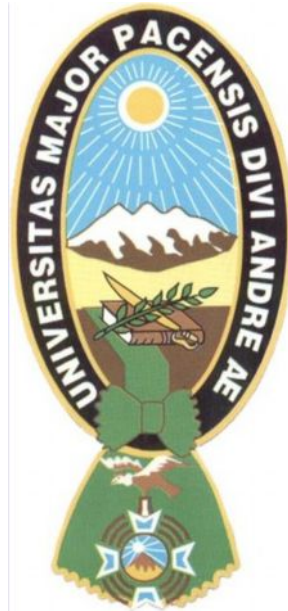


UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMATICA



“METODOLOGIA DE DESARROLLO WEB
ASEGURANDO ACCESIBILIDAD A PERSONAS CON
DISCAPACIDAD VISUAL”

PARA OPTAR AL TITULO DE LICENCIATURA EN INFORMATICA MENCION
INGENIERIA DE SISTEMAS INFORMATICOS

POSTULANTE: UNIV. MICHAEL PEREZ ACEBEY

TUTOR: LIC. FREDDY MIGUEL TOLEDO PAZ

REVISOR: M.SC. CARLOS MULLISACA CHOQUE

LA PAZ BOLIVIA

2014



**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA**



LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

DEDICATORIA

Esta tesis se la dedico a mis padres quienes me han apoyado para poder llegar a esta instancia de mis estudios, ya que siempre han estado presentes para apoyarme moral y psicológicamente.

AGRADECIMIENTOS

Agradezco a dios por haberme cuidado, por haber puesto en mi camino a todos aquellos que fueron parte de este trabajo.

A mi madre Ana María, a mi papi Pedro, a mi hermana Yesenia, a mí cuñado Jorge y a mi sobrino Leonardo por su apoyo y colaboración.

A los docentes de la carrera que fueron parte de mi formación en estos años de estudio, a mi grandes amigos de carrera Mayoli, Rene, Joso, Chicho, Chapaco, Cesar, Pablo, Milenka, y mis amigos de barrio Rubén, Iván, Fernando, Hernán. Quienes me brindaron todo el apoyo para poder seguir adelante y concluir este trabajo. A mi docente y tutor de taller de Licenciatura 2, Lic. Freddy Miguel Toledo Paz por su guía y amistad en el transcurso de la elaboración de esta Tesis. A mi docente Revisor M.Sc. Carlos Mullisaca, por su colaboración. A todos mis compañeros de la carrera de Informática y un agradecimiento especial a Humberto por sus consejos.

RESUMEN

Cuando los sitios web están diseñados correctamente pensando en la accesibilidad, todos los usuarios pueden acceder en condiciones de igualdad a los contenidos, si un sitio web tiene un código semánticamente correcto, se proporciona un texto equivalente y alternativo a las imágenes y a los enlaces le da un valor significativo, esto permite a los usuarios ciegos utilizar lectores de braille o programas de lectura de pantalla para acceder a los contenidos, pero si un sitio web está mal estructurado desordenado sin accesibilidad esto no permite el acceso a la información a personas con discapacidad visual .

Cuando una persona con discapacidad visual no puede acceder a cierto contenido de un sitio web esto es llamado barreras de accesibilidad. Por un lado, están aquellas relacionadas al estilo de la página (colores, tamaño, imágenes, audio y video). Impidiendo el acceso a la información y por otro la mala estructuración del sitio.

Este trabajo desarrolla una metodología que permite analizar todas las barreras que puede llegar a tener una página web dando una solución a cada una, permitiendo la accesibilidad a personas con discapacidad visual.

PALABRAS CLAVE: Accesibilidad, barreras, discapacidad visual, metodología.

ABSTRACT

When websites are properly designed with accessibility in mind, all users can access equal to the contents, if a website has a semantically correct code, an equivalent and alternative to images text is provided and links gives them significant value, this allows blind users to use braille readers or screen readers to access content, but if a website is poorly structured messy without accessibility does not allow access to information for people With visual disabilities When a person with visual disabilities can not access some content from a website that is called accessibility barriers. On one side are those related to the style of the page (color, size, images, audio and video). Preventing access to information. This paper develops a methodology to analyze the barriers that can have a web page giving a solution to each, allowing accessibility for people with visual disabilities.

KEYWORDS: Accessibility barriers, visual impairment, methodology.

INDICE

CAPÍTULO 1	1
1.1. INTRODUCCION	1
1.2. ANTECEDENTES	1
1.2.1. Publicaciones que existen	2
1.3. MARCO TEORICO	2
1.4. PLANTEAMIENTO DEL PROBLEMA	2
1.4.1. Problema general.....	2
1.4.2. Problemas específicos	4
1.5. HIPOTESIS.....	4
1.6. OBJETIVOS.....	4
1.6.1. Objetivo general.....	4
1.6.2. Objetivos específicos.....	4
1.7. JUSTIFICACIÓN	5
1.7.1. Justificación social	5
1.7.2. Justificación técnica	5
1.7.3. Justificación Económica	5
1.8. LIMITES Y ALCANCES	5
1.8.1. Limites.....	5
1.8.2. Alcances.....	6
1.9. APORTES	6
1.9.1. Mejorará el acceso de clientes potenciales.	6
1.9.2. Mejora del posicionamiento en buscadores.....	7
1.9.3. Responsabilidad social	7

1.10.	Metodología de investigación	7
1.11.	HERRAMIENTAS	7
	CAPITULO 2 MARCO TEORICO	9
2.1.	La entidad W3C-WAI y su rol	9
2.2.	Sobre la importancia de tener una Web accesible	10
2.3.	La Web no accesible	11
2.3.1.	Barreras.....	11
2.4.	Las pautas para lograr la accesibilidad Web	11
2.4.1.	UAAG: User Agent Accessibility Guidelines	12
2.4.2.	ATAG: Authoring Tools Accessibility Guidelines.....	13
2.4.3.	WCAG: Web Content Accessibility Guidelines.....	13
2.4.3.1.	WCAG 1.0	13
2.4.3.2.	WCAG 2.0	15
2.4.3.3.	Evaluación y verificación del cumplimiento de las pautas para contenido.....	18
2.5.	Las normativas de Accesibilidad Web.....	19
2.5.1.	Antecedentes.....	19
2.5.2.	Normativas particulares.....	20
2.6.	TIPOS DE ENFERMEDADES VISUALES	20
2.6.1.	Tipos de baja visión.....	20
2.6.1.1.	Degeneración macular	21
2.6.1.2.	Glaucoma.....	21
2.6.1.3.	Retinopatía diabética	22
2.6.2.	Daltonismo	23
2.6.2.1.	Deficiencia rojo verde	23

2.6.2.2. Protanopia y protanomalia (deficiencia de color rojo).....	23
2.6.2.3. Deuteranopia y deuteranomalia (deficiencia de color verde).....	24
2.6.2.4. Tritanopia (deficiencia de color azul)	25
2.6.2.5. Bastón monocromático o acromática (sin color)	26
2.7. INGENIERIA DE SOFTWARE	27
2.7.1. Modelo Lineal Secuencial , evolutivo o cascada.....	29
CAPÍTULO 3. MARCO APLICATIVO.....	32
3.1. Análisis para el desarrollo de web accesible.....	32
3.1.1. Selección de Objetivos	33
3.1.1.1. selección de usuarios.....	33
3.1.1.2. Expectativas de usuario	33
3.1.1.3. Expectativas de la organización.....	34
3.1.2. Reconocimiento del problema.	34
3.1.3. Evaluación y síntesis.	39
3.1.3.1. Páginas con tamaño absoluto.....	39
3.1.3.2. Presencia de elementos intermitentes y rotativos	40
3.1.3.3. Códigos de seguridad o captcha.....	40
3.1.3.4. Uso de tecnologías anexas al html	41
3.1.3.5. Falta De Soporte A La Navegación Por Teclado.....	41
3.1.3.6. Páginas Muy Extensas Sin Marcas o índice	42
3.1.3.7. Uso de tablas anidadas para contener la información.	42
3.1.3.8. Limitaciones en los tiempos de respuesta de las páginas.....	43
3.2. Diseño	44
3.2.1.1. Diseño de la estructura	49

3.2.1.2. Diseño de la página	51
3.2.2. SEPARACION Y ESTRUCTURACION DE CODIGO CSS	56
3.2.3. Barreras en el diseño accesible	60
3.2.3.1. Enlaces de hipertexto.....	66
3.2.3.2. Organización de las páginas: use encabezados, listas y estructura consistente. use css para la maquetación donde sea posible.	68
3.2.3.3. Posicionamiento absoluto	80
3.2.3.4. Marcos accesibles.....	82
3.2.3.5. Problemas de Cataratas protanopia, deuteranopia y tritanopia.....	84
3.3. código.....	85
3.3.1. Controles de formulario accesibles	85
3.3.1.1. Text.....	85
3.3.1.2. Textareas.....	86
3.3.1.3. Radio buttons	87
3.3.1.4. Select menus	87
3.3.2. CAPTCHA.....	88
3.3.3. Javascript	89
3.3.3.1. División De Código.....	89
3.3.3.2. Reglas Del Javascript No Obstructivo.....	89
3.3.3.3. Javascript solo es una mejora.....	90
3.3.3.4. Se comprueba la disponibilidad de un objeto.	92
3.3.3.5. Crea Un Javascript No Específico Para Un Navegador	93
3.3.3.6. No usar variables de otros scripts.	93
3.3.3.7. Efectos De Ratón De Forma Independiente	93

3.4.	pruebas	93
CAPITULO 4.....		97
4.1.	DESCRIPCION DE LA POBLACION	98
4.2.	PRUEBAS DE CHI CUADRADA	100
4.2.1.	Planteamiento De La Hipótesis.....	100
4.2.2.	Región Crítica	101
4.2.3.	Estadístico De Prueba.....	101
4.2.4.	Decisión.....	103
5 CONCLUSIONES Y RECOMENDACIONES		104
5.1.	CONCLUSIONES.....	104
5.2.	CONCLUSIÓN DE LA HIPÓTESIS.....	105
5.3.	RECOMENDACIONES	105
5.3.1.	UTILIZACION DE FLASH	105
5.3.2.	TEXTO JUSTIFICADO O ALINEADO A LA IZQUIERDA	106
5.3.3.	TAMAÑO DE TEXTO.....	106
BIBLIOGRAFIA		108
ANEXOS A: ARBOL DE PROBLEMAS.....		110
	Presencia de elementos intermitentes y rotativos	110
ANEXO B: ARBOL DE OBJETIVOS.....		111

INDICE DE FIGURAS

Figura 1 Percepción de persona con problema de degeneración macular	21
Figura 2 Percepción de una persona con problema de glaucoma.....	22
Figura 3 Percepción de una persona con problema de retinopatía diabética	22
Figura 4 Percepción de una persona con problema de protanopia	24
Figura 5 Percepción de una persona con problema deuterenaopia.....	25
Figura 6 Percepción de una persona con problema de Tritanopia.....	26
Figura 7 Percepción de una persona con problema de bastón monocromático.....	27
Figura 8 modelo evolutivo	29
Figura 9 Esquema propuesto	46
Figura 10 Diseño de acceso a nodos	50
Figura 11 Diseño de página web inaccesible	52
Figura 12 Diseño de página web accesible	52
Figura 13 Diseño navegacional	53
Figura 14 Estructura de una página web.....	54
Figura 15 Ejemplo de mal uso de imágenes.....	62
Figura 16 Prueba realizada por Fangs.....	63

Figura 17 Prueba con herramienta Fangs.....	65
Figura 18 Página web “el diario”.....	66
Figura 19 Prueba realiza por Fangs a la página el diario	67
Figura 20 Pagina web accesible.....	69
Figura 21 prueba realizada por Fangs.....	69
Figura 22 Estructura accesible.....	71
Figura 23 Ejemplo de uso de longdesc	73
Figura 24 Uso correcto de tablas	78
Figura 25 revisión con Juicy Studio Accessibility toolbar.....	79
Figura 26	81
Figura 27	81
Figura 28 Ejemplo de uso de marcos.....	84
Figura 29 percepción de enfermedades visuales	85
Figura 30 página de comprobación de accesibilidad W3C	94
Figura 31 Pagina de validación CSS.....	94
Figura 32 Pagina de comprobación de accesibilidad.....	95
Figura 33 página de comprobación de accesibilidad “TAW”	96

Figura 34 toma de pantalla de prototipo 97

INDICE DE TABLAS

Tabla 1 Ejemplo de mal uso 36

Tabla 2 Resultados de la muestra poblacional grupo 1 98

Tabla 3 Resultados de la muestra poblacional grupo 2 99

Tabla 4 resultados generados por SPSS 101

Tabla 5 resultados generados por SPSS 102

Tabla 6 Pruebas generadas por SPSS 102

INDICE DE ILUSTACIONES

Ilustración 1 Grafico circular de personas con manejo excelente de JAWS 99

Ilustración 2 Grafico circular de personas con manejo regular 100

Ilustración 3 Región de rechazo 103

CAPÍTULO 1

1.1. INTRODUCCION

La accesibilidad en la Web se refiere a la posibilidad de la misma en ser percibida, entendida, interactuada y navegada por personas con algún grado de discapacidad. Se incluyen problemas visuales, auditivos, físicos, cognitivos, neurológicos y del habla. Existen pautas para la creación de contenido Web accesible, así como también leyes en muchos países que promueven el contenido Web accesible. Sin embargo, en el momento de planificación web no se aplica ni una de estos aspectos y estos se cambian una vez terminado el proyecto.

Existen millones de personas con discapacidad que no pueden utilizar la Web. Actualmente, la mayoría de los sitios Web y los software Web presentan barreras de accesibilidad, lo que dificulta o imposibilita la utilización de la Web para muchas personas con discapacidad visual.

1.2. ANTECEDENTES

Muchos países adoptaron las pautas para contenido del consorcio W3C bajo un marco legal. Tomando parte de las pautas para contenido WCAG 1.0, se crearon leyes para garantizar la accesibilidad Web de los sitios en esa jurisdicción [Thatcher, 2006].

Los antecedentes de este movimiento sucedieron alrededor del año 1995 tanto para la Unión Europea como para los Estados Unidos. La primera legislación sobre accesibilidad general en los Estados Unidos fue “American and Disabilities Act”, conocida como ADA, y data del año 1992. Abarca al trabajo, la construcción, la educación, el transporte, las comunicaciones (en donde se incluiría la accesibilidad en la Web), la recreación, la salud y demás servicios. En ese momento el alcance de Internet era limitado y la Web era mucho menos gráfica, por lo tanto garantizar la accesibilidad no requería del soporte pautas ni lineamientos técnicos complejos. Para el año 1998, los Estados Unidos adoptan la “Section 508”, que es la primera normativa específica para accesibilidad Web de ese país [Paciello, 2002].

1.2.1. Publicaciones que existen

- Diseño de páginas web con características de accesibilidad para personas con discapacidad [Terán, 2004].

1.3. MARCO TEORICO

WCAG 2.0 se desarrolla a través del proceso de W3C en colaboración con personas y organizaciones de todo el mundo, con el objetivo de proporcionar un estándar común para la accesibilidad de contenidos web que cumpla con las necesidades de los individuos, organizaciones y gobiernos a nivel internacional. WCAG 2.0 se basa en WCAG 1.0 [WCAG1.0] y está diseñado para aplicarse en términos generales a las diferentes tecnologías web ahora y en el futuro, y para ser comprobable con una combinación de pruebas automatizadas y evaluación humana.

- **JavaScript** es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.
- **HTML5** (HyperText Markup Language, versión 5) es la quinta revisión importante del lenguaje básico de la World Wide Web, HTML. HTML5 especifica dos variantes de sintaxis para HTML: un clásico HTML (text/html), la variante conocida como HTML5 y una variante XHTML conocida como sintaxis XHTML5 que deberá ser servida como XML (XHTML) (application/xhtml+xml). Esta es la primera vez que HTML y XHTML se han desarrollado en paralelo.

1.4. PLANTEAMIENTO DEL PROBLEMA

1.4.1. Problema general

La Organización Mundial de la Salud recoge en sus informes un total de 600 millones de personas con discapacidad. El acceso de estas personas a la tecnología debe tenerse en cuenta en la construcción de una sociedad igualitaria [OMS].

La misma Organización Mundial de la Salud (OMS), a través de la Clasificación Internacional del Funcionamiento, de la Discapacidad y de la Salud (CIF) del año 2001, define la discapacidad como término genérico que incluye déficit, limitaciones en la actividad y restricciones en la participación. Indica los aspectos negativos de la interacción entre un individuo (con una condición de salud) y sus factores contextuales (factores ambientales y personales) [Egea, 2006].

Cuando los sitios web están diseñados pensando en la accesibilidad, todos los usuarios pueden acceder en condiciones de igualdad a los contenidos. Por ejemplo, cuando un sitio tiene un código XHTML semánticamente correcto, se proporciona un texto equivalente alternativo a las imágenes y a los enlaces se les da un nombre significativo, esto permite a los usuarios ciegos utilizar lectores de pantalla o líneas Braille para acceder a los contenidos. Cuando los vídeos disponen de subtítulos, los usuarios con dificultades auditivas podrán entenderlos plenamente. Si los contenidos están escritos en un lenguaje sencillo e ilustrados con diagramas y animaciones, los usuarios con dislexia o problemas de aprendizaje están en mejores condiciones de entenderlos.

Si el tamaño del texto es lo suficientemente grande, los usuarios con problemas visuales puedan leerlo sin dificultad. De igual modo, el tamaño de los botones o las áreas activas adecuado puede facilitar su uso a los usuarios que no pueden controlar el ratón con precisión. Si se evitan las acciones que dependan de un dispositivo concreto (pulsar una tecla, hacer clic con el ratón) el usuario podrá escoger el dispositivo que más le convenga.

Lo mencionado en los párrafos anteriores se puede resumir en Pautas de Accesibilidad; estas pautas explican cómo hacer accesibles los contenidos de la web a personas con discapacidad visual. Las pautas están pensadas para todos los diseñadores de contenidos de la web y para los diseñadores de herramientas de creación. El fin principal de estas pautas es promover la accesibilidad.

En las metodologías de diseño de páginas web estas no contemplan los aspectos para que una página web pueda ser percibida por una persona con discapacidad visual.

¿Será posible integrar las herramientas y métodos que existen de programación web basada en accesibilidad a una metodología de desarrollo para que una página web pueda ser percibida por personas con discapacidad visual?

1.4.2. Problemas específicos

Existen muchos tipos para personas con problemas visuales estos son los problemas más frecuentes al no tener una página web accesible:

- La incapacidad para ver los gráficos debido a deficiencias visuales.
- Las personas ciegas que utilizan programas para lectura no pueden acceder ya que la página no contiene código que facilita el acceso a la información y esto dificulta que el programa de asistencia puedan proporcionar toda la información.
- Dificultades para navegar por sitios que no están bien organizados mal estructurados incapaces de navegar.
- Incapacidad de distinción de colores en los contrastes de la página.

1.5. HIPOTESIS

El desarrollo de una metodología web facilita el acceso a contenido web para personas con discapacidad visual.

1.6. OBJETIVOS

1.6.1. Objetivo general

Desarrollar una metodología basada en el modelo evolutivo de ingeniería de software para el acceso a contenidos web de personas con discapacidad visual.

1.6.2. Objetivos específicos

- Integrar al análisis de requerimientos todas las posibles barreras de percepción de una persona con problemas visuales.

- Integrar en la fase de código elementos para que puedan facilitar la percepción por programas de asistencia a personas con discapacidad visual.
- Tomar en cuenta todos los posibles problemas o barreras que puedan limitar la percepción.
- Integrar todas las pruebas de accesibilidad dadas por la WAI [Web Accessibility Initiative].
- Diseñar un sitio web que sea accesible para las personas ciegas y con discapacidad visual bien estructurada y accesible.
- Acortar la brecha digital que pueda tener una persona con discapacidad visual.

1.7. JUSTIFICACIÓN

1.7.1. Justificación social

La metodología permitirá el acceso a tecnologías basadas en la web para personas con discapacidad visual, así mismo permitirá el acceso a toda la información para acortar la brecha digital.

1.7.2. Justificación técnica

Para el desarrollo de páginas web se utilizarán herramientas de software libre sin ningún costo.

1.7.3. Justificación Económica

Para el desarrollo de una página web accesible no tendrá ningún costo adicional al del proyecto ya que tan solo se tomara en cuenta ciertos aspectos que incluirá la metodología para ser perceptible.

1.8. LIMITES Y ALCANCES

1.8.1. Limites

La investigación procura desarrollar una metodología que dé accesibilidad a las personas con discapacidad visual integrando a la metodología evolutiva (Análisis, diseño, código, prueba) aspectos que no se toman en cuenta en todas las fases de desarrollo de un proyecto web.

La metodología se destina a personas con discapacidad visual que estén en una edad cognitiva comprendiendo lo que es un sitio web y ser capaz de navegarla.

Existen una gran variedad de enfermedades visuales, para lo que esta investigación no puede hacer un estudio exhaustivo y dar solución a cada una de ellas, para esto se ha tomado en cuenta las siguientes discapacidades visuales:

- Personas que tienen problemas de distinción de colores (daltonismo, protanopia, deuteranopia, tritanopia).
- Personas con problemas con baja visión (cataratas).
- personas con ceguera total o parcial.

1.8.2. Alcances

El trabajo está centrado en los aspectos que refieren a los estándares y a la evaluación de la accesibilidad en la Web. Queda fuera del alcance del trabajo todo lo relacionado a la usabilidad, a las tecnologías que hacen uso de la Web.

1.9. APORTES

1.9.1. Mejorará el acceso de clientes potenciales.

La accesibilidad además ayuda a la correcta visualización en todos los dispositivos, ya que no hay que olvidar el aumento del tráfico que se está produciendo desde smartphones y dispositivos móviles como IPads. Tanto la falta de acceso a toda la información, como los problemas de visualización puede provocar la pérdida de clientes y que estos busquen otras alternativas a los servicios.

1.9.2. Mejora del posicionamiento en buscadores

Los spiders de las principales herramientas de los buscadores han mejorado mucho tienen algunos problemas de acceso a la información similares a los de los usuarios con discapacidad. Un ejemplo de ello es el acceso a enlaces generados únicamente con javascript que no permiten el acceso a la información que enlazan. Asimismo, el uso de encabezamientos o metaetiquetas de forma correcta es un factor muy importante en el posicionamiento y la visibilidad de las páginas en buscadores.

1.9.3. Responsabilidad social

Aquellas empresas que se preocupan de la accesibilidad muestran una imagen de mayor responsabilidad social, lo que hace que se perciba su marca de forma más positiva.

1.10. METODOLOGÍA DE INVESTIGACIÓN

La metodología a utilizar será cuantitativa cualitativa descriptiva para especificar propiedades, características y rasgos importantes que puedan tener una persona con discapacidad visual y aplicarlas a la metodología para que esta pueda ser percibida.

1.11. HERRAMIENTAS

• Fangs para Mozilla Firefox

Fangs, una extensión para el navegador Mozilla Firefox que crea una representación textual de una página web similar a como lo haría un lector de pantalla. Con esta herramienta, podemos comprobar fácil y rápidamente como percibirá una página web una persona con discapacidad que utilice un lector de pantalla

Fangs dispone de tres pestañas:

- Screen reader output: muestra la representación lineal de la página, incluye "anuncios" (marcados en gris) que son generados por el lector de pantalla para explicar la estructura de la página web.

- Heading list: muestra una lista de los encabezados de la página, le permite al usuario tener una visión rápida de la estructura y contenido de la página web.
- Links list: muestra una lista de los enlaces de la página.
- Color Contrast Analyser para Páginas Web

Es sobre todo una herramienta para el control de las combinaciones de primer plano y color de fondo para determinar si ofrecen una buena visibilidad de color. También contiene la funcionalidad para crear simulaciones de ciertas condiciones visuales como la ceguera al color.

La relación de contraste ayuda a determinar si el contraste entre dos colores puede ser leído por personas con daltonismo u otros impedimentos visuales.



CAPITULO 2 MARCO TEORICO

La accesibilidad en la Web se refiere a la posibilidad capacidad de la misma en ser percibida, entendida, interactuada y navegada por personas con algún grado de discapacidad. Siguiendo pautas de diseño específicas, las páginas Web pueden ser accedidas por una variedad de usuarios a lo largo de diferentes escenarios. Se incluyen problemas visuales, auditivos, físicos, cognitivos, neurológicos y del habla [WAIintro & Thatcher et al., 2006].

Los usuarios con discapacidad visual operan las computadoras a través de dispositivos especiales, llamados tecnologías asistivas. En ambientes con software conocido, esos dispositivos tienen un funcionamiento esperado si la página web.

En un principio las páginas Web consistían en simples textos planos, y eso permitía que se adaptaran muy fácilmente a cualquier tecnología asistida. Con el tiempo, el HTML fue evolucionando hasta llegar a una Web más gráfica y construida en JavaScript. La complejidad agregada tanto a la estructura como a la presentación de los documentos dificultó el trabajo de los sistemas de asistencia. Todos los elementos en una Web que interfieren con la accesibilidad son denominados barreras de accesibilidad.

Más allá de los números, la importancia radica en conocer que un gran porcentaje de la audiencia de los sitios Web presenta dificultades en el acceso y eso se contrapone a los objetivos originales del fundador Tim Berners Lee [Paciello, 2002].

2.1. LA ENTIDAD W3C-WAI Y SU ROL

Con el fin de crear un estándar en las tecnologías para el desarrollo Web, la W3C tiene un organismo llamado WAI (Web Accessibility Initiative) [WAI].

La WAI, surgida en el año 1997, se dedica a desarrollar estrategias, pautas y recursos para hacer la Web accesible a las personas con discapacidad.

Una pauta (traducido de Guideline), a diferencia de una ley o normativa, es un conjunto de recomendaciones detalladas y organizadas con puntos de control. Cada pauta contiene sugerencias y ejemplos a seguir para lograr la conformidad con cada uno de sus

puntos de control. Existen tres componentes diferenciados sobre los cuales la WAI genera pautas de accesibilidad:

a) Authoring tools and User Agents: Authoring Tool se denomina al software usado para la creación de sitios Web, y un User Agent es un software cliente que se conecta a la red (comúnmente se llama así a los navegadores Web o browsers). Es responsabilidad de las empresas de desarrollo.

b) Tecnologías asistivas: Es la interfaz del lado del usuario (lectores de pantalla, dispositivos Braille, etc.). Es responsabilidad de las empresas fabricantes del hardware y software.

c) Contenido: Está conformado por el conjunto de páginas Web. Es responsabilidad de los desarrolladores y diseñadores.

La idea original de la WAI al realizar esta separación entre partes fue lograr un compromiso mutuo [WAI interdependences, 2005]. Si una parte tomaba la iniciativa y comenzaba a implementar las recomendaciones sobre accesibilidad, el resto haría lo mismo. Además, la responsabilidad de resolver la barrera se repartiría de forma balanceada [Chisholm & Henry, 2005].

Para el éxito de la metodología es contar con una fuerte adhesión de las pautas para contenido.

2.2. SOBRE LA IMPORTANCIA DE TENER UNA WEB ACCESIBLE

Existen diversos factores por los cuales resulta importante tener una Web accesible [WAI social factors & Thatcher et al., 2006]. El principal de ellos es el uso extensivo de la Web como medio de comunicación, Muchos métodos tradicionales están siendo reemplazados por páginas Web. Esta tendencia alcanza a la educación, el comercio, las comunicaciones, la participación civil, el cuidado de la salud, la recreación y las noticias.

Resulta valioso que la Web sea accesible para dar acceso igualitario a las personas con discapacidad para dar una participación más activa e informativa. Por otro lado, la Web ofrece una oportunidad de acceso a la información que no tiene precedentes.

2.3. LA WEB NO ACCESIBLE

2.3.1. Barreras

En general, una barrera de accesibilidad es un obstáculo que impide o dificulta la realización de una determinada tarea o actividad, afectando de esta manera a la plena integración social de esa persona.

Existen múltiples escenarios cotidianos en donde los usuarios pueden encontrarse con barreras. Cada tipo de enfermedad visual es susceptible a determinadas barreras, que generalmente están asociadas a la presentación y al estilo de la página.

Para comprender la complejidad del problema, será necesario estudiar la forma en la que cada grupo percibe la Web según el tipo de discapacidad que presenta [Brewer, 1994]. Se puede establecer una clasificación de las barreras. Por un lado, están aquellas relacionadas al estilo de la página (colores, tamaño, imágenes, audio y video), y por el otro están las barreras vinculadas a la estructura de la información. Este último grupo es el más difícil de detectar porque requiere evaluar la forma en que se percibe una página según su estructura de información [Sloan et al, 2006]. Puede ocurrir que la disposición del contenido no siga un criterio lógico, o que utilice un vocabulario confuso. Ese aspecto conforma lo que es la usabilidad de un sitio junto con su ontología.

Un punto importante es que la mayoría de las barreras aparecen cuando las páginas escapan del esquema habitual del documento de texto. Cada interacción del usuario con elementos multimedia les genera un problema a la accesibilidad.

2.4. LAS PAUTAS PARA LOGRAR LA ACCESIBILIDAD WEB

Como se expuso anteriormente, el consorcio W3 a través de su organismo para la accesibilidad (WAI) desarrolló un modelo de pautas con tres niveles.

En esta sección se enunciarán las pautas en su nivel más elemental. Cada ítem de las pautas posee documentos anexos que explican en detalle cómo alcanzarla y verificar su cumplimiento, disponibles en la página de la WAI.

2.4.1. UAAG: User Agent Accessibility Guidelines

User Agent es todo programa que recupera, reproduce y facilita la interacción entre el usuario y el contenido [UAAG 2.draft, 2009]. Las pautas de accesibilidad para User Agents “UAAG 1.0” fueron liberadas en el año 2002. Apuntan a lograr que los navegadores o cualquier software que interprete contenido Web no se comporte como barrera.

Algunos de los puntos de control son los siguientes:

1. Asegúrese de que la funcionalidad no basada en la Web sea accesible.
2. Asegúrese de que la funcionalidad basada en la Web sea accesible.
3. Dé soporte a las funcionalidades de accesibilidad de las tecnologías que interprete o muestre.
4. Muestre el contenido acorde con la especificación.
5. Facilite el acceso a la programación por parte de terceros.
6. Provea acceso al contenido alternativo.
7. Permita operar el software con un teclado.
8. Provea acceso a manejadores de eventos.
9. Permita la interacción independiente del tiempo.
10. Permita a los usuarios evitar los parpadeos que se presenten.
11. Guarde las preferencias del usuario.
12. Provea búsqueda en el texto.

13. Provea navegación estructurada.

14. Provea barras de herramientas personalizadas.

15. Ayude a los usuarios a eludir mensajes de alerta innecesarios.

Muchos de los puntos están vinculados con la usabilidad pero este trabajo está relacionado con la accesibilidad.

2.4.2. ATAG: Authoring Tools Accessibility Guidelines

De la segunda parte de ATAG 1.0, los puntos centrales son los siguientes:

1. De soporte a tecnologías de contenido Web que permitan la creación de contenido accesible.
2. Asista a los usuarios cuando realicen un chequeo de los problemas de accesibilidad.
3. Asista a los usuarios al reparar los problemas de accesibilidad.
4. Asista a los autores a administrar, editar y reusar descriptores de texto para elementos no textuales.
5. Asista a los autores con plantillas accesibles.
6. Asegúrese de que las acciones para lograr la accesibilidad sean integradas y promovidas.

2.4.3. WCAG: Web Content Accessibility Guidelines

Las pautas WCAG están dirigidas a quienes generan en contenido para la Web. Consisten en recomendaciones puntuales, redactadas de una forma menos genérica que el resto de las pautas anteriores. El foco está puesto en lograr que el contenido sea presentado de forma accesible. La WAI elaboró guías anexas que explotan cada punto y detallan los pasos a seguir para implementarlas y cumplirlas.

2.4.3.1. WCAG 1.0

Es la primera versión de las pautas, y fue liberada en el año 1999. WCAG 1.0 Consiste en catorce principios generales de diseño accesible. Cada uno de ellos define un punto de control sobre el cual el cumplimiento es verificado. Los puntos están priorizados del 1 al 3: [WCAG 1.0].

Prioridad 1: Debe ser cumplido. Es un requerimiento básico.

Prioridad 2: Debería ser cumplido. Remueve ciertas barreras.

Prioridad 3: Podría ser cumplido. Mejoraría la accesibilidad para ciertos grupos.

Del conjunto de prioridades presentes en el sitio, las pautas definen tres niveles de conformidad:

Nivel A: Todos los puntos de control de prioridad 1 son cumplidos.

Nivel AA: Todos los puntos de control de prioridades 1 y 2 son cumplidos.

Nivel AAA: Todos los puntos de control de prioridades 1,2 y 3 son cumplidos.

Las pautas genéricas para la accesibilidad definidas son las siguientes:

1. Provea alternativas equivalentes para contenido visual y auditivo.
2. Asegúrese de que tanto el texto como los gráficos se puedan ver sin color (modo monocromático).
3. Utilice maquetado y hojas de estilo para la estructura del documento.
4. Emplee un lenguaje claro en el contenido.
5. Cree tablas que puedan ser interpretadas por los lectores de forma correcta.
6. Asegúrese de que las páginas que contienen elementos de tecnologías nuevas se puedan ver de forma correcta aun cuando éstas no se encuentren disponibles.
7. Asegúrese de que los objetos móviles o destellantes puedan ser controlados por el usuario.

8. Asegure la accesibilidad de los objetos embebidos en su página.
9. Utilice un diseño independiente del dispositivo de entrada.
10. Utilice soluciones intermedias hasta que las tecnologías asistivas den soporte al diseño original.
11. Siga las pautas, y en caso de resultar imposible, provea una solución alternativa paralela.
12. Provea información de contexto que oriente al usuario.
13. Provea mecanismos claros de navegación.
14. Asegúrese de que los documentos son claros y simples.

Existen ejemplos y casos puntuales en cada uno de los puntos, que de cumplirlos, darán un grado de conformidad global.

La mayoría de las leyes sobre accesibilidad están basadas en el cumplimiento de estas pautas, pese a la antigüedad que tienen. Existen además herramientas para validar y asistir al desarrollo de contenidos, también basadas en 1.0. Sin embargo, la evolución de la Web hizo necesaria una reforma. Esta versión está basada en el HTML y muchas de las recomendaciones están concebidas según las formas de trabajo del HTML y de las hojas de estilo. Las nuevas tecnologías para creación de páginas Web, no basadas en HTML, no pasarían la conformidad más básica de las WCAG 1.0, aun siendo accesibles.

Sobre WCAG 1.0 fueron creadas normativas regionales, y una gran variedad de software para validar la accesibilidad de sitios Web.

2.4.3.2. WCAG 2.0

La versión más reciente al momento de las pautas WCAG fue liberada en diciembre del 2008 luego de dos años de revisiones y mejoras. El modelo 1.0 tenía el inconveniente de la dependencia tecnológica.

Si bien el autor puede seguir WCAG y lograr la conformidad máxima, no hay un control acerca de cómo el usuario está accediendo al contenido. [Kelly et al, 2007]. En diez años la variedad de plataformas y tecnologías asistivas para acceder a la Web creció, y es necesario replantear el significado de la conformidad. Los cambios más importantes que trajo la versión 2.0 son la separación tecnológica y la reorganización de las pautas.

Así como WCAG 1.0 posee pautas con puntos de control de prioridades 1, 2 y 3, WCAG 2.0 está organizada en cuatro principios de diseño. Cada uno de ellos contiene pautas, de las cuales existen tres diferentes criterios de éxito. [WCAG 2.0, 2009].

Los cuatro principios de diseño que siguen las pautas son los siguientes:

1. Perceptibilidad: El contenido debe ser mostrado de modo tal que los usuarios puedan percibirlo.
2. Operabilidad: Los componentes de la interfaz de usuario y la navegación deben ser operables.
3. Comprensibilidad: La información y el manejo de la interfaz de usuario debe ser comprensible.
4. Robustez: El contenido debe ser lo suficientemente robusto como para ser interpretado en múltiples agentes de usuario y tecnologías asistivas.

El nuevo documento de pautas es más extenso y detallado que su antecesor. Ante el aumento de la abstracción, fue necesario ampliar cada pauta con un documento anexo que explica posibles implementaciones. A continuación se mostrará el primer nivel de las pautas:

a) Perceptibilidad

- Proporcione alternativas textuales para el contenido no textual.
- Contenido multimedia dependiente del tiempo. Proporcione alternativas sincronizadas.

- Adaptabilidad: Cree contenidos que puedan presentarse de diferentes maneras (como una composición más simple) sin perder la información ni su estructura.
- Haga más fácil para los usuarios ver y distinguir el contenido, incluyendo la separación entre primer plano y fondo.

b) Operabilidad

- Accesible a través del teclado. Haga que toda la funcionalidad esté accesible por teclado.
- Tiempo suficiente. Proporcione al usuario el tiempo suficiente para leer.
- Ataques. No genere contenido destellante.
- Navegable. Proporcione al usuario ayuda a la hora de navegar y localizar el contenido.

c) Comprensibilidad

- Legibilidad: Haga el contenido textual legible y comprensible.
- Predecible Cree páginas Web cuya apariencia y operabilidad sean predecibles.
- Ayuda a la entrada de datos. Ayude a los usuarios a evitar y corregir errores.

d) Robustez

- Compatible. Maximice la compatibilidad con los navegadores y las tecnologías asistidas.

Criterios de éxito o Requisitos de conformidad para toda la página:

Dentro de cada pauta hay puntos a cumplir con criterios de éxito establecidos, y cada uno una prioridad que va desde A (básico) hasta AAA.

Nivel A: La página satisface con todas las pautas de nivel A, existe una página alternativa que las satisfice.

Nivel AA: La página satisface todas las pautas de nivel A y AA, o bien satisface A y existe una página alternativa que cumple con AA.

Nivel AAA: La página satisface A, AA y AAA, o bien satisface A, AA y existe una página alternativa que lo hace con AAA.

2.4.3.3. Evaluación y verificación del cumplimiento de las pautas para contenido

La WAI desarrolló un conjunto de procedimientos para evaluar el cumplimiento de las pautas para accesibilidad Web. Además de que cada punto de control. Posee un criterio de éxito, se toma al conjunto de páginas para realizar la revisión y calificar con A, AA ó AAA [Evaluating Accessibility].

WCAG 1.0 es más simple de evaluar por estar basada en el HTML y tener criterios de éxito puntuales. Existen validadores automáticos que realizan el trabajo de verificar cada punto de las pautas. Dentro de los más conocidos: Bobby, TAW, Hera. Una de las principales críticas al modelo WCAG 2 es la mayor dificultad para evaluar las pautas que no dependen de una tecnología específica. Fue una debilidad muy criticada cuando circularon los primeros borradores. En consecuencia, la WAI puso el esfuerzo en generar documentos detallando y explicando cada pauta y sus criterios.

El proceso de evaluación de las pautas propuesto por WCAG 2.0 requiere de la intervención de un grupo de personas. Las herramientas automáticas fallan a la hora de detectar barreras que provienen de la percepción.

Por otra parte, el cumplimiento de las pautas y el aseguramiento de la accesibilidad no están directamente asociadas. El paso inductivo hacia la accesibilidad es cuestionable el cumplimiento de las pautas no garantiza en la totalidad de los casos que la página sea accesible [Sloan et al, 2006]; como se mostró, una página con tecnologías no contempladas por las pautas WCAG 1.0 también puede ser accesible pese a no cumplir los criterios de conformidad.

- **Cumplimiento de Pautas No implica Accesibilidad**

Una excepción frecuente es la legibilidad del contenido. Las pautas expresan que el contenido debe ser legible y comprensible, y proporcionan técnicas para conseguirlo. No obstante, el criterio de éxito de este punto consiste en un desafío respecto de su evaluación. Se puede generar contenido comprensible para un cierto grupo de usuarios, pero es incorrecto generalizar hacia todos los demás grupos sin haberlo probado.

2.5. LAS NORMATIVAS DE ACCESIBILIDAD WEB

2.5.1. Antecedentes

Muchos países adoptaron las pautas para contenido del consorcio W3 bajo un marco legal. Tomando parte de las pautas para contenido WCAG 1.0, se crearon leyes para garantizar la accesibilidad Web de los sitios en esa jurisdicción [Thatcher et al, 2006].

Los antecedentes de este movimiento sucedieron alrededor del año 1995 tanto para la Unión Europea como para los Estados Unidos.

La primera legislación sobre accesibilidad general en los Estados Unidos fue “American and Disabilities Act”, conocida como ADA, y data del año 1992. Abarca al trabajo, la construcción, la educación, el transporte, las comunicaciones (en donde se incluiría la accesibilidad en la Web), la recreación, la salud y demás servicios. En ese momento el alcance de Internet era limitado y la Web era mucho menos gráfica, por lo tanto garantizar la accesibilidad no requería del soporte pautas ni lineamientos técnicos complejos. Para el año 1998, los Estados Unidos adoptan la “Section 508”, que es la primera normativa específica para accesibilidad Web de ese país [Paciello, 2002].

La Unión Europea inició diversos proyectos en el año 1994. A través de la entidad “Information Society for All”, lanzada por la Comisión Europea en el año 1999, se iniciaron acciones para liberar normativas que regulen la accesibilidad Web. Los planes de acción “Europe 2000”, basados en WCAG, alcanzan a los sitios del sector público. Por otro lado, existe el “WAB Cluster”, entidad que se encarga de la evaluación y el

monitoreo de los sitios Web de todos los países, generando informes y métricas actualizadas [Thatcher et. al, 2006].

2.5.2. Normativas particulares

- **Section 508**

En el año 1998 el congreso de los Estados Unidos promulgó el “Rehabilitation Act” para exigir que los organismos oficiales hagan su tecnología de la información accesible para las personas con discapacidad. La ley aplica para todos los organismos oficiales cuando desarrollan, mantienen o utilizan de la electrónica y la tecnología de la información. Bajo esta ley, se debe otorgar a empleados con discapacidades y demás miembros acceso público a la información [Wikipedia].

- **Unión Europea eEurope**

La iniciativa eEurope Una sociedad de la información para todos fue lanzada en el año 1999. En el plan de acción del año 2002 fueron incluidos aspectos de la accesibilidad Web. A diferencia de lo que sucede con Section 508, la enmienda de la ley de los Estados Unidos, eEurope emplea abiertamente las pautas de la WAI [eEuropeSidareEurope]. Para el año 2001, se exigía a las páginas Web de la administración pública cumplir con conformidad A de WCAG 1.0.

2.6. TIPOS DE ENFERMEDADES VISUALES

2.6.1. Tipos de baja visión

La agudeza visual de personas con baja visión es muy variable, pero, en general, de baja visión se define como una condición en la cual la visión de una persona no se puede corregir con gafas, lo que interfiere con las actividades diarias como leer y conducir. La baja visión es más común entre los ancianos, pero puede ocurrir en personas de cualquier edad como consecuencia de enfermedades como la degeneración macular, glaucoma, retinopatía diabética o las cataratas. Cada una de estas afecciones hace que los diferentes tipos de efectos en la visión de una persona, sin embargo, aquí están algunos ejemplos:

2.6.1.1. Degeneración macular

La mácula se encuentra cerca del centro de la retina, que es el área en la parte posterior del ojo. El proceso de envejecimiento y el adelgazamiento de los tejidos de la mácula causa la forma más común de degeneración macular, la degeneración macular seca. El resultado es una pérdida gradual de la visión. La degeneración macular seca ocurre cuando los vasos sanguíneos anormales en la parte posterior del ojo empiezan a filtrar líquido o sangre, haciendo borrosa la visión central, siendo a menudo el resultado la pérdida rápida de la visión. En cualquier caso, la zona central de la persona de la vista es la más afectada, por lo que es difícil ver los objetos que la persona está mirando directamente. Las imágenes a continuación son una simulación del efecto de la degeneración macular. El texto puede aparecer roto y poco claro.

Figura 1 Percepción de persona con problema de degeneración macular



Fuente: propia

2.6.1.2. Glaucoma

El glaucoma es causado por un aumento de presión dentro del ojo, lo que provoca daños en el nervio óptico. El resultado final es a menudo el efecto opuesto al de la degeneración macular: la pérdida de la visión periférica y una zona central de la visión borrosa. Puede ser particularmente difícil de leer el texto porque este parece haberse

difuminado, así como parecer borroso. Algunas personas han comparado los efectos del glaucoma a mirar todo a través de una pajita.

Figura 2 Percepción de una persona con problema de glaucoma

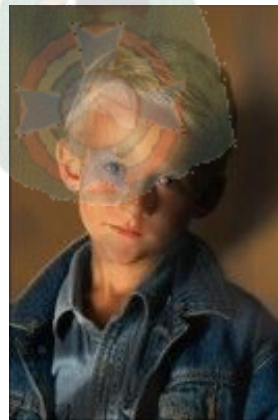


Fuente: propia

2.6.1.3. Retinopatía diabética

Uno de los efectos a largo plazo de la diabetes puede ser la filtración de los vasos sanguíneos de la retina, causando manchas oscuras en el campo de visión en la que las fugas se producen. El texto puede aparecer borroso o distorsionado en estas regiones.

Figura 3 Percepción de una persona con problema de retinopatía diabética



Fuente: propia

2.6.2. Daltonismo

2.6.2.1. Deficiencia rojo verde

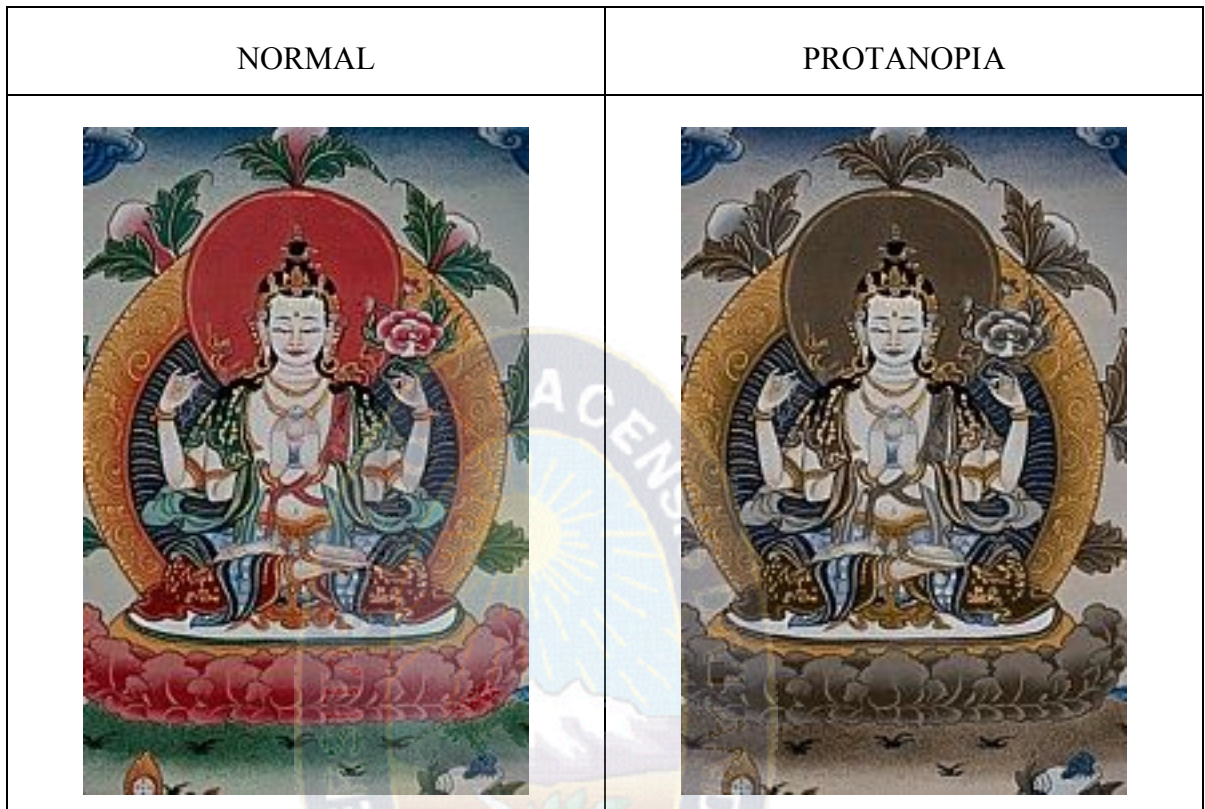
La categoría más amplia y común de daltonismo es la llamada ceguera rojo verde, pero esto no significa que estas personas no pueden ver los rojos o verdes. Ellos simplemente tienen más dificultades para diferenciar entre ellos. No todos los rojos y los verdes son indistinguibles. Sería muy fácil para alguien con una deficiencia rojo verde distinguir entre un color verde claro y rojo oscuro, por ejemplo. Depende mucho, al menos en parte, la oscuridad de los colores. Si el rojo es aproximadamente tan oscuro como el verde, hay una mayor probabilidad de que los colores se confundan.

Además, existe alguna evidencia de que las personas con daltonismo rojo verde vean rojos y verdes como amarillos, naranjas y beises. Esto significa que los amarillos, naranjas y beises se puedan confundir con verdes y rojos. Los colores menos afectados son los azules.

2.6.2.2. Protanopia y protanomalia (deficiencia de color rojo)

Los receptores de color (conos) en los ojos de las personas con protanopia no son sensibles a longitudes de onda largas (las del color rojo). Los rojos parecen más beises y parecen ser un poco más oscuros de lo que realmente son. Los verdes tienden a parecerse a los rojos. Protanomalia es más suave que protanopia, pero el resultado final es similar. Aunque muchas personas con protanomalia pueden distinguir algunos rojos y verdes, aunque no lo puede hacer tan fácil como una persona con visión normal, y, como con protanopia, los rojos tienden también a más oscuros.

Figura 4 Percepción de una persona con problema de protanopia

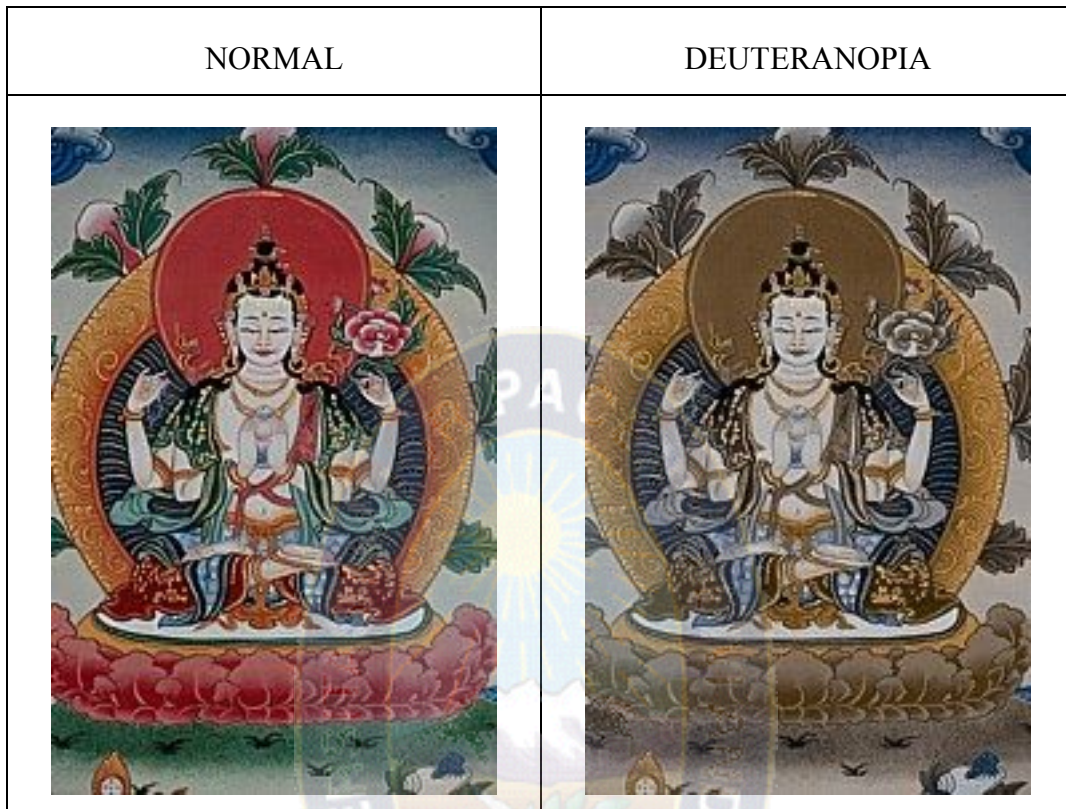


Fuente: propia

2.6.2.3. Deuteranopia y deuteranomalia (deficiencia de color verde)

Deuteranopia y deuteranomalia son las formas más comunes de daltonismo. Las personas con estas condiciones tienen conos que son sensibles a longitudes de onda media (verde), pero el resultado final es similar a la protanopia, con la excepción de que los rojos no se ven tan oscuros. Deuteranomalia es menos grave de las dos condiciones. Aunque los individuos con deuteranomalia probablemente no pueden percibir los rojos y verdes de la misma manera que lo puede ver la gente sin estos problemas, a menudo pueden distinguir entre las tonalidades de rojos y verdes con relativa precisión.

Figura 5 Percepción de una persona con problema deuteranoopia



Fuente: propia

2.6.2.4. TRITANOPIA (DEFICIENCIA DE COLOR AZUL)

Tritanopia es mucho menos común que las otras categorías mencionadas anteriormente. Tritanopia es la insensibilidad a las longitudes de onda cortas (los azules). En general, los azules y verdes se pueden confundir, pero los amarillos también se ven afectados en cuanto a que puede parecer que desaparecen o aparecen como ligeros tonos de rojo.

Figura 6 Percepción de una persona con problema de Tritanopia

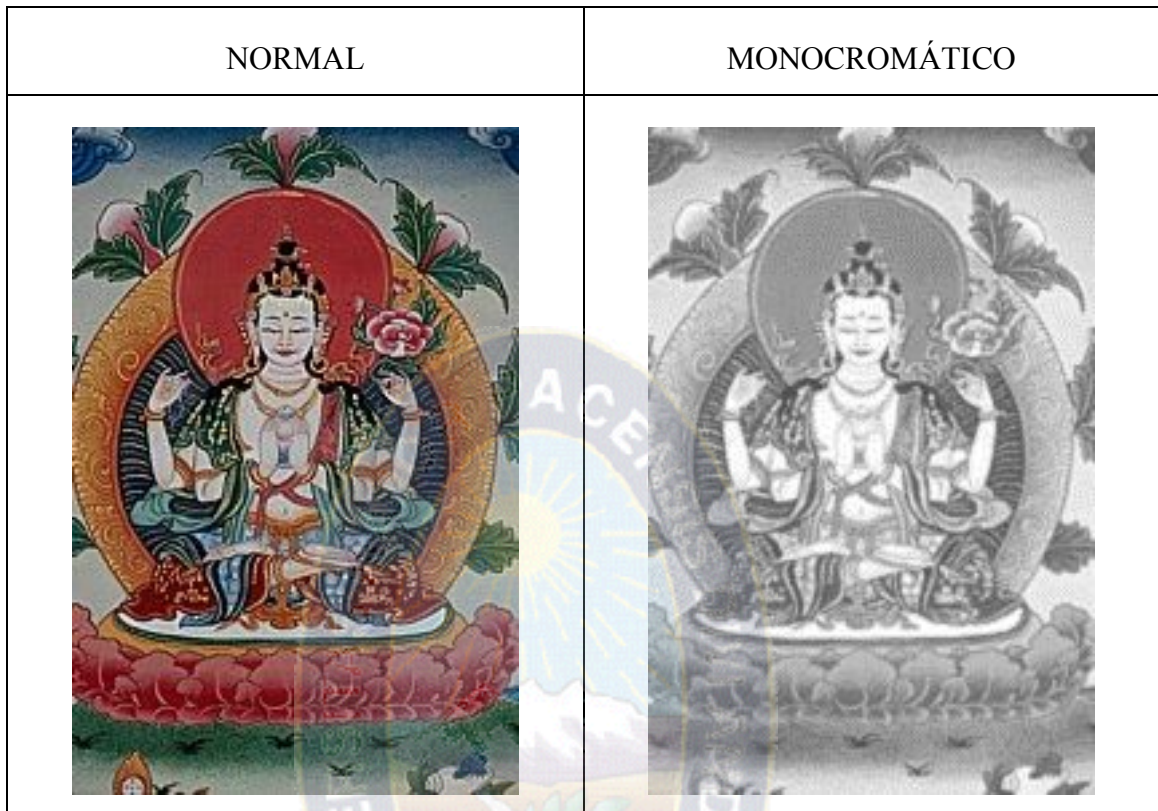


Fuente: [Propia]

2.6.2.5. BASTÓN MONOCROMÁTICO O ACROMÁTICA (SIN COLOR)

Este grupo constituye una minoría muy pequeña pero extrema del daltonismo. Los conos de los ojos no son funcionales, por lo que los bastones (receptores que sólo pueden diferenciar entre claro y oscuro) son la única fuente de información visual. Las personas con acromatía no pueden ver ningún color. El suyo es un mundo de color negro, blanco y tonos de gris. A menudo tienen mala agudeza visual y tiene una aversión a la luz brillante. Este es el único grupo para el que el daltonismo es una etiqueta que se ajusta, ya que todos los demás grupos tienen la capacidad de ver un poco de color.

Figura 7 Percepción de una persona con problema de bastón monocromático



Fuente: [Propia]

2.7. INGENIERIA DE SOFTWARE

Ingeniería de Software es una disciplina o área de las ciencias de la computación que ofrece métodos y técnicas para desarrollar y mantener software de calidad que resuelve problemas de todo tipo [Pressman, 98].

Ingeniería de Software no es una disciplina que solo debe seguirse para proyectos de software o desarrollo de páginas web que se encuentren pensados dentro de ciertas áreas, por el contrario, trata con Áreas muy diversas de las ciencias de la computación, tales como construcción de compiladores, sistemas operativos, o desarrollos en. La Ingeniería de Software abarca todas las fases del ciclo de vida del desarrollo de cualquier tipo de sistemas de información aplicables a áreas tales como los negocios, investigación

científica, medicina, producción, logística, banca, y para el caso particular de este estudio desarrollo de páginas web [Pressman, 98].

Un aspecto muy importante de Ingeniería de Software es que proporciona parámetros formales para lo que se conoce como Gestión de Proyectos de Software y desarrollo web. Esto se refiere a que Ingeniería de Software proporciona diversas métricas y metodologías que pueden usarse como especificaciones para todo lo referente a la administración del personal involucrado en proyectos web, ciclos de vida de un proyecto de software, costos de un proyecto, y en si todo el aspecto administrativo que implica el desarrollar páginas web. Por supuesto que estos aspectos son relevantes para los fines de la creación de esta metodología.

De acuerdo con Pressman [Pressman, 98], Ingeniería en general es el análisis, diseño, construcción, verificación y gestión de entidades técnicas. En general, todo proceso de ingeniería debe comenzar por contestar las siguientes preguntas en este caso para una página web:

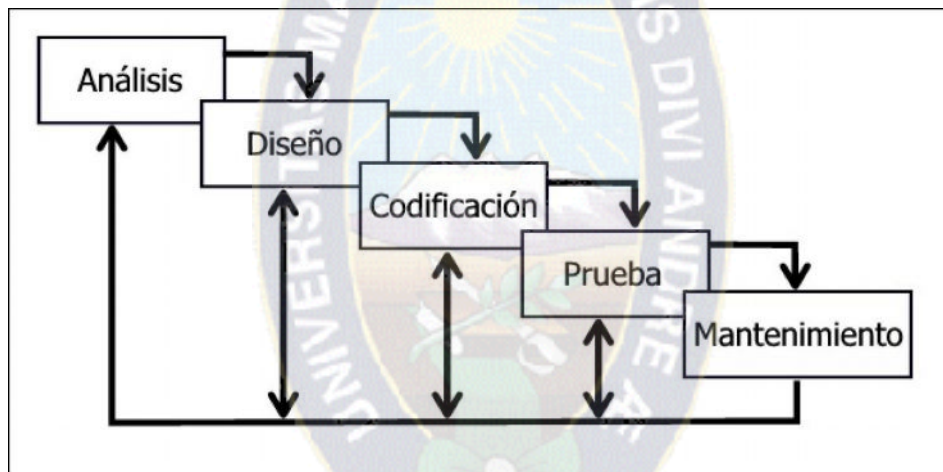
1. ¿Se necesita realmente el sitio Web?
2. ¿Para qué se necesita el sitio Web?
3. ¿Qué es lo que buscarán los usuarios en el sitio Web?
4. ¿Qué imagen se quiere proyectar con el sitio Web a los usuarios?
5. ¿Qué servicios interactivos se integrarán a través del sitio Web?
6. ¿Con qué recursos se cuenta para el desarrollo del sitio Web?
7. ¿Cuánto tiempo debe ser invertido en el desarrollo del sitio Web?
8. ¿Qué impacto causará el sitio Web en los usuarios?

Existen muchas metodologías para el desarrollo de software o páginas web pero estas contemplan a la accesibilidad como un punto no muy primordial solo lo mencionan y no se hace énfasis de hacer que la pagina sea accesible, cada una de estas metodologías

pretende de una manera u otra proporcionar en lo posible un orden para el desarrollo de software y páginas web. Para el caso de esta tesis es necesario apegarse lo más posible a uno de estas metodologías con el fin de tener una organización de actividades que se planean en base a una serie de etapas lógicas e interconectadas entre sí. El modelo de ingeniería de software que esta tesis sigue es el Modelo Lineal Secuencial o evolutivo, que es la más básica de todas las metodologías de desarrollo de software pero a la vez es de la que partieron las demás metodologías.

2.7.1. Modelo Lineal Secuencial , evolutivo o cascada

Figura 8 modelo evolutivo



Fuente: Pressman

De acuerdo con Pressman, el modelo lineal secuencial contempla actividades que deben llevarse a cabo. A continuación se describen estas actividades, y se aterriza cada una de ellas a los fines de este proyecto:

- **Ingeniería y modelado de Sistemas.**

Uno de los modelos estructurados más básicos, y que sirve como bloque de construcción para los demás modelos de ciclo de vida, fue el modelo en lineal secuencial, Al igual que el proceso de Diseño Inclusivo debe estar conducido por las necesidades y objetivos del usuario, en este caso requerimientos de usuarios con discapacidad visual.

- **Análisis de los requisitos del software.**

El proceso de reunión de requisitos se intensifica y se centra especialmente en el software. Dentro del proceso de análisis es fundamental que a través de una colección de requerimientos funcionales y no funcionales, el desarrollador o desarrolladores web completamente la naturaleza de los programas que deben construirse para desarrollar la aplicación, la función requerida, comportamiento, rendimiento e interconexión [Pressman, 98].

- **Diseño.**

La mejor forma de asegurar la accesibilidad de un sitio web es mediante la separación de contenido, presentación, estructura lógica e interacción. La separación entre contenido, presentación y estructura lógica, así como los beneficios derivados de este modo de diseñar, son ampliamente conocidos entre desarrolladores web. No tanto, sin embargo, la abstracción de elementos de interacción [Stephanidis, 1998]. Diferentes usuarios necesitarán diferentes formas de presentación de información, pero también elementos de interacción adaptados. En la Web, el elemento básico de interacción es el enlace, y este, debido a su simplicidad de comportamiento no necesitará de esta abstracción, sin embargo otros elementos más complejos (como listas de selección) deberán presentarse y responder ante la interacción de diferente forma según las características, habilidades y limitaciones del usuario.

- **Generación de Código.**

Esta actividad consiste en traducir el diseño en una forma legible por la máquina en este caso ver que código no se incluye para dar accesibilidad.

- **Pruebas.**

Una vez que se ha generado código, comienzan las pruebas del software o sistema que se ha desarrollado. De acuerdo con Pressman, el proceso de pruebas se centra en los procesos lógicos internos del software, asegurando que todas las sentencias se han comprobado, y en los procesos externos funcionales, es decir, la realización de las

prueba para la detección de errores [Pressman, 98]]. En el caso de una herramienta de software para tratar el trastorno de lateralidad y ubicación espacial, es necesario tener etapas de pruebas tanto para la parte funcional del software, como para la parte aplicativa del mismo. Se requiere poder probar el software con sujetos reales que puedan evaluar el comportamiento del software con el fin de proporcionar retroalimentación a los desarrolladores. Es sumamente importante que durante el proceso de desarrollo no se pierda el contacto con los interesados o solicitantes del desarrollo de software, de esta manera los objetivos de proyecto se mantendrán vigentes y se tendrá una idea clara de los aspectos que tienen que probarse durante el periodo de pruebas.



CAPÍTULO 3. MARCO APLICATIVO

En todo desarrollo de páginas web es de suma importancia el seguir alguna especificación que permita a los desarrolladores el tener una disciplina que haga que todas las etapas del desarrollo de una página web, desde la el análisis de requerimientos hasta las pruebas finales de la página, sean no solo más coherentes sino también más formales para un correcto ordenamiento de las etapas de desarrollo siguiendo unos pasos.

El desarrollo de una página web, al ser una herramienta que pretende tener aplicación dentro del contexto de un problema real destinada a informar a las personas con discapacidad, tiene que seguir un proceso de conjunto de pasos que proporcione los cimientos bajo los cuales se va a desarrollar una web conjuntamente. Es por esto que en este capítulo se detallan los procesos de ingeniería de software, análisis, y diseño que se involucran para el desarrollo de una web asegurando accesibilidad a personas con discapacidad visual.

Uno de los modelos estructurados más básicos, y que sirve como bloque de construcción para los demás modelos de ciclo de vida, fue el modelo en lineal secuencial. En la actualidad, el modelo en cascada es sustituido por los modelos iterativos e incrementales asociados con la tecnología más reciente orientada a objetos, aunque en el fondo el modelo en cascada sigue siendo la base de todos los procesos.

3.1. ANÁLISIS PARA EL DESARROLLO DE WEB ACCESIBLE

Tanto el desarrollador como el cliente tienen un papel activo en la ingeniería de requisitos un conjunto de actividades que son denominadas análisis, El cliente intenta replantear un sistema confuso, a nivel de descripción de datos, funciones y comportamiento, en detalles concretos. El desarrollador actúa como interrogador, como consultor, como persona que resuelve problemas y como negociador.

Pressman establece que la tarea del análisis de requisitos es un proceso de descubrimiento, refinamiento, modelado y especificación. Se refina en detalle el ámbito del software, y se crean modelos de los requisitos de datos, flujo de información y

control, y del comportamiento operativo. Se analizan soluciones alternativas y se asignan a diferentes elementos del software. El análisis de requisitos permite al desarrollador o desarrolladores especificar la función y el rendimiento del software, indica la interfaz del software con otros elementos del sistema y establece las Restricciones que debe cumplir el software.

3.1.1. Selección de Objetivos

Todo proyecto debe contar con objetivos definidos, contra los cuales puede medirse la efectividad de las acciones que se desarrollen.

Denominada esta etapa, resumen de objetivos, donde establecen tres puntos primordiales para esta selección, los cuales son definir los usuarios, lo que los usuarios esperan de la visita, y qué se pretende comunicar a través del sitio Web [Green, 2003].

También se puede denominar selección de usuarios, expectativas de usuario y expectativas de la organización; se describe a continuación:

3.1.1.1. selección de usuarios

Los usuarios o la audiencia se refieren a aquella parte de la población a la que se pretende alcanzar o influir con el sitio web, esto está estrechamente vinculado con la naturaleza del sitio que se propone. Toda la información que se obtenga servirá para establecer parte de los criterios de diseño de contenido, lenguajes, estética, entre otros.

El modelado del usuario se basa en definir clases o perfiles de los usuarios y así poder diseñar con el objetivo de satisfacer las necesidades propias de cada grupo de usuarios [Montero, 2003].

3.1.1.2. Expectativas de usuario

La organización puede o no, ser conocida por los usuarios en potencia del sitio. Al plantearse la puesta en marcha de un sitio web es necesario conocer hacia quién va dirigido éste y qué podrían esperar dichos usuarios al visitarlo.

3.1.1.3. Expectativas de la organización

Es vital establecer o fijar qué resultados pretenden conseguir la organización con la creación del sitio web. Con ello se busca definir el propósito del sitio web, y se deben incluir aspectos que van desde los más generales, relacionados con el público de destino, hasta los más específicos relacionados con la organización.

[Morris, 2001] indican que se debe decidir si lo que se busca con el sitio web es: aumentar ventas, ofrecer servicios especializados y conocimientos, publicidad y patrocinio, incentivos o para publicitar.

3.1.2. Reconocimiento del problema.

Reconocer los elementos básicos del problema tal y como los perciben los usuarios finales:

La existencia de barreras en el acceso a la comunicación imposibilita que las personas con necesidades especiales puedan acceder a la información suministrada por las páginas web, al no cumplir una serie de requisitos imprescindibles y, por tanto, limita el acceso a este colectivo.

Por eso entendemos de que la accesibilidad web es un recurso muy importante para diferentes aspectos de la vida: educación, empleo, gobierno, comercio, sanidad, entretenimiento y muchos otros. Es muy importante que las páginas web sean accesibles para así proporcionar un acceso equitativo en igualdad de oportunidades a las personas con necesidades especiales.

1. Brindar textos Alternativos

En muchas páginas web el contenido de las páginas se ofrece de una sola forma, sin alternativas equivalentes. Así, por ejemplo:

- Las imágenes (y todos los elementos no textuales en general), no ofrecen un texto alternativo que pueda ser leído por lectores de pantalla de personas con ceguera o

por personas que deben deshabilitar las imágenes para navegar más rápido debido a sus conexiones lentas. Lo mismo pasa con los mapas de imágenes.

- Los applets u otros objetos programados no ofrecen un texto alternativo explicativo.
- Los scripts (Javascript, Vbscript) que no ofrecen una alternativa de uso para los usuarios que lo tienen deshabilitados en sus navegadores (por razones de seguridad, por ejemplo) o no pueden utilizarlos por navegar con navegadores parlantes, lectores de pantalla o navegadores de texto.

2. Lenguaje de marcado

El desconocimiento de los lenguajes de marcado en el diseño de páginas web hace que se usen alternativas no accesibles, como por ejemplo:

El uso de unidades de medida absolutas en vez de relativas (en texto, tablas, marcos, etc.) Impide que el usuario pueda agrandar el tamaño del texto de acuerdo a sus necesidades visuales. Otro problema relacionado con esto es

Cuando queremos imprimir una página web y, como dicha página tiene valores de ancho absolutos, queda una parte de la información fuera de la impresión. Asimismo, cuando una página está diseñada para una definición de pantalla específica (como cuando dice “Se ve mejor en 800 x 600” o “1024 x 768”) y se navega con una definición distinta, queda una parte de la información fuera de la ventana del navegador y deberemos estar atentos a que se habilita una barra de desplazamiento para que podamos movernos hacia la derecha (si es que aparece).

El uso de tablas en vez de hojas de estilo para ordenar el diseño de las páginas hace que muchos usuarios con navegadores de voz, textuales o con teléfonos celulares encuentren dichas páginas desordenadas y confusas.

Las tablas tienen una función específica que es el desplegar datos que necesitan ser mostrados en forma tabulada, y no para estructurar o “maquetar” páginas. El uso indiscriminado de tablas en el diseño web provoca confusión en los usuarios de navegadores parlantes, lectores de pantalla y navegadores de texto o que acceden a

través de teléfonos celulares o PDA's. Para evitar estos problemas no se debe usar tablas para organizar la estructura de las páginas (comúnmente se lo llama “maquetar”) ni para ningún efecto visual. Ejemplo:

Tabla 1 Ejemplo de mal uso

<p>Instrucciones para dar cuerda al reloj Por Julio Cortázar</p> <p>Allá al fondo está la muerte, pero no tenga miedo. Sujete el reloj con una mano, tome con dos dedos la llave de la cuerda, remóntela suavemente. Ahora se abre otro plazo, los árboles despliegan sus hojas, las barcas corren regatas, el tiempo como un abanico se va llenando de sí mismo y de él brotan el aire, las brisas de la tierra, la som</p>	<p>Bra de una mujer, el perfume del pan.</p> <p>¿Qué más quiere, qué más quiere? Átelo pronto a su muñeca, déjelo latir en libertad, imítelo anhelante. El miedo herrumbra las áncoras, cada cosa que pudo alcanzarse y fue olvidada va corroyendo las venas del reloj, gangrenando la fría sangre de sus rubies. Y allá en el fondo está la muerte si no corremos y llegamos antes y comprendemos que ya no importa.</p>
---	---

Fuente: Propia

Con un lector de pantalla o un navegador parlante, el usuario oiría el siguiente texto:

“Instrucciones para dar cuerda al reloj bra de una mujer, el perfume del pan.

Por Julio Cortázar ¿Qué mas quiere, qué mas quiere? Átelo... pronto a su muñeca, déjelo latir en libertad, Allá al fondo está la muerte, pero no tenga imítelo...”

3. Uso de la Web por personas invidentes

Obviamente, las personas invidentes no son capaces de ver las cosas tan bien como aquellas que no son videntes. Si bien es cierto que la mayoría de invidentes tienen algún grado de visión, a todos los efectos, se puede decir que las personas invidentes no utilizan sus ojos para acceder a la Web, porque la visión que tienen no es lo suficientemente útil para este tipo de tarea. Esto significa que un monitor de ordenador y

el ratón sería mucho menos útil para una persona invidente. No es que las personas invidentes sean incapaces de moverse o hacer clic con el ratón, es sólo que no saben a ciencia cierta por dónde se mueven o cuando hacen clic en un sitio no son capaces de percibir lo que cambia en la página, ya que no pueden ver lo que está sucediendo en el monitor.

4. Lectores de pantalla

Como resultado, hay programas de software que convierten el texto en voz sintetizada para que los ciegos sean capaces de escuchar a los contenidos web. Estos programas son generalmente llamados lectores de pantalla, a pesar de que hacen algo más de lo que su nombre indica. Los lectores de pantalla permiten a los usuarios navegar a través de contenido web de muchas maneras. El usuario puede simplemente dejar que el lector de pantalla lea toda la pantalla de arriba a abajo, una línea a la vez, o que el usuario pueda utilizar la tecla tabulación para navegar de enlace en enlace. El usuario también puede navegar de un encabezado al siguiente (si el contenido web tiene encabezados), de un marco al siguiente (si hay marcos), o por otros métodos. Muchos lectores de pantalla comunes incluyen JAWS, Window Eyes, y Home Page Reader.

Los lectores de pantalla también puede ser utilizado por aquellos que son sordos e invidentes, pero en lugar de convertir texto en voz, los lectores de pantalla para los sordo-ciegos pueden convertir texto en caracteres Braille actualizables en dispositivos Braille, como el de la siguiente foto:



Figura 9: lector braille

Los dispositivos de este tipo poseen pines pequeños que se pueden subir o bajar para formar los caracteres Braille que el individuo sordo-ciego puede sentir.

Estas tecnologías abren para las personas invidentes. Las personas ciegas ya no necesitan depender de otros para acceder a la información común que otros dan por sentado, tales como periódicos, revistas, estados de cuenta bancarios, y las transcripciones escolares. En el pasado, cuando los estudiantes invidentes querían inscribirse en una clase universitaria, tenían que recurrir a la ayuda de otros para llevar a cabo este propósito. Tenían que hacerlo en el horario de otras personas, y sacrificando parte de su intimidad personal. Ahora, con la ayuda de un lector de pantalla, los estudiantes invidentes son capaces de utilizar el sistema de registro basado en Internet en muchas universidades. Esta nueva capacidad les proporciona un mayor nivel de independencia que antes les era imposible obtener.

Por supuesto, el problema es que todo el contenido tiene que ser accesible a las tecnologías que utilizan las personas invidentes.

5. Limitaciones de los lectores de pantalla

Los lectores de pantalla aunque robustos, tienen limitaciones ya que no puede sustituir completamente a la experiencia visual. Afortunadamente, hay maneras de compensar las deficiencias del lector de pantalla.

- **Imágenes.**

Los lectores de pantalla no pueden describir las imágenes. La única manera que un lector de pantalla puede transmitir el significado de una imagen es mediante la lectura de texto en el documento que sirve como un sustituto o alternativa para esa imagen. Si no hay ningún texto alternativo o atributo alt, el lector de pantalla no precisa de la información necesaria para transmitir el significado de una imagen.

- **Diseño visual.**

Además, los lectores de pantalla no pueden estudiar la totalidad de una página web como un usuario visual podría hacerlo. Un usuario visual puede ver una página web y darse cuenta rápidamente de cómo se organiza para entonces concentrarse en el

contenido más importante. Un lector de pantalla no es capaz de hacer esto. Se lee de forma lineal, una palabra a la vez. No siempre se puede saltar de forma inteligente sobre el contenido de cuerpos extraños, como anuncios o barras de navegación.

- **Tablas de datos.**

Del mismo modo, los lectores de pantalla deben leer de manera lineal tablas de datos lo que puede llegar a ser muy confuso. Imagínese tratando de escuchar una tabla de datos de gran tamaño, con 14 columnas y filas 28. En el momento en que se lee la fila 20, el usuario ha olvidado lo que era el título de la columna 8. Puede ser difícil, si no imposible, interpretar la información.

- **Accesibilidad mediante teclado**

El hecho de que los usuarios de lectores de pantalla usan el teclado como su medio principal de navegación en la Web es un punto que los desarrolladores deben tener en cuenta. Sin siquiera darse cuenta de las consecuencias, algunos desarrolladores web hacen que los sus sitios sean únicamente accesibles mediante el ratón. La mayoría de las veces, la inaccesibilidad del teclado es el resultado de los eventos de JavaScript que dependen ya sea en el clic o el movimiento del ratón. Por ejemplo, cualquier evento provocado por el evento onmouseover sólo funciona cuando el usuario mueve el puntero del ratón sobre ese objeto en particular. No hay forma de acceder a este tipo de contenido utilizando un teclado solo, sin ofrecer otro método, ya sea como complemento o en lugar del evento mencionado.

El algunos casos, no hay necesidad de ofrecer una alternativa, como por ejemplo cuando el evento onmouseover desencadena un cambio visual en una imagen (por ejemplo, haciendo que el brillo de la imagen cambie o que se modifiquen algunos colores de la misma). A las personas invidentes, probablemente no les importe que la imagen cambie de color. Esta información es irrelevante.

3.1.3. Evaluación y síntesis.

3.1.3.1. Páginas con tamaño absoluto

Inconveniente: La página se deforma cuando se intenta aumentar o reducir el tamaño

Causas: Mala definición del layout, Uso de medidas en píxeles y no porcentuales,

Intención de ajustarse a resoluciones fijas, Uso de tecnologías anexas al HTML sin opción de ajuste de tamaño, Incompatibilidades con las herramientas de autor.

Impacto: Usuarios con baja visión, usuarios de dispositivos móviles, notebooks o pantallas no convencionales.

Mitigación tecnológica: Navegadores con lupa.

Escenarios: típicos, Generalizado.

3.1.3.2. Presencia de elementos intermitentes y rotativos

Inconveniente: El contenido no queda fijo, dificultando su lectura.

Causas: Elección de diseño.

Impacto: Usuarios de edad avanzada.

Mitigación tecnológica: Botón de pausa para el contenido rotativo.

Escenarios típicos: Espacio publicitario, periódicos, páginas con animaciones y elementos gráficos dinámicos.

3.1.3.3. Códigos de seguridad o captcha

Captcha es la sigla de “Completely Automated Public Turing test to Tell Computers and Humans Apart”. En general, esta técnica es empleada para evitar que robots programados puedan atacar al sistema en cuestión. Se le muestra al usuario una secuencia de números en un formato gráfico, deformados mediante un algoritmo de ruido aleatorio. El usuario debe interpretar esa secuencia e ingresarla [TuringTest].

Inconveniente: Los lectores de pantalla no pueden interpretar las secuencias (Si lo hicieran, no tendría sentido el sistema como freno ante los robots).

Causas: Necesidad de verificar la presencia de un humano para evitar ataques o estafas.

Impacto: Usuarios con baja visión, usuarios con ceguera.

Mitigación tecnológica: Ninguna ya que eso implicaría romper el algoritmo. La opción alternativa consiste en brindar esa secuencia en formato de audio, o proporcionar algún otro circuito que permita realizar esa operación.

Escenarios típicos: Alta y consulta de servicios (cuentas de correo electrónico, suscripciones, juegos en línea, pagos).

3.1.3.4. Uso de tecnologías anexas al html

Inconveniente: El contenido puede no ser compatible con el navegador o la plataforma.

Causas: Uso de tecnologías diferentes del estándar HTML para mostrar el contenido, que no contemplan una opción alternativa.

Impacto: Usuarios con tecnologías asistivas antiguas, usuarios con plataformas que no poseen o no soportan dichas tecnologías.

Escenarios típicos: Páginas con versiones recientes, páginas con animaciones o interacción gráfica, complementos Active-X bloqueados.

3.1.3.5. Falta De Soporte A La Navegación Por Teclado

Inconveniente: La página impide la interacción con el teclado.

Causas: Uso de tecnologías anexas al HTML sin soporte para teclados y demás dispositivos.

Impacto: Usuarios con problemas de motricidad, usuarios de edad avanzada, usuarios con diversos dispositivos de entrada basados en el teclado.

Escenarios típicos: Páginas que utilizan tecnologías diferentes al HTML, publicidades, animaciones.

Esta barrera además afecta a la usabilidad de una página.

3.1.3.6. Páginas Muy Extensas Sin Marcas o índice

Inconveniente: La cantidad de texto en la página obliga a los usuarios con lectores de pantalla a realizar un recorrido lineal de principio a fin. La navegación se ve entorpecida.

Causas: Falta un índice o de marcas de posición en el texto para separar párrafos y organizar el contenido.

Impacto: Usuarios con lectores de pantalla.

Mitigación tecnológica: Las versiones más recientes permiten la lectura no secuencial con marcas inteligentes.

Escenarios típicos: Contratos, resoluciones, normativas, publicaciones.

3.1.3.7. Uso de tablas anidadas para contener la información.

Es una práctica ampliamente utilizada, que debería ser reemplazada por estilos CSS.

Inconveniente: Los lectores de pantalla no interpretan correctamente el orden de la página, interfiriendo con su interpretación. (Las tablas usuales en el HTML definen las filas, y dentro de cada una las celdas. Eso define un recorrido por filas de izquierda a derecha).

Causas: Uso de tablas anidadas para la disposición de la información. Fue una práctica muy empleada cuando se dejaron de usar marcos en el HTML.

Impacto: Usuarios con lectores de pantalla.

Mitigación tecnológica: Lectores de pantalla que permiten la lectura no secuencial. Se podría mitigar desde la inteligencia del lector, aunque no sería lo correcto.

Escenarios típicos: Prácticamente todas las páginas que no utilizan estilos de forma correcta, o fueron hechas con herramientas de autor que resuelven las cuestiones de diseño con tablas anidadas.

3.1.3.8. Limitaciones en los tiempos de respuesta de las páginas

Es una barrera que también interesa a los fines de la usabilidad.

Inconveniente: La página no permite al usuario tener control de los tiempos, imposibilitando la navegación.

Causas: Presencia de animaciones o temporización de eventos con escaso margen.

Impacto: Usuarios de edad avanzada, baja visión.

Mitigación tecnológica: Desde el navegador, se puede solicitar una pausa o pedido de confirmación para cierto tipo de comportamiento.

Escenarios típicos: Sitios con animaciones, publicidad, contenido rotativo, formularios, exámenes, presentaciones de fotografías. PDFs protegidos.

Inconveniente: Los lectores de pantalla no pueden interpretar el texto de los documentos.

Causas: Por protección del material, el contenido se bloquea para evitar la copia.

Impacto: Usuarios con lectores de pantalla.

Mitigación tecnológica: Desde el software del lector, se podría implementar el reconocimiento de caracteres.

Escenarios típicos: Páginas con normativas, leyes, documentación digitalizada, publicaciones varias.

3.2. DISEÑO

Desde los inicios de la World Wide Web, tanto el diseño de las páginas como de la propia estructura de los hipertextos, ha variado enormemente. Si al principio, los hipertextos en la Web se limitaban a plasmar, sobre este nuevo medio, un texto impreso segmentándolo en pequeños bloques y colocando aquí y allá algunos enlaces, en la actualidad, las páginas web se han convertido en verdaderas obras de diseño gráfico, multimedia e ingeniería informática. Integración de bases de datos, servicios online, inclusión de sofisticadas herramientas de búsqueda y recuperación de información, dinamismo, usabilidad e interactividad, éstas son las claves de las páginas web actuales. Ya no importa sólo el contenido presentado, sino su diseño y, sobre todo, su funcionalidad.

En 1997, David Siegel en Técnicas avanzadas para el diseño de páginas web distinguía 3 generaciones en el desarrollo de las interfaces de la WWW:

- Webs de primera generación:
 - estructura lineal
 - eventual inserción de fotografías y líneas de separación
 - baja definición (proyectados para terminales ASCII en blanco y negro)
- Webs de segunda generación:
 - iconos en vez de palabras subrayadas con azul
 - menú de opciones
 - fondos coloreados o con imágenes
 - bordes azules alrededor de las figuras interactivas
- Webs de tercera generación:
 - hegemonía del diseño sobre la tecnología

- utilización de metáforas
- uso de un layout tipográfico y visual para la descripción de una página bidimensional
- estructura entrada -área central- salida

Sin embargo, hoy podemos hablar de una cuarta generación de webs en las que prima tanto el diseño gráfico como la tecnología. Los web masters ya no son sólo diseñadores gráficos, sino expertos informáticos que han de conocer los nuevos lenguajes, dominar las nuevas herramientas de programación, conectividad a base de datos, seguridad, etc. El hipertexto ya no es una colección de textos enlazados, sino un espacio de intercambio de servicios de todo tipo: culturales, informativos, comerciales, sociales, etc.

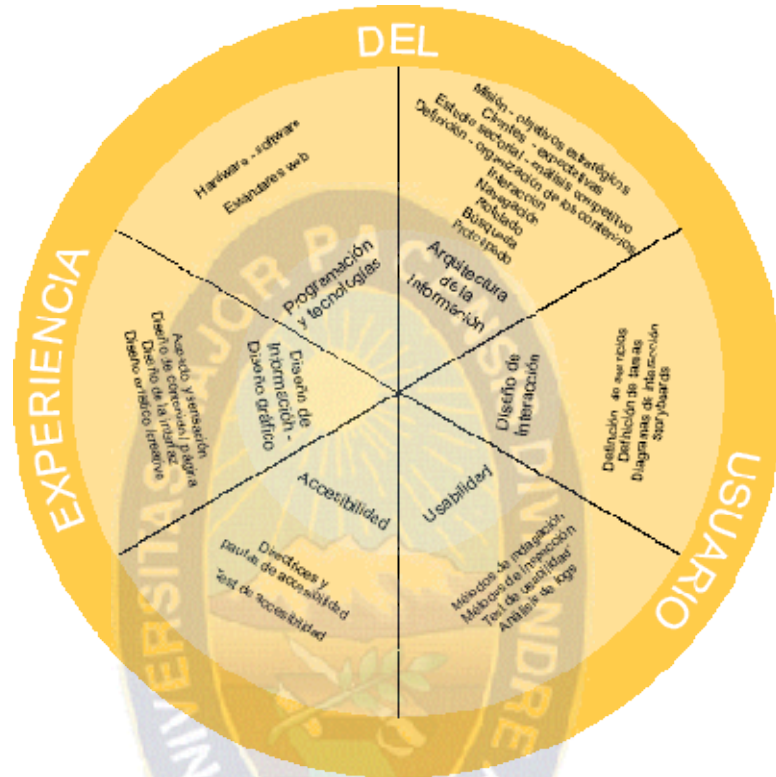
Son muchos los factores que hay que tener en cuenta a la hora de diseñar un sitio web. Sin embargo, dos factores destacan sobre los demás: su contenido y el diseño de la interfaz.

Francisco Tosete Herranz en La experiencia del usuario resume en la siguiente imagen, en forma de rueda, todas las disciplinas implicadas en el diseño de sitios web:

- arquitectura de la información: misión y objetivos estratégicos, clientes y sus expectativas, estudio sectorial/análisis competitivo, definición y organización de los contenidos, interacción, navegación, rotulado, búsqueda, prototipado, etc.
- diseño de la interacción: definición de servicios, definición de las tareas, diagrama de interacción, storyboards, etc.
- usabilidad: métodos de indagación, métodos de inspección, test de usabilidad, análisis de logs, etc.
- accesibilidad: directrices y pautas de accesibilidad, test de accesibilidad, etc.
- diseño de la información/diseño gráfico: aspectos y sensación, diseño de contenido página, diseño de la interfaz, diseño artístico creativo, etc.

- programación y tecnologías: hardware/software, estándares web, etc

Figura 9 Esquema propuesto



Fuente: Francisco Tosete Herranz

La World Wide Web ya no se entiende como un macro texto, sino como un espacio de relación y de servicios de todo tipo. Quizás haya triunfado la visión mercantil de la Web como un hipermercado global para hacer negocios, pero a la par, se han desarrollado otros servicios con fines menos crematísticos y con un enfoque centrado en la transmisión de conocimientos, la puesta en común y el desarrollo colaborativo para crear cultura, pensamiento, ciencia, saber y, en suma, conocimiento.

El proceso de diseño del sitio web abarca muchos y diferentes aspectos como son:

- **Estructuración del conocimiento:** delimitar el ámbito disciplinar, corpus de conocimiento y tema sobre el que vamos a tratar de acuerdo con el fin perseguido:

informar, investigar, educar, vender, etc. La estructuración del conocimiento en la web permite convertir la tradicional estructura secuencial en una estructura multisequencial, por tanto, se atenderá pues más a las relaciones entre conocimientos que a la información aislada.

- **Estructuración de la información:** para ello se debe fragmentar la información en nodos que posteriormente se organizarán estableciendo distintas categorías que atiendan a diferentes aspectos: relaciones jerárquicas, cronológicas, secuenciales, espaciales, etc. Además de los nodos, se precisa la creación de metanodos con información sobre otros nodos: sumarios, índices, tablas de contenido, etc. Para esta fase es muy útil la utilización de mapas conceptuales, no sólo para el diseño, sino con el fin de integrarlos dentro del propio hipertexto a modo de mapa de navegación junto a otras herramientas de navegación. Una de las ventajas que ofrece el hipertexto es la posibilidad de organizar la información de más de una forma, dejando al usuario la elección de en qué modo acceder a ella, por lo tanto, se pueden conjugar varios criterios de organización y ofrecer al usuario distintas formas de acceso para que elija la forma que le sea más cómoda o la más adecuada para los fines que persigue. La web permite una estructuración de la información modular y descentralizada que por medio de los enlaces se podrán conectar a otros módulos, atendiendo a diferentes perspectivas.

- **Planteamiento general del diseño:** hay que utilizar criterios de accesibilidad, plantearse el tipo de servicios y funcionalidades que se van a ofrecer y con qué fin, y cómo se van a presentar al usuario, es decir, diseñar su usabilidad. La finalidad del sitio web debe condicionar la interfaz para adaptarse a las expectativas del usuario (un sitio de ocio no puede tener un diseño serio, poco atractivo y aburrido), pero la interfaz debe ser, ante todo, funcional.

- **Organizar la información:** hay que dotar al sitio de una buena organización para que el usuario pueda localizar lo que busca de la forma más fácil, clara e intuitiva posible. Se deben crear tablas de contenido, índices, diferentes secciones con títulos significativos, etc. Hay que simplificar el contenido y no mezclar varios temas en una misma página, jerarquizar la información no sólo dentro del propio hipertexto, sino dentro de la propia página colocando la información más importante en las ramas

superiores o en la parte superior de la página dejando la información complementaria o más detallada para el final o para las ramas más bajas, agrupar las opciones importantes en la misma página, no ampliar el número de opciones en cada nivel hasta la extenuación. La organización de la información debe anticiparse en todo momento a los deseos del usuario y a las tareas más frecuentes para que el usuario llegue antes a ellas.

- **Tipos de información e interfaz de usuario:** disposición de la página en la pantalla (utilización de marcos o frames, distribución de los contenidos, colores, fondos, presencia de logotipos, etc.); elementos textuales (determinar la amplitud de cada uno de ellos, selección de la información, palabras clave, enlaces, formatos y estilos de letra); selección y elaboración de gráficos; inclusión y uso de audio, vídeo y animaciones; selección de iconos, botones, barras de menús; presencia de campos de ayuda para la orientación del usuario, etc.
- **Interrelación entre la información:** estructuración de los nodos de texto, audio, vídeo e interconexión de todas estas morfologías para convertir el hipertexto en hipermedia; establecimiento de enlaces entre documentos, partes de documentos, fragmentos de información, etc. dentro del propio documento y fuera de él.
- **Creación de un sistema de navegación:** que posibilite una navegación simple, intuitiva, consistente, transparente y flexible. Se trata de interrelacionar la información y la interfaz. Se pueden elaborar un gran número de herramientas que ayuden a la navegación en forma de botones, barras de navegación, uso de metáforas, mapas sensibles, FAQ o Preguntas más frecuentes (Frequently Asked Question) que faciliten la navegación y la hagan comprensible. Todas estas herramientas son imprescindibles si se trata de documentos largos y complejos en los que el peligro de pérdida del contexto aumenta cuando el usuario se aleja de la página principal y se adentra por las ramas inferiores de una estructura profunda o demasiado amplia.
- **Creación de un sistema de búsqueda y recuperación:** Para ello es necesario el uso de herramientas de representación de la información (metadatos, métodos de indización, etc.), formas de presentación de las consultas y los resultados, y otras herramientas de recuperación y búsqueda tanto de la información interna como externa.

Se pueden incorporar motores de búsqueda para que rastreen dentro del propio documento y/o en toda la World Wide Web.

No vamos a analizar aquí de forma pormenorizada cómo se crea una página web, ya que existen en la Web un gran número de manuales que lo explican de forma detallada, sin embargo, sí abordaremos 2 aspectos fundamentales para la accesibilidad de páginas, por un lado, el diseño de la estructura y, por otro, el diseño la página propiamente dicha.

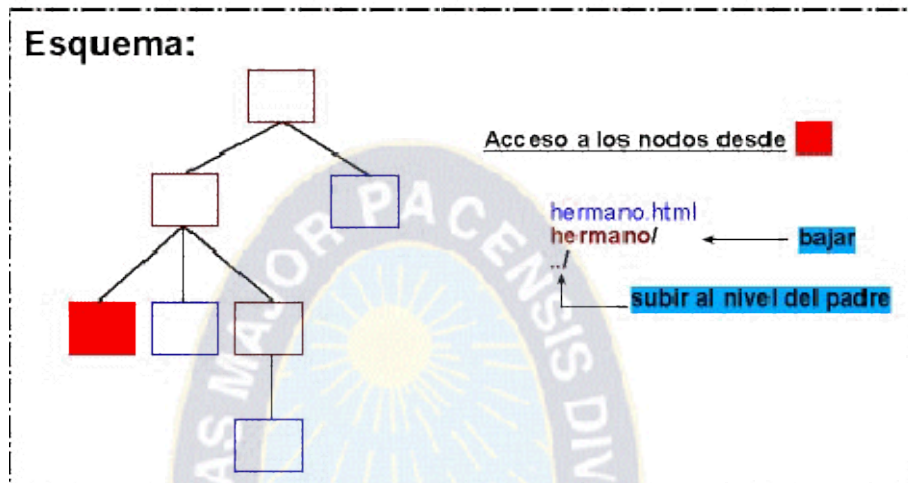
3.2.1.1. DISEÑO DE LA ESTRUCTURA

En la Web no existe una metodología concreta para crear hiperdocumentos o sitios web completos. Por el contrario, existen multitud de estructuras organizativas tanto en lo visual como en lo conceptual, que pueden provocar una gran desorientación al usuario. Es por eso, que a la hora de elaborar un hipertexto se deben establecer una serie de reglas para que el sistema sea coherente (visual y cognoscitivamente), y para que no se produzca duplicación de informaciones o reiteración de esfuerzos a la hora de recopilar o introducir la información, y de crear la red hipertextual. Como la macroestructura común de información debe constituirse a partir de una serie de microestructuras, éstas deben construirse siguiendo unas reglas mínimas que favorezcan su articulación en la estructura global. Por lo tanto, las reglas deben establecer:

- Los tipos de documentos que pueden entrar a formar parte del sistema
- Los datos de identificación del documento: título, autor, descripción, localización, fecha de introducción, fecha de la última consulta, nivel de acceso, etc.
- El formato de introducción del texto: estructuración del texto, posición de las páginas, modelo gráfico utilizado, etc.
- El tipo de relación entre los documentos: normas que indican qué documentos pueden relacionarse, lugares dentro del documento en donde se deben encontrar esas relaciones (en todo el documento o en una parte del mismo), presentación visual de enlaces, etc.

- El acceso a los documentos: recomendaciones para el uso del hipertexto en la búsqueda de información, etc.

Figura 10 Diseño de acceso a nodos



Fuente: propia

Los editores web actuales permiten automatizar una serie de tareas que antes constituían una afanosa labor, como la creación automática de tablas de contenido y herramientas de búsqueda, creación de barras de navegación y exploración o botones con enlaces de trayectoria vertical y horizontal (de páginas primarias a secundarias y viceversa, de orden secuencial anterior-siguiente-arriba, etc.) siguiendo el árbol y jerarquías establecidas en la estructura, etc. Un aspecto clave y fundamental es la facilidad que ofrecen estos editores para organizar los archivos y carpetas del sitio web y para construir la propia estructura del sitio y los enlaces entre documentos, ofreciendo esquemas visuales generales de todo el conjunto de nodos y sus relaciones jerárquicas y horizontales, o esquemas parciales de parte de esa estructura jerárquica y/o en red. Sólo hay que seleccionar la carpeta o archivo elegido, arrastrarla y soltarla sobre el lugar que queramos que ocupe en el árbol que representa la estructura de nodos, y ya tenemos creada de forma automática la red jerárquica y sus relaciones, los enlaces estructurales y de trayectoria, etc.

Por otro lado, los editores web también facilitan otras labores al ofrecer plantillas de diseño general de sitios web y esqueletos completos de estructuras y modelos para webs personales, webs de presencia corporativa, webs de discusión, webs de proyectos, webs de soporte al cliente, webs de colaboración en grupo, etc. Utilizando dichas plantillas, sólo tenemos que introducir el contenido y añadir o quitar las páginas que falten o sobren a lo largo y ancho de la estructura propuesta. Los editores actuales también suelen incluir una serie de herramientas y complementos que facilitan otras labores complejas o tediosas para el diseñador web, como asistentes para interfaz de bases de datos, inclusión de contadores, barras de vínculos, efectos dinámicos, subprogramas java, controles ActiveX, etc.

3.2.1.2. DISEÑO DE LA PÁGINA

Es importante tener en cuenta que la unidad básica de información de un documento hipertextual no es la página, sino la pantalla.

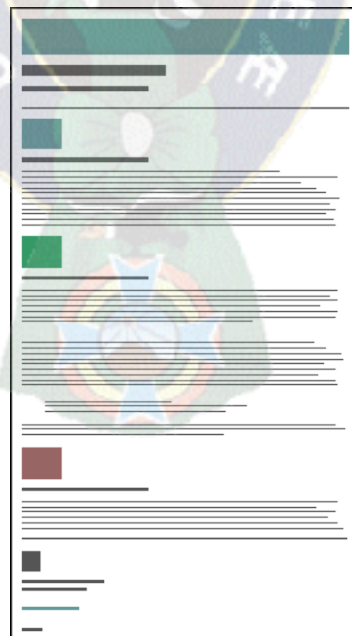
Así pues, el diseño de la página y la disposición de los elementos dentro de ella para ser vistos en pantalla, son uno de los aspectos principales a la hora de diseñar el hiperdocumento. Las páginas deben tener un esquema ordenado y legible de un vistazo. He aquí dos ejemplos de disposición de página, la figura 12 muestra una página ilegible y desordenada, mientras de la figura 13, que sigue un esquema ordenado, facilita la navegación y comprensión del contenido.

Figura 11 Diseño de página web inaccesible



Fuente: propia

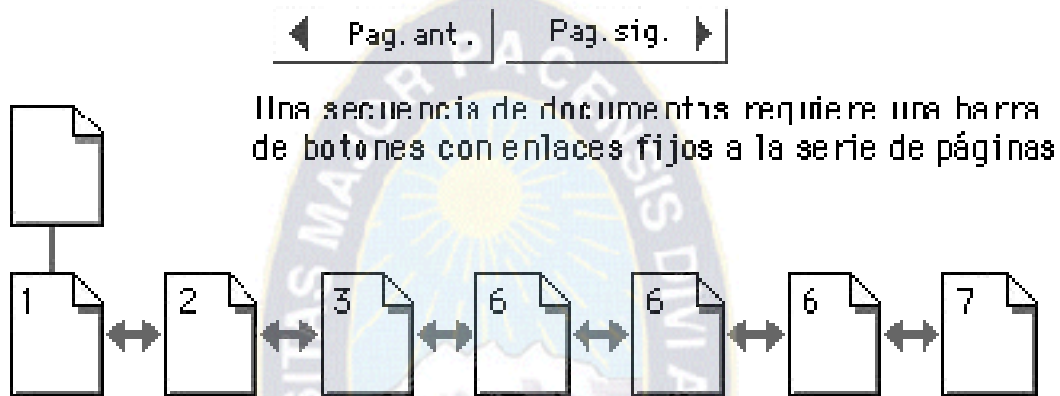
Figura 12 Diseño de página web accesible



Fuente: propia

En el encabezamiento de los documentos es imprescindible que aparezca el título destacado y el uso de gráficos sensibles o botones de cabecera que indiquen los recorridos posibles para orientar la navegación.

Figura 13 Diseño navegacional



Fuente: propia

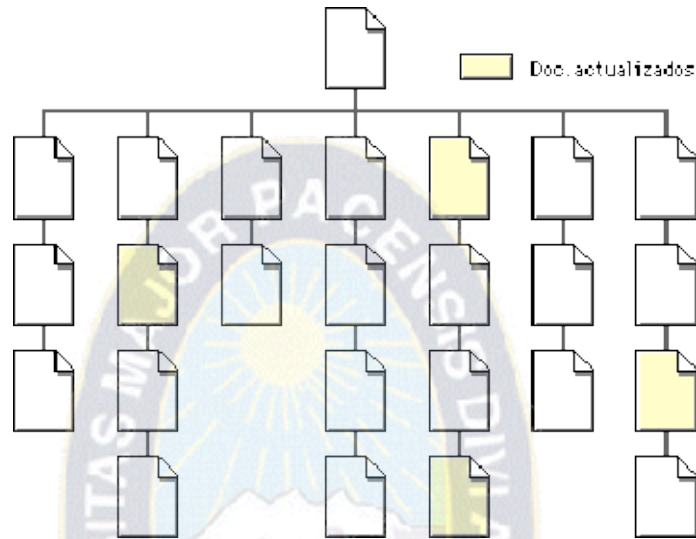
También es corriente la utilización de un logotipo u otro sello gráfico que identifique la imagen institucional u oficial, comercial, etc. del sitio web.

En el diseño de las páginas hay que tener en cuenta una serie factores como: enlaces locales y ayudas a la navegación, encabezamiento de documentos, tipografía (contraste visual, esquema y diseño de páginas, tipos de letras, establecimiento de títulos y subtítulos, etc.). Creación de pies de página con información sobre el autor, e-mail de contacto, enlaces a otras páginas relacionadas, fechas de creación y actualización, etc.

En este sentido, también los editores web actuales facilitan la labor de ofrecer un diseño de páginas homogéneo y coherente para el sitio web ya que ofrecen diferentes modelos y estilos de plantillas para todo un sitio web o para páginas concretas páginas principales, páginas de búsqueda, formularios, página de preguntas más frecuentes, tablas de contenido, página con diseño a dos columnas, etc., o realizan otras labores de forma casi automática, como poner al día las páginas si se utiliza un sistema modular de

actualización para tratar un conjunto enorme de páginas que puedan actualizarse de forma conjunta.

Figura 14 Estructura de una página web



Fuente: propia

Los documentos hipertextuales destinados a ser leídos mediante la pantalla de un ordenador tienen su propio lenguaje. En muchos casos, se convierte en una práctica común convertir un texto en hipertexto, por medio de la simple traslación de un texto a lenguaje HTML y la partición del texto secuencial completo en bloques de texto más pequeños que pasan a convertirse en nodos y que se conectan unos a otros mediante unos enlaces que siguen más o menos la misma estructura lineal que tenía el texto primitivo.

Hay que tener en cuenta que el texto impreso en papel necesariamente presenta una disposición para la lectura secuencial, mientras que el hipertexto permite otro tipo de disposición: la lectura multisequencial y que esta posibilidad o facultad de lectura se lleva a cabo por medio del establecimiento de otro tipo de estructuras que le diferencian del texto tradicional. El hipertexto en la Web implica una nueva manera de presentar el texto y el conocimiento que nada tiene que ver con el texto lineal.

Al transferir textos impresos a hipertexto, la primera operación que hay que realizar es convertir un texto completo en pequeños bloques de texto (o nodos) que luego uniremos en múltiples trayectos o recorridos mediante los enlaces. Sin embargo, este es sólo un primer paso. El diseño de un hipertexto es una labor muy compleja. Los editores de páginas web que, en realidad, son sistemas de gestión de hipertextos, nos facilitan esta labor.

Por supuesto que existen tipos de documentos, en los cuales no tiene ningún sentido separar el texto en bloques de nodos más pequeños para ser puestos en la red. Una novela de intriga se debe leer empezando por el principio y siguiendo, de forma secuencial, hasta el final. La tecnología hipertextual se debe aplicar con algún fin coherente. Es muy útil para las obras de consulta y referencia, los artículos y textos científicos, manuales, etc. Estas obras impresas ya contienen básicamente todo lo que también necesita el hipertexto: capítulos, párrafos, contenido, índices de palabras claves, referencias bibliográficas, etc. Estas obras no se leen página a página y el hipertexto permite hojear los bloques de información ayudado por la estructura de navegación y las búsquedas directas.

Un buen hipertexto tiene que ofrecer al usuario una estructura de navegación fácil de manejar, la fragmentación de la información en bloques debe tener sentido en sí misma y la estructuración de los bloques de información por medio de los enlaces deber ser coherente conceptualmente. El autor de un sitio web debe ofrecer al usuario toda la estructura y la infraestructura necesaria para facilitar la exploración, la navegación y las búsquedas mediante un lenguaje de fácil comprensión y utilización por el lector.

también es un aspecto clave a tener en cuenta que, en la Web, el lector puede convertirse en lector activo, y pasar de ser lector a ser usuario si se incluyen las herramientas adecuadas para que este usuario interactúe no sólo con los contenidos, sino con el propio autor del sitio web (por medio de un formulario, un e-mail de contacto, etc.), o con otros usuarios a través del establecimiento de un foro o una discusión en línea, un chat dentro del propio sitio web, etc.

Por supuesto, una vez creado un sitio web, lo más importante es darlo a conocer. Esta labor se realiza dando de alta el sitio en los principales buscadores e incluyendo metadatos para que los principales robots y agentes indiquen nuestras páginas.

En resumen, los web masters actuales deben afinar diferentes disciplinas a la hora de diseñar un sitio web, ya que la aproximación al diseño del sitio se debe hacer, entre otras, desde múltiples perspectivas: cognoscitiva, conceptual, ergonómica (características del diseño, la dotación física y el software lógico), funcional (qué servicios queremos ofrecer, etc.) y teniendo en cuenta ciertas condiciones de accesibilidad, usabilidad, diseño gráfico, diseño de la interacción, etc.

3.2.2. SEPARACION Y ESTRUCTURACION DE CODIGO CSS

Toda página Web a la que se le quiera dar un diseño CSS y reducirle las barreras al máximo, debe contener como base un documento perfectamente estructurado.

Al separar claramente contenido y diseño, se crea un documento con una estructura lógica muy marcada. Todo lo que pertenece al contenido se puede identificar claramente como tal sin necesidad de medidas auxiliares y el documento, a su vez, es perfectamente comprensible, independientemente de su representación visual.

La representación óptica de los contenidos se regula mediante instrucciones de formato centralizadas que se guardan, inteligentemente, fuera del documento en sí.

Eliminar los obstáculos para tener libre acceso al documento es relativamente fácil y representa un paso gigantesco en la supresión de barreras.

Al documento se le pueden añadir funciones de ampliación (por ejemplo, con JavaScript, PHP y similares), siempre que no dificulten el acceso a los contenidos.

- **Estructura lógica y textos resaltados.**

Las páginas Web son documentos. HTML no describe el aspecto de un documento o su ubicación. HTML describe qué es un documento: un título, una lista, una imagen, una

tabla, etc. Por eso los elementos y atributos de HTML se deben utilizar correctamente a nivel semántico, es decir, conforme a su significado según las especificaciones.

Un HTML semánticamente estructurado es la clave para conseguir una mejor accesibilidad, independencia de la plataforma, eliminar barreras y no menos importante, un diseño impecable.

Otro sinsentido (semántico) es insertar en el texto etiquetas vacías, o de cualquier otro tipo, simplemente para crear un determinado espacio, por ejemplo:

```
<p>Aquí tenemos un párrafo de texto</p>
```

```
<p>&nbsp;</p>
```

```
<p>Aquí tenemos otro párrafo de texto</p>
```

Lo mismo puede decirse para todos los demás elementos del texto, ya sean párrafos, listas, citas, formularios o tablas. No se puede perder el tiempo pensando en la fuente, tamaño o el color que le daremos a un elemento, ni qué distancias guardará respecto a los elementos contiguos. Lo importante aquí no es el aspecto óptico, sino el significado lógico, que deberá ser reconocible por todos desde una persona invidente a quien le leerán la página, o un robot buscador para el que la presentación no cuenta en lo más mínimo.

- **Agrupar los elementos de página**

Lo siguiente que se puede hacer es agrupar diferentes partes del documento en bloques lógicos (contenedores) que más tarde se podrán ordenar o formatear conforme al diseño visual que se desee. La división de la página Web de nuestro ejemplo, clásico y sencillo, es la siguiente:

- Header (encabezado)
- Main menu (menú principal)
- Content (contenido)
- Footer (pie)

Se crea un contenedor para cada área e identifique cada contenedor con alguna señal, por ejemplo, asignándole un ID. La etiqueta HTML <div> cobra aquí un significado especial. Este elemento bloque carece prácticamente de propiedades, ya que no define ningún párrafo determinado ni tabla, ni nada similar; simplemente delimita un área determinada. En el código fuente quedaría así:

```
<body>
<div id="header">Logo</div>
<div id="menu">Navegación principal</div>
<div id="content">Títulos y textos</div>
<div id="footer">Derechos reservados</div>
</body>
```

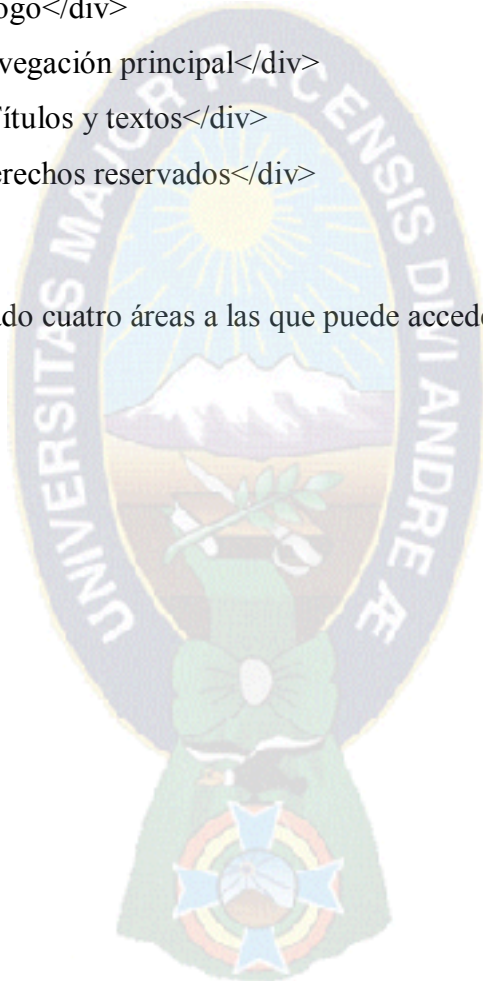
De ese modo ha creado cuatro áreas a las que puede acceder por separado con ayuda de CSS.

```
#header {
Valor propiedad;
}

#menu {
Valor propiedad;
}

#content {
Valor propiedad;
}

#footer {
Valor propiedad;
}
```



```
#header {  
Valor propiedad;  
}
```

Los contenidos de estos contenedores también se puede hacer accesibles y diseñar con relación al contexto, por ejemplo:

```
#content h1 {  
Valor propiedad  
}
```

```
#content p {  
Valor propiedad  
}
```

Con estas dos últimas instrucciones se hace que sólo se tengan en cuenta los títulos `<h1>` y los párrafos `<p>` que aparecen dentro del contenedor con ID `content`.

Naturalmente, los contenedores también se pueden anidar, es decir, se puede crear contenedores dentro de contenedores. Pero se debe preguntar antes seriamente si de verdad necesita contenedores dentro de contenedores, o si no es suficiente con la estructura semántica del contenido. Se quiere mantener el código fuente lo más claro y reducido posible. Con un anidamiento múltiple de contenedores `<div>`, por ejemplo, resulta superfluo que en la mayoría de los casos, ya que se puede obtener el mismo resultado por otros medios, por ejemplo dando formato directamente a los elementos que queremos vincular.

Se considera la página Web nuevamente desde un punto de vista conceptual. Al decidir seguir los estándares ya ha dado el paso más importante en la eliminación de barreras, pues queda implícito que se quiere dar total accesibilidad a la página. Por eso vale la pena dar un paso más y considerar qué más se podría hacer para eliminar barreras. Existen prejuicios inveterados en contra de la creación de páginas Web sin barreras, pero sin prejuicios que al analizar, se desvanecen como las falsas creencias que son.

3.2.3. Barreras en el diseño accesible

3.3.2.1 Imágenes y animaciones: Se usa el atributo “alt” para describir la función de cada elemento visual.

En el apartado 13.8 Cómo especificar texto alternativo de la especificación [HTML 4.01 Specification] se encuentra la siguiente definición:

Definiciones de atributos alt = texto [CS] Este atributo especifica texto alternativo para agentes de usuario que no puedan mostrar imágenes, formularios o aplicaciones. El idioma de este texto alternativo está especificado por el atributo lang varios elementos no textuales (IMG, AREA, APPLET e INPUT) permiten a los autores especificar texto alternativo que sirva como contenido cuando el elemento no pueda ser representado normalmente. El especificar texto alternativo ayuda a los usuarios que no tengan terminales gráficas, a los usuarios cuyos navegadores no soporten formularios, a los usuarios con discapacidades visuales, a aquellos que utilicen sintetizadores de voz, a aquellos que hayan configurado sus agentes de usuario para no mostrar imágenes, etc.

El atributo “alt” debe especificarse para los elementos IMG y AREA. Es opcional para los elementos INPUT y APPLET.

Si bien el texto alternativo puede ser muy útil, se tiene que tratarlo con cuidado. Los desarrolladores deberían seguir las siguientes pautas:

No especificar texto alternativo irrelevante cuando las imágenes incluidas sólo sirven para dar formato a una página, por ejemplo, alt="bola roja" sería inapropiado para una imagen que añade una bola roja para decorar un título o un párrafo. En tales casos, el texto alternativo debería ser la cadena vacía (""). En cualquier caso se aconseja a los autores que eviten usar imágenes para dar formato a las páginas, y que utilicen hojas de estilo en su lugar.

No especificar texto alternativo sin significado. Esto no solamente frustrará a los usuarios, sino que ralentizará a los agentes de usuario que deban convertir texto a salida por voz o Braille.

Los implementadores deberían consultar la sección sobre accesibilidad para información sobre cómo tratar los casos en que se omite el texto alternativo.

Normalmente, los lectores de pantalla tienen opciones para configurar como tratar el atributo alt y qué hace caso de que no esté. Si no está el atributo alt, normalmente leen el valor del atributo src de la imagen, lo cual no suele ser muy útil.

Para que se entienda lo que hacen los lectores de pantalla, que suelen ser difíciles de utilizar para una persona que no está acostumbrada a ellos, vamos a utilizar el programa Fangs para Mozilla Firefox. Este programa ofrece una representación textual de lo que un lector de pantalla lee.

La siguiente imagen muestra una página web con varias imágenes que se corresponde con el siguiente código HTML:

```
<body>
```

```
<h1>
```

```
<a href="anterior.html"></a>
```

```
<a href="indice.html">Índice</a>
```

```
<a href="siguiente.html"></a>
```

```
</h1>
```

```
<p>
```

Lista de carreras facultad ciencias puras y naturales

```
<br />
```

carrera informatica

carrera estadistica

carrera quimica

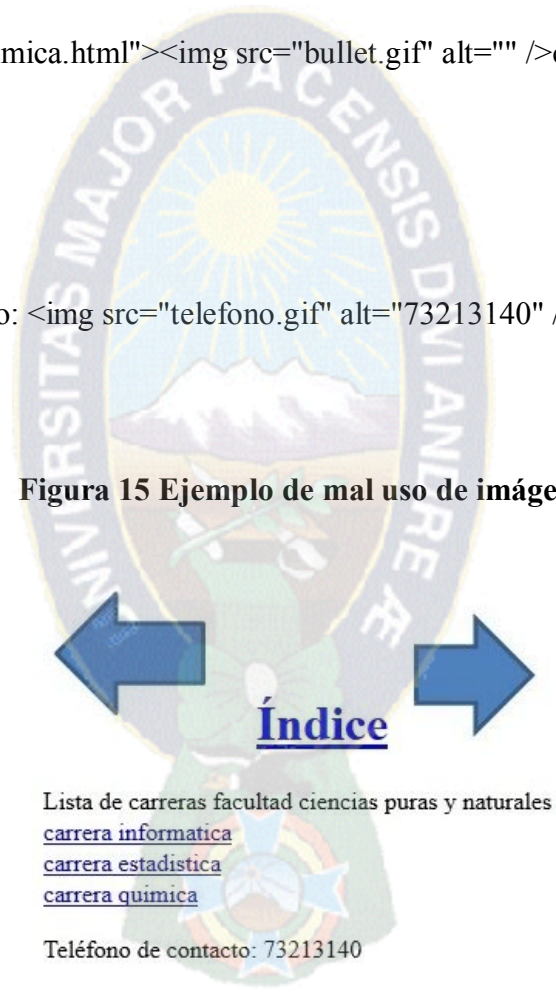
</p>

<p>

Teléfono de contacto:

</p>

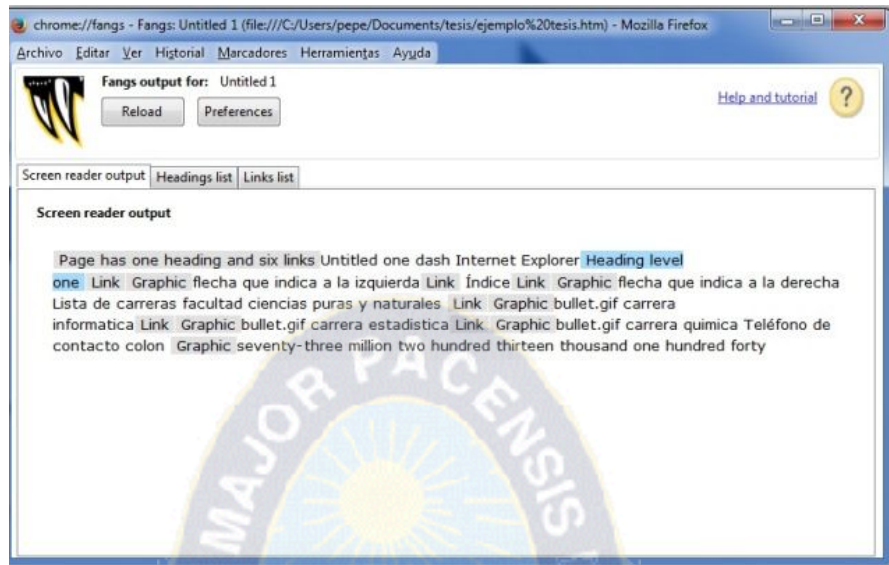
Figura 15 Ejemplo de mal uso de imágenes



Fuente: propia

Con el programa Fangs se obtiene la siguiente representación textual de la página:

Figura 16 Prueba realizada por Fangs



Fuente: propia

Fangs indica la aparición de algunos elementos de la página, como los enlaces (Link) o las imágenes (Graphic). Una vez corregido el texto alternativo de las imágenes de la página web de ejemplo se obtiene lo siguiente:

En esta página existen varios problemas:

- El texto alternativo que tienen las flechas describe las imágenes, pero no indica su función que es moverse hacia la anterior o la siguiente página.
- Las imágenes que se han empleado en las listas tienen como texto alternativo el mismo texto que las acompañan, por lo que aparece repetido dos veces.
- La imagen que contiene un número de teléfono tiene un texto alternativo que no es nada útil ("Teléfono de contacto ").

Para que esta página sea accesible, se puede modificar el texto alternativo de la siguiente forma:

- El texto alternativo de las flechas pasa a ser "Anterior" y "Siguiente". También se podría poner una descripción más larga como "Pasar a la página anterior" si "Anterior" puede ser ambiguo dentro del contexto de la página.
- El texto de las imágenes de las listas lo dejamos vacío, ya que es simplemente decorativo y no aporta ningún contenido útil.
- En el texto alternativo de la imagen con un número de teléfono ponemos el número de teléfono.

Con estos cambios, el código HTML de la página ahora es:

```

<body>

<h1>

<a href="anterior.html"></a>

<a href="indice.html">Índice</a>

<a href="siguiente.html"></a>

</h1>

<p>

Lista de carreras facultad ciencias puras y naturales

<br />

<a href="carrera_informatica.html">carrera
informatica</a>

<br />

<a href="carrera_estadistica.html">carrera estadistica</a>

```


carrera quimica

</p>

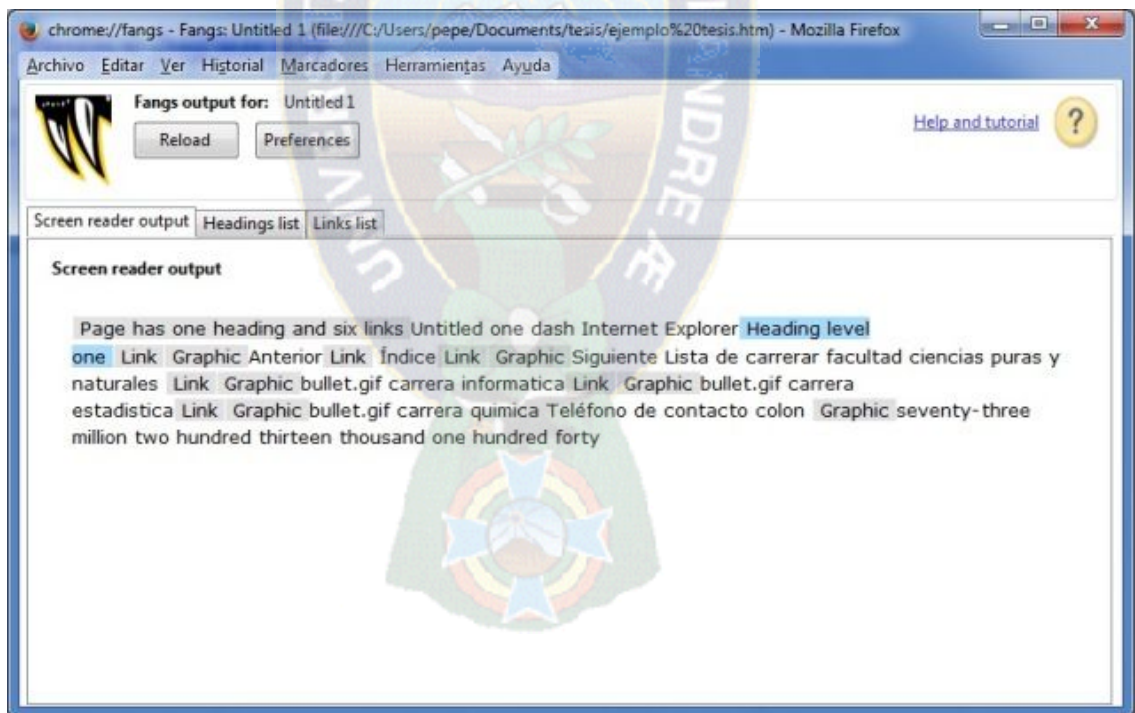
<p>

Teléfono de contacto:

</p>

</body>

Figura 17 Prueba con herramienta Fangs



Fuente: propia

Cuando una imagen no tiene texto alternativo o el texto alternativo es nulo (alt=""), Fangs muestra el valor del atributo src como representación alternativa. En la mayoría de los casos, este valor no es muy útil, ya que suele incluir nombres de directorios y el

nombre del fichero que contiene la imagen puede no tener sentido. Para resolver este problema lo mejor es no incluir las imágenes decorativas directamente en el HTML, sino emplear CSS para su inclusión.

3.2.3.1. Enlaces de hipertexto.

Algunos navegadores y algunos los programas de ayuda que emplean las personas con discapacidad (por ejemplo, los lectores de pantalla) ofrecen al usuario la posibilidad de mostrar, normalmente una ventana aparte, la lista de enlaces que contiene una página web. Esta lista de enlaces normalmente permite activar un enlace y navegar a la página de destino. Si el texto de un enlace no tiene sentido fuera de su contexto, el enlace no tendrá sentido en esta lista de enlaces.

Por otro lado, si los enlaces poseen un estilo especial para resaltarlos, los usuarios suelen fijar su atención en ellos, por lo que es importante que el texto de los enlaces sea lo más claro y significativo.

Un ejemplo el <http://www.eldiario.net>

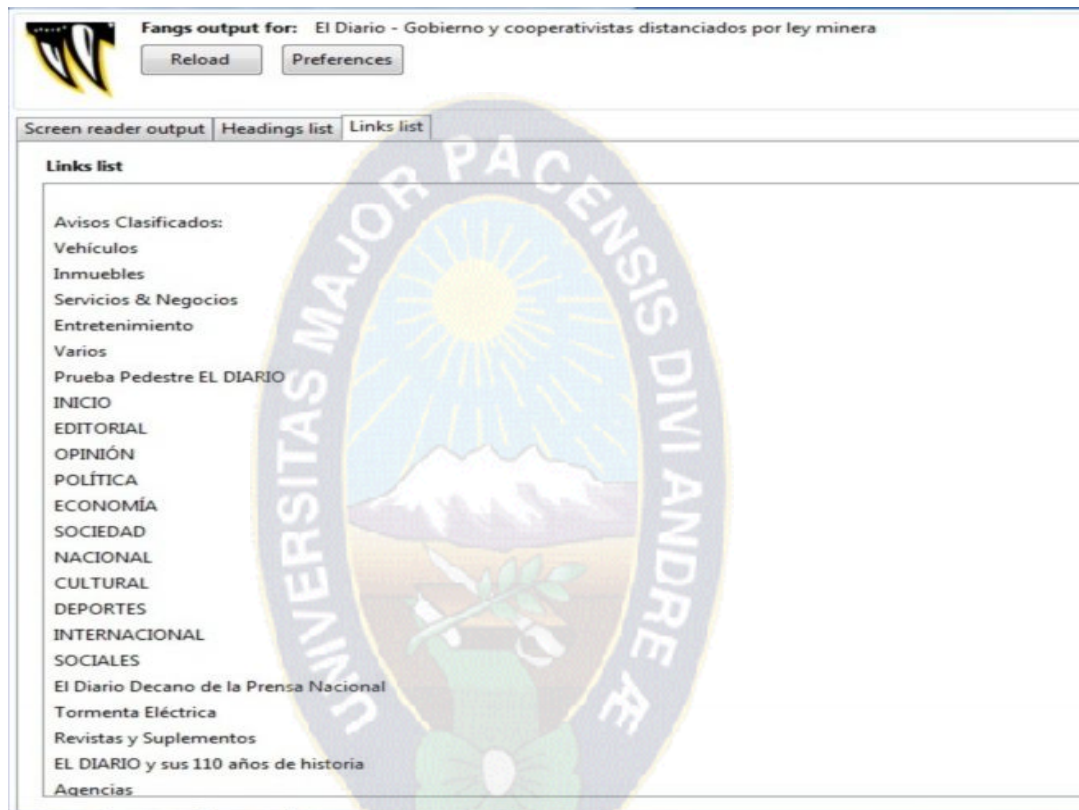
Figura 18 Página web “el diario”



Fuente: www.eldiario.net

Como se puede ver en la siguiente imagen, cuando se consulta la lista de enlaces con Fangs el texto de cada enlace es único y explica correctamente el destino del siguiente enlace:

Figura 19 Prueba realiza por Fangs a la página el diario



Fuente: propia

Cosas a tomar en cuenta en enlaces correctos:

- Tiene que ser claro y corto, pero no tan corto que sea casi inapreciable para el usuario cuando lo escuche.
- Tiene que tener sentido fuera de contexto, por ejemplo, cuando se lea él sólo o como parte de una lista de enlaces de una página.
- Tiene que tener sentido en el contexto, por ejemplo, cuando se lea como parte del resto de la página.

3.2.3.2. Organización de las páginas: use encabezados, listas y estructura consistente. use css para la maquetación donde sea posible.

Las páginas web tienen que estar correctamente estructuradas. Para ello, se tienen que emplear las etiquetas de HTML que definen la estructura de una página, como son: <title>, <h1>, <h2>, ..., , , <p>, <blockquote>.

Título de la página

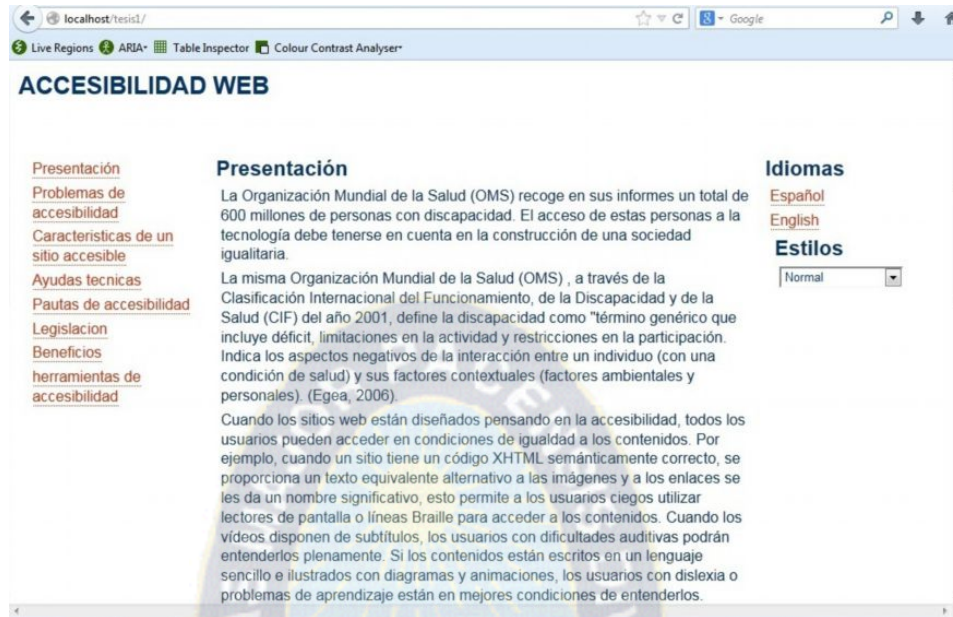
Toda página web debe tener un título definido con la etiqueta <title> que resuma el contenido o la función de la página.

El título de una página tiene que ser descriptivo pero a la vez breve. El título de la página se emplea en los marcadores de los navegadores y en los resultados que muestran los buscadores. El título de la página es lo primero que leen los lectores de pantalla y lo primero que aparece en los navegadores no visuales. Por ello, debe describir correctamente el contenido de la página. Pero como se lee en todas las páginas, debe ser corto y no repetitivo (no debe ser el mismo en todas las páginas). Es conveniente incluir el nombre del sitio web para definir el contexto de la página, pero el nombre debe ser breve y significativo.

Encabezados

El contenido de las páginas se tiene que estructurar con las etiquetas de encabezado <h1>, <h2>, La mayoría de los lectores de pantalla y algunos navegadores como Opera permiten al usuario desplazarse dentro de una página web saltando de un encabezado a otro encabezado. Eso permite llegar de una forma más rápida a la información que se busca. Por ejemplo:

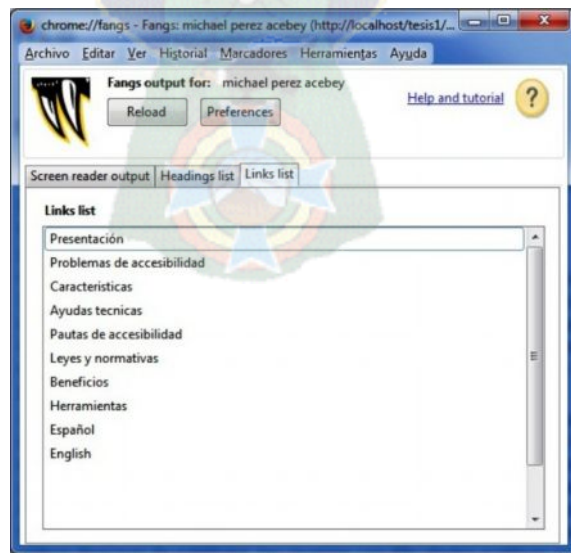
Figura 20 Pagina web accesible



Fuente: propia

En la siguiente imagen de Fangs, que simula el comportamiento de un lector de pantallas, podemos observar la lista de encabezados (Heading list) de la página anterior.

Figura 21 prueba realizada por Fangs



Fuente: propia

El ejemplo Encabezados explica que muchos desarrolladores de páginas web emplean las etiquetas `<div>` y `` para definir encabezados (títulos) de sección y estructurar el contenido. Sin embargo, estas etiquetas no proporcionan contenido semántico, por lo que se debe evitar su uso. Lo correcto es utilizar las etiquetas de encabezado (`<h1> ... <h6>`) y definir su presentación visual con CSS.

Se debe definir un estilo propio para cada sitio web, pero existen algunas reglas de aplicación general:

- Utilizar un único encabezado `<h1>` para el título principal del sitio web, como si fuera el título de un libro.
- Utilizar diferentes encabezados `<h2>` para cada apartado principal de la página (barra de navegación, contenido principal, pie de página), como si fuera la tabla de contenidos del libro. En este nivel no seas muy detallado, intenta que no sean muchos (no más de 10).
- Emplear el resto de encabezados (`<h3> ... <h6>`) para añadir mayor nivel de detalle en la estructura de la página. No todos los apartados pueden necesitar tanto nivel de detalle. Intenta no crear estructuras con mucha profundidad, normalmente no es necesario llegar a `<h5>` y `<h6>`.
- Ser consistente en el uso de los encabezados y no dejar huecos: por ejemplo, no pases de `<h2>` a `<h4>` sin utilizar `<h3>`.
- Si no se quiere que se vean los encabezados en un navegador visual, utiliza CSS para esconderlos. Para ello, no utilices `display: none;`.

Figura 22 Estructura accesible

The image shows a screenshot of the 'Accesibilidad Web' page from the University of Alicante. The page is titled 'Accesibilidad Web' and contains a list of guidelines for web accessibility. Red arrows point to specific elements on the page:

- An arrow points to the main title 'Accesibilidad Web' with the label '<h1>: Título del sitio web'.
- An arrow points to the sub-header 'Pautas de accesibilidad del contenido en la Web 1.0' with the label '<h2>: Título de la página'.
- An arrow points to the section header 'Niveles de accesión' with the label '<h3>: Apartado'.

The page content includes sections like 'Pautas de accesibilidad del contenido en la Web 1.0' and 'Niveles de accesión'. The 'Niveles de accesión' section defines three levels of accessibility based on the number of verification points met.

Fuente: WAI

Cuando un usuario con visión navega por una página web puede echar una vista rápida a toda la página para detectar ciertas zonas con el tamaño de letra más grande o con un color diferente que indican el inicio y fin de los apartados que forman el contenido. De

este modo, puede acceder al contenido que le interesa dentro de una página sin tener que leer toda la página. Los usuarios que emplean un lector de pantallas son capaces de actuar de la misma forma siempre que la página esté correctamente estructurada y etiquetada.

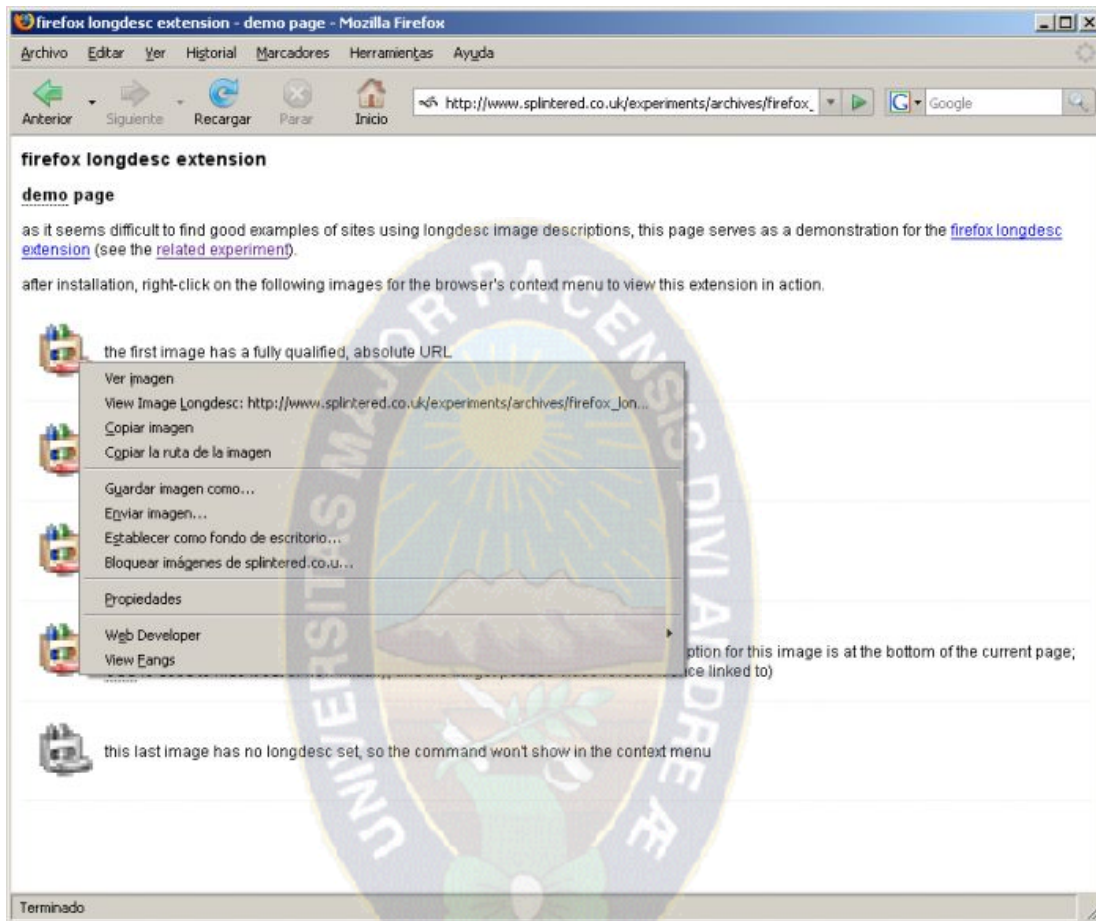
3.3.2.4 Figuras y diagramas: Describalos brevemente en la página o use el atributo longdesc.

El atributo longdesc complementa al atributo alt y se emplea para ofrecer una descripción más larga del elemento que la proporcionada por el atributo alt. Las etiquetas de XHTML que admiten este atributo en la versión 1.0 son: img, iframe y frame. Su valor tiene que ser una Uniform Resource Identifier (URI), la dirección de un recurso en Internet.

Mientras que el atributo alt contiene el texto alternativo de la imagen, el atributo longdesc contiene una dirección de internet a otra página web o a la misma página web donde se encuentra la descripción larga de la imagen. El atributo longdesc se emplea en situaciones donde la descripción es muy larga para ser incluida en el atributo alt, por ejemplo, cuando la imagen es un cuadro o un gráfico.

Pero sí que existe un complemento para Firefox llamado Longdesc que añade al menú contextual (botón derecho del ratón) una nueva opción llamada **View Image Longdesc** que permite navegar a la URL que tenga asignada la imagen, como podemos ver en la siguiente imagen que es una captura de la página firefox longdesc extension - demo page del autor del complemento:

Figura 23 Ejemplo de uso de longdesc



Fuente: Propia

3.3.2.4 Tablas

Una tabla de datos definida con la etiqueta `<table>` es fácil de entender si se puede ver toda ella en su conjunto, pero es muy difícil de entender si sólo se puede ver un dato aislado cada vez. Este problema lo sufren los usuarios que emplean navegadores no visuales, ya que ellos tienen que recorrer las tablas de forma lineal pero la naturaleza de una tabla es bidimensional, por lo que pierden la visión global de la tabla y pierden las referencias que permiten interpretar el contenido de cada celda de la tabla.

Para evitar este problema, se tienen que etiquetar correctamente las tablas para definir su título, se tiene que incluir un resumen que describa brevemente el contenido de la tabla, se tienen que definir los encabezamientos de las columnas y las filas y se tienen que emplear etiquetas y atributos especiales para asociar las celdas de encabezamiento y las celdas de datos para los encabezamientos más complejos.

- **El título**

Una tabla debe tener un título que proporcione una descripción breve de la tabla. Para evitar que existan dudas, el título se tiene que definir dentro de la tabla. Para definir correctamente el título de una tabla, se tiene que emplear la etiqueta `<caption>`.

De acuerdo con la especificación de `caption` en HTML, la etiqueta `<caption>` es opcional y tiene que ser el primer elemento que contenga una tabla (justo después de la etiqueta `<table>` y sólo puede haber un título en una tabla).

- **El resumen**

El resumen permite definir una descripción larga de la tabla que complemente al título de la tabla. La descripción tiene que incluir una explicación sobre el contenido y sobre la estructura de la tabla (por ejemplo, se debe explicar el propósito de la tabla, el número de filas y de columnas que contiene y una descripción de los encabezamientos). Además, también puede explicar la relación que guarda la tabla con el resto de la página.

En HTML, el resumen de una tabla se define con el atributo `summary` de la etiqueta `<table>`. Según la especificación de `summary`, los navegadores no visuales, como los lectores de pantalla o los basados en el uso de una línea braille, deben transmitir el contenido de este atributo a los usuarios.

- **Encabezamientos**

Al recorrer una tabla de forma lineal, se pierde la visión global y es muy difícil identificar el significado de un dato. De forma aislada, el contenido de una celda puede no tener sentido si no se sabe en qué fila y en qué columna está situado. Para evitar esta

situación, podemos usar los encabezamientos (o encabezados), que permiten asociar un dato con su encabezado.

Un encabezado de una tabla se define con la etiqueta <th>. Esta etiqueta es similar a la etiqueta <td> (se puede usar una en el lugar de la otra) y por tanto, ambas definen una celda de una tabla, pero <th> indica que la celda tiene una función especial y contiene un encabezado.

Con la etiqueta <th> se pueden definir tanto encabezados verticales como horizontales.

En el siguiente ejemplo define una tabla sencilla formada por 4 filas y 7 columnas que muestra el promedio de alumnos que ingresan a las carreras de informática, estadística, matemática. La primera fila de la tabla contiene los encabezados de columna y la primera columna contiene los encabezados de fila:

```
<?xml version="1.0" encoding="iso-8859-1"?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="es" lang="es">

<head>

<title>Página con una tabla sencilla</title>

<style type="text/css">

table {

border: 2px solid #000;

margin-left: auto;

margin-right: auto;
```



```

}

th, td {

border: 2px solid #999;

}

caption {

caption-side: bottom;

}

</style>

</head>

<body>

<table summary="tabla de informacion de promedio de alumnos que entran a la facultad
de ciencias puras y naturales se nota un desenso en informatica">

<caption>Alumnos promedio que entran a las carreras de
Informatica,Matematica,Estadistica</caption>

<tr>

<th>Alumnos</th>

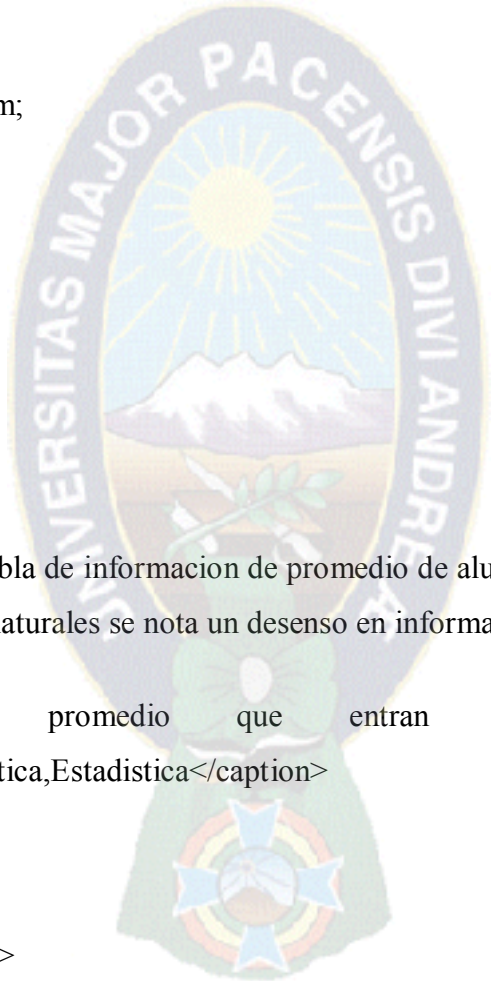
<th>2008</th><th>2009</th><th>2010</th>

<th>2011</th><th>2012</th><th>2013</th>

</tr>

<tr>

```



```

<th>informatica</th>

<td>150</td><td>130</td><td>120</td>

<td>100</td><td>90</td><td>80</td>

</tr>

<tr>

<th>Estadística</th>

<td>103</td><td>104</td><td>106</td>

<td>108</td><td>110</td><td>113</td>

</tr>

<tr>

<th>Matemática</th>

<td>84</td><td>85</td><td>87</td>

<td>85</td><td>87</td><td>90</td>

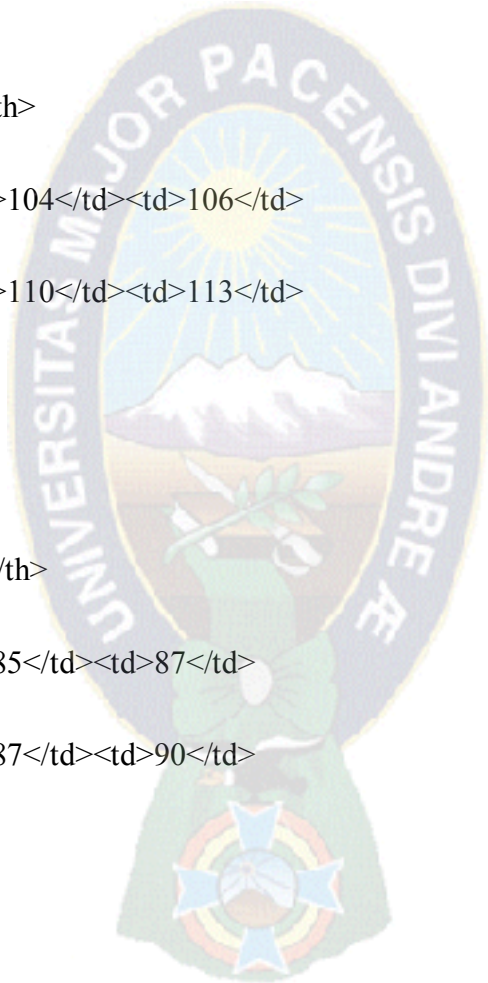
</tr>

</table>

</body>

</html>

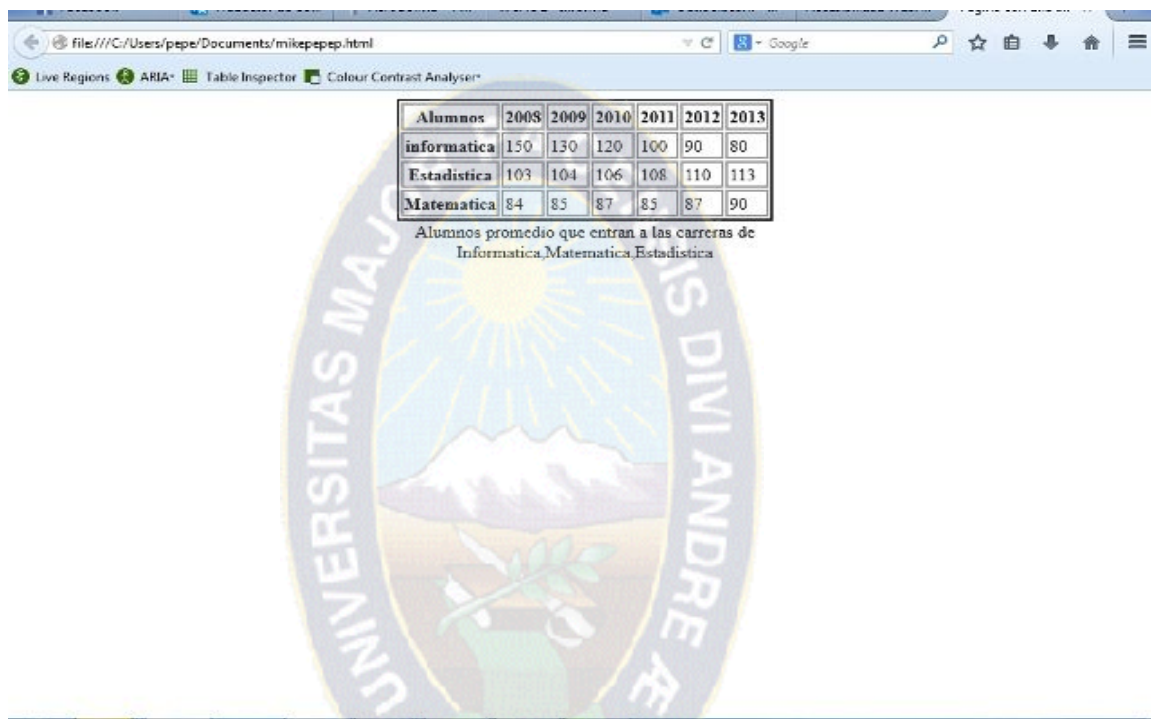
```



En la siguiente imagen se puede ver cómo se visualiza este código HTML en un navegador, el título que se ha definido con la etiqueta <caption> aparece en la parte

inferior de la tabla, mientras que el resumen que se ha definido con el atributo summary no es visible y en principio no es accesible con un navegador normal como Firefox.

Figura 24 Uso correcto de tablas



Alumnos	2008	2009	2010	2011	2012	2013
informatica	150	130	120	100	90	80
Estadística	103	104	106	108	110	113
Matemática	84	85	87	85	87	90

Alumnos promedio que entran a las carreras de
Informatica,Matemática,Estadística

:Fuente: Propia

Sin embargo, podemos emplear el complemento Juicy Studio Accessibility toolbar para revisar la estructura y contenido de una tabla y entonces sí que podemos visualizar el contenido del atributo summary, tal como podemos ver en la siguiente imagen:

Figura 25 revisión con Juicy Studio Accessibility toolbar

Summary: tabla de informacion de promedio de alumnos que entran a la facultad de ciencias puras y naturales se nota un desenso en informatica

Alumnos	2008	2009	2010	2011	2012	2013
informatica	150	130	120	100	90	80
Estadistica	103	104	106	108	110	113
Matematica	84	85	87	85	87	90

Alumnos promedio que entran a las carreras de Informatica,Matematica,Estadistica

Fuente: propia

Encabezamientos más complejos

Para tablas con encabezamientos más complejos, donde pueden existir varios niveles de encabezamiento, se puede emplear los atributos scope y headers para definir la relación que existe entre las celdas de encabezamiento y las celdas de datos.

En tablas sencillas ambos atributos se pueden emplear de forma equivalente, pero para tablas más complejas se tiene que emplear el atributo headers. En principio, el uso de los atributos scope y headers no afecta a la presentación visual de la tabla.

El atributo scope define el conjunto de celdas para las cuales la celda sobre la que se aplica proporciona información de encabezamiento. Puede tomar cuatro posibles valores:

- row: La celda proporciona información de encabezamiento para el resto de celdas de la fila que la contiene.
- col: La celda proporciona información de encabezamiento para el resto de celdas de la columna que la contiene.
- rowgroup: La celda proporciona información de encabezamiento para el resto del grupo de filas que la contiene.
- colgroup: La celda proporciona información de encabezamiento para el resto del grupo de columnas que la contiene.

El atributo headers permite definir una lista de celdas de la tabla que proporcionan información de encabezamiento para la celda actual. El valor de este atributo es una lista separada por espacios en blanco de los identificadores de las celdas de encabezamiento; las celdas de encabezamiento se identifican con el atributo id. Por tanto, el atributo headers permite definir encabezamientos más complejos que con el atributo scope.

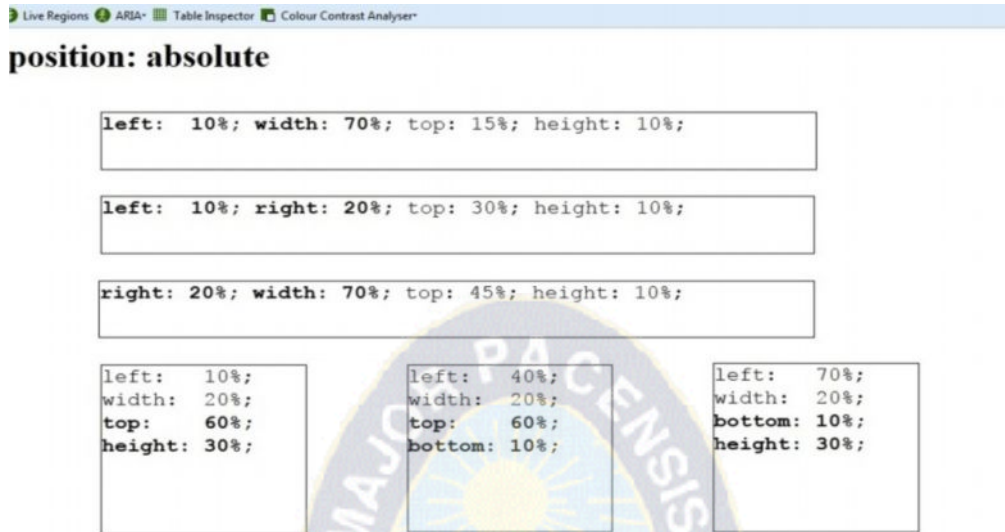
3.2.3.3. POSICIONAMIENTO ABSOLUTO

El posicionamiento absoluto se emplea para establecer de forma exacta la posición en la que se muestra la caja de un elemento. La nueva posición de la caja se indica mediante las propiedades top, right, bottom y left. La interpretación de los valores de estas propiedades es mucho más compleja que en el posicionamiento relativo, ya que en este caso dependen del posicionamiento del elemento contenedor.

Cuando una caja se posiciona de forma absoluta, el resto de elementos de la página se ven afectados y modifican su posición. Al igual que en el posicionamiento relativo, cuando se posiciona de forma absoluta una caja es probable que se produzcan solapamientos con otras cajas.

En el siguiente ejemplo, se posiciona de forma absoluta:

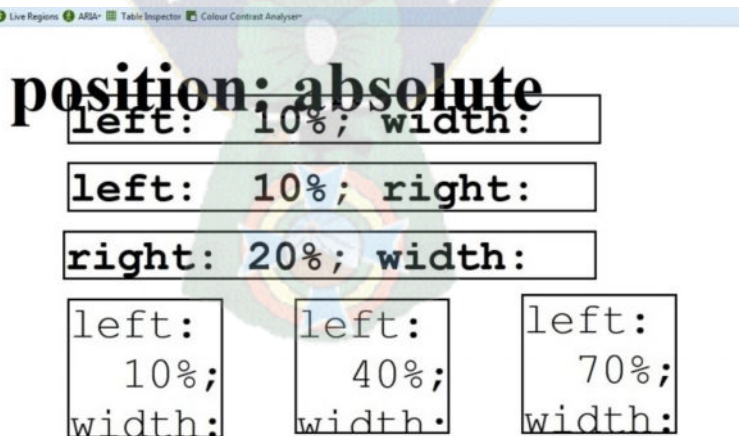
Figura 26



Fuente: propia

Es así como se ve al aumentar la pantalla para personas que tienen problemas de visión baja.

Figura 27



Fuente: propia

Se recomienda que no se use la posición absoluta ya que tiene problemas de accesibilidad.

3.2.3.4. Marcos accesibles

Cada marco debe tener un título y una descripción. Para ello, se tienen que emplear los atributos title y longdesc en la etiqueta <frame />. El atributo longdesc realiza el mismo papel que con la etiqueta se emplea para ofrecer una descripción más larga del elemento que la proporcionada por el atributo title. Por ejemplo:

```
<frameset cols="10%, 90%" title="Biblioteca de documentos electrónicos">  
  
  <frame src="nav.html" name="navega" title="Barra de navegación"  
  longdesc="descripcion.html#barra" />  
  
  <frame src="doc.html" name="contenido" title="Documentos"  
  longdesc="descripcion.html#documentos" />  
  
</frameset>
```

Además, se puede proporcionar una versión alternativa sin marcos para aquellos agentes de usuario (navegadores) que no sean capaces de interpretar los marcos. Para ello se tiene que emplear la etiqueta <noframes>.

El siguiente ejemplo completo muestra cómo crear una página web dividida en varios marcos (como es un ejemplo, el origen de cada marco es la misma página):

```
<?xml version="1.0" encoding="iso-8859-1"?>  
  
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">  
  
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="es" lang="es">  
  
<head>  
  
<title>Página con marcos complejos</title>  
  
</head>
```

```

<frameset cols="10%,*,10%">

  <frameset rows="20%,*">

    <frame src="doc.html" name="promo" title="Promociones" />

    <frame src="nav.html" name="barranavega" title="Barra de navegación global del
siti" longdesc="marcos-longdesc.html#barranavega" />

  </frameset>

  <frame src="doc.html" name="noticia" title="Noticia seleccionada - contenido
principal" longdesc="frameset-desc.html#noticia" />

  <frameset rows="*,20%">

    <frame src="doc.html" name="indice" title="Índice de otros titulares nacionales"
longdesc="frameset-desc.html#titulares" />

    <frame src="doc.html" name="espacioanuncio" title="Publicidad" />

  </frameset>

  <noframes>

    <p><a href="noframes.html">Versión sin marcos</a></p>

  </noframes>

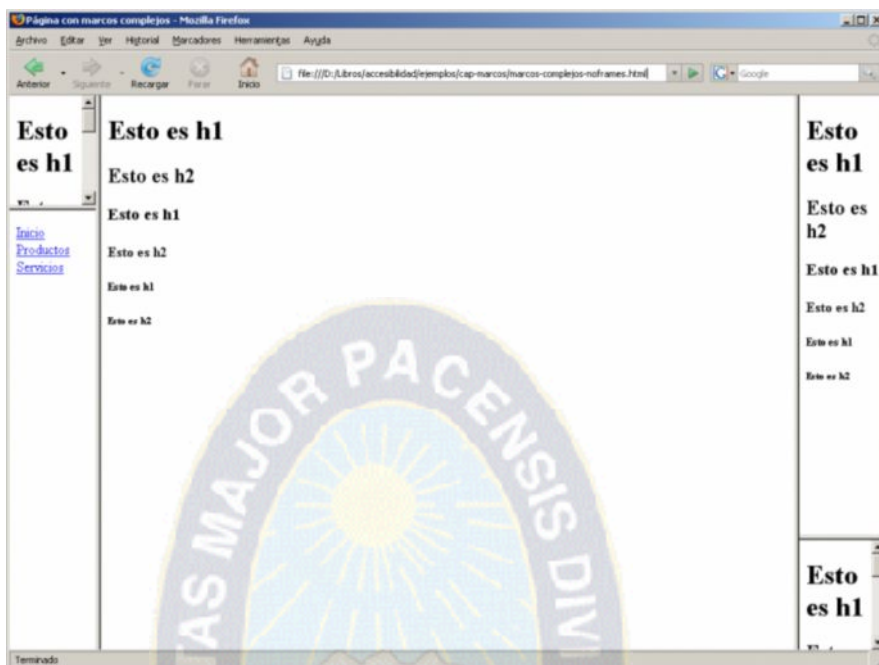
</frameset>

</html>

```

En la siguiente imagen podemos observar cómo se visualiza esta página con **Mozilla Firefox** :

Figura 28 Ejemplo de uso de marcos



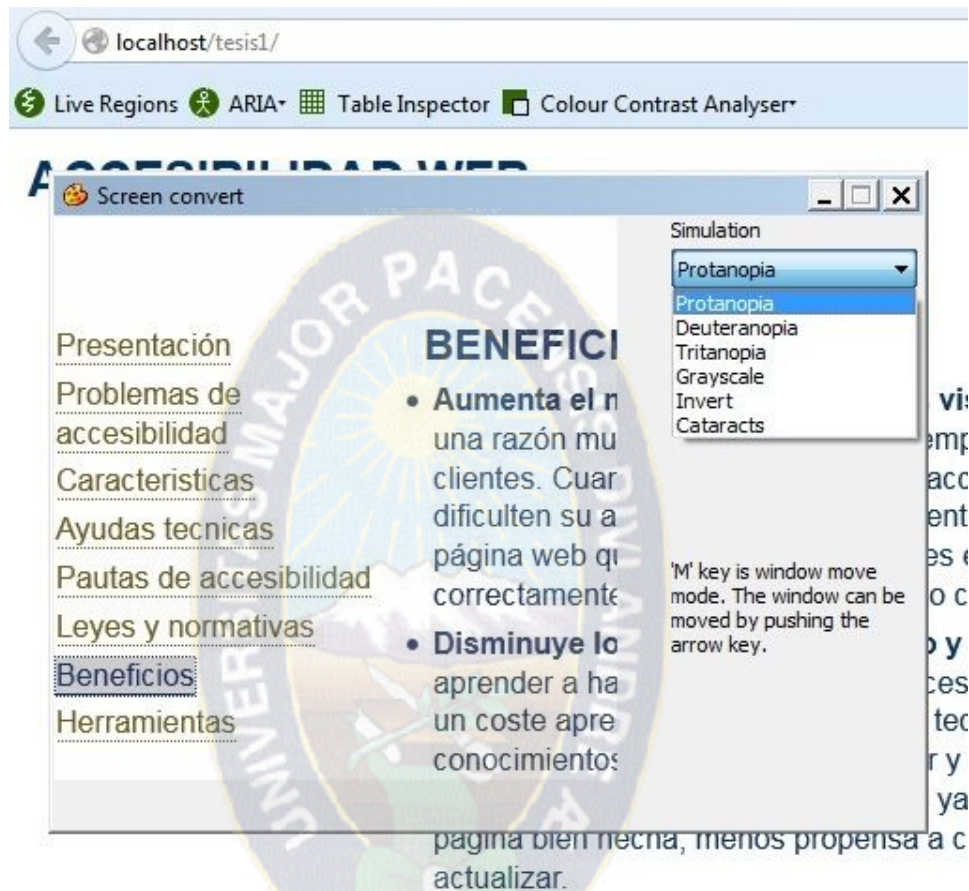
Fuente: Propia

Los navegadores tradicionales no los ignoran completamente. Sin embargo, algunos de los navegadores alternativos sí que los tienen en cuenta.

3.2.3.5. PROBLEMAS DE CATARATAS PROTANOPIA, DEUTERANOPIA Y TRITANOPIA

Para estas enfermedades mencionadas en el capítulo 2 se utiliza la herramienta Colour contrast analyzer que simula las enfermedades visuales así se puede saber si la página web es accesible para este tipo de enfermedades.

Figura 29 percepción de enfermedades visuales



Fuente: propia

3.3. CÓDIGO

3.3.1. Controles de formulario accesibles

3.3.1.1. Text

Nombre:

Aquí está el código HTML:

```
<label for="name">Nombre:</label>
<input id="name" type="text" name="textfield" />
```

Se observa que los valores de for y id son los mismos, por lo que la asociación de la etiqueta con el elemento del formulario es adecuada.

3.3.1.2. Textareas

Pon tu dirección:

Aquí está el código HTML:

```
<label for="address">Pon tu dirección:</label><br />
<textarea id="address" name="addresstext"></textarea>
```

Seleccione su carrera:

- Informatica
- Estadistica
- Matematica

Aquí está el código HTML:

```
<fieldset>
<legend> Seleccione su carrera: </legend>
<input id="inf" type="checkbox" name="toppings" value="inf" />
<label for="inf">Informatica</label><br />
<input id="est" type="checkbox" name="toppings" value="est" />
<label for="est">Estadistica</label><br />
<input id="mat" type="checkbox" name="toppings" value="mat" />
```

```

<label for="matematica">Matematica</label><br />
<input id="olives" type="checkbox" name="toppings" value="olives" />
<label for="olives">Olivas</label>
</fieldset>

```

3.3.1.3. RADIO BUTTONS

Elija tu carrera:

- Informatica
- Estadistica
- Matematica

Aquí está el código HTML:

```

<fieldset>
<legend>Elija un método de envío:</legend>
<input id="overnight" type="radio" name="shipping" value="overnight" />
<label for="Informatica">Informatica</label><br />
<input id="Informatica" type="radio" name="shipping"
value="Informatica" />
<label for="Estaditica"> Estaditica </label><br />
<input id=" Estaditica " type="radio" name="shipping" value="
Estaditica" />
<label for="Matematica">Tierra</label>
<input id=" Matematica " type="radio" name="shipping" value="
Matematica" />
</fieldset>

```

3.3.1.4. SELECT MENUS

Elige tu ciudad favorita?

Aquí está el código HTML:


```
<label for="city">Elige tu ciudad favorita?</label>
<select id="city" name="select">
<option value="1">La paz</option>
<option value="2">Oruro </option>
<option value="3">Cochabamba</option>
<option value="4">Santa Cruz</option>
<option value="5">Beni</option>
<option value="6">Pando</option>
<option value="7">Sucre</option>
<option value="8">Tarija</option>
<option value="9">Potosi</option>
</select>
```

3.3.2. CAPTCHA

Los captchas, esas imágenes con letras y números distorsionados que hay que descifrar para completar muchos formularios son cada vez más difíciles, se aplican más variaciones, más distorsiones y se introduce más ruido para dificultar la resolución automática por parte de los ordenadores, pero a su vez también son más difíciles para las personas.

Desgraciadamente, los captchas que normalmente se emplean presentan problemas de accesibilidad para algunos grupos de usuarios. En concreto, los captcha bloquean el acceso a muchos usuarios que padecen algún tipo de discapacidad. Las personas con visión reducida, como pueden ser los daltónicos que tienen dificultades para distinguir algunas combinaciones de colores o las personas que usan magnificadores de pantalla para ampliar el tamaño de lo que se visualiza en una pantalla pueden tener graves problemas para distinguir el texto que se muestra en un captcha. Por otro lado, es evidente que las personas ciegas que utilizan un lector de pantalla no pueden contestar los captcha basados en imágenes con texto en su interior, ya que estas imágenes no pueden incluir en el atributo "alt" de la etiqueta el texto que aparece escrito en los captcha, ya que entonces un ordenador también lo podría leer y podría pasar la prueba. Además, las personas con algún tipo de discapacidad cognitiva o intelectual como la dislexia, también pueden tener problemas a la hora de interpretar el texto que contiene un captcha.

El problema de los captcha aparece como uno de los principales problemas de accesibilidad de las páginas web en la actualidad.

3.3.3. Javascript

Uno de los problemas con los que nos encontramos a la hora de desarrollar aplicaciones Web con JavaScript, son los problemas de accesibilidad derivados al no contemplar la posibilidad de que un usuario nos visite con un navegador que no interprete este lenguaje. Una técnica para corregir este problema sería el separar JavaScript de las otras 2 capas del desarrollo Web (estructura y diseño), esto recibe el nombre de JavaScript no obstructivo o Javascript accesible.

La frase, “Divide y vencerás” se adapta perfectamente a esta idea, en la cual separaremos cada capa en su respectivo fichero, de forma que en cuanto a mantenimiento (la etapa más costosa y larga del desarrollo de una página web) e incluso la comprensión de la aplicación, se verán afectadas positivamente en cualquier aplicación.

3.3.3.1. División De Código

El desarrollo Web ha cambiado en los últimos años, estamos dejando de mezclar la presentación de la estructura, lo que nos está facilitando el trabajo de mantenimiento, modificación y de las aplicaciones. El código que hace años se usaba está dando lugar a otro más complejo y fácil de mantener.

3.3.3.2. Reglas Del Javascript No Obstructivo

1. Nunca se incluye JavaScript directamente en el documento.

Una de las mayores ventajas del JavaScript es que al igual que CSS permite la ejecución en ficheros externos. Esto permite incluir los ficheros que se necesita para cada página, ajustando al máximo el peso de la aplicación. Además si has se cambia alguna funcionalidad de la aplicación, únicamente se modifica el fichero JS en cuestión. Pero se puede acceder de esta propiedad se tiene que hacer referencia al JavaScript desde el

fichero HTML, añadiendo entre los tags la siguiente línea y haciendo alusión al fichero que se desea adherir al proyecto.

```
<script type="text/javascript" src="scripts.js"></script>
```

3.3.3.3. Javascript solo es una mejora.

se puede usar JavaScript como una mejora, se debe pensar que no siempre se puede disponer de él y que agentes externos pueden deshabilitarlo sin opción a activarlo, así que se debe programar pensando en que cualquier persona en este problema también debe poder ver la página web.

Para ello se puede ver un ejemplo de JavaScript en el que se ve como válida un formulario.

HTML:

```
<form action="index.php" onsubmit="return checkform(this)"></form>
<p><label for="login">Login:</label>
<input type="text" name="login" id="login" /></p>
<p><label for="pw">Password:</label>
<input type="password" name="pw" id="pw" /></p>
<p><input type="submit" value="send" /></p>
</form>
```

JavaScript:

```
function checkform(f)
```

```
{
```

```

var error='';

error+=f.login.value=='?'?'nlogin':';

error+=f.pw.value=='?'?'npassword':';

if (error!='')

{

alert('Por favor ingrese lo siguiente:'+error);

}

return error=='';

```

Esto es perfectamente válido, permite controlar el formulario sin perjudicar a los usuarios que no tienen JavaScript. De modo que en caso de no tenerlo activado, únicamente se tiene que tener en cuenta que en la página de recepción de datos tendrá que validar directamente en el servidor, devolviendo al usuario a la página anterior en caso de que no sean válidos.

Otra forma de resolver el problema sería la siguiente.

HTML:

```

<form action="index.php">

<p><label for="login">Login:</label>

<input type="text" name="login" id="login" /></p>

<p><label for="pw">Password:</label>

<input type="password" name="pw" id="pw" /></p>

<p><input type="button" onclick="checkform()" value="send" /></p>

```

```
</form>
```

JavaScript:

```
function checkform()
{
var f=document.forms[0];

var error='';

error+=f.login.value=='?'?'nlogin? ':'';
error+=f.pw.value=='?'?'npassword? ':'';

if (error!='')
{
alert('Por favor ingrese lo siguiente:'+error);
} else {
f.submit();
}
}
```

3.3.3.4. Se comprueba la disponibilidad de un objeto.

Muchos de los errores en JavaScript se deben a que se intenta acceder a elementos o métodos que no existen en un determinado momento del periodo de ejecución. Para solucionar esto basta con realizar una comprobación antes de utilizar dicho elemento o método.

Antes de usar el elemento o comprobamos que existe, así conseguimos una robustez en la aplicación que el usuario agradecerá. Esta técnica es muy usada para el famoso cross browsing.

3.3.3.5. Crea Un Javascript No Especifico Para Un Navegador

Quizás la regla más complicada de todas, pero la más importante. Al igual que tenemos que tener en cuenta navegadores sin JavaScript se tiene en cuenta todos los navegadores, para ello se usa la regla.

3.3.3.6. No usar variables de otros scripts.

Cuando se crea una función o funcionalidad debemos estar seguros de usar variables locales para dicha función o funcionalidad. De esta forma nos prevenimos que otras funciones o funcionalidades modifiquen nuestras variables.

3.3.3.7. Efectos De Ratón De Forma Independiente

Tener en cuenta que el JavaScript esté activo o no, no es suficiente. Se tiene en cuenta que hay usuarios que no pueden utilizar el ratón para navegar como normalmente se hace, esto obliga a usar el teclado para realizar todas las tareas que requiera la aplicación. Un problema típico es el onchange o onblur en los elementos de un formulario ya que el teclado ejecuta estos eventos de forma distinta que el ratón.

3.4. PRUEBAS

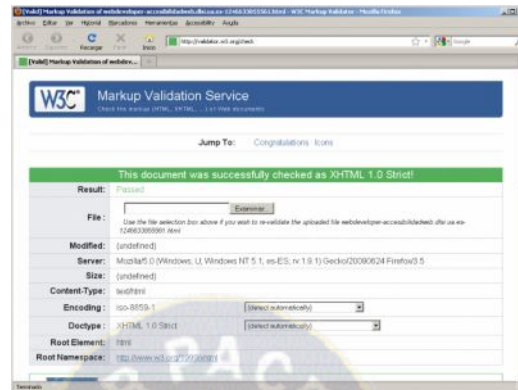
Para ver si la página es accesible se tiene varias páginas para hacer las pruebas correspondientes y así se sabrá si nuestra página es accesible.

En la página Complete List of Web Accessibility Evaluation Tools, el W3C proporciona una lista muy extensa de herramientas de evaluación de la accesibilidad.

Además, en este sitio web está el apartado Herramientas de revisión donde se enumeran algunas de las herramientas de revisión más importantes y más útiles.

En primer lugar hay que revisar la sintaxis y semántica de los lenguajes empleados en la creación de las páginas web. Para revisar y validar el lenguaje XHTML se puede emplear la página Markup Validation Service de W3C. En la siguiente imagen se puede ver la respuesta obtenida para valorar la página:

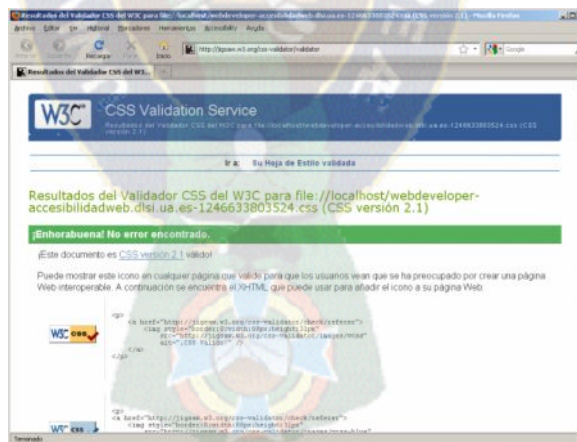
Figura 30 página de comprobación de accesibilidad W3C



Fuente: validator.w3.org

Para revisar y validar el lenguaje CSS se puede emplear la página CSS Validation Service también de W3C. En la siguiente imagen se puede ver la respuesta obtenida al validar esta página:

Figura 31 Página de validación CSS



Fuente: validator.w3.org

Para revisar los puntos de comprobación y las pautas de WCAG 1.0 no existe una herramienta automática que pueda sustituir la revisión de un experto humano. Sin embargo, existen algunas herramientas que pueden ayudar al proceso de revisión, principalmente con aquellos puntos de verificación que sólo supongan la existencia de

ciertos atributos o el correcto uso de algunas etiquetas de XHTML. Existen infinidad de herramientas de este tipo, pero una de las más conocidas y empleadas es TAW, que sólo requiere la introducción de la URL de la página que se quiere validar, tal como se puede ver en la siguiente imagen:

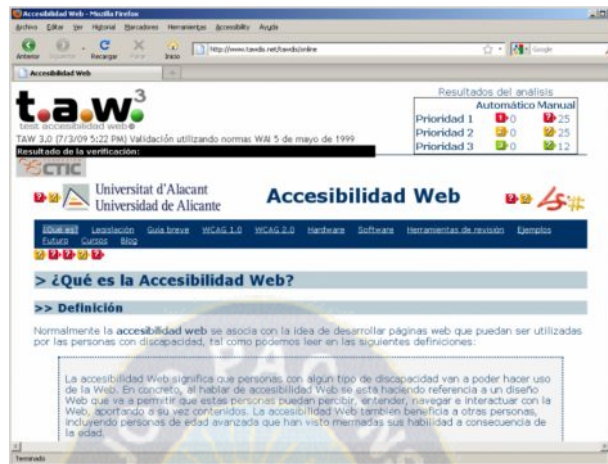
Figura 32 Página de comprobación de accesibilidad



Fuente : www.tawdis.net

En la siguiente imagen se puede observar el resultado de revisar con TAW esta página web, como se puede observar en la esquina superior derecha, la página no contiene ningún error automático de nivel A, AA o AAA explicados en el capítulo 2, pero eso no asegura que esté libre de errores, ya que se indican algunos posibles errores que requieren una revisión manual por un experto:

Figura 33 página de comprobación de accesibilidad “TAW”



Fuente: www.tawdis.net

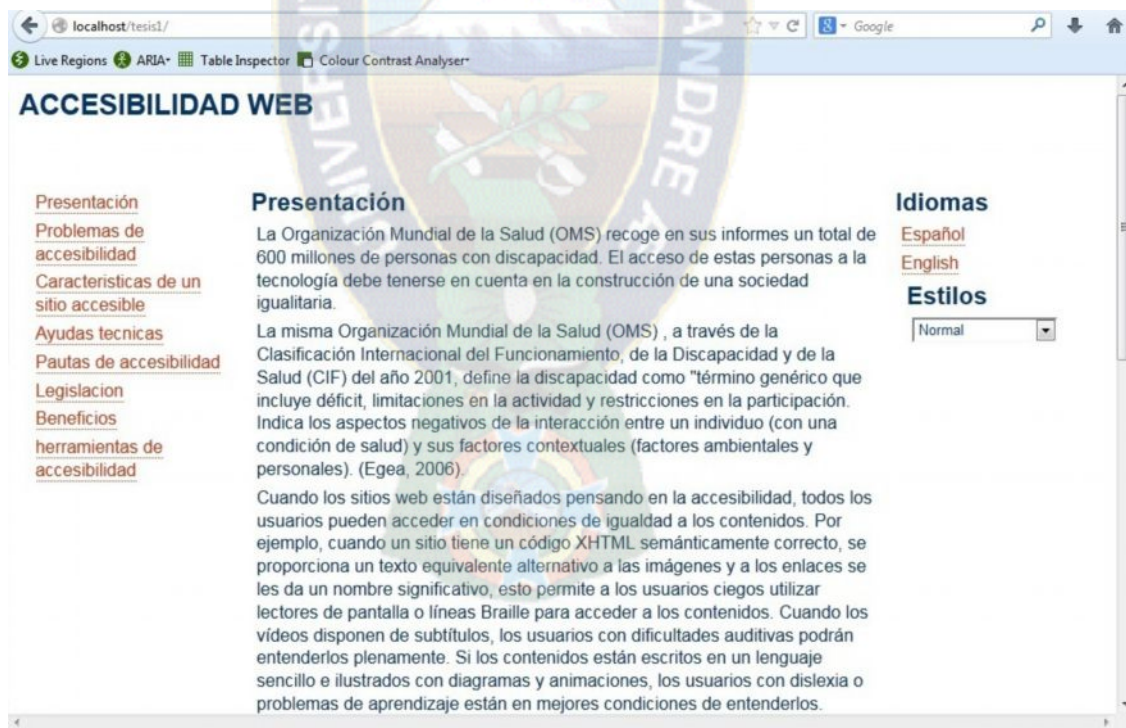
CAPITULO 4

Para comprobar la hipótesis del presente trabajo de investigación se aplicó el método de Chi-cuadrado.

Se realizó una encuesta de una página web aplicando la metodología propuesta y puesta a prueba mediante el cual se comprobó el nivel de aceptación de personas que usan el programa JAWS del INSTITUTO BOLIVIANO DE LA CEGUERA.

El prototipo está hecho según la metodología propuesta destinado a hacer conocer sobre que es la accesibilidad web.

Figura 34 toma de pantalla de prototipo



Fuente: propia

A continuación se realiza un análisis de los resultados obtenidos a personas que accedieron a la página web.

4.1. DESCRIPCION DE LA POBLACION

Para la comparación de la variable se eligió 2 grupos de personas el grupo numero 1 está conformado por personas que tienen un excelente manejo del programa JAWS y el grupo numero 2 está conformado por personas que tienen un manejo regular y principiante del mismo programa se hizo esta clasificación por que las personas que tienen mayor experiencia ya saben sobre los impedimentos o barreras que existen en una página web mientras que el segundo grupo descubre impedimentos o barreras.

Para la comprobación de la hipótesis se considera un margen de error del 0.05% en tablas independientes para cada grupo. Aplicando la siguiente pregunta en la encuesta.

¿EL ACCESO A LA INFORMACION DE LA PAGINA WEB LE PARECE QUE ES?

Con los siguientes resultados:

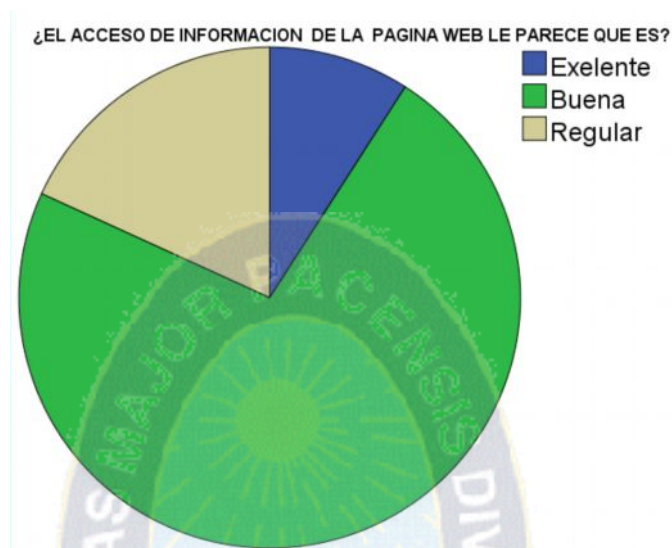
Tabla N° 2 Resultados de la muestra poblacional de personas grupo 1.

Tabla 2 Resultados de la muestra poblacional grupo 1

¿el acceso a la información le parece que es ?

		Frecuencia	Porcentaje	Porcentaje	total
Válidos	Excelente	1	4,0	9,1	9,1
	Buena	8	32,0	72,7	81,8
	Regular	2	8,0	18,2	100,0
	Total	11	44,0	100,0	
Perdidos	Sistema	14	56,0		
Total	25	100,0			

Ilustración 1 Grafico circular de personas con manejo excelente de JAWS



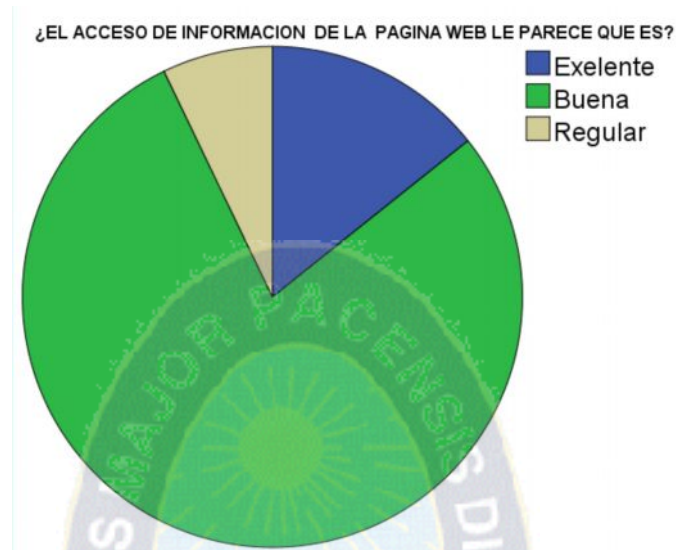
Fuente: propia

Tabla 3 Resultados de la muestra poblacional grupo 2

¿el acceso a la información le parece que es ?

	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado	
Válidos	Excelente	2	8,0	14,3	14,3
	Buena	11	44,0	78,6	92,9
	Regular	1	4,0	7,1	100,0
	Total	14	56,0	100,0	
Perdidos	Sistema	11	44,0		
Total	25	100,0			

Ilustración 2 Grafico circular de personas con manejo regular o principiante de JAWS



4.2. PRUEBAS DE CHI CUADRADA

Las pruebas serán presentadas en los siguientes cuadros con sus respectivas categorías pero ahora incluyendo las columnas de números observado, número esperado y la diferencia tan bien llamado residual, estas tres columnas serán probadas por estadístico de prueba Chi –cuadrado, el programa usado es el SPSS 19, en cuanto para los fines consiguientes se presentara la siguiente información.

4.2.1. Planteamiento De La Hipótesis

Ho= no existe cambios significativo con la metodología

La aplicación del desarrollo de una metodología web no facilita el acceso a contenido web para personas con discapacidad visual.

H1 = Existe cambios significativos con la metodología

La aplicación del desarrollo de una metodología web facilita el acceso a contenido web para personas con discapacidad visual.

4.2.2. Región Crítica

La región crítica por el valor que mostrara el punto de inflexión para determinar el rechazo o la aceptación de la hipótesis nula. Entonces el chi-cuadrado de tablas a un nivel de significación del 95% y con :

$$GL=[J-1][K-1]$$

Que es $(2-1)*(4-1)=3$, entonces será con 3 grados de libertad. Lo que nos da como punto de inflexión un resultado de tablas de 0.35.

4.2.3. Estadístico De Prueba

Como se puede observar el Chi-cuadrado calculado no se encuentra en la región de rechazo de H_0 .

Tabla 4 resultados generados por SPSS

Resumen del procesamiento de los casos

	Casos					
	Válidos		Perdidos		Total	
	N	Porcentaje	N	Porcentaje	N	Porcentaje
¿Uso de JAWS? * ¿EL ACCESO DE INFORMACION DE LA PAGINA WEB LE PARECE QUE ES?	25	100,0%	0	,0%	25	100,0%

Fuente: propia

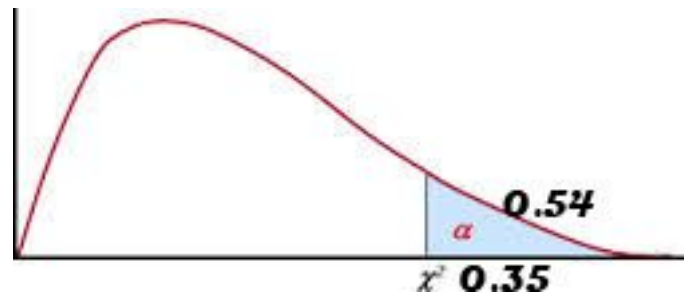
Tabla 5 resultados generados por SPSS
¿El acceso a la información le parece que es ?

		Excelente	Buena	Regular	Total
¿ uso de JAWS?	Excelente	1	8	2	11
	Principiante	3	10	1	14
Total		4	18	3	25

Tabla 6 Pruebas generadas por SPSS

	Valor	gl	Sig. asintótica (bilateral)
Chi-cuadrado de Pearson	1,213	2	,545
Razón de verosimilitudes	1,248	2	,536
Asociación lineal por lineal	1,161	1	,281
N de casos válidos	25		

Ilustración 3 Región de rechazo



Fuente: propia

$$0.54 > 0.325$$

El resultado que da SPSS es de 0.545 que es mayor a 0.35 así que se puede descartar la hipótesis nula y se afirma hipótesis 1 ya que se encuentra en la región de rechazo.

4.2.4. Decisión

Como la prueba de Chi-cuadrada es mayor que el punto de inflexión entonces se rechaza H_0 (hipótesis nula) puesto que el valor de chi-cuadrado pertenece a la zona de rechazo por lo siguiente la hipótesis alternativa es considerada como válida reflejando que la utilización de **la aplicación del desarrollo de una metodología web facilita el acceso a contenido web para personas con discapacidad visual.**

5 CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

Después de desarrollar la metodología de desarrollo web asegurando accesibilidad a personas con discapacidad visual se ha llegado a las siguientes conclusiones:

- El primer objetivo específico “Integrar al análisis de requerimientos todas las posibles barreras de percepción de una persona con problemas visuales.” Este objetivo fue concluido ya que en la metodología propuesta en la fase de análisis se ve todas las barreras que puedan existir.
- El segundo objetivo “Integrar en la fase de código elementos para que puedan facilitar la percepción por programas de asistencia a personas con discapacidad visual” por la investigación de la metodología se pudo ver los impedimentos de la no inclusión del atributo “ALT” y del java script intrusivo así dando recomendaciones de cómo hacer un buen código.
- El tercer objetivo “Tomar en cuenta todos los posibles problemas que puedan limitar la percepción” a que se refiere este estudio con la percepción es como una persona con discapacidad visual ve una página web y herramientas las cuales utilizan para acceder a la información de una página web como percibe una persona con discapacidad visual, En entrevistas con el instituto nacional de la ceguera se ve más a fondo este problema ya que la paginas web bolivianas son inaccesibles en todos los aspectos.
- El cuarto objetivo “Integrar todas las pruebas de accesibilidad dadas por la WAI [Web Accessibility Initiative]” En el estudio de esta metodología se encuentran páginas que se dedican exclusivamente a ver si una web es accesible.
- El quinto objetivo “Diseñar un sitio web que sea accesible para las personas ciegas y con discapacidad visual bien estructurado y accesible” Se hace una página web hablando sobre que es accesibilidad web para en instituto nacional de la ceguera

aplicando todo lo mencionado en la metodología y es bien aceptado ya que en dicen que se puede acceder a toda la información y es muy navegable el sitio.

- El sexto objetivo y ultimo “Acortar la brecha digital que pueda tener una persona con discapacidad visual” Aplicando la metodología propuesta se verá que se puede acceder a toda la información que puede proveer una página web así no se pierde ninguna información.

5.2. CONCLUSIÓN DE LA HIPÓTESIS

Al obtener los resultados de la encuesta realizada después de la prueba de la página web prototipo y realizando los análisis correspondientes con ayuda del método estadístico chi cuadrada se demostró la hipótesis que fue rechazada por lo que se afirma que la hipótesis propuesta “La aplicación del desarrollo de una metodología web facilita el acceso a contenido web para personas con discapacidad visual” Fue aceptada.

5.3. RECOMENDACIONES

Si bien esta metodología agrega los aspectos más importantes para que una página web sea accesible se recomienda tomar en cuentas estos puntos tan bien, no están incluidos pero no son menos significativo también se puede visitar la página de la WAI que da instrucciones para llegar a la accesibilidad:

5.3.1. UTILIZACION DE FLASH

Si se utiliza Flash de forma no accesible se deberá proporcionar una alternativa accesible en HTML y entonces en la declaración de conformidad no incluiríamos javascript como tecnología de la que se depende.

Si en una página se usa una tecnología que no es compatible con la accesibilidad o bien una que sí lo es pero usada de forma no compatible, no debe impedir a los usuarios acceder al contenido del resto de la página. Además, es necesario que la página web como un todo siga cumpliendo con los requisitos de conformidad cuando cualquier

tecnología de la que no se depende está activada en los agentes de usuario así como cuando esté desactivada o no se soporte.

5.3.2. TEXTO JUSTIFICADO O ALINEADO A LA IZQUIERDA

Es decir, el texto no debe justificarse, puesto que algunas personas con discapacidades cognitivas o visuales pueden tener problemas para leer el texto que está justificado a ambos lados. Esta dificultad proviene de que las variaciones entre el espaciado entre las palabras puede provocar que las palabras estén demasiado juntas o por el contrario "ríos blancos", es decir, zonas del texto donde la separación de las palabras dejan grandes zonas continuas de espacio en blanco.

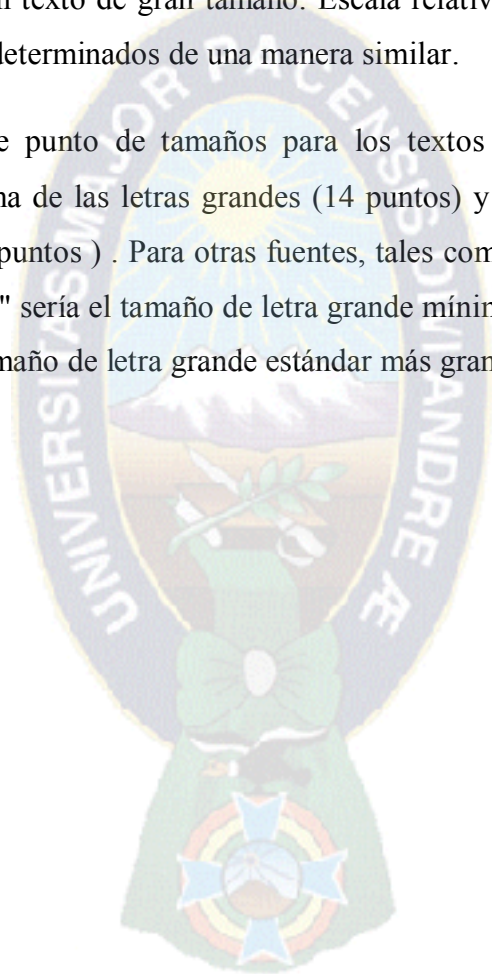
5.3.3. TAMAÑO DE TEXTO

En algunos criterios de conformidad de las WCAG 2.0 (1.4.3, 1.4.6) se establecen diferentes obligaciones en función de si el tamaño de texto es grande o pequeño.

- Las fuentes con trazos extraordinariamente delgadas o características inusuales y características que reducen la familiaridad de sus formas de las letras son más difíciles de leer, especialmente en los niveles más bajos de contraste.
- El tamaño de fuente es el tamaño cuando se entrega el contenido. No incluye el cambio de tamaño que puede ser realizado por un usuario .
- El tamaño real del carácter que un usuario ve depende tanto del tamaño autor - definido y la pantalla del usuario o ajustes de agente de usuario . Para muchas fuentes de texto del cuerpo de corriente, 14 y 18 puntos es aproximadamente equivalente a 1,2 y 1,5 em o al 120 % o 150 % del tamaño del defecto para el texto del cuerpo (suponiendo que la fuente del cuerpo es el 100 %) , pero los autores tendrían que comprobar esto para las fuentes particulares de uso. Cuando las fuentes se han definido en unidades relativas, el tamaño de punto real se calcula por el agente de usuario para su visualización. El tamaño en puntos se debe obtener de la aplicación del usuario, o se calcula sobre la base de medidas de fuente como el

agente de usuario lo hace , al evaluar este criterio de éxito . Los usuarios que tienen baja visión serían responsables de elegir la configuración adecuada.

- Cuando se utiliza texto sin especificar el tamaño de fuente, el tamaño de letra más pequeño utilizado en los principales navegadores de texto no especificado sería un tamaño razonable suponer para la fuente . Si un título de nivel 1 se representa en 14pt negrita o superior en los principales navegadores, entonces sería razonable asumir que es un texto de gran tamaño. Escala relativa puede calcularse a partir de los tamaños predeterminados de una manera similar.
- Los 18 y 14 de punto de tamaños para los textos romanos son tomadas de la superficie mínima de las letras grandes (14 puntos) y el tamaño de fuente estándar más grande (18 puntos) . Para otras fuentes, tales como idiomas Japonés, el tamaño de "equivalentes" sería el tamaño de letra grande mínimo utilizado para esos idiomas y el siguiente tamaño de letra grande estándar más grande.



BIBLIOGRAFIA

[**AboutW3**] (2009) Ian Jacobs “About the World Wide Web Consortium (W3C)”
<http://www.w3.org/Consortium/>.

[**Brewer, 2004**] J. Brewer. “Web Accessibility Highlights and Trends”, 2004.

[**Chisholm & Henry, 2005**] W. Chisholm, S.L. Henry. “Interdependent Components of Web Accessibility”.

[**Clarck, 2003**] J. Clarck. “Building Accessible Web Sites”. 2003. New Riders.

[**Evaluating Accesibility**] (2009) W3C-WAI “Evaluating Web Sites for Accessibility:

[**Kelly et al, 2007**] B. Kelly, L. Neville, E.A. Draffan, S. Fanou. “One World, One Web But Great Diversity”. Located with the 17th International World Wide Web Conference. Copyright 2008

[**Paciello, 2002**] M. Paciello. “Web Accessibility for People with Disabilities”. CMP Books.

[**Sloan et al, 2006**] D. Sloan, B. Kelly, A. Heath, H. Patrie, F. Hamilton, L. Phipps. “Contextual Web Accessibility - Maximizing the benefit of Accessibility Guidelines”.

[**Testimonials**] (2002) W3C-WAI. “Testimonials for User Agent Accessibility Guidelines (UAAG) 1.0 Recommendation”.

[**Thatcher et al, 2006**] “Web Accessibility: Web standards and regulatory compliance”. 2006. Friendsoft (UK).

[**Terán, 2004**]. **M.sc.** Teran “Desarrollo de sitios web accesibles para personas con discapacidad”

[**Pressman, 98**] 1998 Roger S. Pressman “Ingenieria de software”

[**Sloan et al, 2006**] D. Sloan, B. Kelly, A. Heath, H. Patrie, F. Hamilton, L. Phipps. “Contextual Web Accessibility - Maximizing the benefit of Accessibility Guidelines”.

<http://www.w3.org/2002/12/uaag10-testimonials> .

[WAIabout] (2009) W3C-WAI. “About WAI”. <http://www.w3.org/WAI/>

[WCAG 1.0] (2009) W3C-WAI. “Authoring Tool Accessibility Guidelines (ATAG) 2.0 W3C Working Draft 21 May 2009”.<http://www.w3.org/TR/ATAG20/>.

[UAAG 2.0 draft, 2009] W3C-WAI. “User Agent Accessibility Guidelines (UAAG) 2.0 W3C Working Draft 11 March 2009”.

<http://www.w3.org/TR/2009/WD-UAAG20-20090311/>.

[WAI social factors] (2009) S.L. Henry & Andrew Arch. “Social Factors in Developing a Web Accessibility Business Case for Your Organization”.
<http://www.w3.org/WAI/bcase>

[WCAG 2.0, 2009] W3C-WAI. “Web Content Accessibility Guidelines (WCAG) 2.0 W3C Recommendation 11 December 2008”. <http://www.w3.org/TR/WCAG20/>.

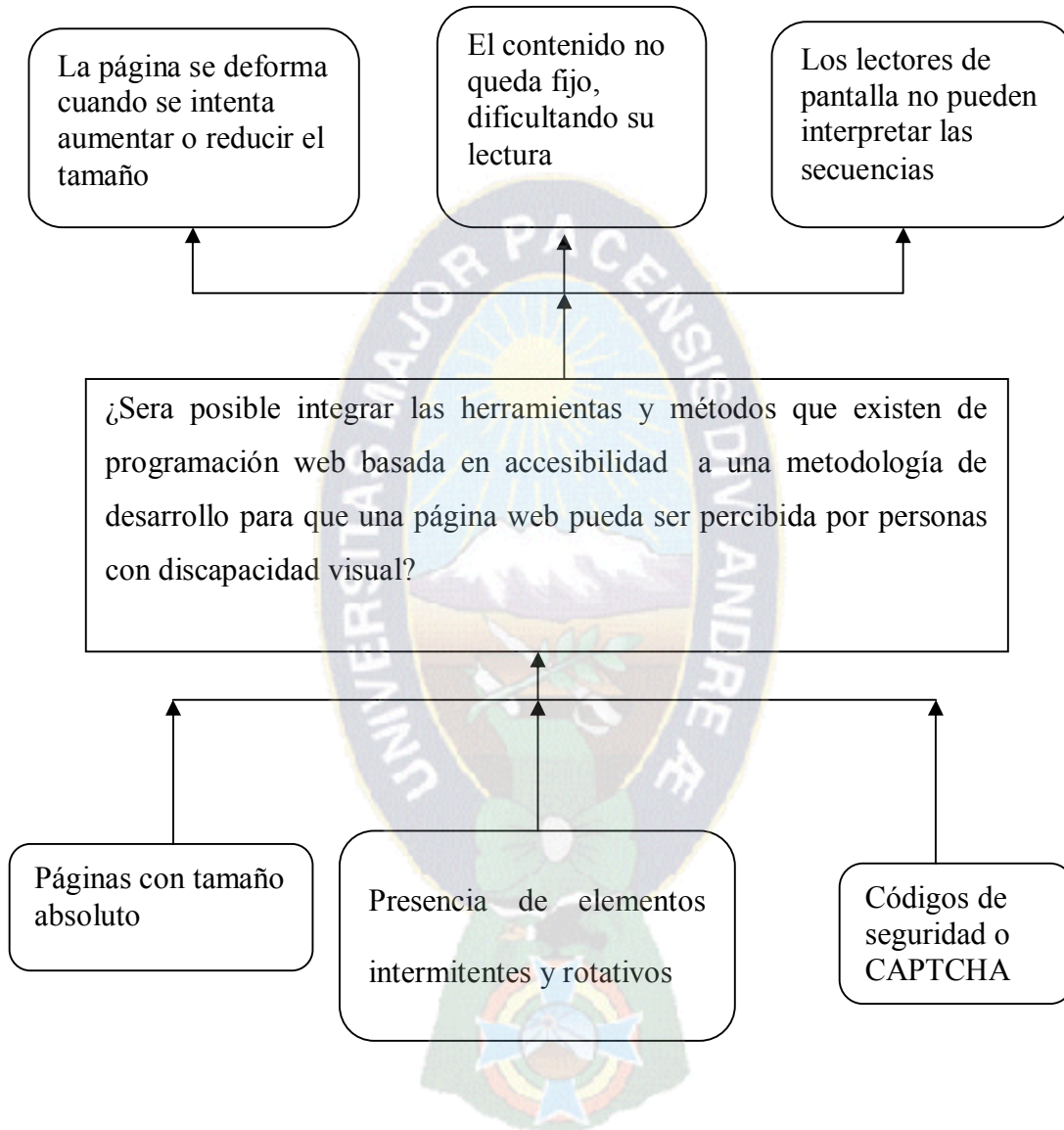
[WAI interdependences, 2005] S. L. Henry. “Essential Components of Web Accessibility”. <http://www.w3.org/WAI/intro/components.php>.

[Wikipedia Section 508](2009) Wikipedia. “Section 508 Amendment to the Rehabilitation Act of 1973”.

http://en.wikipedia.org/wiki/Section_508_Amendment_to_the_Rehabilitation_Act_of_1973.

[Wiscosin] (2001) University of Wiscosin. “Disability as a function of Age”.
<http://trace.wisc.edu/docs/function-aging/>

ANEXOS A: ARBOL DE PROBLEMAS



ANEXO B: ARBOL DE OBJETIVOS

