

Alinhando os Requisitos de Usabilidade com as Diretrizes de Acessibilidade

Simone Bacellar Leal Ferreira

Universidade Federal do Estado do Rio de Janeiro – UNIRIO
Departamento de Informática Aplicada
simone@uniriotec.br

Ricardo Rodrigues Nunes

Universidade Federal do Estado do Rio de Janeiro – UNIRIO
Departamento de Informática Aplicada
ricardornunes@gmail.com

Denis Silva da Silveira

Universidade Federal de Pernambuco – UFPE
Departamento de Ciência da informação
denis.silveira@ufpe.br

Resumo

A interface com o usuário, parte do sistema através da qual os usuários se comunicam para executarem suas tarefas, deve atender as expectativas e necessidades de uma variedade de usuários. Elas devem poder ser acessadas por qualquer pessoa, independentemente de suas capacidades físico-motoras e perceptivas, culturais e sociais. A presente pesquisa tem por objetivo mostrar que, a obtenção de aplicativos *Web* acessíveis vai além de projetar *sites* em conformidade com as diretrizes de acessibilidade; questões de usabilidade devem ser consideradas. Para isso foram observadas interações entre deficientes visuais e diversos *sites* e foram analisados os problemas de usabilidade encontrados. A análise foi feita com base em uma taxonomia dos requisitos não-funcionais de usabilidade. Foi feito o alinhamento dessa taxonomia com as diretrizes de acessibilidades com a finalidade de auxiliar os profissionais de sistemas na identificação de problemas de acessibilidade e usabilidade que podem ser solucionados ou minimizados durante a definição dos requisitos e assim, facilitar a interação de deficientes visuais com a Internet e garantir *sites* com conteúdo compreensível e navegável.

1. Introdução

Com o avanço da Tecnologia da Informação, em especial da Internet, as pessoas e as organizações necessitam cada vez mais de sistemas capazes de fornecer informação com qualidade; essa demanda torna a interface com o usuário (IU) uma parte fundamental dos Sistemas de Informação já que é através dela os usuários se comunicam com os aplicativos para executarem suas tarefas; é importante que ela seja fácil de ser usada. Para isso, a interface deve atender as expectativas e necessidades de todos seus usuários (Ferreira, 2003).

Obter interfaces que satisfaçam a muitos usuários não é trivial, uma vez que existe uma diversidade de pessoas com necessidades distintas. As interfaces devem poder ser acessadas por qualquer pessoa, independentemente de suas capacidades físico-motoras e perceptivas, culturais e sociais (Nicholl, 2001). Ou seja, as interfaces devem ser projetadas em conformidade com as diretrizes de acessibilidade e com foco na usabilidade. Para isso, deve-se dar especial atenção aos requisitos relacionados à entrada de dados e à exibição de informação, de modo a garantir que a informação satisfaça aos requisitos não-funcionais de qualidade, inclusive seu caráter “amigável” e “acessível”.

Ao se projetar Sistemas de Informação, é raro encontrar alguma equipe que tenha a devida preocupação com a usabilidade alinhada com a acessibilidade. No Brasil, de acordo com o censo realizado em 2000 pelo Instituto Brasileiro de Geografia e Estatística (IBGE), cerca de 14,5% da população possui algum tipo de deficiência, sendo que desse grupo, 48,1% apresenta deficiência visual (IBGE, 2008). Esses valores indicam que desenvolver sistemas acessíveis

para essas pessoas não é altruísmo e, que embora tenha um lado social importante, é economicamente viável e deve ser considerado por todos que queiram fazer negócios através da Web (Queiroz, 2008).

A presente pesquisa tem por objetivo mostrar que, a obtenção de aplicativos *Web* acessíveis vai além de projetar *sites* em conformidade com as diretrizes de acessibilidade; questões de usabilidade devem ser consideradas. Para isso, foram observadas interações entre deficientes visuais e alguns *sites*, onde os problemas de usabilidade encontrados foram analisados. Durante a análise, realizada com base na taxonomia dos requisitos não-funcionais (RNF) de usabilidade (Ferreira, 2003), foi feito o alinhamento dessa taxonomia com as diretrizes de acessibilidades com a finalidade de auxiliar os profissionais de sistemas na identificação de possíveis problemas de acessibilidade e usabilidade que podem ser solucionados ou minimizados durante a definição dos requisitos, e assim, facilitar a interação de deficientes visuais com a Internet e garantir *sites* com conteúdo compreensível e navegável.

2. Acessibilidade

Acessibilidade é o termo geral usado para indicar a possibilidade de qualquer pessoa usufruir todos os benefícios de uma vida em sociedade, entre eles, o uso da Internet (Nicholl, 2001) e (NBR 9050, 1994); essa definição, proposta inclusive pela Associação Brasileira de Normas Técnicas (ABNT), apesar de forte impacto, é fundamental, pois acessibilidade de verdade só existe quando “todos” conseguem ter acesso a esses benefícios (Queiroz, 2008).

O termo acessibilidade na Internet é usado, de forma ampla, para definir o acesso universal a todos componentes dessa rede, como *chats*, *e-mail* entre outros. Já o termo acessibilidade na Web, ou *e-acessibilidade*, refere-se especificamente ao componente Web, conjunto de páginas escritas na linguagem HTML; a acessibilidade na Web representa o direito de acesso à rede de informações e eliminação de barreiras arquitetônicas, de comunicação, de acesso físico, de equipamentos e programas adequados, de conteúdo e apresentação da informação em formatos alternativos (Carter, 2007).

A Web desempenha um papel fundamental no avanço que a Internet representa no cotidiano das pessoas com deficiências visuais, facilitando a vida deles; ela permite que eles criem novas formas de relacionamento, encontrem oportunidades de trabalho e formas alternativas de diversão (Queiroz, 2008) e (Petrie, 2006). Mas apesar de indubitavelmente importante, a acessibilidade digital e na Web não é simples. As pessoas com deficiências possuem limitações sensoriais e motoras, que devem ser compensadas de alguma forma, a fim de viabilizar o acesso delas aos recursos computacionais. Para isso, as organizações necessitam adaptar seus *hardwares* e seus *softwares*, a fim de fazer com que um computador possa ser usado por pessoas com deficiências (Harrison, 2005). O problema é que essa adaptação requer um conhecimento técnico de especialistas, que faz com que as organizações não dediquem os esforços necessários ao processo de acessibilização (Carter, 2007).

Com relação à acessibilidade de Sistemas de Informação, existem quatro dificuldades com que podem se deparar usuários com necessidades especiais: uso do *mouse*; utilização do teclado; visualização do monitor e obtenção de sons de dispositivos de áudio.

3. Usabilidade Alinhada com as Diretrizes de Acessibilidade

A usabilidade é a característica que determina se o manuseio de um produto é fácil e rapidamente aprendido, dificilmente esquecido, não provoca erros operacionais, oferece um alto grau de satisfação para seus usuários, e eficientemente resolve as tarefas para as quais ele foi projetado (Nielsen, 2006). Uma aplicação orientada à usabilidade não necessariamente é orientada à acessibilidade, e vice-versa. Ou seja, ela pode ser de fácil uso para usuários comuns, mas inacessível para os com necessidades especiais (Hanson, 2004).

A facilidade que um usuário tem em interagir com uma interface está ligada aos aspectos de usabilidade, mas é também função da capacidade desse usuário detectar, interpretar e responder apropriadamente às informações do sistema. No ambiente computacional, grande parte da informação é apresentada na forma visual, o que faz com que a capacidade de usar computadores dependa muito de uma cadeia complexa de processos visuais (Moloney, 2006).

Como a visão passou a ser a principal forma de se interagir com os sistemas, não importa quão bem projetada seja a interface, ela não estará de acordo com o modelo conceitual dos usuários deficientes visuais e sempre constituirá uma barreira para eles. Esses usuários necessitam de uma tecnologia assistiva capaz de captar as interfaces e torná-las acessíveis. Tecnologia assistiva é o termo usado para identificar qualquer ferramenta, como uma bengala, ou um recurso, como um treinamento em *Braille*, utilizado para proporcionar ou ampliar as habilidades funcionais das pessoas com alguma deficiência e assim promover maior autonomia (Enap, 2007). No caso do acesso de uma pessoa com deficiência visual, acentuada ou total, à Internet, é feito de um programa leitor de tela (*screen readers*), aplicativos associados a programas sintetizadores de voz (Ferreira_B, 2007) e (Queiroz, 2008). Logo, as interfaces devem ser projetadas de modo que, quando acessadas por um leitor de tela, forneçam interações amigáveis. Elas devem oferecer sequências simples e consistentes de interação, mostrando claramente as alternativas a cada passo, sem confundir nem deixar o usuário inseguro, que deve poder se fixar só no problema que deseja resolver (Ferreira, 2003).

Visando o acesso universal a todos, no final dos anos 90, esforços começaram a ser desenvolvidos para promover a acessibilidade em aplicativos Web. No entanto, até o momento, sem muito foco na usabilidade, isto é, na simplificação da navegação e interação; conseqüentemente, problemas de usabilidade continuam existindo (Leporini, 2004) e (Takagi, 2004). Atualmente, através do comitê internacional W3C (*World Wide Web Consortium*), que regula os assuntos ligados à Internet, encontram-se um conjunto de catorze diretrizes para a Acessibilidade do Conteúdo da Web (WCAG 1.0). Essas diretrizes tratam de questões que dificultam o acesso a *sites* por usuários com necessidades especiais (W3C, 2008).

O processo de verificação de acessibilidade de *sites* é realizado através de programas que detectam o código HTML e fazem uma análise do seu conteúdo, verificando se está ou não dentro do conjunto das regras estabelecidas; no final, geram relatórios com a lista dos problemas encontrados que devem ser corrigidos para que o *site* seja considerado acessível. Entre esses *softwares* destacam-se: *WebXact* (antigo *Bobby*, que deixou de ser disponível ao público em 2008), e o *Hera* (um dos mais aderentes aos padrões Web). No Brasil, foi desenvolvido o programa, *daSilva*, que avalia os *sites* de acordo com as do WCAG e do *Modelo de Acessibilidade Brasileiro*, o e-MAG (Acessibilidade, 2008).

Contudo, é importante ressaltar que pessoas com necessidades especiais desenvolvem habilidades específicas. Por exemplo, os deficientes visuais usam as combinações das teclas de tal forma que uma pessoa com visão normal não conseguiria simular. Logo, para se obter um *site* de acesso universal orientado à usabilidade, além de verificá-lo através de programas avaliadores, é fundamental que se modelem as dificuldades e habilidades dos usuários, pois estas norteiam o modelo mental de suas interações. Essa modelagem viabiliza o alinhamento dos requisitos de usabilidade com as diretrizes de acessibilidade de forma a tornar a interação do deficiente harmoniosa, e, ao mesmo tempo garantir conteúdo compreensível e navegável.

Consideram-se problemas de usabilidade quaisquer características observadas capazes de retardar, prejudicar ou inviabilizar a realização de uma tarefa, que aborrecem, constrangem e, às vezes, traumatizam o usuário. Sendo assim, com relação à acessibilidade, os problemas de usabilidade, em geral ocorrem por três motivos: muito foco na conformidade com as diretrizes de acessibilidade e não na usabilidade; muitos programas avaliadores dependem somente de técnicas de verificação sintática dos *sites* para detectar a acessibilidade, com isso, os erros detectáveis se limitam à camada de descrição de *tags* (etiquetas) e não consideram aspectos de

usabilidade; os avaliadores de acessibilidade desconsideraram o fato que raramente os usuários escutam a saída falada de forma passiva. Eles se movimentam pelas páginas usando combinações de teclas e através desse processo, criam seus modelos mentais (Takagi, 2004).

4. Método de Pesquisa

O estudo, de caráter exploratório, teve quatro etapas: (a) escolha da categoria de usuários; (b) pesquisa bibliográfica e documental; (c) pesquisa de campo; (d) escolha do método de análise de usabilidade e alinhamento da taxonomia dos RNF de Usabilidade com as diretrizes de acessibilidade. As etapas foram realizadas concomitantemente.

a. *Escolha da Categoria de Usuários*: Para objeto de estudo optou-se pelos usuários com deficiências visuais; essa decisão foi devido ao fato de a *Internet* contribuir para melhorar a qualidade de vida dos deficientes visuais, permitindo que acessem informações que antes só podiam ser obtidas com a ajuda de uma pessoa, entre outras facilidades (Harrison, 2005).

b. *Pesquisa Bibliográfica e Estudo da Taxonomia dos RNF de Usabilidade*: num primeiro momento, buscou-se compreender o princípio de acessibilidade e suas implicações para *sites*. Nessa etapa, algumas instituições forneceram diferentes *softwares* destinados a usuários deficientes visuais. Esses *softwares* foram usados para navegar em *sites* “comuns”, como jornais, e aprofundar a observação e análise dos vários aspectos levantados na literatura.

c. *Pesquisa de Campo*: Foi conduzida uma pesquisa de campo no Instituto *Benjamin Constant* (IBC), órgão do Ministério da Educação, fundado em 1854, no Rio de Janeiro. Trata-se de um centro de excelência e de referência em matéria de estudos relacionados a deficiências visuais, e tem por principal objetivo promover a educação e integração do deficiente visual em toda a sua amplitude (IBC, 2008). Durante a pesquisa de campo, que durou três meses, foram realizadas atividades de observação em diferentes setores do instituto e foram feitas várias entrevistas informais e seis entrevistas em profundidade com funcionários, alunos e ex-alunos da instituição, que hoje atuam nela, sendo a maior parte composta de deficientes visuais.

d. *Escolha do Método de Análise de Usabilidade e Alinhamento da Taxonomia dos RNF de Usabilidade com as Diretrizes de Acessibilidade*: Nessa etapa estudou-se diversas formas de se analisar a usabilidade de sites. Optou-se pela análise seguindo a taxonomia de Ferreira e Leite (Ferreira, 2003) dos RNF de usabilidade. Essa taxonomia foi escolhida por estar sendo utilizada na análise de usabilidade de sites. Depois de selecionada, procurou-se alinhar os requisitos de usabilidade com as diretrizes de acessibilidade a fim de auxiliar os profissionais de Sistemas de Informação na identificação de problemas de acessibilidade e usabilidade que podem ser solucionados ou minimizados durante a definição dos requisitos.

5. Requisitos Não funcionais de Usabilidade

Deve-se projetar um produto tendo em vista a expectativa dos usuários e a facilidade de uso. O Sistema de Informação, sendo também um produto, não foge à regra; seu projeto deve seguir os princípios de Engenharia a fim de garantir a usabilidade.

A Engenharia de Requisitos, subdivisão da Engenharia de *Software*, procura sistematizar o processo de definição de requisitos dos *softwares*. A necessidade dessa sistematização decorre do fato da complexidade dos sistemas em demandar total compreensão do problema antes de buscar a solução. A Engenharia de Requisitos tem por finalidade propor métodos, técnicas e ferramentas para facilitar o trabalho de determinar o que se espera do sistema (Ferreira, 2003).

Os requisitos podem ser funcionais (RF) ou não-funcionais (NRF). Os RF descrevem o que o sistema faz, isto é, as funções necessárias para cumprir os objetivos do sistema. Já os NRF, entre eles, os de usabilidade, dizem respeito à qualidade do sistema, descrevem suas facilidades e ligam-se diretamente aos fatores humanos, que normalmente são negligenciados

pela Engenharia de *Software*. No entanto, desconsiderar esses fatores na definição de requisitos constitui-se uma das principais razões da insatisfação do usuário com o produto.

A construção de sistemas que levam em consideração aspectos relacionados às características dos usuários e à qualidade do sistema é um processo difícil por vários motivos. O principal deles provavelmente reside na definição de requisitos, primeira etapa no desenvolvimento de um sistema. Como etapa inicial, possui forte impacto na qualidade (ou na falta de qualidade) do desenvolvimento em si e do produto final (Pressman, 2004).

A taxonomia dos RNF de usabilidade, criada por Ferreira e Leite (Ferreira, 2003) com a finalidade de sistematizar o processo de definição desses requisitos e de reduzir os problemas de usabilidade. A tabela 1 mostra a taxonomia que, nesse artigo, foi usada para alinhar a análise de usabilidade com as diretrizes de acessibilidade.

Requisitos Relacionados à Exibição da Informação	Requisitos Relacionados à Entrada de Dados
A) Consistência B) <i>Feedback</i> C) Níveis de Habilidade e Comportamento Humanos D) Percepção Humana E) Metáforas F) Minimização de Carga de Memória G) Eficiência no Diálogo, Movimento e Pensamentos H) Classificação Funcional dos Comandos I) Manipulação Direta J) Exibição Exclusiva de Informação Relevante K) Uso de Rótulos, Abreviações e Mensagens Claros L) Uso Adequado de Janelas M) Projeto Independente da Resolução do Monitor.	A) Mecanismos de Ajuda B) Prevenção de Erros C) Tratamento de Erros

Tabela 1: Taxonomia dos RNF de Usabilidade (Ferreira, 2003)

6. Taxonomia dos RNF de Usabilidade Alinhada com as Diretrizes de Acessibilidade

6.1 Requisitos Relacionados à Apresentação da Informação

Pode-se apresentar a informação de diversas formas: textos, imagens, sons entre outras. No entanto, independentemente da forma, a informação não deve apresentar-se incompleta, ambígua nem ininteligível. É, portanto, necessário obedecer a certos requisitos quando se deseja construir interfaces eficientes no que diz respeito às informações apresentadas.

A seguir são apresentadas práticas que, se seguidas, promovem a acessibilidade e tornam a interface fácil de ser acessada através de tecnologia assistiva, como um leitor de tela.

A) *Consistência*

A consistência, uma das principais características de usabilidade da interface, reduz a frustração provocada por comportamentos inesperados e logicamente incompreensíveis do sistema. Além disso, permite que a pessoa generalize o conhecimento de um aspecto do sistema para os outros, isto é, possibilita que o usuário aprenda certas tarefas e estenda esse conhecimento para as partes do sistema ou aplicativos semelhantes. Assim, quando usar outro programa com características semelhantes, não necessitará aprender de novo (Nielsen, 2006).

Na interface consistente, os menus, os comandos de entrada, as apresentações de informação e as funções têm, todos eles, apresentação, codificação e comportamento semelhantes. Para isso deve-se: manter uma uniformidade na codificação da combinação de teclas de atalhos; exibir sempre as mensagens nas mesmas posições lógicas; preservar a disposição de elementos, como itens nos menus; padronizar o emprego de maiúsculas e, finalmente, disponibilizar sempre comandos globais, como “Sair”, “Salvar”, “Ajuda”, etc.

Consistência com relação a disposição e apresentação da informação: os leitores de tela captam o código HTML da página, analisam esse código e linearizam os elementos do conteúdo na mesma ordem em que aparecem no código. Assim, todos os elementos da tela, como *links*, texto, figuras, etc. são concatenados em uma única coluna, na mesma ordem sequencial em que aparecem no código-fonte. Consequentemente, as pessoas com deficiência visual percebem a página como se fosse um documento textual que eles podem ler linha por linha (através das setas) ou *link por link* (através da tecla *tab*) (Nevile, 2005).

Para se manter a consistência das informações apresentadas através desses programas, é preciso lembrar que a ordem em que um elemento da tela aparece para o usuário que acessa uma página via um leitor de tela não é a mesma ordem em que ele aparece visualmente em um navegador. O que determina quando ele vai aparecer no leitor de tela é a sua posição dentro do código HTML; deve-se tomar cuidado com essa posição. É essencial que o código seja também consistente de modo a garantir que os elementos aparecerão sempre na mesma ordem ao longo das diversas interfaces que constituem o sistema.

Deve-se relacionar a localização dos elementos no navegador e sua respectiva posição no código. Muitas vezes, um elemento importante ocupa uma posição de destaque na página exibida em um navegador, mas ao ser analisado pelo leitor, por se encontrar no fim do código, acaba sendo um dos últimos elementos que o usuário nota. Além de perder sua posição de destaque, ele só será percebido depois de informações menos relevantes (Leporini, 2004).

Consistência com relação ao comportamento: muitos *sites* apresentam certos recursos em algumas de suas interfaces, mas, sem qualquer motivo, não os mantêm em outras interfaces em que seriam pertinentes, ou se os mantêm, os trazem com denominações distintas. Por exemplo, em alguns *sites* de comércio eletrônico, como no “*submarino.com*”, é comum encontrar em algumas telas um mecanismo de busca com a opção de busca de acordo com algum critério, inexistente em outras partes onde um critério semelhante poderia ser aplicado. Nesse *site* na seção destinada à venda de CDs, existe uma “*busca por artista*”. Naturalmente, essa opção não caberia em todo o *site*, mas em algumas seções, como Livraria, poderia existir um mecanismo de busca semelhante, substituindo apenas o critério “artistas” por “autores”. Isso daria um comportamento padronizado às telas, uma boa prática de usabilidade seria não incluir esse critério de busca ou adaptar o mesmo nas telas do *site* em fosse pertinente.

No caso de uma pessoa com visão, isso é um ponto negativo; ela vai precisar de um tempo até perceber, através de uma análise visual, que o mecanismo de busca não existe em todas as seções. Mas esse problema se torna mais crítico quando o usuário for um deficiente visual. Como ele não consegue enxergar o *site*, ele terá que ficar percorrendo o mesmo com o leitor sem saber se o mecanismo foi omitido ou se simplesmente está em outra posição.

Consistência com relação às atualizações: deve-se estender o princípio de uniformidade a todas as interfaces do *site*. Contudo, diferentemente dos aplicativos não destinados à Web, os *sites* são dinâmicos; uma vez disponibilizados, continuam sofrendo infundáveis atualizações, nem sempre realizadas pela equipe que o desenvolveu. E essa atualização pode causar inconvenientes a alguns usuários, como por exemplo, as pessoas com deficiência visual.

Existem dois tipos de alterações que podem ser feitas nos *sites*. A primeira, mais comum, consiste apenas em atualizar o conteúdo sem modificar o *layout* da página, como ocorre nos jornais. Isso não causa estranheza ao deficiente. O segundo tipo, mais problemático, ocorre

quando o projeto do *site* é alterado. Isso leva o usuário cego a ter que reaprender o nome e a posição dos elementos da página. Embora eles não considerem isto um obstáculo, alguns deficientes que participaram dessa pesquisa (Coube, 2005), (Hildebrandt, 2005) e (Livramento, 2005) mencionaram que facilitaria se houvesse uma notificação quando uma nova versão do *site* entrasse em vigor. Uma entrevistada relatou que, quando a página de seu provedor foi modificada, ela e seu marido, também cego, ficaram sem saber o que estava acontecendo, em dúvida se era uma falha do programa ou um erro cometido por eles (Livramento, 2005). Uma sugestão é colocar uma identificação, na página, do número e da data da versão atual. Ou seja, é preciso adotar certos procedimentos com o objetivo de ajudar as equipes que virão a trabalhar na atualização do *site* a preservar a consistência de suas telas.

Consistência com ao uso de ícones, imagens e outros elementos: a diretriz de acessibilidade 1.1 do W3C (W3C, 2008) determina que toda informação não textual seja acompanhada de um texto. Para isso deve-se sempre colocar o atributo “*alt*” (*alternate text*), cujo conteúdo pode ser lido por um leitor de tela. Esse atributo fornece um texto alternativo associado a uma imagem; em grande parte dos navegadores, o conteúdo textual do “*alt*” é exibido para o usuário comum em duas ocasiões: quando ele passa o *mouse* por cima da imagem ou quando a página está sendo carregada e uma imagem ainda não foi carregada, o bloco com o conteúdo textual do “*alt*” aparece até que a figura o substitua.

Se os textos desses atributos de imagens forem apropriadamente escritos, podem fornecer informações úteis sobre o significado das imagens ao serem lidos pelos leitores de tela (Hanson, 2004) e (Harrison, 2005). No entanto, quando um *site* contém uma imagem sem o atributo “*alt*”, essa imagem pode ser detectada ou não, dependendo do programa leitor usado. Alguns leitores não acusam nada, outros indicam a existência de uma “imagem”, mas sem fornecer nenhuma informação para o deficiente do porquê da figura (Queiroz, 2008).

Um ícone é um pictograma que indica visualmente a existência de uma aplicação ou representa uma função, um objeto, uma ação, uma propriedade, um *link* ou qualquer outro conceito (Foley, 1997). Logo, em termos de acessibilidade, ele deve ser projetado de acordo com o que determina as imagens. É comum os *sites* usarem diversos nomes ou ícones para um mesmo assunto, ação ou objeto. Por exemplo, alguns *sites* usam imagens diferentes de carrinhos para indicar a mesma ação de “comprar”, confundindo assim o usuário com visão; o mais apropriado seria usar um único ícone. Da mesma forma, com relação à acessibilidade, deve-se procurar usar o mesmo texto para o atributo “*alt*” de modo a permitir que o deficiente visual identifique imediatamente o elemento sempre pelo mesmo nome.

Outro problema grave e freqüente consiste na adoção de ícones visualmente parecidos, com diferenças sutis, para funções diferentes. Naturalmente, se o usuário prestar atenção, conseguirá perceber uma pequena diferença entre as duas imagens, mas mesmo assim, pode se confundir. Igualmente, adotar a prática de escolher nomes parecidos para o atributo “*alt*”, apesar de alinhada com as diretrizes de acessibilidade, poderá trazer problemas de usabilidade, pois o usuário terá que prestar atenção no nome falado para perceber a diferença.

Outra boa prática de usabilidade é identificar o texto que descreve um *link* de forma informativa e útil (Harrison, 2004); esse texto é captado pelo leitor de tela e é ele que permite que o deficiente saiba para que serve o *link*. Logo, identificar *links* simplesmente com frases como “*clique aqui*” ou “*próximo*” constitui um obstáculo para usuários que dependem de leitores de voz, como é o caso dos deficientes visuais (Queiros, 2008). Semelhantemente aos ícones, os nomes escolhidos devem ser padronizados ao longo de todo o *site*.

Uso consistente e adequado da combinação de cores: a cor, elemento fundamental em qualquer processo de comunicação, interfere nos sentidos, emoções e intelecto da pessoa; pode-se usá-la deliberadamente para atingir um determinado objetivo. Por isso, recomenda-se atenção na escolha da combinação de cores. Cores apropriadas contribuem para a rápida e correta assimilação da informação. O impacto na eficácia da interface depende da relevância da

cor para a execução de uma tarefa, bem como da situação e do ambiente em que se executa essa tarefa (Jackson, 1994) e (Marcus, 1988).

O projetista de interface deve precaver-se contra a adoção de cores inadequadas, a fim de evitar levar ao usuário uma informação incompleta, ambígua ou ininteligível. Nota-se que em alguns *sites*, a escolha das cores se faz aparentemente de forma impensada, sem preocupação com a consistência: a cor existe apenas como elemento estético, oferecendo pouca ajuda para o usuário. Essa negligência muitas vezes torna a navegação confusa.

A cor também deve ser cautelosamente escolhida de forma a facilitar a interação dos deficientes visuais que ainda possuem algum resíduo de visão. Isso porque os deficientes visuais não usam o *mouse* para navegar, já que esse dispositivo exige coordenação visual (mira) (Queiroz, 2008). Eles utilizam a tecla *tab* e combinações de teclas, chamadas teclas de atalho, também utilizadas por pessoas com visão normal para agilizar tarefas (Leporini, 2004). O problema é que essa combinação de teclas, em muitas situações requer mais um aprendizado, o que faz com que deficientes visuais que ainda mantenham alguma visão residual prefiram usar esse resíduo de visão. Por isso, deve-se selecionar de forma criteriosa a cor de fundo, para que o contraste entre fundo e texto facilite a leitura (Hanson, 2004).

B) Feedback

O *feedback* é necessário em todo tipo de comunicação. Nas conversas, as pessoas dão constantemente *feedbacks* a seus interlocutores, por meio de palavras, expressões faciais, corporais, etc. Igualmente, a interação entre o usuário e o computador requer *feedback*, só que nesse caso, ele precisa ser planejado e programado (Foley, 1997).

Bons exemplos de *feedbacks* podem ser achados nos *sites* que mostram para o usuário a seção em que se encontra. Em geral isso é feito através de alguma mudança da apresentação visual que destaque a identificação da seção corrente de modo que o usuário possa localizar rapidamente dentro do *site*. O problema que o *feedback* fornecidos para realçar uma identificação de página é útil apenas para pessoas com visão, pois ele se apóia essencialmente na capacidade visual do observador. Uma forma de fornecer esse *feedback* através do leitor de tela é com o uso da *tag* (etiqueta) “*title*”. Ao se entrar em uma página através do leitor, a informação lida é a *tag* <*title*>. Logo, uma boa prática de indicar ao deficiente visual em que seção do *site* ele se encontra consiste em pôr o nome da seção no *tag* “*title*” (Soares, 2008).

Outro exemplo de *feedback* bem explorado, sobretudo nos buscadores de informação, é o realce dos *links*. Ao passar o *mouse* sobre um *link*, além da mudança do ícone para uma “mãozinha” que indica ser um *link* (recurso do próprio navegador), o *link* assume uma cor diferente. Apesar de um usuário sem visão não notar esse *feedback* visual, ele percebe que se trata de um *link*, porque o leitor de tela indica; mas para ser eficiente, o nome do *link* deve ser bem apropriadamente escolhido.

Uma boa prática de *feedbacks* é indicar os campos de preenchimento obrigatório nos formulários. Em geral isso é feito através do uso de uma formatação de cor ou fonte diferente. O problema é que o programa leitor de tela não reconhece essa mudança de cor. Uma alternativa seria usar um asterisco, mas usuários de leitores de tela comumente desabilitam a pontuação. O ideal é indicar com uma letra que represente a palavra “obrigatório” (Harrison, 2004). Ainda com relação às falhas de *feedback* para os deficientes visuais, existe uma muito comum: algumas interfaces são construídas de tal modo que, quando um deficiente visual encontra um campo a ser preenchido, nenhuma indicação sonora lhe é passada a fim de explicar o que deve ser feito; o usuário apenas ouve um aviso padrão do leitor: “*caixa de edição*”. Deve-se sempre então usar o *tag* “*label*”, pois ele permite que se coloque um texto, que é lido pelo leitor, informando o que deve ser preenchido (Queiroz, 2008).

C) Níveis de Habilidade e Comportamento Humano

O processo de produção de um *software* depende do fator social. Os *softwares* são desenvolvidos dentro de um contexto social. Logo, é fundamental levar em conta o tipo de

formação e o meio social do conjunto de usuários do sistema. Para criar produtos efetivamente úteis para um grupo de pessoas, faz-se necessário identificar e compreender esse grupo: procurar saber como as pessoas realizam suas tarefas, o que pensam do seu ambiente de trabalho e a que tipo de imposições e limitações elas estão sujeitas (Ferreira, 2003).

Variações de personalidade e comportamento também influem no êxito de um sistema. Na maioria dos casos, a personalidade relaciona-se ao estilo cognitivo da pessoa, logo, uma interface usada por duas pessoas com formações semelhantes, mas personalidades diferentes pode ser amigável para uma e não para outra.

Os modelos mentais são representações existentes na mente das pessoas, usadas para explicar, simular, prever ou controlar objetos no mundo. Essas representações são externalizadas através dos modelos conceituais. A elaboração de um modelo conceitual do usuário depende da personalidade, do conhecimento e da experiência prévia de cada pessoa e baseia-se nas expectativas, objetivos e compreensão do usuário a respeito do sistema. Um usuário cria seu modelo a partir de “objetos” que já conhece no seu dia-a-dia; ele procura relacionar os elementos computacionais com esses “objetos” familiares, em uma tentativa de melhor entender a máquina (Pressman, 2004).

Usuários com deficiências, ao acessarem um sistema, usam um ambiente bem diferente das pessoas sem deficiências. Eles relacionam os elementos computacionais com “objetos” de seu dia-a-dia, desenvolvidos para suprir suas necessidades. Além disso, pessoas com deficiências desenvolvem habilidades especiais, como uma excelente audição; raramente eles ficam aguardando de forma passiva a saída falada; eles se movimentam pelas páginas usando combinações complexas de teclas. Através desse processo eles criam seus modelos e tentam navegar de forma lógica pelas páginas.

Esses fatos, por acrescentarem mais dificuldade na interação com *sites* (Hanson, 2004), acabam influenciando seus modelos conceituais (Takagi, 2004). Por isso, o ideal é projetar a interface visando acomodar diferentes personalidades ou, uma personalidade típica das classes de usuários finais. Uma das maneiras de se acomodar personalidades distintas é projetando-se sistemas em conformidade com as diretrizes de acessibilidade. O critério de acessibilidade indica a facilidade com que alguma coisa pode ser utilizada, visitada ou acedida de um modo geral por todas as pessoas, inclusive por aquelas que possuem algum tipo de deficiência. Para promover a acessibilidade, usa-se certas facilidades que ajudam a retirar os obstáculos ou barreiras de acessibilidade do ambiente, conseguindo que estas pessoas realizem a mesma ação que uma pessoa sem nenhum tipo de deficiência (Enap, 2007).

Interfaces acessíveis facilitam também o acesso tanto por pessoas com muita experiência como por aqueles que a utilizam pela primeira vez. Menus, formulários e *prompts* são excelentes ferramentas para os principiantes, pois os ajudam a entender a linguagem do sistema. Bom exemplo disso encontra-se nos caixas eletrônicas de bancos, dotados de interfaces tipicamente orientadas para *prompts* e formulários com a finalidade de torná-los acessíveis às pessoas sem experiência computacional prévia.

Tais facilidades muitas vezes revelam-se lentas para os mais experientes, que em geral preferem usar teclas de função e comandos por meio do teclado (aceleradores). Assim, uma boa interface deve permitir ambas as formas de entrada de comandos (Foley, 1997).

Um elemento importante é o *Language tag*, atributo que indica o idioma utilizado em um documento ou em uma parte de um documento. Leitores de tela detectam automaticamente o *language tag* e escolhem o sintetizador de voz adequado para a leitura nesse trecho.

D) Percepção Humana

Para o usuário com visão, a interface estabelece uma interação visual com o computador; o cérebro, por intermédio do olho recebe e interpreta a informação com base no tamanho, no formato, na cor e em outras características. A especificação adequada da comunicação visual constitui-se o elemento-chave para a obtenção de uma interface amigável (Pressman, 2004).

Apesar da forte tendência de utilizar elementos gráficos no projeto de telas, ainda existe muita informação na forma de texto. A leitura permanece essencial em vários sistemas. O tamanho do texto, o tipo de fonte, o comprimento das linhas, as letras maiúsculas/minúsculas, a localização e a cor são fatores que influem diretamente na percepção da informação. No projeto de *sites*, cumpre lembrar que o modo de percepção da informação difere de pessoa para pessoa. Por exemplo, os mais velhos costumam sentir dificuldade de enxergar fontes pequenas e em geral preferem as cores pastel. Deve-se levar isso em conta.

A sequencialidade é uma das barreiras encontradas por um deficiente visual ao percorrer um *site* com um leitor de tela (ou através de um programa que amplifica a interface): o usuário só consegue acessar uma porção limitada da tela e perde, assim, a noção do contexto geral da página corrente (Leporini, 2004). Embora esses usuários percam a noção de contexto, eles acabam desenvolvendo certas habilidades, como uma excelente audição e memória. Assim, ao acessarem a Web através de leitor de tela, eles não precisam ouvir todas as palavras da página; eles escutam apenas o suficiente para determinar para onde desejam ir. Assim como as pessoas com visão rapidamente percorrem uma página com os olhos, esses usuários cegos o fazem com os ouvidos (Harrison, 2005). Como os leitores de tela processam o conteúdo de um *site* de forma bem diferente do processamento em uma leitura visual e seu uso demanda treinamento e experiência, é importante então que as interfaces sejam simples.

Por essa razão, as *tags* da linguagem HTML, que podem ser identificados pelo leitor de tela, por permitirem que o leitor forneça informações detalhadas sobre a estrutura do *site*, se tornam um elemento importante para o processo de percepção da informação por um cego (Leporini, 2004). Assim, ao se projetar uma interface para ser acessada também por deficientes visuais, deve-se tomar cuidado, pois muitas características visuais, como negrito, sublinhado, itálico, estilo de fontes etc. não são identificadas pelo leitor de tela.

A percepção do sistema e os modelos mentais das pessoas são influenciados por suas experiências; como é pouco provável que pessoas comuns tenham, ao navegar pela Internet, uma experiência semelhante à de pessoas com deficiências, os modelos de pessoas com deficiências tendem a ser distintos dos modelos das demais pessoas (Takagi, 2004).

Outro fator que deve ser levado em conta é o fato da maioria das pessoas, diante de um problema, aplica uma série de heurísticas (diretrizes, regras e estratégias) baseadas na compreensão de situações do mundo real. Assim, a boa interface permite ao usuário interagir com ela por meio de heurísticas coerentes, sólidas. E, no caso de deficientes visuais, essas heurísticas devem ser baseadas em suas peculiaridades.

E) Metáforas

Metáfora, do grego *metaphorá*, “mudança, transposição, transferência”, consiste na técnica de substituir um signo por outro para tornar a comunicação mais efetiva. Em retórica, quer dizer transposição do sentido próprio ao figurado, ou seja, designação de um objeto ou qualidade mediante uma palavra que designa outro objeto ou qualidade que tem com o primeiro uma relação de semelhança. Bons exemplos de metáfora presente em sistemas encontra-se na “mesa de trabalho” (*desktop*), adotada em interfaces do tipo *Macintosh* e *Windows* e o carrinho de compras, amplamente usado em *sites* comerciais (Marcus, 1998).

No que diz respeito à acessibilidade, as metáforas merecem um estudo especial; objetos do cotidiano de pessoas com visão são distintos dos elementos dos deficientes. Por exemplo, de acordo com o Prof. Hercen, que nasceu cego, a metáfora janela (*windows*), que indica a visualização de uma área de trabalho, nada representa para um cego (Hilderbrandt, 2005). Logo, dependendo da situação, uma metáfora aparentemente óbvia, pode nada representar.

F) Minimização de Carga de Memória

No projeto das interfaces, deve-se procurar reduzir ao mínimo o esforço exigido do usuário para memorizar suas características. Por isso recomenda-se que utilizem comandos mnemônicos e, sempre que possível, permitam a manipulação de ícones (Foley, 1997).

Os signos (ícones, comandos textuais, etc.), como elementos essenciais de uma tela, merecem uma produção cuidadosa. É preciso, durante todo o processo de desenvolvimento, constante preocupação com a escolha e com o projeto dos signos, para que não gerem dúvidas e tornem a interface o mais poderosa possível. Os ícones escolhidos devem expressar com precisão seus objetivos. Deve-se evitar dar nomes longos aos comandos, a fim de facilitar sua memorização e de ocupar pouco espaço de tela. O ideal é que o usuário não encontre situações que o levem a hesitar sobre o significado de alguma coisa (Pressman, 2004).

É importante considerar o fato que os deficientes visuais se baseiam muito em sua capacidade de “memorizar” para navegar pelos *sites* (Queiroz, 2008), uma vez que eles precisam decorar uma série de informações (comandos, sequências de operações, alternativas, etc.). A memória humana, extremamente complexa, compõe-se de duas partes: memória de curta duração (*short-term memory* - STM) e memória de longa duração (*long-term memory* - LTM) (Klatzky, 1980). A memória de curta duração armazena os *inputs* sensoriais (visuais, auditivos e táteis), e a de longa duração o conhecimento. A interface de um bom sistema não exige do usuário o uso excessivo de nenhuma das memórias (Pressman, 2004).

Com a memorização, a leitura de tabelas oferece uma barreira ao deficiente visual. Quando ele usa um leitor de tela, as tabelas são, necessariamente, lidas horizontalmente, linha após linha. Como o ele não pode visualizar a tabela inteira e depende, assim, de sua memória para saber em que posições se encontram as diferentes colunas, seria preferível que o leitor de tela lesse o cabeçalho de cada coluna (a primeira célula de cada coluna) antes do dado de cada célula (Livramento, 2005). O HTML oferece recursos que permitem distinguir os cabeçalhos das demais células, abrindo caminho para esse tipo de leitura, desde que essa distinção seja aplicada corretamente no código-fonte. Logo, identificar dos nomes de cada coluna e linha por meio do *tag* “*th*” (*table header*) consiste de uma boa prática de usabilidade.

G) Eficiência no Diálogo, no Movimento e nos Pensamentos

No projeto do *layout* da tela, deve-se procurar minimizar a distância que o *mouse* necessita percorrer entre dois cliques. Elementos usados em geral um logo a seguir do outro, devem ser dispostos próximos. Por exemplo, os ícones de “*recortar*”, “*copiar*” e “*colar*” devem estar próximos na tela de modo que o usuário não perca tempo procurando.

Já com relação aos deficientes visuais, os leitores de tela fornecem funções especiais que permitem ao usuário pular diretamente para os *tags* de cabeçalho (*h1*, *h2*...), elemento chave na estruturação de *sites* de fácil navegação. Com o uso dessa *tag*, os deficientes visuais podem navegar pelos títulos, de forma a obterem uma noção geral da página (Takagi, 2004).

Segundo Maria Luzia do Livramento, *sites* estruturados em parágrafos promovem uma navegação mais objetiva. O deficiente visual gosta de ter a opção de navegar pulando de parágrafo em parágrafo, só se detendo naqueles que considera ser mais importantes; Usuários experientes conseguem identificar rapidamente se desejam ou não prosseguir na leitura de um parágrafo ou pular para o seguinte, aproximando assim seu método de leitura àquele de uma pessoa com visão normal (Livramento, 2005).

Para isso, é fundamental adotar a prática de sinalizar cada parágrafo no código HTML através do *tag* “*P*” (um *tag* de bloco, que marca o início e o final de uma formatação) em vez do *tag* “*br*” (um *tag* único que só indica a ocorrência de um elemento – no caso, a quebra de linha – naquela posição, sem caracterizar o tipo de texto que o precede ou o sucede).

Um dos problemas para quem acessa a Web através de leitores de tela é o fato de a navegação nos *links* ser seqüencial (Leporini, 2004). Essa navegação pode se tornar lenta, pois quando se deseja navegar para uma parte já lida da página, por exemplo, voltar para um *link* posicionado à esquerda do corrente, deve-se pular todos os *links* de navegação para se recomençar a leitura da página e finalmente chegar ao conteúdo desejado. Recomenda-se que os *sites* forneçam recursos que permitam aos usuários pular os *links* repetitivos, acelerando a interação (Takagi, 2004). Para viabilizar uma navegação rápida, devem ser usados “*links de salto*” (*skip links*) de

modo a permitir que os usuários que acessam o *site* com um leitor de tela, pulem os *links* repetitivos e possam ir direto para algum conteúdo principal (Harrison, 2004). Os *links* de salto não são visíveis se o *site* for exibido em um navegador comum, e só são de fato úteis quando o *site* está sendo acessado por um leitor de tela (Takagi, 2004).

H) Classificação Funcional dos Comandos

O menu fornece uma lista de opções e oferece ao usuário a possibilidade de acessar funções e utilitários pouco solicitados. Clicando-se ou passando-se o mouse por cima de um determinado item do menu, pode aparecer um submenu abaixo dele (menu *pull-down* ou menu hierárquico). O menu *pull-down* apresenta a vantagem de só ser chamado quando necessário, economizando assim espaço de tela. Além disso, dá objetividade à tela: itens demais confundem o usuário.

Menu, é palavra francesa e significa cardápio. Normalmente, nos cardápios, as opções são agrupadas abaixo de um título (aves, carnes, massas, etc.). Da mesma forma, nos menus das interfaces, o projetista estabelece um critério para ordenar as opções. Por exemplo, debaixo da opção “Editar” do menu, devem aparecer todos comandos relacionados à tarefa de edição.

Mas o problema é que o recurso de menus *pull-down* só funciona bem se o usuário estiver usando um *mouse*; frequentemente ele causa problemas àqueles que precisam navegar via teclado. Logo, ao mesmo tempo que esse recurso promove a usabilidade, ele dificulta a acessibilidade, pois os submenus, muitas vezes, não são acessíveis quando a navegação é feita pelo teclado. E em um *site* acessível deve-se garantir que todo seu conteúdo possa ser acessado pelo *mouse* e pelo teclado. Para isso, os menus *pull-down* precisam ser projetados de modo a poderem ser acessados via teclado. A navegação via teclado beneficia não só os deficientes visuais, mas todos os usuários que possuem dificuldades ou impossibilidade de usar o *mouse*, como pessoas com problemas de coordenação motora, mal de *parkinson*, paralisia cerebral etc. (Torres, 2008).

Um técnica para acessibilizar menus *pull-down* pode ser através do uso de *JavaScript*, que torna o submenu visível mesmo quando acessado pelo teclado. Isso porque, na maioria das situações onde esse menu ficaria inacessível via teclado, o usuário tem o *JavaScript* habilitado no navegador (Torres, 2008). Mas como alguns *Javascripts* e *VBscripts* as vezes criam dificuldades, dependendo do evento a ser acionado, o *script* pode ser substituído por algum que aceite o teclado, utilizando-se tratamentos redundantes para os eventos do *mouse*. Como exemplo, poderia usar "*onmousedown*" com "*onkeydown*" (Queiroz, 2008).

Deve-se levar em conta também a quantidade de itens nos menus *pull-down*. Um estudo do psicólogo George A. Miller (Miller, 1956) revelou que as pessoas, ao lidar com quantidade de informações sentem mais facilidade quando o número de elementos não excede a sete mais ou menos dois. Segundo essa teoria, os menus devem conter no máximo nove itens.

I) Manipulação Direta

A manipulação direta, apesar de não trazer vantagens para o deficiente visual, deve ser uma preocupação constante do projetista. A vantagem da manipulação direta reside em permitir que o usuário se sinta no controle dos objetos representados no computador. O objeto manipulado deve permanecer visível durante a operação, bem como o efeito dessa operação deve se tornar imediatamente visível (Ferreira, 2003).

Como os deficientes visuais navegam nas páginas através das teclas, sem usar o *mouse*, eles pouco tiram vantagens da manipulação direta. Como é necessário um aprendizado para usar bem as teclas, os deficientes que ainda possuem algum resíduo de visão preferem manipular diretamente os objetos. Como a visão destes é limitada deve-se selecionar de forma criteriosa as cores usadas nos elementos de forma a melhorar sua visão (Hanson, 2004).

J) Exibição Exclusiva de Informação Relevante

Deve-se mostrar só a informação relevante ao contexto corrente, a fim de facilitar sua assimilação. Isto poupa o usuário de ter de explorar vários menus, textos e imagens até encontrar

o que precisa para executar sua tarefa (Pressman, 2004). Com relação à acessibilidade, uma boa prática consiste em se colocar *links* de saltos (*links* internos) para o conteúdo principal, sistema de busca e menu interno. Esses *links* internos, quando aplicados no início das páginas ou em outros lugares estratégicos podem aumentar a usabilidade dos *sites* para quem usa apenas o teclado e evita que esses usuários tenham que ler uma série de informações não relevantes (Queiroz, 2008).

K) Uso de Rótulos, Abreviações e Mensagens Claros

Deve-se tomar cuidado para adotar rótulos consistentes, abreviações padronizadas, cores previsíveis e evitar usar signos arbitrários para representar itens novos; é indispensável critério na escolha desses elementos para evitar dúvidas quanto ao seu significado (Ferreira, 2003). Assim, é importante lembrar que, muitas características visuais, como negrito, sublinhado, itálico, estilo de fontes etc. não são detectadas pelo leitor, e se tornam não perceptíveis; já outros elementos invisíveis, como *labels*, títulos em *links* e atributos “*alt*” em imagens, podem ser utilizadas para realçar a informação (Leporini, 2004). Os textos usados nesses atributos nos elementos de tela deve ser tal que torne a mensagem passada útil. Textos do tipo “*clique aqui*” ou “ *siga*” nada informam sobre a finalidade dos elementos em questão.

Da mesma forma, as mensagens geradas devem ser claras e elucidativas. O projetista deve criar mensagens que não deixem o usuário confuso nem o façam sentir culpado por ter errado.

L) Uso Adequado de Janelas

Ao clicar em um *link* ou em um botão, deve-se evitar que sejam abertas novas janelas. Essa prática, além de poluir a tela e cobrir a janela que o usuário efetivamente está trabalhando, faz também com que o botão “Voltar” seja desativado, uma vez que novas janelas não herdaram o histórico da janela original (Nielsen, 2006). A exceção a essa regra se aplica nos documentos que devem ser apresentados que não são tratados pelos navegadores, como arquivos PDFs, *word*, *excel* entre outros. Isso porque, quando as pessoas terminam de usar tais arquivos, em geral elas clicam na caixa de fechamento em vez do botão “Voltar” (Nielsen, 2006). E no caso da necessidade de se abrir várias janelas, pelo menos uma parte de cada janela deve aparecer na tela. Um outro tipo de janela que deve ser evitado são as janelas *pop-up*. Além de irritarem os usuários, elas distorcem o que as pessoas esperam da Web, que é ter a informação exibida na janela principal do navegador (Nielsen, 2006).

Usuários com deficiências também encontram dificuldade para gerenciar janelas extras. Pessoas com dificuldades de coordenação motora certamente não querem clicar em caixas de fechamento indesejáveis. E usuários com problemas de visão provavelmente nem notarão que uma nova janela foi aberta.

É preciso procurar tirar o máximo de proveito do espaço da tela. Deve-se, portanto, evitar o excesso de interfaces de *sites* com propagandas, *banners* e afins. Além de ocuparem muito espaço de tela, os *sites* costumam apresentar elementos repetitivos, comuns a todas as páginas, como *banners*, menus entre outros, apresentam um obstáculo ao acesso de deficientes visuais: quando esses elementos são detectados antes do conteúdo principal, a leitura se torna lenta e cansativa (Leporini, 2004), (Hilderbrandt, 2005) e (Livramento, 2005).

Também é recomendado que se evite o uso de GIFs animadas, que além de não terem utilidade para os cegos, também dificultam a leitura dos deficientes visuais parciais (Hanson, 2004). De acordo com alguns deficientes, eventualmente, ao se acessar um *site* através do programa leitor, encontra-se uma mensagem acusando, quando não é ainda o fim. Essa mensagem em geral é provocada pelo uso inadequado de imagens animadas ou *banners* (Ferreira, 2005) e (Cerqueira, 2005).

M) Projeto Independente da Resolução do Monitor

A questão da resolução do monitor interfere na usabilidade dos *sites*. Nas interfaces tradicionais, o projetista sabe para que ambiente está trabalhando. Assim, pode controlar cada ponto (*pixel*) da tela que o usuário verá e assegurar-se de que todo elemento projetado vai

aparecer sempre da mesma forma, independentemente da resolução do monitor. Isto possibilita a escolha adequada dos aspectos visuais da interface, como tipo e tamanho de fonte.

Na Web, porém, como o usuário pode acessá-la de diversas maneiras, o projetista não tem esse controle. Um dos princípios básicos para o bom planejamento na construção de *sites* independentes da resolução consiste em definir a aparência dos componentes por meio de porcentagens do espaço disponível, em vez de projetar os elementos com um tamanho fixo, determinado pelo número de *pixels* (Nielsen, 2006). Não se pode esquecer que muitas pessoas e organizações, sobretudo nos países menos desenvolvidos, possuem monitores de baixa resolução. O projeto independente da resolução é a chave para acessibilidade, pois permite que o *site* possa ser acessado em qualquer dispositivo, inclusive nos móveis, como celulares.

6.2. Requisitos Relacionados à Entrada de Dados

Consome-se muito tempo de trabalho na escolha de comandos, na digitação de dados e em outros *inputs*. Quanto à entrada de dados, recomendam-se alguns procedimentos para tornar as interfaces mais poderosas e promover a acessibilidade (Pressman, 2004). Ao entrarem com dados, os usuários com deficiência visual, principalmente as pessoas com deficiência acentuada ou total, não utilizam o *mouse* e sim o teclado, que se tornou uma solução facilitadora capaz de ser usada por qualquer cego devido a uma norma internacional de datilografia: todos teclados produzidos em conformidade com as normas técnicas regionais possuem, na parte inferior das teclas J e F (no lado alfanumérico) e 5 (no lado numérico), um alto relevo com a forma de um ponto ou traço que funciona como guia para o cego posicionar as mãos, assim como faz uma pessoa com treinamento em datilografia (Queiroz, 2008).

A) Mecanismos de Ajuda

Deve-se fornecer informação de ajuda para toda ação de entrada (Foley, 1997), bem como dicas quando o *mouse* passar sobre algum item. Isto agiliza o trabalho, pois possibilita ao usuário saber para que serve certo item sem precisar recorrer ao sistema de ajuda textual.

Como exemplo de mecanismos de ajuda citam-se a indicação de campos obrigatórios. Essa prática faz com que se poupe tempo de, após enviar o formulário, ter que se retornar para fornecer alguma informação faltante. Em geral isso é feito com uma cor diferente para os campos obrigatórios ou com o uso de um asterisco. Mas é importante considerar o fato que os leitores de tela não reconhecem mudanças de cor e consideram o asterisco uma pontuação. Logo, uma boa solução é o uso da palavra “obrigatório” (Harrison, 2004).

Uma das formas com que as pessoas enviam dados pela Internet é através do preenchimento de formulários. Como o deficiente visual navega pelos campos dos formulários usando a tecla *tab*, para facilitar a entrada de dados, os campos a serem preenchidos e os botões de busca, se importantes, devem, preferencialmente, ficar localizados na parte superior da página (Leporini, 2004).

Quando uma pessoa navega com um leitor de tela encontra um campo a ser preenchido, é importante que seja indicado qual o dado deve preencher. Isto é feito através do *tag* “*label*” que coloca o texto que será lido pelo leitor (Queiroz, 2008). Esse *tag* permite também atribuir uma tecla de acesso rápido a cada campo do formulário, além de ampliar a área de clique para campos de caixa de seleção e “*radio button*”, o que facilita o preenchimento do formulário por pessoas com deficiência visual branda. Deve-se evitar o uso de um valor *default* no campo, pois mesmo lido pelo leitor, exige que o usuário apague o valor (Harrison, 2004).

Uma outra forma de se enviar dados é através dos botões de comando, como o botão de “enviar”, ou de “submeter”; estes não precisam do “*label*”, pois podem ser lidos através do atributo “*value*”; no entanto, deve-se evitar colocar palavras do tipo “clique aqui” ou “siga” nesse atributo, pois elas nada indicam sobre a finalidade do botão. Se o botão possuir uma imagem no lugar de um texto, basta colocar o atributo “*alt*” (Queiroz, 2008).

B) Prevenção de Erros

Uma boa interface procura evitar que seus usuários cometam erros. O bom projeto não permite que o usuário escolha uma opção inválida e, só depois, receba a mensagem de erro.

Deve-se desabilitar ou inibir os itens do menu que o usuário não pode selecionar no estado corrente do sistema. Por exemplo, se o usuário não tiver selecionado um objeto da tela, a interface deve impedir a seleção das opções “Recortar” e “Copiar”. Essa prática também é importante quando se pensa em usuários que acessam um sistema através de um leitor de tela. Quando um item está desabilitado o leitor sonoriza: “nome do item em questão desabilitado”.

Um bom projeto de interface deve sempre prestar ao usuário informações que contribuam para evitar erros. Por exemplo, se houver limite de quantidade ou de tipo de caracteres, o usuário deve obter essa informação antes de digitar o texto; os campos de preenchimento obrigatório e também aqueles que só aceitam números devem ser indicados e é importante informar o número mínimo e máximo de caracteres para a senha.

A boa interface minimiza o número de ações necessárias para a entrada de dados, ou seja, exige do usuário o mínimo de digitação. Se o preenchimento de cadastros costuma ser inibidor para qualquer usuário, para pessoas com deficiências visuais ele representa um obstáculo maior, podendo até levar ao abandono do *site*. O simples ato de restringir o acesso por senhas, que, por razões de segurança, não são soletradas pelos leitores de texto já dificulta o acesso dos usuários. Logo, tal prática só deve ser aplicada quando realmente necessária.

Existem maneiras de se minimizar a entrada de dados. Por exemplo, pode-se exibir automaticamente endereços eletrônicos anteriormente digitados, o que poupa o usuário de re-digitar a informação. Outra forma é exibir listas de opções (*listboxes*), por exemplo, para sexo e estado, em que o usuário simplesmente seleciona a(s) letra(s) correspondente(s) à sua escolha. Esse recurso não só reduz o número de ações de entrada como também a incidência de erros, por impedir que o usuário, por distração, digite uma opção inválida.

C) Tratamento de Erros

Erros são inevitáveis. A boa interface proporciona ao usuário algum meio de corrigir os erros rapidamente. Assim, além de aumentar a produtividade, encoraja o usuário a explorar o sistema, uma das melhores maneiras de aprender suas características. Basicamente, existem dois tipos de erros: o funcional e o sintático (Foley, 1997).

7. Conclusões

O presente trabalho teve por objetivo alinhar os RNF de usabilidade com as diretrizes de legislação de acessibilidade. Esse alinhamento tem por finalidade de auxiliar os profissionais de sistemas na identificação de possíveis problemas de acessibilidade e usabilidade que podem ser solucionados ou minimizados durante o processo de definição dos requisitos e assim, facilitar a interação de deficientes visuais com a Internet e garantir *sites* com conteúdo compreensível e navegável.

Foi feita uma pesquisa de campo no IBC, referência no Brasil em matéria de educação e reeducação de deficientes visuais, com a finalidade de se conhecer melhor os usuários com deficiências visuais. O motivo de se ter optado por deficientes visuais foi o fato de Internet muito contribuir para melhorar a qualidade de suas vidas, permitindo que eles criem novas formas de relacionamento, encontrem oportunidades de trabalho e formas alternativas de diversão. A pesquisa permitiu compreender como esses usuários percebem e interagem com os *sites* e identificou barreiras que eles precisam superar para acessar a informação. Com o conhecimento adquirido na pesquisa de campo e na literatura, foram identificados os tipos de imposições e limites a que esses usuários estão sujeitos, o que possibilitou uma melhor compreensão de suas necessidades e habilidades especiais.

Feito isso, selecionou-se uma maneira de avaliar a usabilidade de *sites*, que foi a proposta pela taxonomia dos RNF de usabilidade (Ferreira, 2003). Essa taxonomia foi criada com a

finalidade de sistematizar o processo de definição desses requisitos e de reduzir os problemas de usabilidade e têm sido muito utilizada na análise de usabilidade de diversos *sites* e pode auxiliar os projetistas na etapa inicial de definição de requisitos. Posteriormente procurou-se alinhar a análise de usabilidade feita com base na taxonomia com as diretrizes de acessibilidade. Esse alinhamento teve por finalidade auxiliar os profissionais de sistemas de informação na identificação e na solução de possíveis de acessibilidade que podem ser minimizados durante o processo de definição dos requisitos. Com isso, percebeu-se que a obtenção de aplicativos Web acessíveis vai além de projetar páginas Web em conformidade com as diretrizes de acessibilidade; questões de usabilidade devem ser consideradas.

8. Referências Bibliográficas

- Acessibilidade Brasil. *Recursos de acessibilidade*. - <<http://www.acesso brasil.org.br/>>. Acesso em: 20/4/2008.
- Carter, J.A e Fourney, D.W. - “Techniques to assist in developing accessibility engineers” - *Proceedings of 9th international ACM SIGACCESS conference on Computers and accessibility* - Pg: 123-130. 2007.
- Enap – *Material do curso de “e-Mag - Modelo de Acessibilidade de Governo Eletrônico”* – ministrado pela Escola Nacional de Administração Pública. Jan/ 2007.
- Ferreira, S. B.L; Leite, J. C. S. P - Avaliação da usabilidade em sistemas de informação: o caso do sistema submarino. *Revista de Administração Contemporânea – RAC*, São paulo, v. 7, n. 2, Pg: 115-137. Abr./Jun. 2003.
- Ferreira, S.B.L; Chauvel, M.A e Ferreira, M.L. “e-Acessibilidade: Tornando Visível o Invisível” - *Revista de Práticas Administrativas* - Ano 3, v. 3, n 4 Pg: 5-15. Jan/fev/2007.
- Foley, J.D. et al. *Computer graphics: principles and practice*. Reading, Addison-Wesley, 1997.
- Hanson, V.L.-The User Experience: Designs and Adaptations-*ACM Conference Proceeding Series-Proceedings of International Cross-Disciplinary Workshop on Web*–Pg:1-11. 2004.
- IBC – Instituto Benjamim. *Acessibilidade*.-<<http://www.ibr.gov.br/>>. Acesso em: 25/4/2008.
- IBGE – *Censo* <<http://www.ibge.gov.br/home/presidencia/noticias/>> Acesso em 17/01/2008.
- Harrison, S. M. - Opening the eyes of those who can see to the world of those who can't: a case study - *Technical Symposium on Computer Science Education - Proceedings of the 36th SIGCSE technical symposium on Computer science education* – Pg: 22-26. 2005.
- Klatzky, R. L. *Human memory*. 2. ed. Nova York: W. H. Freeman. 1980.
- Leporini B., Andronico, P. & Buzzi, M. - Designing search engine user interfaces for the visually impaired - *ACM International Conference Proceeding Series - Proceedings of the international cross-disciplinary workshop on Web accessibility* – Pg: 57-66. 2004.
- Marcus, A.: “Color: A Tool for Computer graphics Communication” Tutorial - *Color in Computer Graphics* nº. 24- SIGGRAPH. 1987.
- Miller, G. A. The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review*, n. 63, Pg: 81-97, Washington, DC, Publisher-American Psychological Association. 1956.
- Moloney, K.P., Jacko, J.A., Vidakovic, B., Sainfort, F., Leonard, V. K. E Shi, B. – “Leveraging data complexity: Pupillary behavior of older adults with visual impairment during HCI” - *ACM Transactions on Computer-Human Interaction* – v. 13 , n. 3 Pg: 376-402. 2006.
- NBR 9050: *Associação Brasileira de Normas Técnicas*. Acessibilidade de Pessoas Portadoras de Deficiências a Edificações, Espaço, Mobiliário e Equipamento Urbanos. ABNT. RJ. 1994.
- Nevile, Liddy “Adaptability and accessibility: a new framework” - *Proceedings of the 19th conference of the computer-human interaction special interes group of Australia on Computer-human interaction* - Pg: 1-10. Nov/2005.
- Nicholl, A.R.J. “O Ambiente que Promove a Inclusão: Conceitos de Acessibilidade e Usabilidade”. *Revista Assentamentos Humanos*, Marília, v3, n. 2, Pg: 49-60. 2001.

Nielsen, J.; Loranger, H. *Prioritizing web usability*. Indianápolis - New Riders. 2006.
Petrie, H., Hamilton, F., King, N. & Pavan-Remote usability evaluations With disabled people - *Proceedings of SIGCHI conference on Human Factors in computing systems*. Canada. 2006.
Pressman, R. *Software Engineering- Practioner's Approach*—McGraw-Hill, Nova York. 2004.
Queiroz, M.A. – *Bengala Legal* - <<http://www.bengalalegal.com/>> - Acesso 2/4/2008.
Soares, H – *Usabilidade* -< <http://horaciosoaes.blogspot.com>> - Acesso 27/02/2008.
Takagi, H., Asakawa, C., Fukuda K. & Maeda J.: Accessibility designer: visualizing usability for the blind - *ACM SIGACCESS Conference on Assistive Technologies - Proceedings of the ACM SIGACCESS conference on Computers and accessibility*. Pg: 177-184. Atlanta. 2004.
Tangarife, T. & Mont'Alvão, C. - Estudo comparativo utilizando uma ferramenta de avaliação de acessibilidade para Web - *Proceedings of the 2005 Latin American conference on Human-computer interaction* – pg. 313-318 - México –2005.
Torres, B. *Acessibilidade* - <<http://brunotorres.net/>> - Acesso 27/02/2008.
W3C. *XHTML2 working group page*. <<http://www.w3.org/MarkUp/>>. Acesso em: 16/2/2008.

Entrevistas

Coube, José Elias - Professor de Informática do IBC(cego) – 31/06/2005.
Ferreira_B, Gerson F. – Coordenador Geral de Informática do IBC (cego) – 10/06/2005.
Hilderbrandt, Hercen – Professor de Informática do IBC (cego) – 29/06/2005.
Cerqueira, Maria de Fátima Carvalhal – ex aluna do IBC(cega) – 09/07/2005.
Souza, José Francisco –Diretor da Divisão de Reabilitação do IBC(10% de visão)–22/07/2005.
Livramento, M.L. – Revisora de Textos da Imprensa Braille do IBC (cega) – 05/07/2005.