

1. Usabilidad en la web

En plena era digital y de las TIC, el número de contenidos proporcionados a través de la web se ha visto aumentado de manera exponencial. Además, el usuario cada vez cuenta con más formación relativa al uso de las nuevas tecnologías, incluyendo el web. Es por ello por lo que las webs más recientes tienen una apariencia y unas funcionalidades que costaba imaginar hace apenas unos años.

La norma ISO 9241 (Ergonomics of human-system interaction), es un estándar internacional que trata sobre la ergonomía de la interacción humano-computadora (IHC), proporcionando pautas y principios para el diseño y la evaluación de sistemas interactivos, teniendo en cuenta la usabilidad y la experiencia del usuario. La serie ISO 9241 aborda diferentes aspectos de la interacción humano-computadora, estructurados en las diferentes partes de la norma.

La ergonomía de la interacción ordenador-persona aborda diversos aspectos para garantizar un entorno de trabajo digital adecuado y amigable.

- **Diseño de Interfaces de Usuario (UI):** Se centra en crear interfaces gráficas, menús, iconos y otros elementos visuales que sean intuitivos y fáciles de entender para los usuarios. La idea es minimizar la carga cognitiva y facilitar la navegación.
- **Diseño de Experiencia del Usuario (UX):** Considera la experiencia global del usuario al interactuar con el sistema informático. Busca hacer que la experiencia sea eficiente, teniendo en cuenta aspectos emocionales y psicológicos.
- **Disposición de Hardware y Mobiliario:** Incluye la disposición de teclados, ratones, monitores y otros dispositivos periféricos para minimizar la fatiga y prevenir problemas de salud.
- **ISO 9241-11:** Principios de usabilidad y diseño centrado en el usuario. Proporciona principios generales y directrices para el diseño centrado en el usuario. <https://www.iso.org/obp/ui/#iso:std:iso:9241:-11:ed-2:v1:en>
- **ISO 9241-110:** Diálogos de comandos. Se centra en el diseño de diálogos de comandos entre usuarios y sistemas. El **principio de diálogo Usuario-Sistema**, se centra a la comunicación fluida y efectiva entre el usuario y el sistema durante la interacción, siendo recomendable para su cumplimiento que se tengan en cuenta las siguientes recomendaciones: una retroalimentación inmediata, claridad en la presentación de la información, consistencia en la interfaz, entre otros.
- **ISO 9241-12:** Presentación de información. Aborda la presentación visual y organización de información en interfaces de usuario.

- **ISO 9241-171:** Directrices de accesibilidad para la ergonomía de la interacción. Proporciona directrices específicas para mejorar la accesibilidad en la interacción humano-computadora.

La familia de estándares ISO/IEC 25000, conocida como SQuaRE (System and Software Quality Requirements and Evaluation), es una familia de normas que tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software. Esta familia de normas ISO/IEC 25000 se encuentra compuesta por cinco divisiones.



Tabla 1. Comparativa Less y Sass. Fuente: <https://iso25000.com/>

El apartado ISO/IEC 25010, se centra en el modelo de calidad del software, este modelo es la piedra angular en torno a la cual se establece el sistema para la evaluación de la calidad del producto. En este modelo se determinan las características de calidad que se van a tener en cuenta a la hora de evaluar las propiedades de un producto software determinado.

La calidad del producto software se puede interpretar como el grado en que dicho producto satisface los requisitos de sus usuarios. El modelo de calidad del producto definido por la ISO/IEC 25010 se encuentra compuesto por las ocho características de calidad: adecuación funcional, eficiencia de desempeño, compatibilidad, usabilidad, fiabilidad, seguridad, mantenibilidad y potabilidad.



Tabla 1. Comparativa Less y Sass. Fuente: <https://iso25000.com/>

En concreto, sobre usabilidad se define que se trata de la capacidad del producto software para ser entendido, aprendido, usado y resultar atractivo al usuario, al ser usado bajo determinadas condiciones. Algunas de las subcaracterísticas directamente vinculadas a la usabilidad son: aprendicibilidad, operabilidad, protección contra errores de usuario o estética de la interfaz de usuario.

2. Accesibilidad

El concepto de *accesibilidad* está relacionado de manera directa con el de usabilidad. En este caso, la *accesibilidad* se puede definir como la posibilidad de acceso a un determinado portal web, en lugar de la facilidad de uso. Por tanto, es evidente que un determinado portal web debe ser accesible antes que usable.

Este hecho, en el fondo, quiere decir que el acceso al portal web debe facilitarse para todos los usuarios potenciales, más allá de las limitaciones técnicas de cada usuario (software, hardware, etc.) o de las limitaciones individuales de cada uno (discapacidades, dominio de un determinado idioma, etc.). De esta forma, un sitio web accesible debe tener en cuenta la gran diversidad de potenciales usuarios que puede llegar a tener.

2.1. El consorcio World Wide Web (W3C)

El consorcio World Wide Web (W3C) es una comunidad internacional donde las organizaciones miembro se encargan del desarrollo de estándares que aseguran el crecimiento y el acceso a la web. Fue creada en 1994 con un conjunto de objetivos que permitieran desarrollar tecno- logías interoperables.

Dentro de este marco se hace necesario desarrollar estrategias, directrices y recursos para garantizar el acceso por igual a la web, es así como aparece la Web Accessibility Initiative o Ini- ciativa para la accesibilidad a la web (WAI). Esta iniciativa desarrolló las Directrices de accesibi- lidad para el contenido web 2.0, más conocido como WCAG 2.0, donde se recogen las pautas y las técnicas que permitan ofrecer soluciones accesibles para el software y contenido web. Este conjunto de pautas fue aprobado bajo el estándar internacional ISO/IEC 40500:2012. En junio de 2018 fue sustituido por la versión WCAG 2.1.

Los cuatro principios que regulan este funcionamiento son que el diseño debe ser perceptible, operable, comprensible y robusto.

Cada uno de los principios que se analizarán a continuación, presentan un conjunto de criterios que serán representados con una notación “x.y.z”, donde cada uno de los elementos presenta un significado concreto. Bajo cada principio, los números y letras siguientes indican directrices y criterios de éxito específicos.

- **x: Principio** (1. Perceptible, 2. Operable, 3. Comprensible y 4. Robusto).
- **y: Directrices**
- **z: Criterios de Éxito** (Cada directriz tiene varios criterios de éxito que detallan condiciones específicas que deben cumplirse para satisfacer la directriz).

La numeración es jerárquica, proporcionando una estructura organizada para abordar diferentes aspectos de la accesibilidad web. La numeración de los criterios específicos dentro de las Pautas de Accesibilidad para el Contenido Web (WCAG) versión 2.1, resultan parte esencial del sistema de numeración utilizado para organizar y referenciar los criterios de accesibilidad en WCAG.

1. *Principio 1. Diseño perceptible.* La información y los componentes de la interfaz de usuario deben ser mostrados a los usuarios de manera que pueda ser entendida. Este principio se centra en garantizar que la información y las operaciones de usuario sean presentadas de manera que puedan ser percibidas. Para ello se establecen cuatro directrices:

- *Texto alternativo.* Proporcionar texto alternativo para el contenido que no sea textual (caracteres grandes, lenguaje braille, lenguaje oral, símbolos o lenguaje más simple).
- *Contenido multimedia dependiente del tiempo.*
- *Adaptable.* Crear contenido que pueda ser presentado de diferentes formas sin perder información o estructura.
- *Distinguible.* Facilitar a los usuarios la posibilidad de ver y escuchar el contenido incluyendo la distinción entre lo más y menos importante.

A continuación, se muestran diferentes casos prácticos basados en pautas y técnicas específicas para evaluar el principio de Diseño Perceptible.

1. **Contraste de Color (Criterio 1.4.3).** El texto (y las imágenes de texto) tiene un contraste de color de al menos 4.5:1 con respecto a su fondo, excepto para el texto grande, el texto en imagen y el texto incrustado. Es decir, debe existir contraste suficiente entre el texto y el fondo para que sea legible. Para comprobar que este criterio se cumple, así como otros, se utilizarán herramientas de verificación de contraste de color para evaluarlo. En el ejemplo, el color gris oscuro de la línea de texto ofrece mejor contraste de color que la de título, con respecto al color de fondo (eso se comprobará después con las herramientas de evaluación de accesibilidad).

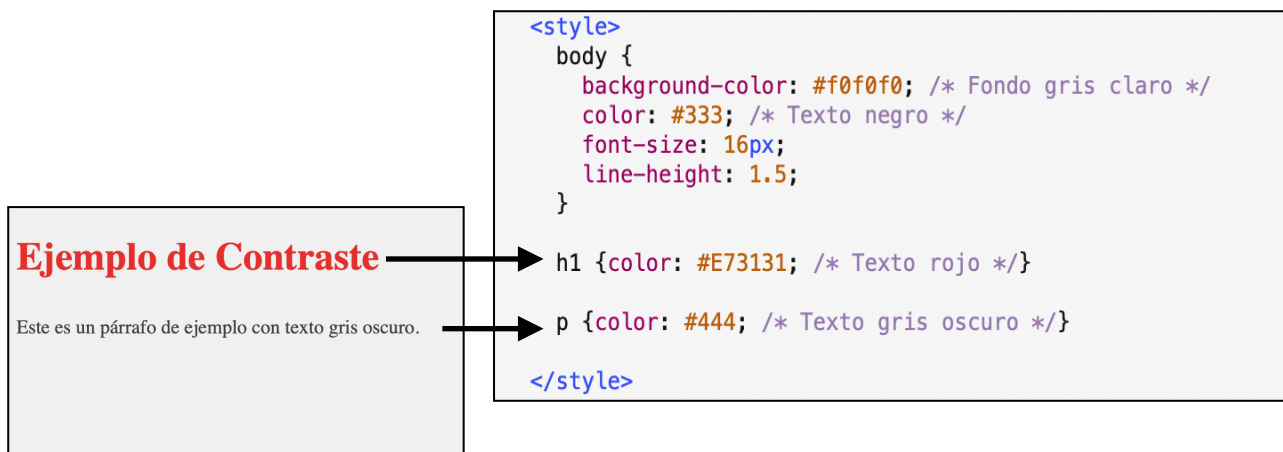


Figura 1. Ejemplo criterio de contraste

2. **Adaptabilidad del Texto (Criterio 1.4.4).** Permite a los usuarios ajustar el tamaño del texto sin perder información o funcionalidad, esto está directamente vinculado con la responsividad del contenido. Esto se puede realizar utilizando la regla de medios **@media** para aplicar estilos adicionales cuando el ancho de la pantalla es igual o inferior a 600px. Esto nos va a permitir ajustar el tamaño del texto en dispositivos más pequeños.
3. **Multimedia Alternativo (Criterio 1.2.1 y 1.2.2).** Se deberán proporcionar alternativas textuales para contenido no textual, como imágenes y multimedia, para esto se puede utilizar la propiedad **alt**, seguida de la descripción textual, para la imagen o el vídeo.
4. **Adaptabilidad del Contenido (Criterio 1.4.10).** El contenido deberá ser adaptable y podrá presentarse de diferentes maneras, como en dispositivos móviles, sin perder información o funcionalidad. Este criterio también queda relacionado con la regla de medios **@media**, que permite que el contenido sea adaptable a dispositivos móviles.
5. **Contrastar y Presentar (Criterio 1.4.11).** Se debe proporcionar alguna manera de pausar, detener, ocultar o ajustar la velocidad de cualquier contenido que se mueva, parpadee o destelle automáticamente durante más de cinco segundos. Por ejemplo, si un elemento de animación con CSS está desplazándose por la pantalla se deberá incorporar algún método para detenerlo, por ejemplo un botón.



Figura 1. Ejemplo criterio de contraste

- 6. Reconocimiento de Componentes (Criterio 1.3.1).** Los elementos interactivos, como los formularios, deben tener etiquetas y asociaciones adecuadas para que los usuarios puedan entender y completarlos. Por ejemplo, en este ejemplo campo del formulario está etiquetado con la etiqueta <label> que se asocia al campo correspondiente mediante el atributo for y el atributo id, esto permite el usuario pulse sobre Correo Electrónico y se marque el campo que se debe completar.

Un formulario web con tres campos de entrada y un botón. El primer campo está etiquetado como 'Nombre:' y es un cuadro de texto vacío. El segundo campo está etiquetado como 'Correo Electrónico:' y es un cuadro de texto con un cursor de texto y un ícono de lupa a la derecha. El tercer campo está etiquetado como 'Mensaje:' y es un cuadro de texto más pequeño. A la derecha del campo 'Mensaje' hay un botón azul con el texto 'Enviar Mensaje'.

Figura 1. Ejemplo criterio de contraste

- 7. Contrastar sin Tecnología (Criterio 1.4.12).** Este criterio persigue que la información sea presentada sin depender exclusivamente de la percepción del color, para ello se debe implementar el sitio para ser comprensibles sin depender exclusivamente del color, por ejemplo, utilizando estilos, texto y estructura para garantizar que la información sea accesible para todos los usuarios, independientemente de su percepción del color
- 8. Sentido de la Información (Criterio 1.3.3).** Se utilizarán técnicas que ayuden a los usuarios a comprender o interpretar la información, por ejemplo, utilizando encabezados (h1 y h2).

2. *Principio 2. Diseño operable.* Los componentes de la interfaz de usuario y la navegación deben ser manejables. Este principio se centra en asegurar que los usuarios puedan interactuar y operar con la interfaz y el contenido de manera efectiva. Para ello se establecen cuatro directrices:

- *Teclado accesible.* Poder controlar todas las funciones desde el teclado.
- *Tiempo suficiente.* Proporciona tiempo suficiente a los usuarios para leer y utilizar el contenido.
- *Ataques epilépticos.* No diseñar contenido que pueda causar ataques epilépticos.
- *Navegación.* Proporciona formas para ayudar a los usuarios a navegar, a buscar contenido y a determinar donde están estos.

Se muestran diferentes casos prácticos basados en pautas y técnicas específicas para evaluar el principio de Diseño Operable.

- 1. Teclado (Criterio 2.1.1).** Todas las funciones importantes y las áreas interactivas del sitio web deberán resultar accesibles y utilizables utilizando a través del teclado, para ello es conveniente

realizar pruebas de navegación sin el uso del ratón, por ejemplo, si se tiene un botón este debe ser accesible y se podrá utilizar tanto con el ratón como con el teclado, para ello se la escucha de eventos asociados a la tecla Enter.

2. **Tiempo Suficiente (Criterio 2.2.1).** Los usuarios deberán tener suficiente tiempo para leer y utilizar el contenido, por lo tanto, se debe evitar que el contenido cambie o se actualice automáticamente añadiendo alguna opción para pausar, detener o ajustar el tiempo. Se pueden utilizar algunas funciones Javascript como setInterval.
3. **Parpadeo y Destello (Criterio 2.3.1).** Es necesario incorporar las herramientas necesarias para evitar el uso de elementos que parpadeen o destellen automáticamente, puesto que podría causar convulsiones, es conveniente proporcionar una opción para desactivar o ajustar el parpadeo, por ejemplo, con Javascript.

```
<script>
  var parpadeo = document.getElementById('parpadeo');

  function detenerParpadeo() {
    parpadeo.style.animation = 'none';
  }
</script>
```

Figura 1. Ejemplo criterio de contraste

4. **Navegación Intuitiva (Criterio 2.4.2).** La estructura del sitio web debe ser intuitiva y predecible, para esto se recomienda el uso de encabezados, etiquetas y enlaces.
5. **Distracciones (Criterio 2.2.2 y 2.3.2).** Se debe minimizar la existencia de distracciones en el sitio web, por ejemplo, evitando el uso de contenido que parpadee, destelle o cambie automáticamente sin la capacidad de controlar o pausar.
6. **Navegación por Búsqueda (Criterio 2.4.5):** Proporciona una funcionalidad de búsqueda para ayudar a los usuarios a encontrar contenido específico de manera eficiente. Se propone incluir un pequeño formulario con un campo de entrada de texto y un botón de búsqueda, este proporciona una funcionalidad de búsqueda que puede ayudar a los usuarios a encontrar contenido específico en la página.
7. **Navegación Consistente (Criterio 3.2.3).** El sitio web deberá presentar consistencia en la navegación y el diseño, para que de esta forma los usuarios puedan prever el comportamiento de la interfaz. Por ejemplo, en un sitio web cualquiera la consistencia se logra a través del uso de estilos similares para el encabezado y la barra de navegación, así como la colocación de los elementos comunes en lugares predecibles.

3. *Principio 3. Diseño comprensible.* La información y las operaciones de usuarios deben ser comprensibles. Para ello se establecen tres directrices:

- *Legible*. Hacer contenido de texto legible y comprensible.
- *Previsible*. Presentar una apariencia y una forma de utilizar las páginas web previsibles.
- *Asistencia a la entrada de datos*. Ayudar a los usuarios a evitar y corregir errores.

1. **Lenguaje Claro (Criterio 3.1.2)**. Se debe utilizar un lenguaje claro en todo el contenido del sitio web, evitando texto innecesario o expresiones complejas. La información debe resultar fácil de entender para una amplia audiencia.
2. **Predecibilidad (Criterio 3.2.1)**. El diseño de los elementos contenidos en un sitio web deben conseguir que la navegación y la interacción resulten predecibles, es decir, los usuarios tendrían que ser capaces de anticipar cómo funcionarán todas las partes del sitio.
3. **Entrada de Datos (Criterio 3.3.1)**. Es conveniente proporcionar sugerencias y ayudas claras, por ejemplo con ayuda para que los usuarios puedan completar los campos de entrada, incluyendo sugerencias sobre el formato correcto de la información, instrucciones sobre qué tipo de datos se espera en un campo específico, indicando si es necesario el uso de un formato específico, entre otros.
4. **Identificación de Errores (Criterio 3.3.3)**. Los usuarios deben ser informados de manera clara y comprensible sobre cualquier error en formularios u otro tipo de interacciones, de esta forma si los usuarios cometen errores al completar formularios o al interactuar con la interfaz, se indicará cuál es el motivo.

Nombre:
Diana

Introduce una dirección de correo electrónico

Correo Electrónico:
correoprueba@gmail.com

Enviar

Figura 1. Ejemplo criterio de contraste

5. **Consistencia (Criterio 3.2.2)**. Se debe mantener la consistencia en el diseño del sitio web, sobre todo en la presentación del contenido para permitir que los usuarios puedan comprender y prever cómo funcionará la interfaz.
6. **Ayuda Contextual (Criterio 3.3.2)**. Como se ha expuesto en criterios previos, es conveniente proporcionar ayuda contextual para facilitar a los usuarios la comprensión de la información o de la funcionalidad específica.
7. **Entrada de Usuario (Criterio 3.3.4)**. Los usuarios deberán poder corregir o deshacer acciones involuntarias y tener información sobre cómo hacerlo. Por ejemplo, ante la entrada incorrecta de

datos en un formulario, será conveniente incluir un script para impedir que el formulario se envíe en el caso de tener datos no válidos, así como para mostrar una alerta al usuario, lo que permitirá a los usuarios corregir o deshacer la acción de enviar el formulario si lo hacen involuntariamente.

4. *Principio 4. Diseño robusto.* El contenido ha de ser robusto para que pueda ser interpretado por una gran variedad de agentes de usuario, incluyendo las tecnologías de asistencia. Para ello se establece una directriz.

- *Compatible.* Maximizar la compatibilidad con los agentes de usuario actuales y futuros, incluyendo las tecnologías de asistencia.

El principio de "Diseño Robusto" es otro de los cuatro principios fundamentales de las Pautas de Accesibilidad para el Contenido Web (WCAG).

1. **Compatibilidad con Tecnologías de Asistencia (Criterio 4.1.2).** El contenido deberá ser compatible con tecnologías de asistencia como lectores de pantalla y navegadores de voz. Por ejemplo, mediante el uso del atributo **aria-label** sobre un botón para proporcionar un texto descriptivo adicional para usuarios de tecnologías de asistencia, como lectores de pantalla.
2. **Compatibilidad con Navegadores (Criterio 4.1.1).** El sitio web deberá ser, en la medida de lo posible, compatible con una amplia variedad de navegadores
3. **Detección de Cambios (Criterio 4.1.3).** Los cambios en el contenido y el estado de la interfaz de usuario han de ser detectables y notificados al usuario, esto resulta clave sobre para los casos en los que se utilizan tecnologías de asistencia. Por ejemplo, utilizando **aria-live="polite"** para notificar a las tecnologías de asistencia sobre el cambio.

Tal y como se recoge en la recomendación del W3C del 11 de diciembre de 2008 para las Pautas de Accesibilidad para el Contenido Web (WCAG) 2.0, para que una página web sea conforme con las WCAG 2.0 deben satisfacerse todos los requisitos de conformidad siguientes:

1. *Nivel de conformidad.* Se recogen un conjunto de criterios de conformidad, redactados en forma de enunciados verificables sobre el contenido web, y que son utilizados para verificar la adecuación a la accesibilidad de un sitio web. Se distinguen tres niveles de conformidad que definen el grado de accesibilidad; uno de ellos se debe satisfacer por completo.
 - *Nivel A:* para lograr conformidad con el nivel A (el mínimo), la página web satisface todos los criterios de conformidad del nivel A, o proporciona una versión alternativa conforme. Se deben satisfacer 25 criterios.

- *Nivel AA*: para lograr conformidad con el nivel AA, la página web satisface todos los criterios de conformidad de los niveles A y AA, o se proporciona una versión alternativa conforme al nivel AA. Se deben satisfacer 13 criterios.
 - *Nivel AAA*: para lograr conformidad con el nivel AAA, la página web satisface todos los criterios de conformidad de los niveles A, AA y AAA, o proporciona una versión alternativa conforme al nivel AAA. Se deben satisfacer 23 criterios.
2. *Páginas completas*. La conformidad se aplica a páginas web completas, y no se puede alcanzar si se excluye una parte de la página.
 3. *Procesos completos*. Cuando una página web es parte de una serie de páginas web que presentan un proceso, todas las páginas en ese proceso deben ser conformes con el nivel especificado o uno superior.
 4. *Uso de tecnologías exclusivamente según métodos que sean compatibles con la accesibilidad*. Para satisfacer los criterios de conformidad solo se depende de aquellos usos de las tecnologías que sean compatibles con la accesibilidad.
 5. *Sin interferencia*. Si las tecnologías se usan de una forma que no es compatible con la accesibilidad, o están usadas de una forma que no cumplen los requisitos de conformidad, no deben impedir a los usuarios acceder al contenido del resto de la página.

2.2. Normativa para accesibilidad en aplicaciones web y dispositivos móviles. RD 1112/2018 y UNE 301549.

El Real Decreto 1112/2018 es un marco legal en España que fija y establece las normas de accesibilidad y usabilidad de los sitios web y aplicaciones móviles del sector público. En la actualidad el desarrollo de sitios web y aplicaciones móviles para la Administración Pública está cada vez más extendido, en aras de una Transformación Digital esencial para creación de nuevos activos estratégicos.

Esta normativa resulta clave para todo desarrollo que aplica a los sitios web y aplicaciones móviles de los organismos del sector público, incluidos organismos gubernamentales, administraciones locales y otras entidades vinculadas al sector público.

Se establecen principios de accesibilidad universal que los sitios web y aplicaciones móviles deben cumplir para garantizar el acceso a la información y servicios para todas las personas, independientemente de sus capacidades y discapacidades. En el Real Decreto se establece tres niveles de conformidad con las normas de accesibilidad, identificados como A, AA y AAA, siguiendo las Pautas de Accesibilidad al Contenido en la Web (WCAG) 2.1 del W3C.

Asimismo la UNE 301549:2019 es una norma técnica española que establece los requisitos de accesibilidad y usabilidad para contenidos web, software y dispositivos móviles, en concreto, sobre los contenidos en servicios públicos electrónicos, encontrándose plenamente alineada con las Directrices de Accesibilidad para Contenidos Web (WCAG) del World Wide Web Consortium (W3C). Esta norma

aborda aspectos como la percepción visual, la audición, la movilidad, la comprensión y la operabilidad.

2.3. Atributo aria (Accesible Rich Internet Application)

Este atributo aglutina un conjunto de atributos en HTML que son utilizados para mejorar la accesibilidad de las aplicaciones web, especialmente aquellas que utilizan tecnologías e interacción, como JavaScript. Estos atributos resultan especialmente útiles para proporcionar información a tecnologías de asistencia, como los lectores de pantalla, para que estos se puedan comunicar de manera efectiva la interfaz de usuario a las personas con discapacidades. Se describen a continuación, algunos de los atributos aria más comunes en la actualidad.

Toma nota:

Para ampliar información sobre aria puedes consultar: <https://www.w3.org/TR/wai-aria/>

- **aria-label.** Añade una etiqueta de texto que describe la función de un elemento.
`<button aria-label="Cerrar ventana" onclick="cerrarVentana()">Cerrar</button>`
- **aria-labelledby.** Asocia el elemento con un elemento de la página que actúa como su etiqueta.
`<h2 id="titulo-ventana">Ventana de Configuración</h2>
<div role="dialog" aria-labelledby="titulo-ventana">
 <!-- Contenido de la ventana de configuración -->
</div>`
- **aria-describedby.** Asocia el elemento con un elemento de la página que proporciona una descripción más detallada.
`<input type="text" aria-describedby="instrucciones-input">
<p id="instrucciones-input">Ingrese su nombre completo</p>`
- **aria-hidden.** Permite determinar si el elemento y sus descendientes serán visibles o estarán ocultos para las tecnologías de asistencia (siendo true oculto)
`<div aria-hidden="true"> Contenido no visible para las tecnologías de asistencia</div>`
- **aria-live.** Se utiliza para posibilitar que las actualizaciones del contenido sean anunciadas de inmediato a los usuarios o deben esperar hasta que el usuario solicite una actualización. Este atributo puede tener varios valores para indicar cómo se deben manejar las actualizaciones del contenido en tiempo real: **off** (para desactivar que se notifiquen de forma automática las

actualizaciones), **assertive** (para activar que las actualizaciones del contenido se realicen de forma inmediata, interrumpiendo cualquier otro contenido que se esté reproduciendo) y **polite** (para activar que las actualizaciones del contenido se realicen después de que el usuario ha completado la acción actual).

```
<div aria-live="assertive">¡Nuevo mensaje recibido!</div>
```

2.4. Herramientas para la evaluación de accesibilidad

- Lighthouse. Herramienta de auditoría de código abierta desarrollada por Google, ofrece informes detallados sobre el rendimiento, la accesibilidad, las mejores prácticas, el SEO, entre otras. (<https://developers.google.com/web/tools/lighthouse>). Resulta sencillo de utilizar y está disponible desde las herramientas para desarrolladores de Chrome, bastará con seleccionar la opción Lighthouse y cargar el sitio web que sea desea auditar. Solo puede auditar páginas en http o https.
- WAVE (Web Accessibility Evaluation Tool). Herramienta en línea que permite evaluar la accesibilidad de un sitio web, devolviendo informes detallados en el que se indican problemas específicos. (<https://wave.webaim.org/>).
- Validador HTML de W3C. Realiza una validación de la gramática del sitio web, comprobando la conformidad de los documentos HTML con respecto a las gramáticas del W3C. Permite tanto la inspección de ficheros en local como a través de URL.
- Validador de CSS de W3C. Realiza una validación de la gramática centrada en la verificación de las hojas de estilo CSS y otros documentos HTML con hojas de estilo.
- WCAG Contrast Checker. Esta herramienta permite verificar el contraste de color entre dos colores, dando así cumplimiento a las pautas de contraste de color de WCAG. (<https://contrastchecker.com/>)
- Axe DevTools (Extensión para Chrome y Firefox). Una extensión que se integra directamente en las herramientas de desarrollo del navegador para realizar auditorías de accesibilidad en tiempo real. (<https://www.deque.com/axe/devtools/>)
- Color Contrast Analyzer. Se trata de una herramienta de escritorio que permite analizar el contraste de color directamente en la interfaz de usuario. Permite seleccionar los colores para ser evaluados e incluye un resumen de cumplimiento de criterios WCAG.
- Contrast-ratio. Se trata de una herramienta en línea utilizada para evaluar el contraste entre dos colores según la fórmula establecida por el Web Content Accessibility Guidelines (WCAG). El contraste entre colores es un factor crucial para garantizar la accesibilidad web, especialmente

para personas con discapacidades visuales.