



ASOCIACIÓN DE INVESTIGACIÓN
EN SOFTWARE INTELIGENTE

USABILIDAD Y ACCESIBILIDAD DE APLICACIONES WEB UTILIZANDO UWE



INGENIERÍA WEB

Mayo 2007
La Paz - Bolivia

1. INTRODUCCIÓN	1
2. USABILIDAD EN LA WEB	1
2.1. Definición	2
2.2. Problemas de usabilidad	3
2.3. Características de la usabilidad	5
2.4. Guías de diseño	6
2.4.1. Tiempo de respuesta	6
2.4.2. Disposición del texto	7
2.5. Diseño de página	11
2.6. Estructura de la navegación	15
3. ACCESIBILIDAD EN LA WEB	17
3.1. Definición	18
3.2. Pautas de accesibilidad	19
3.3. Pautas de accesibilidad al contenido Web 1.0 (WCAG 1.0)	19
3.3.1. Temas de diseño accesible	20
3.4. Diseño accesible	27
4. MÉTRICAS PARA LA USABILIDAD Y ACCESIBILIDAD EN LA WEB	29
4.1. Métricas principales de usabilidad	30
4.2. Métricas principales de accesibilidad	32
5. INTEGRACIÓN DE CRITERIOS DE USABILIDAD Y ACCESIBILIDAD	34
5.1. Análisis de requerimientos	35
5.2. Diseño de la aplicación	36
5.2.1. Modelo conceptual	36
5.2.2. Modelo de navegación	36
5.2.3. Diseño de interfase y modelo de presentación	37
BIBLIOGRAFÍA	41

1. INTRODUCCIÓN

El desarrollo de aplicaciones Web va evolucionando día a día, cada vez más métodos y herramientas se desarrollan para construir aplicaciones funcionales. Sin embargo, existen aspectos todavía descuidados. Uno de estos aspectos es la calidad de aplicaciones Web, ya que, los desarrolladores se inclinan por priorizar la funcionalidad y el diseño y olvidar factores que en determinado momento llegan a influir en el éxito o fracaso de las aplicaciones. Los criterios de calidad son variados y cada uno tiene su grado de importancia. El presente trabajo se enmarca en el estudio y aplicación de la *usabilidad* y la *accesibilidad*, como criterios fundamentales para la construcción de aplicaciones Web de calidad.

La usabilidad, en palabras simples, se puede entender como la facilidad de uso y de comprensión de una Aplicación Web. Esta definición resumida no manifiesta el verdadero significado de usabilidad y todas sus implicaciones. La usabilidad contempla un espectro mucho más amplio y se basa en varios estudios que reflejan la preocupación por que los usuarios comprendan las aplicaciones desarrolladas, las utilicen de manera correcta, y quizás lo más importante, sientan la necesidad de utilizar estas aplicaciones habitualmente. Todo esto con el fin de que la información publicada sea siempre recibida y con ello se obtengan más beneficios, ya sean comerciales, académicos, etc.

Facilidad de uso también significa acceder a la información con el menor grado de dificultad posible. Se podría pensar que con la Web la información ya no significa un problema de acceso, a menos que existan dificultades técnicas o aspectos relacionados con seguridad. Aún en situaciones normales esto podría debatirse, pero cuando un usuario vive con algún tipo de discapacidad, la accesibilidad es un tema que debe estudiarse con más profundidad. Entonces, la accesibilidad podría entenderse como el grado de facilidad con que cualquier usuario puede acceder a la información.

Para ambos criterios de calidad existen guías, las cuales están basadas en estudios sobre la interacción de los usuarios con las aplicaciones Web. La interfaz de usuario es el punto focal de esta investigación, puesto que es ahí donde empieza la interacción de los usuarios con las aplicaciones Web, y es ahí donde se adquiere o se pierde la confianza del público. A diferencia de lo que generalmente se piensa, la confianza no se logra, tanto con un diseño atractivo como con un diseño usable y accesible. De hecho las consignas de estos criterios son: “menos es más”, “simple es completo” y “práctico es útil”. Entonces, se plantea el trabajo de conocer las guías para hacer aplicaciones usables y accesibles, y comprobar su efectividad.

2. USABILIDAD EN LA WEB

El crecimiento de usuarios de Web ha permitido conocer diferentes criterios de los usuarios, que de acuerdo a las tareas u objetivos que pretenden realizar, ellos evalúan el uso de las Aplicaciones Web para posteriormente establecer una elección rigurosa de los sitios que visitaran. Es normal que a los usuarios prefieren un uso fácil de la aplicación.

La elección que los usuarios establecen al visitar diferentes sitios, viene determinada por factores de calidad en las Aplicaciones Web, cuya calidad implica: establecer un proceso de desarrollo, utilizar una metodología, herramientas y técnicas en el desarrollo de las Aplicaciones; pero en este desarrollo se debe considerar esencialmente los contextos de uso y las necesidades de los usuarios [Kappel et al., 2003].

En el desarrollo de una Aplicación Web de calidad se debe priorizar técnicas de Usabilidad para tomar en cuenta los diferentes usuarios que van a utilizar la aplicación, sin dificultad o confusión alguna, ya que los desarrolladores deben conocer el comportamiento de los usuarios frente al computador, esto significa saber que pasos van a tomar en la navegación del sitio, y asegurar que en cada paso el usuario sabe donde se encuentra y que pueda acceder fácilmente a la información que necesita. Mantener dicho lineamiento genera que los usuarios volverán al sitio con diferentes fines, que es la idea fundamental de toda Empresa representada por una aplicación.

“La usabilidad ha asumido una parte importante de la economía en Internet, mucho más que en el pasado” [Nielsen, 2000].

2.1. Definición

Usabilidad es un término adaptado de la palabra en inglés *“usability”*, que se puede interpretar como facilidad de uso, este factor recientemente ha causado un gran impacto en el desarrollo de las Aplicaciones Web; ya que, en el pasado, muchas veces se confundía el objetivo de dichas Aplicaciones.

La Organización Internacional para la Estandarización define a la usabilidad como el alcance en la capacidad de un producto para ser utilizado, por usuarios específicos, para alcanzar metas específicas como la efectividad, la eficiencia y la satisfacción, en un contexto específico de uso [ISO 9241-11, 1998].

El Instituto Electrónico e Ingenieros Electrónicos (IEEE) define la usabilidad como la facilidad con la que un usuario puede aprender a operar, realizar entradas e interpretar salidas de un sistema.

La definición que presenta ISO, describe el aspecto más importante que es considerado en el diseño del software utilizable por basarse en los usuarios, las metas del producto y el contexto de uso. El carácter sistemático de la definición sobre los factores que le acompañan no son mecanismos suficientes a considerar como roles generales que se puedan señalar en las Aplicaciones Web usables. Lo que se debe hacer es que la definición ISO/IEC vaya cubriendo usuarios, el contexto de usuarios en cada caso individual, y el control del desarrollo de procesos con mecanismos apropiados de regeneración [Keppel et al., 2003].

*Tabla 1 Modelo de Usabilidad acordado por Nielsen y Shneiderman
Fuente: [Kappel, 2003]*

ISO/IEC 9241-11	Nielsen (1994)	Shneiderman y Plaisant (2005)
Eficiencia	Eficiencia Aprendizaje Compresión	Velocidad de ejecución Tiempo de instrucción Retención sobre un tiempo
Efectividad	Errores/Seguridad	Grado de errores por usuario
Satisfacción	Satisfacción	Satisfacción subjetiva

Para acercarse a los tres criterios de efectividad, eficiencia y satisfacción, Nielsen (1994) conjuntamente con Shneiderman y Plaisant (2005) identifican métricas para la evaluación concreta de cada uno de los criterios de ISO representada en la *Tabla 1*. Sin embargo, Welie et al. (1999) describe mecanismos adicionales que pueden influenciar positivamente a estas métricas (*Figura 1*).

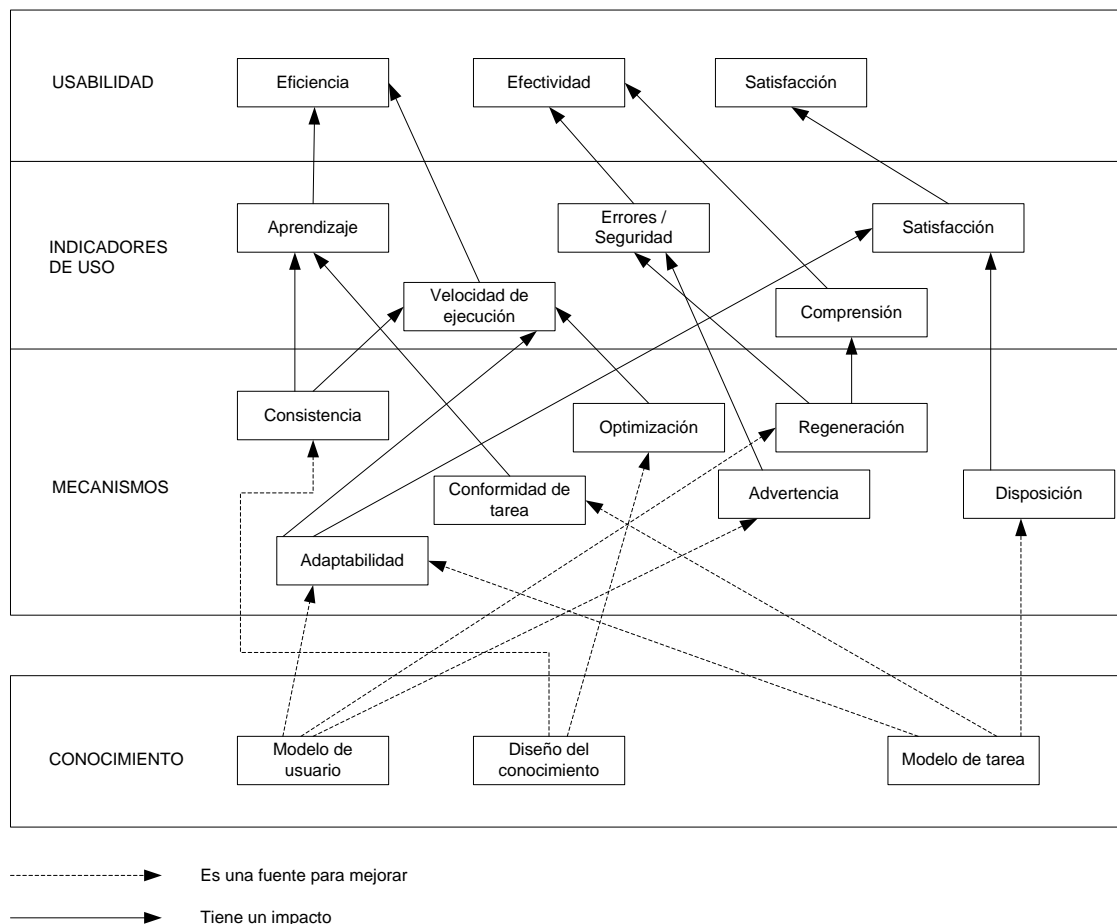


Figura 1: Modelo de las capas de Usabilidad acordado por Welie et al.
Fuente: [Kappel G., 2003]

Así la usabilidad de las Aplicaciones Web llega a ser un factor de calidad esencial que se basa en los usuarios, contextos de usuario y desarrollo de procesos, determinando así el grado de utilidad que los usuarios le dan, quienes se apoyan en la eficiencia, efectividad y satisfacción de la aplicación. El construir una Aplicación Web utilizable implica tomar en cuenta técnicas de usabilidad durante todo el proceso de desarrollo, dichas técnicas deben ser aplicadas en el diseño de página, sitio, navegación y contenido, formando así la estructura general de la aplicación.

2.2. Problemas de usabilidad

El uso que le dan los usuarios a un sitio Web puede llegar a tener dramáticas consecuencias en la elección futura de sitios para navegar, pues ellos evaluarán las fallas o el éxito del sitio de acuerdo a diferentes factores.

Treinta y cuatro problemas de usabilidad se identificaron inicialmente de acuerdo a un estudio sistemático¹ ejecutado desde 1994, de los cuales ocho problemas han constituido ser más que un problema y no han cambiado, los cuales se presentan a continuación [Nielsen y Loranger, 2006]:

¹ Se consideran ventanas emergentes a todas aquellas páginas que “aparecen” sin la solicitud expresa del usuario. Generalmente contienen información publicitaria.

a. Links que no cambian de color cuando son visitadas

Los usuarios necesitan entender donde han estado, para entender donde están, y a donde podrían ir. Los tres se interrelacionan debido a que: tener conocimiento del pasado de la navegación ayuda a entender la situación actual en la navegación. Conociendo la situación pasada y presente, a la vez, hace más fácil la elección de donde ir luego.

b. Ruptura del botón atrás para retroceder

En la Web las personas deben saber que pueden retroceder en la navegación o pueden alterar sus acciones sin dificultad alguna mediante el botón *Atrás*, pero algunos programadores deshabilitan el botón con diferentes fines. Los usuarios deben tener la facilidad de explorar un sitio libremente, manteniendo la confianza de que pueden salir de cualquier problema que hallen, ya que es un principio fundamental de la interacción humano-computadora.

c. Apertura de nuevas ventanas en el navegador

Los usuarios de la Web cuando pulsan un link o un botón de alguna página, donde normalmente se espera que dicho enlace reemplace la última página que visitaron; resulta que simplemente aparece una nueva ventana. Desde el punto de vista de los usuarios, es incomodo tener que obtener una nueva ventana porque no les permitirá volver atrás si así lo quisieran, para no llenarse de ventanas y no romper la navegación libre.

d. Ventanas emergentes

Las ventanas emergentes² distraen y perjudican la captura de información que ellos pretenden obtener de la página principal en donde se encuentran. Los usuarios al verse incómodos por dicha situación, instalan algún software para bloquear las ventanas emergentes, que también resulta molesto.

e. Diseño de elementos que parecen publicidad

Los usuarios, al navegar en la Web buscan la información que requieren, ignorando otros elementos que a simple vista otros desean imponerles. Los elementos de publicidad desvían la atención de los usuarios y los desvían de cumplir sus metas, motivo por el cual los usuarios crean un sistema de autodefensa de dichos elementos y simplemente no los leen.

f. Violación de la convenciones del diseño de paginas Web

Este problema implica que los usuarios de la Web generalmente utilizan la experiencia, al conocer diferentes aplicaciones Web, para mantener las normas y convenciones que se estandarizan en dichas aplicaciones. Entonces es frecuente que se busquen similitudes en la presentación y uso que se les quiera dar.

² Se consideran ventanas emergentes a todas aquellas páginas que “aparecen” sin la solicitud expresa del usuario. Generalmente contienen información publicitaria.

g. Contenido sutil y propaganda exagerada sin utilidad.

Uno de los grandes problemas es que las empresas al posesionarse en la Web no establecen lo que hacen en un lenguaje claro para el usuario. Los usuarios tienen poca paciencia por el tiempo que disponen y cuando se les presenta una descripción muy florida e información escasa tienden a abandonar la aplicación e irse a otro lado. Presentar un contenido sutil y pobre en un lenguaje confuso, no sólo hace que se malinterprete el objetivo real de la empresa, además provoca que sus productos no se vendan. El objetivo de las empresas que poseen aplicaciones Web debería ser vender los beneficios, no sus características.

h. Contenido denso y texto poco escaneable³

Los bloques densos de texto son rechazados por los usuarios de la Web. Este tipo de páginas le sugiere al usuario que trabajará mucho para obtener la información que necesita. Además los textos densos muchas veces están escritos en términos burocráticos y técnicos especializados en un área, que no son de conocimiento general para los diferentes usuarios.

En base a los problemas que frecuentemente tienen las aplicaciones Web, se describen tres razones por las que un problema de usabilidad puede, en realidad, no ser un problema, tales razones son [Nielsen y Loranger, 2006]:

- *La tecnología ha mejorado.* La evolución de la usabilidad en aplicaciones Web se debe a que la tecnología avanzada ha aliviado algunas de las dificultades que los usuarios han tenido.
- *Los usuarios priorizan el diseño de la Aplicación Web.* Cuando las personas se acostumbran al conocer el diseño, están menos desconcertados en las tareas que tengan que realizar.
- *Los diseñadores limitan las malas costumbres.* En mayor cantidad se ha mostrado que los diseñadores van entendiendo el diseño particular, utilizando técnicas para reforzar los problemas, por lo cual generan menos problemas.

2.3. Características de la usabilidad

En la Web el manejo de la usabilidad también es particular debido a que se trata de un entorno con muchas posibilidades de comercialización e intercambio de ideas y productos. Si bien, las aplicaciones Web son en sí software, la forma en que se utilizan y como se muestran tienen una dimensión totalmente nueva y diferente. Manhartberger y Musil (2002), citado en Hitz et al. (2003), resumen algunas de las diferencias más notables:

a. El software ya no se vende en cajas.

Las aplicaciones Web no vienen en cajas ostentosas y que dan un toque de misterio acerca de qué es lo que realmente tienen. El término WYSIWYG⁴, especifica una importante característica de las aplicaciones Web, todo lo que uno puede obtener de una Aplicación Web se muestra inmediatamente tal y como es.

³ La palabra “escanear”, y sus derivados, proviene de la palabra inglesa “scan”; describe la acción de ojear, un objeto, un texto, etc., buscando información relevante. También se entiende como el acto de digitalizar un documento a través de un dispositivo de entrada de datos como el “scanner”. Es un término técnico aceptado internacionalmente.

⁴ *What You See Is What You Get*, cuya traducción es “Lo que ves es lo que obtienes”

b. No se puede entrenar a los navegadores.

Al probar cualquier otro producto lo que generalmente se hace es entrenar a los usuarios. En este proceso se adapta e impone al usuario las restricciones de este. Sin embargo, en la Web no es factible hacer esto, en consecuencia se debe hacer que la Aplicación Web se explique a sí misma.

c. Productos reales (no) se venden en sitios Web.

A pesar de que muchas aplicaciones se diseñan con el propósito de vender productos, una buena cantidad de estudios señalaron que el porcentaje de clientes que de veras compran en la Web, es relativamente bajo. A pesar de tener la intención original de comprar, muchos usuarios llegan a virtualmente perderse en el camino de encontrar una forma de comprar y de pagar el producto.

d. En Internet no se sabe la verdadera identidad de los usuarios.

La comunicación en la Web es unilateral. Las aplicaciones Web no pueden reaccionar como un vendedor real y adaptarse rápidamente al cliente.

El último punto en particular es un problema fundamental de usabilidad en la Web: *la categorización de los usuarios*. Es bastante difícil levantar supuestos acerca de los usuarios y de lo que esperan de la aplicación. Mientras que se pueden hacer suposiciones de forma aproximada en el proceso de desarrollo de aplicaciones convencionales, la Web en cambio, es un territorio incierto para el desarrollador. Además, de diferentes factores que le conciernen al usuario, las condiciones concernientes al lado técnico del uso de las aplicaciones Web son extremadamente heterogéneas [Hitz et al, 2003].

Debido a que el usuario es quizás lo más relevante a la hora de probar cuan usable es un sitio es importante analizar la experiencia de este al navegar. Nielsen y Loranger (2006) describen los resultados de una prueba de un sitio en la que el 66% de las veces los usuarios completaron la tarea de forma exitosa. La cuestión es ¿Por qué la gente usa la Web si se falla en una tercera parte de las veces? En realidad, no se falla tanto. El fracaso ocurre cuando se utiliza una aplicación por primera vez. La mayoría de las personas invierten su tiempo de navegación en sitios en los que han tenido éxito anteriormente, entonces la medida en cuanto a completar tareas exitosamente es alta. Puesto que los usuarios escogen sitios basados en su experiencia inicial con estos, aquellos con elevada usabilidad tienen mayor oportunidad de ser seleccionados.

2.4. Guías de diseño

2.4.1. Tiempo de respuesta

Para prevenir que los usuarios de una aplicación interactiva pierdan el sentido de la interacción y el flujo de control, los tiempos de respuesta del sistema en todas las fases de la interacción deben estar condicionados según ciertos umbrales asociados con el mecanismo del proceso de la información del ser humano: el tiempo de respuesta del sistema equivalente a 0,1 segundos se siente como una respuesta inmediata; un tiempo de 3 segundos es tolerable, sin perder el “hilo” de lo que se está haciendo; después de 8 a 10 segundos, el usuario ya está distraído con algo más. Para aplicaciones Web, observar tales valores es un reto si se consideran las circunstancias técnicas de las conexiones de Internet que no son fácilmente calculables. Es por esto que se recomienda diseñar páginas “livianas”. En este sentido, se debe limitar el número de imágenes y otros elementos de gran

volumen, y utilizar métodos de compresión o reducir su formato, resolución o profundidad en el color, a fin de minimizar su tamaño. Si no se puede evitar utilizar imágenes de gran tamaño, es recomendable mostrar primero la versión pequeña de la imagen y descargar la imagen grande sólo si el usuario lo solicita. Una página que contenga un conjunto de imágenes pequeñas de muestra no debería exceder los 50 Kbytes. Otro método para acelerar la descarga de un sitio Web es especificar la dimensión de las imágenes en el texto HTML y así permitir que el browser represente la página antes de que las imágenes estén completamente descargadas [Hitz, 2003].

2.4.2. Disposición del texto

El texto en una Aplicación Web es un elemento sumamente importante y de trato delicado. En primer lugar, el texto debe ser legible y fácil de leer, obstaculizar la lectura rápida es un problema común de muchas aplicaciones saturadas de elementos que dificultan la lectura. Además, el texto debe ser claro y debe presentar la idea y la información precisa, no debe ser extenso y debe omitir detalles innecesarios. Existen diversas formas de manejar el texto dentro de una aplicación. En principio se considera qué tipo de información se quiere dar a conocer, ya que no es lo mismo un texto escrito como apoyo académico como un tutorial, a un texto escrito para informar acerca de las propiedades de un producto o un servicio. Debe considerarse también el tipo de lector al que se quiere llegar, no es lo mismo escribir para un joven de 25 años que podría leer letra pequeña con menos dificultad que para una persona adulta mayor que podría tener problemas visuales. Entonces, la escritura para la Web debe ser tratada con cuidado para que no sea el factor por el cual una aplicación no sea visitada.

Dos son los aspectos que deben ser considerados al disponerse un texto en una Aplicación Web:

1) Tipografía, legibilidad de la página y facilidad de lectura

Los esquemas correctos de tipografía y color son componentes esenciales para un buen diseño visual. Lo que generalmente se ve son diseños basados en marcas, gustos personales, dejándose de lado las necesidades de los usuarios y los resultados de estas prácticas son: texto muy pequeño o difuso, texto que no puede cambiar de tamaño fácilmente, color del texto contrastante de forma inadecuada con el fondo, y texto eclipsado por bastantes elementos de diseño alrededor [Nielsen y Loranger, 2006].

Leer texto en una pantalla es significativamente menos eficiente que en papel, es por esto que el texto debe ser distribuido cuidadosamente, en especial en aplicaciones Web con bastante texto [Hitz, 2003]. Una práctica común es confundir el diseño de publicaciones tales como revistas, periódicos, etc.; con el diseño de páginas Web. Parecería natural basarse en un estilo de diseño similar, pero ya en muchos aspectos la Web ha demostrado que “similar” no es lo mismo que “igual”, en este caso como en muchos, el medio de difusión es lo que hace la diferencia.

Una composición colorida y con bastantes elementos de diseño puede ser eficiente en una publicación impresa, ya que atrae la atención del lector, crea un ambiente para una mejor lectura y psicológicamente genera sentimientos placenteros. No ocurre lo mismo en las páginas Web porque una composición de estas características, distrae la atención del lector, en un determinado momento llega a ser fastidioso y aumenta los niveles de estrés [Nielsen y Loranger, 2006].

Fijar la distribución del texto debe ser evitado, así como las dobles o triples columnas, más que todo para permitir al browser adaptar la paginación y los diferentes tamaños de ventanas. Otro aspecto

que se debe diferenciar es la tipografía. En la lectura convencional, los tipos de caracteres serif⁵ son más adecuados para grandes trozos de texto, pero tienden a verse borrosos en la pantalla debido al anti-aliasing⁶ algorítmico, así que es mejor utilizar un tipo de letra sans-serif⁷ como Verdana, especialmente diseñado para verse en pantalla, con el adecuado tamaño de letra. Los textos gráficos, textos largos con letra mayúscula y los caracteres muy pequeños deben evitarse, tampoco es conveniente que los fondos tengan estructuras o patrones, ya que puede afectar sobremanera la legibilidad de los elementos de encima [Hitz, 2003] [Nielsen y Loranger, 2006].

Puesto que leer en la pantalla es más agotador, normalmente los usuarios tienden a leer diagonalmente, es decir, que escanean buscando palabras clave. En vista de este comportamiento, el texto debe ser conciso y distribuido en pequeños párrafos y lleno de palabras clave. Además, se debe utilizar el lenguaje del usuario sin complicarlo con términos de difícil entendimiento [Hitz, 2003].

En consecuencia las siguientes son las cuatro guías fundamentales para la tipografía en la Web [Nielsen y Loranger, 2006]:

a. Utilizar fuentes de tamaño común, a 10 puntos o más.

Al escoger el tamaño de fuente es recomendable un tamaño largo para evitar alienar a los lectores. El texto pequeño disminuye la legibilidad simplemente porque no es distintivo, incluso para gente con visión normal. El texto pequeño tiende a perder su forma característica en la pantalla, especialmente cuando se usa negrita o cursiva. El antialiasing también debe evitarse debido a que obliga a forzar la vista.

*Tabla 2: Tamaños de Fuente Recomendados
Fuente: Modificado de [Nielsen y Loranger, 2006]*

Tipo de Audiencia	Tamaño de Puntos
Audiencia General	10-12
Adultos mayores y personas con visión disminuida	12-14
Niños pequeños y otros lectores principiantes	12-14
Adolescentes/ Adultos jóvenes	10-12

Un tamaño de letra reducido no es la solución para incluir más texto en una página. Tener más contenido no significa que la gente lea más, de hecho lo más probable es que lea menos. El texto denso aleja al lector. Hay que tomar en cuenta que el tamaño de letra también varía dependiendo del tipo de letra, en algunos se ve más grande que en otros a pesar de que tengan el mismo tamaño de fuente.

b. Evitar páginas ocupadas y saturadas con diseño.

Un elevado número de colores y estilos de fuentes combinados con imágenes, gráficos y elementos como fondo, hacen que la página se vea saturada y ocupada. Se deben utilizar los elementos de diseño visual sabiamente, para ayudar a comunicar y crear orden, y no solo para decorar la página. Se debe procurar que todos los elementos que forman parte de la página cumplan con un objetivo y

⁵ Tipos de letras cuyos caracteres tienen alargamientos, por ejemplo Times New Roman

⁶ Anti-aliasing es una acción que suaviza los bordes dentados de los textos y formas basadas en vectores. Los diseñadores utilizan esto para refinar el aspecto de las letras, pero puede crear nuevos problemas como el texto borroso.

⁷ Tipos de letras cuyos caracteres no tienen alargamiento y son más bien lisos, por ejemplo Arial

evitar colocar elementos porque sí. Nada da muestras de un diseño amateur más que un fondo ocupado y saturado. No sólo luce mal, también desvirtúa la capacidad de la página para ser leída.

c. Utilizar texto negro en fondo blanco.

El texto negro sobre un fondo blanco o algo similar es más fácil de leer. En general, los colores oscuros son mejores para el texto, y los colores desaturados van mejor como fondo. En concordancia con una fuente y un tamaño correctos, un buen contraste de color asegura la legibilidad y una buena lectura de un sitio. Una vez más, leer en línea es mucho más difícil que leer en papel, entonces un buen contraste en el texto y en el fondo lo hace más fácil. Aún cuando el texto blanco en un fondo negro tiene el mismo contraste que a la inversa, la capacidad de lectura no es igual, especialmente en letras pequeñas. Los colores claros tienden a desvanecerse en el fondo oscuro. Si los fondos oscuros no pueden evitarse se puede aliviar el problema con otro color oscuro diferente del negro. Aunque el contraste es algo más bajo, es menos borroso.

Evitar usar colores similares como un fondo blanco y texto gris claro. El contraste bajo puede causar un esfuerzo visual y disconformidad. Esto se intensifica más para las personas con visión disminuida. También se debe tener cuidado al usarse colores vibrantes y complementarios como el morado y el amarillo. Mientras estos colores técnicamente son muy contrastantes, combinados producen un efecto vibrante en el texto de computadora que es difícil de leer.

En síntesis es bueno procurar utilizar el blanco y negro combinados o utilizar colores que varíen significativamente en intensidad. Una forma de verificar si la combinación de colores es distinguible es ver la pantalla en escala de grises, si los elementos relevantes son distinguibles es prueba de un buen contraste.

d. Movimiento, letras mayúsculas y el texto gráfico deben ser mínimos.

Las personas se sienten molestas con texto parpadeante o moviéndose en un sitio Web porque los distrae de lo que quieren hacer realmente. Asocian el texto dinámico con publicidad y procuran ignorarlo. Es irónico que el texto móvil es una de las tácticas utilizadas para atraer a los usuarios, de hecho termina espantando al usuario al considerarse como publicidad y de poca confianza.

El texto rico en letras mayúsculas reduce la rapidez al leer en un 10% aproximadamente. Además, aparentan figuras reconocibles tales como bloques, simulando o dando la sensación de que son murallas. El uso de letras mayúsculas en textos grande se ve infantil y amateur además de agresivo (da la sensación de gritar) y poco profesional. Es mejor reservar las letras mayúsculas para encabezados cortos y títulos.

El texto gráfico es apropiado a veces para partes del texto tales como botones, pero no para grandes bloques de texto. Los diseñadores que quieren usar fuentes “elegantes” no estandarizadas a veces tienden a sortear la capacidad del browser haciendo del texto una imagen. Escoger juiciosamente áreas para imágenes de texto es bueno en la mayoría de los casos, pero los textos gráficos en un sitio Web son problemáticos por muchas razones:

- *Los gráficos causan agrandamiento de archivos.* Recuérdese que la mitad de los usuarios en el mundo aún utilizan conexiones Dial-up, y tienen poca paciencia con los sitios que se descargan lentamente. Es común que las personas abandonen el sitio antes de que haya terminado de descargarse.

- *Los gráficos basados en texto no se pueden buscar*, y se escucha siempre la queja de que los motores de búsqueda no encuentran palabras en las aplicaciones Web.
- *El texto gráfico no es seleccionable*. Cuando se hace una búsqueda, las personas gustan de seleccionar un área de contenido y pegarla en un editor de texto para imprimir o como referencia futura.
- *El tamaño del texto gráfico no es modificable*. Incluso cuando el texto del sitio Web puede cambiarse de tamaño, los gráficos se quedan del mismo tamaño.
- *Los lectores de pantalla no pueden leer texto gráfico*. Texto alterno debe ser especificado.

2) Escritura y redacción para la Web

Escribir y redactar para la Web significa hacerlo lo suficientemente claro y conciso para que el usuario logre su objetivo y quiera seguir leyendo. Un contenido desorganizado y pobre suele inhabilitar al usuario para completar tareas básicas en la Web, tales como elegir productos. Cuando los usuarios encuentran sitios que facilitan el encontrar respuestas tienden a confiar y revisarlos. Un estilo visual puede ayudar a incitar la curiosidad y hacer interesante el contenido, pero cuando hay poco o ningún contenido valioso los clientes o usuarios tienden a irse. El contenido debe ser un punto focal en la Aplicación Web [Nielsen y Loranger, 2006].

Una Aplicación Web debe tener un buen contenido descriptivo, con pequeñas notas publicitarias que describan los elementos disponibles, un tono atractivo y una clara descripción enfatizando sólo lo esencial. Utilizar lenguaje simple y común también ayuda a llegar a un mayor número de usuarios que comprendan de lo que se está hablando y se sientan familiarizados con la situación. Esto de hecho atrae al lector a seguir viendo y asegura su entendimiento. Muy por el contrario, lo que no debería encontrarse en un escrito para la Web es texto superfluo, que enfatice demasiado los detalles auxiliares sin dejar claro cuales son los verdaderos motivos de la página. Asimismo, utilizar lenguaje con muchos términos técnicos o incomprensibles para la mayor parte del público hace que el lector asuma que no podrá entender el resto de la lectura y por lo tanto deje la página. Si la inclusión de tales términos es necesaria, entonces se debe procurar que el lector finalmente entienda el significado de alguna palabra, aclarando su definición o enlazarlo a un lugar donde pueda descifrarla. [Nielsen y Loranger, 2006]. Lo mismo se aplica al utilizarse jerga o expresiones muy locales, que si bien tiene la ventaja de poseer un tono familiar; sus significados, que no todos conocen, son más difíciles de encontrar en un diccionario o en otro lugar de la Web.

La Web es entendida como un medio para encontrar información rápida y el usuario siempre está ideando estrategias para acelerar la búsqueda y la comprensión de lo que busca. Ponerle obstáculos no es una buena forma de que se vuelva visitante habitual de la aplicación. Los usuarios no tienden a buscar información de la misma manera. Confían en las indicaciones visuales que les digan que están cerca de encontrar las respuestas. Las personas dirigen su atención a estas áreas e ignoran el resto [Nielsen y Loranger, 2006].

Esto no sugiere que el lector nunca lee información en la Web. Cuánto leen las personas, depende de sus objetivos y el nivel de información que requieren. En general las personas primero exploran rápidamente el texto para encontrar puntos clave, y si es necesario, revisan más detalles después. Si las personas pueden reconocer rápidamente las indicaciones hacia la información requerida entonces se acoplan al resto de las lecturas. Por el contrario, si no encuentran rápidamente algo significativo o se sienten abrumados abandonarán el camino. El escritor Web debe conocer a su “público”. Debe

estar familiarizado con sus intereses, su cultura, sus necesidades y limitaciones. La información debe ser filtrada y traducida a una forma en que el usuario potencial pueda digerir. Por otro lado, no hay que cometer el error de ser condescendiente con la audiencia. No se debe subestimar al usuario ni enmarcarlo en estereotipos [Nielsen y Loranger, 2006].

2.5. Diseño de página

El diseño de los sitios es cada vez más importante para la usabilidad, ya que los usuarios podrían no llegar a las páginas adecuadas, a menos que el sitio esté estructurado en función a las necesidades del usuario y que contenga un esquema de navegación que le permita encontrar lo está buscando [Nielsen, 2000]. A continuación se describen los factores preponderantes que se deben tomar en cuenta cuando se diseña una página:

1) Espacio de la pantalla

Las páginas Web deben estar presididas por un contenido que despierte la atención del usuario. Como norma el contenido debe ocupar al menos la mitad del diseño de una página y, preferiblemente, hasta un 80%. La navegación debe mantenerse por debajo del 20% del espacio de las páginas de destino, aunque las opciones de navegación pueden alcanzar proporciones más altas en las páginas de inicio y en las páginas de navegación intermedias.

Un principio general de todo el diseño de interfaces de usuario consiste en ver los elementos de diseño y eliminarlos uno por uno. Si el diseño funciona también sin un cierto elemento de diseño, es mejor quitarlo. Simplicidad es mejor que complejidad, especialmente en la Web, donde cada cinco bytes guardados es un milisegundo menos de tiempo de descarga.

2) Diseño de plataforma cruzada

En el diseño de Interfase Gráfica de Usuario (GUI) tradicional se controla cada uno de los píxeles de la pantalla, es decir, cuando se diseña un cuadro de diálogo, se puede estar seguro de que tendrá el mismo aspecto en la pantalla de usuario. Se sabe qué sistema se está diseñando, se sabe que fuentes instaladas tiene, se conoce el tamaño normal que tendrá la pantalla y se tiene la guía de estilos del fabricante, que indica las reglas necesarias para combinar los widgets de interacción. Pero en la Web no se puede aplicar muchas de las presunciones que se aplican al diseño GUI.

Los usuarios pueden acceder a la Web a través de computadoras tradicionales, pero también podría usar un dispositivo de mano, un teléfono móvil, etc. En la Web uno se tiene que acomodar a un factor de 100 en un área de pantalla entre las mini computadoras y las estaciones de trabajo, así como un factor de 1.000 de ancho de banda entre módems y conexiones T3.

La mayoría de las páginas Web solo funcionan bien en un monitor de 17 pulgadas que se ejecute a una resolución de al menos 1024 X 768 píxeles. Todo lo que pueda estar debajo no es válido, y los usuarios tendrán que desplazarse por la pantalla para ver todas las partes de la página. Al asegurar una orientación fácil dentro de una página, se debe evitar la necesidad de desplazamiento horizontal, porque los usuarios tienen que mover la estructura de la ventana permanentemente de un lado a otro. En contraste, el desplazamiento vertical es considerado un “pecado perdonable” [Kappel et al., 2003].

3) Diseño independiente de la resolución

Dado que no existe forma de saber el tamaño que pueden tener las pantallas de los usuarios, hay que diseñar para todas las resoluciones de pantalla. El principio básico del diseño independiente de la resolución consiste en no usar nunca un ancho de píxel⁸ fijo para ninguna tabla, marco o elemento de diseño (exceptuando las barras finas en la parte lateral de la página). En vez de usar tamaños fijos, habrá que especificar diseños como porcentajes del espacio disponible.

Los elementos gráficos también deben ser diseñados pensando en las distintas resoluciones. En concreto, los iconos deben seguir funcionando cuando se muestren con una resolución de 100 dpi's⁹ o más. Cuanto mayor sea la resolución, más pequeño será el gráfico, por lo que, para seguir siendo legible, todo texto que este incrustado en el gráfico deberá usar un tamaño de fuente relativamente grande. Por regla general, no conviene incluir texto en los gráficos, ya que al hacerlo se vuelve más lenta la transmisión y ocasiona más trabajo, al tener que traducir la interfaz de usuario a distintos idiomas.

4) Separar el significado de la presentación

El diseño original de la Web y su formato de datos subyacente, HTML, estaban basados en la codificación del significado de la información, y no en su presentación. El estilo de codificación fue elegido por Tim Berners Lee, porque quería que la Web fuera un sistema de información universal. Por tanto no podía saber que equipo tendrían los distintos usuarios, lo que hacia necesario mantener los detalles de la presentación fuera del archivo en sí. La manera exacta en que aparecían las páginas ante el usuario vendría determinado por el propio equipo del usuario.

Para la presentación de una Aplicación Web, es que los menús y los elementos de la navegación se colocan en el margen izquierdo de la página y en la parte superior. En general no se deben cargar las paginas excesivamente que influye en la memoria de almacenamiento [Kappel et al., 2003].

5) Vinculación

Los vínculos constituyen la parte más importante del hipertexto: conectan las páginas y permiten a los usuarios ir a nuevos sitios de la Web. Existen tres tipos principales de vínculos:

- *Vínculos de navegación estructural.* Estos vínculos esbozan la estructura del espacio de información y hacen posible que los usuarios acudan a otras partes de ese espacio.
- *Los vínculos asociativos del contenido de la página.* Estos vínculos suelen ser palabras subrayadas (aunque puede tratarse de mapas de imágenes) que señalan a las paginas con mas información relacionada con el texto del ancla.
- *Véanse también listas de referencias adicionales.* Estos vínculos se proporcionan para ayudar a los usuarios a encontrar lo que quieren si la página no es la adecuada.

⁸ Factor de medición de imágenes emitidas por la pantalla del monitor de la computadora.

⁹ *Dot per inches*, puntos por pulgada. Factor de medición de imágenes digitales.

Los vínculos deben tratarse de acuerdo a los siguientes parámetros:

1) Descripción de los vínculos

Los vínculos de hipertexto están anclados en el texto sobre el que el usuario hace clic para seguir el vínculo. Estas anclas no deben ser muy largas, ya que los usuarios buscan páginas para ver lo que son capaces de hacer los vínculos en una página determinada.

La regla de diseño más antigua de la Web consiste en evitar el uso de “*Haga clic aquí*” como texto de un vínculo de hipertexto. Subrayar las palabras importantes es vital, pero sería aun mejor incluir texto que proporcionara breve resumen del tipo de información disponible.

2) Títulos de los vínculos

La explicación de los vínculos se denomina título de vínculo¹⁰, tales explicaciones pueden dar a los usuarios una visión preliminar de dónde va a llevarles el vínculo y optimizar así su navegación. La información adecuada que se puede incluir en un vínculo es:

- El nombre del sitio que conduce al vínculo (si es destino del sitio actual).
- El nombre del subsitio al que conduce el vínculo (si se permanece en el sitio actual, pero se necesita ir a una parte distinta de éste).
- Los detalles añadidos acerca del tipo de información que se va a encontrar en la página de destino, como el modo en que se relaciona con el texto de ancla y con el contexto de la página actual.
- Los avisos sobre los problemas que pueden haber al otro lado del vínculo.

3) Colorear los vínculos

La mayor parte de los navegadores emplean dos colores diferentes para mostrar los vínculos: los vínculos con las páginas que el usuario no ha visto antes, suelen aparecer en azul, mientras que los vínculos con las páginas que el usuario ha visto antes aparecen en morado o en rojo. Es importante para la usabilidad mantener esta codificación en los colores de sus vínculos.

Cuando se usan colores no estándar, los usuarios pierden la facultad de ver con claridad qué partes del sitio han visitado y que partes quedan por explorar.

4) Vínculos Externos

Al establecer un vínculo externo, este podrá ser diseñado de dos formas:

- La primera consiste en que el puntero cambie el color cuando éste esté en un vínculo saliente. También sería posible cambiar la forma del puntero dependiendo del nivel de calidad del sitio a partir de un servidor conocido.
- Una segunda opción es la de mostrar una breve información del sitio que pretende visitar, cuando el puntero esté en el link externo. Dicha información puede ser la dirección de la página destino.

¹⁰ El único inconveniente es que los títulos de vínculos añadirán aproximadamente una décima de segundo al tiempo de descarga de una página Web normal si la conexión se realiza vía MODEM (Nielsen, 2000).

La importancia de los vínculos externos, es que los diseñadores deben proporcionar al usuario los más importantes y útiles vínculos, de tal manera que sean una guía para ellos.

5) Vínculos Internos

Los vínculos internos se constituyen como uno de los medios que generan tráfico en un mismo sitio, la existencia de estos son favorables ya que pueden permitir publicidad gratuita. La manera más fácil de permitir los vínculos entrantes consiste en que haya URL permanentes con cada una de sus páginas. Siempre que los usuarios vean un URL de una de sus páginas, podrán copiarlo como vínculo de hipertexto en sus propias páginas, en la creencia de que “desaparecerán” en una determinada fecha.

La mejor forma de alentar los vínculos entrantes consiste en disponer de un contenido tan bueno que los demás quieran crear un vínculo con él. Esta vinculación facilita a los contenidos que quieren referenciar a los autores u otros.

6) Marcos

Los marcos rompen el modelo unificado de la Web e introducen una nueva forma de ver los datos que no se hayan integrado bien en todos los demás aspectos de la Web. Con los marcos, la visión de la información que tiene el usuario en la pantalla viene determinada por una secuencia de acciones de desplazamiento, en vez que sea por una sola acción.

El tema principal del uso de los marcos es el de asegurar que los URL sigan funcionando. Para hacerlo, todos los vínculos de hipertexto deben tener un atributo `TARGET="_top"` en su etiqueta de ancla (por ejemplo, ``). Añadir el atributo de destino `_top` hace que el navegador borre todos los marcos y que los sustituya por un nuevo framset. El framset de destino puede tener muchos marcos que sean idénticos a los del framset de salida y que permanezcan guardados en el navegador, pero al forzar una recarga completa, el navegador obtendrá un nuevo URL como destino. Esto significa que las acciones de navegación, como los marcadores, funcionen de nuevo y que el URL esté disponible para que otras personas enlacen con él.

7) Imprimir

El diseño Web debe proporcionar versiones imprimibles de documentos extensos. Las impresiones son preferibles si el usuario posee un sistema de archivado con soporte papel, con carpetas y cuadernos que contienen toda la información relacionada con un determinado proyecto.

La recomendación es la de generar dos versiones de todos los documentos Web extensos. Una versión debería ser optimizada para su visualización en línea fragmentándola apropiadamente en muchos archivos que usaran numerosos vínculos de hipertexto y una hoja de estilo orientada a la pantalla. Otra versión debería mantener el documento completo en un solo archivo común con un diseño optimizado para la impresión [Nielsen, 2000].

Actualmente, los usuarios necesitan descargar la versión imprimible de forma manual, pero es de esperar que los navegadores implementen el estándar para especificar versiones alternativas de documentos.

2.6. Estructura de la navegación

Lanzar al mercado apresuradamente aplicaciones Web sin arquitecturas de información efectivas, restringe a los usuarios de conseguir la información que buscan. Cuando esto sucede, los usuarios pueden rendirse o, aún peor, ir a un sitio diferente. Una Aplicación Web pobremente diseñada hace más que lento el trabajo de los usuarios, puede realmente disuadirlos de usar la Aplicación Web. Un sitio bien estructurado da a los usuarios lo que quieren cuando lo requieren [Nielsen y Loranger, 2006].

La navegación dentro una Aplicación Web representa, un criterio importante particularmente para la usabilidad. La estructura de la navegación, especifica cómo la interfaz de usuario global está organizada en los contextos de interacción, colecciones, y grupos, y que se presentan a los usuarios, y cómo los usuarios navegan entre éstos. Un buen diseño de navegación muestra a los usuarios dónde están, dónde se localizan la información requerida, y cómo conseguir la información que necesitan de una manera metódica. Una arquitectura de información apropiada hace al usuario explorar cómodamente y asegura que pueda volver fácilmente a la página previamente vista. [Hitz et al., 2006][Constantine y Lockwood, 2002][Nielsen y Loranger, 2006].

Según Hitz y sus colegas (2006) y Cueva (2004) las interfaces de navegación tiene que ayudar a los usuarios en:

- *¿Dónde estoy?*, retroalimentación constante en la ubicación actual dentro de la estructura.
- *¿Qué puedo hacer?* o *¿Qué puedo encontrar aquí?*, información clara acerca del contenido en la página actual.
- *¿Dónde he estado?*, mecanismos que faciliten esta información.
- *¿Dónde puedo ir?*, información clara de los ítems alcanzables en el próximo paso de interacción.

En una Aplicación Web bien organizada, los usuarios pueden desenvolverse libremente, enfocados en sus tareas, sin tener que preocuparse por la estructura del sitio. Es trabajo del diseñador preocuparse por la estructura de la Aplicación Web, no de los usuarios [Nielsen y Loranger, 2006].

En general la estructura navegacional de una Aplicación Web, tiene que visualizar la ubicación actual del usuario y sus movimientos alternativos. Una arquitectura deficiente de la información siempre llevará a una deficiente usabilidad. Las dos reglas más importantes son: tener una estructura y hacer que ésta refleje el punto de vista del usuario sobre la Aplicación Web, de la información o de los servicios. La estructura de una Aplicación Web debe estar determinada por las tareas que los usuarios van a llevar a cabo en su Aplicación Web. La mayor parte de las aplicaciones Web poseen una estructura jerárquica [Cueva, 2004].

A continuación se resume las guías de diseño para la estructura navegacional de una Aplicación Web, sugerida por Nielsen y Loranger (2006), de acuerdo a una prueba de usuario realizada por los autores, enfocados en cuatro elementos: menús, nombres de categorías, enlaces y arquitectura de la información:

1) Consistencia navegacional

Mantener una estructura navegacional consistente ayuda a visualizar a los usuarios su ubicación y opciones actuales, minimiza conjeturas. Los elementos navegacionales actúan como un puente para ayudar a las personas a desplazarse de un área a la otra.

Cuando la navegación cambia drásticamente de una página a la siguiente, las personas deben cambiar su centro de atención: de usar el sitio a deducir cómo usarlo. Las aplicaciones Web con una navegación inestable hacen a los usuarios inseguros e indecisos.

Una buena estructura navegacional es predecible y hace a las personas sentirse cómodas explorando la Aplicación Web, que no necesita ser estudiada o memorizada porque refleja de forma correcta cómo debe representarse la información en el espacio Web. Tiene sentido y orden, y hay poca o ninguna ambigüedad sobre donde están los ítems. Los usuarios pueden avanzar, retroceder, explorar, y sentirse seguros de que no perderán su camino.

2) Ítems principales visibles

Los temas principales deben ser estáticos y aparecer en seguida para que los usuarios puedan navegar rápidamente a través de las opciones. Cualquier tipo de navegación dinámica necesita ser fácil de operar. Los menús que son demasiado sensibles e incluso cambian al más ligero movimiento del ratón pueden ser más complejos y terminarán con las oportunidades de conseguir cualquier interés de los usuarios comunes, las personas con discapacidades, y los usuarios Web principiantes.

No esconder los ítems de navegación principales. Es más rápido para todos inspeccionar las opciones cuando son simplemente visibles.

En síntesis, evitar la navegación extravagante y atractiva. Los usuarios lo repudian. Ahorrar juicios creativos para aquellas áreas de la Aplicación Web en que los usuarios estén realmente interesados.

3) Reducir el desorden y evitar la redundancia

No mantener áreas de navegación múltiples para los mismos tipos de enlaces. Las categorías que están duplicadas o indistinguibles complican la interfaz, haciendo el trabajo de los usuarios difícil. Los usuarios deben esforzarse para deducir la diferencia entre los enlaces con nombres similares. No exagerar con enlaces esparciéndolos sobre la página, debido a que realmente disminuye la probabilidad que los espectadores los vean. Con muchos elementos compitiendo, todos los ítems pierden importancia. Es mejor ofrecer algo claramente en una ubicación. Reducir la redundancia minimiza el desorden, hace más fácil para las personas encontrar información.

4) Especificar enlaces y nombres de etiquetas

Asegurar que los usuarios puedan entender las etiquetas navegacionales fácilmente. Mantener los nombres de los enlaces tan breves y específicos como sea posible para maximizar la exploración. Palabras o nombres de categorías inventadas son problemáticas porque los usuarios no las entienden. Si se necesita tener nombres inventados, siempre debe explicarse su significado. Los usuarios tienden a omitir palabras sin sentido o significado. Nombrar los enlaces empezando con palabras claves o una palabra de información. Los enlaces que empiezan con frases idénticas o

redundantes exigen a los usuarios que lean cuidadosamente para entender las diferencias. No usar instrucciones genéricas como el “Haga Clic Aquí” como nombres de enlaces. En cambio, debe ayudarse a las personas a diferenciar los enlaces rápidamente dándoles nombres informativos.

5) Menús desplegables verticales

Los menús desplegables tienden a ser estrechos y permitir espacios pequeños para describir nombres de categorías. En lugar de arriesgar y proporcionar una lista larga de ítems cuyos significados se solapan o son inciertos, es mejor permitir a las personas hacer clic en el título principal y entonces llevarlos a otra página que claramente lista sus opciones con las descripciones apropiadas.

6) Evitar menús multi-nivel

Usar poco menús horizontales flotantes y mantener los niveles a no más de dos, más de eso cubre la página y es difícil de usar. Con demasiados niveles, la navegación se convierte en un problema, al punto de que para localizar la opción deseada es necesaria una constante expansión y contracción de sub-menús.

Asegurarse de que los menús dinámicos permanezcan el tiempo suficiente para que las personas hagan sus selecciones. Los menús volátiles que requieren precisión y que se los abra y se los cierre a los movimientos del ratón más ligeros son difíciles de controlar. Los usuarios hábiles podrían tener la experiencia para controlar los menús dinámicos sin mucha dificultad, pero una persona en promedio se esforzará demasiado.

7) Opciones de clicqueo

Asegurarse que las personas puedan decir lo que es fácilmente clicqueable y lo que no. No se les debe obligar a que pulsen clic en toda la pantalla para deducir dónde están los enlaces. Proporcionése enlaces de tratamientos visuales que son normalmente asociados con la capacidad de hacer clic: principalmente el texto coloreado y subrayando. No se debe depender solamente del cursor para indicar enlaces.

8) Acceso directo desde la página de inicio

Una de las estrategias de diseño más exitosas, es la colocación de enlaces directos en la página de inicio a un número muy pequeño de operaciones prioritarias. No importa qué bien se estructure una arquitectura de información o qué transparente sea la representación en su sistema de navegación, los usuarios pueden perderse o impacientarse si deben navegar a través de múltiples niveles. Los enlaces directos acortan y simplifican esto.

Desgraciadamente, no puede presentarse enlaces directos a un número grande de características en la página de inicio. Una lista larga de enlaces directos frustra otros objetivos de la página de inicio, como proveer las bases del sitio e informar a los usuarios sobre el alcance global de sus opciones.

3. ACCESIBILIDAD EN LA WEB

Hoy en día se ha incrementado el uso de sistemas publicados vía Web. Sin embargo los usuarios con restricciones físicas, sensoriales o cognitivas suelen encontrar problemas para acceder a la

información. No solo los usuarios con discapacidad son un ejemplo de la exclusión por falta de accesibilidad, similar es el caso de personas que cuentan con equipos o con visualizadores pequeños, conexiones de bajo ancho de banda, navegadores especiales, etc.

Ante la gran variedad de discapacidades, los diseñadores de aplicaciones Web se han mostrado reacios, argumentando que la accesibilidad limita la libertad de su diseño [Díaz, Montero y Aedo, 2005]. Lo que se intenta demostrar es el uso de pautas para un diseño accesible que no limite la libertad del diseñador.

3.1. Definición

Accesibilidad Web significa acceso a la Web por cualquiera, sin importar la discapacidad e independencia de los equipos y aplicaciones utilizados para su acceso: navegadores gráficos, textuales o de voz; ayudas técnicas (lectores de pantalla, ratones de cabeza, magnificadores, etc.); empleando únicamente el teclado, etc. [Nogales y Martín, 2007].

La accesibilidad es una de las propiedades de una Aplicación Web, que garantiza que esta pueda ser accedida y usada por todos los usuarios potenciales, independientemente de sus limitaciones propias o las derivadas del contexto de uso. Por tanto, incluye el uso de cualquier tipo de navegador, computador o elemento de acceso alternativo [Díaz et al., 2005].

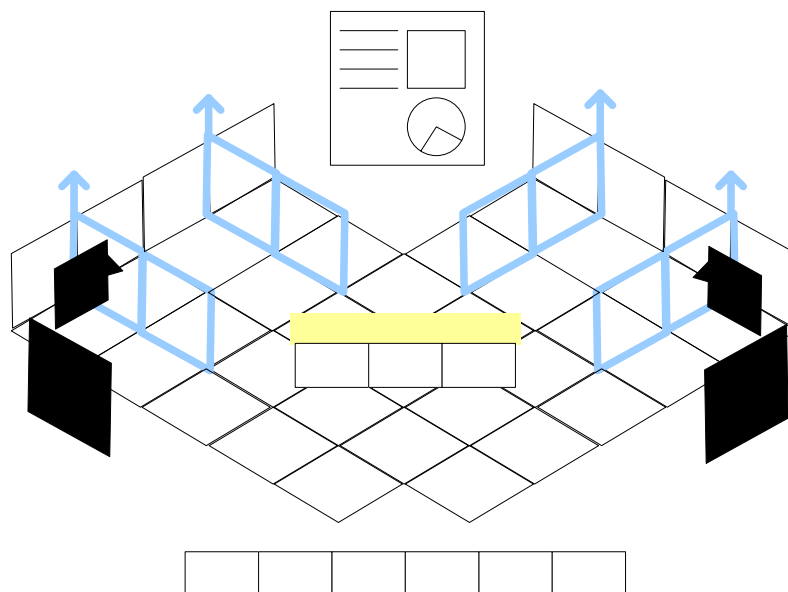


Figura 2. Los componentes y pautas para la accesibilidad Web
Fuente: [Gerti, 2003]

Finalmente se puede concluir que la accesibilidad permite que la información pueda ser presentada con independencia de los equipos y aplicaciones utilizados para su acceso: navegadores gráficos, textuales o de voz, etc. y que los usuarios con algún tipo de discapacidad puedan también tener acceso a la información proporcionada por la Aplicación Web.

3.2. Pautas de accesibilidad

El consorcio de la World Wide Web (W3C o World Wide Web Consortium) fue creada para lograr todo el potencial de la Web desarrollando protocolos que promuevan su evolución y aseguren su interoperabilidad [W3C, 2004]. Sus actividades están desglosadas en varios grupos de trabajo. Una de ellas es la Iniciativa de Accesibilidad a la Web (WAI: Web Accessibility Initiative).

A continuación se describen los trabajos desarrollados por la WAI relacionados al desarrollo de pautas de accesibilidad en cuanto se refiere a: contenido Web como a las herramientas de autor y los agentes de usuario [WAI, 2005]. Las siguientes son las pautas más importantes:

a. Pautas de accesibilidad al contenido Web

Las *pautas de accesibilidad al contenido Web* (WCAG: Web Content Accessibility Guidelines) son recomendaciones para que las páginas Web sean accesibles para todos con ayuda de la tecnología estándar existente [WCAG, 2004].

b. Pautas de accesibilidad a los agentes de usuario

Las *pautas de accesibilidad a los agentes de usuario* (UAAG: User Agent Accessibility Guidelines) son recomendaciones para que los navegadores y programas multimedia sean accesibles para todos y para que estas herramientas puedan cooperar mejor con los dispositivos de tecnología asistencial [UAAG, 2004].

c. Pautas de accesibilidad a las herramientas de autor

Las *pautas de accesibilidad a las herramientas de autor* (ATAG: Authoring Tool Accessibility Guidelines) son recomendaciones para que las herramientas de diseño de páginas Web sean accesibles para todos, así como el resultado generado por ellas [ATAG, 2004].

3.3. Pautas de accesibilidad al contenido Web 1.0

Las pautas de accesibilidad al contenido (WCAG 1.0) son:

- Explican cómo hacer accesible el contenido Web a personas con discapacidades.
- Están dirigidas a todos los desarrolladores de contenido Web y de herramientas de autor.
- Su objetivo primordial es promover la accesibilidad.
- No desaniman a los desarrolladores de contenido a que usen imágenes, vídeo, etc., sino que explican cómo hacer el contenido multimedia más accesible para una audiencia más amplia.

Al diseñar páginas Web, considérese que muchos usuarios pueden estar operando en contextos muy diferentes:

- No ser capaces de ver, oír, moverse, o procesar algunos tipos de información.
- Tener dificultades para leer o comprender el texto;
- No tener o poder usar un teclado o ratón;
- Tener una pantalla de sólo texto o pequeña, o una conexión a Internet lenta;
- No hablar o comprender de forma fluida la lengua del documento;

- Estar en una situación en que sus ojos, oídos o manos estén ocupados o interferidos;
- Tener una versión antigua de un navegador, o uno muy diferente, o de voz, o un Sistema Operativo distinto.

Cada elección de diseño accesible generalmente beneficia a varios grupos de discapacitados y a la comunidad Web como un todo, así las directrices tratan cuestiones de accesibilidad y dan soluciones de diseño accesible [Nogales y Martin, 2007].

3.3.1. Temas de diseño accesible

a. Asegurar una transformación "elegante"

Se aborda fundamentalmente desde la directriz 1 hasta la directriz 11.

Algunas claves para diseñar páginas que se transformen elegantemente según Nogales y Martin son [Nogales y Martin, 2007]:

- Separar la estructura de presentación.
- Proporcionar texto (incluyendo equivalentes de texto).
- Crear documentos que funcionen incluso si un usuario no puede ver u oír.
- Crear documentos que no contemplen sólo un tipo de hardware.

b. Hacer el contenido comprensible y navegable

Se aborda fundamentalmente desde la directriz 12 hasta la directriz 14. Esto incluye hacer el lenguaje claro y sencillo, pero también proporcionar mecanismos comprensibles para navegar dentro y entre páginas. Incluir herramientas de navegación e información de orientación maximiza la accesibilidad y usabilidad. No todos los usuarios pueden usar claves visuales. Los usuarios también pierden información contextual cuando sólo pueden ver una porción de una página. Sin información de orientación, los usuarios pueden no comprender tablas grandes, listas, menús, etc. [Nogales y Martin, 2007].

c. Directrices y controles

WAI (2004) afirma que existen 14 **directrices** o **pautas** (principios generales de diseño accesible), que incluyen:

- **Número** de la directriz.
- **Declaración** de la directriz.
- La **lógica** de la directriz y algunos **grupos de usuarios** a los que beneficia.
- Una lista de definiciones de **controles**, que explican cómo se aplica la directriz escenarios típicos de desarrollo de contenidos, y que incluyen:
 - **Número** del control.
 - **Declaración** del control.
 - **Prioridad** del control (los de prioridad 1 están resaltados con hojas de estilo).
 - **Notas informativas** opcionales, **ejemplos** clarificadores y **referencias cruzadas** a directrices o controles relacionados.
 - Un **enlace** a una sección del Documento de Técnicas donde se tratan implementaciones y ejemplos del control.

d. Prioridades

Cada control tiene un **nivel de prioridad** basado en su impacto sobre la accesibilidad

Prioridad 1 **Tiene que** cumplirse. Cumplirlo es un requisito básico para que algunos grupos puedan usar documentos Web.

Prioridad 2 **Debería** cumplirse. Cumplirlo eliminará barreras significativas para acceder a documentos Web.

Prioridad 3 **Puede** abordarse. Cumplirlo mejorará el acceso a documentos Web.

Algunos controles especifican un nivel de prioridad que puede cambiar en ciertas condiciones (indicadas).

e. Conformidad

WAI (2004) afirma que existen tres **niveles de conformidad** o adecuación con las directrices:

- "A": se satisfacen todos los controles de Prioridad 1.
- "Doble-A": se satisfacen todos los controles de Prioridad 1 y 2.
- "Triple-A": se satisfacen todos los controles de Prioridad 1, 2 y 3.

Las alegaciones de conformidad con las directrices pueden usar una de estas dos formas.

- Forma 1: Especificar el título de las directrices, su URI, el nivel de conformidad y el alcance cubierto por la alegación (página, sitio o porción definida del sitio).
- Forma 2: Incluir en cada página de la que alega la conformidad uno de los tres iconos proporcionados por el W3C enlazados con la apropiada explicación del W3C de la alegación.

Directriz 1. Proporcionar alternativas equivalentes al contenido auditivo y visual.

Se debe proporcionar contenido que, cuando sea presentado al usuario, transmita esencialmente la misma función o propósito que el contenido auditivo o visual.

- Equivalentes textuales de información no textual (gráficos, fotografías, videos, voz grabada, etc.). El texto puede ser presentado al usuario visualmente, como habla sintetizada. Para que sea útil, el texto debe transmitir la misma función o propósito que la imagen, ilustrar información específica geográfica, o contener un enlace que proporciona cierta información. Además los equivalentes textuales pueden ayudar a todos los usuarios a encontrar páginas rápidamente.
- Equivalentes no textuales del texto como iconos, habla pregrabada, vídeo de una persona que traduce el texto a lenguaje de signos. Una descripción auditiva es un ejemplo equivalente no textual, de información visual, que beneficia a usuarios que no pueden verla.

Controles:

1.1 Prioridad 1 Proporcionar un equivalente textual para cada elemento no textual (p.ej., vía "alt", "longdesc", o en el contenido del elemento). *Incluye:* imágenes, representaciones gráficas de texto (incluyendo símbolos), regiones de mapas de imagen, animaciones (como GIFs animados), applets y objetos, arte ASCII, marcos, scripts, imágenes usadas como viñetas de listas, espaciadores, botones gráficos, sonidos (reproducidos con o sin interacción del usuario), ficheros de audio autónomos, pistas audio de vídeo, y vídeo.

1.2 Prioridad 1 Proporcionar enlaces de texto redundante para cada región activa de un mapa de imagen del servidor.

1.3 Prioridad 1 Hasta que los agentes de usuario (AU) puedan leer en voz alta automáticamente el equivalente textual de una pista visual, proporcionar una descripción auditiva de la información importante de la pista visual de una presentación multimedia.

1.4 Prioridad 1 Para cualquier presentación multimedia basada en tiempo, sincronizar alternativas equivalentes con la presentación.

1.5 Prioridad 3 Hasta que los AU representen equivalentes textuales para los enlaces de mapas de imagen del cliente, proporcionar enlaces de texto redundante para cada región activa.

Directriz 2. No confiar solo en el color.

Asegurar que el texto y gráficos son comprensibles cuando se ven sin color

Si sólo se usa el color para transmitir información, las personas que no pueden diferenciar entre ciertos colores y los usuarios con dispositivos no visuales o que no trabajen con color no recibirán la información. Cuando el color de primer plano y de fondo están demasiado próximos en el mismo matiz, pueden no tener suficiente contraste al ser vistos en dispositivos monocromos o por personas con diferentes tipos de déficit de color [WAI, 2004].

Controles:

2.1 Prioridad 1 Asegurar que toda la información transmitida con color está disponible también sin él, por ejemplo, por el contexto o el marcado.

2.2 (Prioridad 2 para imágenes, Prioridad 3 para texto) Asegurar que las combinaciones de colores de primer plano y fondo tienen suficiente contraste vistas en una pantalla en blanco y negro.

Se recomienda el uso de propiedades CSS relativas al color para asignar colores diversos a los vínculos.

Se recomienda que no se hagan llamadas explícitas a objetos coloreados cuando el usuario deba intervenir para activar un enlace o un evento.

Se recomienda que se examine el documento en un monitor en blanco y negro para verificar que éste puede ser leído correctamente sin el uso de colores.

Se recomienda que exista un fuerte contraste entre el color del fondo del documento y el de primer plano. Para verificar este hecho se puede realizar una copia impresa en escala de grises.

Directriz 3. Utilizar marcado y hojas de estilo y hacerlo con propiedad.

Se debe marcar los documentos con los elementos estructurales apropiados. Controlar la presentación con hojas de estilo en lugar de elementos y atributos de presentación.

Utilizar de forma impropia el marcado (no acorde con la especificación) dificulta la accesibilidad. Usarlo para efectos de presentación dificulta a los usuarios con software especializado comprender la organización de la página o navegar por ella. Usar marcado de presentación en lugar de estructural para transmitir estructura, dificulta representar una página de forma inteligible para otros dispositivos [WAI, 2004].

Controles:

3.1 Prioridad 2 Cuando exista un lenguaje de marcado apropiado, debe usarse en lugar de imágenes para transmitir la información. Evitar también usar imágenes para representar texto; usar en su lugar texto y hojas de estilo.

3.2 Prioridad 2 Crear documentos válidos respecto a gramáticas formales publicadas.

3.3 Prioridad 2 Utilizar hojas de estilo para controlar la composición y la presentación.

3.4 Prioridad 2 Utilizar unidades relativas en lugar de absolutas en los valores de atributos del lenguaje de marcado y en los valores de propiedades de las hojas de estilo.

3.5 Prioridad 2 Utilizar elementos de encabezado para transmitir la estructura del documento y hacerlo de acuerdo a la especificación.

3.6 Prioridad 2 Marcar con propiedad las listas e ítems de listas.

3.7 Prioridad 2 Marcar las citas. No utilizar marcado de citas para efectos de formateo como el sangrado.

Directriz 4. Clarificar el uso del lenguaje natural

Utilizar un marcado que facilite la pronunciación o interpretación de texto abreviado o en otro idioma.

Cuando los desarrolladores de contenido marcan los cambios en el lenguaje natural en un documento, los sintetizadores de habla y dispositivos de lenguaje braille pueden cambiar automáticamente a la nueva lengua, haciendo el documento más accesible a usuarios multilingües. Deberían asimismo identificar el lenguaje natural predominante del contenido de un documento (a través del marcado o cabeceras HTTP), así como proporcionar expansiones de abreviaturas y acrónimos [WAI, 2004].

Controles:

4.1 Prioridad 1 Identificar claramente los cambios en el lenguaje natural del texto de un documento y de cualquier equivalente de texto.

4.2 Prioridad 3 Especificar la expansión de cada abreviatura o acrónimo en un documento donde ocurra por primera vez.

4.3 Prioridad 3 Identificar el lenguaje natural primario de un documento.

Directriz 5. Crear tablas que se transformen elegantemente.

Asegurar que las tablas tienen el marcado necesario para ser transformadas por navegadores accesibles y otros agentes de usuario.

Las tablas deberían ser usadas para marcar información verdaderamente tabular ("tablas de datos"). Los desarrolladores de contenido deberían evitar usarlas para componer páginas ("tablas de composición").

Algunos AU permiten navegar entre celdas de tablas y acceder a encabezados y otra información de celdas de tablas. A menos que se marque adecuadamente, esas tablas no proporcionarán a los AU la información apropiada.

Los siguientes controles beneficiarían directamente a los usuarios que acceden a una tabla a través de medios auditivos, o que ven sólo una porción de la página a la vez [WAI, 2004].

Controles:

5.1 Prioridad 1 Para tablas de datos, identificar encabezados de fila y columna.

5.2 Prioridad 1 Para tablas de datos que tienen dos o más niveles lógicos de encabezados de filas o columnas, utilizar el marcado para asociar celdas de datos y encabezados.

5.3 Prioridad 2 No utilizar tablas para composición a menos que la tabla tenga sentido cuando sea linearizada. Si así no fuera, proporcione una alternativa equivalente (podría ser una versión linearizada).

5.4 Prioridad 2 Si se utiliza una tabla para composición, evitar el uso de marcado estructural con fines de formato visual.

5.5 Prioridad 3 Proporcionar resúmenes para tablas.

5.6 Prioridad 3 Proporcionar abreviaturas para etiquetas de encabezados en tablas.

Directriz 6. Asegurar que las páginas que presentan nuevas tecnologías se transforman elegantemente.

Asegurar que las páginas son accesibles incluso cuando las tecnologías más nuevas no están soportadas o están desactivadas.

Aunque se anima a los desarrolladores de contenido para que usen nuevas tecnologías que resuelvan problemas generados por las tecnologías existentes, deberían saber cómo hacer que sus páginas funcionen aún con navegadores más antiguos y personas que desactivan características [WAI, 2004].

Controles:

6.1 Prioridad 1 Organizar los documentos de manera que puedan ser leídos sin hojas de estilo.

6.2 Prioridad 1 Asegurar que los equivalentes para el contenido dinámico son actualizados cuando el contenido dinámico cambia.

6.3 Prioridad 1 Asegurar que las páginas sean accesibles cuando scripts, applets, u otros objetos de programación sean desactivados o no soportados. Si esto no es posible, proporcionar información equivalente o una página alternativa accesible.

6.4 Prioridad 2 Para scripts y applets, asegurar que los gestores de eventos sean independientes del dispositivo de entrada.

6.5 Prioridad 2 Asegurar que el contenido dinámico es accesible o proporcionar una presentación o página alternativa.

Directriz 7. Asegurar el control del usuario, de los cambios de contenido que dependen del tiempo.

Asegurar que objetos o páginas que se mueven, parpadean, desplazan, autoactualizan puedan ser congelados o detenidos.

Algunas personas con discapacidades cognitivas o visuales son incapaces de leer texto en movimiento, con la suficiente rapidez o en absoluto. El movimiento puede también causar tal distracción que el resto de la página se haga ilegible para usuarios con discapacidades cognitivas. Los lectores de pantalla no pueden leer texto en movimiento. Los usuarios con discapacidades físicas pueden no ser capaces de moverse con velocidad o precisión suficiente para interactuar con objetos en movimiento [WAI, 2004].

Controles:

7.1 Prioridad 1 Hasta que los AU permitan a los usuarios controlarlo, evitar que la pantalla parpadee.

7.2 Prioridad 2 Hasta que los AU permitan a los usuarios controlarlo, evitar que el contenido destelle (esto es, cambiar la presentación a un ritmo regular, como si se encendiera y apagase).

7.3 Prioridad 2 Hasta que los AU permitan al usuario congelar el movimiento del contenido, evitar el movimiento en las páginas.

7.4 Prioridad 2 Hasta que los AU proporcionen la capacidad para detener el refresco, no crear páginas que se auto - refresquen periódicamente.

7.5 Prioridad 2 Hasta que los AU proporcionen la capacidad de detener una redirección, no usar marcado para redirigir páginas automáticamente. En su lugar, configurar el servidor para realizar redirecciones.

Directriz 8. Asegurar la accesibilidad directa de las interfaces de usuario incrustadas.

Asegurar que la interfaz de usuario sigue los principios del diseño accesible: acceso a la funcionalidad independiente del dispositivo, operabilidad de teclado, voz automática, etc.

Cuando un objeto incrustado tiene "interfaz propia", la interfaz, como la del propio navegador, debe ser accesible. Si la interfaz del objeto incrustado no puede ser accesible, se debe proporcionar una solución alternativa accesible [WAI, 2004].

Controles:

8.1 Prioridad 1 si la funcionalidad es importante y no se presenta en otro sitio, en otro caso Prioridad 2. Introducir elementos de programa tales como scripts y applets directamente accesibles o compatibles con tecnologías de ayuda.

Directriz 9. Diseñar con independencia del dispositivo.**Utilizar características que permitan la activación de elementos de página por medio de una variedad de dispositivos de entrada.**

Acceso independiente del dispositivo significa que el usuario puede interactuar con el AU o el documento con el dispositivo de entrada (o salida) preferido (ratón, teclado, voz, puntero de cabeza u otro. Generalmente las páginas que permiten interacción de teclado son también accesibles a través de entrada de habla o una interfaz de línea de comandos [WAI, 2004].

Controles:

9.1 Prioridad 1 Proporcionar mapas de imagen cliente en lugar de mapas de imagen servidor, excepto donde las regiones no puedan ser definidas con una figura geométrica disponible.

9.2 Prioridad 2 Asegurar que cualquier elemento que tenga su propia interfaz pueda ser operado de forma independiente del dispositivo.

9.3 Prioridad 2 Para los scripts, especificar gestores de eventos lógicos en lugar de dependientes del dispositivo.

9.4 Prioridad 3 Crear un orden de tabulación lógico para los enlaces, controles de formulario y objetos.

9.5 Prioridad 3 Proporcionar atajos de teclado para los enlaces importantes, controles de formulario y grupos de éstos.

Directriz 10. Utilizar soluciones provisionales.**Utilizar soluciones de accesibilidad provisionales de forma que las tecnologías de ayuda y los navegadores antiguos operen correctamente.**

Estos controles están clasificados como "provisionales" [WAI, 2004].

Controles:

10.1 Prioridad 2 hasta que los AU permitan a los usuarios desactivar las ventanas generadas, no permitir que aparezcan nuevas ventanas ni cambie la actual sin informar al usuario.

10.2 Prioridad 2 Hasta que los AU soporten asociaciones explícitas entre etiquetas y controles de formulario, para todos los controles de formulario con etiquetas asociadas implícitamente asegurar que la etiqueta está situada apropiadamente.

10.3 Prioridad 3 Hasta que los AU (incluyendo tecnologías de ayuda) representen correctamente texto contiguo, proporcionar una alternativa textual lineal (en la página actual o en otra) para todas las tablas que representan texto en columnas paralelas con texto multilínea.

10.4 Prioridad 3 Hasta que los AU manejen controles vacíos correctamente, incluir en cuadros de edición y áreas de texto, caracteres por defecto.

10.5 Prioridad 3 Hasta que los AU (incluyendo tecnologías de ayuda) representen enlaces adyacentes de forma separada, incluya caracteres imprimibles, no de entre enlaces adyacentes.

Directriz 11. Usar las tecnologías y directrices del W3C (World Wide Web Consortium).

Utilizar las tecnologías del W3C (de acuerdo a la especificación) y siga las directrices de accesibilidad. Donde no sea posible usar una tecnología del W3C, o hacerlo resulte en un material que no se transforme elegantemente, proporcionar una versión alternativa del contenido que sea accesible.

Convertir documentos a lenguajes de marcado del W3C no siempre crea un documento accesible. Por esto, se debe validar cada página en su accesibilidad y usabilidad después del proceso de conversión. Si una página no se convierte fácilmente, o bien revisar la página hasta que su representación original se convierta apropiadamente o proporcionar una versión HTML o de texto llano [WAI, 2004].

Controles:

11.1 Prioridad 2 Utilizar tecnologías W3C cuando estén disponibles y sean apropiadas para una tarea y utilizar las últimas versiones cuando estén soportadas.

11.2 Prioridad 2 Evitar las características reprobadas (o desaconsejadas) de las tecnologías W3C.

11.3 Prioridad 3 Proporcionar información de manera que los usuarios puedan recibir documentos de acuerdo a sus preferencias (lengua, tipo de contenido, etc.).

11.4 Prioridad 1 Si, a pesar de los esfuerzos, no se puede crear una página accesible, proporcionar un enlace a una página alternativa que use tecnologías W3C, sea accesible, tenga información (o funcionalidad) equivalente y sea actualizada tan a menudo como la página inaccesible (original).

Los desarrolladores de contenido solo deberían recurrir a páginas alternativas cuando fallen otras soluciones puesto que las páginas alternativas generalmente se actualizan con menos frecuencia que las "primarias".

Directriz 12. Proporcionar información de contexto y orientación.

Proporcionar información de contexto y orientación para ayudar a los usuarios a comprender páginas o elementos complejos.

Agrupar elementos y proporcionar información contextual sobre las relaciones entre elementos puede ser útil para todos los usuarios. Las relaciones complejas entre partes de una página pueden ser difíciles de interpretar para personas con discapacidades cognitivas y visuales [WAI, 2004].

Controles:

12.1 Prioridad 1 Establecer títulos en cada marco para facilitar su identificación y navegación.

12.2 Prioridad 2 Describir el propósito de los marcos y cómo se relacionan entre ellos si no es obvio sólo por sus títulos.

12.3 Prioridad 2 Dividir los bloques grandes de información en grupos más manejables donde sea natural y apropiado.

12.4 Prioridad 2 Asociar etiquetas explícitamente con sus controles.

Directriz 13. Proporcionar mecanismos de navegación claros.

Proporcionar mecanismos de navegación claros y consistentes (información de orientación, barras de navegación, un mapa del sitio, etc.) para aumentar la probabilidad de que un usuario encuentre lo que está buscando en el sitio.

Controles:

13.1 Prioridad 2 Identificar claramente el destino de cada enlace.

El texto del enlace (ancla) debería ser suficientemente significativo para tener sentido leído fuera del contexto. El texto del enlace debe ser también conciso.

13.2 Prioridad 2 Proporcionar metadatos para añadir información semántica a las páginas y sitios.

13.3 Prioridad 2 Proporcionar información sobre la disposición general de un sitio al describir la disposición, resaltar y explicar las características de accesibilidad disponibles.

13.4 Prioridad 2 Utilizar mecanismos de navegación de forma consistente.

13.5 Prioridad 3 Proporcionar barras de navegación para destacar y dar acceso a los mecanismos de navegación.

13.6 Prioridad 3 Agrupar enlaces relacionados, identificar el grupo (para los AU) y, hasta que lo hagan los AU, proporcionar una forma de salir del grupo.

13.7 Prioridad 3 Si se proporcionan funciones de búsqueda, se debe habilitar diferentes tipos de búsqueda para diferentes niveles de habilidad y preferencias.

13.8 Prioridad 3 Situar información que se distinga claramente al comienzo de encabezados, párrafos, listas, etc.

13.9 Prioridad 3 Proporcionar información sobre colecciones de documentos (esto es, documentos que comprenden múltiples páginas).

13.10 Prioridad 3 Proporcionar un medio para ingresar a un "ASCII art" cuando éste ocupe varias líneas.

Directriz 14. Asegurar que los documentos sean claros y simples.

Asegurar que los documentos sean claros y simples de manera que puedan ser comprendidos más fácilmente.

Una composición de página consistente, gráficos reconocibles y un lenguaje fácil de comprender benefician a todos los usuarios. En particular, ayudan a usuarios con discapacidad cognitiva o que tienen dificultades para leer [WAI, 2004].

Controles:

14.1 Prioridad 1 Utilizar el lenguaje más claro y simple apropiado para el contenido de un sitio.

14.2 Prioridad 3 Complementar el texto con presentaciones gráficas o auditivas donde faciliten la comprensión de la página.

14.3 Prioridad 3 Crear un estilo de presentación que sea consistente a lo largo de las páginas.

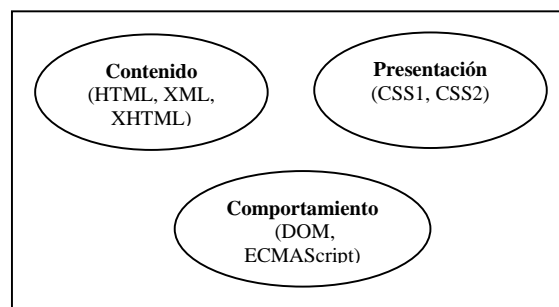


Figura 3: Separación del contenido, la presentación y el comportamiento.

Fuente: [Díaz et al, 2005]

3.4. Diseño accesible

El acceso a la información ha sido uno de los objetivos de la Web, en consecuencia el desarrollo por parte del consorcio W3C de estándares Web. Con la adopción del XML (eXtensible Markup Language) como estándar Web, se dispuso de un formato adaptable y universal para estructurar documentos e información. El uso de los estándares Web divide las páginas en tres componentes: contenido, presentación y comportamiento. Al estar separado el contenido de la presentación se consigue llegar a más usuarios con menos trabajo. El correcto uso de los estándares Web implica la reducción de problemas de accesibilidad [Días et al, 2005].

A continuación se proporciona un detalle de los estándares para el contenido, la presentación y el comportamiento:

a. Contenido

El lenguaje HyperText Markup Language (HTML) ha sido considerado como un lenguaje de publicación estándar de la Web, hace uso de un reducido conjunto de etiquetas estructurales y semánticas apropiadas para la realización de documentos relativamente simples [HTML, 2004].

La constante invención de elementos para HTML trajo graves problemas de compatibilidad de documentos entre diferentes plataformas y navegadores. El consorcio W3C solucionó el problema con la creación de lenguajes extensibles de XML (Extensible Markup Language), dando lugar al lenguaje XHTML (eXtensible HyperText Markup Language) [Díaz et al, 2005].

Lenguaje de marcado de hipertextos extensible

El Consorcio W3C define XHTML (Extensible HyperText Markup Language) como una familia de módulos y documentos presentes y futuros que reproduce, engloba y extiende HTML, reformulado en XML [XML, 2004]. XHTML proporciona una serie de ventajas desde el punto de vista de la accesibilidad:

- Al estar basado en XML, puede trabajar correctamente con otros lenguajes de marcado, pautas y protocolos basados en XML
- El soporte a XHTML está garantizado en la más modernas versiones de navegación así mismo en navegadores antiguos
- Manejado de manera correcta, es completamente portable a navegadores Web, lectores de pantalla, navegadores de texto, dispositivos móviles y otros agentes de usuarios especializados.
- Su utilización permite romper el hábito de utilizar etiquetas de marcado para ajustar la presentación
- Ofrece la posibilidad de testear el código utilizando validadores [Validator Service, 2004], lo que evita el esfuerzo de la revisión manual y evita muchos errores de accesibilidad básicos y
- XHTML 1.0 es un puente hacia futuras versiones de HTML, ya que su adaptación hacia ellas será mucho más sencilla desde HTML [Díaz et al, 2005].

b. Presentación

En un principio se consideró que era decisión del navegador el como presentar de la mejor manera las páginas al usuario, lo que produjo la proliferación de diferentes formas de visualización del contenido. La W3C decidió abordar el problema mediante el uso de *hojas de estilo*, especificadas mediante el estándar Cascading Style Sheet [Díaz et al, 2005].

Hojas de estilo

Las hojas de estilo (*Cascading Style Sheet* (CSS)) constituyen una especificación para la presentación de documentos HTML [Cascading Style Sheet, 2004]. Define el lenguaje para la creación de hojas de estilo, que funcionan como una plantilla, permitiendo a los desarrolladores y diseñadores Web definir un estilo para elementos HTML y aplicarlos a tantos elementos y documentos deseen [Díaz et al, 2005]. CSS proporciona una serie de ventajas a la accesibilidad:

- Separa la estructura del documento de la presentación
- Permite que los usuarios eliminen estilos del autor si lo desea
- Si la hoja de estilo del usuario contiene el operador **important**, tiene prioridad desde cualquier atributo aplicable en una hoja de estilo del autor
- Proporciona control exacto sobre aspectos como el tamaño de la fuente, el color y el estilo.
- Utiliza fuentes y colores del sistema para que las páginas se adapten a las necesidades y preferencias del usuario
- Usa unidades relativas en lugar de unidades absolutas en los atributos del lenguaje
- Controla la representación de los textos
- Controla la composición y alineación de los elementos y
- Permite la herencia de estilos [Accessibility Features of CSS, 2004].

c. Comportamiento

El W3C propone el uso de *Modelo de objetos de Documento (Document Object Model o DOM)* como modelo de objetos estándar de manera que se puedan crear comportamientos sofisticados y efectos que puedan funcionar en múltiples plataformas y navegadores.

4. MÉTRICAS PARA LA USABILIDAD Y ACCESIBILIDAD EN LA WEB

Una *métrica* (medida) es un valor numérico o nominal asignado a características o atributos de un ente computado a partir de un conjunto de datos observables y consistentes con la intuición [Cueva, 2005].

Se entiende como *ente* o *entidad* un objeto tangible o intangible, que está o exhibe comportamiento en el mundo real. El mundo real puede ser el mundo físico (hombre ordenador, teléfono), o un mundo intangible o abstracto (organización, evaluación, proyecto). Se entiende por atributo la característica o propiedad de un objeto o ente. La métrica debe ser en todo momento una medida válida, la medida debe ser una caracterización numérica apropiada del atributo de un ente, mostrando que se satisfaga la condición de representación, esto es, que la correspondencia entre el dominio empírico y el nuevo dominio numérico o simbólico preserve la relación de manera que estudiando y analizando los números se puedan explicar y conjeturar sobre el ente del mundo real. Cualquier medida que satisfaga la condición de representación es una métrica válida [Cueva, 2005].

Kitchenham et al. (1996) afirman que para decidir si una métrica es válida es necesario al menos confirmar [Cueva, 2005]:

- *La validez del atributo*: Su el atributo en cuestión es realmente exhibido por el ente que se desea evaluar.
- *La validez de la unidad*: si la unidad de medición a ser usada es apropiada para medir el atributo.
- *La validez del instrumento*: si el modelo subyace al instrumento de medición es válido y el mismo está plenamente calibrado.
- *La validez del protocolo*: si se ha adoptado un protocolo aceptable para la medición de modo que sea repetible y replicable.

4.1. Métricas principales de usabilidad

1) Evaluación de la usabilidad

La principal actividad en el proceso de usabilidad es la evaluación. La evaluación de la usabilidad puede ayudar a determinar cual es el nivel actual de la aplicación y si de hecho el diseño elegido realmente funciona. Los datos que se recaban mediante la observación del usuario frente a la aplicación y ver su desempeño, es información muy valiosa que ayudan en definitiva a detectar posibles falencias del sistema.

Existen diferentes técnicas para evaluar un sistema. Su uso depende de variables tales como costo, disponibilidad de tiempo, personal calificado para interpretar los datos, entre otros factores. A continuación se describen brevemente algunos de estos métodos [Baeza y Rivera, 2002]:

a. Inspección formal de usabilidad.

Un grupo de expertos realizan una especie de juicio de la interfaz, con uno de los participantes actuando como moderador, destacando las fortalezas y las debilidades de la aplicación.

b. Testeo de usabilidad

Se realizan pruebas de desempeño de un grupo de usuarios utilizando el sistema a probar y se graban los resultados para un análisis posterior. Esta actividad se puede desarrollar en un laboratorio con condiciones controladas o directamente en el lugar donde se va a utilizar el sistema.

c. Pensar en voz alta

Se le pide al usuario que realice una serie de tareas específicas. El usuario debe de expresar sus acciones oralmente. Dentro de las instrucciones dadas al usuario de prueba no se le pide que *explique* sus acciones, simplemente que cada paso que realice lo diga en voz alta (generalmente el mismo usuario da una serie de explicaciones sin pedírselo de manera explícita).

d. Evaluación heurística y de estándares.

En el área de Interfaces de Usuario existen una serie de estándares y de heurísticas ampliamente aceptadas (y probadas). En este tipo de evaluación un equipo de especialistas en usabilidad realiza una revisión conforme a estas normativas.

e. Caminata cognitiva.

Un grupo de expertos simula la manera en como un usuario *caminaría* por la interfaz al enfrentarse a tareas particulares.

Una manera bastante efectiva, y económica, de recabar información sobre los usuarios es mediante la aplicación de encuestas. Las encuestas deben de ser escritas y revisadas por un panel de especialistas para asegurarse de que se van a evaluar factores críticos de la interfaz. Las encuestas on-line pueden ser colocadas en un sitio Web ya en existencia, enviada por correo directamente a un

grupo de usuarios, enviada a listas de correos o colocada en grupos de noticias [Baeza y Rivera, 2002].

2) Métricas de éxito

La métrica más simple para medir la usabilidad de un sitio es la tarifa del éxito al realizar una tarea representativa. Es una métrica burda pero fácil de recolectar y constituye una estadística reveladora. Tarea terminada tiene peso 1, tarea a medio terminar tiene un peso 0,5 y no realizada 0 [Cueva, 2005].

$$\text{Éxito} = (n^{\circ} \text{ tareas terminadas} + (n^{\circ} \text{ medias } 0,5)) / n^{\circ} \text{ total de tareas}$$

Además de la tarifa del éxito, las medidas principales de la usabilidad son el funcionamiento de la tarea, el índice de errores, y la satisfacción subjetiva. Es posible medir la calidad del resultado según el tiempo que llevó realizar la tarea, si se acabó con éxito, cuánto se disfrutó ejecutando la tarea y si se la hizo de la manera más eficiente [Cueva, 2005].

3) Métricas de usabilidad sobre contenido del sitio (lectura)

Se considera el tiempo que toma realizarse una tarea, los errores producidos, el tiempo que toma el recordar la estructura del sitio, la satisfacción subjetiva y la memoria [Cueva, 2005]. La memoria toma dos medidas:

- *Memoria del conocimiento.* porcentaje basado en el número de respuestas correctas menos el número de respuestas incorrectas.

$$(n^{\circ}C - n^{\circ}I) * 100/n^{\circ}T$$

- *Memoria de memoria.* cuenta del porcentaje basada en el número de artículos recordados tras la prueba menos el número de los no recordados.

$$(n^{\circ}RC - n^{\circ}RI) * 100/n^{\circ}T$$

4) Métricas de confiabilidad

Se toman en cuenta las medidas de enlaces rotos (cantidad de enlaces rotos, densidad de enlaces rotos, concentración por nodo, etc.), las páginas muertas (páginas sin enlaces de retorno dentro de ella), los enlaces no implementados, los enlaces inválidos y los enlaces a e-mails mal contruidos [Cueva, 2005].

5) Métricas de contenido

Resulta del análisis de medidas tales como la cantidad de palabras por página, el promedio de palabras por página, el promedio de longitud de párrafos, la legibilidad del texto, el porcentaje de énfasis en las páginas (uso de negritas, itálicas y mayúsculas), la cantidad total de imágenes, el porcentaje de redundancia de imágenes, y otras medidas que dan cuenta de la disposición del contenido dentro de la página [Cueva, 2005].

6) Métricas y heurísticas de funcionalidad

Dos aspectos principalmente se tratan con estas métricas: la búsqueda y recuperación (búsqueda restringida, búsqueda global y personalización de la recuperación), La navegación y exploración (indicadores de camino, etiquetas de posición actual, permanencia de los controles contextuales, estabilidad, enlaces con título y el promedio de enlaces por página). También se analizan las deficiencias o resultados inesperados independientes de browsers, el testeo de páginas, cantidad total de Applets, cantidad de documentos con JavaScript, la cantidad de campos en las formularios [Cueva, 2005].

7) Métricas de eficiencia

Se realiza considerando aspectos como las páginas de acceso rápido (el tiempo de descarga estará en función del tamaño de la página estática y la velocidad en línea de conexión establecida), el soporte a versión solo texto, la cantidad de imágenes con título (se debe proveer texto alternativo, *ATL*, para cada imagen u objeto gráfico), porcentaje de presencia ATL y la cantidad máxima de marcos.

4.2. Métricas principales de accesibilidad

1) Herramientas de accesibilidad

Dado que la accesibilidad es un concepto muy amplio, exige a los diseñadores tener en cuenta muchos detalles. Por este motivo se han desarrollado herramientas para tratar la accesibilidad las cuales pueden clasificarse de acuerdo a varios conceptos [Evaluations Tools, 2004].

2) Herramientas de validación automática

Estas herramientas analizan el cumplimiento de un conjunto de pautas de accesibilidad (usualmente las WCAG o Sección 508). Entre estas se mencionan: Bobby™ Accessibility Tool (Bobby™), Wave Accessibility Tool (Wave), Test de Accesibilidad a la Web (TAW), etc. [Nogales y Martin, 2007].

Se puede también mencionar a herramientas que verifican si el código HTML, XHTML o CSS cumple el estándar correspondiente, lo que permite detectar código que está en desuso, se puede mencionar entre estas herramientas: W3C CSS Validator , W3C HTML Validator Service, WDG HTML Validator [Nogales y Martin, 2007].

3) Herramientas de ayuda a la corrección

Se puede mencionar a las herramientas de ayuda a la corrección general: Tidy, AccRepair, A-Prompt [Nogales y Martin, 2004].

4) Accesibilidad como medida de calidad

La usabilidad es uno de los atributos que hay que medir, clasificar y valorar. Según la norma ISO 9126, la usabilidad se refiere a la capacidad de software de ser comprendido, aprendido, usado y resultar atractivo al usuario. La accesibilidad y la usabilidad están estrechamente relacionadas, pues de ambas disciplinas comparten objetivos comunes como la satisfacción, efectividad y eficiencia del software [UsableNet, 2004]. Por otro lado, según las pautas WCAG 2.0 para que una Aplicación Web sea accesible se han de cumplir los principios de perceptibilidad, manejabilidad,

comprensibilidad y robustez. Estos atributos se describen en la norma ISO 9126, por tanto se pueden definir métricas que midan dichos atributos.

La W3C publicada mediante la WAI recomienda unas pautas de accesibilidad para el desarrollo de contenido Web. En este sentido se establecen métricas que cuantifican la información visual presente en las páginas, y dará una idea aproximada de la información no accesible en la Web. En la mayor parte de las herramientas existentes no proporcionan métricas que permitan definir a priori la calidad de una página Web en cuanto a su accesibilidad. Algunas de las posibles métricas para definir esta información son [González, Macías, Lozano, Nieto y Sánchez, 2001]:

- Cantidad total de imágenes en el sitio.
- Cantidad total de imágenes distintas en el sitio analizado.
- Porcentaje de información visual distinta, calculada como la relación entre la cantidad de imágenes distintas y la cantidad total de imágenes.

A cada una de las imágenes que aparecen en la Web se le puede asociar o un texto alternativo o bien una descripción larga del contenido de la imagen. Si el texto incluido es suficientemente descriptivo, podrá ser utilizado por los lectores y proporcionar la información que no es accesible directamente a través de la imagen. Por tanto, se puede medir [González et al., 2001]:

- Cantidad de imágenes con texto alternativo.
- Cantidad de imágenes con texto alternativo significativo. El que una imagen posea un texto alternativo no implica que el texto sea de calidad y por tanto que proporcione información adecuada.
- Porcentaje de imágenes con texto alternativo
- Porcentaje de imágenes con texto alternativo significativo.
- Porcentaje de texto alternativo descriptivo.

La interrelación entre el contenido y la presentación es una barrera en la presentación adecuada de la información según las necesidades del usuario final, por lo que se puede medir [González et al., 2001]:

- Si se utiliza HTML 4.0, el número de páginas que utilizan un DTD para determinar la estructura del documento.
- Número de aplicaciones de la etiqueta FONT en el código HTML. El uso de esta etiqueta debe ser evitado, un usuario con deficiencia visual necesita de un tamaño de letra mayor, si el tamaño de la fuente no está embebido en el código, su adaptación será más simple.

Además de la presentación de la información es importante mantener la funcionalidad de las páginas incidiendo en su interconexión mediante los enlaces, y además, el estado de éstos. Respecto a la interconexión de las páginas [González et al., 2001]:

- Número de páginas muertas. Se contarán aquellas páginas que no poseen ningún enlace, por lo que una vez que se ha llegado a ella no se puede seguir navegando, a no ser por los controladores del navegador, si es que existen.
- Porcentaje de páginas muertas.

Respecto al estado de los enlaces que interconectan las páginas se han de contabilizar especialmente aquellos cuyo estado es inaceptable. Si la funcionalidad es crítica para cualquier usuario, para un usuario con limitaciones lo es aún más [González et al., 2001]:

- Número de enlaces rotos internos. Considerando enlace interno a aquél que hace referencia a una página del sitio que se está analizando, se contará aquellos enlaces cuya dirección es incorrecta.
- Número de enlaces rotos externos. Como enlace externo se consideran aquellos que no pertenecen al sitio evaluado.
- Número total de enlaces.
- Porcentaje de número de enlaces rotos internos.
- Porcentaje de número de enlaces rotos externos.

5. INTEGRACIÓN DE CRITERIOS DE USABILIDAD Y ACCESIBILIDAD

Las guías de usabilidad y accesibilidad pueden ser integradas arbitrariamente en un modelo de procesos de desarrollo. Esta sección trata la integración de estas guías dentro de un método de desarrollo, en este caso UML-Based Web Engineering (UWE), se basa en las fases de análisis y diseño ilustrando la dependencia del uso de los criterios de usabilidad y accesibilidad. Se considera que estos criterios de calidad pueden ser transportados a este modelo de procesos sin mayor dificultad, es importante considerar que el proceso debe ser iterativo y estas iteraciones deben ser controladas de manera que las metas de usabilidad y accesibilidad se cumplan. Para que esto ocurra se pueden especificar los puntos de control a medida que se vaya aplicando el método dentro del espectro de la usabilidad y accesibilidad.

Para construir un modelo de proceso para el desarrollo usable y accesible, en el caso de las aplicaciones Web, los desarrolladores suelen tener que implementar la aplicación requerida en periodos breves de tiempo, lo que hace que sea necesario una metodología específica para el desarrollo de aplicaciones Web.

El modelo de proceso dependerá del tamaño de la aplicación y de las características del equipo de desarrollo, de acuerdo con Nielsen (2001), los modelos de procesos iterativos son los que mejor se adaptan al desarrollo de aplicaciones Web usables y accesibles. Hay que tomar en cuenta dos aspectos en el desarrollo de una Aplicación Web: si se desarrolla desde cero una Aplicación Web o se mejora la usabilidad y accesibilidad de una aplicación existente.

Como se puede apreciar, en ambos escenarios es importante prever y planificar evaluaciones periódicas de la usabilidad y accesibilidad durante su proceso de desarrollo. Cuando todas las especificaciones y pautas aplicadas (según la política interna de usabilidad y accesibilidad adoptada en la organización o las especificaciones del cliente sobre la usabilidad y accesibilidad que el producto deberá cumplir) hayan sido incorporadas terminará el proceso de desarrollo comenzando así la fase de mantenimiento o monitorización [Díaz et al, 2005].

Por otro lado, si es que se está mejorando la usabilidad y accesibilidad en una aplicación existente, se tendrá que realizar una evaluación inicial. Así se podrán detectar los problemas de accesibilidad para luego analizarlos y posteriormente diseñar el modo de corregirlo. En la fase de mantenimiento habrá que llevar a cabo evaluaciones frecuentes de estos criterios de calidad para determinar si los

cambios realizados en la aplicación influyen negativamente en el nivel de accesibilidad requerido [Díaz et al., 2005].

UWE es un método que proporciona guías para la construcción de modelos de forma sistemática y con fases acertadas enfocadas en personalización y en estudio de casos de uso. Las actividades de modelado principales son el *análisis de requerimientos*, el *diseño conceptual*, el *diseño de navegación* y el *diseño de presentación*, y producen los siguientes artefactos [Koch et al., 2001; Hennicker y Koch, 2001; Koch y Kraus, 2002]:

- Modelo de Casos de Uso
- Modelo Conceptual
- Modelo de Espacio de Navegación y Modelo de Estructura de Navegación
- Modelo de Presentación

5.1. Análisis de requerimientos

Primeramente, la integración con los criterios de usabilidad y accesibilidad al análisis de requerimientos empieza con la formulación de una “visión” describiendo el funcionamiento básico y las características de desempeño, los casos de uso esenciales. Así se traza el camino para crear los modelos de usuario en el análisis de requerimientos y controla las fases siguientes dentro de los criterios de calidad tratados. El análisis de la Aplicación Web a realizarse debe fundamentarse en el estudio de otras aplicaciones Web y la lectura relevante. Con esto no se está promoviendo la copia de aplicaciones existentes, sino la consideración de las mejores prácticas y lo que no debe hacerse frente a la propuesta propia y los propios requerimientos para trabajar en el concepto de la aplicación que se quiere desarrollar [Hitz, et al., 2003].

Uno de los más importantes resultados de la fase de análisis para el éxito de la aplicación de criterios de calidad es una clara definición de las metas cualitativas y cuantitativas por parte del analista de sistemas Web y los expertos en usabilidad y accesibilidad. Estas metas deben ser formuladas concretamente para que puedan ser ejecutadas y comprobadas efectivamente en el curso del proceso. Otra actividad importante al preparar los requerimientos es la especificación de los perfiles de los usuarios de manera que pueda diseñarse un proceso *centrado en el usuario*. En tal sentido, para formar una base para el análisis de requerimientos la categorización de los usuarios está caracterizada, no primeramente por la observación de los usuarios individuales, sino utilizando modelos de tareas y de uso, deducidas de los requerimientos inherentes de las tareas a ser realizadas [Hitz, et al., 2003].

Luego, para describir los requerimientos de la aplicación, utilizando UWE, se usa un *modelo de casos de uso*. Este modelo describe un trozo de comportamiento de la aplicación sin revelar su estructura interna. El modelo de casos de uso está conformado por dos elementos de modelado principales, llamados casos de uso y actores. Los pasos para dirigir este proceso de casos de uso son [Koch et al., 2001]:

1. Identificar a los actores
2. Para cada actor identificar las actividades que desempeña
3. Agrupar las actividades en casos de uso
4. Establecer relaciones entre los actores y los casos de uso
5. Establecer relaciones de tipo «include» y «extend» entre los casos de uso.

6. Simplificar el modelo de casos de uso mediante la definición de relaciones de herencia entre actores y casos de uso

5.2. Diseño de la aplicación

5.2.1. Modelo conceptual

El diseño conceptual, con UWE, está basado en el análisis de requerimientos previamente establecido. Incluye a los objetos involucrados en la interacción entre el usuario y la aplicación, especificado en los casos de uso. Apunta a la construcción de modelos de clase con estos objetos, que intentan ignorar tanto como sea posible los caminos de navegación y los pasos de presentación. Como método para construir este modelo de clases para el dominio se siguen técnicas de modelado tales como [Koch et al., 2001]:

1. Encontrar clases
2. Especificar los atributos y las operaciones más relevantes.
3. Determinar las asociaciones entre clases.
4. Agregar clases e identificar la composición de clases.
5. Definir las jerarquías de herencia
6. Definir las restricciones

5.2.2. Modelo de navegación

El *diseño de navegación* es un paso crítico en el diseño de la Aplicación Web. Por un lado, los enlaces aumentan la navegabilidad, por otro lado, sin embargo, incrementan el riesgo de perder la orientación. Construir un modelo de navegación no sólo es de gran ayuda para la documentación de la estructura, también permite acceder a un crecimiento estructurado de la navegabilidad. Este modelo se comprime en el modelo de espacio de navegación y el modelo de estructura de navegación. El primero especifica *qué* objetos pueden ser visitados mediante una navegación a través de la aplicación. *Cómo* estos objetos son alcanzados está definido por el modelo de la estructura de navegación [Koch et al., 2001].

El *modelo de espacio de navegación* es construido con las clases de navegación y las asociaciones de navegación y están representadas gráficamente por un diagrama de clases de UML. Se utilizan varias guías para ser seguidas por el desarrollador [Koch et al., 2001]:

1. Incluir clases del modelo conceptual que son relevantes para la navegación como las clases de navegación. Si la clase conceptual no es un objetivo de visita en el modelo de casos de uso, es irrelevante en el proceso de navegación, y por ende omitido en el modelo.
2. Mantener la información de las clases omitidas (si fuera necesario) como atributos de otras clases en el modelo de espacio de navegación. Todos los otros atributos de clases de navegación se ponen en mapas directamente como atributos de la clase conceptual correspondiente. En cambio, excluir los atributos de las clases conceptuales que son consideradas irrelevantes para la presentación en el modelo.
3. Las asociaciones del modelo conceptual son mantenidas en el modelo de navegación. Las asociaciones pueden ser añadidas para la navegación directa y, de esta manera, evitar caminos de navegación demasiado largos.
4. Añadir asociaciones adicionales basadas en la descripción de los requerimientos o de los escenarios descritos por el modelo de casos de uso.

5. Añadir limitaciones para especificar restricciones en el espacio de navegación.

El *modelo de estructura de navegación* describe cómo la navegación es soportada por elementos de acceso tales como índices, visitas guiadas, cuestionarios y menús. Técnicamente, los caminos de navegación junto con los elementos de acceso son representados por los modelos de clase que pueden ser sistemáticamente contruidos del modelo de espacio de navegación en dos pasos: El primer paso consiste en realzar el modelo de espacio de navegación con índices, visitas guiadas y cuestionarios. El segundo consiste en derivar menús directamente del modelo realzado. Los menús representan posibles elecciones de navegación. El resultado es un diagrama de clases UML contruido con estereotipos UML, los cuales están definidos según mecanismos de extensión UML. Para el modelado deben seguirse ciertas reglas que son resumidas a continuación [Koch et al., 2001]:

1. Reemplazar todas las asociaciones bidireccionales que tengan cardinalidad mayor que uno en ambos extremos de la asociación por dos asociaciones unidireccionales correspondientes.
2. Reemplazar todas las asociaciones bidireccionales que tengan cardinalidad mayor que uno en un extremo de la asociación con una asociación unidireccional con un extremo dirigido de la asociación en el extremo con cardinalidad mayor que uno. La navegación en la otra dirección está garantizada por el uso de árboles de navegación introducidas más tarde en el diseño.
3. Considerar solo aquellas asociaciones del modelo de espacio de navegación, que tengan cardinalidad mayor que uno en el extremo dirigido de la asociación.
4. Para cada asociación de esta clase, escoger uno o más elementos de acceso para entender la navegación.
5. Resaltar el modelo de espacio de navegación correspondientemente. Los nombre de los roles en este modelo son ahora movidos hacia los elementos de acceso. Si dos o más alternativas son introducidas en el paso 3, distinguirlas mediante el cambio de nombres de roles de la asociación por medio de búsquedas o el criterio del índice usado.

El paso siguiente es añadir menú y se sigue las reglas [Koch et al., 2001]:

1. Considerar esas asociaciones que tienen como fuente una clase navegación.
2. Asociar a cada clase navegación, que tiene en el modelo previo al menos una asociación de salida, una clase menú correspondiente. La asociación entre una clase navegación y su correspondiente clase menú es una composición.
3. Reorganizar un menú en un menú con submenús.
4. Introducir por cada rol, que ocurre en el modelo previo al lado final de una asociación dirigida un menú ítem correspondiente. Por defecto, el nombre del rol es usado como nombre constante del menú ítem.
5. Cualquier asociación del modelo previo que tiene como origen a una clase navegación ahora se convierte en una asociación del menú ítem correspondiente introducido en el paso 4. Notar que todos los pasos en el método anterior pueden ser realizados en una forma automática. Como resultado se obtiene un modelo de estructura de navegación comprensible de la aplicación.

5.2.3. Diseño de interfaz y modelo de presentación

Los diseñadores de la interfaz son los encargados de aplicar los criterios de usabilidad y accesibilidad una vez obtenidos los resultados del análisis de requerimientos y el desarrollo del modelo conceptual y el modelo de navegación. Este modelo define la estructura básica de la Aplicación Web, facilitando a los desarrolladores “ver más allá” de la aplicación. Las versiones

iniciales de este modelo están basadas en las subfunciones importantes del sistema, quizás sean las especificadas en el núcleo de los casos de uso. El modelo se enfoca en la representación básica de aspectos, metáforas, y la navegación y los principios de interacción. Es por eso que el modelo conceptual debe ser sincronizado en una base caso por caso para ilustrar como puede ser convertido en algo real [Hitz, et al., 2003].

Subsecuentemente, el modelo es evaluado por los expertos en usabilidad y accesibilidad. Tal inspección de calidad identifica las potenciales debilidades en el diseño de forma temprana, por ejemplo, secuencias inconsistentes, estrés potencialmente excesivo para el usuario, confusión de la estructura de navegación, inaccesibilidad para personas con capacidades disminuidas, etc. Entonces, tan pronto como el modelo conceptual ha sido establecido (en el sentido de un enfoque interactivo), los diseñadores de interfase y los implementadores pueden conjuntamente elaborar un diseño detallado de la interfase de usuario [Hitz, et al., 2003].

El diseño de presentación, en UWE, soporta la construcción de un modelo de presentación basado en el modelo de estructura de navegación e información adicional, se recolecta durante el análisis de requerimientos. Este modelo de presentación consiste en un conjunto de vistas que muestran el contenido y la estructura de los nodos simples, es decir cómo cada nodo es presentado al usuario y cómo el usuario puede interactuar con ellos. Se propone la construcción de *sketches*¹¹, *storyboards*¹² y modelos de flujo de presentación [Koch et al., 2001].

Para esto, primeramente, se realiza el *modelo de interfase de usuario abstracta* que implica el diseño de los elementos anteriormente mencionados. El diseño de storyboard puede ser considerado un paso tan opcional como el diseño relacionado a la interfase de usuario. Los sketches dan un primer *look and feel* de la interfase. Después de haber producido las diferentes vistas de interfases de usuario (sketches) los escenarios de la realización del storyboard pueden ser desarrollados, los cuales muestran secuencias de las vistas en el orden en el cual el usuario puede navegar de una vista a otra. El objetivo es visualizar la organización de la estructura de la Aplicación Web de una manera más intuitiva que en la fase de modelado de la estructura de navegación. Ambos, los sketches de las vistas así como la realización de los storyboards de los escenarios, son medios muy útiles para la comunicación entre un cliente y el diseñador Web.

Para diseñar el modelo de storyboard se empieza con el modelo de navegación de la Aplicación Web. Cada interfase de usuario abstracta es representada como una composición de clases. Las siguientes reglas pueden ser usadas para construir el modelo de presentación basado en las vistas de interfase de usuario [Koch et al., 2001]:

1. Construir una clase de presentación para cada clase de navegación que ocurra en el modelo de estructura de navegación. La clase presentación define una plantilla apropiada para presentar las instancias de la clase tomando en cuenta los atributos dados. Los elementos de interfase estereotipados tales como <<text>>, <<image>>, <<audio>>, <<video>> son usados por atributos de tipos primitivos y <<collections>> es usado para listas.

¹¹ Se trata de un anglicismo conocido, cuya traducción al español es “bosquejo”, es por esta razón que se utilizará la expresión en su idioma original

¹² Se trata de un anglicismo conocido, cuya traducción al español es “guiones gráficos”, es por esta razón que se utilizará la expresión en su idioma original

2. Construir una clase presentación para cada menú e índice que ocurra en el modelo de estructura de navegación. Esta presentación normalmente consiste de una lista de anclas. Los estereotipos `<<anchor>>` o `<<anchored collection>>` son usados para este propósito.
3. Construir una clase presentación para cada pregunta y visita guiada. Para las preguntas usar un estereotipo `<<form>>` y para las visitas guiadas usar un menú con los ítems *next* y *previous* (permite navegar al siguiente y al objeto previo sin una visita guiada)
4. Construir clases de presentación para soportar la navegación como composición de clases de presentación derivadas de las estructuras de acceso. Son usadas para reflejar el camino de navegación. Es decisión del diseñador donde incluir estas clases de presentación derivadas.
5. Añadir anclas a las clases de presentación para permitir la creación, destrucción y ejecución de operaciones sobre objetos del modelo conceptual. Los requerimientos funcionales de estas anclas provienen del modelo de caso de uso.
6. Determinar qué elementos de presentación deberían ser presentadas juntas al usuario (en una sola ventana). Las clases de presentación correspondientes deben ser compuestas en la vista de interfase de usuario (estereotipadas por `<<UI view>>`). Ya que el usuario necesita siempre una combinación de datos conceptuales y facilidades en la navegación, típicamente una vista de interfase de usuario consiste de la clase de presentación construida para la clase de navegación y de una clase de presentación construida para facilitar la navegación.
7. Construir escenarios de storyboard representadas por secuencias de vistas de interfases de usuario (opcional). Para este propósito se introducen enlaces que conectan un ancla (dentro de una vista UI) con otra vista UI para mostrar de esta manera los posibles flujos de presentación que puede ser causado por interacción de usuarios

Como paso siguiente se construye el *modelo de estructura y flujo de presentación*. El enfoque de este paso es modelar las dinámicas de la presentación mostrando dónde los objetos de navegación y los elementos de acceso serán presentadas al usuario, es decir en qué frames o ventanas el contenido es desplegado y qué contenido será reemplazado cuando un enlace es activado. Primero que nada, el diseñador tiene que especificar si es que una sola técnica es usada, si es que los frames son usados, y si es así, dentro de cuantos está dividido. En el caso de una ventana sin frames el resultado obviamente proviene del modelo de storyboard y no es necesaria una representación gráfica [Koch et al., 2001].

El modelo de presentación requiere que el diseñador tome algunas decisiones tales como el número de ventanas a ser usadas y el número de frames en la que está dividida. Por lo tanto, la construcción de la estructura de presentación no puede ser automatizada por completo, pero existen ciertas guías que el diseñador debe seguir [Koch et al., 2001]:

1. Escoger entre una técnica para ver una ventana o múltiples ventanas. En caso de elegir una técnica para ver múltiples ventanas planificar cómo muchas ventanas serán usadas.
2. Escoger el estilo del frameset, es decir con o sin frames. En el primer caso especificar cuántos frames tiene cada frameset.
3. Representar la estructura de presentación con un diagrama de clases UML (opcional).
4. Establecer el escenario para el modelo de interacción, o sea definir cuál camino de navegación del diagrama de estructura de navegación será modelado. Un camino de navegación relacionada a una caso de uso.
5. Representar a los usuarios, ventanas y los objetos frame en la dimensión horizontal.
6. Especificar un mensaje de *despliegue* para cada objeto de presentación que debería ser presentada al usuario (en una ventana o frame). El parámetro del mensaje de despliegue es el objeto de presentación correspondiente.

7. Incluir un mensaje de *selección* para cada acción de usuario que selecciona un ancla o un botón. Los nombres de anclas o botones son los parámetros del mensaje.
8. Especificar un mensaje de *llenado* y un mensaje de *introducción* para cada acción de usuario, que consiste en suministrar datos en una forma con preguntas. Esta forma es el parámetro de los mensajes.
9. Incluir un mensaje por cada apertura y cada cierre de una ventana.

BIBLIOGRAFÍA

Usabilidad

Constantine, L., & Lockwood, L. (2002). Usage-Centered Engineering for Web Applications. Obtenido el 21 de Abril de 2007, de <http://www.forUse.com>

Cueva, J. (2004). Usabilidad en la Web. Obtenido el 14 de Noviembre de 2006, de <http://www.di.uniovi.es/~cueva/asignaturas/doctorado/2004/UsabilidadWeb.pdf>

Díaz P., Montero S., Aedo I. (2005). Ingeniería de la Web y Patrones de Diseño. Pearson Educación, S.A.

Gerti K., Birgit P., Siegfried R., Werner R. (2003). Web Engineering. John Wiley & Sons, Ltda..

Hitz, M., Leitner, G. y Melcher, R. (2006). *Usability of Web Applications*. En Kappel, G., Pröll, B., Reich, S. y Retschitzegger, W. (Eds.), Web Engineering - The Discipline of Systematic Development of Web Applications (pp. 219-246), Alemania: John Wiley & Sons Ltd.

Nielsen, J. (2000). Usabilidad Diseño de Sitios Web. Madrid: Pearson Educación S.A..

Nielsen, J., & Loranger, H. (2006). *Prioritizing Web Usability*. Berkeley: New Readers.

Nogales T., Martin B. (2007). http://www.bib.uc3m._nogales/cursos/accesibilidad.html

Moner Cano, D. y Sabate Alsina, J. (2004). Usabilidad el guión multimedia. 1(1), 3-24

Accesibilidad

Díaz P., Montero S., & Aedo I. (2005). Ingeniería de la Web y Patrones de Diseño. Madrid (España): Pearson Educación S.A.

Gerti K., Birgit P., Siegfried R., & Werner R. (2003). Web Engineering. Heidelberg (Alemania): John Wiley & Sons, Ltda..

Universidad Carlos III de Madrid, Nogales T., Martin B. (2007). Obtenida el 16 de abril de 2007, de http://www.bib.uc3m._nogales/cursos/accesibilidad.html

World Wide Web Consortium. (2004). Obtenida el 11 de abril de 2007, de <http://www.w3.org>
Web Accessibility Initiative.(2005). Obtenida el 11 de abril de 2007, de <http://www.w3.org/WAI>

Web Contents Accessibility Guidelines. (2004). Obtenida el 11 de abril de 2007, de <http://www.w3.org/2004/09/wai-nav/intro/wcag.html>

User Agent Accessibility Guidelines. (2004). Obtenida el 13 de abril de 2007, de <http://www.w3.org/2004/09/wai-nav/intro/uaag.html>

Authoring Tool Accessibility Guidelines. (2004). Obtenida el 13 de abril de 2007, de <http://www.w3.org/2004/09/wai-nav/intro/atag.html>

HiperText Markup Language (HTML). (2004). Obtenida el 13 de abril de 2007, de <http://www.w3c.org/MarkUp>

Estensible Markup Language. (2004). Obtenida el 14 de abril de 2007, de <http://www.w3.org/XML>

W3C HTML Validator Service. (2004). Obtenida el 14 de abril de 2007, de <http://validator.w3.org>

Cascading Style Sheets. (2004). Obtenida el 16 de abril de 2007, de <http://www.w3.org/Style/CSS>

Accessibility Features of CSS. (2004). Obtenida el 16 de abril de 2007, de <http://www.w3.org/TR/CSS-access>

Document Object Model. (2004). Obtenida el 16 de abril de 2007, de <http://www.w3.org/DOM>

Document Object Model . Level 2. Core Specification. (2004). Obtenida el 16 de abril de 2007, de <http://www.w3.org/TR/2000/REC-DOM-Level-2-Core-20001113/>

Ecma International. (2004). Obtenida el 17 de abril de 2007, de <http://eee.ecma-international.org>

Evaluations Tools Demonstration. (2004). Obtenida el 17 de abril de 2007, de <http://www.w3.org/WAI/EO/2004/02/tools.html>

Bobby™ Accessibility Tool. (2004). Obtenida el 17 de abril de 2007, de <http://bobby.watchfire.com>

WAVE 3.0 Accessibility Tool. (2004). Obtenida el 17 de abril de 2007, de <http://wave.Webaim.org/index.jsp>

Test de Accesibilidad a la Web . (2004). Obtenida el 17 de abril de 2007, de <http://tawdis.net>

EvalAccess. (2004). Obtenida el 18 de abril de 2007, de <http://sipt07.si.ehu.es/evalaccess/>

W3C CSS Validator. (2004). Obtenida el 18 de abril de 2007, de <http://jigsaw.w3.org/css-validator>

WDG HTML Validator. (2004). Obtenida el 18 de abril de 2007, de <http://www.htmlhelp.com/tools/validator>

Tidy. (2004). Obtenida el 18 de abril de 2007, de <http://tidy.sourceforge.net>

AccRepair. (2004). Obtenida el 18 de abril de 2007, de <http://www.hisoftware.com/access/repair.html>

A-Prompt. (2004). Obtenida el 18 de abril de 2007, de <http://aprompt.snow.atoronto.ca/>

Referencias Adicionales

Cueva, L., (2005). Métricas de Usabilidad en la Web. Obtenido el 30 de abril de 2007, <http://www.di.uniovi.es/~cueva/asignaturas/doctorado/2004/MetricasUsabilidad.pdf>

Baeza, R., Rivera, C., (2002). Ubicuidad y Usabilidad en la Web. Depto. de Ciencias de la Computación, Universidad de Chile. Obtenido el 30 de abril de 2007, de <http://www.dcc.uchile.cl/~rbaeza/inf/usabilidad.html>

González, J., Macías, M., Lozano, A., Nieto, M.A., Sánchez, F., (2001). Métricas de calidad centradas en accesibilidad para KAI. Área de Lenguajes y Sistemas Informáticos, Departamento de Informática, Universidad de Extremadura.

Koch, N., (2000). Software Engineering for Adaptive Hypermedia Systems. Reference Model, Modeling Techniques and Development Process. Ludwig-Maximilians-Universität München, Germany.

Koch, N., Kraus, A., Hennicker, R. (2001). The Authoring Process of the UML-based Web Engineering Approach. Institute of Computer Science Ludwig-Maximilians University of Munich. Munchen Germany.

Koch, N., Kraus, A., (2002). The Expressive Power of UML-based Web Engineering. Ludwig-Maximilians University of Munich. Munchen Germany

Manchón, E. (2003). Errores comunes en el diseño de sitios Web. Junio, 5, 2006. Creative Commons. http://www.alzado.org/articulo.php?id_art=49.