PRÁCTICO INTEGRADOR – DESARROLLO WEB

CÁCERES MÁXIMO GABRIEL

1.- Características de HTML 5:

**NUEVAS ETIQUETAS SEMÁNTICAS**

<header>: Encabezado de una página o sección.

<footer>: Pie de página o de sección.

<nav>: Navegación (por ejemplo un menú con enlaces internos o externos).

<article>: Contenido independiente y reutilizable (Entrada de blog, noticia, comentario).

<section>: Agrupar contenido relacionado dentro de una página.

<aside>: Información complementaria (Barras laterales, enlaces relacionados).

Mejoran la accesibilidad y el SEO al describir el propósito de cada bloque.

**MULTIMEDIA NATIVA**

* Etiquetas <audio> y <video> para reproducir contenido sin plugins externos.
* Soporte para subtítulos y descripciones (<track>)

**GRÁFICOS Y ANIMACIONES**

* <canvas> para dibujar gráficos y animaciones por código.
* Integración directa de SVG (Gráficos vectoriales escalables).

**FORMULARIOS MEJORADOS**

* Nuevos tipos de entrada: email, url, date, color, range, etc.
* Validación nativa con atributos como required, pattern, min, max.
* Placeholders y autocompletado mejorados.

**CONTENIDO MATEMÁTICO**

* Soporte para MathML para mostrar fórmulas matemáticas.

**ADAPTABILIDAD Y COMPATIBILIDAD**

* Diseño **responsive** (adaptación al diferentes tamaños de pantalla) más sencillo.
* Mejor soporte en navegadores modernos.
* Optimización para dispositivos móviles.

**APIs y FUNCIONALIDADES EXTRA**

* Geolocalización ( Gelocation API ).
* Almacenamiento local ( localStorage, sessionStorage )
* Web Workers para procesos en segundo plano.
* Offline con Application Cache (luego reemplazado por Service Workers)

**2.-** Próximas novedades de HTML, CSS y JS

|  |  |
| --- | --- |
| **HMTL** | * **Elementos semánticos extendidos**: se están discutiendo nuevas etiquetas y atributos para mejorar la accesibilidad y la interoperabilidad con lectores de pantalla. * **Mejoras en formularios**: validaciones más potentes y atributos como autocomplete y inputmode más refinados para distintos dispositivos. * **Integración con Web Components**: mayor soporte nativo para encapsulación y reutilización de componentes sin depender de frameworks externos. |
| **CSS** | Según Caronte Studio, 2025 trae varias propiedades nuevas y más soporte en navegadores.   * View(): Animaciones ligadas a la visibilidad del elemento en pantalla. * Auto-height: Transiciones suaves de altura sin “saltos” * Anchor positioning: Posicionar elementos relativos a otros de forma mas intuitiva. * Field-sizing: Ajustar automáticamente el tamaño de campos de texto. * Font-size-adjust: Mejor control de legibilidad entre distintas tipografías. |
| **JavaScript** | De acuerdo con NTT DATA, las tendencias clave incluyen:   * Importacion nativa de JSON: ahora se puede hacer import data from ‘./archivo.json’ sin hacks. * Mejoras en Set: métodos para uniones, intersecciones y diferencias directamente en el lenguaje. * Frameworks y librerías con IA: herramientas como ShadCN o V0 que generan interfaces casi completas a partir de descripciones. * APIs más modernas: mejoras en Web Animations, WebGPU y soporte más amplio para Streams y OffscreenCanvas. |

3.-Web Storage

Web Storage es una funcionalidad de HTML5 que permite a las aplicaciones web **guardar datos directamente en el navegador del usuario,** de forma más rápida, segura y con mayor capacidad que las cookies tradicionales.

Tipos de Almacenamiento:

* localStorage: Permanente (hasta que el usuario o el script lo borre), los datos persisten incluso si se cierra y vuelve a abrir el navegador. Ideal para preferencias o configuraciones.
* sessionStorage: Temporal (solo mientras la pestaña este abierta), los datos se eliminan al cerrar la pestaña o ventana. Útil para información de una sesión puntual.

4.- IFRAME

El elemento <iframe> (inline frame) en HTML sirve para incrustar otra página web (o un recurso HTML) dentro de la página actual. Básicamente, es como una "ventana" o "marco" que muestra contenido proveniente de otra URL.

Características clave:

**Contenido independiente:** lo que se carga dentro del <iframe> funciona de forma aislada respecto a la página padre.

**Fuente definida por src**: se indica con la URL del recurso.

**Aislamiento de estilos y scripts:** el CSS y JavaScript del documento padre no afectan al contenido del iframe (y viceversa), salvo que se configuren explícitamente.

**Uso frecuente:** insertar mapas de Google Maps, videos de YouTube, widgets externos, documentos PDF, etc.

Ejemplo básico de uso:

<iframe

src="https://www.example.com"

width="600"

height="400"

title="Ejemplo de iframe">

</iframe>