

# Modernização de software desktop legado para suporte à acessibilidade para usuários com deficiência visual: um estudo de caso

Livia Cristina Gabos Martins, Bruno Elias Penteado, Márcia Mayumi Haga, Geysler Niclevicz da Silva

MStech, Bauru - SP

{livia.gabos, bruno.penteado, marcia.haga, geysler.silva}@mstech.com.br

**Abstract.** *Software systems tend to evolve over time, either by changes in functional requirements, such as changes in business rules, applicable laws, etc. as in non-functional requirements: new levels of performance, new technologies and platforms, etc. in order to remain useful to those who use them. This paper shows the modernization of a legacy software to meet accessibility requirements for the targeted group of users with visual impairments. The software under study is a desktop program to control the computer labs. In this case study a reengineering process was applied, exploring white box techniques, to create a new layer of service that interprets the commands and controls its interface, implemented in Adobe Flash technology. As a result, we could use all the features of the software on the Windows platform, by communicating a screen reader DOSVOX using MSAA - interface communication platform.*

**Resumo.** *Sistemas de software tendem a evoluir ao longo do tempo, seja por modificações em requisitos funcionais, tais como: mudanças em regras de negócio, leis aplicáveis, etc. quanto em requisitos não funcionais: novos patamares de desempenho, novas tecnologias e plataformas, etc., de modo a se manterem úteis a quem os usa. Este artigo mostra a modernização de um software legado para atender requisitos de acessibilidade direcionados para o grupo de usuários com deficiência visual. O software em estudo trata-se de um programa desktop para controle de laboratórios de informática. Neste estudo de caso foi aplicado um processo de reengenharia, explorando técnicas de caixa branca, para a criação de uma nova camada de serviço que interprete os comandos e controles da sua interface, implementados na tecnologia Adobe Flash. Como resultado, foi possível utilizar todos os recursos do software, na plataforma Windows, através da comunicação do leitor de tela DosVox, utilizando O MSAA - interface de comunicação da plataforma.*

## 1. Introdução

Nos últimos anos houve grandes investimentos para aumentar a inclusão digital em todo o país [Inclusão Digital, Programa Brasil Conectado, 2011]. Muitos programas têm como principal objetivo estimular principalmente nos jovens, a utilização do computador, facilitando sua entrada na sociedade digital. Essa inclusão digital também se dá para as pessoas com deficiências físicas, com base na Lei 8213/91 [Brasil, Lei nº

8.213, de 24 de julho de 1991], também chamada como *Lei de Cotas*, que prevê uma porcentagem de contratação para portadores de deficiências.

De acordo com dados da OMS (Organização Mundial da Saúde) [WHO, 2009] cerca de 314 milhões de pessoas são deficientes visuais em todo o mundo, dos quais 45 milhões são cegos. No Brasil, de acordo com os dados do IBGE (Instituto Brasileiro de Geografia e Estatística) do censo de 2000, 16,6 milhões de pessoas possuem algum grau de deficiência visual, sendo que quase 150 mil se declararam cegos [IBGE, 2005].

Neste contexto, foi utilizado para o escopo deste trabalho um software desktop para controle de laboratórios de informática amplamente usado em programas educacionais de governos (estaduais e municipais) chamado *blueControl*. Criado e mantido pela empresa de tecnologia para educação MStech [MStech, 2011] sua utilização por deficientes visuais se fazia necessária, para que atendessem a necessidade de seus usuários e de modo a se adequar às leis.

Neste trabalho são discutidas as técnicas empregadas para a modernização do software citado e os resultados obtidos. Na seção 2 é feita uma breve introdução ao conceito de acessibilidade em software. Na seção 3 é feita revisão dos conceitos de reengenharia usados no escopo deste trabalho. Na seção 4 é descrito o software *blueControl*, usado neste estudo de caso. Na seção 5 é descrito o método adotado para o problema da modernização do software para adequação à acessibilidade. Na seção 6 são descritos os resultados obtidos com a avaliação feita até o presente momento. Na seção 7 é feita a discussão sobre os resultados.

## 2. Acessibilidade

No contexto digital, acessibilidade significa que pessoas com diferentes tipos de deficiência podem entender, navegar e interagir com sistemas computacionais, e cujas diretrizes podem trazer benefícios até mesmo àqueles sem deficiência [W3C, 2005].

Voltado para o desenvolvimento Web acessível, a WAI (*Web Accessibility Initiative*), divisão da W3C, estabeleceu um padrão conhecido como WCAG 1.0 – Diretrizes para Acessibilidade do Conteúdo Web 1.0 (*Web Content Accessibility Guidelines*) [W3C, 1999], para desenvolvedores e designers Web. Este padrão define técnicas para que pessoas com deficiências, como incapacidade de ver, ouvir, ou se movimentar adequadamente – chamados de deficientes visuais, auditivos e motores, respectivamente – possam ler e entender as informações disponíveis na internet. Atualmente não existe, para nosso conhecimento, um padrão específico para implementação de acessibilidade para aplicações desktop

Os padrões descritos são voltados para ajudar na comunicação com as *tecnologias assistivas* - termo utilizado para identificar recursos e serviços que contribuem para proporcionar ou ampliar habilidades funcionais de pessoas com tipos diversos de deficiências [Sonza, 2008], como por exemplo: leitores e ampliadores de tela, utilizados por deficientes visuais; reconhecedores de fala, usados para acionar comandos por voz; dispositivos para entrada de dados alternativos ao teclado e mouse; dentre outros.

Neste estudo foi utilizado o ambiente DOSVOX [Borges, 2002] como tecnologia assistiva para o ambiente de teste. Ele pode ser utilizado apenas para o sistema operacional Windows, se comunicando com o usuário através de síntese de voz,

permitindo o uso de computadores por deficientes visuais. Ele é considerado um ambiente e não apenas um leitor de tela, devido a muitas características que ele possui, sem a necessidade de o usuário navegar pelo sistema operacional diretamente. Outras ferramentas de tecnologia assistiva, tais como NVDA [NVDA Project, 2010] e JAWS [Freedom Scientific, 2011] não foram utilizados neste estudo de caso, uma vez que não foram distribuídos junto com *blueControl*.

### 3. Reengenharia de software

Podemos definir informalmente um software legado como aquele que executa tarefas úteis para uma organização, mas que foi desenvolvido utilizando técnicas atualmente consideradas obsoletas [Ward e Bennett, 1995]. Organizações que utilizam um software legado não fazem atualizações devido ao custo e riscos envolvidos na substituição. Por sua constante utilização, não se encontram mais erros de grande relevância no sistema legado. Normalmente, tais softwares não possuem documentação ou, quando ela existe, está desatualizada. Isso se torna um problema grave, principalmente quando os autores do software não estão mais por perto para apoio de mudanças.

De acordo com Seacord, Plakosh e Lewis [2003], as atividades de evolução de um software podem ser categorizadas em 3 classes: (i) *manutenção*: pequenas alterações são feitas de modo incremental e iterativo (novas funcionalidades e correções de defeitos); (ii) *modernização*: modificações mais amplas e complexas que a manutenção, podendo haver reestruturação do sistema, porém conservando porções significantes do sistema existente; (iii) *substituição*: reconstrução do sistema a partir do zero.

No escopo deste trabalho a categoria de modernização foi a adotada, pois o software necessitou de grandes modificações apenas em parte de sua estrutura.

Ainda de acordo com os autores, dependendo do nível de entendimento exigido do sistema em questão, duas estratégias de modernização podem ser tomadas: *caixa-preta* ou *caixa-branca*. Na estratégia de *caixa-preta* é necessário ter o conhecimento do comportamento externo do sistema, por meio de suas entradas e respectivas saídas de dados. Esta estratégia também conhecida como *wrapping*. Na estratégia de caixa branca, é exigido o conhecimento do funcionamento interno do sistema legado, de modo mais amplo e profundo, podendo envolver até conhecimento do código-fonte usado. Esta estratégia também é conhecida por *reengenharia de software*.

Como será visto na seção 4, o conhecimento do funcionamento interno do software *blueControl* foi necessário para que sua adaptação à questão de acessibilidade fosse realizada. Deste modo, neste artigo será discutida a aplicação de metodologia de reengenharia em um software para recriação e implementação de acessibilidade em sua interface, de modo a torná-lo usável por portadores de deficiência visual.

### 4. O software *blueControl*

O software tratado no artigo é chamado *blueControl*, criado pela empresa de tecnologia para educação MStech. Trata-se de um software de gestão de laboratório de informática em escolas, responsável por controlar o acesso aos computadores dos alunos ou por membros da comunidade que usam os computadores da escola, por meio do computador do responsável pelo laboratório. Com ele, é possível disponibilizar acesso para utilização dos computadores do laboratório, bloquear softwares inadequados, gerenciar impressão de documentos, gerar relatório de ocorrências e de uso dos computadores,

entre outras funcionalidades. Ele é operado por um monitor técnico que gerencia todos os computadores ligados àquela rede e de onde podem ser executadas as funcionalidades descritas acima.

O *blueControl* é utilizado normalmente em programas educacionais que disponibilizam acesso a laboratórios de computadores para aulas, treinamento e acesso ao público em geral. Os responsáveis pelos laboratórios das escolas ou instituições em que eles vão atuar recebem treinamento sobre as suas funcionalidades e da utilização em seu cotidiano, como por exemplo, sobre como habilitar os computadores de uma sala para uma aula. Eles são contratados pelo governo obedecendo a *Lei de Cotas*.

A impossibilidade de usar o software *blueControl* por deficientes visuais acarreta na exclusão de tais pessoas das atividades cotidianas de controle dos laboratórios de informática, ficando afastadas ou aguardando uma resolução do problema por meio da ajuda de terceiros. O uso dessa ferramenta no dia-a-dia dos laboratórios mostrou que a maior parte de suas funcionalidades não atendia às necessidades dos deficientes visuais responsáveis pelos laboratórios, sendo necessário um reprojeto de sua interface com o usuário.

Quando o software *blueControl* foi projetado, não houve a preocupação em torná-lo acessível para pessoas portadores de deficiências visuais, tampouco foi aplicada uma técnica de engenharia de software que incentivasse a modernização do código fonte. Por isso, foram enfrentados problemas pela equipe de desenvolvimento para adaptá-lo.

## 5. Método utilizado

A aplicação *blueControl* foi originalmente implementada em 3 camadas: i) camada de negócio, com o uso da linguagem C#, da .Net Framework 2.0; ii) camada de persistência, com a base de dados SQL Server 2005, iii) camada de interface, usando a tecnologia Flash Action Script 2.0.

Normalmente, quando aplicadas práticas de acessibilidade no desenvolvimento de sites usando a tecnologia Flash, é possível que os leitores de tela consigam identificar as informações do site por causa da tecnologia MSAA (Microsoft Active Accessibility) [Microsoft, 2000]. O MSAA é uma tecnologia baseada em COM (Componente Object Model) existente apenas no sistema operacional Windows, que possibilita a comunicação entre as aplicações e o sistema operacional. Além da tecnologia MSAA, utilizada no Windows, existe a *IAccessible2* [Linux\_Foundation, 2009], uma nova API que complementa o trabalho da MSAA e que pode ser utilizada pelo ambiente Linux e Mac OS.

Assim, foi necessário introduzir uma nova camada de mensagens para criar uma comunicação entre o software *blueControl* e o sistema operacional Windows, capturando as informações da interface e disponibilizando para a tecnologia MSAA.

Como a camada de regras de negócio foi desenvolvida em C#, a função dessa nova camada de serviço seria passar as informações de acessibilidade da interface do usuário para o sistema operacional, por meio do MSAA, onde atuam os leitores de tela, conseguindo identificar as informações.

Para que houvesse essa comunicação, a interface do *blueControl* foi alterada para que fosse possível fazer a navegação por teclado - modo em que os deficiente

visuais mais utilizam o computador - possibilitando assim o reconhecimento do objeto ou componente que estava sob o foco da navegação. Para isso, foram utilizadas técnicas de reengenharia para tornar possível modificar toda a estrutura da sua interface, sem que fossem perdidas regras de negócio e outras informações importantes contidas apenas no seu código. Utilizando a técnica de engenharia reversa foi possível recriar toda a interface, adicionando as mudanças relacionadas à acessibilidade em conjunto.

A sinalização entre a camada de interface e de negócio foi feita através do comando *fscommand* do Action Script [Adobe, 2010], que permite que o arquivo SWF (formato do arquivo resultado do Flash) se comunique com a base que o hospeda. Desta maneira, por meio do MSAA, a camada de negócio envia as informações para o sistema operacional e o leitor de tela consegue identificar as informações (Figura 1).

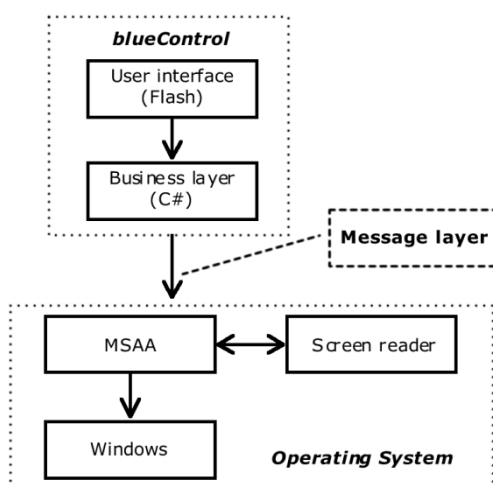


Figura 1 – Diagrama de comunicação.

Para o desenvolvimento das adaptações foram seguidos os padrões da WCAG 1.0. Este padrão é dividido em três níveis de prioridades, sendo: i) prioridade 1 - devem ser satisfeitas inteiramente, ii) prioridade 2 - deveriam ser satisfeitas e iii) prioridade 3 - podem ser satisfeitas. Essas prioridades foram estabelecidas para a identificação de qual usuários com deficiências seriam atendidos ao implementar cada nível de prioridades. Quanto maior o nível, mais abrangente se torna a aplicação da acessibilidade. Atualmente, os três níveis foram satisfeitos, respeitando as diferenças de conceitos e procedimentos entre as plataformas web e desktop.

Testes de regressão dos requisitos funcionais do sistema, no nível de sistema, foram feitos de forma manual, de modo a verificar a integridade dos dados finais resultantes. Assim, pode-se afirmar que a atualização do sistema não trouxe o surgimento de novos defeitos sobre as funcionalidades legadas.

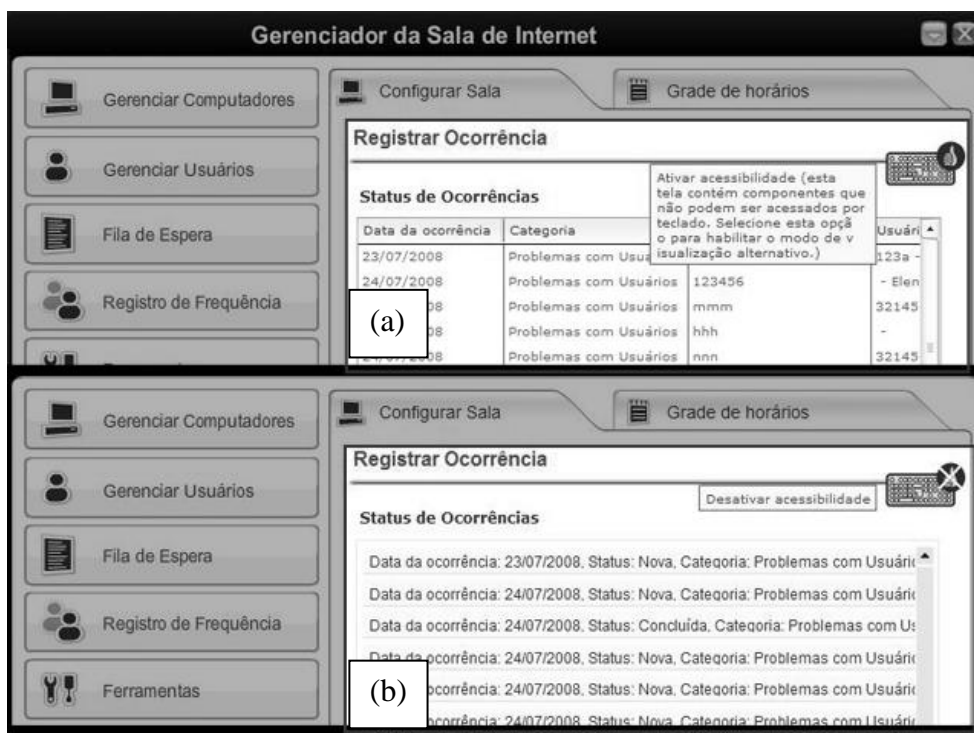
## 6. Avaliação e resultados

Após todas as modificações em sua estrutura e interface e análise preliminar em ambiente de teste, o software foi disponibilizado para testes em campo. Um pequeno grupo de três pessoas foi selecionado para avaliar preliminarmente o *blueControl*. O grupo foi restrito a pessoas com deficiência visual, devido ao foco deste estudo. Eles foram classificados de acordo com a definição da OMS sobre a cegueira e baixa visão [WHO e DIMDI, 2007], já possuindo treinamento sobre o uso do software. Nos testes

foram gravados áudio e vídeo para que as informações e respostas dos usuários fossem analisadas posteriormente.

Em posse da nova versão do software - com as alterações que implementam recursos de acessibilidade - os usuários foram instruídos a realizar suas tarefas de gestão do laboratório já estabelecidas por um roteiro de tarefas a serem feitas. Segundo a idéia do método Think Aloud [van Someren, Barnard, Sandberg, 1994] foi solicitado que os participantes falassem o que estavam fazendo e pensando em determinados momentos ou quando tinham alguma dificuldade. De acordo com a facilidade/dificuldade, tempo de execução da tarefa, opiniões colocadas durante as tarefas, entre outros parâmetros, pudemos avaliar o software de forma qualitativa, coletando informações subjetivas em termos de execução das tarefas do roteiro.

O DosVox foi escolhido para ser o software leitor de tela por ser um ambiente gratuito, de ampla utilização em programas governamentais brasileiros e por apresentar boa aceitação à utilização do Flash em ambiente desktop.



**Figura 2 – Demonstração de alternativa de acessibilidade para o componente DataGrid**

Existiram respostas negativas com relação à utilização do software: alguns componentes, como o Calendário, precisavam de uma entrada alternativa, já que não poderia fornecer o acesso direto às informações necessárias (por exemplo, um dia específico no futuro), tornando-o difícil de ser utilizado com grande frequência. O componente *DataGrid*, usado para mostrar uma tabela de registros, surgiu como um problema também. Uma alternativa foi a versão simplificada do *DataGrid*, com a fusão das colunas em uma única (Figura 2). Além disso, o excesso de explicação de textos em algumas formas dificultou a execução de algumas tarefas fluente. Na Figura 2 (a), pode-se ver o componente *DataGrid* em seu estado natural. Na Figura 2 (b), pode-se ver as

modificações feitas, sem as colunas dividindo o conteúdo, facilitando para que o leitor de tela compreenda todo o seu conteúdo.

A solução geral para esses componentes foi uma boa navegação pelo teclado. Embora a aplicação tivesse alguns problemas, os conceitos básicos da WCAG foram cumpridos adequadamente, com pequenos ajustes.

## 7. Discussão e conclusões

A aceitação pelo grupo de usuários foi positiva, com resultado final satisfatório. Nota-se que apesar do tema acessibilidade ter ganhado força, ainda há poucos trabalhos sobre a questão da acessibilidade aplicada a softwares e, na maioria dos casos, o tema principal está vinculado ao desenvolvimento de aplicações web.

Pelo resultado prévio junto aos usuários, as escolhas de abordagens de modernização do software *blueControl* se mostraram oportunas. Com as técnicas aplicadas neste estudo de caso foi possível adaptar uma aplicação desktop legada, comercializada e com manuais de uso prontos, para atender aos níveis de acessibilidade estabelecidos para aplicações Web, garantindo o respeito às leis e inclusão de usuários com deficiência visual no seu trabalho diário.

Embora este trabalho tenha utilizado apenas um leitor de tela, o DOSVOX, outras ferramentas de leitores de tela mais populares devem ser testadas para avaliar a abrangência desta abordagem. Como próximo passo, a aplicação de pesquisa qualitativa, com base nas recomendações WCAG, para os deficientes visuais que trabalham diariamente com *blueControl* permitirá avaliar quais são as conquistas dessa abordagem, podendo estender as adaptações a outros tipos de deficiência, como a deficiência motora. Dependendo das modificações exigidas, novos ciclos de reengenharia podem ser necessários.

Em trabalhos futuros, serão avaliados se houve melhora de desempenho dos usuários com deficiência quando estes usam o software novo, além da discussão das melhores práticas observadas durante o projeto.

## REFERÊNCIAS

- Adobe (2010). "Public Methods". Documentação Adobe Flash, <http://www.adobe.com/livedocs/flash/9.0/ActionScriptLangRefV3/flash/system/package.html>, Maio, 2010.
- Borges, J. A (2002). "O que é o Dosvox". Núcleo de Computação Eletrônica da UFRJ – Universidade Federal do Rio de Janeiro. <http://intervox.nce.ufrj.br/dosvox/intro.htm>, Abril, 2010.
- Brasil. Lei nº 8.213, de 24 de julho de 1991. "Dispõe sobre os Planos de Benefícios da Previdência Social e dá outras providências". Diário Oficial [da República Federativa do Brasil], Brasília, DF, v. 4, 25 jul. 1991. Seção 1, p. 14809.
- Freedom Scientific (2011). "JAWS for Windows Screen Reading Software". <http://www.freedomscientific.com/products/fs/jaws-product-page.asp>. Fevereiro, 2011.

- IBGE (2005). "IBGE e CORDE abrem encontro internacional de estatísticas sobre pessoas com deficiência". [http://www.ibge.gov.br/home/presidencia/noticias/noticia\\_visualiza.php?id\\_noticia=438&id\\_pagina=1](http://www.ibge.gov.br/home/presidencia/noticias/noticia_visualiza.php?id_noticia=438&id_pagina=1), Maio, 2010.
- Inclusão Digital (2011). "Governo lança edital para investir R\$ 165 milhões na inclusão digital". <http://www.inclusaodigital.gov.br/noticia/governo-lanca-edital-para-investir-r-165-milhoes-na-inclusao-digital>, Abril, 2011.
- Linux Foundation (2009). "IAccessible2". The Linux Foundation. <http://www.linuxfoundation.org/collaborate/workgroups/accessibility/iaccessible2>, Dezembro, 2010.
- Microsoft Corporation (2000). "Microsoft Active Accessibility: Architecture". [http://msdn.microsoft.com/en-us/library/ms971310.aspx#actvaccess\\_topic1](http://msdn.microsoft.com/en-us/library/ms971310.aspx#actvaccess_topic1), Maio, 2010.
- MStech (2011). <http://www.mstech.com.br/>, Janeiro, 2011.
- NVDA Project, (2010). "About NVDA". General Features. <http://www.nvda-project.org/wiki/About>, Maio, 2010.
- Programa Brasil Conectado (2011). "Programa nacional de Banda Larga". <http://www4.planalto.gov.br/brasilconectado>, Abril, 2011.
- Seacord, R., Plakosh D., Lewis, G. (2003), "Modernizing Legacy Systems: Software Technologies", Engineering Process and Business Practices, Addison-Wesley.
- Sonza, A.P. (2008). "Ambientes virtuais acessíveis sob a perspectiva de usuários com limitação visual". Doutorado. Universidade Federal do Rio Grande do Sul. Centro de Estudos Interdisciplinares em Novas Tecnologias da Educação. Programa de Pós-Graduação em Informática na Educação. <http://www.lume.ufrgs.br/handle/10183/14661>, Fevereiro, 2011.
- van Someren, M. W., Barnard, Y. F. , Sandberg, J. (1994). The think aloud method: a practical guide to modelling cognitive processes. 1ª edição. New York: Academic Press, 1994.
- W3C (1999). "Web Content Accessibility Guidelines 1.0". <http://www.w3.org/TR/WCAG10>, Março, 2010.
- W3C (2005). "Introduction to Web Accessibility". <http://www.w3.org/WAI/intro/accessibility.php>, Janeiro, 2011.
- WARD, M. P.; BENNETT, K. H.(1995) "Formal Methods for Legacy Systems". Journal of Software Maintenance: Research and Practice, v. 7, n. 3, p. 203-219.
- WHO (2009). "World Health Organization - Visual impairment and blindness – Fact Sheet no. 282". <http://www.who.int/mediacentre/factsheets/fs282/en/>, Março, 2010.
- WHO, DIMDI, (2007). "World Health Organization - Chapter VII - Diseases of the eye and adnexa". <http://apps.who.int/classifications/apps/icd/icd10online/?gH53.htm+h540>, Dezembro, 2010.