evaluación

04

Régimen de
Aprobación

05

TPO

06

TPO – Criterios de
Evaluación



01

Introduccion

Objetivos de la materia

Web

Cómo se desarrolla una aplicación web utilizando tecnologías **front-end** (HTML5, CSS3, Javascript, React).

Servicios

Cómo desarrollar un servicio de procesamiento de datos para una aplicación web con tecnología **back-end** (Java + Springboot).

Integracion

Como debe ser una Integración Web con un servicio backend.



HTML

¿Qué es HTML?

- Lenguaje de marcado, no de programación
- Utiliza etiquetas (tags) para estructurar contenido
- Interpretado por los navegadores

```
<!DOCTYPE html>
<html>
  <head>
    <title>This is a title</title>
  </head>
  <body>
    <div>
      <p>Hello world!</p>
    </div>
  </body>
</html>
```

¿Para qué sirve HTML?

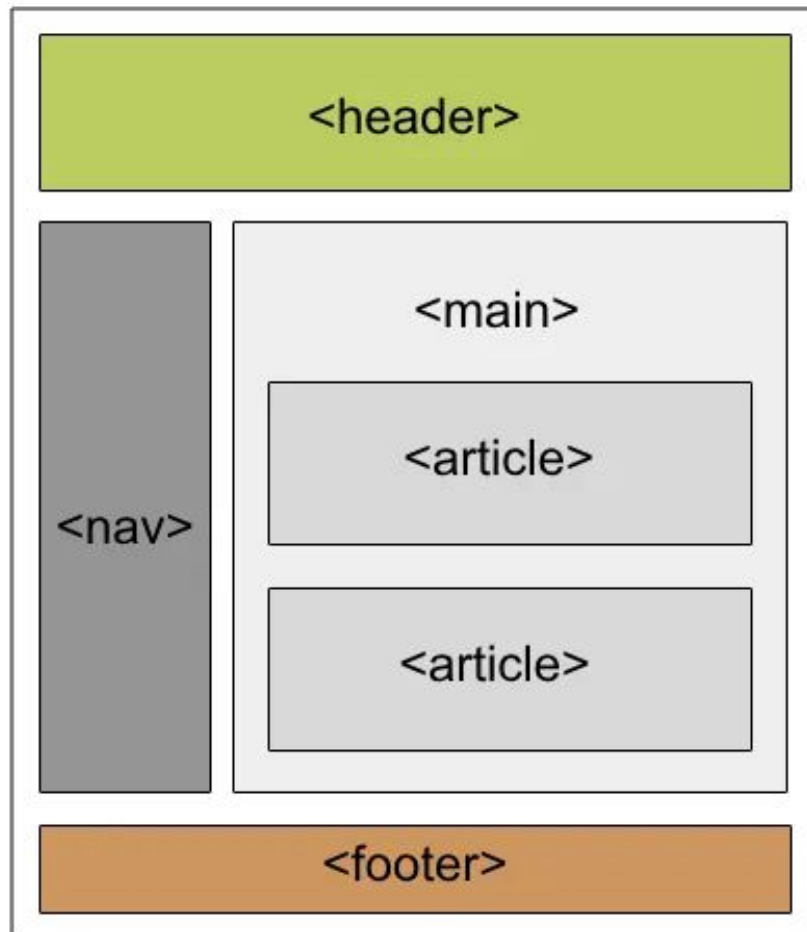
- Estructurar contenido con etiquetas semánticas
- Insertar imágenes, videos y formularios
- Enlazar páginas entre sí (**hipervínculos**)
- Base para aplicar estilos (**CSS**) y lógica (**JavaScript**)

HTML Semantico

No solo dice “que mostrar” si no que “representa”. Mejora accesibilidad, SEO, mantenibilidad y claridad

Algunas etiquetas

- **<header>** Representa la cabecera de una sección o página.
- **<nav>** Define una sección de navegación.
- **<main>** Contiene el contenido principal de la página.
- **<article>** Representa un contenido independiente, como un artículo de blog o una noticia.
- **<section>** Define una sección temática dentro del contenido.
- **<footer>** Representa el pie de una sección o página.



CSS

¿Qué es CSS?

- Lenguaje de estilos
- Define presentación visual de HTML
- Basado en reglas: selectores + declaraciones

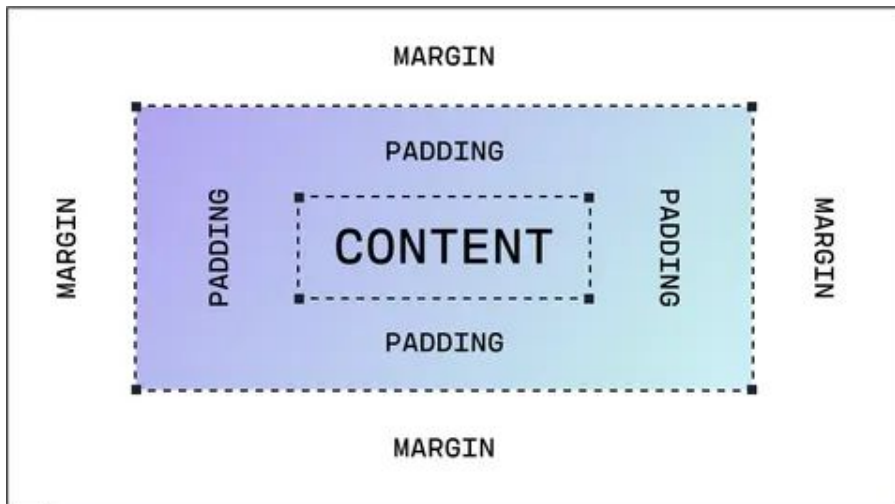
¿Para qué sirve CSS?

- Separar contenido de estilo
- Estilos responsivos
- Mejora experiencia visual
- Trabajar con frameworks (**Bootstrap**, **Tailwind**)



CSS - Declaraciones Básicas

Margin vs Padding






Unidad REM



CSS – Declaraciones Básicas

Colores

COLOR	NAME	HEX CODE	RGB CODE
	IndianRed	#CD5C5C	rgb(205, 92, 92)
	LightCoral	#F08080	rgb(240, 128, 128)
	Salmon	#FA8072	rgb(250, 128, 114)



CSS - posicionamiento

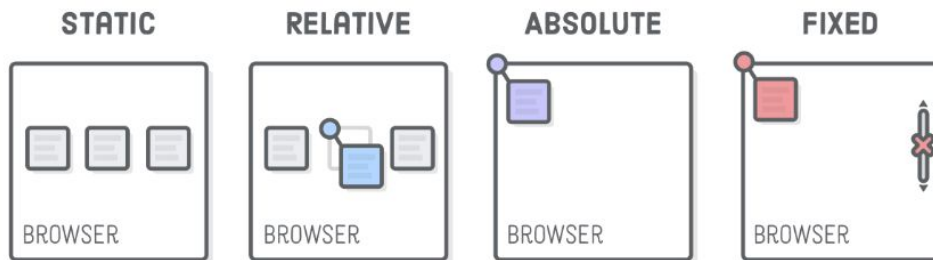
static: el elemento es posicionado de acuerdo al flujo normal del documento. Las propiedades top, right, bottom, left, and z-index no tienen efecto. Este es el valor por defecto

relative: El elemento es posicionado de acuerdo al flujo normal del documento, y luego es desplazado con relación a sí mismo, con base en los valores de top, right, bottom, and left

fixed: El elemento es removido del flujo normal del documento, sin crearse espacio alguno para el elemento en el esquema de la página. Es posicionado con relación al bloque contenedor inicial establecido por el viewport o ventana gráfica

absolute: El elemento es removido del flujo normal del documento. Es posicionado relativo a su ancestro posicionado más cercano, si lo hay; de lo contrario, se ubica relativo al bloque contenedor inicial.

Sticky: El elemento es posicionado de acuerdo al flujo normal del documento, y luego es desplazado con relación a su ancestro que se desplace más cercano y su bloque contenedor



CSS - Display

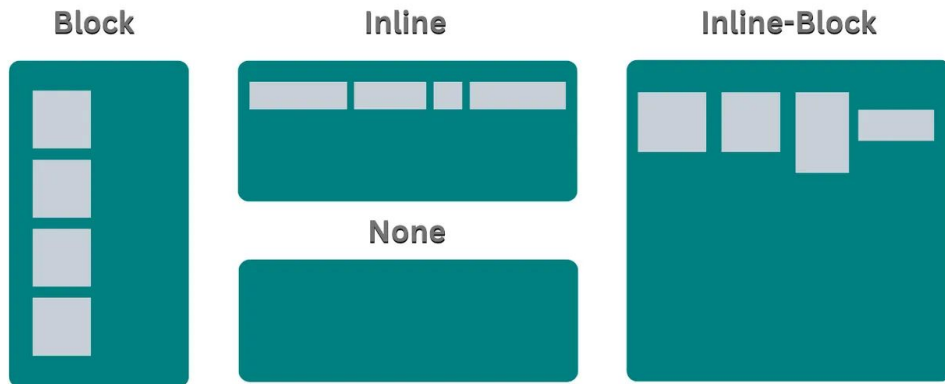
La propiedad CSS display especifica si un elemento es tratado como block or inline element y el diseño usado por sus hijos. Establece los tipos de visualización interna y externa de un elemento.

inline: el elemento inline se comporta como un inline-block, de mínima anchura

block: uno abajo del otro

inline-block: se comporta como inline pero se pueden definir height y width values

none: no lo muestra



CSS – Flexbox

Distribuye espacio entre elementos

- **flex-direction, flex-wrap, justify-content, align-items**
- [Playground – Flex](#)



CSS – Extra visuales

Mejoras

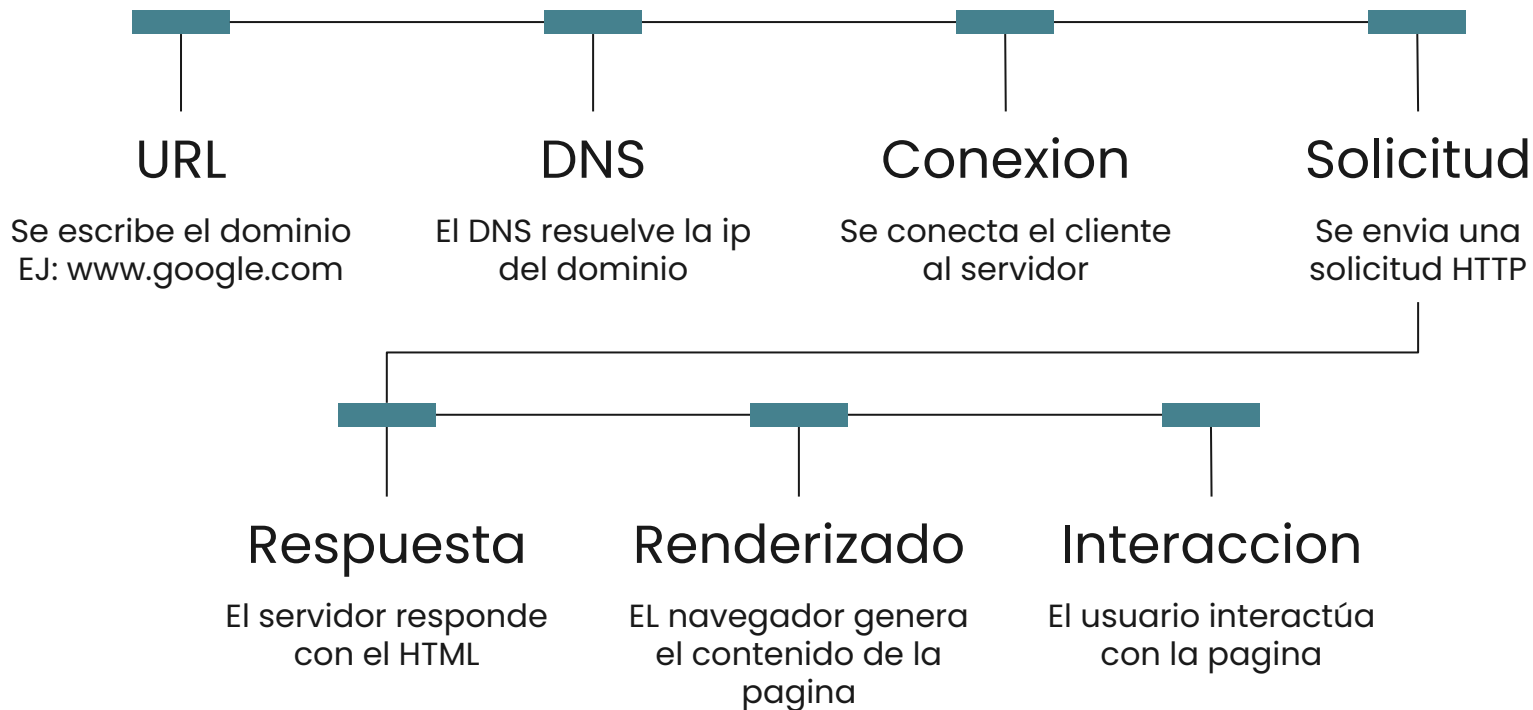
- **Animaciones** con [Animate.css](#)
- **Iconos** con [FontAwesome](#)
- **Tipografías** con Google Fonts

Media Queries

- Diseño adaptable a distintos dispositivos
- Sintaxis **@media**
- Frameworks que ya incorporan esto: Bootstrap, Materialize



¿Qué ocurre al cargar una página web?



02

Cursada y TPO

Mecanismos de evaluación

Trabajo práctico obligatorio

- Realizado en grupos compuestos por **5 personas**.
- El mismo contará de **dos presentaciones parciales y una entrega final**.

Participación activa en clase, practicas continuas en clases

Régimen de Aprobación

Aprobación de la cursada

- 75% de asistencia a las clases
- Examen Final Obligatorio: aprobación de 2 parciales con calificación ≥ 4 para encontrarse en condiciones de presentarse en los turnos de exámenes finales
- Se podrá rendir 1 solo recuperatorio de algunas de las dos entregas desaprobadas.

Promoción de la materia:

- Aprobación de la cursada sin tener que haber realizado ningún recuperatorio
- Promoción Directa: aprobación de 2 parciales con calificación ≥ 8 obtendrán la aprobación directa de la asignatura. El segundo parcial tiene carácter de integrador.

Examen final:

- En el caso de no promocionar la materia y haber aprobado la cursada se tomará un examen final presencial individual.

TPO



Trabajo práctico obligatorio

TPO – Régimen de Aprobación

Se evaluará el funcionamiento completo de los casos de uso solicitados en el TPO.

- Gestión de usuarios
- Catálogo de productos
- Carrito de compras
- Gestión de productos
- Mi Perfil.

Se evaluará el desempeño individual de cada uno de los participantes a partir de:

- **Commits y additions** del repositorio de código: Se tendrán en cuenta aquellos que se produzcan en las ramas principales (develop, master, main) a partir de la cantidad generada. Se tendrá en cuenta que las **aportaciones sean proporcionales** entre los miembros del grupo.

Se evaluará la presentación del TPO a través de un examen oral en el que se tendrá en cuenta tanto el resultado grupal como la aportación individual

Gracias!
