



Escola Politècnica Superior
d'Enginyeria de Vilanova i la Geltrú

UNIVERSITAT POLITÈCNICA DE CATALUNYA

PROYECTO FINAL DE CARRERA

TÍTULO: Accesibilidad de la Web e Intranet de la EPSEVG

AUTORES: Laura Ortiz Hernández
Rafael de la Rosa Iglesias

TITULACIÓN: Ingeniería Técnica de Telecomunicaciones, Esp. en Sistemas Electrónicos.

DIRECTOR: Víctor Agramunt Ciurana

PONENTE: Rafael Morillas Varón

DEPARTAMENTO: Ingeniería Telemática

FECHA: 26-06-2009

TÍTULO: Accesibilidad de la Web e Intranet de la EPSEVG

APELLIDOS: Ortiz Hernández

NOMBRE: Laura

TITULACIÓN: Ingeniería Técnica de Telecomunicaciones

ESPECIALIDAD: Sistemas electrónicos

PLAN: 1995

APELLIDOS: de la Rosa Iglesias

NOMBRE: Rafael

TITULACIÓN: Ingeniería Técnica de Telecomunicaciones

ESPECIALIDAD: Sistemas electrónicos

PLAN: 1995

DIRECTOR: Víctor Agramunt Ciurana

PONENTE: Rafael Morillas Varón

DEPARTAMENT: Ingeniería Telemática

CALIFICACIÓN DEL PFC

TRIBUNAL

PRESIDENTE

SECRETARIO

VOCAL

FECHA DE LECTURA:

Este Proyecto tiene en cuenta aspectos medioambientales: Sí No

PROYECTO FINAL DE CARRERA

RESUMEN (máximo 50 líneas)

En este proyecto se ha adaptado la nueva web de la EPSEVG a la nueva ley española publicada en el **BOE número 312 de 29/12/2007** que obliga a los organismos públicos a cumplir como mínimo con el segundo nivel de accesibilidad web.

La accesibilidad web consiste en eliminar las barreras que dificultan o imposibilitan el acceso a la web a personas con discapacidad visual, auditiva o cognitiva/neuronal, entre otras. Gracias al presente proyecto, se han eliminado dichas barreras, consiguiendo que la web de la EPSEVG ya no sea únicamente accesible para personas sin ningún tipo de discapacidad.

Para conseguir nuestro fin, se han seguido unas pautas de diseño creadas por la WAI, una actividad desarrollada por el W3C, consorcio industrial internacional cuyo objetivo es facilitar el acceso a Internet a las personas con discapacidad. Dichas pautas, "Pautas de Accesibilidad al Contenido en la Web 1.0", están divididas en tres niveles de accesibilidad. Como se ha indicado anteriormente, la ley española obliga a cumplir como mínimo el segundo nivel. En el presente proyecto no sólo se ha querido seguir la normativa sino que se ha intentado cumplir con el tercer nivel. No siempre se va a poder conseguir que una web sea accesible al 100% ya que depende de muchos factores tales como el tamaño del sitio, complejidad, hardware, herramientas utilizadas y diseño de la página.

Algunas de las medidas tomadas para cumplir con los mencionados niveles son:

- Garantizar un gran contraste de colores, tipos de letra y estilos apropiados a personas con dificultades visuales, así como evitar los parpadeos, para personas con epilepsia.
- Asegurar que toda la información transmitida a través de colores, esté también sin color y garantizar que personas con ceguera, que utilizan dispositivos de voz tengan acceso al contenido web.
- Proporcionar textos alternativos en imágenes o en todos aquellos elementos no textuales.
- Maquetar las páginas con hojas de estilo CSS y organizar el documento para que también pueda ser leído sin la utilización de hojas de estilo.

Para garantizar la accesibilidad del sitio web se ha testado la página utilizando herramientas como **TAWDIS**, una aplicación web que indica los errores técnicos y manuales del sitio. También se ha utilizado software especial por voz, el programa **JAWS**, para corroborar la lectura correcta de la página.

Todas estas medidas se han adaptado al diseño inicial de la web, para que sea una página atractiva y agradable de consultar.

Palabras clave (máximo 10):

| | | | |
|-------------------|----------------------|--------------|------------|
| Accesibilidad Web | CSS | Discapacidad | Validación |
| HTML | T.A.W | W3C | Estilo |
| Legislación | Pautas accesibilidad | | |

Índice

| | |
|--|----|
| Índice..... | 11 |
| Agradecimientos | 15 |
| 1 Introducción..... | 17 |
| 2 Declaración de intenciones..... | 21 |
| 3 Objetivos | 23 |
| 3.1 Accesibilidad web | 23 |
| 3.2 Estudiar HTML y PHP | 23 |
| 3.3 Estudiar hojas de estilo CSS | 23 |
| 3.4 Herramientas necesarias en la validación | 24 |
| 3.5 Pautas a seguir para la accesibilidad web | 24 |
| 3.6 Conocimiento del programa JAWS | 24 |
| 3.7 Analizar resultados y extraer conclusiones | 24 |
| 4 Ámbito de aplicación | 25 |
| 4.1 Introducción..... | 25 |
| 4.2 ¿Qué es la Accesibilidad Web? | 25 |
| 4.3 Pautas, prioridades y Legislación..... | 26 |
| 4.4 Beneficios..... | 27 |
| 4.5 Limitaciones | 28 |
| 4.6 Estado actual..... | 28 |
| 5 Estado del arte | 29 |
| 5.1 Introducción..... | 29 |
| 5.2 Accesibilidad web | 29 |
| 5.2.1 Accesibilidad web..... | 29 |
| 5.2.2 Legislación | 29 |
| 5.2.3 Estado actual | 30 |
| 5.3 W3C i WAI | 30 |
| 5.4 Estándares | 32 |
| 5.5 Tecnologías web | 32 |
| 5.5.1 Introducción al código abierto | 32 |
| 5.5.2 Tipos de páginas web | 33 |
| 5.5.3 HTML | 34 |
| 5.5.4 PHP | 34 |

| | | |
|-------|---|----|
| 5.5.5 | Hojas de estilo o CSS | 34 |
| 5.6 | Servidor de pruebas Kato | 35 |
| 5.7 | Herramientas utilizadas | 36 |
| 5.7.1 | Dreamweaver | 36 |
| 5.7.2 | Internet Explorer y Mozilla Firefox..... | 37 |
| 5.8 | Software de Validación | 39 |
| 5.8.1 | Validador de CSS de W3C..... | 39 |
| 5.8.2 | TAW | 40 |
| 5.8.3 | CONTRAST-A..... | 42 |
| 5.8.4 | JAWS..... | 43 |
| 6 | Diseño. | 45 |
| 6.1 | Introducción..... | 45 |
| 6.2 | Discapacidades a las que afecta la accesibilidad web | 45 |
| 6.3 | En qué consiste cada discapacidad y cambios a realizar en la Web..... | 46 |
| 6.3.1 | Discapacidades visuales | 46 |
| 6.3.2 | Discapacidades auditivas..... | 48 |
| 6.3.3 | Discapacidades físicas | 49 |
| 6.3.4 | Discapacidades cognitivas y neurológicas..... | 50 |
| 6.3.5 | Falta de memoria | 51 |
| 6.3.6 | Discapacidades múltiples | 52 |
| 6.3.7 | Condiciones relacionadas con el envejecimiento. | 53 |
| 6.4 | Medidas tomadas en la eliminación de las barreras..... | 53 |
| 6.4.1 | Medidas generales | 53 |
| 6.5 | Tablas..... | 58 |
| 6.5.1 | Prioridades | 59 |
| • | Prioridad 1 | 59 |
| • | Prioridad 2 | 59 |
| • | Prioridad 3 | 59 |
| 6.5.2 | Creación de tablas accesibles..... | 60 |
| 6.6 | Formularios..... | 64 |
| • | Prioridad 2 | 64 |
| • | Prioridad 3 | 65 |
| 6.6.1 | Creación de formularios accesibles..... | 65 |
| 6.7 | WCAG 2.0..... | 71 |

| | |
|--|-----|
| 6.8 Cambios en el código PHP | 74 |
| 6.9 Página Accesibilidad | 77 |
| 7 Resultados obtenidos. | 79 |
| 7.1 Introducción..... | 79 |
| 7.2 Cambios visibles realizados en la web..... | 79 |
| 7.3 Intranet. | 85 |
| 7.4 Validaciones | 87 |
| 8 Futuras vías de trabajo | 91 |
| 9 Bibliografía | 93 |
| Anexo 1 - Manual de Tablas Accesibles | 95 |
| Anexo 2 - Manual de Formularios Accesibles | 97 |
| Anexo 3 - Abreviaturas..... | 101 |
| Anexo 4 - Atajos de Teclado..... | 105 |
| Anexo 5 - Puntos de verificación WCAG 1.0 | 107 |
| Anexo 6. Correspondencia entre la norma UNE 139803:2004 y los puntos de verificación WAI | 113 |
| Anexo 7. Árbol de decisiones..... | 115 |



Agradecimientos

Queremos agradecer a todas las personas que nos han apoyado a lo largo de la realización del proyecto. A nuestros amigos, que siempre han estado a nuestro lado, a los compañeros de trabajo y a todas las personas que han colaborado en la validación de la web.

Agradecimientos especiales a nuestro tutor del proyecto, por ofrecernos su ayuda y estar presente en todo momento.

1 Introducció

El presente proyecto se centra en hacer accesibles la nueva web e Intranet de la EPSEVG que podrán ser visitadas en breve.

La accesibilidad web consiste en que personas discapacitadas tengan acceso a Internet, disminuyendo las barreras de diseño que lo dificultan y así, hacer que puedan acceder a la información.

Debido al desarrollo tecnológico y de la información, la web de la EPSEVG necesitaba cambiar el lenguaje de programación. La web anterior estaba programada en código ASP, por eso, se creó una nueva web en PHP, lenguaje de programación especialmente diseñado para el desarrollo de páginas web dinámicas, consiguiendo una página más dinámica y con más información.

La nueva página web ha sido realizada en base a Joomla!, sistema de administración de contenidos de código abierto construido con PHP. El lenguaje de marcado utilizado para construir la página web ha sido HTML. Pero en este proyecto no se ha realizado la web, sino, su accesibilidad.

Con el fin de conseguir una página más accesible se ha tenido que modificar el código PHP de la nueva web. Por este motivo, antes de realizar el presente proyecto se ha tenido que estudiar y obtener los conocimientos necesarios de PHP y HTML para poder modificar el código.

La principal medida a tomar para hacer accesible una web es separar la estructura o el contenido de la presentación. Por este motivo se utilizan las hojas de estilo para maquetar. Estas contienen todo el estilo que se aplica a la web. Se utilizan para definir la presentación de un documento estructurado y escrito en HTML. Se ha tenido que modificar la mayoría de los archivos CSS existentes para adaptarlos al nuevo diseño, lo que implica una preparación previa: estudio y adquisición de conocimientos sobre CSS.

Una vez adquiridos todos los conocimientos previos necesarios -que debido a que somos estudiantes de Ingeniería Técnica de Telecomunicaciones especializada en Sistemas Electrónicos carecíamos de ellos-, se empezó con el desarrollo del proyecto, haciendo los cambios oportunos en la web. Todos estos cambios han requerido de la utilización nuevas herramientas cuyos conocimientos se han ido asimilando a medida que se iba avanzando con el proyecto.

Cada uno de los pasos realizados a lo largo del proyecto, como pueden ser cambios en el CSS o en el código, han requerido de validaciones previas, que consisten en cumplir con las pautas de accesibilidad web creadas por el W3C¹, consorcio internacional formado por más de 300 asociaciones, cuyo objetivo es facilitar el acceso a Internet a las personas con discapacidad. No se ha realizado ningún cambio que no cumpla con estas pautas.

¹ World Wide Web Consortium

Las "Pautas de Accesibilidad al Contenido en la Web 1.0" (WCAG 1.0) consisten en 14 pautas con sus respectivos puntos de verificación, divididas en tres prioridades. No son legislación pero son el modelo a seguir para realizar un sitio web accesible.

En diciembre de 2008 entró en vigencia una nueva ley que obliga a los organismos públicos españoles a cumplir como mínimo con la segunda prioridad. Como ninguna ley española puede basarse en unas pautas que no hayan sido creadas por un organismo oficial de normalización, las WCAG 1.0 se tuvieron que reescribir en la Norma UNE 139803:2004.

El 11 de diciembre de 2008 el W3C aprobó las "Pautas de Accesibilidad al Contenido en la Web 2.0" (WCAG 2.0), una mejora de las anteriores, que consiste en 12 pautas. Actualmente no se han incorporado a la legislación española, pero es conveniente tenerlas en cuenta.

Con este proyecto se ha querido cumplir con la normativa, pero no solo con eso, sino, con poder hacer la nueva web e Intranet de la EPSEVG lo más accesible posible, teniendo en cuenta las WCAG 1.0, WCAG 2.0 y la Norma UNE 139803:2004, tarea no fácil ya que cuanto más accesible es un sitio web, más difícil resulta su realización y eso se traduce en un aumento del tiempo de dedicación.

A veces, no bastará con el diseño web accesible sino que se tendrá que complementar con ayudas técnicas de software, para que, por ejemplo, con un sintetizador de voz, personas ciegas tengan acceso a la web.

A parte del desarrollo implementado del proyecto, existe el desarrollo escrito, la elaboración de la memoria. Al no ser un proyecto en el que los resultados sean visibles al 100%, sino que una parte muy pequeña de éste es visible, provoca que su desarrollo se complique.

La memoria de este proyecto se ha dividido de la siguiente manera:

Agradecimientos.

Índice.

Introducción. En este punto se explica la preparación previa realizada para llevar a cabo este proyecto. Se hace una breve introducción a la accesibilidad web y se describen los contenidos básicos de cada capítulo de la presente memoria.

Declaración de intenciones. Explicación del por qué de este proyecto.

Objetivos. En este capítulo se exponen los principales objetivos propuestos para el desarrollo de dicho proyecto.

Ámbito de aplicación. Se detalla, tal y como indica su nombre, donde puede ser aplicado este proyecto. También se explica lo que es la accesibilidad web, sus beneficios, limitaciones, legislación y el estado actual, para conseguir un mayor entendimiento de su ámbito de aplicación.

Estado del arte. En este punto se describen una a una todas las tecnologías y aplicaciones utilizadas en la realización de este proyecto, así como las instituciones que han dado un gran impulso a la accesibilidad web. Se define el marco histórico, historia, inicios de cada una de las tecnologías y su situación actual.

Diseño. En este apartado se explican detalladamente las pautas a seguir y los cambios realizados en el diseño y código del sitio web para conseguir su accesibilidad.

Resultados conseguidos. En este capítulo se explican los resultados obtenidos. También se muestran las validaciones de las diferentes páginas que forman la web, demostrando la accesibilidad del sitio.

Conclusiones. Se verificará si los objetivos planteados han sido adquiridos y se dará una valoración global del proyecto.

Futuras vías de trabajo. Se exponen las posibilidades de desarrollo de esta materia, de la accesibilidad web.

Anexos. Se añade la información y documentación adicional complementaria al proyecto que facilita su comprensión.

Bibliografía. Se numeran las fuentes consultadas que han servido de gran ayuda en la realización de este proyecto.

2 Declaración de intenciones

La idea de realizar el presente proyecto surgió a causa del acto de inauguración del Curso Académico 2008/2009 celebrado el 25 de setiembre de 2008 en la EPSEVG, donde la Sra. Teresa Palahí Joan, Delegada Territorial de la ONCE en Catalunya, habló sobre las medidas de Accesibilidad Web para personas con discapacidad auditiva, como, por ejemplo, la transcripción escrita en pantalla o la interpretación del lenguaje de signos, entre otras.

Esta conferencia nos llamó mucho la atención ya que, mientras la señora Teresa exponía las dificultades que tienen las personas con discapacidad auditiva o las que tienen las personas ciegas a la hora de acceder a una página web, nos poníamos en el lugar de éstas y pensamos en la falta de libertad que tienen para navegar por la red.

Teniendo prevista la realización de otro proyecto con el SIO², propusimos un cambio: realizar un proyecto sobre Accesibilidad Web. Nuestro tutor -autor de la nueva web de la EPSEVG-, no se opuso al cambio de proyecto y nos dio la posibilidad de poder hacer accesible la nueva página web y la Intranet de la escuela. La construcción de la Intranet de la EPSEVG es un proyecto paralelo al nuestro realizado por un becario del SIO, al que nosotros añadiremos la accesibilidad web.

Necesitamos mucha preparación y una adquisición de conocimientos en materias con las que nunca antes habíamos trabajado, como eran los CSS o los lenguajes HTML o PHP; conocimiento de las pautas de accesibilidad marcadas por el W3C o de la reciente ley española que obliga a los organismos públicos a cumplir como mínimo con la segunda prioridad.

Nuestras intenciones eran muy claras, conseguir la máxima Accesibilidad web de la página de la escuela para facilitar el acceso web a personas que realmente necesitan tener sitios accesibles.

Se han creado unos pequeños manuales de cómo hacer tablas y formularios accesibles para que personas que desarrollen otros proyectos como el nuestro puedan incorporar la accesibilidad a las aplicaciones web que realicen.

Creemos que el presente proyecto es una ayuda para todos, ya que es una ayuda para la escuela, para las personas discapacitadas y para los desarrolladores de otros proyectos, que podrán fijarse en el presente y hacer sitios accesibles.

² Sistemas de Información y Organización de la EPSEVG

3 **Objetivos**

Para el desarrollo del proyecto se han propuesto varios objetivos que son partícipes de uno principal. El principal objetivo es hacer que la nueva web e Intranet de la EPSEVG sean accesibles y que se visualicen correctamente en cualquier navegador.

Para realizar tal objetivo, se necesitan de unos conocimientos que han sido adquiridos previamente antes del desarrollo de dicho proyecto. Los objetivos son detallados a continuación:

- Conocer el término accesibilidad web
- Estudiar HTML y PHP
- Estudiar hojas de estilo CSS
- Conocer las herramientas necesarias para la validación de la página
- Conocer las pautas de accesibilidad a seguir
- Conocer software por voz
- Analizar resultados y extraer conclusiones

3.1 **Accesibilidad web**

Como se ha comentado con anterioridad, el proyecto se centra en hacer que la nueva web e Intranet de la EPSEVG sean accesibles. Primero se ha de tener claro el concepto de accesibilidad, esto conlleva a buscar y recopilar información al respecto. Una vez se tiene toda la información se aplican los conocimientos en la elaboración del proyecto.

3.2 **Estudiar HTML y PHP**

Para empezar a desarrollar el proyecto ha sido necesario aprender HTML, que es el lenguaje de marcado predominante en la construcción de cualquier página web. También se ha estudiado PHP, que es el lenguaje de programación utilizado en la creación de esta web. Ha sido necesario conocer estos lenguajes previamente porque se ha tenido que modificar el código para realizar los cambios pertinentes y así tener una web e Intranet accesibles.

3.3 **Estudiar hojas de estilo CSS**

Las hojas de estilo CSS ha sido lo que más se ha tenido que modificar, ya que se ha cambiado el estilo de la web e Intranet para hacerlas accesibles. Esto, ha requerido un estudio a fondo del diseño de la web, que incluye los CSS para poder realizar las modificaciones pertinentes en el código. Los CSS u Hojas de Estilo en Cascada es un lenguaje formal utilizado principalmente para separar la estructura o el contenido de un documento de su presentación. Para que una web sea accesible ha de cumplir esta parte al 100%.

3.4 Herramientas necesarias en la validación

Existen una serie de herramientas creadas por el W3C, un consorcio internacional cuyo principal objetivo es facilitar el acceso a Internet a las personas con discapacidad, que validan la URL indicada, mostrando los errores cometidos y las pautas a seguir para conseguir que el sitio web sea accesible. Estas herramientas son los validadores de CSS del W3C, de la página web o el de código XHTML.

3.5 Pautas a seguir para la accesibilidad web

Existen unas pautas creadas por el W3C, que aún no siendo normativa son esenciales para crear un sitio web accesible, por tanto, se han tenido que aplicar y validar constantemente para ver si los resultados obtenidos eran los esperados.

3.6 Conocimiento del programa JAWS

El programa JAWS es un sintetizador de voz que utilizan las personas invidentes para acceder a la información de la web. Se han tenido que adquirir los conocimientos necesarios para el buen funcionamiento de este y así poder comprobar si todas las modificaciones hechas en el código eran las correctas. Por este motivo, nos pusimos en contacto con la Cátedra de Accesibilidad y nos ofrecieron la oportunidad de contactar con un trabajador de la ONCE, los conocidos TIFLO, para que nos explicara el funcionamiento de este programa.

3.7 Analizar resultados y extraer conclusiones

El primer y último objetivo marcado es hacer la web e Intranet de la EPSEVG accesible. Las pautas de accesibilidad creadas por el W3C se dividen en tres niveles de aplicación. Según lo indicado en la nueva ley vigente desde diciembre de 2008, se obliga a los organismos públicos a cumplir como mínimo con el segundo nivel de accesibilidad, que implica el cumplimiento del primer nivel. Pero se ha querido llegar más lejos, para que sea lo más accesible posible y no quedarse en un segundo nivel, sino, aplicar el tercero en medida de lo posible. Para conseguir tal fin, se ha tenido que verificar cada uno de los pasos descritos anteriormente.

4 Àmbito de aplicació

4.1 Introducció

En este punto se explicará el ámbito de aplicación de este proyecto, así como una explicación más extensa de la accesibilidad web, que es la parte principal de dicho proyecto.

El ámbito de aplicación es básicamente un sitio o página web. En este proyecto, lo que se pretende realizar es que un sitio web, en este caso la nueva web e intranet de la EPSEVG, sea lo más accesible posible. No obstante, los métodos utilizados son totalmente válidos para la realización de cualquier web, ya que no interfieren de manera negativa en el diseño, sino, todo lo contrario, facilitan el acceso a la información a los usuarios y hacen que la web sea más fácil de entender.

El presente proyecto trabaja básicamente con los conceptos de accesibilidad web y lo que ello conlleva. Por tanto, a continuación se procederá a explicar estos conceptos.

4.2 ¿Qué es la Accesibilidad Web?

La accesibilidad web es un término que hace referencia a que personas con discapacidad tengan acceso a la web, ya que muchas veces se encuentran barreras, sobretodo en el diseño y en el código web, que dificultan o imposibilitan el acceso de estas personas a la información.

Existen millones de personas con discapacidad que no pueden utilizar la Web. El porcentaje se sitúa, en muchas poblaciones entre el 10 y el 20 %. No todas las discapacidades afectan al acceso a tecnologías de información como Internet, ya que personas que necesiten silla de ruedas no se verán perjudicadas, por ejemplo. Principalmente, estas barreras son las dificultades que pueden tener personas con discapacidad física visual, auditiva y cognitiva/neurológica.

Discapacidades a las que hace referencia la Accesibilidad Web:

- Visuales: En sus distintos grados, desde la baja visión a la ceguera total, además de problemas para distinguir colores (daltonismo).
- Auditivas: Sordera o deficiencias auditivas.
- Cognitivas: Dificultades de aprendizaje (dislexia, discalculia, etc.) o discapacidades cognitivas que afecten a la memoria, la atención, las habilidades lógicas, etc.

La accesibilidad web hace que los contenidos de las páginas sean más usables para todos los usuarios, incluyendo las personas mayores que ven mermadas sus capacidades con la edad.

El documento "[Cómo utilizan la Web personas con discapacidad](#)" muestra las dificultades que tienen las personas con discapacidad para acceder a Internet e incluye algunos escenarios basados en ejemplos reales.

Cuanto más software y sitios Web accesibles estén disponibles, más personas con discapacidad podrán utilizar la Web y contribuir de forma más eficiente.

Con esta idea de acceso universal a Internet, nace la **Iniciativa de Accesibilidad Web**, conocida como **WAI**³. Se trata de una actividad desarrollada por el **W3C**, un consorcio industrial internacional formado por más de 300 asociaciones, cuyo objetivo es facilitar el acceso a Internet a las personas con discapacidad, desarrollando pautas de accesibilidad, mejorando las herramientas para la evaluación de accesibilidad Web y abriendo nuevos campos a través de la investigación en este área.

Estas pautas están dirigidas a los diseñadores de sitios Web, a las personas que verifican la accesibilidad de estos sitios y a todos los interesados en que personas con discapacidad tengan acceso a Internet.

Un principio básico de la accesibilidad Web es la flexibilidad con el objetivo de satisfacer diferentes necesidades, situaciones y preferencias. Es por este motivo que de la accesibilidad Web no solamente se benefician personas con discapacidad, sino también organizaciones y personas sin discapacidad, ya que facilita su acceso.

4.3 Pautas, prioridades y Legislación

Las "Pautas de Accesibilidad al Contenido en la Web 1.0" (WCAG 1.0) son una especificación del W3C, desarrolladas por la WAI. Consiste en catorce pautas con sus respectivos puntos de verificación. Estos puntos definen cómo aplicar estas pautas a las páginas web. Un apéndice de estas pautas, la "Lista de puntos de verificación para las Pautas de Accesibilidad al contenido en la Web 1.0" presenta los puntos de verificación clasificados por prioridades y conceptos, para facilitar su búsqueda.

En diciembre de 2008 el W3C aprobó las "Pautas de Accesibilidad al Contenido en la Web 2.0" (WCAG 2.0). Estas pautas son una mejora de las anteriores. Se dividen en cuatro principios: perceptibilidad, operabilidad, comprensibilidad y robustez. Por debajo de estos principios están las pautas. Son 12 pautas que proporcionan los objetivos básicos para lograr un sitio Web accesible. Para cada pauta se proporcionan los criterios de éxito verificables que se dividen en tres niveles de conformidad: A, AA y AAA. Para cada una de las pautas y criterios de éxito se han documentado una amplia variedad de técnicas. Las técnicas son informativas y se agrupan en dos categorías: las suficientes para cumplir con los criterios de éxito y las aconsejables, que van más allá de los requisitos de cada criterio de éxito individual y que permite a los autores cumplir mejor con las pautas.

Tanto para la WCAG 1.0 como para la WCAG 2.0 existen tres niveles de prioridad con sus correspondientes niveles de adecuación:

- Nivel de adecuación "**A**" que cumple con los criterios mínimos de accesibilidad. Este nivel incluye los puntos de verificación de la prioridad 1.
- Nivel de adecuación "**Doble A**" (**AA**) que con cumple los criterios extendidos de la accesibilidad. Este nivel incluye los puntos de verificación de las prioridades 1 y 2.

³ Web Accessibility Initiative

- Nivel de adecuación “Triple A” (**AAA**) que cumple con los criterios de accesibilidad máxima. Este nivel incluye los puntos de verificación de las prioridades 1, 2 y 3.

A continuación se detallarán los tres niveles de prioridad según el W3C:

- Prioridad 1. “Un desarrollador de contenidos de páginas Web **tiene** que satisfacer este punto de verificación. De otra forma, uno o más grupos de usuarios encontrarán imposible acceder a la información del documento. Satisfacer este punto de verificación es un requerimiento básico para que algunos grupos puedan usar los documentos Web”.
- Prioridad 2. “Un desarrollador de contenidos de páginas Web **debe** satisfacer este punto de verificación. De otra forma, uno o más grupos encontrarán dificultades en el acceso a la información del documento. Satisfacer este punto de verificación eliminará importantes barreras de acceso a los documentos Web”.
- Prioridad 3. “Un desarrollador de contenidos de páginas Web **puede** satisfacer este punto de verificación. De otra forma, uno o más grupos de usuarios encontrarán alguna dificultad para acceder a la información del documento. Satisfacer este punto de verificación mejorará la accesibilidad de los documentos Web”.

Estos niveles de prioridad pueden variar bajo ciertas condiciones, indicadas en las Pautas de Accesibilidad. Estas pautas no tienen carácter legislativo y por tanto, la especificación "Pautas de Accesibilidad al Contenido en la Web" no es una normativa, pero actualmente existe una ley que obliga, desde diciembre de 2008 a las instituciones públicas españolas a cumplir como mínimo la segunda prioridad de las pautas de accesibilidad. Esta ley se detalla en el siguiente capítulo.

4.4 Beneficios

Los principales beneficios que ofrece la accesibilidad web, son:

- Aumenta el número de potenciales visitantes de la página web:

Esta es una razón muy importante para una empresa, como por ejemplo la EPSEVG, que pretenda captar nuevos clientes. Cuando una página web es accesible disminuye las barreras que dificultan su acceso. Una página web que cumple los estándares es más probable que se visualice correctamente en cualquier dispositivo con cualquier navegador.

- Reduce el tiempo de carga de las páginas web y la carga del servidor web:

Al separar el contenido de la información sobre la presentación de una página web mediante CSS se logra reducir el tamaño de las páginas web y, por tanto, el tiempo de carga.

- Facilita el acceso a Internet de los usuarios con discapacidad.

4.5 Limitaciones

La finalidad de la accesibilidad es conseguir que todos los sitios web, incluyendo los multimedia, funcionen bien para todos los usuarios, por tanto, los sitios web accesibles no tienen porqué ser aburridos, sombríos, etc. No se deberían diseñar de manera muy diferente a un sitio normal, sino que han de ser diseñados con flexibilidad: flexibles para que los usuarios puedan acceder a ellos mediante teclado, ratón, voz, etc., según las necesidades del usuario, y flexibles respecto al diseño, para que no se conviertan en algo pesado y aburrido.

Esta flexibilidad, muchas veces, se convertirá en algo difícil de realizar, ya que no siempre será posible hacer una página web accesible al 100%. La facilidad o dificultad de hacer sitios web accesibles depende de una gran variedad de factores, como el tamaño del sitio, el hardware, su complejidad, las herramientas utilizadas en su creación, etc.

La realización de un sitio web accesible requiere de unos conocimientos previos y de un aumento de las horas invertidas en la creación, ya que cada pequeño avance ha de ser validado con las herramientas de validación oportunas, para cumplir con los estándares. Este aumento de horas se convierte en un incremento de coste, pero una vez se tienen los conocimientos para el desarrollo de una página Web accesible, el coste de desarrollar y mantener la página web se puede ver reducido frente a una no accesible, debido a la utilización de hojas de estilo o a las actualizaciones de la página, ya que una página web accesible es una menos propensa a contener errores y más sencilla de actualizar. El rendimiento y la amplitud del número de visitas de los sitios accesibles suelen ser rentables.

4.6 Estado actual

Uno de los principales motivos por los que todavía, organismos públicos y universidades no hayan adaptado sus páginas web a la accesibilidad, estando obligados a hacerlo por ley, es que requiere una gran inversión de tiempo para aprender nuevas tecnologías y aplicarlas a la web. Tiempo en el que programadores podrían invertirlo en realizar otras tareas, ya que la realización de un sitio o una página web se alarga más de lo normal al tratar de hacerla accesible. No es un resultado visible, ya que no se podrá apreciar visualmente todo el trabajo realizado, pero se necesita retocar el código, en este caso PHP, modificar CSS y sobretodo, aprender previamente lenguajes de programación, hojas de estilo y otras utilidades y aplicaciones web para hacer posible la validación de las páginas, entre otras cosas.

Otro de los motivos es que el público al que va dirigido es minoritario, a pesar de ser un grupo muy grande y en aumento, ya que cada vez son más los discapacitados de nuestra sociedad.

Conseguir que una página web sea accesible no es nada fácil, por este motivo somos la primera universidad catalana y una de las primeras del estado español en cumplir con la nueva ley.

5 Estado del arte

5.1 Introducción

En este capítulo se da una visión del marco histórico de las tecnologías utilizadas en la realización de este proyecto, así como de los programas y organismos que se han dedicado al auge de la accesibilidad web y que también, en algún momento nos han servido de gran ayuda.

Se comentarán los inicios de cada una de las tecnologías, programas y organismos requeridos, de su funcionalidad y de la situación actual de cada uno de ellos.

5.2 Accesibilidad web

5.2.1 Accesibilidad web

Hablar de *Accesibilidad Web* es hablar de un acceso universal a la Web, es querer reducir al máximo posible las barreras existentes en las páginas web. Claro está que no todo el mundo tiene acceso a Internet, debido a su situación geográfica, economía, etc. Pero cierto es, que el número de usuarios de la web está en constante aumento. Al hablar de accesibilidad web se está haciendo referencia a un diseño web que va a permitir que personas con discapacidad puedan percibir, entender, navegar e interactuar con la web.

Para algunas personas con discapacidad, las barreras de acceso a la web pueden significar la falta de acceso a la información general de la web e información específica, relacionada con el empleo, con las intranets del puesto de trabajo, con actividades o programas educativos, con la incapacidad de acceder al comercio en la Web, etc.

5.2.2 Legislación

*La Orden PRE/1551/2003, de 10 junio, por la que se desarrolla la disposición final primera del Real Decreto 209/2003, de 21 de febrero de 2003, que regula los registros y las notificaciones telemáticas, así como la utilización de medios telemáticos para la sustitución de certificados por los ciudadanos, establece que el registro telemático y el servicio de notificación telemática deberán cumplir los requerimientos en materia de accesibilidad establecidos por la **Iniciativa de Accesibilidad Web (WAI)** del W3C y en particular las especificaciones de la Recomendación de 5 de mayo de 1999 sobre **Pautas de Accesibilidad del Contenido en la Web (WCAG 1.0)**, en su nivel AA. En el Anexo 5 se detalla cada uno de los puntos de verificación de la WCAG1.*

Con esto se quiere decir que la Administración General del Estado debería cumplir con el nivel de accesibilidad Doble A (AA), desde el 1 de Enero de 2006, pero actualmente, desde el 31 de diciembre de 2008, la normativa española obliga a las Administraciones

Públicas a satisfacer, como mínimo el nivel dos de prioridad de la accesibilidad Web. A continuación se puede observar el texto legislativo, así como el número de BOE en el que fue publicado:

Jefatura del Estado (BOE número 312 de 29/12/2007)

Artículo 27. Obligaciones previas a la contratación. Dieciocho.

«A partir del 31 de diciembre de 2008, las páginas de Internet de las Administraciones Públicas satisfarán, como mínimo, el nivel medio de los criterios de accesibilidad al contenido generalmente reconocidos. Excepcionalmente, esta obligación no será aplicable cuando una funcionalidad o servicio no disponga de una solución tecnológica que permita su accesibilidad.»

Como ninguna ley española puede basarse en unas pautas que no hayan sido creadas por un organismo oficial de normalización, las WCAG 1.0 se tuvieron que reescribir en la Norma [UNE 139803:2004](#). En el Anexo 6, se muestra la correspondencia entre la norma UNE 139803:2004 y los puntos de verificación WAI.

En diciembre de 2008 se publicaron las “Pautas de Accesibilidad al Contenido en la Web 2.0” (WCAG 2.0). Estas pautas no son normativa pero es recomendable tenerlas en cuenta a la hora de realizar un sitio web accesible.

5.2.3 Estado actual

Anteriormente, en lo que al marco histórico se refiere, se creaban dos páginas diferentes para un mismo sitio web, es decir, una web accesible y la otra no, ya que resultaba mucho más sencillo de realizar. El interaccionar las dos páginas, el convertirlas en una sola no es tarea fácil, puesto que han de cumplir muchas pautas adicionales, se ha de validar página a página para asegurarse de que cada una de ellas cumpla con las pautas de accesibilidad, con la correcta utilización de CSS, con el contraste de colores adecuados, etc. El llevar a cabo esta tarea conlleva implícitamente el aprendizaje de nuevos programas y conocimientos que no sería necesarios tenerlos si a la hora de diseñar la página no se tuviera en cuenta la accesibilidad.

Según se ha podido comprobar, a rasgos generales se puede decir que a la accesibilidad web en España aún le queda un largo recorrido. Decir esto cuando se ha visto en el punto anterior que la ley española obliga a cumplir como mínimo el segundo nivel de prioridad de accesibilidad web, no parece alentador.

A modo de comprobación adjuntamos una de las últimas noticias sobre accesibilidad web, demostrando incluso que las instituciones públicas más fuertes no cumplen con la normativa. Por eso, muchas instituciones, han sido multadas con sanciones entre los 30000€ y los 90000€, entre ellas la Universidad Autónoma de Madrid (UAM).

<http://solidaridaddigital.discapnet.es/SolidaridadDigital/Noticias/Accesibilidad/DetalleNoticia.aspx?id=5411>

5.3 W3C i WAI

En 1989 Tim Berners-Lee creó la World Wide Web; desarrolló el primer servidor y el primer programa cliente en 1990; creó la primera versión del HTML; y finalmente, fue el director y fundador del consorcio W3C.

El W3C se creó en octubre de 1994 en el Laboratorio de Ciencias Informáticas del Instituto de Tecnología de Massachusetts (MIT/LCS), en colaboración con el CERN⁴, DARPA⁵ y la Comisión Europea. El objetivo principal de este consorcio es facilitar el acceso a internet a las personas con discapacidad.

En octubre de 1996 se publicó la primera Recomendación del W3C “Gráficos de Red Portátiles PNG 1.0” y en diciembre de ese mismo año se publicó el CSS nivel 1, hojas de estilo en cascada cuyo principal objetivo es separar la estructura del contenido de las páginas web. En 1998 saldría una nueva versión de CSS, el CSS nivel 2.

El compromiso del W3C de encaminar la Web a su máximo potencial incluye promover un alto grado de accesibilidad para las personas con discapacidad. El grupo de trabajo permanente Web Accessibility Initiative (WAI), creado en febrero de 1997, en coordinación con organizaciones al rededor de todo el mundo, persigue la accesibilidad Web a través de cuatro áreas de trabajo principales: tecnología, herramientas, formación y difusión, e investigación y desarrollo.

El WAI está dirigido por Judy Brewer, Directora del Programa Internacional, y por Daniel Dardailler como Director de Proyectos.

A continuación se nombrarán las acciones posteriores más significativas del W3C:

- Diciembre 1997: el HTML 4.0 añade tablas, scripting, internacionalización y accesibilidad. Estas nuevas características permitían la utilización de hojas de estilo, hacer la web más dinámica y proporcionar un contenido más internacional y más accesible para usuarios discapacitados.
- 5 de mayo de 1999: publicación de las “Pautas de Accesibilidad al Contenido en la Web 1.0”.
- Agosto de 2000: nacen los Gráficos Vectoriales Escalables 1.0 (SVG) que servirán de base a los teléfonos móviles de nueva generación.
- Mayo de 2001: el Esquema XML proporciona la vía para la construcción de aplicaciones más potentes y versátiles.
- Enero de 2002: se lanza la Actividad de Servicios Web que proporciona que diferentes aplicaciones puedan interoperar y que funcionen en una gran variedad de entornos y/o plataformas.
- Mayo 2003: el W3C publica su Política de Patentes libre de derechos de autor.
- Marzo 2004: el W3C le da voz a la Web con VoiceXML 2.0.

4 Conseil Européen pour la Recherche Nucléaire u Organización Europea para la Investigación Nuclear

5 Defense Advanced Research Projects Agency o Agencia de Investigación de Proyectos Avanzados de Defensa

- 11 de diciembre de 2008: publicación de las “Pautas de Accesibilidad al Contenido en la Web 2.0”.

5.4 Estándares

Los estándares web son un conjunto de recomendaciones dadas por la W3C y otras organizaciones internacionales para crear e interpretar documentos basados en la web.

Un estándar pasa por los siguientes estados:

- Working Draft (borrador de trabajo).
- Last Call (última convocatoria).
- Proposed Recommendation (propuesta de recomendación).
- Candidate Recommendation (recomendación candidata).

El estándar finaliza con la aprobación de la *Recommendation*, el equivalente a una homologación de la propuesta, es decir, un nuevo estándar público y abierto a la web. Es muy importante la presencia de la W3C, ya que de esta forma se evita que fabricantes se hagan con el monopolio de la explotación web.

5.5 Tecnologías web

5.5.1 Introducción al código abierto

PHP, así como Apache y MySQL forman parte del grupo de programas de software de código abierto. El término “código abierto” se tomó en 1998 cuando Netscape publicó el código fuente de su navegador “Navigator”. A partir de este momento, un grupo de programadores informáticos crearon formalmente las Iniciativas de código Abierto (*OSI*⁶) proponiendo mejoras y presentando sugerencias para aumentar el rendimiento del navegador. El primer sistema operativo en recibir la denominación de código abierto fue Linux. Posteriormente, se le fueron sumando otros sistemas operativos y programas.

Características de los programas de código abierto:

- El software de código abierto es de libre distribución, es decir, gratuito, y está disponible para todos y por definición, no puede exigir ningún tipo de licencia o cuota de ventas.
- Según la OSI, los programas de código abierto no pueden depender de ninguna tecnología o de un concreto estilo de interfaz. Tampoco deben pertenecer a ninguna plataforma concreta, ya que han de ser tecnológicamente neutrales para que todos puedan disponer de ellos.

- Si se distribuye un programa de código abierto junto con otros programas, éstos podrán ser de código abierto o comerciales, ya que el uso de unos no debe restringir el uso de otros.
- No pueden discriminar a personas ni a grupos sociales y no se pueden limitar a ser utilizados por un solo campo profesional. Por ejemplo, un programa diseñado para el sector de las finanzas no debería estar limitado a ese sector, sino que ha estar disponible a cualquier persona que pudiera necesitarlo, pudiéndolo modificar para adaptarlo a sus necesidades.

En la actualidad, todos los programas con el sello OSI Certification se consideran código abierto, ya que han pasado la definición. Todos estos programas se encuentran disponibles en la Web.

5.5.2 Tipos de páginas web

Podemos diferenciar dos tipos de páginas web, páginas web estáticas o dinámicas. El contenido de las páginas web estáticas viene predeterminado, pero el de las dinámicas se genera al momento de visualizar la página o cuando se solicita al servidor. A continuación se proporciona una tabla con las diferencias más básicas entre los dos principales tipos de páginas:

| PÁGINAS WEB ESTÁTICAS | PÁGINAS WEB DINÁMICAS |
|--|---|
| Ausencia de movimiento y funcionalidad | Aumenta el número de posibilidades respecto al diseño y a su desarrollo. |
| Se realizan en XHTML o HTML. | Se utilizan diversos lenguajes y técnicas de programación. Puede realizarse con software de libre distribución. |
| No se pueden utilizar funcionalidades como bases de datos, foros, etc. | Permite un gran número de funcionalidades como bases de datos, foros, etc. |

Hoy en día, la mayoría de programadores y diseñadores optan por hacer páginas web dinámicas, ya que aportan grandes ventajas respecto a las estáticas. Entonces, para ello, se necesitan otros lenguajes de programación como PHP. Este lenguaje, junto con HTML es el que se ha utilizado en la realización de la página e Intranet de la EPSEVG.

5.5.3 HTML

HTML⁷ es el lenguaje de marcado⁸ predominante en la construcción de páginas web. Se trata de un conjunto de etiquetas que indican la forma en la que se ha de presentar el texto y los otros elementos del navegador.

Fue creado originalmente por el físico nuclear Tim Bernes-Lee, creador de la World Wide Web, en el CERN en el año 1991. Se describieron diferentes elementos para un diseño inicial relativamente simple de HTML. El borrador estándar HTML 3.0 fue propuesto por la W3C en 1995. Actualmente la última versión de HTML existente es HTML 4.0.

El diseño en HTML, a parte de cumplir con las especificaciones propias del lenguaje debe respetar unos criterios de accesibilidad web siguiendo unas pautas y normativas creadas por la W3C, aunque muchos países, incluida España, tienen especificaciones propias como es la Norma UNE 139803:2004 .

5.5.4 PHP

PHP⁹, se trata de un lenguaje de programación del lado del servidor¹⁰, de código abierto interpretado de alto nivel, que puede estar embebido en páginas HTML. Este lenguaje está diseñado especialmente para desarrollo web, para crear páginas HTML dinámicas. Su sintaxis es parecida a Java o C.

Fue creado originalmente por Rasmus Lendorf en 1994. El equipo de desarrollo que se encarga de mantener las actualizaciones disponibles para este lenguaje es *"The PHP Group"*, que inicialmente fue liderado por los creadores de PHP. Han existido muchas versiones de PHP, pero la más reciente es la 5.2.9-1 (para Windows) y data del 10 de marzo del 2009, aunque está previsto el lanzamiento de la rama 6 de PHP. La primera versión fue el PHP 1.0, llamado *"PHP Tools"*.

PHP puede ser desplegado en la mayoría de servidores web y en casi todos los sistemas operativos de manera gratuita, sin coste alguno. Actualmente, PHP se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores. Aunque PHP tenga que competir con otros lenguajes nuevos menos poderosos, la utilización de éste sigue en alza.

5.5.5 Hojas de estilo o CSS

Las hojas de estilo en cascada o CSS¹¹ son un lenguaje formal utilizado para definir la presentación de un documento estructurado y escrito en HTML. El W3C es el encargado

7 Hiper Text Markup Language

8 Forma de codificar un documento que, junto con el texto, incorpora etiquetas o marcas que contienen información adicional acerca de la estructura del texto o de su presentación.

9 Hypertext Pre-Processor

10 Los lenguajes de programación del lado del servidor son aquellos en los que el código se interpreta en el servidor y no en el navegador del usuario: el resultado del código PHP interpretado por el servidor HTTP se envía al navegador del usuario en forma de HTML, un documento, imagen, etc.

11 Cascading Style Sheets

de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

Las hojas de estilo aparecieron poco después que el lenguaje de etiquetas SGML¹², alrededor del año 1970. El SGML es un sistema para la organización y etiquetado de documentos. Nació para dar estilos a los documentos electrónicos. Aunque el gran impulso de los lenguajes de las hojas de estilo se dio con la aparición de Internet y del gran crecimiento que experimentó HTML, basado en el SGML.

El W3C propuso la creación de un lenguaje de hojas de estilo específico para las páginas web. Se realizaron nueve propuestas de las cuales sólo dieron frutos dos:

- CHSS (Cascading HTML Style Sheets). Fue realizada por Hakon Viwm Lie.
- SSP (Stream-based Style Sheet Proposal). Fue realizada por Bert Bos.

Entre 1994 y 1995 Lie y Bos se unieron para crear un lenguaje formado con lo mejor de las dos propuestas, llamado CSS. En 1995 la W3C apostó por el desarrollo y estandarización del CSS y a finales de 1996 publicó la primera versión oficial CSS1. La versión que utilizan actualmente la mayoría de navegadores es la CSS2.1. Se está trabajando en la versión CSS3, pero actualmente, sólo consta de borradores.

La idea principal que se encuentra detrás del desarrollo de CSS es separar el contenido o estructura de un documento de su presentación. Los CSS permiten a los desarrolladores web controlar el estilo y el formato de múltiples páginas web al mismo tiempo.

Actualmente los navegadores modernos implementan CSS1 correctamente aunque existen pequeñas diferencias de implementación según las marcas y versiones de navegador. CSS2 se está implementado parcialmente en los navegadores más recientes. La ventaja más clara de las hojas de estilo es que a nivel de usuario se mejora la calidad de la información visualizada a través de la página, permite estructurar los estilos de la página y así hacerla más accesible. Por el contrario, a nivel del programador existe una desventaja y es el hecho de aprender un nuevo lenguaje a la hora de crear un sitio web.

5.6 Servidor de pruebas Kato

Kato es el servidor web Apache donde se encuentran todas las páginas en PHP i las bases de datos MySQL. Trabaja con un sistema operativo tipo Linux, con distribución Ubuntu Server versión 7.04. La versión de Apache es la 2.2.3, la de MySQL es la 5.0.38 y la de PHP la 5.2.1.

Está previsto cambiar la máquina por una más potente, con más memoria, para que la ejecución de las páginas sea más rápida. En este caso se instalarían versiones más nuevas de todas las aplicaciones: Ubuntu Server versión 8.10, PHP versión 5.2.4 y MySQL versión 5.0.51.

12 Standard Generalized Markup Language o Lenguaje de Marcado Generalizado.

5.7 Herramientas utilizadas

5.7.1 Dreamweaver

Adobe Dreamweaver es un editor de páginas web. Este programa está indicado para el diseño, edición y construcción de páginas web basados en estándares. Es el editor más utilizado en el sector del diseño y la programación web debido a su funcionalidad y a su integración con otras herramientas. Actualmente, da soporte a los estándares establecidos por el W3C. El principal competidor de este editor es el Microsoft Expression Web.

Una página web puede ser creada con un simple procesador de texto, como el Bloc de Notas. El bloc de notas fue creado por Microsoft en el 1985 y siempre va incluido en sus sistemas operativos. Hoy en día la utilización de procesadores de texto para la creación de sitios web está en desuso y cada vez más se utilizan editores gráficos, ya que facilitan la tarea.

La primera versión del Macromedia Dreamweaver data de 1997. Fue creado inicialmente por Macromedia, pero actualmente es producido por Adobe Systems. La última versión es el Dreamweaver CS4.

Dreamweaver, es un editor que permite crear páginas web en HTML ocultando el código y viendo los resultados que se van obteniendo. Por ejemplo, resulta realmente sencillo hacer una página web con tablas mediante este editor, ya que consiste en insertar una tabla e ir añadiendo el contenido donde se quiera. Por este motivo ha sido muy criticado por muchos profesionales porque introduce código inútil que hace que la página web

sea más larga, cuando uno de los objetivos es hacer las páginas con el menor código posible para que de este modo, entre otras cosas, tarden menos tiempo en realizar la carga en el navegador. En versiones posteriores se ha conseguido mejorar este hándicap aumentando el soporte a los CSS y permitiendo otras maneras de crear páginas web sin tablas. Otro motivo de crítica fue que en el pasado, no cumplía con los estándares del W3C. Ese problema ha sido solucionado en las últimas versiones de este programa.

En la siguiente figura podemos observar como la ventana se divide en tres partes:

- La parte superior muestra el código de la página que se está editando.
- La parte inferior muestra el diseño de la página, tal y como se vería en el navegador.
- La parte derecha muestra la estructura de carpetas donde se guardan los ficheros que se editan.

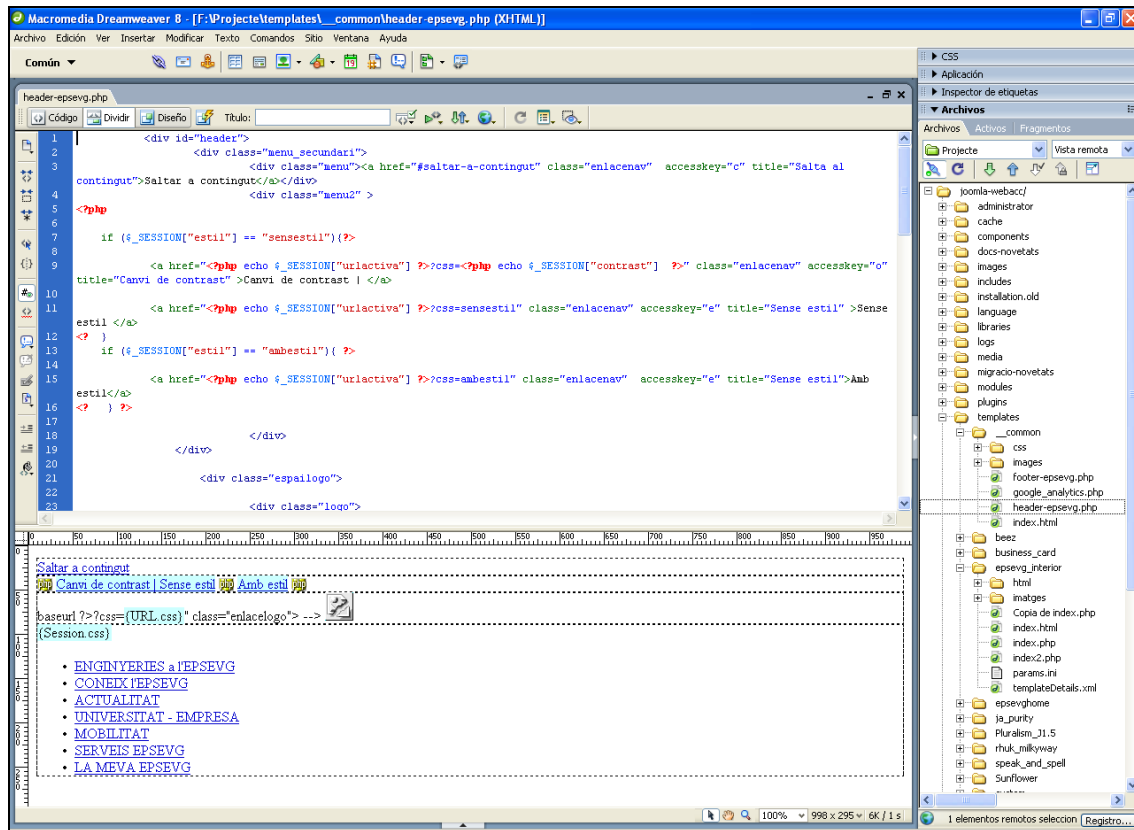


Figura 1. Macromedia Dreamweaver.

5.7.2 Internet Explorer y Mozilla Firefox

Internet Explorer es un navegador web, producido por Microsoft para el sistema operativo Windows. Más tarde sería producido para Sun Solaris y Apple Macintosh. El

proyecto de Internet Explorer lo inició en el verano de 1994 Thomas Reardon y posteriormente fue dirigido por Benjamin Slivka. Fue lanzado por primera vez como complemento de Microsoft Plus! Para Windows 95 en el año 1995. Posteriormente, se lanzó gratuitamente mediante descarga. Su versión más reciente es la 8.0.

Ha sido el navegador web más utilizado desde 1999. El segundo navegador más utilizado es Mozilla Firefox.

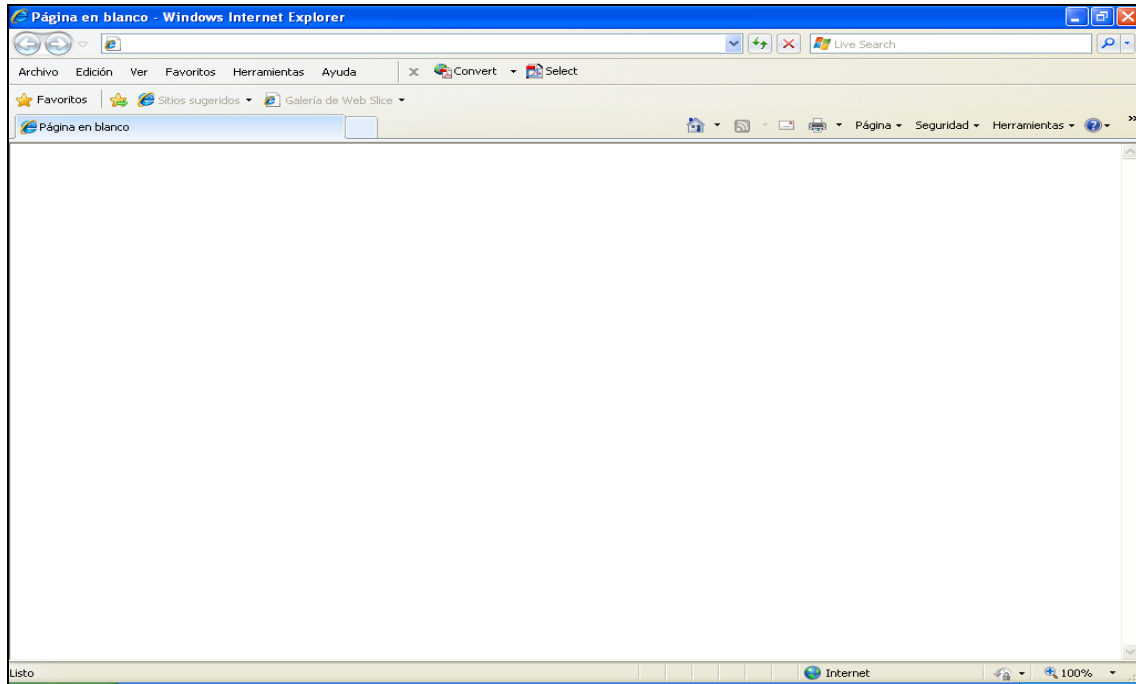


Figura 2. Navegador Internet Explorer.

Mozilla Firefox es un navegador libre y de código abierto. Es un descendiente de Mozilla Application Suite, desarrollado por la Corporación Mozilla, la Fundación Mozilla y un gran número de voluntarios externos. Comenzó como una rama experimental del proyecto a cargo de Dave Hyatt y Blake Ross. Su última versión es la 3.0.

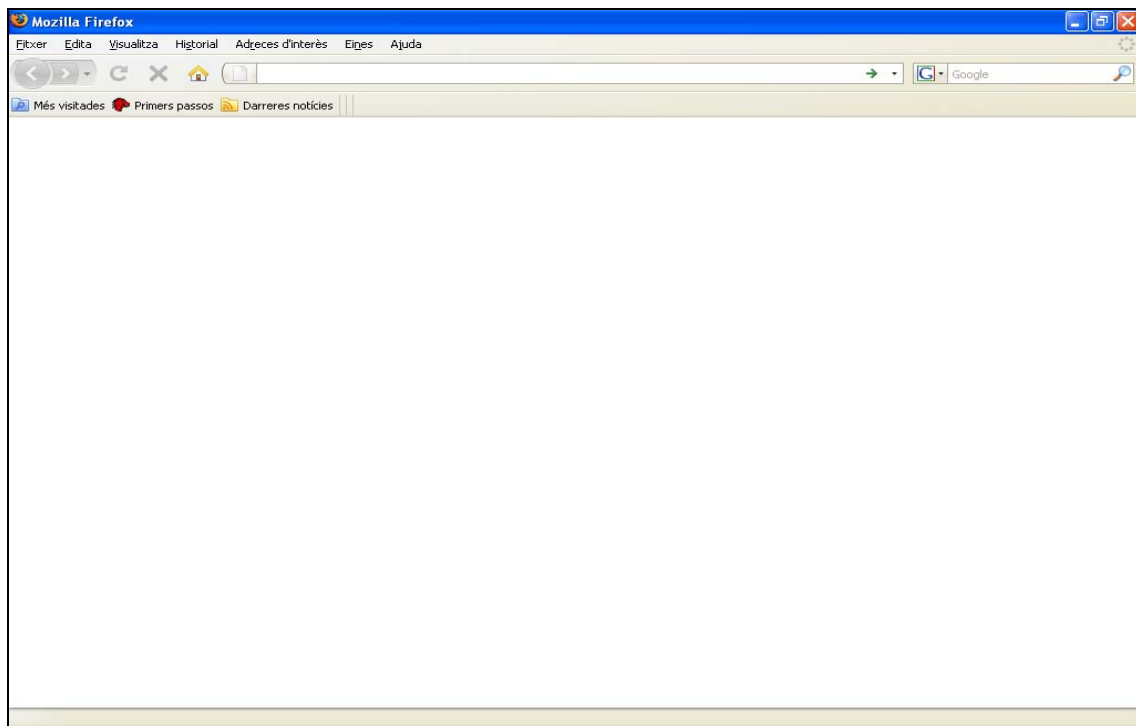


Figura 3. Navegador Mozilla Firefox

En el presente proyecto se han utilizado diferentes navegadores con un fin muy claro, eliminar todas las funciones ajenas a un navegador, con lo cual se mejora código e interfaz.

Cada avance del proyecto no es comprobado únicamente mediante las herramientas de validación que se explican a continuación, sino que también es comprobado en los distintos tipos de navegador existentes, Internet Explorer, Mozilla Firefox e incluso Chrome (navegador de Google) con el fin de que no existan diferencias de un navegador a otro a la hora de mostrar la página web de la EPSEVG.

5.8 Software de Validación

5.8.1 Validador de CSS de W3C

El servicio de Validación de CSS del W3C es un software libre creado por el W3C en el año 1994 con el fin de ayudar a los diseñadores y desarrolladores web a validar Hojas de Estilo en Cascada (CSS). Puede utilizarse mediante un servicio gratuito en la web, <http://jigsaw.w3.org/css-validator/>, o puede descargarse y ser usado como un programa java.

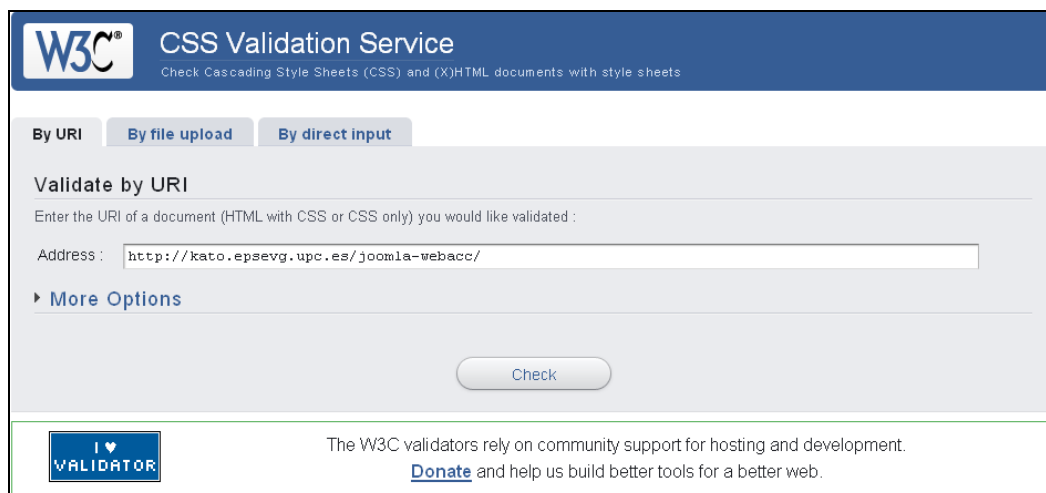


Figura 4. Validador CSS del W3C.

Esta herramienta compara las hojas de estilo con las especificaciones CSS, ayudando a encontrar errores comunes, errores tipográficos, o usos incorrectos de CSS, también muestra cuando el CSS presenta algún riesgo en cuanto a usabilidad. Lo único que hay que hacer es introducir la URL a analizar.

Con esto no se dice que el validador sea una herramienta útil y confiable, pero es una herramienta de software, y como todo software, tiene algunos errores e incidencias. La verdadera referencia sobre Hojas de Estilo en Cascada son las [Especificaciones de CSS](#).

5.8.2 TAW

TAW¹³ es una herramienta que permite analizar, estudiar y validar páginas web con el objetivo de hacerlas accesibles a personas con discapacidad. Existen otras herramientas, como por ejemplo HERA, que realizan la misma función, siendo la más utilizada TAW.

El análisis de las páginas se realiza de acuerdo a las Pautas de Accesibilidad emitidas por la WAI permitiendo, además, configurar la forma de ofrecer los resultados del análisis.

TAW se creó en el año 2000 fruto de un convenio con el CEA-PAT¹⁴, organismo dependiente del Inmerso. Fue ideado para permitir el análisis automático y manual de la accesibilidad web. En el 2002 la Fundación CTIC¹⁵ realizó un nuevo desarrollo para necesidades más específicas.

Toma como referencia las "**Pautas de Accesibilidad para el Contenido Web. WCAG 1.0**" publicadas por el W3C en el 1999. Su uso es completamente gratuito y el único requisito para su ejecución es disponer de un navegador web. Lo único necesario es acceder al portal web www.tawdis.net e indicar la página que queremos analizar: esta herramienta revisará todas las páginas y elementos que componen el sitio. Indica donde está la irregularidad, la prioridad a la que pertenece y las pautas necesarias para su corrección, así como los errores manuales y los automáticos a corregir.

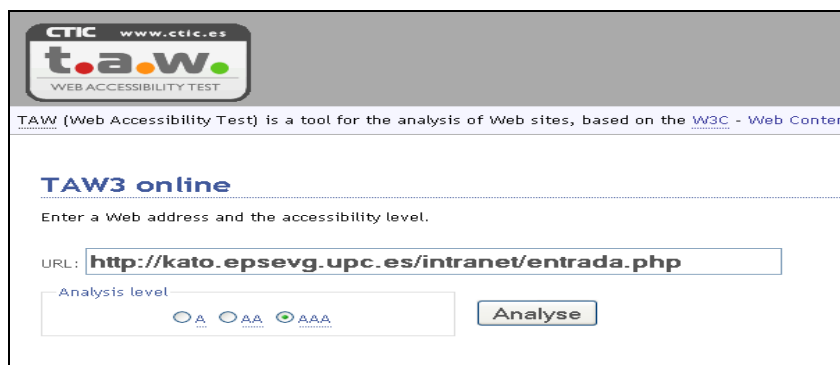


Figura 5. T.A.W.

En la siguiente figura se observa que existen dos tipos de errores, los automáticos y los manuales. Para cumplir con las pautas de accesibilidad se han de reducir los niveles automáticos a cero según la prioridad que se quiera obtener. Los errores manuales se deberán reducir lo máximo posible, aunque muchas veces no se podrá realizar ya que muchos de estos errores son avisos o comprobaciones a tener en cuenta.

13 Test de Accesibilidad Web.

14 Centro Estatal de Autonomía Personal y Ayudas Técnicas.

15 Centro Tecnológico de la Información y la Comunicación



Figura 6. Errores T.A.W.

Existe una versión descargable y gratuita del TAW que ofrece un proceso de validación más detallado. Permite escanear todas las páginas que componen el sitio web de una tirada, mientras que con la aplicación web, se han de introducir manualmente una a una; indica las pautas superadas y nos ofrece un resumen del análisis.

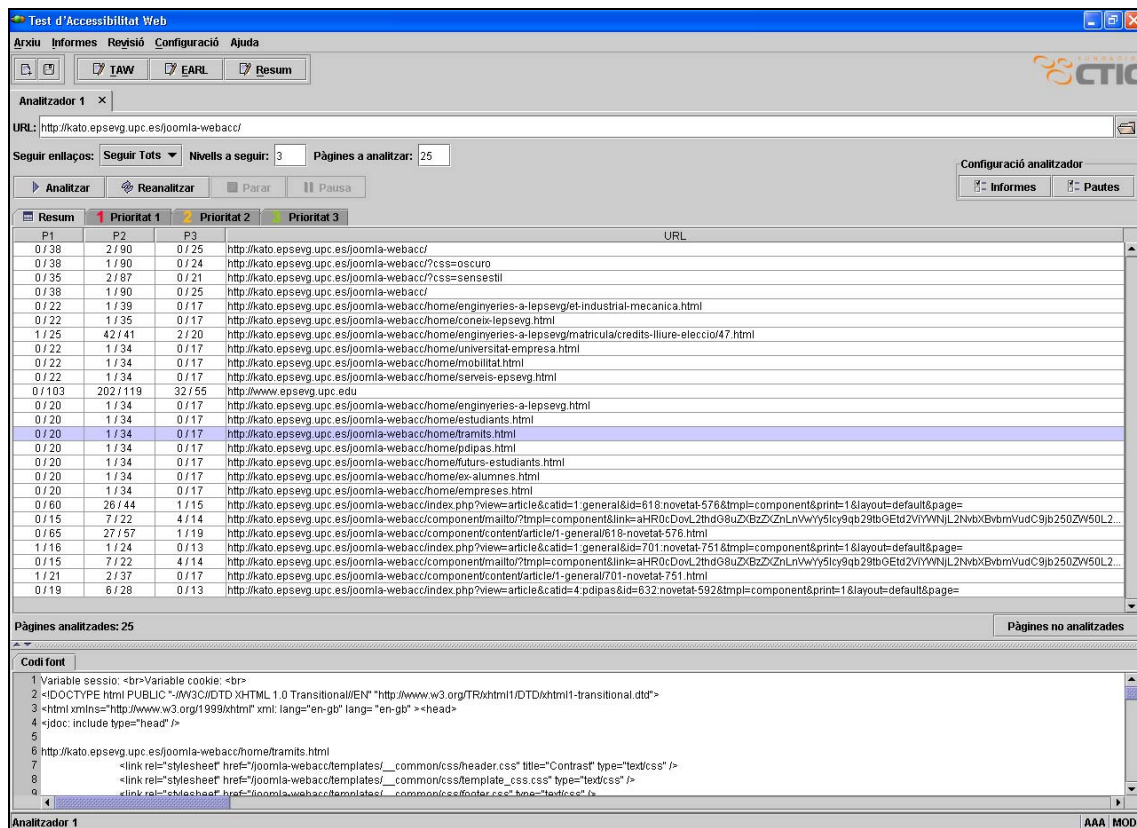


Figura 7. Versión T.A.W. descargable.

Desde finales del 2005 se entregan anualmente los “Premios TAW a la Accesibilidad Web”, divididos en diferentes categorías que premian las Webs más accesibles del estado español.

5.8.3 CONTRAST-A

Contrast-A es una herramienta que permite elegir paletas de color para contrastar dos colores. Este pequeño programa está especialmente diseñado para comprobar si dos colores crean un contraste accesible para aplicarlos a un sitio web.

Permite escoger dos colores pertenecientes a una paleta de color, compararlos y verificar si se cumplen las prioridades de accesibilidad web definidas por el W3C. Una vez escogidas las combinaciones de colores, hace una simulación sobre cómo verían la combinación personas con diferentes trastornos visuales, como por ejemplo, daltonismo e indica si la combinación pasa la guía de accesibilidad del W3C según el brillo y color y según la luminancia. A parte, se puede ver el resultado de las combinaciones de color elegidas en dos textos: en uno se observa el fondo de uno de los colores elegidos y las letras de otro color, y en el segundo texto se observa la combinación de colores invertida.

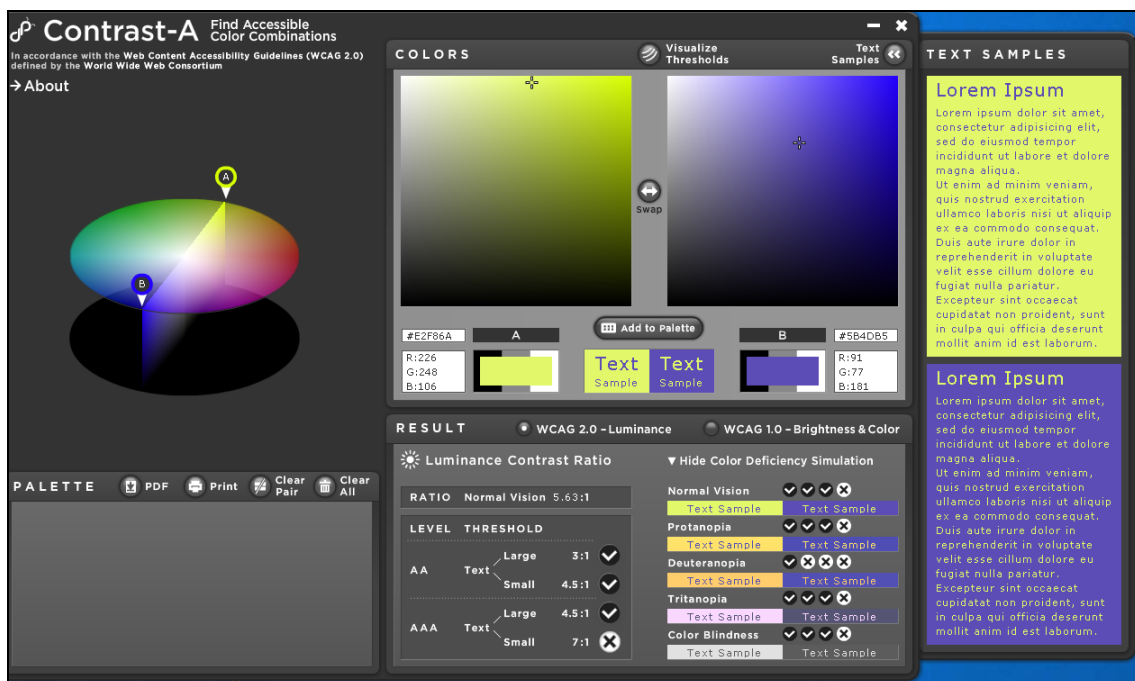


Figura 8. Contrast-A.

5.8.4 JAWS

El programa JAWS¹⁶, es un software lector de pantalla que permite a una persona ciega o con visión reducida utilizar aplicaciones y programas que se ejecuten en cualquier versión de Windows. El mismo software viene con un sintetizador de voz integrado 'Eloquence' en su programación que permite utilizar la tarjeta de sonido de cualquier PC para reproducir su voz sintetizada, es decir, el programa convierte el contenido de la pantalla en sonido, de manera que el usuario pueda acceder o navegar por él sin necesidad de verlo.

JAWS nació en 1989 como iniciativa de Ted Henter, un motociclista que perdió la visión en 1978 en un accidente de automóvil. Ted Henter y su compañero Rex Skipper escribieron el código original JAWS para MSDOS a mediados de los años 80, y la versión 2.0 a mediados de los 90. En 1994 Oppermann escribió JAWS for Windows 1.0. Hasta la fecha, la versión JAWS Windows 10.0 es la última del mercado.

JAWS está ideado para personas ciegas o con visión reducida, por lo tanto, la navegación por los diferentes programas y navegadores existentes se hace mediante teclado y no por mouse. Este es uno de los motivos por los cuales se ha hecho complicado llegar a entender y saber utilizar el programa. La forma correcta para que una persona "no ciega" pueda habituarse a trabajar con JAWS, por ejemplo, los autores del presente proyecto o cualquier programador que se aventure en el tema accesibilidad web, es vendarse los ojos, desconectar el mouse y simplemente escuchar y conocer los atajos de teclado.

La oportunidad de llegar a conocer y dominar el programa ha sido toda una experiencia, incluso ha habido una cita con un TIFLO¹⁷ el cual de forma voluntaria -ya que los cursos de instrucción en JAWS se han de pagar-, nos enseñó el manejo de JAWS tanto el punto de vista del programador como el punto de vista del usuario.

¹⁶ *Job Access With Speech*

¹⁷ *Técnico perteneciente al grupo ONCE*

6 **Diseño.**

6.1 **Introducción**

Este capítulo intenta explicar todo el diseño que se ha llevado a cabo para conseguir que la Web e Intranet de la EPSEVG sean lo más accesible posible.

Para explicar todos los cambios realizados en la Web, se hará una previa explicación de las discapacidades y trastornos que contempla la accesibilidad Web y que se han tenido en cuenta en la realización de este proyecto.

Se explicarán las barreras con las que se encontrarían los usuarios discapacitados que quisieran acceder a esta Web, si no fuera accesible. También se nombrarán algunas de las ayudas técnicas que pueden necesitar los diferentes usuarios y se citarán las soluciones realizadas para eliminar esas barreras. Una vez citadas, se explicará cada uno de los cambios realizados. Estos cambios se basan en la aplicación de las WCAG 1.0. Se relacionará cada cambio con la pauta de accesibilidad aplicada y el punto de la Norma UNE 139803:2004 asociado a esta que aparecerá entre paréntesis. Los cambios se han estructurado en cambios generales, tablas y formularios. En los cambios generales, se explicarán aquellos que han requerido modificación de código y se citarán los que no, ya que con el diseño que tenía la web ya se cumplían.

Se nombrarán los puntos de verificación de la WCAG 2.0 que, aun no siendo obligatorio pero sí recomendable, se cumplen.

Se explicarán los archivos de código modificados y creados en el diseño de la web.

Finalmente, se explicará el contenido de la página de Accesibilidad, que contiene una pequeña explicación de la accesibilidad web, los atajos de teclado utilizados y los recursos externos de accesibilidad.

6.2 **Discapacidades a las que afecta la accesibilidad web**

Discapacidades visuales

- Ceguera
- Baja visión
- Daltonismo

Discapacidades auditivas

- Sordera
- Duro de oído

Discapacidades físicas

- Discapacidades motrices

Discapacidades cognitivas y neurológicas

- Dislexia, discalculia
- Trastorno de déficit de atención

- Discapacidad cognitiva
- Falta de memoria
- Discapacidades de salud mentaTrastornos de epilepsia

Discapacidades múltiples

Condiciones relacionadas con el envejecimiento

6.3 En qué consiste cada discapacidad y cambios a realizar en la Web.

6.3.1 Discapacidades visuales

6.3.1.1 Ceguera

Es la pérdida total o parcial del sentido de la vista. Existen varios tipos de ceguera dependiendo del grado y del tipo de pérdida de visión.

Para acceder a la Web, muchas personas ciegas necesitan lectores de pantalla, software que interpreta el texto que se muestra por pantalla y lo dirige a un dispositivo de síntesis de voz o a una línea de braille. Algunas personas invidentes utilizan navegadores solo texto o navegadores de voz en vez de lectores de pantalla. Mediante tabulaciones con el teclado se mueven por los enlaces.

Barreras de la ceguera total

- Imágenes sin texto alternativo o imágenes complejas sin descripción.
- Utilización de tablas, que muchas veces resultan difíciles de leer por los lectores de pantalla o navegadores de voz, ya que la mayoría no tienen sentido cuando se leen celda a celda o de forma lineal.
- Formularios que no permiten que al tabular se lean en el orden adecuado o que contengan etiquetas inadecuadas.
- Formatos de documentos no estándar que dificulten la interpretación mediante los lectores de pantalla.
- Navegadores y herramientas de edición que no permitan soporte de teclado para todos los comandos o que no empleen el interfaz de programación del sistema operativo.

Cambios a realizar

- Proporcionar texto alternativo a todo elemento no textual y descripciones a las imágenes complejas que necesiten ser explicadas para poder entender mejor el contenido.
- Hacer tablas accesibles proporcionando encabezados de fila y columna que serán interpretados por los lectores de pantalla, cosa que facilita la orientación de los usuarios respecto la información. Otras de las medidas tomadas en hacer las tablas accesibles son la agrupación de la información si la tabla es compleja, adición de resúmenes, títulos, etc.

- Hacer formularios accesibles proporcionándoles atajos de teclado a los controles más importantes, crear un orden de tabulación, etc.

6.3.1.2 Baja visión

Son alteraciones oculares y funcionales severas. Las personas que las padecen no se incluyen dentro del grupo de los videntes ni dentro del grupo de las personas ciegas. La condición de baja visión implica una extensa gama de posibilidades visuales que afectan sobre todo a la visión de los colores, del contraste, del campo visual y de la agudeza visual, entre otras.

Algunas personas con este tipo de problemas utilizan pantallas muy grandes o necesitan cambiar el tamaño de letra e imágenes. Otras emplean ampliadores de pantalla, software que amplía una parte de la pantalla para facilitar su visualización, pero que a veces, al visualizar imágenes, por ejemplo y hacerlas más grandes, se reduce la extensión del área visible del documento, dificultando la visión del contorno textual. Algunos ampliadores proporcionan dos vistas: una ampliada y otra con el tamaño por defecto de navegación.

Otras personas utilizan combinaciones específicas de colores y de fondo o eligen un determinado tipo de letra adecuado a sus dificultades visuales. Como se puede ver, existen muchas clases de baja visión y cada una de ellas tiene un tipo de barreras diferentes que le impiden ver la página con normalidad.

Barreras

- Sitios web que al ser ampliados son de difícil navegación.
- Páginas con colores de poco contraste que no se puede cambiar.
- Desactivación de las imágenes de fondo o necesidad de ver las páginas sin ningún tipo de estilo.
- Dificultad de comprensión de los sistemas multimedia. Son sistemas que utilizan más de un medio de comunicación en la presentación de la información, como texto, imágenes, videos o sonidos.
- Muchas de las barreras descritas para la ceguera dependiendo del tipo de deficiencia y de su grado.

Cambios a realizar

- Proporcionar opciones de cambio de contraste de los colores que definen la letra y el fondo de la página con el suficiente contraste para ser percibidas por personas con deficiencias en la percepción del color o en pantallas en blanco y negro.
- Opción de poder ver la página sin aplicarle ningún estilo.
- Proporcionar a los sistemas multimedia subtítulos o descripciones que mejoren su comprensión.

6.3.1.3 Daltonismo

Falta de sensibilidad a ciertos colores, es decir, no distinguir entre dos colores como pueden ser el rojo y el verde o el azul y el amarillo. Según el grado de daltonismo también puede ser que la persona no perciba ningún color.

Barreras

- Color de texto con insuficiente contraste con el color de fondo.
- No poder ver la página web sin aplicarle ningún estilo.
- Colores para dar énfasis al texto como único indicador, ya que si no se diferencian ciertos colores no se aprecia que se está ante palabras o citas importantes.

Cambios a realizar

- Proporcionar opciones de cambio de contraste de los colores que definen la letra y el fondo de la página con el suficiente contraste para ser percibidas por personas con deficiencias en la percepción del color o en pantallas en blanco y negro.
- Opción de poder ver la página sin aplicarle ningún estilo.
- A parte de dar color al texto como modo de indicador, se tendría que proporcionar algún otro tipo de identificador, como el cambio de tamaño de la letra o del estilo u otro cambio de estilo como subrayado o negrita, para una mayor apreciación.

6.3.2 Discapacidades auditivas

6.3.2.1 Sordera

La sordera es la pérdida de la audición total o parcial incorregible de un oído o de ambos. Una persona sorda será incapaz o tendrá problemas para escuchar.

Barreras

- Sitios web que requieren entrada de voz. No sería nuestro caso.
- Falta de imágenes en páginas con mucho texto que pueden ralentizar la lectura a personas que utilicen lengua de signos para comunicarse.

Cambios a realizar

- Proporcionar subtítulos o descripciones que mejoren la comprensión de los contenidos sonoros.

6.3.2.2 Dureza de oído

Se definen las personas duras de oído como aquellas que tienen sordera ligera o moderada.

Barreras

- Falta de subtítulos o transcripciones de sonido.

Cambios a realizar

- Proporcionar subtítulos o descripciones que mejoren su comprensión a los contenidos sonoros.

6.3.3 Discapacidades físicas

6.3.3.1 Discapacidades motrices

Incluyen debilidad, limitaciones de control muscular como movimientos involuntarios, falta de coordinación o parálisis, limitaciones de sensibilidad, problemas con las articulaciones o incluso falta de miembros. Algunos tipos de discapacidades incluyen dolor que imposibilita el movimiento, que puede ser de manos, brazos, o de otras partes del cuerpo.

Si las discapacidades afectan a las manos o brazos, este tipo de personas pueden tener



dificultades o imposibilidad de utilizar ratones y/o teclados, entonces, suelen utilizar unos ratones especiales. También disponen de teclados con una distribución de las teclas diferente a los teclados normales, adaptadas a su movimiento. Hay personas que necesitan un ratón de cabeza, ratón especial que permite el guiado del cursor con ligeros movimientos de cabeza. Está indicado para personas que tengan dificultades con la utilización de un teclado normal, como personas con Parkinson, ELA, distrofia muscular, parálisis cerebral, tetraplejia, etc.

Otro tipo de personas necesita de un licornio, un casco con una varilla en el que en uno de sus extremos puede incorporarse un lápiz o un pequeño puntero, diferentes accesorios para poder teclear en el ordenador, pintar, etc. Lo utilizarían aquellas personas que tienen afectadas las extremidades superiores pero que conservan un adecuado control de los músculos que dirigen la cabeza. Un apuntador bucal sería un objeto similar al licornio pero diseñado para sujetarlo con la boca.



También pueden necesitar software de reconocimiento de voz o sistemas de seguimiento de ojos, un servicio que permite el total control del ordenador por el movimiento de los ojos. Esto sustituye al ratón. Iriscom permite al usuario colocar el puntero del ratón en cualquier lugar de la pantalla del ordenador simplemente mirando a ese punto.

Iriscom proporciona al usuario acceso completo a todas las funciones de Windows y a los nuevos sistemas Mc Intosh.

Barreras

- Páginas con restricción de tiempo, ya que pueden necesitar más tiempo de lo normal para acciones como rellenar un formulario, por ejemplo. No es nuestro caso.
- Navegadores y herramientas de edición que no soporten alternativas de teclado para comandos de ratón, ya que con los apuntadores de cabeza pueden activar comandos mediante combinaciones de teclas simultáneas o secuencias de teclas individuales.
- Formularios que no permiten su navegación con el tabulador o que no lo hacen en el orden correcto.

Cambios a realizar

- Realizar formularios accesibles ya que, entre otras prestaciones, permiten un orden lógico de tabulación.

6.3.4 Discapacidades cognitivas y neurológicas

6.3.4.1 Dislexia y discalculia

La dislexia es una deficiencia relacionada con la lectura, la escritura y el aprendizaje. Su causa es una alteración de las zonas cerebrales que controlan el lenguaje. Es un trastorno específico del aprendizaje. Estas personas pueden tener dificultades con el lenguaje escrito, invirtiendo letras, números, sílabas o palabras. También presentan dificultades en la pronunciación de palabras, invirtiendo sílabas o sustituyendo palabras por otras, así como dificultad en el aprendizaje y problemas de concentración.

La discalculia es similar. Se diferencian en que ésta solo hace referencia a la falta de aprendizaje en el ámbito matemático, como la confusión de números, la mala interpretación de cantidades en operaciones o en mecanismos matemáticos y aritméticos.

Personas con estos problemas pueden requerir de lectores de pantalla con síntesis de voz para facilitar la comprensión del texto. Si el trastorno es de interpretación de sonidos, de procesar lo escuchado, entonces, estas personas requieren de subtítulos para facilitar la comprensión de un video, por ejemplo.

Barreras

- Falta de un texto alternativo que se pueda convertir en sonido para complementar la información visual.

Cambios a realizar

- Proporcionar un texto alternativo o equivalente a todo elemento no textual.

6.3.4.2 Trastorno de déficit de atención

Las personas con trastorno de déficit de atención pueden tener problemas para concentrarse en la información.

Barreras

- Elementos visuales o sonoros que no se pueden desactivar fácilmente, ya que las personas que padecen este trastorno pueden necesitar desactivar las animaciones para mejorar su concentración en el contenido del sitio.
- Falta de organización clara y regular de los sitios.

Cambios a realizar

- Opción de desactivar las animaciones que aparecen en el sitio web.
- Organización clara y regular de la información del sitio.

6.3.4.3 Discapacidad cognitiva

Alteración neurológica que determina el nivel cognitivo y funcional de la persona, que se caracteriza por una disminución de la capacidad de comprensión o de cognición.

Barreras

- Uso de lenguaje complejo sin necesidad.
- Falta de gráficos en los sitios web, ya que estas personas puede que necesiten gráficos que les ayude a la comprensión del sitio.
- Falta de una organización del contenido de la web clara y coherente.
- Algunas personas con este tipo de discapacidad puede que no sean capaces de comprender el tipo de información que debe ir en un formulario.

Cambios a realizar

- Utilizar el lenguaje apropiado más claro y simple para el contenido de un sitio web.
- Organización clara y regular de la información del sitio web.
- Se debe colocar información distintiva al comienzo de títulos de sección, párrafos, listas, etc.
- Se deben incluir valores por omisión en los controles vacíos de los formularios.

6.3.5 Falta de memoria

Este tipo de personas pueden presentar falta de memoria a corto plazo, a largo plazo o incluso, alguna pérdida de habilidad en el lenguaje.

Barreras

- Falta de organización coherente de un sitio web. Este tipo de personas puede depender de una estructura coherente de navegación por el sitio web.

Cambios a realizar

- Organización de la información de forma coherente, que facilite la comprensión de los usuarios y que siga un orden de tabulación adecuado.

6.3.4.1 Discapacidades de salud mental

Las personas con estos tipos de discapacidad pueden tener problemas de concentración con la información de un sitio web. A parte, pueden sufrir efectos secundarios producidos por los medicamentos, como temblor de manos o visión borrosa, efectos que empeorarán su acceso a la información. Por eso, muchas personas con estos trastornos pueden necesitar ampliadores de pantalla.

Barreras

- Elementos visuales o sonoros que no se puedan desactivar, ya que estas personas, a veces necesitan desactivarlos para mejorar su concentración.

Cambios a realizar

- Opción de desactivar las animaciones que aparecen en el sitio web.

6.3.5.1 Trastornos de epilepsia

La epilepsia es un trastorno neurológico crónico caracterizado por una generación de crisis epilépticas que suelen ser transitorias con manifestaciones, como movimientos convulsivos y que pueden llevar asociado pérdidas de conciencia.

Los usuarios con epilepsia foto-sensitiva pueden tener ataques desencadenados por parpadeos o destellos que oscilen entre los 4 y los 49 destellos por segundo (Hz), con un nivel máximo de 20 destellos por segundo, así como los cambios bruscos de oscuridad a iluminación, como las luces estroboscópicas. Este parpadeo hace referencia a toda la pantalla o a la ventana.

Barreras

- El uso de determinadas frecuencias visuales o sonidos puede provocar ataques epilépticos.

Cambios a realizar

- Evitar provocar destellos en pantalla.
- Poder desactivar animaciones o ciertas frecuencias de sonidos.

6.3.6 Discapacidades múltiples

Hay personas que sufren la combinación de diferentes discapacidades. Esto, les dificulta todavía más el acceso a la información. También merma la flexibilidad de la página web. Entonces, hay que tener en cuenta los posibles casos de discapacidades y sus posibles combinaciones para tener un sitio web lo más accesible posible. Las barreras que se encontrarían estas personas ya han sido explicadas anteriormente, y serían una combinación de muchas de las nombradas en cada caso. Las soluciones serían las

mismas soluciones planteadas anteriormente, asociadas a cada una de las barreras nombradas.

6.3.7 Condiciones relacionadas con el envejecimiento.

Los cambios de la capacidad funcional producidos en los usuarios debido al envejecimiento pueden influir en una reducción de las capacidades auditivas, así como de la visión, la memoria o la destreza. Esta reducción de las capacidades puede verse agravada con el tiempo. Las barreras con las que se pueden encontrar estas personas ya han sido mencionadas anteriormente y sus soluciones han sido descritas.

6.4 Medidas tomadas en la eliminación de las barreras

6.4.1 Medidas generales

6.4.1.1 Prioridad 1

1 Punto de verificación 1.1 (Requisito 4.4.3 de la Norma UNE 139803:2004)

“Proporcionar texto equivalente para todo elemento no textual”

Mediante el atributo **alt** se asigna un texto equivalente a cualquier elemento no textual. Los elementos no textuales incluyen imágenes, animaciones, botones gráficos, etc. Si el elemento textual no proporciona información importante, se puede definir un texto equivalente nulo (“”).

El atributo **longdesc** se utiliza para describir una imagen. Este atributo deriva a otra página web donde está la descripción. Algunas aplicaciones de usuario no manejan este atributo, por lo que se recomienda adjuntar a la imagen un enlace con el texto “D” que apunte a la página donde se encuentra la información detallada de la imagen.

2 Punto de verificación 2.1 (Requisito 4.2.2 de la Norma UNE 139803:2004)

“Toda la información transmitida a través de colores, también ha de estar disponible sin color”

Para llevar a cabo este punto de verificación, se ha creado un link en la página web que aplica unas hojas de estilo CSS diferentes. La página web, inicialmente, se verá con las letras oscuras y el fondo claro.



Figura 9. Link Cambio de Contraste.

Si se quiere ver la página con contraste, hay que pulsar el link para aplicar el contraste de colores: fondo oscuro y letras blancas.



Figura 10. Cabecera con cambio de contraste aplicado

Para realizar estos cambios se han tenido que crear nuevos CSS y a parte, se ha tenido que modificar el código.

3 Punto de verificación 4.1 (Requisito 4.4.5 de la Norma UNE 139803:2004)

“Identifique claramente los cambios en el idioma del texto del documento y en cualquier texto equivalente (por ejemplo, leyendas).”

Se identificará el idioma mediante el atributo **lang**.

4 Punto de verificación 6.1 (Requisito 4.2.1 de la Norma UNE 139803:2004)

“Organizar el documento para que pueda ser leído sin hoja de estilo”

Se ha creado un link que al ser pulsado se puede ver la página sin ningún estilo aplicado. Para la realización de este punto de verificación, se ha tenido que modificar el código PHP.



Figura 11. Link Sin estilo.

5 Punto de verificación 7.1 (Requisito 4.2.3 de la Norma UNE 139803:2004)

“Hasta que las aplicaciones de usuario permitan controlarlo, evitar provocar destellos de pantalla”

6 Punto de verificación 14.1 (Requisito 4.4.2 de la Norma UNE 139803:2004)

“Utilizar el lenguaje apropiado más claro y simple para el contenido del sitio”

Todo el contenido Web de las páginas tiene un orden lógico. Se utilizan etiquetas del tipo <h1> para marcar títulos, etiquetas <p> para párrafos, etc.

6.4.1.2 Prioridad 2

1 Punto de verificación 2.2 (Requisito 4.2.7 de la Norma UNE 139803:2004)

“Asegurar que las combinaciones de los colores de fondo y primer plano tengan el suficiente contraste para que sean percibidas por personas con deficiencias de percepción de color en pantallas en blanco y negro.”

Según la norma UNE, el W3C ha creado dos fórmulas para facilitar la comparación de contraste entre colores:

- Brillo de color:

$$((\text{Valor rojo} \times 299) + (\text{valor verde} \times 587) + (\text{valor azul} \times 114)) / 1000$$

La diferencia entre el brillo del color del fondo y el brillo del color del primer plano debe ser superior a 125.

- Diferencia de color:

$$\begin{aligned} & (\text{Máximo (valor rojo 1, valor rojo 2)} - (\text{mínimo (valor rojo 1, valor rojo 2)}) + \\ & (\text{máximo (valor verde 1, valor verde 2)} - \text{mínimo (valor verde 1, valor verde 2)}) + \\ & (\text{máximo (valor azul 1, valor azul 2)} - \text{mínimo (valor azul 1, valor azul 2)}) \end{aligned}$$

La diferencia entre el color de fondo y el color del primer plano debe ser superior a 500.

En este proyecto no se han utilizado estas fórmulas para validar el contraste de colores. Los colores aplicados han sido validados previamente con el programa Contrast-A para asegurar que la combinación elegida es accesible.

2 Punto de verificación 3.2 (Requisito 4.1.2 de la Norma UNE 139803:2004)

“Cree documentos que estén validados por las gramáticas formales publicadas.”

Todas las páginas de la Web han sido validadas a nivel de CSS, XHTML y Accesibilidad Web por los validadores del W3C.

3 Punto de verificación 3.3 (Requisito 4.2.5 de la Norma UNE 139803:2004)

“Utilización de hojas de estilo para la maquetación. “

Toda la web ha sido maquetada con hojas de estilo CSS. Estos estilos han tenido que ser modificados en su mayoría para poder realizar los cambios pertinentes. Se han creado hojas de estilo nuevas porque han sido requeridas, como por ejemplo, para el cambio de contraste de colores.

4 Punto de verificación 3.4 (Requisito 4.2.4 de la Norma UNE 139803:2004)

“Utilización de unidades relativas en lugar de absolutas al especificar los marcadores de lenguaje y en las hojas de estilo.”

Existen dos tipos de unidades de medida de los elementos HTML: unidades absolutas y unidades relativas. Siempre se recomienda trabajar con unidades relativas. En el caso de esta web se ha trabajado con la unidad **em**, unidad relativa que es proporcional al tamaño de letra utilizado, así que si se aumenta la fuente se aumentan las distancias

proporcionalmente. En cambio, si se utiliza otra unidad, como el píxel, que a pesar de ser una unidad de medida relativa depende de la resolución de pantalla y, a no ser que se esté diseñando un documento muy específico en el que ya se conozca el dispositivo en el que será visualizado, no es recomendable utilizarlo. En el caso de la accesibilidad web, son muchos y muy diferentes los dispositivos que acceden a ella.

Según la norma UNE pueden utilizarse medidas fijas como puntos o milímetros en las hojas de estilo para la impresión de las páginas web. Pero en atributos como Font-size de los elementos CSS se deben utilizar unidades relativas como **em**.

5 Punto de verificación 3.5 (Requisito 4.3.5 de la Norma UNE 139803:2004)

“Utilice elementos de encabezado para transmitir la estructura lógica y utilícelos de acuerdo con la especificación.”

Se han marcado las páginas correctamente con la utilización de los encabezados de <h1> a <h6>. Entre encabezados no puede existir una diferencia superior a un orden.

6 Punto de verificación 3.6 (Requisito 4.3.7 de la Norma UNE 139803:2004)

“Marcar correctamente las listas y los ítems de las listas.”

Se han marcado las listas mediante el atributo cuando no se requerían listas ordenadas. Para listas ordenadas se utiliza el atributo . Dentro de cada una de las listas se han marcado los ítems mediante el atributo .

7 Punto de verificación 7.2 (Requisito 4.2.10 de la Norma UNE 139803:2004)

“Hasta que las aplicaciones de usuario permitan controlarlo, evitar el parpadeo del contenido.”

8 Punto de verificación 7.4 (Requisito 4.5.4 de la Norma UNE 139803:2004)

“Hasta que las aplicaciones de usuario proporcionen la posibilidad de detener las actualizaciones, no crear páginas que se actualicen automáticamente de forma periódica.”

9 Punto de verificación 11.1 (Requisito 4.1.1 de la Norma UNE 139803:2004)

“Utilice tecnologías W3C cuando estén disponibles y sean apropiadas para la tarea y las últimas versiones que sean soportadas.”

Se ha seguido la WCAG 1.0 para el cumplimiento de la nueva Ley y se ha tenido en cuenta la última versión, la 2.0.

10 Punto de verificación 11.2 (Requisito 4.1.3 de la Norma UNE 139803:2004)

“Evitar características desaconsejadas por las tecnologías W3C.”

No se han utilizado elementos desaconsejados por el W3C.

11 Punto de verificación 12.3 (Requisito 4.3.6 de la Norma UNE 139803:2004)

“Divida los bloques largos de información en grupos más manejables cuando sea natural y apropiado.”

Se han dividido los bloques largos de información en formularios y tablas utilizando etiquetas tales como `<caption>` o `<fieldset>`. Estas se explicaran más adelante.

12 Punto de verificación 13.1

“Identifique claramente el objetivo de cada vínculo”.

Todos los links o vínculos han sido nombrados adecuadamente, consiguiendo que el usuario sepa en cada momento dónde va dirigido.

13 Punto de verificación 13.2 (Requisito 4.1.4 de la Norma UNE 139803:2004)

“Proporcione metadatos para añadir información semántica a las páginas y sitios.”

Los metadatos proporcionan al usuario información sobre el sitio: autor, título, lenguaje, descripción, etc. Estos se incluyen añadiendo dentro de la etiqueta `<head>` el atributo **meta**. Las etiquetas más utilizadas son *Keywords* (palabras clave), *author*, *language*, *description* o *robots* (proporciona instrucciones a los rastreadores sobre cómo rastrear o indexar).

En Joomla! existen dos tipos de metadatos: uno general para toda la web que proporciona el nombre y el título de ésta y que es generado automáticamente. El segundo es propio de cada página y ha de ser introducido manualmente.

14 Punto de verificación 13.4 (Requisito 4.5.8 de la Norma UNE 139803:2004)

“Utilización de los mecanismos de navegación de forma coherente.”

La maquetación coherente de páginas, los gráficos reconocibles y el lenguaje fácilmente comprensible benefician a todos los usuarios. En particular, ayudan a personas con discapacidades cognitivas o con dificultades en la lectura.

6.4.1.3 Prioridad 3

1 Punto de verificación 4.2 (Requisito 4.4.10 de la Norma UNE 139803:2004)

“Especifique la expansión de cada abreviatura o acrónimo cuando aparezcan por primera vez en el documento.”

La primera vez que aparezca una abreviatura en un documento, se añadirá el atributo **title** con el nombre completo. La utilización de **abbr** se explicará más adelante, en el punto de tablas.

En el Anexo 3, se muestran cada una de las abreviaturas que aparecen en los encabezados de las tablas que forman la Web.

2 Punto de verificación 4.3

“Identifique el idioma principal de un documento.”

Se identificará el idioma mediante el atributo **lang**.

3 Punto de verificación 9.4

“Creación de un orden lógico de tabulación.”

El orden de tabulación permite tener la información ordenada, que será interpretada por los lectores de pantalla. Este orden ha de estar también en tablas y formularios. Con el atributo **tabindex** se asigna el orden.

4 Punto de verificación 9.5 (Requisito 4.5.15 de la Norma UNE 139803:2004)

“Proporcionar atajos de teclado para los vínculos más importantes.”

A través del atributo **accesskey** se consigue proporcionar un atajo de teclado. Se han introducido atajos de teclado en los vínculos más importantes. Estos atajos se explican en la web haciendo “click” en el link de Accesibilidad que aparece en la cabecera.

En el Anexo 4, se muestran los atajos de teclado a los enlaces más comunes de la web.

5 Punto de verificación 10.5 (Requisito 4.5.14 de la Norma UNE 139803:2004)

“Hasta que las aplicaciones de usuario interpreten claramente los vínculos contiguos, incluya caracteres imprimibles (rodeados de espacios), que no sirvan como vínculo, entre los vínculos contiguos”

Cuando se tienen links próximos entre si, se ha incluido entre link y link una separación visible por el usuario.

6 Punto de verificación 13.5 (Requisito 4.5.11 de la Norma UNE 139803:2004)

“Proporcionar barras de navegación para destacar y dar acceso al mecanismo de navegación.”

7 Punto de verificación 13.8 (Requisito 4.4.9 de la Norma UNE 139803:2004)

“Localice la información destacada al principio de los encabezamientos, párrafos, listas, etc.”

8 Punto de verificación 14.3 (Requisito 4.2.12 de la Norma UNE 139803:2004)

“Crear un estilo de presentación coherente para todas las páginas.”

6.5 Tablas

En este punto se nombran y detallan las pautas de accesibilidad que se han aplicado en el diseño de todas y cada una de las tablas de la Web e Intranet. Una vez nombradas las pautas, se explicará mediante un ejemplo la forma de crear una tabla accesible, de esta forma se podrá interpretar mejor cada una de las pautas nombradas. En el Anexo 1, se muestra el manual sobre tablas accesibles.

6.5.1 Prioridades

- **Prioridad 1**

1 Punto de Verificación 5.1 (Requisito 4.3.2 de la Norma UNE 139803:2004)

“En las tablas de datos, identificar los encabezamientos de fila y columna.”

2 Punto de verificación 5.2 (Requisito 4.3.3 de la Norma UNE 139803:2004)

“Para las tablas de datos que tienen dos o más niveles lógicos de encabezamientos de fila o columna, utilizar marcadores para asociar las celdas de encabezamiento y las celdas de datos.”

- **Prioridad 2**

1 Punto de verificación 5.3 (Requisito 4.2.8 de la Norma UNE 139803:2004)

“No utilizar tablas para maquetar, a menos que la tabla tenga sentido cuando se alinee.”

2 Punto de verificación 5.4 (Requisito 4.2.9 de la Norma UNE 139803:2004)

“Si se utiliza una tabla para maquetar, no utilizar marcadores estructurales para realizar un efecto visual de formato.”

- **Prioridad 3**

1 Punto de verificación 5.5

“Proporcionar resúmenes a las tablas”

2 Punto de verificación 5.6

“Proporcionar abreviaturas para las etiquetas de encabezado.”

3 Punto de verificación 10.3 (Requisito 4.2.13 de la Norma UNE 139803:2004)

“Hasta que las aplicaciones de usuario interpreten correctamente los textos contiguos, proporcionar un texto lineal alternativo para todas las tablas que maquetan texto en paralelo, en columnas de palabras.”

6.5.2 Creación de tablas accesibles.

- Estructura básica de una tabla.

```
<table width="451" height="300" border="1">
  <tr>
    <th>Primera columna de la primera fila </th>
    <th>Segunda columna de la primera fila </th>
    ...
  </tr>

  <tr align="right" bgcolor="#FFFF99">
    <td bordercolor="#6699FF">Primera columna de la segunda fila
</td>
    <td>Segunda columna de la segunda fila </td>
    <td >Tercera columna de la segunda fila </td>
    ...
  </tr>

  <tr height="150">
<td> Primera columna de la tercera fila </td>
    <td>Segunda columna de la tercer fila </td>
    ...
  </tr>
</table>
```

A continuación se explicarán las etiquetas básicas que forman una tabla:

<table> y </table>

Esta etiqueta define donde empieza y acaba la tabla. Entre estas dos etiquetas se han de poner todos los atributos y etiquetas que la forman.

<tr> y <td>

Corresponden a las filas y las columnas respectivamente. Cuando se quiera poner un encabezado se sustituirá la etiqueta **<td>** por **<th>**. Se podrá observar que el contenido del encabezado aparecerá en negrita y centrado en la celda. Se irán añadiendo tantos **<tr>** y **<td>** como filas o columnas se quiera.

Mediante diferentes atributos se pueden modificar total o parcialmente las propiedades de la tabla, como pueden ser las dimensiones, color, bordes, etc. A continuación se explicarán algunos de ellos, concretamente, los utilizados en el ejemplo:

- **width y height**

Estos dos atributos permiten modificar la anchura y la altura de la tabla o de una parte de ella.

- **border**

Cuando se quiera modificar el borde de la tabla o parte de este se utilizará el atributo **border**. También se podrán modificar celdas, filas y columnas.

Mediante el atributo **bordercolor** también se podrá cambiar el color del borde escogido.

- **align**

Cambia el alineado del texto para toda la tabla o para una parte de ella. El atributo **align** puede tomar los siguientes valores: **center** (centro), **justify** (justificado), **left** (izquierda) o **right** (derecha).

- **bgcolor**

Este atributo define un color de fondo.

Con todos los atributos definidos anteriormente y otros muchos otros existentes, se consigue modelar las tablas.

El código de ejemplo utilizado anteriormente para la creación de la estructura básica de una tabla, consiste en una tabla formada por tres filas y cinco columnas. La siguiente imagen muestra los cambios de estilo explicados.

| | | | | |
|------------------------------------|------------------------------------|------------------------------------|------------------------------------|-----------------------------------|
| Primera columna de la primera fila | Segunda columna de la primera fila | Tercera columna de la primera fila | Cuarta columna de la primera fila | Quinta columna de la primera fila |
| Primera columna de la segunda fila | Segunda columna de la segunda fila | Tercera columna de la segunda fila | Cuarta columna de la segunda fila | Quinta columna de la segunda fila |
| Primera columna de la tercera fila | Segunda columna de la tercer fila | Tercera columna de la tercera fila | Cuarta columna de la tercer a fila | Quinta columna de la tercera fila |

Ejemplo 1. Tabla.

- **Estructura básica de una tabla accesible.**

A continuación se observa la estructura de una tabla accesible para posteriormente explicar los atributos y etiquetas que la forman.

En el Anexo 1 se muestra el manual de Tablas Accesibles.

```
<table title="Título que será mostrado" summary="Resumen del
contenido de la tabla">
  <caption> Título de la tabla </caption>

  <thead>
    <tr>
      <th abbr="abreviatura" scope="col">Título de columna
1</th>
      <th scope="col">Título de columna 2</th>
      ...
    </tr>
  </thead>

  <tfoot>
    <tr>
      <th scope="row">PIE 1</th>
      <th>PIE 2</th>
      ...
    </tfoot>

  <tbody>
    <tr>
      <td scope="row">1ª celda de fila 1</td>
      <td>2ª celda de fila 1</td>
      ...
    </tr>
    <tr>
      <td scope="row">1ª celda de fila 2</td>
      <td>2ª celda de fila 2</td>
      ...
    </tr>
    <tr> ... </tr>
  </tbody>

</table>
```

Seguidamente, se explicarán las etiquetas que hacen que la tabla sea accesible y las pautas a seguir para conseguir su accesibilidad.

Anteriormente, se ha visto que en una tabla se podían modificar sus propiedades mediante una serie de atributos. Si se quiere que la tabla sea accesible, la modificación de estas propiedades se hará mediante hojas de estilo CSS, es decir, *no se aplicará ningún cambio de estilo a las tablas accesibles mediante atributos*. Esto quiere decir que si en una celda queremos que la letra aparezca en negrita, por ejemplo, sin ser esa celda el encabezado de fila o columna, no se debe utilizar th para darle estilo a la celda, sino que se deberá hacer mediante otros mecanismos como la utilización de hojas de estilo.

<title> y </title>

Proporcionan un texto que será mostrado cuando el “mouse” se sitúe encima de la tabla.

Summary

Proporciona un resumen sobre el contenido de la tabla, su relación con el documento, la relación entre las celdas u otras relaciones que no son evidentes a la hora de hacer un análisis estructural de la tabla, pero que a simple vista sí lo son. Estos resúmenes son muy útiles para los software lectores de voz y muy necesarios si no se proporciona un título.

<caption> y </caption>

Contienen el título de la tabla, aportando información sobre su contenido.

Las tablas accesibles se dividen en tres partes: header o cabecera, body o cuerpo y foot o pie. Puede ser que no todas las tablas contengan las tres partes, ya que pueden carecer de alguna de ellas.

La etiqueta **<thead>** define el comienzo de la cabecera de la tabla. Y **</thead>** el final. El contenido de la celda aparecerá en negrita y centrado. Dentro de esta etiqueta, se definirán las filas con **<tr>** y las celdas, pertenecientes a las columnas, con **<th>** que sustituye a **<td>**. Cuando en la cabecera se tengan títulos que ocupen más de 15 caracteres, se utilizará el atributo **abbr**. En él se definirá la abreviatura del título, la cual, no puede contener más de 15 caracteres.

Mediante la etiqueta **<tfoot>** se identifica el pie de tabla. Tiene la misma estructura que **<thead>**: para definir las filas se utilizará **<tr>** y para las columnas **<th>**. Dentro de **<tbody>** estará el cuerpo de la tabla donde se definirán las filas y columnas de manera tradicional, con **<tr>** y **<td>**. Esta etiqueta se debe acompañar del atributo **scope**, que más adelante se detallará.

Aunque la tabla siga una estructura de cabecera, cuerpo y pie de tabla, a la hora de escribir el código no se seguirá ese orden sino el siguiente: *thead*, *tfoot* y *body*. Siempre habrá que ordenarlo de la misma manera.

Existen atributos que vinculan la información que contienen las celdas que forman la tabla: *scope*, *axis* y *headers*. La elección entre utilizar *scope* o *header* no afecta al resultado, siempre y cuando la relación entre el encabezado y las celdas de datos sea correcta, la elección dependerá de la complejidad de la tabla. En este caso, como es una tabla sencilla se utilizará **scope**. Si la tabla fuera más compleja, se utilizaría **header**.

El atributo **scope** se utiliza solamente con las etiquetas **<th>** y **<td>**. Tiene cuatro posibles valores: **row** (celda referida al inicio de una fila), **col** (celda referida al inicio de una columna), **rowgroup** (celda referida al inicio de un conjunto de filas) y **colgroup** (celda referida al inicio de un conjunto de columnas).

Seguidamente, se puede observar la tabla que correspondería al ejemplo anterior, habiéndole añadido previamente filas y columnas.

Tabla de prueba

| Título de columna 1 | Título de columna 2 | Título de columna 3 |
|---------------------|---------------------|---------------------|
| 1ª celda de fila 1 | 2ª celda de fila 1 | 3ª celda de fila 1 |
| 1ª celda de fila 2 | 2ª celda de fila 2 | 3ª celda de fila 2 |
| 1ª celda de fila 3 | 2ª celda de fila 3 | 3ª celda de fila 3 |
| 1ª celda de fila 4 | 2ª celda de fila 4 | 3ª celda de fila 4 |
| PIE 1 | PIE 2 | PIE 3 |

Ejemplo 2. Tabla accesible.

En general, todo lo explicado anteriormente quedaría plasmado en los 4 puntos de accesibilidad que a continuación se mencionan:

- No aplicar estilos a las tablas. Los estilos se han de aplicar mediante hojas de estilo CSS.
- Proporcionar un resumen de la tabla (*summary*), abreviaturas para las etiquetas del encabezado (*abbr*), texto que muestre información de la tabla (*title*) y un título (*caption*).
- Identificar encabezados de fila y columna (*thead*, *th*), grupos estructurales (*thead*, *tbody* y *tfoot*) y etiquetar elementos de tabla (*scope*, *headers* y *axis*).
- No utilizar tablas para maquetar.

Estas pautas de accesibilidad Web pueden estar sometidas a variaciones y excepciones según el diseño y contenido de la página Web.

6.6 Formularios

A continuación, se nombran y detallan las pautas de accesibilidad que se han aplicado en el diseño de todos y cada uno de los formularios de la Web e Intranet. Una vez nombradas las pautas, se explicará mediante un ejemplo la forma de crear un formulario accesible, de esta forma se podrá interpretar mejor cada una de las pautas nombradas. En el Anexo 2, se muestra el manual sobre Formularios Accesibles.

• Prioridad 2

1 Punto de verificación 10.2 (Requisito 4.4.8 de la Norma UNE 139803:2004)

“Hasta que las aplicaciones de usuario soporten explícitamente la asociación entre control de formulario y etiqueta, para todos los controles de formularios con etiquetas asociadas implícitamente, asegurarse de que la etiqueta está colocada adecuadamente.”

2 Punto de verificación 12.4 (Requisito 4.4.7 de la Norma UNE 139803:2004)

“Asocie explícitamente las etiquetas con sus controles”.

- **Prioridad 3**

1 Punto de verificación 10.4 (Requisito 4.4.14 de la Norma UNE 139803:2004)

“Hasta que las aplicaciones de usuario manejen correctamente los controles vacíos, incluir caracteres por defecto en los cuadros de edición y áreas de texto.”

6.6.1 Creación de formularios accesibles

A continuación se explica la estructura básica de un formulario para posteriormente, poder explicar cómo crear formularios accesibles.

- **Estructura básica de un formulario.**

```
<FORM METHOD="post" ACTION="mis_datos.php">

<p>Tu nombre <input type="text" name="nombre" size=30"></p>
<p>Tu color preferido

<select size="1" name="color">
<option value="rojo">Rojo</option>
<option value="verde" selected="selected">Verde</option>
<option value="azul">Azul</option>
...
</select></p>

<p>¿Te gusta el deporte? <input type="checkbox" name="deporte"
value="ON"></p>
<p>¿Mar o montaña?</p>
<blockquote>
<p>Mar<input type="radio" value="mar" checked
name="preferencias"></p>
<p>Montaña <input type="radio" name="preferencias"
value="montaña"></p>
</blockquote>
<p>Aficiones</p>
<p><textarea rows="5" name="aficiones"
cols="28"></textarea></p>
<p><input type="submit" value="Enviar datos" name="enviar">
<input type="reset" value="Restablecer" name="B2"></p>

</FORM>
```

Los formularios forman parte del lenguaje de programación HTML, por tanto, todas las etiquetas y atributos usados corresponden a este lenguaje. Posteriormente, los formularios podrán ser tratados con PHP.

Todo formulario empieza y acaba con las etiquetas **<form>** y **</form>** respectivamente. Dentro de estas etiquetas es donde se especificarán todos los atributos que formarán parte de nuestro formulario.

La etiqueta **action** especifica el script PHP que va a tratar la información recogida en nuestro formulario.

Method indica la manera en que se van a enviar los datos del formulario. Puede tomar como valor *get* o *post*. **Get** envía los archivos o datos mediante url, esto quiere decir, que al recargar la página, nuestras variables aparecerán en la barra de direcciones. No es muy apropiado utilizar *get* en formularios, sobretodo, si se quiere enviar información confidencial, ya que se estarán mostrando todos los datos del envío. En cambio, **post** envía datos al servidor para ser procesados: envía las variables de forma “oculta” y además, no se tiene límite de envío de datos, cosa que con *get*, sí.

Cada campo del formulario ha de ir precedido por los atributos **<p>** y **</p>**: cada elemento irá en un párrafo diferente. Si se quiere que varios campos aparezcan alineados en la página, irán todos dentro del mismo párrafo.

Mediante la etiqueta **input** se definen los elementos de control del formulario. Se les dará un nombre a través del atributo **name** para que se pueda hacer referencia a ellos desde hojas de estilo o scripts. También se les asignará un valor, que será el que se envíe a través del formulario asignado por **value**. El tipo de control que se quiera definir depende de lo especificado en el atributo **type**. A continuación, se explicará cada uno de los valores que puede tomar *type*:

- **Text**

Crea un control de entrada de texto de una línea. Se pueden especificar los caracteres de los que constará esta línea de texto mediante el atributo **size**.

- **Password**

Crea un control de entrada de texto en el que los caracteres aparecen escondidos, por ejemplo, bajo asteriscos. Este tipo de control se utiliza normalmente cuando los datos son importantes, como contraseñas, para que no sean vistos por otros usuarios.

- **Checkbox**

Crea una casilla de verificación. Estas casillas son interruptores de encendido/apagado que pueden ser conmutados por el usuario. Para hacer que una casilla aparezca marcada previamente se utiliza el atributo **checked**. Este atributo solo es válido en casillas de verificación y radiobotones.

- **Radio**

Crea un *radiobotón*, botón circular seleccionable por el usuario.

- **Submit**

Crea un botón de envío de formulario. Un formulario puede contener más de un botón de envío.

- **Image**

Crea un botón de envío gráfico.

- **Reset**

Crea un botón de reinicialización. Borra los valores introducidos anteriormente.

- **Button**

Crea un botón pulsador.

- **Hidden**

Crea un botón oculto para el usuario cuyo valor será enviado con el formulario. Este tipo de botones suele usarse para almacenar información entre intercambios cliente/servidor, que se perdería, debido a la naturaleza no persistente del protocolo HTTP.

- **File**

Crea un control de selección de fichero. Permite al usuario enviar los contenidos de un fichero mediante un formulario.

La etiqueta **select** crea un menú desplegable. Con la etiqueta **<option>** se crea cada una de las opciones de menú. Para preseleccionar una opción de menú, que podrá ser cambiada por el usuario, se hará mediante el atributo **selected** y su valor ha de ser *selected*. Para los controles tipo radiobotón, se preseleccionará la opción mediante el atributo **checked**.

Textarea. Mediante esta etiqueta se inserta un campo de texto.

Con la etiqueta **<blockquote> </blockquote>** se sangra el texto. Se pueden insertar otras etiquetas que varíen el estilo del formulario para aproximarlos a nuestro gusto.

A continuación se puede ver el formulario creado en el ejemplo anterior:

Tu nombre

Tu color preferido ▼

¿Te gusta el deporte?

¿Mar o montaña?

Mar

Montaña

Aficiones

Ejemplo 3. Formulario.

- Estructura básica de un formulario accesible

```

<FORM METHOD="post" ACTION="mis_datos.php">
<fieldset>
  <legend>Datos personales</legend>
  <p>
    <label for="nombre">Nombre:</label>
    <input type="text" id="nombre" accesskey="n" tabindex="1"
size="30">
    <label for="primer-apellido">Primer apellido:</label>
    <input tabindex="2" accesskey="p" type="text" id="primer-
apellido"
      size="30">
    ...
  </p>
</fieldset>

<fieldset>
  <legend>Curiosidades</legend>
  <p>
    <label for="color-favorito">Tu color favorito:</label>
    <select tabindex="4" accesskey="c" size="1" id="color-favorito">
    <optgroup label="Azul">
      <option label="Azul cielo" value="azul cielo">Azul
cielo</option>
      <option label="Azul marino" value="azul marino">Azul
marino</option>
      <option label="Turquesa" value="Turquesa">Turquesa</option>
    </optgroup>
    <option value="Rojo">Rojo</option>
    <option value="Verde" selected="selected">Verde</option>
    </select>
  </p>
  <p>
    <label for="deporte">¿Te gusta el deporte ?</label>
    <input tabindex="5" accesskey="d" type="checkbox" id="deporte"
value="ON">
    <p>¿Mar o montaña?</p>
    <p>Mar <input tabindex="6" accesskey="m" type="radio" checked
name="preferencias" value="mar"></p>
    <p>Montaña <input tabindex="7" type="radio"
name="preferencias" value="montaña"></p>
    <p>
      <label for="aficiones">Aficiones</label>
    </p>
    <p><textarea tabindex="8" accesskey="a" rows="5" name="aficiones"
cols="28">teclea su nombre</textarea></p>
    <p><input tabindex="9" accesskey="e" type="submit" value="Enviar datos"
name="enviar">
    <input tabindex="10" accesskey="r" type="reset" value="Restablecer"
name="B2"></p>
</fieldset>

</FORM>

```

A continuación se detallarán cada una de las etiquetas y atributos que hacen que un formulario sea accesible.

La etiqueta **<label>** se asocia a un control de formulario. Cada control ha de llevar asociada explícitamente esta etiqueta. A través del atributo **for** se asocia un rótulo al control. El valor de este atributo ha de ser el mismo que el valor del atributo **id** del control asociado.

Estas etiquetas deben aparecer a la izquierda de los cuadros combinados, campos de edición de texto y listas de selección, y a la derecha de los botones de edición y cuadros de verificación, cuando la etiqueta y el control asociado van en la misma línea. Si la etiqueta va en una línea distinta de la del control, siempre debe aparecer en la línea inmediatamente anterior a él.

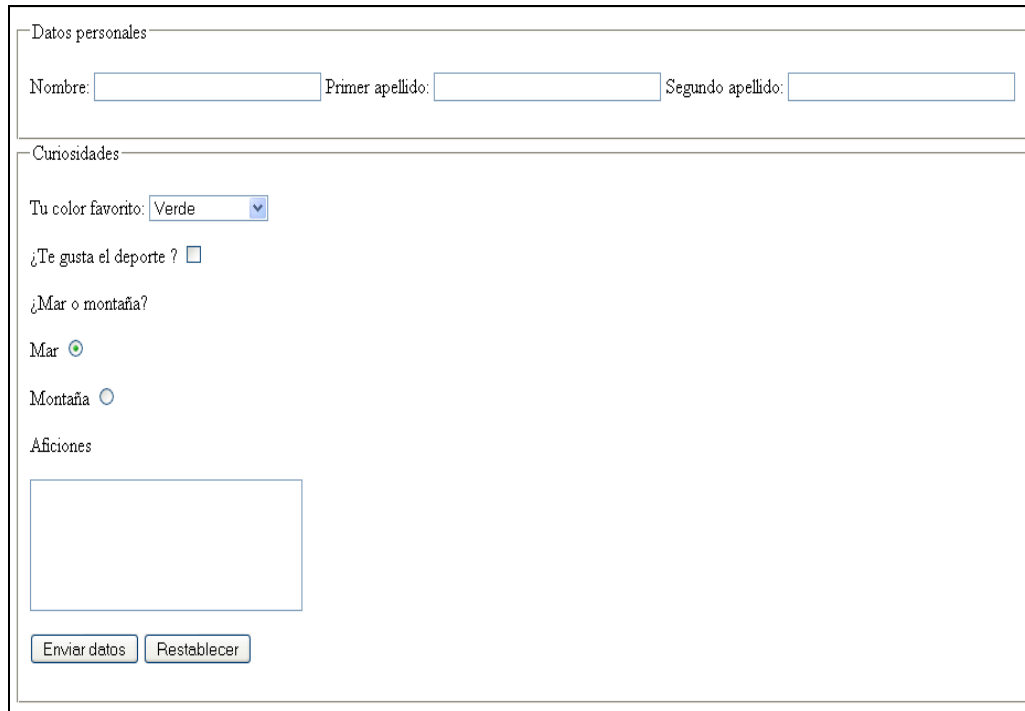
El atributo **tabindex** establece un orden lógico de tabulación. Cuando se recorre el formulario con el tabulador, se le puede asignar el orden de lectura.

Igual que sucede en las tablas, es necesario en los formularios accesibles proporcionar atajos de teclado. Mediante el atributo **accesskey** se consigue este fin. Según la versión del navegador y el sistema operativo que se tenga, las teclas de control del teclado serán diferentes. Por ejemplo, para Firefox 2.0 con sistema operativo Windows, se accederá al teclado presionando a la vez las teclas ALT y SHIFT con la letra asignada para ese campo. En cambio, si el navegador es Internet Explorer 5+ y el sistema operativo es Windows, se tendrá que presionar ALT a la vez que la letra asignada, soltar y presionar INTRO. Esta información debe estar presente en la web.

En el caso de tener formularios largos y difíciles de seguir, se utilizará el atributo **fieldset**. Este atributo agrupa los campos del formulario según la información que contengan ayudando a entender formularios largos y complejos. Otra propiedad de este atributo es hacer más accesible la navegación por voz. Mediante el atributo **legend** se dará un título a la agrupación formada por *fieldset*.

Cuando en el formulario exista un menú desplegable con una lista de información muy larga, es necesario, siempre y cuando sea apropiado, agrupar la información del menú. Para ello se utiliza la etiqueta **<optgroup>** que es la que especifica cada grupo de opciones. Cada una de las opciones que forma el grupo se identifica con la etiqueta **<option>**. El atributo **label**, que será usado en las dos etiquetas, nombra el campo. Este nombre suele ser más corto que el del campo que se describe dentro de la etiqueta *option*. En el caso del ejemplo, sería el mismo nombre. Un aspecto a tener en cuenta es que *label* ha de estar a la izquierda del campo que se describe, menos para radio buttons o check boxes donde puede ir a la derecha del campo.

A continuación se podrá ver el formulario explicado anteriormente:



The image shows a web form with two main sections. The first section, titled 'Datos personales', contains three text input fields labeled 'Nombre:', 'Primer apellido:', and 'Segundo apellido:'. The second section, titled 'Curiosidades', contains a dropdown menu for 'Tu color favorito:' with 'Verde' selected, a checkbox for '¿Te gusta el deporte?', and radio buttons for '¿Mar o montaña?' with 'Mar' selected. Below this is a text area for 'Aficiones' and two buttons: 'Enviar datos' and 'Restablecer'.

Ejemplo 4. Formulario accesible.

En general todo lo explicado anteriormente quedaría plasmado en los 5 puntos de accesibilidad que a continuación se detallan:

- Todos los campos de un formulario (*textarea*, *radio*, *checkbox*, etc.) deben tener asociados atajos de teclado (*accesskey*), un orden coherente de tabulación (*tabindex*), una agrupación de campos según lógica y temática, siempre que sea posible (*fieldset*) y una etiqueta que de nombre a cada control (*label*).
- Si se utilizan tablas dentro de un formulario, hacer que éstas sean accesibles.
- Se ha de proporcionar un texto equivalente a todo elemento no textual (*alt*, *longdesc*).
- No modificar el estilo del formulario mediante atributos. Se hará mediante hojas de estilos CSS.
- Se han de incluir, por defecto, caracteres en los cuadros de edición y áreas de texto.

6.7 WCAG 2.0

Las pautas de accesibilidad seguidas para hacer accesible esta web han sido las Pautas de Accesibilidad al Contenido en la Web 1.0 (WCAG 1.0) del W3C. El 31 de diciembre de 2008 se aprobó una nueva ley que obliga a los organismos públicos del estado español a cumplir como mínimo con la segunda prioridad (lo que implica cumplir la primera prioridad) de la WCAG 1.0.

Como ninguna ley española puede basarse en unas pautas que no hayan sido creadas por un organismo oficial de normalización, las WCAG 1.0 se tuvieron que reescribir en la Norma UNE 139803:2004. Se ha leído y aplicado la Norma y asociado con el punto de verificación del W3C.

El 11 de diciembre de 2008 el W3C aprobó las Pautas de Accesibilidad al Contenido en la Web 2.0 (WCAG 2.0), una mejora de las anteriores que consiste en 12 pautas y 60 puntos de verificación. Actualmente, estas pautas no se han incorporado a la legislación española, pero en la realización de este proyecto se han tenido en cuenta. Se han ido comprobando paralelamente los puntos que se cumplían. De todas formas, no ha sido posible validar estas pautas ya que no existe ningún validador para ellas. Se han validado las de la recomendación WCAG 1.0.

A continuación se nombrarán cada uno de los criterios de éxito que se cumplen total o parcialmente de la recomendación WCAG 2.0.

- **Principio 1**

- Pauta 1.1 → Criterio de éxito 1.1.1. Contenido no textual.
- Pauta 1.3. → Criterio de éxito 1.3.1. Información y sus relaciones.
- Pauta 1.3. → Criterio de éxito 1.3.2. Secuencia con significado.
- Pauta 1.3. → Criterio de éxito 1.3.3. Características sensoriales.
- Pauta 1.4. → Criterio de éxito 1.4.1. Uso del color.
- Pauta 1.4. → Criterio de éxito 1.4.2. Control del audio.
- Pauta 1.4. → Criterio de éxito 1.4.3. Contraste (mínimo).
- Pauta 1.4. → Criterio de éxito 1.4.4. Tamaño del texto.
- Pauta 1.4. → Criterio de éxito 1.4.6. Contraste (aumentado).
- Pauta 1.4. → Criterio de verificación 1.4.7. Bajo o sin sonido de fondo.
- Pauta 1.4. → Criterio de verificación 1.4.8. Presentación visual.
- Pauta 1.4. → Criterio de verificación 1.4.9. Imágenes de texto (sin excepción).

- **Principio 2.**

- Pauta 2.1. → Criterio de verificación 2.1.1. Teclado.
- Pauta 2.1. → Criterio de verificación 2.1.2. Teclado no bloqueado.
- Pauta 2.2. → Criterio de verificación 2.2.2. Pausar, parar, ocultar.
- Pauta 2.2. → Criterio de verificación 2.2.3. Sin tiempo

- Pauta 2.2. → Criterio de verificación 2.2.4. Interrupciones.
- Pauta 2.2. → Criterio de verificación 2.2.5. Re-autenticación.
- Pauta 2.3. → Criterio de verificación 2.3.1. Tres destellos (flashes) o debajo del umbral.
- Pauta 2.3. → Criterio de verificación 2.3.2. Tres destellos.
- Pauta 2.4. → Criterio de verificación 2.4.1. Accesos directos.
- Pauta 2.4. → Criterio de verificación 2.4.2. Título de la página.
- Pauta 2.4. → Criterio de verificación 2.4.3. Orden del foco.
- Pauta 2.4. → Criterio de verificación 2.4.4. Propósito de los enlaces (en su contexto).
- Pauta 2.4. → Criterio de verificación 2.4.6. Encabezados y etiquetas.
- Pauta 2.4. → Criterio de verificación 2.4.7. Visibilidad del foco.
- Pauta 2.4. → Criterio de verificación 2.4.9. Propósito de los enlaces (enlaces sin contexto).
- Pauta 2.4. → Criterio de verificación 2.4.10. Encabezados de sección.
- **Principio 3.**
 - Pauta 3.1. → Criterio de verificación 3.1.1. Idioma de la página.
 - Pauta 3.1. → Criterio de verificación 3.1.4. Abreviaturas.
 - Pauta 3.2. → Criterio de verificación 3.2.1. Foco.
 - Pauta 3.2. → Criterio de verificación 3.2.3. Navegación consistente.
 - Pauta 3.2.4. → Criterio de verificación Identificación consistente.
 - Pauta 3.3. → Criterio de verificación 3.3.1. Identificación de errores.
 - Pauta 3.3. → Criterio de verificación 3.3.3. Sugerencias de error.
 - Pauta 3.3. → Criterio de verificación 3.3.6. Prevención de errores (todos).
- **Principio 4.**
 - Pauta 4.1. → Criterio de verificación 4.1.1. Análisis.
 - Pauta 4.1. → Criterio de verificación 4.1.2. Nombre, función, valor.

6.8 Cambios en el código PHP

La página Web de la EPSEVG está formada por una gran cantidad de archivos, organizados en carpetas que contienen el código HTML y PHP. Se ha trabajado con los archivos contenidos en la carpeta *templates*. La Web está formada por varios templates. El template asociado a la página principal se llama *epsevghome.php* y el template asociado a las páginas interiores se llama *epsevg_interior.php*. En estos dos templates se ha modificado el archivo *index.php*. Dichos archivos contienen el código PHP referente a la estructura de la página principal y de las páginas interiores. Cada una de estas páginas se introduce en Joomla!, el gestor de contenidos que se ha utilizado para realizar la Web.

Los archivos modificados en la realización de este proyecto han sido principalmente los dos archivos *index.php*, el *header-epsevg.php* y los CSS. También se han creado nuevos CSS y los archivos *estils-homepage.php* y *estils-interior.php*.

En la Intranet también se han creado archivos nuevos y modificados otros ya existentes. Principalmente se han modificado los archivos *index.php* y los archivos CSS, así como también se ha creado el archivo *header-accesibilitat.php* y nuevos CSS para aplicar el cambio de contraste.

A continuación se explicarán los cambios más significativos realizados en el código PHP.

1. Creación de los archivos *estils-homepage.php*, *estils-interior.php* y *header-accesibilitat.php*.

Se han creado dos archivos que contienen las instrucciones PHP necesarias para aplicar los CSS a la Web. El archivo *estils-homepage.php* contiene las instrucciones de la página principal y el archivo *estils-interior.php* las que se aplican a las páginas interiores.

La página Web de la EPSEVG puede ser visualizada con tres estilos diferentes: el estilo "normal" (estilo por defecto cuando se accede a la Web por primera vez), el cambio de contraste y sin estilo. El objetivo de la creación de estos archivos es mantener el estilo aplicado a la Web hasta que se indique lo contrario, independientemente del número de páginas que se visiten dentro del sitio Web. La primera vez que se accede a la Web de la EPSEVG se visualizará la página "normal", con el fondo blanco y las letras oscuras, sin tener aplicado ningún tipo de contraste. El contraste será cambiado cada vez que se clique el link *Canvi de contrast*: la primera vez que sea clicado, la página será visualizada con el fondo oscuro y las letras blancas; la segunda vez tendrá el fondo blanco y las letras oscuras; la tercera vez, fondo oscuro y letras blancas; y así sucesivamente. Si teniendo un estilo aplicado accedemos a cualquiera de las otras páginas que forman el sitio, estas se han de visualizar con el mismo estilo, tanto con contraste aplicado, sin él o sin estilo. Se tendrá que mantener el estilo aplicado independientemente de la página en la que nos encontremos o de las páginas que se visiten en ese momento. En definitiva, total libertad de movimientos por las diferentes páginas que componen la web aplicándoles a todas el mismo estilo hasta nueva orden.

Si se quiere visualizar la página sin ningún estilo, al aplicárselo de nuevo se tendrá que mantener el contraste anterior. Para ello se han creado estos dos archivos con las

instrucciones PHP necesarias para la aplicación de los diferentes estilos en cada unas de las situaciones planteadas. En el Anexo 7 se muestra el árbol de decisiones en el que se pueden observar las variables sesión creadas y el valor tomado en cada caso.

La variable sesión *css* (`$_SESSION["css"]`) puede tomar tres posibles valores: *claro*, *oscuro* y *sensestil*. Esta variable indica el CSS aplicado. Si su valor es *oscuro* aplicará el CSS de contraste, fondo oscuro y letras blancas. Si la variable toma por valor *claro* aplicará el CSS de fondo claro y letras oscuras. Este último es el CSS aplicado por defecto. En cambio, si la variable sesión toma por valor *sensestil* la página será visualizada sin ningún estilo aplicado.

Cada vez que la variable *css* cambie de valor, es decir, cada vez que se pulsen los links *Canvi de contrast* o *Sense estil*, las variables sesión *estil* y *contrast* estarán preparadas para que cuando se vuelva a pulsar alguno de los dos enlaces se puedan aplicar los CSS adecuados.

En la Intranet, el archivo *header-accesibilitat.php* que se encuentra en la carpeta *lib* contiene las mismas instrucciones que los archivos *estils-interior* i *estils-homepage*, ya que realizan la misma función.

Mediante la instrucción *require_once* estos archivos se incluyen en otros, así se evita que contengan más líneas de código. En la Intranet, el archivo *header-accesibilitat* es llamado desde *app_gestio_usuaris.php*, *app_index.php* y *index.php* que se encuentran en la carpeta *apps*, en la subcarpeta *lib*. También se llaman desde el archivo *ban_index.php* que se encuentra dentro de la carpeta *banner* y desde *index.php*. En la Web el archivo *estils-interior* es llamado desde los *index.php* que se encuentran en los templates *epsevg-home.php* y *epsevg-interior.php*.

En las siguientes imágenes se puede observar la estructura de carpetas y subcarpetas que siguen los archivos en la Web e Intranet sucesivamente.

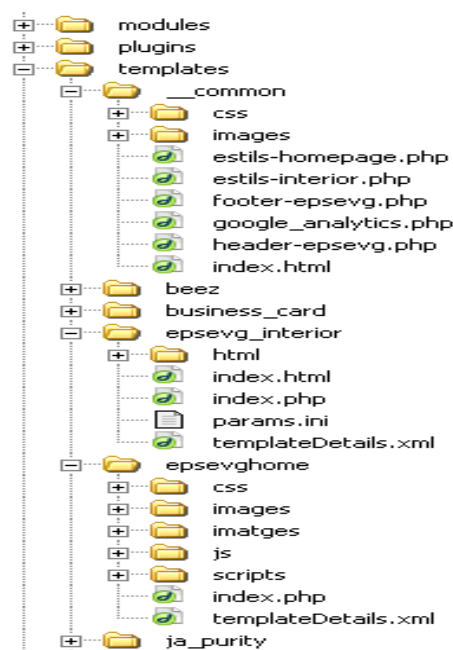


Figura 12. Estructura de los archivos en la Web.

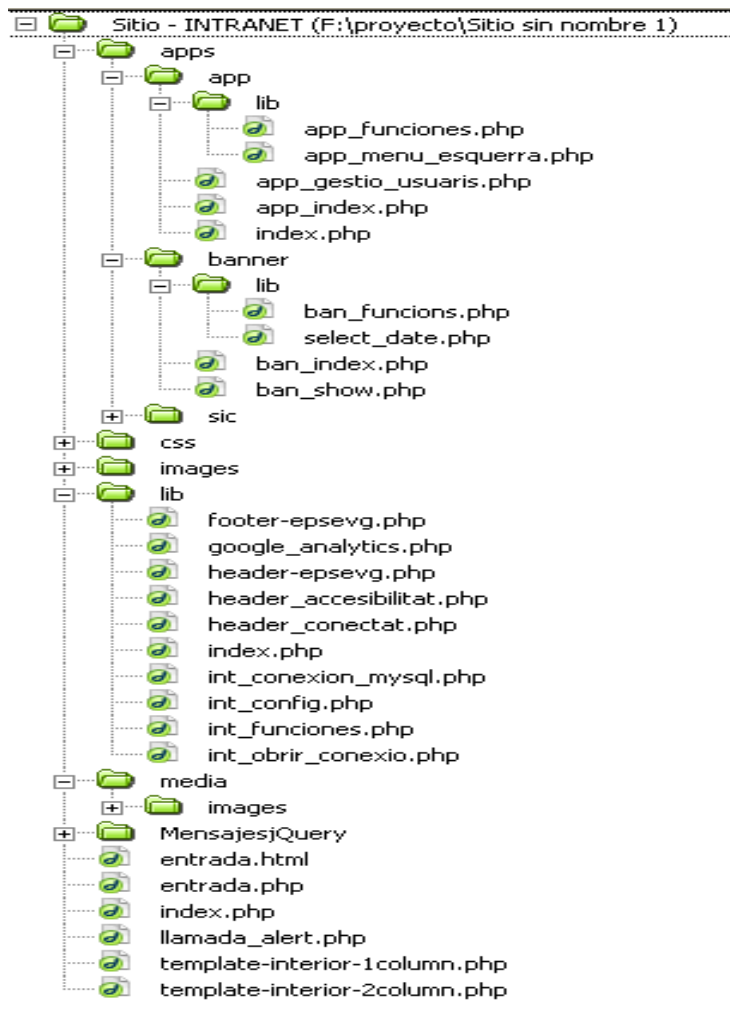


Figura 13. Estructura de los archivos de la Intranet.

2. Cookies.

Para mantener el estilo aplicado, independientemente de si se visualiza la página con contraste o sin él, se han creado unas cookies que mantienen el CSS. Estas cookies han sido añadidas a los archivos `index.php` de los templates *epsevg* y *epsevg_interior*.

En la Intranet se han añadido las mismas cookies a los archivos *app_gestio_usuaris.php*, *app_index.php* i al *index.php* que se encuentran dentro de la carpeta *apps* en la subcarpeta *lib*. También han sido añadidas al archivo *ban_index.php* que se encuentra dentro de la carpeta *banner* y a los *index.php* que se encuentran en la carpeta raíz y dentro de *lib*.

3. Llamada a las hojas de estilo.

Desde el archivo *header-epsevg.php* que se encuentra en la carpeta '*__common*' se llama a las hojas de estilo CSS. Este archivo contiene el código HTML correspondiente a los links de la cabecera. En este archivo se añadieron las instrucciones PHP necesarias para cargar los estilos CSS.

En la Intranet se han realizado los mismos cambios en el *header-epsevg.php* que se encuentra dentro de la carpeta *lib*.

4. Modificación y creación de hojas de estilos CSS.

Cada una de las páginas que forman la Web de la EPSEVG se divide en tres partes: cabecera, cuerpo o body y pie. A cada una de las tres partes se añade un CSS diferente, que define el estilo de cada una. Estos CSS son comunes a todas las páginas.

A continuación se hará un listado de los CSS aplicados a la página Web y de los estilos que recogen cada uno de ellos.

- Header.css. Contiene el estilo asociado a la cabecera cuando no se le ha aplicado ningún cambio de contraste.
- Header-contrast.css. Contiene el estilo asociado a la cabecera cuando se le aplica el cambio de contraste. Este CSS ha sido creado por nosotros.
- Template_home_css.css. Contiene el estilo asociado a todo el cuerpo de la página cuando no se le aplica ningún contraste.
- Template_home_css.css. Contiene todo el estilo asociado al cuerpo de la página cuando se le aplica el cambio de contraste.
- Footer.css. Contiene todo el estilo asociado al pie de la página cuando no le es aplicado ningún contraste.
- Footer-contrast.css. Contiene todo el estilo asociado al pie de la página cuando se le es aplicado el cambio de contraste.

Estos estilos son comunes a todas las páginas que forman la Web, ya que se ha de crear un estilo de presentación coherente a todas ellas.

Se han realizado los cambios pertinentes en los CSS. Se han creado los CSS de contraste caracterizados por tener las letras claras y el fondo oscuro y se han hecho las modificaciones requeridas en todos ellos.

En la Intranet se encuentran los CSS *header.css* que contiene el estilo asociado a la cabecera, *footer.css* que contiene el estilo asociado al pie de página y los archivos *menus.css* y *template_css.css* que contienen el estilo asociado al cuerpo de la página. Se han modificado estos archivos para que la Web e Intranet se asemejen y se han creado nuevas hojas de estilo para aplicar el contraste: el archivo *contrast.php* contiene el estilo asociado a la cabecera cuando se realiza el cambio de contraste y el archivo *footer-contrast.php* contiene el estilo asociado al pie de página cuando se le aplica el cambio de contraste.

6.9 **Página Accesibilidad**

Se ha creído conveniente crear un link de Accesibilidad. Al acceder a este, el usuario encontrará información sobre:

- Política de accesibilidad

En este punto se hace hincapié en el tema de accesibilidad. Se nombran las pautas, normas y leyes a seguir y se incluyen los correspondientes links.

- Atajos de teclado

En este punto se muestra la forma de utilizar los atajos de teclado y ver como varían estos en la utilización de un tipo de navegador u otro. Ver anexo 4.

- Recursos externos de accesibilidad.

Se muestran algunas de las herramientas utilizadas para la validación de nuestra web y sus respectivos enlaces.

Se ha creído conveniente que el link “Accesibilidad” aparezca en la cabecera de la Web. La siguiente imagen muestra exactamente el lugar del link.



Figura 14. Link accesibilidad.

7 Resultados obtenidos.

7.1 Introducció

En este apartado se detallarán los resultados obtenidos con este proyecto. Básicamente, el resultado principal es que la Web e Intranet de la EPSEVG son accesibles.

A continuación se explicarán los cambios visuales realizados en la web y los realizados en la Intranet de la EPSEVG que contribuyen a que sean accesibles. Se adjuntan imágenes que hacen visibles estos cambios. También se mostrarán algunas de las validaciones realizadas y los resultados obtenidos.

7.2 Cambios visibles realizados en la web.

Web anterior de la EPSEVG:



Figura 15. Web anterior de la EPSEVG.

Nueva Web de la EPSEVG:



Figura 16. Nueva Web de la EPSEVG.

Esta imagen de la nueva web de la EPSEVG corresponde a la última versión de ésta, después de hacer todas las modificaciones pertinentes para lograr su accesibilidad.

A continuación se detallarán los cambios realizados en la web, los que afectan a toda la web y los cambios propios de alguna de sus partes como el encabezado (header), cuerpo (body) o pie (foot).

- Cambios propios del encabezado.

En la siguiente imagen se puede observar el encabezado de la página antes de haberle aplicado ningún cambio.



Figura 17. Cabecera inicial.

Este encabezado ha sufrido muchas modificaciones desde que se empezó con el proyecto. Con la inclusión del link de cambio de contraste, se decidió que los colores de este encabezado debían mantenerse cuando se le aplicara el contraste, ya que toda la página iba a tener el fondo oscuro y las letras blancas. Por este motivo se invirtieron los colores del encabezado cuando se veía la página sin aplicarle ningún cambio de contraste.

Al encabezado se añadieron unos links: el de la izquierda conduce directamente al contenido de la página, saltando los menús de navegación; el link de Accesibilidad muestra una página en la que se explica el significado de accesibilidad web y en la que se detallan los atajos de teclado. El link de *Canvi de Contrast* permite cambiar el contraste de la página, aplicando un fondo oscuro con las letras blancas. El link *Sense Estil* permite ver la página web sin aplicarle ningún estilo. Estos cambios realizados han sido explicados previamente en el capítulo de Diseño.

A continuación, se puede observar en las siguientes imágenes los encabezados pertenecientes a la página web cuando no se le aplica ningún contraste, cuando sí se le es aplicado y cuando no se le aplica ningún estilo.



Figura 18. Cabecera Web.



Figura 19. Cabecera Web aplicando un cambio de contraste.

[Saltar a contingut](#)

- [Amb estil](#)



Escola Politècnica Superior
d'Enginyeria de Vilanova i la Geltrú
UNIVERSITAT POLITÈCNICA DE CATALUNYA

- [Futurs Estudiants](#)
- [Estudiants](#)
- [PDIPAS](#)
- [Empreses](#)
- [Antics Alumnes](#)
- [Incomming Students](#)

Contacta'ns

- 93 8967701
- vine a l'UPC de Vilanova
- [GRAUS i ENGINYERIES a l'EPSEVG](#)
- [CONEXIÓ EPSEVG](#)
- [ACTUALITAT](#)
- [SERVEIS DEL CAMPUS](#)
- [RECERCA](#)
- [CONTACTA](#)

Enginyeries a l'EPSEVG

- [Graus EEES](#)
 - [Grau en Enginyeria Mecànica](#)
 - [Informació acadèmica](#)
 - [Matèries](#)
 - [Grau en Enginyeria Elèctrica](#)
 - [Grau en Enginyeria de Disseny Industrial i Desen](#)

Figura 20. Web sin estilo.

- Cambios que afectan a toda la web.
- Cuando se clicla el link de *Canvi de Contrast*, se aplica un cambio de contraste que afecta a toda la página.



Figura 21. Web interior EPSEVG.



Figura 22. Web interior con cambio de contraste.

- Cuando se recorren los links de una página mediante teclado, se puede observar que éstos aparecen en otro color para indicar al usuario en qué lugar de la página está situado. Estos cambios se han realizado en el CSS de la página principal e interiores y en los CSS creados para realizar el cambio de contraste. Afectan a todos los links que puedan aparecer en la página.

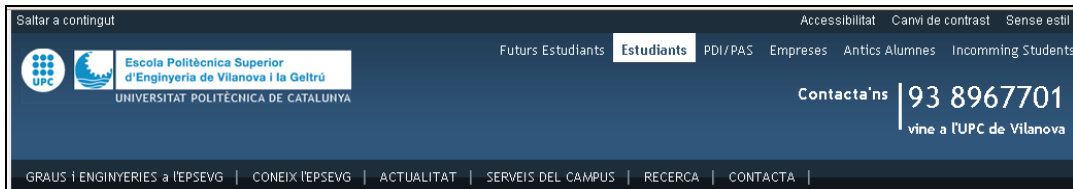


Figura 23. Focus.

- Se han modificado las tablas para que sean accesibles.

| Enginyeries a l'EPSEVG | |
|--|---|
| <ul style="list-style-type: none"> ↳ Graus EEEES <ul style="list-style-type: none"> • Grau en Enginyeria Mecànica <ul style="list-style-type: none"> • Informació acadèmica • Matèries • Grau en Enginyeria Elèctrica • Grau en Enginyeria de Disseny Industrial i Desenvolupament del Producte • Grau en Enginyeria Electrònica Industrial i Automàtica ↳ Enginyeries Tècniques ↳ Enginyeria Automàtica ↳ Doble titulació UPC.UOC ↳ Diplomes d'angles Tècnic ↳ Curs Introductorí ↳ Horaris ↳ Calendari acadèmic ↳ Matricula ↳ Normatives i Reglament ↳ Període Avaluació Final | |
| <p>banner de pagina</p> <p>Home > Graus i Enginyeries a l'EPSEVG > Graus EEEES > Grau en Enginyeria Mecànica > Informació acadèmica</p> <h2>Informació Acadèmica Grau en Enginyeria Mecànica</h2> | |
| Durada | Quatre anys |
| Càrrega lectiva | 240 crèdits |
| Nota de tall PAU * | 5,00 |
| Nota de tall CFGS * | 5,00 |
| Nombre de places | 230 (entre els quatre graus de l'àmbit d'enginyeria industrial que ofereix l'EPSEVG) |
| Accés comú: | L'accés als estudis de l'àmbit de l'enginyeria industrial per a l'Escola Politècnica Superior d'Enginyeria de Vilanova i la Geltrú (UPC), és per preinscripció comuna a les 230 places ofertades, distribuïdes en els següents plans d'estudis: <ul style="list-style-type: none"> ✓ Grau en Enginyeria Elèctrica (60 places), ✓ Grau en Enginyeria Electrònica Industrial i Automàtica (40 places), ✓ Grau en Enginyeria Mecànica (140 places). |
| Assignació d'estudis: | El primer curs és comú per a tots aquests estudis. Un cop superat, l'estudiant podrà demanar els estudis en els que vol continuar, ordenats per ordre de preferència. El centre assignarà l'especialitat en funció de la sol·licitud de l'estudiant i de l'expedient acadèmic cursat al centre. |
| Organització dels estudis | Aquells estudiants que hagin superat el primer curs dels graus de l'àmbit industrial de l'EPSEVG, tenen també la possibilitat incorporar-se al segon curs del Grau en Disseny Industrial i Desenvolupament del Producte, en funció del seu expedient acadèmic i places disponibles. |
| | Els estudis s'organitzen en vuit semestres (quatre anys) i cal cursar 240 ECTS: 60 crèdits d'assignatures bàsiques, 126 crèdits d'assignatures obligatòries, 30 crèdits d'assignatures optatives i un treball final de grau de 24 crèdits. |
| | * Nota orientativa corresponent a l'estudi d'Enginyeria Tècnica Industrial, especialitat en Mecànica, curs 2008/2009 |

Figura 24. Tabla accesible en la Web.

- Se les ha aplicado el contraste de colores correspondiente para que puedan ser vistas cambiando el contraste.

banner de pagina
 Home > Graus i Enginyeries a l'EPSEVG > Graus EEES > Grau en Enginyeria Mecànica > Informació acadèmica

Informació Acadèmica Grau en Enginyeria Mecànica

| | |
|----------------------------------|---|
| Durada | Quatre anys |
| Càrrega lectiva | 240 crèdits |
| Nota de tall PAU * | 5.00 |
| Nota de tall CFGS * | 5.00 |
| Nombre de places | 230 (entre els quatre graus de l'àmbit d'enginyeria industrial que ofereix l'EPSEVG) |
| Accés comú: | <p>L'accés als estudis de l'àmbit de l'enginyeria industrial per a l'Escola Politècnica Superior d'Enginyeria de Vilanova i la Geltrú (UPC), és per preinscripció comuna a les 230 places ofertades, distribuïdes en els següents plans d'estudis:</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Grau en Enginyeria Elèctrica (60 places), <input checked="" type="checkbox"/> Grau en Enginyeria Electrònica Industrial i Automàtica (40 places), <input checked="" type="checkbox"/> Grau en Enginyeria Mecànica (140 places). |
| Assignació d'estudis: | <p>El primer curs és comú per a tots aquests estudis. Un cop superat, l'estudiant podrà demanar els estudis en els que vol continuar, ordenats per ordre de preferència. El centre assignarà l'especialitat en funció de la sol·licitud de l'estudiant i de l'expedient acadèmic cursat al centre.</p> <p>Aquells estudiants que hagin superat el primer curs dels graus de l'àmbit industrial de l'EPSEVG, tenen també la possibilitat incorporar-se al segon curs del Grau en Disseny Industrial i Desenvolupament del Producte, en funció del seu expedient acadèmic i places disponibles.</p> |
| Organització dels estudis | <p>Els estudis s'organitzen en vuit semestres (quatre anys) i cal cursar 240 ECTS: 60 crèdits d'assignatures bàsiques, 126 crèdits d'assignatures obligatòries, 30 crèdits d'assignatures optatives i un treball final de grau de 24 crèdits.</p> <p>* Nota orientativa corresponent a l'estudi d'Enginyeria Tècnica Industrial, especialitat en Mecànica, curs 2008/2009</p> |

Figura 25. Pàgina con tabla accesible y cambio de contraste.

- Se ha añadido texto alternativo a los links y a las imágenes que lo han necesitado. El texto alternativo es visible cuando nos situamos con el ratón encima del link, de la imagen o del elemento no textual.

7.3 Intranet.

Se ha hecho accesible la navegación dentro de la Intranet. La mayoría de los cambios realizados en la web EPSEVG se han aplicado a la Intranet, todo y que existen grandes diferencias, sobretodo en el diseño.

La siguiente imagen muestra la página principal de la Intranet. El cambio más significativo de esta página ha sido el hacer accesible el formulario que aparece en ella, todo y que a simple vista no pueda ser apreciado.



Figura 26. Acceso a la Intranet.

Una vez los usuarios se registran, validan y acceden al interior de la intranet, entran en la siguiente página:

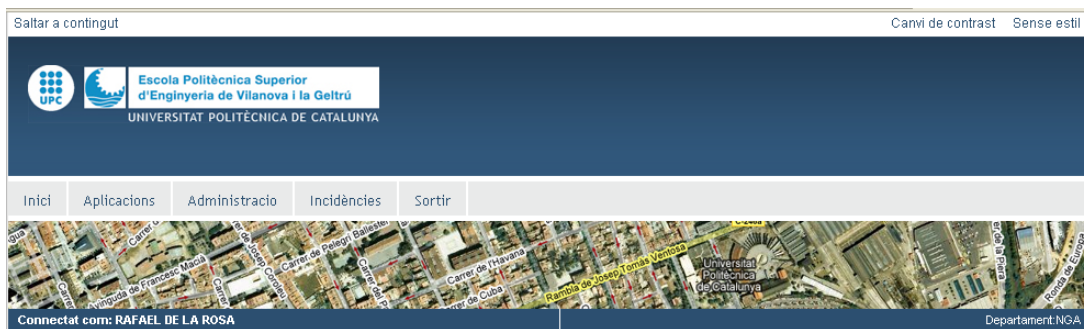


Figura 27. Página interior de la Intranet.

Se puede observar que la cabecera de la Intranet es algo diferente a la de la Web pero manteniendo las partes fundamentales de esta como son los links de cambio de contraste, sin estilo y saltar al contenido. En la siguiente imagen se muestra el cambio de contraste.

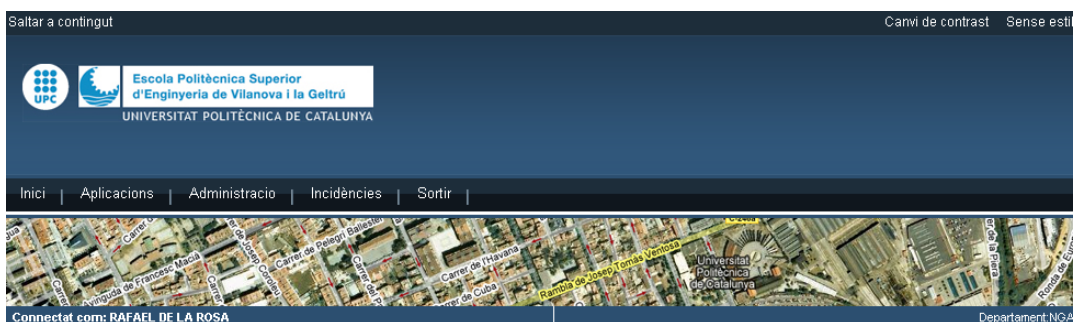


Figura 28. Interior de la Intranet con cambio de contraste.

A cada una de las tablas y formularios, igual que en la Web, se les han aplicado los cambios pertinentes para que sean accesibles.

7.4 Validaciones

El principal objetivo de este proyecto es hacer que la nueva web de la EPSEVG sea accesible. Para demostrarlo se han tenido que hacer modificaciones en el código para poder cumplir con la WCAG 1.0. Para verificar cada uno de los cambios realizados se han utilizado diferentes herramientas que comprueban la accesibilidad web, los CSS y el código XHTML. Se ha tenido que validar la web e Intranet página a página en cada una de las herramientas utilizadas.

- validación de código XHTML.

Si presentaba errores se han tenido que corregir modificando el código de la página.

- validación de CSS

Se ha tenido que comprobar que los CSS cumplieran los estándares y que no había errores en ellos.

Los errores CSS no se han eliminado por completo ya que si se quería que la página fuese vista en todos los navegadores por igual, se tenían que añadir al CSS unos hacks que son reconocibles por algunas versiones de los navegadores y mejoran la visión de la página en éstos. Estos hacks no son reconocidos por los validadores de CSS y dan error. Por ese motivo se ha llegado a un compromiso, el código no cumple con los estándares al 100% (tres errores de CSS) pero a cambio, se puede ver la web en cualquier navegador.

- validación de las pautas de accesibilidad.

Todas las páginas validadas presentaban un gran número de errores, como se puede observar en la siguiente imagen.



Figura 30. Errores T.A.W. después de la validación.

Con este proyecto se concluye lo siguiente:

- Se han realizado análisis completos de la web para comprobar su accesibilidad, las hojas de estilo y el código XHTML. Se han conseguido reducir la mayoría de los errores encontrados, pero se ha querido llegar a un compromiso con el diseño.
- Se ha tenido que modificar el código para conseguir los objetivos propuestos.
- La nueva página Web de la EPSEVG ha sido testeada por personas con diferentes tipos de discapacidad. Reuniendo sus opiniones se ha llegado a la conclusión de que la página es muy accesible: acceder al contenido de los enlaces resulta bastante rápido para personas invidentes gracias a la utilización de encabezados, ya que JAWS ofrece una opción de desplazamiento por los diferentes encabezados de la Web. Lo mismo sucede con las listas. Es una página en la que no abundan los botones ni la existencia de flash lo cual facilita la revisión de la página con el programa JAWS. Las imágenes están etiquetadas como tal, los enlaces están bien definidos y no existen elementos dinámicos.

8 Futuras vías de trabajo

El presente proyecto ha sido diseñado con el fin de estar en funcionamiento en el nuevo curso académico 2009/10-1. Actualmente para poder navegar por la nueva Web se ha de ir a la URL <http://kato.epsevg.upc.es/joomla/>. Esta web no es oficial a día de hoy, no por problemas de diseño o accesibilidad, sino por falta de contenido. El contenido de la Web ha de estar formado por gran parte de la información que hay en la página Web actual más la información referente a los nuevos GRADOS que ofrecerá la EPSEVG. La información referente a los Grados no se puede publicar en su totalidad porque aún se está trabajando en ella.



The screenshot shows the website interface for the EPSEVG. At the top, there are navigation links: 'Saltar a contingut', 'Accessibilitat', 'Canvi de contrast', and 'Sense estil'. Below these are links for 'Futurs Estudiants', 'Estudiants', 'PDI/PA5', 'Empreses', 'Antics Alumnes', and 'Incomming Students'. The main header includes the UPC logo and the school name: 'Escola Politècnica Superior d'Enginyeria de Vilanova i la Geltrú, UNIVERSITAT POLITÈCNICA DE CATALUNYA'. A contact number '93 8967701' is displayed with the text 'Contacta'ns' and 'vine a l'UPC de Vilanova'. A navigation menu includes 'GRAUS I ENGINYERIES a l'EPSEVG', 'CONEIX l'EPSEVG', 'ACTUALITAT', 'SERVEIS DEL CAMPUS', 'RECERCA', and 'CONTACTA'. The main content area features a 'banner de pagina' with a breadcrumb trail: 'Home > Graus i Enginyeries a l'EPSEVG > Graus EEES > Grau en Enginyeria de Disseny Industrial i Desenvolupament del Producte > Matèries'. The title is 'Matèries Grau en Enginyeria de Disseny Industrial i Desenvolupament del Producte'. Below the title, it lists 'Crèdits d'Assignatures del Grau' with a table of subjects and credits.

| Formació bàsica | Optatives (36) |
|---|----------------------------|
| Enginyeria Gràfica (18) | |
| Física (12) | |
| Gestió d'Empreses (6) | |
| Informàtica (6) | |
| Matemàtiques (18) | |
| Química (6) | |
| Formació comuna de la branca industrial | Treball final de Grau (24) |
| Enginyeria i Societat (12) | |
| Tecnologia específica | |
| Ciència i Enginyeria de Materials (6) | |
| Enginyeria Gràfica (12) | |
| Evolució del Producte i Societat (6) | |
| Mecànica (18) | |
| Metodologia del Disseny (18) | |

Figura 31. Información de Grados.

Los creadores del proyecto ya hemos dado el primer gran paso, conseguir que una escuela de la Universidad Politécnica de Catalunya disponga de Web Accesible.

Una vez la Web esté en funcionamiento necesitará de un mantenimiento en lo que a Accesibilidad se refiere, es decir, que la escuela requerirá de alguna persona encargada de hacer accesible cada nuevo contenido que se agregue a la página. Para facilitar esta tarea se han creado unos manuales para realizar tablas y formularios accesibles. Estos se encuentran en los anexos de este proyecto con el fin de que cualquier persona con conocimientos básicos de HTML, PHP y hojas de estilo pueda con un poco de tiempo y dedicación mantener nuestro trabajo.

9 Bibliografía

Libros consultados.

Luis Miguel Cabezas Granados. Prólogo de Zeev Suraski y Andi Gutmans. *Manual imprescindible de PHP5*. Ediciones ANAYA MULTIMEDIA (GRUPO ANAYA, S.A.), 2008.

Michael K. Glass, Yann Le Scouarnec, Elizabeth Naramore, Gary Mailer, Jeremy Stolz, Jason Gerner. *Desarrollo Web con PHP, Apache y MySQL*. Ediciones ANAYA MULTIMEDIA (GRUPO ANAYA, S.A.), 2004.

Webs visitadas.

- Pautas de accesibilidad al contenido en la web 1.0:

http://www.discapnet.es/web_accesible/wcag10/WAI-WEBCONTENT-19990505_es.html

- PHP

<http://www.php.net/manual/es/>

- Lista de puntos de verificación según la prioridad:

http://www.discapnet.es/web_accesible/wcag10/checkpoint-list.html

- Web provisional de la escuela:

<http://kato.epsevg.upc.es/joomla/>

- BOE:

http://www.boe.es/aeboe/consultas/bases_datos/doc.php?coleccion=iberlex&id=2007/22440&txtlen=1000# analisis

- TAW:

<http://tawdis.net/>

- Nueva normativa de accesibilidad:

<http://www.hipertext.net/web/pag296.htm>

- Inteco, descarga de la norma UNE:

http://www.inteco.es/Accesibilidad/Normativa_1/Descarga/DescargaUNE_139803

- Pautas accesibilidad 2.0 W3C:

<http://www.codexemplar.org/traduccion/pautas-accesibilidad-contenido-web-2.0.htm#intro-layers-guidance>

- Puntos de verificación de las pautas 2.0:

<http://gweos.net/blog/2009/01/28/guias-practicas-para-profesionales-web-puntos-de-verificacion-de-las-pautas-de-accesibilidad-al-contenido-web-wcag-20/>

- Biblioteca EPSEVG

<http://biblioteca.upc.es/bib340/>

- Web ONCE

<http://www.once.es/new/home/>

Programas utilizados.

CONTRAST-A.

JAWS 9.0.

Macromedia Dreamweaver 8.

TAW3.

Documentación on-line

- Test de Accesibilidad Web (TAW).

http://aui.es/IMG/pdf_test_accesibilidad_CTIC.pdf

- Manual de páginas web accesibles.

<http://www.carm.es/ctra/cendoc/publicaciones/>

- Manual de HTML:

<http://www.aulaclic.es/html/index.htm>

- Manual de CSS:

<http://www.desarrolloweb.com/>

- Manual de formularios:

http://www.webtaller.com/construccion/lenguajes/html/lecciones/formularios_html.php

Anexo 1 - Manual de Tablas Accesibles

Estructura básica de una tabla accesible:

```
<table title="Título que será mostrado" summary="Resumen del contenido de la tabla">
  <caption> Título de la tabla </caption>

  <thead>
    <tr id="cabecera" >
      <th scope="col">Título de columna 1</th>
      <th scope="col">Título de columna 2</th>
      ...
    </tr>
  </thead>

  <tfoot>
    <tr id="pie">
      <th scope="row">PIE 1</th>
      <th>PIE 2</th>
      ...
    </tr>
  </tfoot>

  <tbody>
    <tr>
      <td scope="row">1ª celda de fila 1</td>
      <td>2ª celda de fila 1</td>
      ...
    </tr>
    <tr>
      <td scope="row">1ª celda de fila 2</td>
      <td>2ª celda de fila 2</td>
      ...
    </tr>
    <tr> ... </tr>
  </tbody>

</table>
```

Se pueden observar en negrita todos aquellos atributos o etiquetas que hacen que la tabla sea accesible. A continuación se explicará cada uno de ellos y algunas de las pautas a seguir.

- No aplicar estilos a las tablas. Los estilos se han de aplicar mediante hojas de estilo CSS.
- No utilizar tablas para maquetar la página.

- **<title> y </title>**

Proporcionan un texto que será mostrado cuando el “*mouse*” se sitúe encima de la tabla.

- **summary**

Proporciona un resumen sobre el contenido de la tabla, su relación con el documento, la relación entre las celdas u otras relaciones que no son evidentes a la hora de hacer un análisis estructural de la tabla, pero que a simple vista sí lo son. Estos resúmenes son muy útiles para los lectores de voz y muy necesarios si no se proporciona un título.

- **<caption> y </caption>**

Contienen el nombre o título de la tabla, aportando información sobre su contenido.

- **<thead>, <tbody> y <tfoot>**

Las tablas accesibles se dividen en tres partes: **header** o cabecera, **body** o cuerpo y **foot** o pie. Puede ser que no todas las tablas contengan las tres partes, ya que pueden carecer de alguna de ellas.

La etiqueta **<thead>** define el comienzo de la cabecera de la tabla, y **</thead>** el final. El contenido de la celda aparecerá en negrita y centrado. Dentro de esta etiqueta, se definirán las filas con **<tr>** y las celdas, pertenecientes a las columnas, con **<th>** que sustituye a **<td>**. Dentro de **<tbody>** estará el cuerpo de la tabla donde se definirán las filas y columnas de manera tradicional, con **<tr>** y **<td>**. Esta etiqueta se debe acompañar del atributo **scope**, que se detallará más adelante. Mediante la etiqueta **<tfoot>** se identificará el pie de tabla. Tiene la misma estructura que **<thead>**, para definir las filas se utilizará **<tr>** y para las columnas **<th>**.

Aunque la tabla siga una estructura de cabecera, cuerpo y pie de tabla, a la hora de escribir el código no se seguirá ese orden sino el siguiente: *thead*, *tfoot* y *body*. Siempre habrá que ordenarlo de la misma manera.

- **scope, axis y headers**

Existen atributos que vinculan la información que contienen las celdas que forman la tabla: **scope**, **axis** y **headers**. La elección entre utilizar *scope* o *headers* no afecta al resultado, siempre y cuando la relación entre el encabezado y las celdas de datos sea correcta, la elección dependerá de la complejidad de la tabla. En este caso, como es una tabla sencilla se utiliza *scope*. Si la tabla fuera más compleja, se utilizaría *header*.

El atributo **scope** se utiliza solamente con las etiquetas **<th>** y **<td>**. Tiene cuatro posibles valores:

- . *row* (celda referida al inicio de una fila)
- . *col* (celda referida al inicio de una columna)
- . *rowgroup* (celda referida al inicio de un conjunto de filas)
- . *colgroup* (celda referida al inicio de un conjunto de columnas).

Anexo 2 - Manual de Formularios Accesibles

Estructura básica de un formulario accesible:

```
<FORM METHOD="post" ACTION="mis_datos.php">
<fieldset>
  <legend>Datos personales</legend>
  <p>
    <label for="nombre">Nombre:</label>
    <input type="text" id="nombre" accesskey="n"
tabindex="1" size="30">
    <label for="primer-apellido">Primer
apellido:</label>
    <input tabindex="2" accesskey="p" type="text"
id="primer-apellido" size="30">
    ...
  </p>
</fieldset>

<fieldset>
  <legend>Curiosidades</legend>
  <p>
    <label for="color-favorito">Tu color
favorito:</label>
    <select tabindex="4" accesskey="c" size="1"
id="color-favorito">
      <optgroup label="Azul">
        <option label="Azul cielo" value="azul
cielo">Azul cielo</option>
        <option label="Azul marino" value="azul
marino">Azul marino</option>
        <option label="Turquesa"
value="Turquesa">Turquesa</option>
      </optgroup>
      <option value="Rojo">Rojo</option>
      <option value="Verde"
selected="selected">Verde</option>
    </select>
  </p>
  <p>
    <label for="deporte">¿Te gusta el deporte ?</label>
    <input tabindex="5" accesskey="d" type="checkbox"
id="deporte" value="ON">
    <p>¿Mar o montaña?</p>
    <p>Mar <input tabindex="6" accesskey="m"
type="radio" checked name="preferencias" value="mar"></p>
    <p>Montaña <input tabindex="7" type="radio"
name="preferencias" value="montaña"></p>
  <p>
    <label for="aficiones">Aficiones</label>
  </p>
</form>
```

```
<p><textarea tabindex="8" accesskey="a" rows="5"
name="aficiones" cols="28">tecleee su nombre</textarea></p>
<p><input tabindex="9" accesskey="e" type="submit" value="Enviar
datos" name="enviar">
<input tabindex="10" accesskey="r" type="reset"
value="Restablecer" name="B2"></p>
</fieldset>
</FORM>
```

- Se pueden observar en negrita todos aquellos atributos o etiquetas que hacen que el formulario sea accesible. A continuación se explicará cada uno de ellos y algunas de las pautas a seguir.
- No modificar el estilo del formulario mediante atributos. Se hará mediante hojas de estilos CSS.
- Se han de incluir, por defecto, caracteres en los cuadros de edición y áreas de texto.
- **<label> y </label>**

La etiqueta *<label>* se asocia a un control de formulario. Cada control ha de llevar asociada explícitamente esta etiqueta. El atributo *for* asocia un rótulo al control. El valor de este atributo ha de ser el mismo que el valor del atributo *id* del control asociado.

- **tabindex**

Este atributo establece un orden lógico de tabulación: se le asigna un orden de lectura para cuando se recorra el formulario con el tabulador.

- **accesskey**

Proporciona atajos de teclado. Según la versión del navegador y el sistema operativo que se tenga, las teclas de control del teclado serán unas u otras. Por ejemplo, para Firefox 2.0 con sistema operativo Windows, se accederá al teclado presionando a la vez las teclas ALT y SHIFT con la letra asignada para ese campo. En cambio, si el navegador es Internet Explorer 5+ y el sistema operativo es Windows, se tendrá que presionar ALT a la vez que la letra asignada, soltar y presionar INTRO.

- **fieldset**

Agrupar los campos del formulario según la información que contengan. Ayuda a entender los formularios largos y hace más accesible la navegación por voz.

- **Legend**

Este atributo le da un título a la agrupación.

- **<optgroup> y </optgroup>, <option> y </option>, label.**

Cuando se tiene un menú desplegable con una lista de información muy larga, es necesario, siempre y cuando sea apropiado, agrupar la información del menú. Para ello se utiliza la etiqueta *<optgroup>* que es la que especifica cada grupo de opciones. Cada



una de las opciones que forma el grupo se identifica con la etiqueta `<option>`. El atributo *label*, que será usado en las dos etiquetas, nombra el campo. Este nombre suele ser más corto que el del campo que se describe, el especificado dentro de la etiqueta *option*. En el caso del ejemplo, sería el mismo nombre. Un aspecto a tener en cuenta es que *label* ha de estar a la izquierda del campo que se describe, menos para radio buttons o check boxes donde puede ir a la derecha del campo.

Anexo 3 - Abreviaturas

Abreviaturas que aparecen en los encabezados de las tablas que forman la Web.

Tablas **Informació Acadèmica** GRADO:

| Palabras | Abreviatura |
|---------------------------|----------------|
| Nota de tall PAU | nota tall PAU |
| Nota de tall CFGS | nota tall CGFS |
| Nombre de places | places |
| Assignació d'estudis | assignació |
| Organització dels estudis | organització |

Tablas **Accés als estudis** INGENIERÍA:

| Palabras | Abreviatura |
|---|----------------|
| Sortides professionals | sortida |
| Requisits d'accés | requisits |
| Assignatures optatives que es recomana haver escollit a l'ensenyament secundari | recomanacions |
| Matèries de la modalitat de secundària en les quals cal anar especialment preparat | preparació |
| Nombre de places de nou ingrés | places |
| Preu mitjà d'un quadrimestre | preu quadri |
| Durada i càrrega lectiva | durada |
| Objectius docents | objectius |
| Organització dels estudis* | organització |
| Fase selectiva: durada i matèries en les quals es necessita fer una preparació especial | fase selectiva |
| Accés a segons cicles de la UPC | segon cicle |

*Este campo solo aparece en **Accés als estudis** de la Ingeniería Técnica Industrial, especialidad en Electricidad.

Tablas **Diplomes d'especialització** INGENIERÍA:

| Palabras | Abreviatura |
|------------------|-------------|
| Àrea optativitat | área opt |

Tabla **Doble Titulació UPC-UOC**:

| Palabras | Abreviatura |
|-----------------------------|-------------|
| Diplomat en C. Empresarials | diplomat |

Tablas **Diplomes d'Anglès Tècnic**:

| Palabras | Abreviatura |
|--|----------------|
| Introducció a l'anglès Tècnic i comprensió lectora | intro |
| Comunicació oral en anglès tècnic | com. oral |
| Anglès acadèmic: ciència, tecnologia i societat | anglès |
| Comunicació escrita en anglès tècnic | com.escrita |
| Cursos semi- o no presencials | no presencials |

Tablas **Curs Introductor**:

| Palabras | Abreviatura |
|------------------------------------|-------------|
| Matemàtiques (MAT) | mates |
| Introducció a l'enginyeria | intro |
| Horaris de Grups Curs 2008/2009 Q1 | horari Q1 |

Tablas **Matrícula/Documentació**:

| Palabras | Abreviatura |
|-------------------------------|----------------|
| Acolliment a la via lenta | acolliment |
| Trasllat d'expedient | trasllat |
| Sol·licitud de convalidacions | convalidacions |

Tablas **Deduccions de matrícula o matricules condicionals:**

| Palabras | Abreviatura |
|--|----------------|
| Estudiantat que opti a una beca | beca |
| Ser membre de família nombrosa | fam. nombrosa |
| Haver obtingut matrícula d'honor a batxillerat | matr. Honor |
| Ser personal de les universitats públiques catalanes | personal uni |
| Haver obtingut un dels premis de les diferents Olimpíades del coneixement | premis |
| Experimentar un grau de minusvalidesa igual o superior al 33% i sense cap títol universitari | minusvalidesa |
| Tenir 65 anys o més i no tenir cap títol universitari | més de 65 anys |
| Ser víctima, cònjuge, fill o filla de víctima d'actes terroristes sense cap títol universitari | víctima |

Anexo 4 - Atajos de Teclado

Para facilitar el acceso a los enlaces más comunes de la web, se han definido los siguientes atajos de teclado:

i: Acceso a la página principal

c: Cambia a contenido

a: Accesibilidad

o: Cambio a alto contraste

e: Cambio de estilo

l: Logo EPSEVG

El uso de las teclas accesibles vienen condicionadas por el tipo de navegador utilizado:

- Internet Explorer 5 o superiores: ALT + Tecla de acceso rápido + Tecla ENTER
- Internet Explorer 4.0: ALT + Tecla de acceso rápido
- Mozilla Firefox hasta 1.5: ALT + Tecla de acceso rápido
- Mozilla Firefox 2.0: ALT + SHIFT + Tecla de acceso rápido
- Netscape: CTRL + Tecla de acceso rápido
- Opera: SHIFT + ESCAPE + Tecla de acceso rápido
- MacOS: CONTROL + Tecla de acceso rápido
- Internet Explorer 4.5 (Mac): No soporta atajos de teclado
- Netscape 6 y anteriores: No soporta atajos de teclado.

Anexo 5 - Puntos de verificación WCAG 1.0

Prioridades

Cada punto de verificación tiene un nivel de prioridad asignado por el Grupo de Trabajo fundamentado en su impacto sobre la accesibilidad:

- **Prioridad 1**

Un desarrollador de contenidos de páginas Web tiene que satisfacer este punto de verificación. De otra forma, uno o más grupos de usuarios encontrarán imposible acceder a la información del documento. Satisfacer este punto de verificación es un requerimiento básico para que algunos grupos puedan usar los documentos Web.

- **Prioridad 2**

Un desarrollador de contenidos de páginas Web debe satisfacer este punto de verificación. De otra forma, uno o más grupos encontrarán dificultades en el acceso a la información del documento. Satisfacer este punto de verificación eliminará importantes barreras de acceso a los documentos Web.

- **Prioridad 3**

Un desarrollador de contenidos de páginas Web puede satisfacer este punto de verificación. De otra forma, uno o más grupos de usuarios encontrarán alguna dificultad para acceder a la información del documento. Satisfacer este punto de verificación mejorará la accesibilidad de los documentos Web.

Algunos puntos de verificación tienen especificado un nivel de prioridad que puede variar bajo ciertas condiciones (que se indican).

Puntos de verificación Prioridad 1:

En general

[1.1](#) Proporcione un texto equivalente para todo elemento no textual (Por ejemplo, a través de "alt", "longdesc" o en el contenido del elemento). *Esto incluye:* imágenes, representaciones gráficas del texto, mapas de imagen, animaciones (Por ejemplo, GIFs animados), "applets" y objetos programados, "ascii art", marcos, scripts, imágenes usadas como viñetas en las listas, espaciadores, botones gráficos, sonidos (ejecutados con o sin interacción del usuario), archivos exclusivamente auditivos, banda sonora del vídeo y vídeos.

[2.1](#) Asegúrese de que toda la información transmitida a través de los colores también esté disponible sin color, por ejemplo mediante el contexto o por marcadores.

[4.1](#) Identifique claramente los cambios en el idioma del texto del documento y en cualquier texto equivalente (por ejemplo, leyendas).

[6.1](#) Organice el documento de forma que pueda ser leído sin hoja de estilo. Por ejemplo, cuando un documento HTML es interpretado sin asociarlo a una hoja de estilo, tiene que ser posible leerlo.

[6.2](#) Asegúrese de que los equivalentes de un contenido dinámico son actualizados cuando cambia el contenido dinámico.

[7.1](#) Hasta que las aplicaciones de usuario permitan controlarlo, evite provocar destellos en la pantalla.

[14.1](#) Utilice el lenguaje apropiado más claro y simple para el contenido de un sitio.

Y si utiliza imágenes y mapas de imagen

[1.2](#) Proporcione vínculos redundantes en formato texto para cada zona activa de un mapa de imagen del servidor.

[9.1](#) Proporcione mapas de imagen controlados por el cliente en lugar de por el servidor, excepto donde las zonas sensibles no puedan ser definidas con una forma geométrica.

Y si utiliza tablas

[5.1](#) En las tablas de datos, identifique los encabezamientos de fila y columna.

[5.2](#) Para las tablas de datos que tienen dos o más niveles lógicos de encabezamientos de fila o columna, utilice marcadores para asociar las celdas de encabezamiento y las celdas de datos.

Y si utiliza marcos ("frames")

[12.1](#) Titule cada marco para facilitar su identificación y navegación.

Y si utiliza "applets" y "scripts"

[6.3](#) Asegúrese de que las páginas sigan siendo utilizables cuando se desconecten o no se soporten los scripts, applets u otros objetos programados. Si esto no es posible, proporcione información equivalente en una página alternativa accesible.

Y si utiliza multimedia

[1.3](#) Hasta que las aplicaciones de usuario puedan leer en voz alta automáticamente el texto equivalente de la banda visual, proporcione una descripción auditiva de la información importante de la banda visual de una presentación multimedia.

[1.4](#) Para toda presentación multimedia tempodependiente (por ejemplo, una película o animación) sincronice alternativas equivalentes (por ejemplo, subtítulos o descripciones de la banda visual) con la presentación.

Y si todo lo demás falla

[11.4](#) Si, después de los mayores esfuerzos, no puede crear una página accesible, proporcione un vínculo a una página alternativa que use tecnologías W3C, sea accesible, tenga información (o funcionalidad) equivalente y sea actualizada tan a menudo como la página (original) inaccesible.

Puntos de verificación Prioridad 2:

En general

[2.2](#) Asegúrese de que las combinaciones de los colores de fondo y primer plano tengan el suficiente contraste para que sean percibidas por personas con deficiencias de percepción de color o en pantallas en blanco y negro [Prioridad 2 para las imágenes. Prioridad 3 para los textos].

[3.1](#) Cuando exista un marcador apropiado, use marcadores en vez de imágenes para transmitir la información.

[3.2](#) Cree documentos que estén validados por las gramáticas formales publicadas.

[3.3](#) Utilice hojas de estilo para controlar la maquetación y la presentación.

[3.4](#) Utilice unidades relativas en lugar de absolutas al especificar los valores en los atributos de los marcadores de lenguaje y en los valores de las propiedades de las hojas de estilo.

[3.5](#) Utilice elementos de encabezado para transmitir la estructura lógica y utilícelos de acuerdo con la especificación.

[3.6](#) Marque correctamente las listas y los ítems de las listas.

[3.7](#) Marque las citas. No utilice el marcador de citas para efectos de formato tales como sangrías.

[6.5](#) Asegúrese de que los contenidos dinámicos son accesibles o proporcione una página o presentación alternativa.

[7.2](#) Hasta que las aplicaciones de usuario permitan controlarlo, evite el parpadeo del contenido (por ejemplo, cambio de presentación en periodos regulares, así como el encendido y apagado).

[7.4](#) Hasta que las aplicaciones de usuario proporcionen la posibilidad de detener las actualizaciones, no cree páginas que se actualicen automáticamente de forma periódica.

[7.5](#) Hasta que las aplicaciones de usuario proporcionen la posibilidad de detener el redireccionamiento automático, no utilice marcadores para redirigir las páginas automáticamente. En su lugar, configure el servidor para que ejecute esta posibilidad.

[10.1](#) Hasta que las aplicaciones de usuario permitan desconectar la apertura de nuevas ventanas, no provoque apariciones repentinas de nuevas ventanas y no cambie la ventana actual sin informar al usuario.

[11.1](#) Utilice tecnologías W3C cuando estén disponibles y sean apropiadas para la tarea y use las últimas versiones que sean soportadas.

[11.2](#) Evite características desaconsejadas por las tecnologías W3C.

[12.3](#) Divida los bloques largos de información en grupos más manejables cuando sea natural y apropiado.

[13.1](#) Identifique claramente el objetivo de cada vínculo.

[13.2](#) Proporcione metadatos para añadir información semántica a las páginas y sitios.

[13.3](#) Proporcione información sobre la maquetación general de un sitio (por ejemplo, mapa del sitio o tabla de contenidos).

[13.4](#) Utilice los mecanismos de navegación de forma coherente.

Y si utiliza tablas

[5.3](#) No utilice tablas para maquetar, a menos que la tabla tenga sentido cuando se alinee. Por otro lado, si la tabla no tiene sentido, proporcione una alternativa equivalente (la cual debe ser una versión alineada).

[5.4](#) Si se utiliza una tabla para maquetar, no utilice marcadores estructurales para realizar un efecto visual de formato.

Y si utiliza marcos ("frames")

[12.2](#) Describa el propósito de los marcos y cómo éstos se relacionan entre sí, si no resulta obvio solamente con el título del marco.

Y si utiliza formularios

[10.2](#) Hasta que las aplicaciones de usuario soporten explícitamente la asociación entre control de formulario y etiqueta, para todos los controles de formularios con etiquetas asociadas implícitamente, asegúrese de que la etiqueta está colocada adecuadamente.

[12.4](#) Asocie explícitamente las etiquetas con sus controles.

Y si utiliza "applets" y "scripts"

[6.4](#) Para los scripts y applets, asegúrese de que los manejadores de eventos sean independientes del dispositivo de entrada.

[7.3](#) Hasta que las aplicaciones de usuario permitan congelar el movimiento de los contenidos, evite los movimientos en las páginas.

[8.1](#) Haga los elementos de programación, tales como scripts y applets, directamente accesibles o compatibles con las ayudas técnicas [Prioridad 1 si la funcionalidad es importante y no se presenta en otro lugar; de otra manera, Prioridad 2].

[9.2](#) Asegúrese de que cualquier elemento que tiene su propia interfaz pueda manejarse de forma independiente del dispositivo.

[9.3](#) Para los "scripts", especifique manejadores de evento lógicos mejor que manejadores de evento dependientes de dispositivos.

Puntos de verificación Prioridad 3:

En general

[4.2](#) Especifique la expansión de cada abreviatura o acrónimo cuando aparezcan por primera vez en el documento.

[4.3](#) Identifique el idioma principal de un documento.

[9.4](#) Cree un orden lógico para navegar con el tabulador a través de vínculos, controles de formulario y objetos.

[9.5](#) Proporcione atajos de teclado para los vínculos más importantes (incluidos los de los mapas de imagen de cliente), los controles de formulario y los grupos de controles de formulario.

[10.5](#) Hasta que las aplicaciones de usuario (incluidas las ayudas técnicas) interpreten claramente los vínculos contiguos, incluya caracteres imprimibles (rodeados de espacios), que no sirvan como vínculo, entre los vínculos contiguos.

[11.3](#) Proporcione la información de modo que los usuarios puedan recibir los documentos según sus preferencias (por ejemplo, idioma, tipo de contenido, etc.).

[13.5](#) Proporcione barras de navegación para destacar y dar acceso al mecanismo de navegación.

[13.6](#) Agrupe los vínculos relacionados, identifique el grupo (para las aplicaciones de usuario) y, hasta que las aplicaciones de usuario lo hagan, proporcione una manera de evitar el grupo.

[13.7](#) Si proporciona funciones de búsqueda, permita diferentes tipos de búsquedas para diversos niveles de habilidad y preferencias.

[13.8](#) Localice la información destacada al principio de los encabezamientos, párrafos, listas, etc.

[13.9](#) Proporcione información sobre las colecciones de documentos (por ejemplo, los documentos que comprendan múltiples páginas).

[13.10](#) Proporcione un medio para saltar sobre un ASCII art de varias líneas.

[14.2](#) Complemente el texto con presentaciones gráficas o auditivas cuando ello facilite la comprensión de la página.

[14.3](#) Cree un estilo de presentación que sea coherente para todas las páginas.

Y si utiliza imágenes o mapas de imagen

[1.5](#) Hasta que las aplicaciones de usuario interpreten el texto equivalente para los vínculos de los mapas de imagen de cliente, proporcione vínculos de texto redundantes para cada zona activa del mapa de imagen de cliente.

Y si utiliza tablas

[5.5](#) Proporcione resúmenes de las tablas.

[5.6](#) Proporcione abreviaturas para las etiquetas de encabezamiento.

[10.3](#) Hasta que las aplicaciones de usuario (incluidas las ayudas técnicas) interpreten correctamente los textos contiguos, proporcione un texto lineal alternativo (en la página actual o en alguna otra) para *todas* las tablas que maquetan texto en paralelo, en columnas de palabras.

Y si utiliza formularios

[10.4](#) Hasta que las aplicaciones de usuario manejen correctamente los controles vacíos, incluya caracteres por defecto en los cuadros de edición y áreas de texto.

Anexo 6. Correspondencia entre la norma UNE 139803:2004 y los puntos de verificación WAI

El contenido de la Norma UNE 139803:2004 está basado en los puntos de control definidos en las Directrices de Accesibilidad de los Contenidos Web de WAI (WCAG), en su versión 1.0.

La siguiente tabla permite realizar consultas de los puntos de control de WAI partiendo de los requisitos de la Norma UNE 139803. Esta tabla está organizada por los apartados de la norma española y por la prioridad de cada uno de los requisitos.

| | Principios generales | | Presentación | | Estructura | | Contenido | | Navegación | | Scripts, objetos de programación y multimedia | | Situaciones excepcionales | |
|-------------|----------------------|------|--------------|------|------------|------|-----------|-------|------------|------|---|---------|---------------------------|------|
| | UNE | WAI | UNE | WAI | UNE | WAI | UNE | WAI | UNE | WAI | UNE | WAI | UNE | WAI |
| Prioridad 1 | | | 4.2.1 | 6.1 | 4.3.1 | 12.1 | 4.4.1 | 6.2 | 4.5.1 | 13.1 | 4.6.1 | 6.3 | 4.7.1 | 11.4 |
| | | | 4.2.2 | 2.1 | 4.3.2 | 5.1 | 4.4.2 | 14.1 | 4.5.2 | 9.1 | 4.6.2 | 8.1 | | |
| | | | 4.2.3 | 7.1 | 4.3.3 | 5.2 | 4.4.3 | 1.1 | 4.5.3 | 1.2 | 4.6.3 | 1.3 | | |
| | | | | | | | 4.4.4 | 4.3 | | | 4.6.4 | 1.4 | | |
| | | | | | | | 4.4.5 | 4.1 | | | | | | |
| Prioridad 2 | 4.1.1 | 11.1 | 4.2.4 | 3.4 | 4.3.4 | 12.2 | 4.4.6 | 6.5 | 4.5.4 | 7.4 | 4.6.5 | 8.1 | | |
| | 4.1.2 | 3.2 | 4.2.5 | 3.3 | 4.3.5 | 3.5 | 4.4.7 | 12.4 | 4.5.5 | 7.5 | 4.6.6 | 6.4/9.3 | | |
| | 4.1.3 | 11.2 | 4.2.6 | 3.1 | 4.3.6 | 12.3 | 4.4.8 | 10.2 | 4.5.6 | 10.1 | 4.6.7 | 6.4/9.2 | | |
| | 4.1.4 | 13.2 | 4.2.7 | 2.2 | 4.3.7 | 3.6 | | | 4.5.7 | 13.3 | | | | |
| | | | 4.2.8 | 5.3 | 4.3.8 | 3.7 | | | 4.5.8 | 13.4 | | | | |
| | | | 4.2.9 | 5.4 | 4.3.9 | 5.5 | | | 4.5.9 | 9.4 | | | | |
| | | | 4.2.10 | 7.2 | | | | | | | | | | |
| | | | 4.2.11 | 7.3 | | | | | | | | | | |
| Prioridad 3 | | | 4.2.12 | 14.3 | 4.3.10 | 5.6 | 4.4.9 | 13.8 | 4.5.10 | 13.9 | | | | |
| | | | 4.2.13 | 2.2 | | | 4.4.10 | 4.2 | 4.5.11 | 13.5 | | | | |
| | | | | | | | 4.4.11 | 14.2 | 4.5.12 | 13.7 | | | | |
| | | | | | | | 4.4.12 | 13.10 | 4.5.13 | 13.6 | | | | |
| | | | | | | | 4.4.13 | 10.3 | 4.5.14 | 10.5 | | | | |
| | | | | | | | 4.4.14 | 10.4 | 4.5.15 | 9.5 | | | | |
| | | | | | | | 4.4.15 | 11.3 | 4.5.16 | 1.5 | | | | |

