

Análise dos Desafios para Programar sem Enxergar: estudo de caso na disciplina Linguagem de Programação 1

Raul Santos Gonçalves¹, Robson Santos Santana¹, Francisco Anacreonte Bezerra de Souza Neto¹, Saullo Cruz Benevides¹, Naiara Silva dos Santos²

¹Discentes da Universidade Estadual do Sudoeste da Bahia (UESB)- Jequié, BA - Brazil

²Docente do Departamento de Ciências e Tecnologias-DCT
Universidade Estadual do Sudoeste da Bahia (UESB)- Jequié, BA - Brazil

{raul santos121, robson santos9999, franciscoanacreonte, saullocb}@gmail.com, Robson santos_18@hotmail.com, nssantos@uesb.edu.br

Abstract. *Screen readers make it possible for blind programmers to implement the code because they can hear what they are typing, but these programs have problems. This article describes the case study and analyzes the problems faced by blind programmers in the act of programming. Data were collected during the classes of the Programming Language 1 course. The DOSVOX screen reader and the C language compilers were used: Dev C ++, Falcon and CodeBlocks.*

Resumo. *Os leitores de tela possibilitam que os programadores cegos possam implementar os códigos, pois conseguem ouvir o que estão digitando, porém esses programas apresentam problemas. Este artigo descreve o estudo de caso e faz uma análise dos problemas enfrentados por programadores cegos, no ato de programar. Os dados foram coletados durante as aulas da disciplina de Linguagem de Programação 1. Foi utilizado o leitor de tela DOSVOX e os compiladores da linguagem C: Dev C++, Falcon e CodeBlocks.*

1. Introdução

A tecnologia está evoluindo mais rápido do que a capacidade humana, diz Friedman conforme Frabasile (2018), conseqüentemente, diversos problemas têm sido solucionados pelos programadores. O ato de programar exige a leitura constante do código e sua compreensão. Para facilitar os seus trabalhos, os programadores utilizam um Ambiente de Desenvolvimento Integrado (IDE), no qual apresentam recursos visuais como os *debuggers*, que indicam os erros encontrados no código. A visão, portanto, é necessária para a escrita, depuração e manutenção dos programas. Conforme Baker (2015) “o uso de recursos visuais apresentam dificuldades para programadores cegos, pois eles não conseguem acessar rapidamente as mesmas informações disponíveis para os desenvolvedores com visão”, então surgem alguns desafios de se programar sem a possibilidade de enxergar.

Os recursos tecnológicos permitem reduzir as barreiras enfrentadas por deficientes visuais, o que acarreta a possibilidade desses a desenvolver com o auxílio de programas de leitura de texto, além de outras ferramentas. De acordo com Portaria n.º1.679/1999 [Ministério da Educação 1999], são precisos alguns requisitos básicos para uma Universidade dispor de um melhor auxílio aos Deficientes Visuais - DV's, como: máquina de datilografia braile, impressora braile acoplada a computador, sistema de síntese de voz,

etc. Essa Portaria dispõe sobre requisitos de acessibilidade de pessoas portadoras de deficiências, para instruir os processos de autorização e de reconhecimento de cursos, e de credenciamento de instituições. Na década de 2006-2016, a SBC apontou como um dos cinco grandes desafios "Acesso participativo e universal dos cidadãos brasileiros ao conhecimento" [SBC 2017].

O presente artigo tem como objetivo identificar e analisar os problemas enfrentados por deficientes visuais no ato de programar, desde a concepção do problema a ser resolvido até a sua compilação e testes, examinando também os recursos tecnológicos disponíveis e propor soluções que visam sanar ou reduzir os problemas enfrentados. Algumas barreiras foram possíveis de ser identificadas, a partir deste estudo de caso na disciplina de Linguagem de Programação 1. O artigo está organizado da seguinte forma: na seção 2 é apresentada uma breve fundamentação teórica, na seção 3 é descrita a metodologia, na seção 4 são discutidos os problemas detectados e na seção 5 são apresentados os resultados.

2. Fundamentação Teórica

Os recursos tecnológicos permitem reduzir as barreiras enfrentadas por deficientes visuais, e permite que venham a desenvolver com o auxílio de programas de leitura de texto, que transformam o texto da tela do computador em áudio, como o JAWS for Windows, Virtual Vision, NVDA e DOSVOX. Segundo Cardoso, D'Ascenzi e Montserrat Neto (2009) o DOSVOX é um software para computadores e desktops, é um sistema que, por meio da síntese de voz, se comunica com o usuário. Nesse sentido, Souza (2008) afirma que "o ambiente Dosvox possui internamente seus próprios aplicativos, e tem como objetivo ser uma solução completa para que pessoas com deficiência visual possam fazer uso de computadores da plataforma PC".

Para pessoas com deficiência visual que resolvem seguir a carreira de desenvolvedor de software, as melhores ferramentas de apoio são os leitores de tela. Cheong (2010) aponta que traduzir as informações visuais para auditivas não é tão adequado e eficiente para tarefas avançadas da computação como a programação, um programador cego não poderá entender as estruturas dos códigos somente com o auxílio de um leitor de tela. Um recuo no código, por exemplo, são vários espaços em branco consecutivos que um leitor de tela não conseguiria traduzir.

Os autores Baker, Milne e Ladner (2015) argumentam que o campo da programação para desenvolvedores cegos ainda é jovem, ferramentas de desenvolvimento feitas por cegos ainda não são tão comuns. Nessas circunstâncias pode-se dizer que: IDEs acessíveis para desenvolvedores com deficiência visual ainda são uma nova área de pesquisa. Há muito pouca exploração que feita para melhorar o desenvolvimento e a experiência na programação para desenvolvedores com DV. [POTLURI et.al. 2018 tradução].

Existem alguns projetos para ensino de programação para deficientes visuais, como o Project Torino [Microsoft 2017] que consiste em uma linguagem de programação tátil, é basicamente um Scratch físico, em que você posiciona os blocos em diferentes arranjos, a fim de escrever um programa, porém para se programar com as linguagens comumente trabalhadas nas Universidades, como: C, C++ e Java, ainda é utilizada a combinação leitor de voz e IDE, onde o DV pode definir quantas palavras por minuto devem ser ditas.

3. Metodologia de Pesquisa

O estudo de caso é um método de pesquisa que utiliza, geralmente, dados qualitativos, coletados a partir de eventos reais, com o objetivo de explicar, explorar ou descrever fenômenos atuais inseridos em seu próprio contexto. Caracteriza-se por ser um estudo detalhado e exaustivo de poucos, ou mesmo de um único objeto, fornecendo conhecimentos profundos [Eisenhardt 1989].

A primeira etapa do trabalho foi o delineamento da pesquisa, seguida da pesquisa bibliográfica para identificar os recursos tecnológicos utilizados por pessoas com deficiência visual, a segunda etapa consistiu da observação, durante as aulas práticas da disciplina de Linguagem de Programação 1, como o deficiente visual programa, ou seja, o desenho da pesquisa. A seguir, foi feita a coleta de dados: levantados por intermédio de anotações dos problemas detectados durante a resolução de exercícios na aula. Por fim, a organização do material coletado durante as observações. Para a análise dos dados, selecionamos as informações levantadas no caso, descartando as desnecessárias, e organizamos em categorias definidas.

3. Problemas Detectados com o uso dos leitores de tela

Sem o leitor de tela, não seria possível que a pessoa com deficiência visual viesse a programar de forma autônoma. Porém, no nosso estudo de caso foram observados alguns problemas apresentados pelos leitores de tela durante a implementação do código, que geram erros sintáticos, como por exemplo: alguns caracteres digitados não são pronunciados pelo reprodutor de tela; não diferencia letras minúsculas e maiúsculas - que ocasiona erro sintático quando a linguagem é sensível ao contexto; não identifica caracteres depois do ponto e vírgula, caracteres duplicados escritos de forma sequencial são lidos apenas uma vez, etc. Todos esses problemas que podem gerar erros sintáticos fazem com que o código escrito não seja compilado.

O processo de compilação é uma tradução de um código escrito em uma linguagem de programação para a linguagem de máquina, com conseqüente criação do código objeto (executável). Para um código ser compilado é feita a análise léxica e sintática do código, caso ocorra algum erro, o código objeto não é gerado. Os problemas sintáticos, citados anteriormente, fazem com que o programa executável não seja criado, porém os ambientes de desenvolvimento não apresentam recursos para informar ao programador com deficiência visual que o seu código não foi compilado. Mesmo o programador com visão comete erros, como: esquecer de colocar ponto e vírgula, parêntese ou chave, utilização de variáveis que não foram declaradas, espaço entre símbolos que deveriam estar juntos, etc. Todos esses erros sintáticos são exibidos nas IDEs em um local específico, e o leitor de tela não faz a leitura desse campo, logo o programador com deficiência visual não sabe qual foi o erro cometido, só é possível saber que houve o erro, pois o código objeto não foi criado.

A voz robotizada dos leitores de tela também foi identificada como um problema, pois diversas vezes o programador não entendia o que deveria ser feito, ao se ler um problema a ser resolvido. Essas são dificuldades identificadas na etapa da implementação e depuração do código. Há também os problemas em tempo de execução: o programa executa, porém não o faz corretamente.

4. Resultados Esperados e Conclusão

Após a apresentação do problema de pesquisa, traçamos possíveis soluções para os problemas dos leitores de tela, acessibilidade nas universidades e no ensino de programação, que poderão ser desenvolvidos durante nossa pesquisa.

A fim de melhorar a acessibilidade dos programadores com DV, é importante que seja feita algumas adaptações nos leitores de tela para que possa ser possível fazer a leitura do campo das IDEs que são mostrados os erros sintáticos. É necessário também que seja adicionada uma função no leitor que possa diferenciar as letras maiúsculas das letras minúsculas. Uma possível solução também seria a criação de uma IDE que tenha o seu próprio reproduzidor de voz sem a necessidade de um software externo. Um reproduzidor capaz de identificar todos os erros de compilação, além de emitir um efeito sonoro ao ser identificado um erro em tempo de execução. Além dos itens anteriores, trabalhar é um dos maiores desafios enfrentados pelas pessoas com DV, por isso, é importante a criação ou alteração de um software para facilitar a adaptação de materiais didáticos, tornando assim, os ambientes educativos mais acessíveis e inclusivos.

Referências

- Baker, C., Milne, L. e Ladner, R. (2015) "StructJumper: A Tool to Help Blind Programmers Navigate and Understand the Structure of Code", <https://catiembaker.github.io/StructJumper.pdf>, Acesso em: 30 jan. 2020.
- Cardoso, D., D'Ascenzi, I, Monserrat Neto, J. (2009) "DOSVOX: a História de uma revolução de cegos". In: Revista SBC Horizontes.
- Cheong, C. (2010) "Coding without sight", In: Andrew Burge (ed.) Eighth Annual Hawaii International Conference on Education, Honolulu, Hawaii, 7-10 January 2010, pp. 1-12.
- Eisenhardt, K.M. (1989) Building theories form case study research. Academy of Management Review. New York, New York, v. 14 n. 4
- Frabasile, D. (2018) "A tecnologia está evoluindo mais rápido do que a capacidade humana", diz Friedman", <https://epocanegocios.globo.com/Tecnologia/noticia/2018/03/tecnologia-esta-evoluindo-mais-rapido-do-que-capacidade-humana-diz-friedman.html>
- Ministério da Educação (1999) "Portaria n.º 1.679, de 2 de dezembro de 1999", http://portal.mec.gov.br/sesu/arquivos/pdf/c1_1679.pdf, Acesso em: 26 nov 2019.
- Microsoft (2017). "Project Torino". <https://blogs.microsoft.com/ai/project-torino-microsoft-creates-physical-programming-language-inclusive-visually-impaired-children/>
- Potluri, V., Vidya, Y., Vaithilingam, P., Swaminathan, M., Ivengar, S., Srinivasa, G. (2018). CodeTalk: Improving Programming Environment Accessibility for Visually Impaired Developers
- SBC (2017) "Grandes Desafios da Pesquisa em Computação no Brasil – 2016 – 2026", http://www2.sbc.org.br/ce-si/arquivos/GranDSI-BR_Ebook-Final.pdf, Acesso em: 11 dez 2019.
- Souza, E. (2008) "Avaliação de usabilidade do sistema Dosvox na interação de cegos com a Web", in: Revista Benjamin Constant