

# Herramienta Software para Administrar el Desarrollo de Sitios Web Accesibles según Pautas WCAG 1.0

Juan I. Cavalieri, Pedro L. Alfonzo, Sonia I. Mariño, Maria V. Godoy

Facultad de Ciencias Exactas y Naturales y Agrimensura. Universidad Nacional del Nordeste. Argentina, 9 de Julio 1449, 3400 Corrientes, Argentina

juanicav@hotmail.com, plalfonzo@hotmail.com, simarinio@yahoo.com, mvvgg2001@yahoo.com

**Resumen-** La accesibilidad a los contenidos web, es un campo en desarrollo y representa una manera de encarar proyectos con posibilidades de crecimiento y perdurabilidad en el tiempo. Por lo expuesto es relevante que los desarrolladores de software y las empresas, elaboren sitios accesibles, por lo cual el trabajo propuesto contribuye a esta temática. Se presenta una herramienta software orientada al análisis de las pautas WCAG 1.0, a ser utilizada en etapas tempranas de desarrollo de proyectos software, siendo los resultados registrados y desplegados en distintos formatos (reportes y gráficos).

**Palabras clave:** Accesibilidad Web, Validadores, Desarrollo web, WCAG 1.0.

## I. INTRODUCCIÓN

En la sociedad del conocimiento, las empresas demandan servicios que respondan de manera rápida a las peticiones de los usuarios y sean sólidos para permanecer operativo el mayor tiempo posible, a pesar de las fallas o la falta de seguridad inherente para transmitir información de alta confidencialidad: siendo mínimas las restricciones solicitadas referentes a la accesibilidad a sus contenidos.

Se coincide con [1], en que la accesibilidad a los contenidos web, es un campo en desarrollo y atendible, considerando que representa una manera de encarar proyectos con posibilidades de crecimiento y perdurabilidad en el tiempo.

Por otra parte, la AW en la República Argentina está logrando mayor difusión desde que en el mes de noviembre del año 2010 se proclamó la Ley N° 26.653 [2], de Accesibilidad a la Información y se aprobó su reglamentación en el año 2013 [3].

Para verificar si un sitio web es accesible, existen herramientas automáticas disponibles en la web que evalúan y elaboran informes detallados del cumplimiento de los puntos de verificación de las Pautas de Accesibilidad al Contenido en la Web 1.0 o WCAG 1.0 [4], y las Pautas de Accesibilidad al Contenido en la Web 1.0 o WCAG 2.0 [5] propuestas por la W3C [6], como ser TAW [7], HERA [8], EXAMINATOR [9], UNICORNIO [10], Access Monitor [11], Tingtun Checkers [12], F.A.E. 2.0. [13], Wave [14], Cinthia Says [15], AChecker [16].

Las herramientas mencionadas realizan un análisis completo del sitio, pero requieren del control en forma manual del evaluador del sitio web, en aquellos puntos de verificación en que se carece de certeza del cumplimiento.

Como una superación a estos validadores web, se han desarrollado herramientas software disponible para apoyar la gestión de las normas de AW. Entre ellas se mencionan los trabajos descriptos en [17] y [18].

En [17], se presenta una herramienta de ayuda para el análisis de sitios web, permite tanto la evaluación semi-

automática como la gestión de los procesos completos de evaluación. Facilita la evaluación de la accesibilidad mediante la comprobación de las pautas WCAG 1.0.

En [18], se reporta un prototipo de sistema experto basado en casos para la generación de reportes de accesibilidad de páginas web atendiendo las pautas WCAG 2.0.

Por lo expuesto, es importante que los desarrolladores de software y las empresas, elaboren sitios accesibles, por lo cual el trabajo propuesto contribuye a esta temática. Se propone una herramienta software orientada al análisis de las pautas WCAG 1.0, a ser utilizada en etapas tempranas de desarrollo de proyectos software; y de esta manera aseguraría la accesibilidad a los contenidos del sitio desarrollado, antes de su puesta en producción.

## II. METODOLOGÍA

La metodología utilizada el desarrollo de la herramienta informática que permite gestionar la accesibilidad a los contenidos en proyectos software consistió en:

Etapa 1: Revisión, selección y estudio de herramientas para evaluar la accesibilidad a los contenidos web.

Se realizó la revisión, selección y estudio de herramientas automáticas propuestas por la W3C [6], para evaluar la accesibilidad web.

Etapa 2: Selección del validador HERA [8] la cual realiza el análisis de sitios web basándose en la normativa WCAG 1.0., permitiendo acceder a su código fuente.

Etapa 3: Desarrollo del software HADWA (Herramienta de Apoyo al Desarrollo Web Accesible)

Se procedió a la construcción del software como soporte al análisis de la WCAG 1.0 [4] de sitios web. Para el análisis de la accesibilidad, incluida en el software de gestión se basó en el código fuente de la herramienta HERA.

## III. RESULTADOS

### A. Descripción de la herramienta

El producto software se desarrolló considerando una secuencia de funcionamiento (Fig. 1), vinculada estrechamente con la definición exacta de funciones que desarrolla cada uno de los usuarios del sistema: i) Programador; ii) Revisor; iii) Jefe de testing y iv) Administrador.

Figura 1. Esquema de funcionamiento del sistema.

Se observa en la Fig. 1 que el proceso consiste en iniciar una revisión – usuario Programador- para luego ser evaluada en profundidad por personas especializadas –usuario Revisor-, quienes realizan las verificaciones manuales que la herramienta no pueda comprobar y emiten los informes necesarios. Estos informes pasan al jefe de testing quien decide si se deben

realizar modificaciones en el sitio o si se encuentra aprobado según las normas aplicadas.

En el caso de que la revisión no culmine de forma exitosa y requiera de modificaciones, se retroalimentarán los resultados hacia el comienzo del proceso. En este punto se deberá comenzar una nueva revisión sobre la página, atendiendo a las modificaciones y resultados provenientes del informe de revisión. Existe un sector denominado “Administración” que si bien no participa en forma activa del proceso de revisión de una página web, constituye un nexo de comunicación entre los diferentes usuarios que participan en el sistema, para supervisar el funcionamiento general de la herramienta.

Cabe destacar que al solicitar la revisión de una página web, el sistema verifica su disponibilidad ante una revisión pendiente y, en caso de ser posible asigna en forma automática a un revisor considerando la cantidad de trabajo pendiente. De esta manera se asegura la distribución del trabajo en forma equitativa y automática entre los revisores de la organización.

Otra funcionalidad disponible es la sección de informes estadísticos. Se destaca dado que las herramientas analizadas carecen de una opción similar y se limitan a realizar una evaluación del sitio y desplegar un resultado.

El modelo de diseño que guió el desarrollo se fundamentó en la determinación de funciones específicas para asignar un experto en el proceso de evaluación pertinente. Es aconsejable que quien desarrolla y valida el sitio sean diferentes personas, para asegurar el correcto desempeño de las funcionalidades definidas en la herramienta software.

### B. Funcionamiento de la herramienta

A continuación se describen los diferentes módulos que componen el sistema desarrollado y los perfiles de usuario que acceden al mismo.

#### ▪ Modulo “Nueva Revisión” - perfil “Programador-

Permite enviar la revisión de una página web (Fig. 2). El programador solicitará una nueva revisión al finalizar el proceso de programación de una página.

Al pulsar el botón “Revisar” se pueden obtener 3 resultados diferentes. i) se informa que la dirección URL ingresada no es consistente o no existe (Fig. 3); ii) la página se encuentra en un proceso de evaluación aún no finalizado, por lo que se deberá esperar a que termine el proceso completo antes de ingresar nuevamente la dirección de la página (Fig. 4) y iii) se informará si el resultado fue exitoso (Fig. 5). En tal caso, el sistema evaluará la página ingresada y almacenará los datos en la base de datos. De esta manera se logra disponer de los resultados del análisis automático su análisis y empleo en cualquier momento sin depender de la herramienta para la evaluación. Así, en la siguiente sección del esquema de funcionamiento de la herramienta, el revisor tendrá que recuperar los datos almacenados para continuar con la evaluación y no tener que considerar el tiempo que implica analizar cada una de las pautas. Además, se ingresan otros datos como la fecha, programador encargado y otros datos automáticos que resulten de interés. Por último, la herramienta asigna de manera automática al revisor encargado de continuar con la evaluación. Esto lo hace tomando aquellas pendientes de cada uno de los revisores que componen el grupo de trabajo y considera con mayor posibilidad a aquel que contenga menos trabajos pendientes.

Aun cuando el usuario programador no puede realizar evaluaciones, la herramienta despliega aquellas pendientes de finalización sin la posibilidad de interactuar con ellas. Para ello, se brinda la posibilidad de contar con un filtro previo para

realizar una selección más acotada. Luego se mostrarán las evaluaciones pendientes que cumplan con el criterio de selección indicado y los datos que puedan resultar relevantes para el programador, tal como se puede observar en la Fig. 6.

#### ▪ Modulo “Revisar Página” – Perfil “Revisor”-

El modulo recupera los datos almacenados, atinentes a la solicitud previa de revisión establecida en la sección anterior; el revisor deberá verificar y completar aquellas pautas WCAG 1.0 no evaluadas en forma automática. Para ello, debe seleccionar la prioridad de evaluación con la que se trabajará (Fig. 7). Esto es de gran importancia dado que permite al usuario poder continuar con cualquier grupo de prioridades sin tener que realizar el análisis en forma secuencial. Además, se presenta una breve descripción de la pauta analizada, a fin de conceptualizar lo revisado. En color verde se observan los resultados correctos, mientras que en rojo se indican los resultados que no superaron el análisis. En general las páginas no cuentan con todas las pautas implementadas, por lo que se indica en negro con la leyenda “no aplicable” cuando no corresponde la verificación de dicho punto. Por último, entre las opciones de evaluación posible, se encuentran las pautas que deben ser analizadas en forma manual. Esto se debe a que es necesario interpretar y evaluar según el juicio de un revisor especializado, algo que el sistema no puede realizar por sí solo. Esto se verá indicado en un campo de selección de color amarillo de manera que resalte y no pase desapercibido, y en él se podrá seleccionar la opción que mejor se adapte según lo evaluado. Todo lo descripto se puede visualizar en la Fig. 8, donde se observa parte del análisis de una página web. Además, se permite el ingreso de comentarios si se considera necesario. Una vez que el revisor ha finalizado la evaluación, deberá indicar el fin de dicho proceso al final de la última prioridad. Allí cambiará el campo de selección de manera que los datos sean almacenados de forma definitiva y cambien su estado actual de “Pendientes” a “Finalizados”. Lo expuesto permite al programador generar una nueva evaluación sobre la página y al revisor obtener una nueva tarea.

Al igual que en la sección anterior, el revisor puede tener acceso a las evaluaciones finalizadas en forma restringida. Ingresando a la sección de “Revisiones finalizadas” podrá ver el estado actual de las evaluaciones que fueron realizadas, junto con los datos que resulten relevantes. Esto permite que pueda conocer el estado de una evaluación realizada pero no tenga acceso a la generación de informes.

#### ▪ Modulo “Generar informes” – Perfil “Jefe de testing”

Este módulo ingresa al final del proceso de evaluación de una página. Brinda la información necesaria para determinar si su resultado fue exitoso o requiere modificaciones futuras. El actor denominado “Jefe de testing” es quien puede visualizar las revisiones finalizadas y generar los informes necesarios. Éstos serán remitidos en versión impresa para el análisis y toma de decisiones.

Al igual que en el caso del programador, el jefe de testing puede observar las revisiones pendientes sin tener acceso a las mismas. Lo expuesto permite llevar un control y seguimiento, se puede observar la fecha de solicitud y la última fecha de revisión, entre otros. Además, se ofrece la posibilidad de disponer de un filtro previo para una selección más acotada.

En la sección de revisiones finalizadas, un filtro de selección permite agrupar a las páginas por proyecto, seleccionar un programador, seleccionar un revisor o restringir la búsqueda a un rango de fechas posibles (Fig. 9). La Fig. 10 permite visualizar las revisiones finalizadas que cumplen con el patrón de búsqueda ingresado en el filtro, se observa que, junto

a la información referente a la revisión, se dispone de las opciones “ver informe” y “ver estadísticas”.

Seleccionando la opción “Ver informe” (Fig. 10), se despliega la interfaz de la Fig. 11, con información correspondiente a la revisión de la página seleccionada y los comentarios del revisor. Se visualiza por prioridad, de manera que la información se despliegue ordenada, facilitando la lectura. Al finalizar el informe (sin importar la prioridad seleccionada) se puede optar por “Generar informe”. Pulsando en el mismo, se pueden solicitar los datos necesarios para completar el informe: comentarios del jefe de testing y solicitud de modificaciones (Fig. 12). Una vez rellenos los campos e indicada su finalización, se muestra la versión final, la cual incluye los datos previamente ingresados y los puntos de verificación sin la posibilidad de seleccionar prioridad. Información desplegada como vista previa de impresión del informe generado. Lo expuesto se visualiza en la Fig. 13. Si el jefe de testing define como listo el informe final, debe seleccionar la opción “Descargar” (Fig. 14), versión que podrá observar un breve título informativo y los datos de las etapas previas a la descarga. Una información relevante es la adición al final del documento del detalle de los puntos que no superaron la revisión. Esto permite al programador encargado disponer de un mayor detalle de las pautas que deben modificarse o corregirse (Figuras 15 y 16).

Como se mencionó anteriormente, se ofrece la posibilidad de visualizar algunos datos estadísticos sobre la página seleccionada.

Filtradas las opciones, en la sección “Ver estadísticas”, se observa la distribución de resultado en los diferentes puntos de verificación. Internamente el sistema inicializa un contador para determinar cuántos puntos finalizaron con resultado “Bien”, “Mal”, “No aplicable” y “No verificado”. Además se muestra la duración en días de la página seleccionada desde la solicitud de revisión hasta su finalización.

Finalmente, se muestra el número de veces que la página ha ingresado al circuito funcional de revisión y la evolución de las páginas, luego de sucesivas mejoras. Como se muestra en la Fig. 17, los datos se grafican para facilitar la comprensión.

▪ Modulo “Administrar” – Perfil “Administrador”

Este módulo gestiona las funciones básicas de los usuarios: alta, baja y modificación, no interviene en forma activa en el proceso de evaluación de una página web. Además, se puede tener acceso al listado de revisiones pendientes y finalizadas sin la posibilidad de actuar sobre ellas. De esta manera, el “Administrador” actúa como nexo de comunicación entre los diversos usuarios y la supervisión del funcionamiento del sistema.

La opción “Listado de activos” (Fig. 18) despliega el listado de los usuarios en condición de utilizar la herramienta. Sobre el margen derecho, se dispone de la opción “Más información” para ver los datos complementarios de cada usuario. La opción “Modificar” permite cambiar o actualizar los datos del usuario.

La opción “Listado de inactivos” despliega los usuarios dados de baja (Fig. 19). Al tratarse una baja lógica, se podrá habilitarlos nuevamente para continuar utilización la herramienta.

#### IV. RESULTADOS

La sociedad del conocimiento promueve el acceso abierto a los datos, la información y los códigos software a través de diversas normativas.

En lo atinente a la AW, las, leyes vigentes y el constante crecimiento de soluciones basadas en los servicios de internet demandan la necesidad de satisfacer las normas de accesibilidad WCAG 1.0 y WCAG 2.0. Siguiendo lo pautado en la normativa existen puntos de evaluación manual, es decir, que requieren la intervención de un especialista en el tema; mientras que aquellos puntos de evaluación automática son medidos por la herramienta descrita en este trabajo

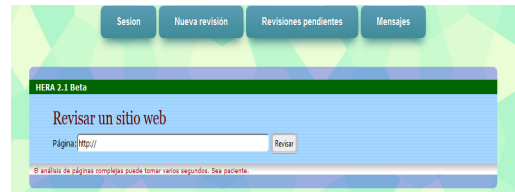


Fig. 2. Interfaz de ingreso de URL.

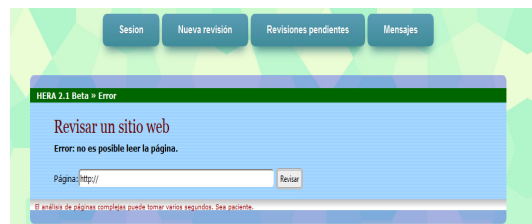


Fig. 3. Interfaz de error en URL ingresada.

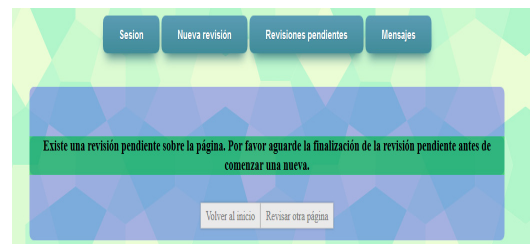


Fig. 4. Interfaz de revisión pendiente sobre la página ingresada.

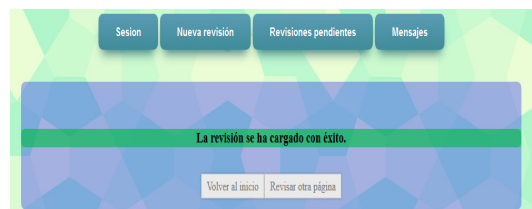


Fig. 5. Interfaz de revisión solicitada correctamente.

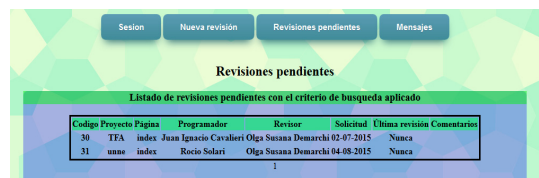


Fig. 6. Interfaz de revisiones pendientes por programador.

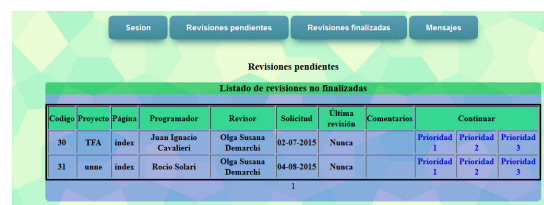


Fig. 7. Interfaz de selección de revisión pendiente a trabajar.

Proyecto: unne  
 Pagina revisada: index  
 Programador: Rocio Solari  
 Revisor: Olga Susana Demarchi  
 Fecha de solicitud: 04-08-2015  
 Última Revisión: Nunca

Comentarios:

**Resultados obtenidos**

Punto	Prioridad	Breve descripción	Estado
1.1	1	Proporcione un texto equivalente para todo elemento no textual	Mal
1.1.01	1	Imágenes	--
1.1.02	1	Botones gráficos	No aplicable
1.1.03	1	Áreas de mapas de imágenes	Mal
1.1.04	1	Scripts	--
1.1.05	1	Elementos incrustados	No aplicable
1.1.06	1	Applets	No aplicable
1.1.07	1	Objetos	No aplicable
1.1.08	1	Marcos en línea	No aplicable
1.1.09	1	Archivos de sonido	No aplicable

Fig. 8. Interfaz de proceso de revisión.

[Sesion](#)   [Revisiones pendientes](#)   [Revisiones finalizadas](#)   [Mensajes](#)

**Revisiones finalizadas**

**Listado de revisiones finalizadas**

Programador:   
 Revisor:   
 Proyecto:   
 Fecha de solicitud:  A

Nota: al dejar todos los campos vacíos, no se aplicará ningún filtro

Fig. 9. Interfaz de filtro de revisiones finalizadas

[Sesion](#)   [Revisiones pendientes](#)   [Revisiones finalizadas](#)   [Mensajes](#)

**Revisiones finalizadas**

**Listado de revisiones finalizadas con el criterio de búsqueda aplicado**

Código	Proyecto	Página	Programador	Revisor	Solicitud	Última revisión	
27	unne	index	Juan Ignacio Cavalieri	Olga Susana Demarchi	25-06-2015	25-06-2015	<a href="#">Ver informe</a> <a href="#">Ver estadísticas</a>

1

Fig. 10. Interfaz de selección de revisiones finalizadas

Proyecto: unne  
 Pagina revisada: index  
 Programador: Juan Ignacio Cavalieri  
 Revisor: Olga Susana Demarchi  
 Fecha de solicitud: 25-06-2015  
 Última Revisión: 25-06-2015

Comentarios del revisor:

**Resultados obtenidos**

Punto	Prioridad	Breve descripción	Estado
1.1	1	Proporcione un texto equivalente para todo elemento no textual	Mal
1.1.01	1	Imágenes	Bien
1.1.02	1	Botones gráficos	No aplicable
1.1.03	1	Áreas de mapas de imágenes	Mal
1.1.04	1	Scripts	Bien
1.1.05	1	Elementos incrustados	No aplicable
1.1.06	1	Applets	No aplicable
1.1.07	1	Objetos	No aplicable

Fig. 2. Interfaz de primera etapa del informe final

Sesion Revisiones pendientes Revisiones finalizadas Mensajes

### Revisiones finalizadas

Por favor antes de generar el informe complete la siguiente información

Comentarios:

Requiere modificaciones:

SI

Finalizar informe

Fig. 3. Interfaz de segunda etapa del informe final

### Informe final de accesibilidad

Proyecto: unne  
 Pagina revisada: index  
 Programador: Juan Ignacio Cavalieri  
 Revisor: Olga Susana Demarchi  
 Jefe de testing: Oscar Alberto Cavalieri  
 Fecha de solicitud: 25-06-2015  
 Última Revisión: 25-06-2015  
 Requiere modificación: NO

Comentarios del revisor:

Comentarios del Jefe de testing:

Prioridad A

Punto	Prioridad	Breve descripción	Estado
1.1	1	Proporcione un texto equivalente para todo elemento no textual	Mal
1.1.01	1	Imágenes	Bien

Fig. 4. Interfaz de la tercera etapa del informe final.

		formulario y los grupos de controles de formulario.	
10.3	3	Hasta que las aplicaciones de usuario (incluidas las ayudas técnicas) interpreten correctamente los textos contiguos, proporcione un texto lineal alternativo para todas las tablas que maquetan texto en paralelo, en columnas de palabras.	No verificado
10.4	3	Hasta que las aplicaciones de usuario manejen correctamente los controles vacios, incluya caracteres por defecto en los cuadros de edición y áreas de texto.	No aplicable
10.5	3	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.	Bien
11.3	3	Proporcione la información de modo que los usuarios puedan recibir los documentos según sus preferencias.	No verificado
13.5	3	Proporcione barras de navegación para destacar y dar acceso al mecanismo de navegación.	No verificado
13.6	3	Agrupe los vinculos 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.	No verificado
13.7	3	Si proporciona funciones de búsqueda, permita diferentes tipos de búsquedas para diversos niveles de habilidad y preferencias.	No verificado
13.8	3	Localice la información destacada al principio de los encabezamientos, párrafos, listas, etc.	No verificado
13.9	3	Proporcione información sobre las colecciones de documentos.	No verificado
13.10	3	Proporcione un medio para saltar sobre un ASCII art de varias líneas.	No verificado
14.2	3	Complemente el texto con presentaciones gráficas o auditivas cuando ello facilite la comprensión de la página.	No verificado
14.3	3	Cree un estilo de presentación que sea coherente para todas las páginas.	No verificado

Descargar

Fig. 5. Interfaz del pie de la tercera etapa del informe final.

## Informe de accesibilidad de la página index, correspondiente al proyecto denominado unne

### Análisis realizado bajo la normativa WCAG 1.0

Proyecto:	unne		
Página revisada:	index		
Programador:	Juan Ignacio Cavalieri		
Revisor:	Olga Susana Demarchi		
Jefe de testing:	Oscar Alberto Cavalieri		
Fecha de solicitud:	25-06-2015		
Última Revisión:	25-06-2015		
Requiere modificación:	NO		
Comentarios del revisor:			
Comentarios del Jefe de testing:			
<b>Prioridad A</b>			
Punto	Prioridad	Breve descripción	Estado
1.1	1	Proporcione un texto equivalente para todo elemento no textual	Mal
1.1 01	1	Imágenes	Bien
1.1 02	1	Botones gráficos	No aplicable
1.1 03	1	Áreas de mapas de imágenes	Mal
1.1 04	1	Scripts	Bien
1.1 05	1	Elementos incrustados	No aplicable
1.1 06	1	Applets	No aplicable
1.1 07	1	Objetos	No aplicable
1.1 08	1	Marcos en línea	No aplicable
1.1 09	1	Archivos de sonido	No aplicable
1.1 10	1	Archivos multimedia	No aplicable
1.1 11	1	Marcos	No aplicable

Fig. 6. Interfaz de parte superior del informe final descargado.

Detalle de los puntos a corregir
<p>Punto 1.1 03: Cada área de un mapa de imagen (un elemento asociado a &lt;img&gt;, &lt;object&gt; o &lt;input&gt; que define zonas activas, usualmente utilizadas como enlaces) debe llevar el atributo "alt" con un texto alternativo que describa el propósito del área.</p>
<p>Punto 5.1: Se identifican las tablas y sus celdas. Las celdas de encabezados (&lt;th&gt;) se destacan con un fondo cyan. Verifique que esos encabezados se utilicen sólo en tablas de datos.</p>
<p>Punto 5.2: Se identifican las tablas y sus celdas. Las celdas de datos (&lt;td&gt;) y las de encabezados (&lt;th&gt;) que tengan algún atributo de asociación se destacan con un fondo cyan.</p>
<p>Punto 3.2 02: HERA utiliza los servicios del revisor de CSS del W3C para verificar la sintaxis de las hojas de estilo. Si el resultado es indefinido, utilice el icono para abrir la página del revisor.</p>
<p>Punto 3.3 01: Verifique que el maquetado y presentación de la página se realicen mediante hojas de estilo y no con elementos y atributos del lenguaje de marcas.</p>
<p>Punto 3.3 03: Verifique que no se utilicen atributos de presentación como "color", "bgcolor" o "face", por ejemplo.</p>
<p>Punto 3.4 01: Se indican las tablas y celdas de tablas con atributos como "height" y "width". Verifique que los valores para esos atributos estén expresados en porcentajes, no en medidas absolutas.</p>
<p>Punto 3.4 02: Se indican las tablas y celdas de tablas con atributos como "height" y "width". Verifique que los valores para esos atributos estén expresados en porcentajes, no en medidas absolutas.</p>
<p>Punto 3.5: Se identifican los encabezados. Debe existir, al menos un título principal y no deben mezclarse los niveles de encabezados aleatoriamente sólo para aprovechar que se presentan en distintos tamaños de fuente.</p>
<p>Punto 3.6: Se identifican los distintos tipos de lista. Todo contenido que conforme una lista de elementos debe estar marcado como tal lista y no debe utilizarse el marcado de lista para conseguir efectos de presentación (Sangrado, por ejemplo). Si existen listas anidadas, debe comprobarse el correcto anidamiento de las mismas.</p>
<p>Punto 3.7: Se identifican los elementos &lt;q&gt; y &lt;blockquote&gt;. Se debe verificar que aquello marcado como cita sea realmente una cita y que no se utilicen estos elementos sólo para dar formato al texto, por ejemplo, para conseguir que el texto aparezca sangrado.</p>
<p>Punto 7.2: Se señalan el elemento &lt;blink&gt;, los scripts y los elementos de programación que pueden causar un efecto de destello. Verifique además, a simple vista, que no haya elementos que destellen.</p>

Fig. 7. Interfaz final del informe.

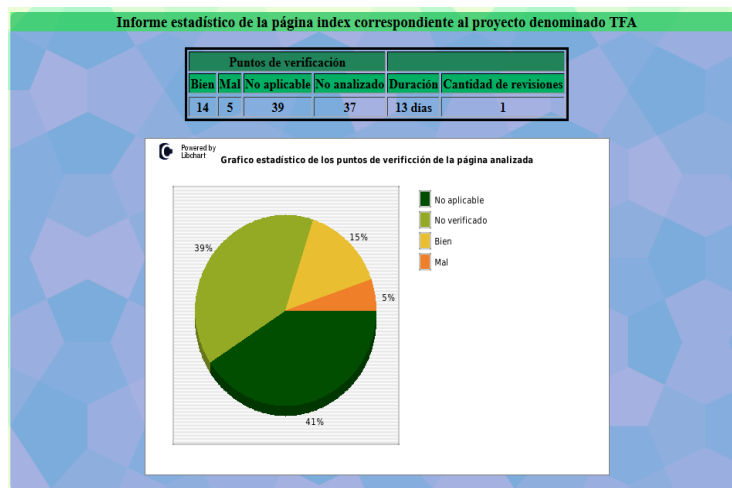


Fig. 8. Informe estadístico.

[Sesion](#)   [Mensajes](#)   [Usuarios](#)   [Revisiones pendientes](#)   [Revisiones finalizadas](#)

**Panel de administración de usuarios**

[Alta](#)   [Listado de activos](#)   [Listado de inactivos](#)

**Listado de de usuarios activos**

DNI	Apellido	Nombres	Mail	Trabajos pendientes			
35438477	Cavalieri	Juan Ignacio	juanicav@hotmail.com	0	Modificar	Dar de baja	Mas información
8219701	Cavalieri	Oscar Alberto	oscaracav@gmail.com	0	Modificar	Dar de baja	Mas información
11240784	Demarchi	Olga Susana	olgacavalieri@hotmail.com	2	Modificar	Dar de baja	Mas información

1

[Exportar a Excel](#)

Fig. 18. Interfaz de listado de usuarios activos.

**Panel de administración de usuarios**

[Alta](#)   [Listado de activos](#)   [Listado de inactivos](#)

**Listado de de usuarios inactivos**

DNI	Apellido	Nombres	Mail	Trabajos pendientes			
37326818	Solari	Rocio		0	Modificar	Dar de alta	Mas información

1

[Exportar a Excel](#)

Fig. 19. Interfaz de listado de usuarios inactivos.

Se diseñó y desarrolló una herramienta para la evaluación de la AW de sitios web locales y remotos, siendo los resultados registrados y desplegados en distintos formatos (reportes y gráficos).

La propuesta se ha basado en herramienta de AW de código abierto aportando a la comunidad del software libre. Entre las ampliaciones incorporadas se destacan la interfaz de usuario que establece la secuencia de evaluación interviniendo diferentes perfiles de usuarios expertos sin necesidad de interferir entre ellos.

Se prevé a futuro incorporar opciones orientadas a brindar diversos controles, reportes y otras funcionalidades que pudieran surgir del uso.

#### REFERENCIAS

[1] P. L. Alfonzo, S. I. Mariño, J. I. Cavalieri y A. Gomez Codutti, *Accesibilidad Web: Su abordaje en congresos argentinos de informática en el período 2012-2013*. Novatica: Revista de la asociación de Técnicos de informática. 2014, no.229.

[2] INFOLEG. (2010). Aprobación de la Reglamentación de la Ley 26.653. Disponible en: < http://www.infoleg.gov.ar/infoleg

Internet/anexos/175000-179999/175694/norma.htm> [Accedido: 10 de marzo 2016].

[3] INFOLEG. (2013). Aprobación de la Reglamentación de la Ley N° 26.653. Disponible en: <http://infoleg.gov.ar/infolegInternet/anexos/210000-214999/210143/norma.htm> [Accedido: 10 de marzo 2016].

[4] WCAG (1999). Web Content Accessibility Guidelines 1.0. Disponible en: < http://www.w3.org/TR/WCAG10/ >[ Accedido: 14 de marzo de 2016].

[5] WCAG (2008). Web Content Accessibility Guidelines 2.0. Disponible en: <http://www.w3.org/TR/WCAG20>[ Accedido: 14 de marzo de 2016].

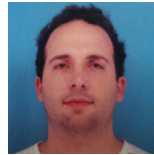
[6] W3C. Consorcio World Wide Web. Disponible en: <http://www.w3c.es/> [Accedido: 10 de marzo 2016].

[7] TAW. Test de Accesibilidad Web. Disponible en: <http://www.tawdis.net/> [ Accedido:2 de marzo 2016].

[8] HERA. Validador Automático. Disponible en: <http://www.sidar.org/hera>[ Accedido: 2 de marzo 2016].

[9] EXAMINATOR. Herramienta de evaluación automática para las Pautas de Accesibilidad para el Contenido Web (WCAG) 2.0. Disponible en: <http://www.accessible.com.ar/examinator> [Accedido: 2 de marzo 2016].

- [10] UNICORNIO Validador Unificado. Validador de código. Disponible en: <[http://validator.w3.org/unicorn/?ucn\\_lang=es.http://www.redalyc.org/pdf/904/90426810009.pdf](http://validator.w3.org/unicorn/?ucn_lang=es.http://www.redalyc.org/pdf/904/90426810009.pdf)> [Accedido: 2 de marzo 2016].
- [11] Access Monitor. Validador automático para las normas WCAG. Disponible en: <<http://www.acessibilidade.gov.pt/accessmonitor/>> [Accedido: 2 de marzo 2016].
- [12] TINGTUN CHECKER. Validador automático. Disponible en: <<http://accessibility.tingtun.no/en/pagecheck2.0/>> [Accedido: 2 de marzo 2016].
- [13] FAE. Functional Accesibility Evaluator 2.0. Validador automático. Disponible en: <<http://fae20.cita.illinois.edu/>> [Accedido: 2 de marzo 2016].
- [14] WAVE. Web Accesibility evaluation tool. Disponible en: <<http://wave.webaim.org/>> [Accedido: 2 de marzo 2016].
- [15] Cinthia Says. Validador automático. Disponible en: <<http://www.cynthiasays.com/Home.aspx>> [Accedido: 2 de marzo 2016].
- [16] Checker. Open Source Web Accesibility evaluation tool. Disponible en: <<http://www.atutor.ca/achecker/>> [Accedido: 2 de marzo 2016].
- [17] L. Mena, P. Latorre y E. Lafuente, WebA (Web Análisis): Herramienta de ayuda para el diseño y para la evaluación de la usabilidad y accesibilidad de sitios web. VII Congreso Internacional de Interacción Persona-Ordenador, 2006. Disponible en: <<http://aipo.es/articulos/4/15.pdf>> [Accedido: 15 de febrero de 2016].
- [18] M. P. Cabrera Prieto, L. Sojos y F. Jasenia, Estudio de la normativa WCAG 2.0 y análisis de accesibilidad web para la creación de un prototipo de sistema experto basado en casos, que permita generar reportes de accesibilidad de páginas web, tomando como base un sitio web ecuatoriano. Proyecto de tesis, Ingeniería en Sistemas, Universidad Politécnica Salesiana, Ecuador, 2014. Disponible en: <<http://dspace.ups.edu.ec/handle/123456789/6290>> [Accedido: 15 de febrero de 2016].



**Juan I. Cavalieri.** Lic. en Sistemas de Información. Graduado de la Facultad de Ciencias Exactas y Naturales y Agrimensura – Ex Becario de la Universidad Nacional del Nordeste (UNNE), Desarrollador software.



**Pedro L. Alfonso.** Docente-Investigador. Experto en Estadística y Computación (Facultad de Ciencias Exactas y Naturales y Agrimensura – UNNE). Especialista en Ingeniería de Software (Universidad Nacional de la Plata). Magíster en Ingeniería de Software (Universidad Nacional de

La Plata).



**Sonia I. Mariño.** Docente–Investigadora, Profesora Titular, Dedicación Exclusiva, del Departamento de Informática de la Facultad de Ciencias Exactas de la Universidad Nacional del Nordeste. Licenciada en Sistemas. Es Magíster en Informática y Computación. (UNNE - Universidad de Cantabria - España). Magíster en Epistemología y Metodología de la Investigación Científica (Facultad de Humanidades - UNNE).



**María V. Godoy.** Docente–Investigadora, Profesora Titular, Dedicación Exclusiva, del Departamento de Informática de la Facultad de Ciencias Exactas de la Universidad Nacional del Nordeste. Experta en Estadística y Computación, y Licenciada en Sistemas. Es Magíster en Informática y Computación. (UNNE – Universidad de Cantabria - España).