¿Qué es la accesibilidad web?

Es la práctica continua de asegurarnos que todo lo que creamos para la web se puede usar, interpretar y operar por una variedad de personas en una variedad de situaciones.

* Se usa la palabra “ALL7” para referirse a la accesibilidad ( buscar info sobre accesibilidad con esta palabra).
* Es estimado que alrededor del 20% de las personas del mundo vive con una discapacidad.
* Usamos la accesibilidad para incluir a las personas en situación de discapacidad, para mejorar la usabilidad de nuestros proyectos y porque en algunos lugares es por ley.
* En USA, se encuentra la Americans with Disabilities Act (ADA), que llevaron a la empresa de pizza DOMINO’S por no tener una web accesible para los usuarios.

¿Qué es WCAG?

Es inglés es “Web Content Accessibility Guidelines” o en español las “Pautas de Accesibilidad para el Contenido WEB”

* Es creado por la W3C: World Wide Web Consortium. Creadores de los estándares que usamos para construir la web.
* WAI – Web Accessibility Initiative. Iniciativa dentro de W3C cuya meta es asegurar la accesibilidad web.
* Dentro de WCAG hay 12 criterios de conformidad que debemos respetar y cada criterio se divide en 3 niveles.
* Nivel A es el más bajo, accesibilidad básica, Nivel AA, nivel intermedio, Nivel AAA, nivel avanzado al cual debemos apuntar si trabajamos en servicios del gobierno, universidades o servicios esenciales de la vida cotidiana.

Criterios de conformidad WCAG

## Perceptible

* Los criterios de conformidad bajo este principio se enfocan en asegurar que todo el contenido que es importante se pueda interpretar de varias maneras:
  + Hay que asegurar que estamos proporcionando alternativas textuales para todo contenido no textual:  
    1.2 Proporcionar alternativas para los medios tempodependientes (como videos o audios, queremos ofrecer opciones como subtítulos o guiones).
  + Crear contenido que se pueda presentar de diferentes formas sin perder información o estructura.
  + Facilitar a los usuarios ver y oír el contenido, incluyendo la separación entre el primer plano y el fondo

## Operable

* Cuando hablamos de ser operable, queremos asegurarnos que nuestro sitio es fácil de usar y navegar, que se puede navegar de diferentes maneras y con mouse y/o teclado
  + Proporcionar acceso a toda la funcionalidad mediante del teclado
  + Darle a las usuarias suficiente tiempo para leer y usar el contenido (tener esto en cuenta cuando hacemos time outs, comunicarlos y dar opciones para pedir más tiempo si es necesario)
  + Tener mucho cuidado de no usar elementos que brillan o se mueven muy rápido ya que pueden provocar ataques, espasmos o convulsiones
  + Asegurarnos que nuestros usuarios pueden navegar, encontrar contenido y determinar dónde se encuentran en nuestros sitios

## Comprensible

* Hacer que nuestro sitio web sea comprensible nos asegura que un rango de personas lo pueden usar desde la persona que vive con una discapacidad mental que necesita patrones comunes para saber cómo usar la web hasta la persona que va de carrera y necesita consultar algo rápido
  + Tener en cuenta los tamaños de texto y contraste de colores para que los textos resulten legibles y comprensibles
  + Hacer que las páginas web aparezcan y operen de manera predecible.
  + Dar instrucciones para evitar errores y oportunidades para corregirlos cuando ocurren

## Robusto

* Cuando hacemos productos que funcionan en muchos lados, abrimos las posibilidades que cualquier persona los use sin importar su ubicación, máquina, navegador y mucho más
  + Maximizar la compatibilidad con las aplicaciones de usuario actuales y futuras, incluyendo las tecnologías asistivas

Lighthouse

Nos ayuda a identificar y solucionar problemas en nuestro sitio, que van desde performance, accesibilidad y experiencia de usuario.

Desde la herramienta del desarrollador, buscamos el apartado Lighthouse, que viene incluido en los navegadores.

Navegación con TAB

Con el teclado siempre se deben enfocar los elementos interactivos.

* Un div es un elemento que no recibe foco
* Los inputs, button, a reciben foco del teclado.

HTML SEMÁNTICO

ES UNA DE LAS HERRAMIENTAS MÁS PODEROSAS QUE TENEMOS AL ALCANCE AL MOMENTO DE IMPLEMENTAR LA ACCESIBILIDAD NUESTRA WEB.

Es importante recordar que HTML es un lenguaje de marcado que nos permite, a través de sus etiquetas, organizar información en una página web para que sea legible.

**¿Qué es lo que difiere de aplicar semántica o no?**

**EL USUARIO NO VA A SABER QUE ESTAMOS USANDO HTML SEMÁNTICO, PERO EL NAVEGADOR SI.**

**Del lado del usuario, si hacemos uso de div para todo y utilizamos estilos y aplicamos dinamismo, para el usuario va a ser lo mismo, pero el navegador no va a saber nada de esto.**

**Cuando usamos HTML semántico, usamos una variedad de elementos que son creados con diferentes significados y que el navegador lee e implementa, la presentación y algunos detalles de las interacciones.**

**CUANDO UN SITIO ESTÁ GENERADO CON HTML SEMÁNTICO, SE GENERA ALGO LLAMADO “ÁBOL DE ACCESIBILIDAD”**

**El árbol de accesibilidad es lo que usan los lectores de pantalla para mapear sus interacciones y leerle el contenido de los usuarios.**

Al principio, los sitios webs se estructuraban con tablas. Al ser esto tan condicionante, aparecieron los elementos <div>, un contenedor genérico al cual se le da estilo. Lo que representa está relacionado a lo que contiene. A la par, se generó un lenguaje de apoyo, llamado CSS.

En el HTML cada elemento tiene su propio modelo de contenido, el cual describe el tipo de contenido que este debe contener y parámetros adicionales como el texto alternativo o alt para describir los contenidos de una imagen a los rastreadores.

Los modelos de contenido se dividen en las siguientes categorías

* Elementos para seccionar
* Elementos de metadata
* Interactivos
* Fraseo, de títulos
* Elementos para incrustar

Al dividir el contenido en secciones por temas o funcionalidad, logramos visualizar el esquema del documento.

Antes del lanzamiento de HTML5, la mejor manera para describir el esquema o índice del contenido, era utilizando los elementos de títulos que van del H1 – H6. Hoy en día siguen siendo muy importantes para identificar y generar secciones dentro del contenido.

Con el lanzamiento de HTML5, surgieron nuevos elementos enfocados en aumentar el poder semántico y estructural del lenguaje:

* Article
* Aside
* Nav
* Section
* Footer
* Header
* Main

Nuevos elementos como main o nav, que describen secciones de una página de internet y nuevos elementos como figure o time que describen el tipo de contenido que contienen.

Entender el significado semántico de estos nuevos elementos permite estructurar con detalle el esquema del documento. Lo que ayuda al lector a visualizar y entender mejor el contenido.

Además permite a personas con discapacidad. Navegar y consumir tu contenido con mayor facilidad. Además de que ayuda a los buscadores de Google rastrear y entender mejor el contenido de tu sitio de internet.

Con el surgimiento de los asistentes digitales y la búsqueda por voz, hoy en día la semántica en el código de tu página es fundamental.

NUEVOS ELEMENTOS ESTRUCTURALES

<header>

Este elemento se utiliza para agrupar las piezas de introducción del contenido.

Los elementos que encontramos dentro de la cabecera o introducción de una página de internet generalmente son: elementos de identidad o branding como el logotipo, nombre de la organización, slogan y elementos de navegación del sitio.

<nav>

Este elemento es para agrupar elementos de navegación del sitio, muy importante utilizarlo en el menú principal

<section>

<header>

<h1>Título de sección - noticia </h1>

<nav><a href=”#”>menu1</a><a href=”#”>menu2</a><a href=”#”>menu3</a></nav>

</header>

</section>

<main>

Utilizado para indicar el contenido más importante de la página, este nuevo elemento resulta muy importante para los lectores de pantalla y dispositivos de asistencia a discapacitados. Lo importante es utilizarlo para agrupar justo el contenido deseado y no agrupar aquí elementos que estén fuera del tema principal de la página, como los anuncios y las barras laterales.

<article>

La definicio de este elemento indica uqe se debe utilizar para agrupar contenido que por sí solo tiene razón de ser y significado. Un buen ejemplo es utilizarlo en un blog para agruar cada uno de los artículos o publicaciones.

Ejemplo:

<article>

<h2>título del artículo 1</h2>

<p>contenido del artículo 1</p>

</article>

<article>

<h2>título del artículo 2</h2>

<p>contenido del artículo 2</p>

</article>

<section>

Este elemento tiene un valor semántico más general y es utilizado para agrupar secciones o subsecciones de contenido. El significado de que es una sección se vuelve muy personal y subjetivo. Lo importante es tener consistencia en tu propia definición y uso del elemento.

<aside>

Dentro de este elemento va el contenido que está fuera del tema principal de la página web.

<footer>

El pie de página o footer, generalmente contiene los avisos legales, información de derechos de autor e información de contacto. Por lo general también es la última sección de la página.

Ejemplo:

<main>

<section>

<article>

<h2>título del artículo 1</h2>

<p>contenido del artículo 1</p>

</article>

<article>

<h2>título del artículo 2</h2>

<p>contenido del artículo 2</p>

</article>

</section>

<aside>

<p>barra lateral</p>

</aside>

</main>

<footer>

<p> aviso de privacidad </p>

</footer>

<figure>

Este element describe contenido de flujo con significado propio, muy utilizado para describir imágenes, video, ilustraciones, diagramas, fragmenos de tócivo, etc.

Por lo general este elemento viene acompañado de una leyenda con la descripción o información relacionada del elemento.

<figcaption>

Es el elemento que contiene la leyenda o información relevante acerca del elemento figure antecesor.

<figure>

<img src="imagen.jpg" alt="descripción" width="304" height="228">

<figcaption>Leyenda de imágen</figcaption>

</figure>

<blockquote></blockquote>

Crea citas en bloque, mara las citas a otros autores o documentos.

Está definido como elemento en bloque, es decir display: inline;

<cite>

Marca una referencia a una fuente o el autor de un texto citado.

<cite></cite

Es un elemento en línea

<p> <cite> Galileo </cite> dijo: “… y sin embargo, se mueve”. </p>

<q>: El elemento de cita en línea

El elemento HTML <q> indica que el texto adjunto es una cita corta en línea. La mayoría de los navegadores modernos implementan esto rodeando el texto entre comillas. Este elemento está destinado a citas breves que no requieren saltos de párrafo; para citas de bloque independiente, utiliza el elemento [<blockquote>](https://developer.mozilla.org/es/docs/Web/HTML/Element/blockquote).

<em> </em>

El elemento HTML <em> es el apropiado para marcar con énfasis las partes importantes de un texto. El elemento <em> puede ser anidado, con cada nivel de anidamiento indicando un mayor grado de énfasis.

<strong></strong>

marca énfasis en un texto, coloreando en negrita ese fragmento.

DIFERENCIAS ENTRE ARTICLE Y SECTION

El elemento <article> especifica un contenido independiente y autónomo. El elemento <section> define la sección en un documento.

### Elemento article

Puede considerarse una forma especializada o **específica** de <section>. Su finalidad es **almacenar contenido que podría estar solo**, fuera del contenido que le rodea, como el post de un blog.

Los tipos de contenido que encajan con <article>

* Video acompañado de una descripción.
* Artículos de noticias.
* Comentarios de un blog.

<article>

<h2>Elementos interesantes de HTML5</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer eu ex erat...</p>

</article>

<article>

<h2>Que función tiene DOCTYPE?</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer eu ex erat...</p>

</article>

### Elemento section

Es el**más genérico** de todos los elementos estrcuturales, pretende **agrupar contenido que está relacionado**. Sin embargo, esto no quiere decir que sea un contenedor genérico como un <div>. El contenido debe estar relacionado.

El elemento <section> podría usarse para contener los 2 elementos de <article> que hemos hecho.

<section>

    <article>

        <h2>Elementos interesantes de HTML5</h2>

        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit...</p>

    </article>

    <article>

        <h2>Que función tiene DOCTYPE?</h2>

        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit...</p>

    </article>

</section>

Pero eso no termina ahí, hay que tener en cuenta que el elemento <section> es apropiado si los elementos que contiene**son listados explícitamente** en el esquema del documento.

El esquema del documento se refiere al modelo seccionado de contenido en HTML5, donde cada nueva estructura crea su **propio esquema**.  Dicho esquema se genera con las cabeceras (h1-h6) contenidos en cada elemento.

Esto quiere decir que section **debe tener su propia cabecera** al igual que la tienen los elementos <article>.

<section>

   <h1>Artículos sobre HTML5</h1>

    <article>

        <h2>Elementos interesantes de HTML5</h2>

        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit...</p>

    </article>

    <article>

        <h2>Que función tiene DOCTYPE?</h2>

        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit...</p>

    </article>

</section>