

**UNIVERSIDADE FEDERAL DE PELOTAS**  
**Centro de Desenvolvimento Tecnológico**  
**Programa de Pós-Graduação em Computação**



Tese

**Diretrizes de Acessibilidade em Ambientes de Desenvolvimento Integrado para  
Estudantes Cegos**

**Eliana Zen**

Pelotas, 2024

**Eliana Zen**

**Diretrizes de Acessibilidade em Ambientes de Desenvolvimento Integrado para Estudantes Cegos**

Tese apresentada ao Programa de Pós-Graduação em Computação do Centro de Desenvolvimento Tecnológico da Universidade Federal de Pelotas, como requisito parcial à obtenção do título de Doutor em Ciência da Computação.

Orientadora: Prof. Dra. Tatiana Aires Tavares

Pelotas, 2024

Universidade Federal de Pelotas / Sistema de Bibliotecas  
Catalogação da Publicação

Z54d Zen, Eliana

Diretrizes de acessibilidade em ambientes de desenvolvimento integrado para estudantes cegos [recurso eletrônico] / Eliana Zen ; Tatiana Aires Tavares, orientadora. — Pelotas, 2024.  
223 f. : il.

Tese (Doutorado) — Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, 2024.

1. Acessibilidade. 2. Programação de Computadores. 3. Estudantes cegos. 4. Diretrizes. I. Tavares, Tatiana Aires, orient. II. Título.

CDD 005

**Eliana Zen**

**Diretrizes de Acessibilidade em Ambientes de Desenvolvimento Integrado para Estudantes Cegos**

Tese aprovada, como requisito parcial, para obtenção do grau de Doutor em Ciência da Computação, Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas.

**Data da Defesa:** 22 de março de 2024

**Banca Examinadora:**

Profa. Dra. Tatiana Aires Tavares (orientadora)  
Dra. em Engenharia Elétrica (UFRN)

Prof. Dr. Marilton Sanchotene de Aguiar  
Dr. em Ciência da Computação (UFRGS)

Prof. Dr. Marcelo Medeiros Eler  
Dr. em Ciência da Computação (USP)

Profa. Dra. Andreia Sias Rodrigues  
Dra. em Ciência da Computação (UFPEl)

Prof. Dr. Leomar Soares da Rosa Júnior  
Dr. em Microeletrônica (UFRGS)

Este trabalho é dedicado, especialmente, aos meus dois amores, Pedro e Otavio, por acreditarem em mim, por compreenderem as exigências desta jornada e permanecerem ao meu lado, celebrando minhas conquistas e me apoiando nos momentos de dificuldade.

## **AGRADECIMENTOS**

Gostaria de expressar minha gratidão a todos que, de alguma forma, contribuíram para a conclusão deste trabalho. Durante esta jornada, tive a honra e o privilégio de conhecer pessoas especiais, que me auxiliaram e tornaram este momento ainda mais marcante e significativo.

Em primeiro lugar, gostaria de expressar minha profunda gratidão à minha família. Em especial, aos meus amores, meu marido, Pedro, e meu filho, Otavio, que nunca deixaram de acreditar que esse momento chegaria, e sempre estiveram ao meu lado, oferecendo apoio incondicional e encorajamento. Suas palavras de incentivo foram a luz que guiou meus passos durante esta jornada desafiadora. À minha mãe, Carmen, e à minha irmã, Ana Maria, que não só me auxiliaram, mas também foram uma fonte constante de apoio e carinho para o Otávio, tornando-se um alicerce essencial durante este percurso. À minha cunhada Rosane e à Michele, pelas palavras de incentivo e pelo amor e atenção dirigidos ao Otavio em todos os momentos.

À minha orientadora, professora Tatiana, gostaria de expressar minha profunda gratidão por todo o apoio, orientação e aconselhamento fornecidos ao longo deste percurso acadêmico. Sua disposição em compartilhar seus conhecimentos e experiências foi inestimável, tornando-se mais do que uma mentora, mas também uma amiga. Os momentos compartilhados e o aprendizado adquirido sob sua orientação foram essenciais para o meu crescimento pessoal e profissional.

Ao meu colega e agora amigo, Vinícius, gostaria de expressar minha gratidão por estar sempre disponível, apoiando-me e ouvindo minhas angústias. Agradeço pelas conversas, conselhos e momentos de descontração compartilhados.

Aos meus colegas de trabalho, os professores do CIET, especialmente Anderson e Eric, que dedicaram seu tempo para conversar quando eu precisava desabafar e trocar ideias. Agradeço também ao Gustavo por concordar em participar do Projeto de Pesquisa, contribuindo para a construção deste trabalho. Além disso, gostaria de expressar minha gratidão à Renira, minha companheira de idas a Pelotas, e aos meus alunos, por aceitarem participar da pesquisa.

Não posso deixar de expressar minha gratidão à Fabiana e à Emanuele, que por vezes assumiram os cuidados da Easy sem hesitar, permitindo que eu me ausentasse com total tranquilidade.

Enfim, a todos que, direta ou indiretamente, participaram e contribuíram para essa conquista, gostaria de expressar meu profundo e sincero agradecimento.

*“Para as pessoas sem deficiência, a tecnologia torna as coisas mais fáceis. Para as pessoas com deficiência, a tecnologia torna as coisas possíveis.”*

MARY PAT RADABAUGH (1993)

## RESUMO

ZEN, Eliana. **Diretrizes de Acessibilidade em Ambientes de Desenvolvimento Integrado para Estudantes Cegos**. Orientadora: Tatiana Aires Tavares. 2024. 223 f. Tese (Doutorado em Ciência da Computação) – Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2024.

A Tecnologia Assistiva (TA) tem desempenhado um papel fundamental na integração e inclusão de pessoas com deficiência na sociedade. Especificamente para pessoas cegas, a adoção de TA é essencial para assegurar o acesso equitativo às Tecnologias de Informação e Comunicação (TICs). No entanto, nos cursos de Computação, o simples uso de TA pode não ser suficiente para garantir a compreensão dos conceitos abordados nas disciplinas, nem o acesso completo aos recursos oferecidos pelas ferramentas essenciais à formação profissional dos estudantes. Isso porque os leitores de tela, principal recurso de acessibilidade utilizado por estudantes cegos para interagir com sistemas digitais, realizam uma leitura linear do conteúdo disponibilizado na interface gráfica. Esta limitação pode exigir mais tempo e esforço cognitivo dos usuários. Esse desafio é particularmente evidente em Ambientes de Desenvolvimento Integrados (IDEs), que possuem interfaces fortemente baseadas em elementos gráficos e podem não ser completamente acessíveis aos leitores de tela se os requisitos de acessibilidade não forem adequadamente implementados. Nesse contexto, as diretrizes de design desempenham um papel crucial ao oferecer recomendações para o desenvolvimento de IDEs que atendam às necessidades de acessibilidade dos estudantes cegos. Este foi o enfoque dado nesta Tese de Doutorado, que buscou especificar um conjunto de diretrizes para apoiar o projeto e a implementação de IDEs que considerem os requisitos de acessibilidade necessários para facilitar a interação desses estudantes. Para alcançar esse objetivo, o conjunto de diretrizes foi elaborado seguindo uma metodologia que englobou os seguintes estágios: Exploratório, Descritivo, Correlacional, Seleção, Especificação, Validação e Refinamento. Em cada uma dessas fases, foram conduzidos estudos que incluíram revisão da literatura, questionários, entrevistas e grupo focal, contando com a participação de estudantes, professores, programadores profissionais, analista de acessibilidade e desenvolvedor de IDEs. Essas etapas culminaram na elaboração das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos. Os resultados obtidos nas etapas de validação indicam que este conjunto de Diretrizes cumpre sua missão de auxiliar o desenvolvimento de IDEs acessíveis a esses estudantes.

Palavras-chave: Acessibilidade; Programação de Computadores; Estudantes cegos; Diretrizes.

## ABSTRACT

ZEN, Eliana. **Accessibility Guidelines in Integrated Development Environments for Blind Students**. Advisor: Tatiana Aires Tavares. 2024. 223 f. Thesis (Doctorate in Computer Science) – Technology Development Center, Federal University of Pelotas, Pelotas, 2024.

Assistive Technology (AT) has played a pivotal role in integrating and including people with disabilities in society. Adopting AT is essential for blind people to ensure equitable access to Information and Communication Technologies (ICTs). However, in Computer Science courses, more than merely using AT may be required to guarantee an understanding of the concepts covered in the subjects and complete access to the resources offered by essential tools for their professional development. This is because screen readers, the primary accessibility tool blind students use to interact with digital systems, perform a linear reading of the content provided in the graphical interface. This limitation may require more time and cognitive effort from users. This challenge is particularly evident in Integrated Development Environments (IDEs), which have interfaces heavily reliant on graphical elements and may not be fully accessible to screen readers if accessibility requirements are not adequately implemented. In this context, design guidelines are crucial in providing recommendations for developing IDEs that meet the accessibility needs of blind students. This was the focus of this doctoral thesis, which sought to specify a set of guidelines to support the design and implementation of IDEs that consider the accessibility requirements necessary to facilitate the interaction of these students. To achieve this goal, a set of guidelines was developed following a methodology encompassing the following stages: Exploratory, Descriptive, Correlational, Selection, Specification, Validation, and Refinement. In each phase, studies included a literature review, survey, interviews, and focus groups involving students, professors, professional programmers, accessibility analysts, and IDE developers. These steps culminated in elaborating Accessibility Guidelines in IDEs for Blind Students. The results obtained in the validation stages indicate that this set of guidelines fulfills its mission of assisting in developing IDEs accessible to these students.

Keywords: Accessibility; Computer programming; Blind students; Guidelines.

## LISTA DE FIGURAS

Figura 1	Intersecção entre as áreas de conhecimento de Tecnologia Assitiva e Interação Humano Computador com Programação de Computadores . . . . .	20
Figura 2	Modelo e notação de processos de negócios . . . . .	34
Figura 3	Processo de elaboração das Diretrizes . . . . .	37
Figura 4	Estágio Exploratório . . . . .	42
Figura 5	Estágios Descritivo, Correlacional, de Seleção e de Especificação .	58
Figura 6	Grupo Focal com Profissionais do IFFar-SVS . . . . .	64
Figura 7	Estágios de Validação e Refinamento (1ª iteração) . . . . .	67
Figura 8	Avaliação do contexto geral das recomendações por estudantes de computação . . . . .	74
Figura 9	Organização dos grupos de atividade prática do Grupo Focal . . . .	76
Figura 10	Estrutura básica da classe Java necessária para a atividade prática do Grupo Focal . . . . .	76
Figura 11	Exemplo de resolução para o problema solicitado na atividade prática do Grupo Focal . . . . .	77
Figura 12	Estágio de Validação (2ª Iteração) . . . . .	84
Figura 13	Avaliação do contexto geral das recomendações por Programadores Cegos . . . . .	90
Figura 14	Avaliação Individual das Recomendações preliminares por Desenvolvedor de IDE . . . . .	95
Figura 15	Pergunta: "A recomendação é apresentada de forma clara e compreensível?" . . . . .	187
Figura 16	Pergunta: "O texto da recomendação esclarece a forma como ela deve ser aplicada?" . . . . .	188
Figura 17	Pergunta: "Os detalhes foram suficientes para a compreensão da recomendação?" . . . . .	188
Figura 18	Pergunta: "A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?" . . . . .	189
Figura 19	Pergunta: "As sugestões de implementação apresentadas em "Como fazer?" são úteis?" . . . . .	189

Figura 20	Pergunta: "Há algum termo técnico ou jargão que precisa de mais explicação?" . . . . .	190
Figura 21	Pergunta: "A recomendação é apresentada de forma clara e compreensível?" . . . . .	191
Figura 22	Pergunta: "O texto da recomendação esclarece a forma como ela deve ser aplicada?" . . . . .	191
Figura 23	Pergunta: "Os detalhes foram suficientes para a compreensão da recomendação?" . . . . .	192
Figura 24	Pergunta: "A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?" . . . . .	192
Figura 25	Pergunta: "As sugestões de implementação apresentadas em "Como fazer?" são úteis?" . . . . .	193
Figura 26	Pergunta: "Há algum termo técnico ou jargão que precisa de mais explicação?" . . . . .	193
Figura 27	Avaliação individual das Diretrizes realizada por Desenvolvedor de IDE . . . . .	195

## LISTA DE TABELAS

Tabela 1	Desafios e Barreiras identificados na Revisão da Literatura . . . . .	45
Tabela 2	Categorias e Códigos adotados na Análise de Conteúdo . . . . .	50
Tabela 3	Desafios e Barreiras Educacionais - Categoria: Acomodações . . . . .	51
Tabela 4	Desafios e Barreiras de Interação - Categoria: Tarefas e Desafios de Programação . . . . .	54
Tabela 5	Dificuldades de Interação com IDEs por Pessoas com Deficiência Visual . . . . .	59
Tabela 6	Categorias e Diretrizes Preliminares . . . . .	61
Tabela 7	Reestruturação das Diretrizes da Categoria Categoria 3 (Navegação no Código) . . . . .	82
Tabela 8	Resumo de categorias e diretrizes . . . . .	107
Tabela 9	Diretriz 1.1 Semântica . . . . .	108
Tabela 10	Diretriz 2.1 Informações sobre erros . . . . .	110
Tabela 11	Diretriz 2.2 Variáveis e constantes . . . . .	111
Tabela 12	Diretriz 2.3 Pontos de interrupção . . . . .	113
Tabela 13	Diretriz 3.1 Rótulos . . . . .	114
Tabela 14	Diretriz 3.2 Movimentação pelo código . . . . .	114
Tabela 15	Diretriz 3.3 Contexto e nível de escopo . . . . .	116
Tabela 16	Diretriz 3.4 Estratégias de navegação . . . . .	117
Tabela 17	Diretriz 3.5 Numeração de linhas . . . . .	119
Tabela 18	Diretriz 3.6 Atalhos . . . . .	119
Tabela 19	Diretriz 3.7 Ajuda e Documentação . . . . .	121
Tabela 20	Diretriz 4.1 Espaçamento . . . . .	122
Tabela 21	Diretriz 4.2 Autocompletar . . . . .	123
Tabela 22	Diretriz 4.3 Comentários . . . . .	124
Tabela 23	Diretriz 5.1 Dobra de Código . . . . .	125
Tabela 24	Diretriz 5.2 Visão geral do código . . . . .	126
Tabela 25	Diretriz 6.1 Construção de Interface Gráfica . . . . .	128
Tabela 26	Diretriz 6.2 Validação da interface gráfica . . . . .	129
Tabela 27	Diretriz 7.1 Alertas sonoros . . . . .	130
Tabela 28	Diretriz 8.1 Idioma . . . . .	131
Tabela 29	Diretriz 8.2 Leitura contextual . . . . .	132

## LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i>
AVA	Ambiente Virtual de Aprendizagem
IA	Inteligência Artificial
IBGE	Instituto Brasileiro de Geografia e Estatística
IDE	<i>Integrated Development Environment</i>
IEC	<i>International Electrotechnical Commission</i>
IFFar	Instituto Federal Farroupilha
IFFar-SVS	Instituto Federal Farroupilha Campus São Vicente do Sul
IHC	Interação Humano-Computador
INEP	Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira
ISO	<i>International Organization for Standardization</i>
JAWS	<i>Job Access With Speech</i>
LDB	Lei de Diretrizes e Bases da Educação Nacional
NVDA	<i>NonVisual Desktop Access</i>
ODS	Objetivos do Desenvolvimento Sustentável
OMS	Organização Mundial de Saúde
ONU	Organização das Nações Unidas
PNAD	Pesquisa Nacional por Amostra de Domicílios Contínua
RSL	Revisão Sistemática da Literatura
TA	Tecnologia Assistiva
TICs	Tecnologias de Informação e Comunicação
TCLE	Termo de Consentimento Livre e Esclarecido
TUIV	Termo para Uso de Imagem e/ou Som da Voz
VSCoDe	<i>Visual Studio Code</i>
WHO	<i>World Health Organization</i>

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	17
1.1	Problema e Questão de Pesquisa	19
1.2	Objetivos	21
1.3	Organização do Trabalho	21
<b>2</b>	<b>CONTEXTUALIZAÇÃO TEÓRICA E ESTADO DA ARTE</b>	23
2.1	Acessibilidade	23
2.2	Tecnologia Assistiva	24
2.3	Leitores de Tela	26
2.4	Ambientes de Desenvolvimento Integrado (IDEs)	28
2.5	Abordagens para Elaboração de Diretrizes de Acessibilidade	30
2.5.1	Embasamento para a Análise Qualitativa	30
2.5.2	Embasamento Metodológico	33
2.6	Considerações Parciais	35
<b>3</b>	<b>PERCURSO METODOLÓGICO</b>	36
3.1	Caracterização da Pesquisa	36
3.2	Processo de elaboração das Diretrizes	36
3.2.1	Estágio Exploratório	38
3.2.2	Estágio Descritivo	38
3.2.3	Estágio Correlacional	38
3.2.4	Estágio de Seleção e Especificação	39
3.2.5	Estágio de Validação	39
3.2.6	Estágio de Refinamento	40
3.3	Considerações Parciais	40
<b>4</b>	<b>DESAFIOS E BARREIRAS DE INTERAÇÃO</b>	42
4.1	Revisão da Literatura	43
4.1.1	Caracterização do Estudo	43
4.1.2	Resultados	45
4.2	Entrevistas e Questionários	48
4.2.1	Caracterização do Estudo	48
4.2.2	Análise dos Dados	49
4.2.3	Resultados	50
4.3	Considerações Parciais	55

<b>5</b>	<b>ESPECIFICAÇÃO DAS DIRETRIZES</b>	<b>57</b>
<b>5.1</b>	<b>Versão Preliminar das Diretrizes</b>	<b>57</b>
5.1.1	Categorização das Diretrizes	58
5.1.2	Estrutura das Recomendações Preliminares	60
5.1.3	Recomendações Preliminares	61
<b>5.2</b>	<b>Cartilha</b>	<b>63</b>
<b>5.3</b>	<b>Considerações Parciais</b>	<b>65</b>
<b>6</b>	<b>VALIDAÇÃO E REFINAMENTO DAS DIRETRIZES (1 ITERAÇÃO)</b>	<b>67</b>
<b>6.1</b>	<b>Caracterização do Estudo</b>	<b>68</b>
<b>6.2</b>	<b>1ª Etapa de Validação: Questionário</b>	<b>68</b>
6.2.1	Resultados	70
<b>6.3</b>	<b>2ª Etapa de Validação: Grupo Focal</b>	<b>75</b>
6.3.1	Resultados	77
<b>6.4</b>	<b>Refinamento das Diretrizes</b>	<b>81</b>
<b>6.5</b>	<b>Considerações Parciais</b>	<b>83</b>
<b>7</b>	<b>VALIDAÇÃO E REFINAMENTO DAS DIRETRIZES (2 ITERAÇÃO)</b>	<b>84</b>
<b>7.1</b>	<b>Caracterização do Estudo</b>	<b>84</b>
<b>7.2</b>	<b>Validação por Programadores Cegos</b>	<b>86</b>
7.2.1	Resultados	86
<b>7.3</b>	<b>Validação por Analista de Acessibilidade</b>	<b>91</b>
7.3.1	Resultados	91
<b>7.4</b>	<b>Validação por Desenvolvedor de IDE</b>	<b>93</b>
7.4.1	Resultados	94
<b>7.5</b>	<b>Refinamento das Diretrizes</b>	<b>96</b>
<b>7.6</b>	<b>Considerações Parciais</b>	<b>106</b>
<b>8</b>	<b>DIRETRIZES DE ACESSIBILIDADE EM IDEs PARA ESTUDANTES CEGOS</b>	<b>107</b>
<b>8.1</b>	<b>CATEGORIA 1: Compreensão do Código</b>	<b>108</b>
<b>8.2</b>	<b>CATEGORIA 2: Depuração de Código</b>	<b>110</b>
<b>8.3</b>	<b>CATEGORIA 3: Navegação no Código</b>	<b>114</b>
<b>8.4</b>	<b>CATEGORIA 4: Edição de Código</b>	<b>122</b>
<b>8.5</b>	<b>CATEGORIA 5: <i>Skimming</i> de Código</b>	<b>125</b>
<b>8.6</b>	<b>CATEGORIA 6: Interface Gráfica de Usuário</b>	<b>127</b>
<b>8.7</b>	<b>CATEGORIA 7: Sobrecarga Auditiva</b>	<b>130</b>
<b>8.8</b>	<b>CATEGORIA 8: Leitura de Código-fonte</b>	<b>131</b>
<b>8.9</b>	<b>Outras Recomendações</b>	<b>133</b>
<b>8.10</b>	<b>Considerações Parciais</b>	<b>134</b>
<b>9</b>	<b>CONCLUSÕES</b>	<b>135</b>
<b>9.1</b>	<b>Principais Contribuições</b>	<b>137</b>
<b>9.2</b>	<b>Limitações da Pesquisa</b>	<b>139</b>
<b>9.3</b>	<b>Direcionamento para Trabalhos Futuros</b>	<b>139</b>
	<b>REFERÊNCIAS</b>	<b>141</b>
	<b>APÊNDICE A PARECER CONSUBSTANCIADO N. 54297421.9.0000.5574</b>	<b>152</b>
	<b>APÊNDICE B PARECER CONSUBSTANCIADO N. 68361023.1.0000.5574</b>	<b>156</b>

<b>ANEXO A</b>	<b>QUESTIONÁRIO ENVIADO AOS DOCENTES DE PROGRAMAÇÃO DE COMPUTADORES . . . . .</b>	<b>163</b>
<b>ANEXO B</b>	<b>TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO (TCLE) ENVIADO A ESTUDANTES E EGRESSOS . . . . .</b>	<b>172</b>
<b>ANEXO C</b>	<b>TERMO PARA USO DA IMAGEM E/OU SOM DA VOZ (TUIV) ENVIADO A ESTUDANTES E EGRESSOS . . . . .</b>	<b>175</b>
<b>ANEXO D</b>	<b>ROTEIRO DE ENTREVISTAS REALIZADAS COM ESTUDANTES COM DEFICIÊNCIA VISUAL . . . . .</b>	<b>177</b>
<b>ANEXO E</b>	<b>ROTEIRO DE GRUPO FOCAL COM ESTUDANTES DA DISCIPLINA DE INTERAÇÃO HUMANO COMPUTADOR . . . . .</b>	<b>179</b>
<b>ANEXO F</b>	<b>ROTEIRO DE GRUPO FOCAL COM PEDAGOGAS, PROFESSORAS DE EDUCAÇÃO ESPECIAL E PROFESSORES DE PROGRAMAÇÃO DE COMPUTADORES . . . . .</b>	<b>182</b>
<b>ANEXO G</b>	<b>E-MAIL CONVITE ENVIADO PARA PROGRAMADORES CEGOS E DESENVOLVEDORES DE IDE . . . . .</b>	<b>185</b>
<b>ANEXO H</b>	<b>RESPOSTAS PARA AS PERGUNTAS QUE UTILIZAVAM ESCALA <i>LIKERT</i> AO QUESTIONÁRIO DE AVALIAÇÃO DAS DIRETRIZES DE ACESSIBILIDADE EM IDES PARA ESTUDANTES COM DEFICIÊNCIA VISUAL ENVIADO A ESTUDANTES DE COMPUTAÇÃO . . . . .</b>	<b>187</b>
<b>ANEXO I</b>	<b>RESPOSTAS PARA AS PERGUNTAS QUE UTILIZAVAM ESCALA <i>LIKERT</i> AO QUESTIONÁRIO DE AVALIAÇÃO DAS DIRETRIZES DE ACESSIBILIDADE EM IDES PARA ESTUDANTES CEGOS ENVIADO A PROGRAMADORES CEGOS . . . . .</b>	<b>191</b>
<b>ANEXO J</b>	<b>RESPOSTAS PARA AS PERGUNTAS QUE UTILIZAVAM ESCALA <i>LIKERT</i> AO QUESTIONÁRIO DE AVALIAÇÃO DAS DIRETRIZES DE ACESSIBILIDADE EM IDES PARA ESTUDANTES CEGOS ENVIADO AO DESENVOLVEDOR DE IDE . . . . .</b>	<b>194</b>
<b>ANEXO K</b>	<b>QUESTIONÁRIO DE AVALIAÇÃO DAS DIRETRIZES DE ACESSIBILIDADE EM IDES PARA PESSOAS COM DEFICIÊNCIA VISUAL . . . . .</b>	<b>196</b>
<b>ANEXO L</b>	<b>TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO (TCLE) GRUPO FOCAL COM ESTUDANTES DE COMPUTAÇÃO . . . . .</b>	<b>221</b>

# 1 INTRODUÇÃO

A deficiência é tema dos direitos humanos de relevância mundial, amplamente debatido internacionalmente por meio de tratados, convenções, eventos e legislações específicas (Guimarães et al., 2016). Uma pessoa com deficiência é alguém com um impedimento de longo prazo de natureza física, mental, intelectual ou sensorial. Essa condição, quando combinada com barreiras sociais, pode dificultar sua capacidade de participação na sociedade em igualdade de condições com os demais (Brasil, 2015).

De acordo com a Organização Mundial da Saúde (OMS), aproximadamente 1,3 bilhão de pessoas em todo o mundo enfrentam algum tipo de deficiência significativa, correspondendo a cerca de 16% da população global (WHO, 2024). Dentre essas pessoas, cerca de 45 milhões são cegas (WHO, 2023). No Brasil, estima-se que cerca 18.6% da população com 2 anos ou mais apresente algum tipo de deficiência (visual, auditiva, motora e mental ou intelectual), o que corresponde a 8,9% da população (PNAD, 2023).

Esses dados têm impactos significativos na educação, uma vez que a deficiência visual afeta um número substancial de estudantes matriculados em cursos superiores no país. Em 2021, 63.404 alunos com deficiência, transtornos globais do desenvolvimento ou altas habilidades/superdotação estavam matriculados em cursos de graduação. Desse total, 23.654 possuíam alguma deficiência visual, sendo 3.482 cegos (INEP, 2022).

A inclusão e a acessibilidade para pessoas com deficiência também é um dos temas abordados pela Organização das Nações Unidas (ONU), que estabeleceu 17 Objetivos de Desenvolvimento Sustentável (ODSs) (ONU, 2024). Esses ODSs constituem um “plano para alcançar um futuro melhor e mais sustentável para todos” até 2030, visando diminuir as desigualdades e superar os principais desafios de desenvolvimento enfrentados no Brasil e no mundo (Burton et al., 2021). O quarto ODS, intitulado "Educação de qualidade", busca assegurar uma educação inclusiva, equitativa e de excelência, promovendo oportunidades de aprendizado para todas as pessoas.

Outras iniciativas semelhantes têm surgido ao longo dos últimos anos, visando proteger os direitos das pessoas com deficiência, reduzir as desigualdades e combater a

discriminação. Especificamente no âmbito educacional, normas e leis desempenham um papel crucial na promoção da inclusão e na garantia da permanência das pessoas com deficiência no ambiente escolar (Porte; Trindade, 2022). No Brasil, a Lei de Diretrizes e Bases da Educação Nacional (LDB), (Brasil, 1996), reconhece a igualdade de acesso e permanência na escola para pessoas com deficiência, garantindo o atendimento educacional especializado aos educandos nas instituições de ensino. A Lei Brasileira de Inclusão da Pessoa com Deficiência, (Brasil, 2015), visa assegurar e promover os direitos e as liberdades essenciais das pessoas com deficiência em um contexto de igualdade. Além disso, o Decreto 6.949, de 2009 (Brasil, 2004) promulga a Convenção Internacional sobre os Direitos das Pessoas com Deficiência.

Essa abordagem inclusiva tem sido impulsionada por políticas públicas e movimentos que visam assegurar os direitos das pessoas com deficiência (Sales; Torres, 2022). Tais iniciativas promovem a inclusão social e fortalecem a cidadania desses indivíduos, oferecendo respaldo legal para poderem conquistar diversos espaços, incluindo o acesso a instituições de ensino públicas e privadas em todos os níveis e modalidades educacionais (Beda et al., 2022).

As Tecnologias de Informação e Comunicação (TICs) têm desempenhado papel fundamental para assegurar que as pessoas com deficiência tenham pleno acesso à informação em todas as suas formas (Murillo-morales; Miesenberger, 2020), ampliando a disponibilidade de produtos, serviços e informações (Paciello, 2000), especialmente no ambiente educacional. Contudo, muitos desses ambientes ainda podem carecer de melhorias em termos de acessibilidade e usabilidade, o que impõe desafios significativos e limita o acesso pleno e eficaz para muitos usuários. Uma pesquisa conduzida pelo “Movimento Web Para Todos” em 2021 mostrou que o número de sites brasileiros aprovados em todos os testes de acessibilidade representa menos de 1% do total (MWPT, 2021).

Essas limitações podem impactar significativamente a interação de usuários com deficiência visual, especialmente considerando que, na maioria dos sistemas, a aquisição das informações ocorre predominantemente através do sentido da visão. Esse impacto se torna ainda mais relevante quando se trata de cursos na área de Computação, que dependem amplamente de pistas visuais, impondo desafios tanto no processo de ensino-aprendizagem quanto na execução de atividades práticas e no uso de tecnologias (Rajaselvi et al., 2021).

Sobretudo nas disciplinas de Programação de Computadores, a utilização de ferramentas altamente dependentes de elementos gráficos, aliado às características intrínsecas da maioria das linguagens de programação<sup>1</sup>, impõem barreiras significativas para pessoas com deficiência visual realizarem tarefas relacionadas a desenvolvimento de software.

---

<sup>1</sup>Linguagem artificial usada para controlar o fluxo de operações de um computador (Sharma, 2020).

Os Ambientes de Desenvolvimento Integrado (IDEs), do inglês *Integrated Development Environment*, necessários nessas disciplinas, dependem muito de recursos visuais e metáforas, o que os torna não totalmente acessíveis. Ao tentarem utilizá-los, os estudantes cegos podem se deparar com desafios significativos decorrentes de problemas de acessibilidade. Essas dificuldades podem obrigá-los a recorrer a ferramentas alternativas, que muitas vezes contém funcionalidades reduzidas e diferem daquelas utilizadas pelos colegas.

Esses problemas de acessibilidade podem não apenas afetar o aprendizado dos estudantes com deficiência, mas também criar uma desigualdade no acesso aos recursos educacionais, comprometendo assim a qualidade e a equidade do ensino. Tal situação pode resultar em um aumento do tempo necessário para concluir as atividades e demandar um esforço cognitivo adicional, o que, por sua vez, pode gerar frustração e ter um impacto negativo no engajamento e na dedicação dos estudantes. Além disso, caso não recebam treinamento adequado e não adquiram familiaridade com essas tecnologias ao longo do curso, a formação profissional desses estudantes pode ser comprometida, visto que o domínio dos recursos e ferramentas disponíveis para auxiliar nas tarefas de programação é fundamental para sua atuação profissional.

Reconhecendo a importância de garantir que todos os estudantes tenham acesso igualitário aos recursos essenciais para o aprendizado de programação, melhorar a acessibilidade dessas ferramentas torna-se fundamental. Neste sentido, este trabalho descreve o processo de elaboração de Diretrizes de Acessibilidade em Ambientes de Desenvolvimento Integrado para Estudantes Cegos. Essas diretrizes priorizam a utilização dos principais recursos e funcionalidades oferecidos por esses ambientes, considerando a necessidade de proporcionar uma experiência inclusiva e acessível.

É importante ressaltar que essas diretrizes não se concentram em funcionalidades extremamente avançadas ou em recursos específicos, mas sim nas funcionalidades associadas às tarefas e recursos essenciais para o aprendizado de programação de computadores.

## **1.1 Problema e Questão de Pesquisa**

A programação de computadores abrange atividades como escrever, testar, depurar e manter o código-fonte de programas (Sharma, 2020), no intuito de instruir um computador a solucionar problemas (Hermans; Aldewereld, 2017). Para programar, é necessário conhecer uma ou mais linguagens de programação, compreender o contexto ou problema abordado e utilizar corretamente as instruções ou comandos textuais específicos (sintaxe) da linguagem adotada (Gomes et al., 2015).

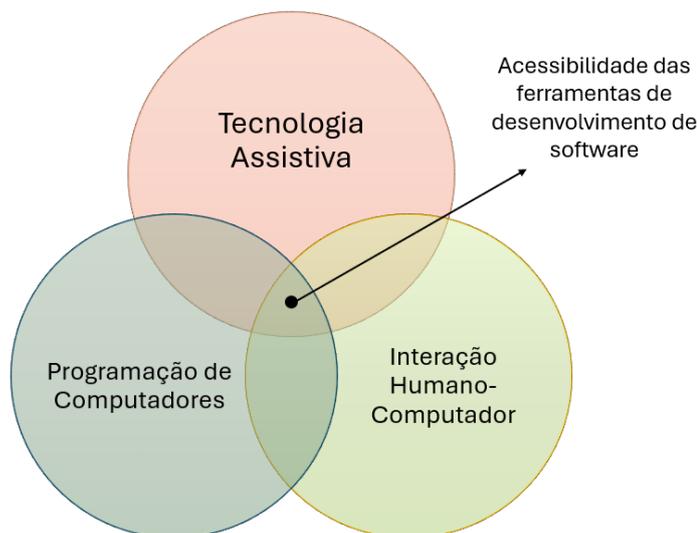
Por essa razão, essa não é uma área de estudo trivial, pois demanda a compreensão de conceitos abstratos (Lahtinen; Ala-mutka; Järvinen, 2005) e o desenvolvimento

de um conjunto de habilidades necessárias para conceber programas capazes de resolver problemas reais. Neste sentido, aprender a programar pode ser especialmente desafiador para estudantes que estão iniciando nessa área (Gomes et al., 2015).

No caso específico de estudantes cegos, algumas dificuldades adicionais podem ser encontradas, já que muitas linguagens de programação dependem fortemente de elementos visuais e apresentam uma sintaxe complexa. Se os estudantes optarem por utilizar editores de texto simples, enfrentarão o desafio de verificar a consistência do programa e realizar a leitura correta das linhas de comando. Por outro lado, se forem fornecidas ferramentas para apoiar a construção de programas, eles precisarão interagir com interfaces gráficas de usuário, que podem não ser muito acessíveis (Sánchez; Aguayo, 2006).

Reconhece-se como fundamental, portanto, conhecer as barreiras e desafios enfrentados por estudantes cegos ao interagir com ferramentas de desenvolvimento de software utilizando leitores de tela (Figura 1). Essa compreensão é essencial para desenvolver soluções que promovam a inclusão e a acessibilidade, garantindo que esses estudantes tenham igualdade de oportunidades no ambiente educacional e profissional.

Figura 1 – Intersecção entre as áreas de conhecimento de Tecnologia Assistiva e Interação Humano Computador com Programação de Computadores



Fonte: Próprio autor (2024)

Diante do contexto e do problema de pesquisa apresentados, formulou-se a seguinte questão de pesquisa para orientar os estudos científicos conduzidos nesta Tese de Doutorado:

***"Quais estratégias devem ser adotadas para promover a acessibilidade em Ambientes de Desenvolvimento Integrado para estudantes com deficiência visual?"***

A definição da questão de pesquisa serviu como base para estabelecer os objetivos a serem alcançados neste estudo, os quais serão detalhados na próxima Seção.

## 1.2 Objetivos

O objetivo principal desta Tese foi estabelecer um conjunto de Diretrizes de Acessibilidade para apoiar o desenvolvimento de Ambientes de Desenvolvimento Integrado acessíveis a estudantes cegos. Para atingir esse objetivo, foram estabelecidos os seguintes objetivos específicos:

- **OE1:** Identificar os recursos de TA mais utilizados por pessoas com deficiência visual para interagir com sistemas digitais, bem como as principais barreiras e limitações encontradas por eles durante a interação;
- **OE2:** Investigar as principais barreiras e limitações encontradas por estudantes com deficiência visual para aprender a programar e realizar tarefas relacionadas à programação de computadores;
- **OE3:** Investigar as principais barreiras e limitações encontradas por estudantes com deficiência visual para interagir com IDEs;
- **OE4:** Elaborar um conjunto preliminar de recomendações para aprimorar a acessibilidade de IDEs para estudantes com deficiência visual;
- **OE5:** Avaliar as recomendações preliminares com diferentes grupos de especialistas do domínio da aplicação; e,
- **OE6:** Refinar as estratégias preliminares para elaboração do conjunto de Diretrizes de Acessibilidade em IDEs para estudantes cegos.

Pretende-se, com esta Tese de Doutorado, fornecer orientações para analistas e desenvolvedores durante o processo de criação e atualização de Ambientes de Desenvolvimento Integrado, tornando-os mais acessíveis para atender às necessidades dos estudantes cegos.

## 1.3 Organização do Trabalho

Neste capítulo, foi apresentado e contextualizado o tema abordado nesta Tese de Doutorado, incluindo a descrição do problema de pesquisa, bem como a questão de pesquisa e os objetivos que nortearam este estudo.

No Capítulo 2, são discutidos os principais conceitos encontrados na literatura científica envolvidos desenvolvimento da pesquisa em questão, além de abordar as tecnologias diretamente relacionadas ao tema desta Tese e os trabalhos que guiaram a condução do estudo.

O Capítulo 3 descreve as abordagens metodológicas adotadas e as etapas executadas para a elaboração do conjunto de diretrizes.

O Capítulo 4 detalha os procedimentos adotados para o levantamento das barreiras e desafios enfrentados por estudantes com deficiência visual ao interagir com IDEs.

No Capítulo 5, é descrito o processo de compilação e categorização das barreiras e desafios enfrentados por estudantes com deficiência visual ao interagir com IDEs, que resultou na versão preliminar das Diretrizes de Acessibilidade em IDEs para Estudantes com Deficiência Visual.

A validação desse conjunto preliminar de recomendações é apresentada no Capítulo 6, juntamente com os resultados obtidos. Essa etapa contou com a participação de estudantes de cursos superiores de computação e resultou na 1ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos.

Posteriormente, a versão atualizada das recomendações foi submetida a uma nova etapa de validação, envolvendo a participação de três programadores cegos, um analista de acessibilidade e um professor que também atua como desenvolvedor de IDE. Os resultados dessa 2ª validação, bem como o processo de revisão e análise dos resultados obtidos são apresentados no Capítulo 7.

A 2ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos é descrita no Capítulo 8. Por fim, no Capítulo 9, são discutidas as conclusões do estudo, incluindo as principais contribuições, a identificação das limitações da pesquisa e uma análise das perspectivas e oportunidades para futuras investigações.

## 2 CONTEXTUALIZAÇÃO TEÓRICA E ESTADO DA ARTE

Este Capítulo apresenta a base teórica que sustenta o desenvolvimento desta Tese de Doutorado. Primeiramente, explora-se o conceito de Acessibilidade e Tecnologia Assistiva (TA), com destaque para os leitores de tela, principal recurso de acessibilidade utilizado por usuários com deficiência visual para interagir com sistemas digitais (Zen et al., 2022, 2023). Em seguida, são abordadas as características e peculiaridades dos Ambientes de Desenvolvimento Integrado (IDEs), seguido dos estudos que fundamentam o percurso metodológico adotado em cada etapa deste trabalho.

### 2.1 Acessibilidade

Segundo a Lei 13.146, de 6 de julho 2015, (Brasil, 2015), a acessibilidade refere-se a habilidade de uma pessoa, independentemente de suas limitações físicas ou de mobilidade, alcançar e utilizar espaços, móveis, equipamentos urbanos, edifícios, meios de transporte, informações e comunicações, incluindo seus sistemas e tecnologias associadas, bem como outros serviços e instalações acessíveis ao público.

A acessibilidade, portanto, incorpora a ideia de que todas as pessoas têm direito de serem incluídas na sociedade, independente de deficiências, localização geográfica, barreiras de linguagem ou outros fatores (Geraldo, 2016), abordando um grupo amplamente definido de usuários (ISO 9241-171, 2018):

- Pessoas com deficiências físicas, sensoriais e cognitivas presentes no nascimento ou adquiridas durante a vida;
- Idosos que podem se beneficiar de novos produtos e serviços, mas que apresentam capacidades físicas, sensoriais e cognitivas reduzidas;
- Pessoas com deficiências temporárias, como uma pessoa com um braço quebrado ou alguém que tenha esquecido seus óculos, e;
- Pessoas com dificuldades em situações particulares, como uma pessoa que trabalha em um ambiente barulhento ou que tem as duas mãos ocupadas por outras atividades.

No contexto de sistemas interativos, a acessibilidade está relacionada à capacidade do usuário de acessar um sistema e interagir com ele, sem que a interface imponha obstáculos (Barbosa; Silva, 2010). Entretanto, desenvolver interfaces que atendam às diversas necessidades dos usuários não é uma tarefa trivial, pois uma dificuldade percebida por uma pessoa com determinada deficiência pode não ser a mesma barreira enfrentada por outras pessoas com a mesma condição (Silva; Ferreira; Ramos, 2016).

Em se tratando de pessoas com deficiência visual, a incorporação da acessibilidade no design da interface é um requisito exigente (Khan; Khusro, 2019). Isso ocorre porque a movimentação e a interação de pessoas com deficiência visual com o computador difere significativamente dos procedimentos adotados por outros indivíduos (Mendonça et al., 2008), já que a tela e o mouse podem não ser úteis (Geraldo, 2016).

## **2.2 Tecnologia Assistiva**

Tecnologia Assistiva engloba dispositivos e tecnologias desenvolvidos para proporcionar independência, qualidade de vida e inclusão social para pessoas com deficiência (Bersch, 2008), auxiliando-as a superar as adversidades pelas quais podem passar quando precisam depender de suas próprias capacidades (Ismaili et al., 2017). Abrange a pesquisa, a fabricação e o uso de equipamentos, recursos ou estratégias utilizadas para potencializar as habilidades funcionais deficitárias das pessoas com deficiência (CAT, 2009), podendo ser tanto dispositivos de hardware quanto de software, desenvolvidos pela indústria ou personalizado pelos próprios usuários finais (Albusays; Ludi; Huenerfauth, 2017).

O desenvolvimento de TA se beneficia de avanços tecnológicos em campos já consolidados, envolvendo a colaboração de especialistas de diversas áreas do conhecimento (CAT, 2009).

Em se tratando de pessoas com deficiência visual, o aprimoramento dos recursos de Tecnologia Assistiva, aliado aos constantes avanços das TICs, desempenham um papel fundamental ao facilitar a integração desses usuários no ambiente digital, promovendo a igualdade de oportunidades na sociedade (Guimarães et al., 2016). Isso possibilita um acesso mais eficaz e abrangente às informações e serviços (Mattheiss et al., 2017), proporcionando liberdade e independência para que pessoas com deficiência visual possam realizar tarefas difíceis, sem depender da assistência de quem enxerga (Albusays; Ludi; Huenerfauth, 2017).

Entretanto, nesses ambientes, a disponibilização e a apreensão das informações ocorrem, principalmente, por meio da visão (Torres; Costa; Lourenço, 2016), o que pode representar um desafio para pessoas com deficiência visual, visto que a tela e o mouse podem não ter utilidade (Geraldo, 2016). Por esse motivo, em algumas situa-

ções, mesmo que apoiados por TA, pessoas com deficiência visual podem encontrar dificuldades ao interagir com as interfaces digitais, já que a maioria dos recursos disponíveis foca apenas na extensão da interface gráfica tradicional (Hadwen-Bennett; Sentance; Morrison, 2018).

Neste sentido, no intuito de compreender quais são os principais recursos de TA mais utilizados por pessoas com deficiência visual para interagir com sistemas digitais, além de elencar os principais desafios e limitações enfrentados por esses usuários, foi conduzida uma Revisão Sistemática da Literatura (RSL). Os resultados obtidos permitiram afirmar que os leitores de tela destacam-se como o recurso de TA mais utilizado por pessoas com deficiência visual, especialmente cegos. Além disso, apesar dos muitos avanços que têm sido realizados no desenvolvimento de TA para auxiliar o acesso às TICs por esse grupo de usuários, esses recursos ainda carecem de pesquisas dedicadas a aperfeiçoá-los. Dentre os principais desafios identificados, destacam-se (Zen et al., 2022, 2023):

- A navegação por meio de leitores de tela é mais limitada do que a navegação visual, e os usuários com deficiência visual podem experimentar sobrecarga de informações (Alves; Cardoso; Freire, 2018), (Jariwala; Marghitu; Chapman, 2020). Isso ocorre, principalmente, porque os leitores de tela produzem fluxos de texto lineares (Khurana et al., 2018), realizando uma leitura sequencial do conteúdo, começando pela parte superior à esquerda e seguindo pela direita (Abreu et al., 2019);
- A maioria dos aplicativos apresenta GUIs (do inglês *Graphical User Interface*) visualmente densas que são mais adequadas para interação com dispositivo de indicação visual, como mouse ou telas sensíveis ao toque (Jariwala; Marghitu; Chapman, 2020);
- Os elementos de design multimídia não são completamente comunicados ao usuário cego por meio do uso de áudio (Muniandy; Sulaiman, 2017), o que torna a tarefa de navegar em uma tela lenta e pesada. Além disso, o conteúdo visual, principalmente na *web*, pode não ter descrições textuais correspondentes, ou ainda, as informações textuais que existem podem ser intercaladas com conteúdo irrelevante (Vanukuru, 2020);
- Os leitores de tela não conseguem distinguir o tipo de aplicativo que está sendo utilizado, comunicar o conteúdo de uma imagem ou descrever o *layout* de uma interface (Muniandy; Sulaiman, 2017);
- Em se tratando de navegação *web*, mesmo que auxiliados por atalhos de teclado, os usuários de leitores de tela frequentemente precisam pular sequencialmente

por uma longa lista de elementos até chegar ao item desejado (Aydin et al., 2020).

- Usuários de leitores de tela têm à sua disposição uma série de atalhos, que ajudam a navegar no conteúdo de maneira mais rápida. No entanto, pode ser difícil lembrar de todos eles (Argyropoulos; Paveli; Nikolarazi, 2019) e, muitas vezes, eles se sobrepõem aos atalhos dos aplicativos utilizados;
- A utilização de leitores de tela pode ser prejudicada se o usuário estiver em um ambiente barulhento ou com privacidade limitada (Khan et al., 2020).
- A maioria dos pesquisadores, projetistas e desenvolvedores possuem visão normal e, portanto, geralmente uma compreensão limitada das necessidades das pessoas com deficiência visual (Mattheiss et al., 2017). Além disso, as limitações diárias vividas por essa população podem não ser completamente compreendidas por pessoas com visão normal (Velázquez; Sánchez; Pissaloux, 2016).

Por meio da RSL executada, também foi possível verificar que alguns usuários com deficiência visual utilizam *Display Braille*<sup>1</sup> para interagir com sistemas digitais. No entanto, devido ao alto custo e à dificuldade de acesso, são poucos os usuários que conseguem adquiri-los.

O protocolo adotado para a execução da Revisão Sistemática da Literatura, bem como os detalhes a respeito dos dados obtidos foram publicados em Zen et al. (2022) e Zen et al. (2023).

Considerando que os leitores de tela se destacam como o principal recurso de TA utilizado por pessoas com deficiência visual, compreender as características específicas desses softwares é fundamental para entender as dificuldades enfrentadas por estudantes cegos ao programar utilizando esses recursos. A próxima seção, portanto, apresenta informações a respeito do funcionamento e a aplicação desses dispositivos.

## 2.3 Leitores de Tela

Leitores de telas são softwares comumente usados por pessoas com deficiência visual para interagir com sistemas computacionais, pois auxiliam os usuários a acessar o conteúdo exibido na tela (Vanukuru, 2020). Esses programas convertem a informação textual exibida em um discurso de voz sintetizado, permitindo que o usuário ouça a informação (Dias; Dias, 2019).

A leitura feita pelo leitor de telas normalmente se dá de maneira sequencial, começando pela parte superior à esquerda e avançando para a direita (Abreu et al., 2019).

---

<sup>1</sup>Dispositivo eletromecânico que traduz informações visuais em caracteres Braille (Albusays; Ludi; Huenerfauth, 2017), atualizados dinamicamente ao longo do tempo (Leonardis; Claudio; Frisoli, 2017).

Esses softwares permitem que os usuários naveguem pelos elementos da interface gráfica usando atalhos de teclado ou gestos específicos. Além disso, a maioria dos leitores de tela oferece combinações de teclas de atalho para acelerar o processo de navegação e pular blocos de conteúdo irrelevante (Aydin et al., 2020).

Entre os leitores de tela mais amplamente reconhecidos e utilizados, destacam-se:

- *JAWS (Job Access With Speech)*. Desenvolvido pela empresa *Freedom Scientific* para plataforma *Microsoft Windows*, fornece saída de fala em diversos idiomas e *Displays Braille* (Freedomscientific, 2022);
- *NVDA (NonVisual Desktop Access)*. Desenvolvido pela *NV Access* com auxílio de uma comunidade global de colaboradores, é gratuito e de código aberto para plataforma *Microsoft Windows* (NVDA, 2022). Um grande vantagem é o fato de não precisar ser instalado, podendo ser utilizado por meio de uma mídia portátil;
- *Virtual Vision*. Software brasileiro desenvolvido pela empresa *MicroPower* para *Microsoft Windows* (Micropower, 2022);
- *Orca*. Software livre, de código aberto, para plataformas *Linux* e *Solaris*, que fornece saída por fala e *Display Braille* (Gnomehelp, 2022);
- *DosVox*. Desenvolvido pela Universidade Federal do Rio de Janeiro (UFRJ), realiza a comunicação através de síntese de voz em língua portuguesa para plataforma *Microsoft Windows*. Grande parte das mensagens sonoras emitidas é feita em voz humana gravada (Intervox, 2002);
- *Narrador*. Aplicativo de leitura de tela integrado ao Sistema Operacional *Microsoft Windows*, versões 10 e 11 (Microsoft, 2024);
- *VoiceOver*. Leitor de telas para plataformas *watchOS*, *iOS*, *iPadOS* e *macOS*, permitindo saída através de fala sintetizada e *Display Braille* (Apple, 2022); e,
- *TalkBack*. Leitor de telas desenvolvido pela Google que está incluso em dispositivos móveis que utilizam a plataforma *Android* (Google, 2022).

Além dos recursos básicos oferecidos pelos leitores de tela, cada software pode disponibilizar um conjunto diversificado de funcionalidades adicionais. Alguns programas podem oferecer um conjunto mais amplo de recursos integrados, enquanto outros proporcionam a flexibilidade de desenvolver e adicionar *plug-ins* para estender suas funcionalidades. Além disso, esses softwares permitem que os usuários ajustem as configurações de acordo com suas preferências e necessidades individuais, incluindo a capacidade de alterar a velocidade e o timbre da voz sintetizada, configurar atalhos

de teclado personalizados e adaptar o comportamento do software para melhor se adequar ao seu fluxo de trabalho.

No contexto de utilização de IDEs, uma série de limitações persiste quando os usuários contam com leitores de tela para interagir com essas ferramentas. A natureza intrínseca desses recursos de TA, que fornecem informações de maneira sequencial aos seus usuários, combinada com o design do conteúdo na interface gráfica dessas ferramentas e as especificidades das linguagens de programação, impõe diversos desafios de navegação tanto para programadores profissionais quanto para estudantes com deficiência visual (Mountapmbeme; Okafor; Ludi, 2022). Tais desafios ressaltam a importância de analisar as principais características dos IDEs, o que será apresentado na próxima Seção.

## **2.4 Ambientes de Desenvolvimento Integrado (IDEs)**

IDEs são ferramentas que fornecem uma ampla gama de recursos e funcionalidades para auxiliar os programadores em várias etapas do desenvolvimento de software (Murphy, 2019), auxiliando na melhoria da produtividade e eficiência de analistas e desenvolvedores de sistemas (Potluri et al., 2018).

Esses recursos desempenham um papel importante no processo moderno de desenvolvimento de software (Albusays; Ludi; Huenerfauth, 2017), pois combinam editor, compilador, interpretador, depurador, sistema de versionamento, suporte a formatação e ferramentas de modelagem em um único ambiente. Entretanto, proporcionar todos esses recursos resulta em interfaces complexas, incluindo estruturas de menus encapsuladas, inúmeras janelas e extensas funcionalidades que podem ser confusas para os usuários com deficiência visual (Petrausch; Loitsch, 2017).

Os editores de texto nesses sistemas geralmente incluem recursos visuais como indentação para indicar o nível de escopo, cores diferentes para realçar a sintaxe e vários outros recursos para ajudar os programadores a compreender a estrutura do código e navegar por ele com mais facilidade (Albusays; Ludi; Huenerfauth, 2017). Muitas dessas informações são transmitidas por meio de metáforas visuais em IDEs e podem não ser acessíveis aos leitores de tela. Essa acessibilidade inadequada dificulta, ou mesmo impede, que usuários com deficiência visual façam uso de muitos recursos oferecidos por esses ambientes (Potluri et al., 2018).

Além disso, os usuários de leitores de tela também enfrentam a limitação de acesso a recursos mais avançados nas IDEs, e levam mais tempo para se moverem em uma grande base de código. Frequentemente, são obrigados a percorrer o código linha por linha ou recorrer aos recursos de busca para localizar diferentes partes do código-fonte (Baker; Milne; Ladner, 2015). Mealin; Murphy-hill (2012) destacam ainda, que muitas linguagens de programação fazem uso extensivo de caracteres não alfanuméricos,

como colchetes, parênteses e chaves, o que pode representar um desafio quando se utiliza leitores de tela como recurso de Tecnologia Assistiva.

Outro aspecto a considerar é a complexidade da sintaxe de muitas linguagens de programação, que pode resultar em um maior número de erros de digitação e dificultar o processo de depuração do código.

Além desses desafios, Noschang et al. (2014) destacam que estudantes de computação brasileiros possuem baixa fluência no idioma inglês, o que cria uma barreira adicional para aprender a programar e utilizar IDEs.

No Brasil, embora não existam índices oficiais que caracterizem a falta de fluência em língua inglesa entre os estudantes que concluem o ensino médio, o índice EPI (*English Proficiency Index*), que classifica países e territórios por suas habilidades de inglês, coloca o Brasil em 70º lugar dentre 113 nações pesquisadas em 2023 (EPI, 2023).

Outra dificuldade enfrentada pelos estudantes está relacionada à compreensão das mensagens de erro exibidas pelos IDEs. Além de serem apresentadas em língua inglesa, muitos compiladores/interpretadores geram mensagens de erro genéricas que podem não ser úteis para orientá-los na correção dos erros (Noschang et al., 2014).

Diante deste contexto, Noschang et al. (2014) defendem a utilização de IDEs projetadas com finalidade didática nas disciplinas introdutórias de programação de computadores, visando mitigar esses problemas e permitir que o tempo e a energia dos estudantes sejam direcionados para a compreensão da lógica de programação em si.

No caso específico de estudantes cegos, esses obstáculos podem ser ainda mais desafiadores, pois mesmo que utilizem ferramentas de cunho didático, enfrentam desafios relacionados à acessibilidade, o que pode limitar ainda mais sua capacidade de interação e compreensão do ambiente de desenvolvimento. Por outro lado, se reconhece a importância de possibilitar que os esses estudantes conheçam as ferramentas de desenvolvimento de software profissional e aprendam a utilizar com qualidade e eficiência todos os recursos oferecidos por elas. Essa familiarização pode contribuir substancialmente para a sua formação profissional, preparando-os para o mercado de trabalho, além de proporcionar uma maneira de realizar tarefas com maior produtividade.

Justifica-se essa afirmação a partir do estudo de Mealin; Murphy-hill (2012), em que os autores identificaram diversas dificuldades enfrentadas por programadores cegos ao desempenhar tarefas de programação profissional (como a complexidade de uso e falta de acessibilidade dos IDEs) e argumentam que podem ser resultado da falta de treinamento no uso dessas tecnologias durante a sua formação acadêmica, afetando sua capacidade de utilizá-las adequadamente no desempenho de suas atividades profissionais.

## 2.5 Abordagens para Elaboração de Diretrizes de Acessibilidade

A área de Interação Humano-Computador (IHC) apresenta uma vasta quantidade de conjuntos de normas, padrões, princípios, recomendações e heurísticas que oferecem sugestões para direcionar o processo de desenvolvimento de sistemas interativos (Barbosa; Silva, 2010). Esses documentos podem abordar aspectos como usabilidade, acessibilidade ou experiência do usuário, representando uma maneira eficaz de integrar critérios de design derivados de experiências bem-sucedidas compartilhadas por outros profissionais (Abascal; Nicolle, 2005).

Enquanto as normas são necessariamente permanentes, as diretrizes são dinâmicas e podem ser modificadas e melhoradas quando surgem novos conhecimentos ou, no caso específico da acessibilidade, quando novas barreiras, desafios ou limitações são identificadas. À medida que seu escopo de aplicação se estabiliza, as diretrizes podem eventualmente se tornar padrões ou normas (Abascal; Nicolle, 2005).

Diretrizes podem ser genéricas, aplicáveis a qualquer sistema interativo, ou específicas, concentrando-se em determinadas categorias, plataformas tecnológicas ou produtos individuais (Barbosa; Silva, 2010). No entanto, diretrizes genéricas podem não contemplar as características das aplicações de domínio específico, que demandam de recomendações adaptadas às suas particularidades. Isso ressalta a importância de desenvolver diretrizes direcionadas aos desafios específicos de determinados domínios de aplicação (Quiñones; Rusu; Rusu, 2018).

Nesse contexto, diante da necessidade de proporcionar uma experiência de aprendizado inclusiva e eficaz de programação de computadores para estudantes cegos, preparando-os adequadamente para a atuação profissional, foram desenvolvidas diretrizes específicas de acessibilidade em IDEs para esses usuários. Esse processo de desenvolvimento se fundamentou em três estudos, os quais embasaram as diferentes etapas envolvidas na concepção do conjunto apresentado nesta Tese.

### 2.5.1 Embasamento para a Análise Qualitativa

A análise qualitativa conduzida neste Estágio da pesquisa se fundamentou nos estudos publicados por Baker; Bennett; Ladner (2019) e Mountapmbeme; Okafor; Ludi (2022), que serão descritos a seguir.

Por meio de entrevistas semiestruturadas, Baker; Bennett; Ladner (2019) investigaram as barreiras enfrentadas por estudantes cegos matriculados em cursos na área de Computação. Os resultados dessa pesquisa revelaram uma ampla gama de problemas encontrados por eles para concluírem seus cursos, os quais foram agrupados em três categorias distintas:

1. **Tecnologias.** Envolve a identificação e análise dos recursos tecnológicos essenciais utilizados pelos estudantes, incluindo leitores de tela, IDEs e outras

ferramentas relevantes.

2. **Acomodações.** Aborda as dificuldades enfrentadas pelos estudantes durante as aulas, no acesso aos materiais de apoio e na realização das atividades.
3. **Impactos Sociais.** Relacionado à influência que as atitudes dos professores e dos demais alunos têm sobre a experiência educacional desses estudantes.

A Categoria "**Acomodações**" foi a mais extensivamente abordada por Baker; Bennett; Ladner (2019), sendo subdividida pelos autores em três subcategorias distintas:

1. **Aulas/Palestras (*Lectures*).** Abrange problemas gerados porque a pessoa que está se comunicando não verbaliza completamente o que está escrevendo no quadro, digitando no computador, ou ainda, deixa de descrever gestos ou ações utilizados para explicar o conteúdo. Quando o estudante cego não tem acesso ao mesmo conjunto de informações que seus colegas, sua compreensão e participação podem ser comprometidas;
2. **Materiais (*Materials*).** Associado à dificuldade para obter os materiais de apoio em um formato acessível e em tempo hábil; e,
3. **Tarefas/Atividades (*Assignments*).** Ocorre quando o estudante recebe tarefas distintas do restante da turma, utiliza recursos ou ferramentas diferentes dos colegas, depende de assistência humana, recebe notas baseadas apenas nos componentes acessíveis ou é dispensado da realização de algumas atividades.

O estudo realizado por Baker; Bennett; Ladner (2019) proporcionou uma base fundamental para a identificação do conjunto de Categorias e Códigos adotados na análise qualitativa dos desafios e obstáculos educacionais enfrentados por estudantes cegos descrita no Capítulo 4.

O trabalho publicado por Mountapmbeme; Okafor; Ludi (2022), por outro lado, descreve uma RSL que buscou identificar os esforços de pesquisa de acessibilidade voltados para tornar os IDEs acessíveis para programadores profissionais com deficiência visual e alunos com deficiência visual aprendendo a programar. Foram analisados 70 artigos publicados entre 2000 e 2020 com o propósito identificar as barreiras de acessibilidade enfrentadas por estudantes e programadores profissionais e as soluções propostas para lidar com essas barreiras. Os resultados obtidos na RSL foram organizados em quatro Categorias (Mountapmbeme; Okafor; Ludi, 2022):

- **Paradigmas de Programação.** Referente aos diferentes estilos ou abordagens de programação: baseada em texto, baseada em blocos, tangível e audível;

- **Tarefas e Desafios de Programação.** Investigou um conjunto de barreiras identificadas no processo de programação: compreensão, navegação, depuração, edição e revisão de código-fonte;
- **Tecnologia Assistiva.** Analisou os principais recursos tecnológicos utilizados por pessoas com deficiência visual para programar: leitor de tela, *Display Braille*, teclado e software ampliador de tela; e,
- **Mecanismos de interação.** Identificou os principais dispositivos de entrada e saída utilizados por pessoas com deficiência visual.

A categoria "Tarefas e Desafios de Programação" foi amplamente analisada pelos autores, que organizaram as informações obtidas nos seguintes códigos (Mountapbeme; Okafor; Ludi, 2022):

- **Compreensão do código.** Relacionado aos métodos utilizados para se obter uma compreensão do código-fonte do programa, sua estrutura e a lógica adotada para a implementação.
- **Navegação no código.** Diz respeito às estratégias para se obter um entendimento geral do código, o que ele faz e qual é a sua estrutura, envolvendo barreiras associadas à: leitura linha por linha, em sequência, realizadas pelos leitores de tela; necessidade de procurar informações no código sem perder a posição de foco do cursor; navegação no ambiente de programação; e, localização dos recursos disponibilizados pelos IDEs.
- **Depuração de código.** Aborda questões relativas a tarefas típicas de depuração, como encontrar e compreender erros no código ou analisar o comportamento do programa.
- **Edição de código.** Envolve a dificuldade em identificar o uso da indentação para indicar o nível de escopo, a utilização de cores distintas para destacar a sintaxe do código, o uso extensivo de caracteres não alfanuméricos e a complexidade das instruções e comandos das linguagens de programação.
- **Skimming<sup>2</sup> de código.** Aborda a dificuldade em obter uma visão geral de alto nível do código. Os problemas decorrem da leitura sequencial (linha por linha) realizada pelos leitores de tela ou mesmo pela falta de acesso a alguns recursos do IDE (como dobra de código, que permite recolher/expandir uma seção de código).

---

<sup>2</sup>Não foi encontrada uma tradução adequada para o termo "*Skimming*", preferindo-se mantê-lo em língua inglesa.

Os códigos relacionados na Categoria "Tarefas e Desafios de Programação" foram especialmente úteis para esta Tese de Doutorado, pois foram adotados na análise qualitativa descrita no Capítulo 4 e, posteriormente, contribuíram para a definição das categorias utilizadas na classificação das diretrizes propostas (Capítulo 5).

## 2.5.2 Embasamento Metodológico

Diversas metodologias têm sido propostas e podem ser adotadas para a elaboração de um novo conjunto de diretrizes de usabilidade (Nasr; Alsaggaf; Sinnari, 2023). Nesta Tese de Doutorado, o processo de elaboração das diretrizes se fundamentou na metodologia proposta por Quiñones; Rusu; Rusu (2018), que descreve um processo formal de desenvolvimento e validação de diretrizes de usabilidade e/ou *User eXperience* (UX), incluindo oito estágios (Figura 2):

1. **Estágio Exploratório.** Abrange a coleta de informações relevantes para a elaboração do novo conjunto de heurísticas. Essas informações podem ser obtidas de diferentes fontes, tais como artigos, livros, experimentos, etc.;
2. **Estágio Experimental.** Trata-se de um estágio opcional, onde é preciso decidir se serão analisados apenas os dados obtidos no Estágio Exploratório ou se serão realizados novos experimentos para coletar informações complementares que não tenham sido identificadas;
3. **Estágio Descritivo.** Nesta etapa, é necessário selecionar e destacar os tópicos mais relevantes de todas as informações coletadas nas etapas anteriores, a fim de formalizar os principais conceitos associados à pesquisa;
4. **Estágio Correlacional.** A fase em que são determinadas as características das diretrizes a serem elaboradas e como elas serão categorizadas, agrupando-as de acordo com aspectos específicos. A criação de categorias não apenas organiza as informações coletadas, mas também contribui para a redução da complexidade do conjunto de dados;
5. **Estágio de Seleção.** Envolve o processo de manter, adaptar e/ou descartar os conjuntos de heurísticas selecionadas nas etapas anteriores. Pode ocorrer que algumas heurísticas existentes cubram parcialmente as características e atributos, enquanto outras não abordem nenhum dos elementos necessários, implicando na necessidade de criar novas heurísticas para avaliar esses elementos;
6. **Estágio de Especificação.** Abrange a formalização do novo conjunto de diretrizes propostas usando um modelo padrão.
7. **Estágio de Validação.** Esta etapa especifica o processo de validação do conjunto de diretrizes propostas por meio de experimentos. Por se tratar de uma

etapa crítica do processo, Quiñones; Rusu; Rusu (2018) propõem três tipos de validação: (1) avaliação heurística; (2) parecer de especializado; e, (3) teste de usuário.

8. **Estágio de Refinamento.** Envolve refinar e melhorar o novo conjunto de heurísticas com base nos resultados obtidos no Estágio de Validação.

Figura 2 – Modelo e notação de processos de negócios



Fonte: Adaptada de Quiñones; Rusu; Rusu (2018)

Embora as etapas sejam apresentadas sequencialmente na Figura 2, Quiñones; Rusu; Rusu (2018) afirmam que o desenvolvimento das diretrizes pode ser realizado de forma iterativa. Inclusive, em circunstâncias específicas, é possível que: (1) alguns estágios se tornem opcionais, não sendo realizadas por algum motivo; (2) alguns estágios se sobreponham e sejam executados simultaneamente; e (3) um estágio possa ser interrompido, permitindo retornar a um estágio anterior.

Ainda, Quiñones; Rusu; Rusu (2018) afirmam que é possível repetir todas as etapas, algumas etapas ou apenas uma etapa da metodologia, bem como realizar quan-

tas iterações forem necessárias. A metodologia também não determina que o processo finalize na Etapa 8 (Estágio de Refinamento).

## 2.6 Considerações Parciais

Este Capítulo apresentou os conceitos considerados mais relevantes para a elaboração de diretrizes destinadas a promover a acessibilidade em Ambientes de Desenvolvimento Integrado para estudantes cegos. Entende-se que a concepção dessas recomendações requer a identificação dos principais recursos de Tecnologia Assistiva utilizados por eles, bem como compreender o funcionamento desses recursos e suas limitações. Além disso, é fundamental identificar as características específicas dos IDEs e como elas influenciam na interação com a Tecnologia Assistiva.

Por fim, foram detalhados os estudos que serviram de fundamentação para as diferentes etapas envolvidas na elaboração do conjunto de Diretrizes de Acessibilidade em IDEs para estudantes cegos propostas nesta Tese.

O protocolo e os resultados alcançados na RSL citada na Seção 2.2, culminaram em duas publicações:

- ZEN, Eliana et al. *Assistive technology to assist the visually impaired in the use of icts: A systematic literature review. In: XVIII Brazilian Symposium on Information Systems. 2022. p. 1-8.*  
<https://dl.acm.org/doi/10.1145/3535511.3535529>.
- ZEN, Eliana et al. *Tecnologia Assistiva para auxiliar a interação entre pessoas com deficiência visual e sistemas computacionais: Um Mapeamento Sistemático da Literatura. iSys-Brazilian Journal of Information Systems, v. 16, n. 1, p. 6: 1-6: 27, 2023.*  
<https://sol.sbc.org.br/journals/index.php/isys/article/view/2890>.

## **3 PERCURSO METODOLÓGICO**

Um método consiste de uma série de passos que se utiliza para obter um conhecimento confiável, livre da subjetividade do pesquisador e o mais próximo possível da objetividade empírica (Gil et al., 2002). Neste sentido, este Capítulo descreve todas as etapas executadas e abordagens metodológicas adotadas para a construção do conjunto de Diretrizes de Acessibilidade em IDEs para estudantes cegos de que trata esta Tese.

### **3.1 Caracterização da Pesquisa**

Quanto à abordagem a pesquisa caracteriza-se como qualitativa, pois busca compreender e explicar os significados, contextos e características dos dados coletados (Gerhardt; Silveira, 2009). Neste tipo de pesquisa, pretende-se identificar e analisar as experiências e percepções dos participantes, contribuindo para uma compreensão da temática estudada (Gibbs, 2009).

Quanto à natureza, a pesquisa pode ser classificada como aplicada, já que pretende gerar conhecimentos para aplicação prática (Gerhardt; Silveira, 2009), identificando problemas e propondo soluções (Thiollent, 2022). Quanto aos objetivos, a pesquisa pode ser classificada como exploratória, porque visa estudar um determinado tema e proporcionar visão geral sobre o assunto (Gil et al., 2002).

Essa escolha metodológica é coerente com a natureza do problema investigado, que pretende compreender as experiências e percepções dos participantes, bem como propor soluções práticas para questões relacionadas à acessibilidade em IDEs para estudantes cegos.

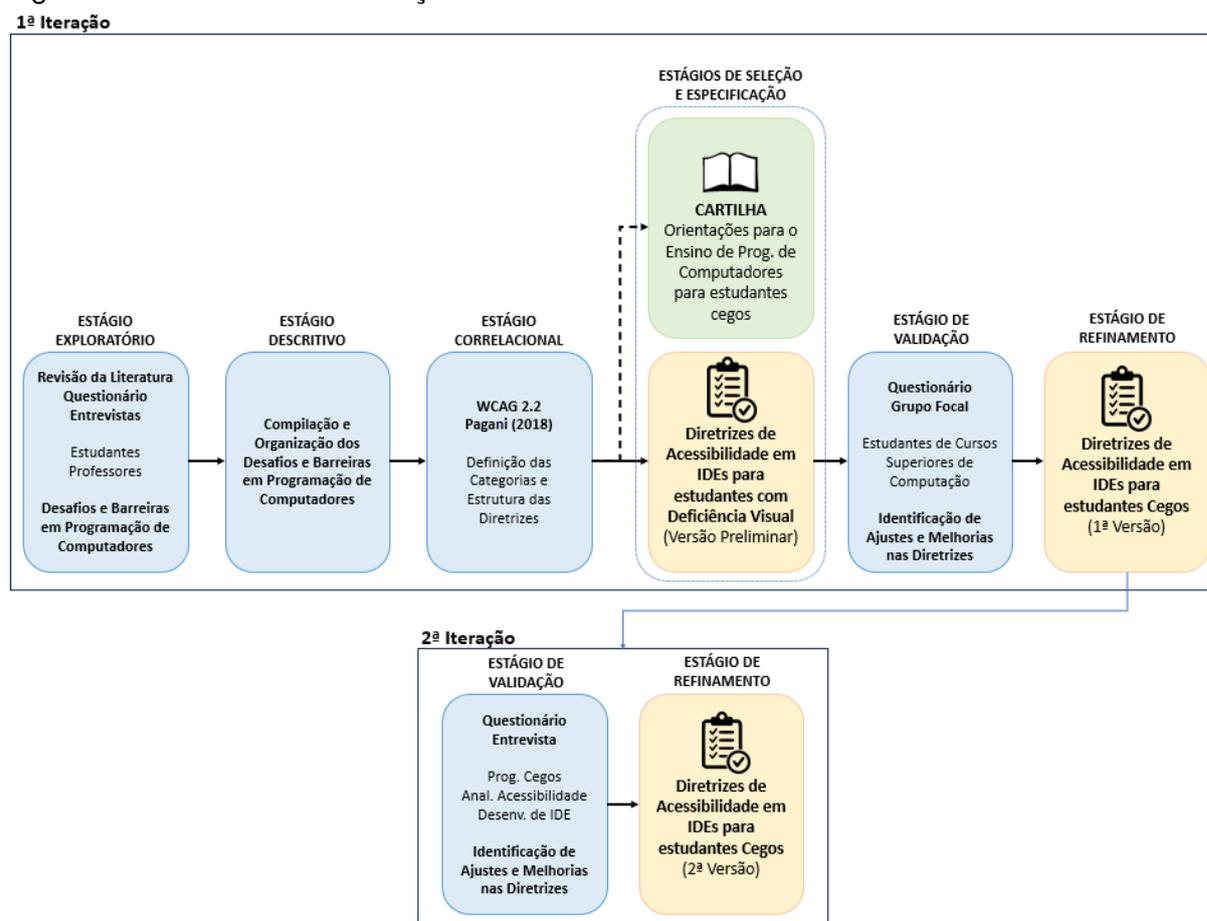
### **3.2 Processo de elaboração das Diretrizes**

Este trabalho foi conduzido predominantemente baseado em procedimentos de pesquisa bibliográfica e estudos exploratórios envolvendo questionários, entrevistas e grupo focal. O processo de elaboração da Diretrizes foi fundamentado na metodologia proposta por Quiñones; Rusu; Rusu (2018), descrita no Capítulo 2. Embora não trate

especificamente de diretrizes de acessibilidade, esta metodologia foi adotada porque fornece um protocolo de pesquisa claro e organizado. Além disso, a acessibilidade pode ser entendida como uma pré-condição para a usabilidade (Leporini; Paternò, 2004), visto que para que um usuário possa tirar proveito do apoio computacional oferecido por um sistema, não podem existir barreiras que o impeçam de interagir com sua interface (Barbosa; Silva, 2010).

A pesquisa contou com dois ciclos iterativos. Na primeira etapa da iteração, todos os estágios propostos por Quiñones; Rusu; Rusu (2018) foram executados, com exceção do Estágio Experimental, etapa opcional da metodologia. Na 2ª iteração, apenas os Estágios de Validação e Refinamento foram executados. A Figura 3 ilustra um resumo da execução dos Estágios e ciclos iterativos realizados para a elaboração das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos.

Figura 3 – Processo de elaboração das Diretrizes



Fonte: Adaptado de Quiñones; Rusu; Rusu (2018)

Mais detalhes sobre cada uma das etapas apresentadas na Figura 3 são apresentados nas subseções a seguir.

### 3.2.1 Estágio Exploratório

No Estágio Exploratório, Quiñones; Rusu; Rusu (2018) determinam que seja realizada a coleta de informações específicas relacionadas ao domínio de aplicação das diretrizes que serão desenvolvidas. Nesta pesquisa, o Estágio Exploratório envolveu a realização das seguintes atividades:

1. Revisão da literatura para identificar as principais barreiras encontradas por pessoas com deficiência visual ao aprender a programar, realizar tarefas associadas à programação de computadores e interagir com IDEs. Nesta etapa foi utilizada a metodologia *Backward snowballing*, onde a lista de referência de um artigo é usada para encontrar artigos relacionados; e,
2. Levantamento das dificuldades enfrentadas por estudantes com deficiência visual nas disciplinas de Programação de Computadores. Os dados foram obtidos por meio de questionários (enviados a professores que lecionaram estas disciplinas para estudantes com deficiência visual) e entrevistas com estudantes e egressos de Cursos de Computação que possuem deficiência visual;

As informações obtidas foram analisadas a partir de uma abordagem qualitativa, utilizando a técnica de análise de conteúdo, que abrange um conjunto de técnicas destinadas a analisar as comunicações, visando obter, por meio de procedimentos sistemáticos e objetivos de descrição do conteúdo, indicadores que possibilitem inferências sobre os conhecimentos relacionados às condições de produção/recepção das mensagens (Bardin, 2015). Os procedimentos adotados e os resultados obtidos no Estágio Exploratório são descritos no Capítulo 4.

### 3.2.2 Estágio Descritivo

No Estágio Descritivo os resultados identificados no Estágio Exploratório foram compilados e analisados. As *inputs* desta fase foram as barreiras e desafios encontrados por pessoas com deficiência visual para aprender a programar, realizar tarefas relacionadas à programação de computações e interagir com IDEs.

Os dados coletados foram organizados visando selecionar e enfatizar tópicos relacionados às barreiras de interação com IDEs destinados à programação baseada em texto e realizada por meio de utilização de leitor de tela como recurso de TA. Os resultados obtidos neste Estágio são apresentados no Capítulo 5.

### 3.2.3 Estágio Correlacional

No Estágio Correlacional, os dados obtidos nos Estágios Exploratório e Descritivo foram consolidados e categorizados para formar os atributos específicos do domínio da aplicação (Quiñones; Rusu; Rusu, 2018). Neste estudo, para desenvolver as

Diretrizes de Acessibilidade em IDEs para Estudantes Cegos, foram analisadas as recomendações apresentadas no conjunto de Diretrizes WCAG 2.2 (WCAG 2.2, 2023).

Este Estágio também contemplou a especificação das Categorias adotadas para a classificação das Diretrizes de Acessibilidade propostas, bem como a definição da estrutura das Diretrizes, que considerou aspectos como a organização, estrutura e forma de redação das WCAG 2.2. Além disso, o modelo proposto por Britto; Pizzolato (2018), que apresenta um guia de recomendações de acessibilidade para interfaces *web* com foco em aspectos do Autismo, também como utilizado como base. Nesse guia, cada Diretriz é estruturada da seguinte forma: (1) Descrição; (2) Por que realizar e por que é importante para a pessoa com Transtorno do Espectro Autista (TEA)?; (3) Como realizar?; (4) Exemplos; e (5) Referências. Este estudo foi adotado como referência nesta Tese de Doutorado por que justifica a importância de contextualizar a razão por trás da definição da diretriz em questão ("Por que realizar?") e fornece sugestões práticas de implementação ("Como Realizar?" e "Exemplos"). Entende-se que esses aspectos são fundamentais para aqueles que aplicarão a Diretriz na prática. Os procedimentos adotados e resultados obtidos são descritos no Capítulo 5.

### **3.2.4 Estágio de Seleção e Especificação**

Nesta pesquisa, o Estágio de Seleção e o Estágio de Especificação foram conduzidos conjuntamente, utilizando os resultados obtidos nos Estágios Exploratório, Descritivo e Correlacional, para elaboração da versão preliminar das diretrizes.

Todas as barreiras e desafios de interação com IDE enfrentados por pessoas com deficiência visual foram classificados nas Categorias definidas no Estágio Correlacional. Em seguida, essas dificuldades passaram pela fase de seleção, envolvendo o agrupamento, adaptação e criação de novas diretrizes. Nesta etapa, foi analisada a relação entre as dificuldades abordadas por cada categoria, verificando quais dessas dificuldades estariam relacionadas a uma mesma diretriz e quais gerariam novas diretrizes. Posteriormente, procedeu-se à especificação do texto das diretrizes preliminares. Os resultados obtidos são apresentados no Capítulo 5.

As informações obtidas nos Estágios Exploratório também permitiram elaborar um Cartilha que contém orientações para o ensino de Programação de Computadores para estudantes cegos, apresentada no Capítulo 5.

### **3.2.5 Estágio de Validação**

No Estágio de validação, optou-se por utilizar parecer especializado para validação das diretrizes. Esta técnica consiste em perguntar a especialistas, pesquisadores ou praticantes, sobre a utilidade, eficiência e eficácia do conjunto de heurísticas para avaliar um determinado produto.

Este Estágio foi conduzido em dois ciclos, cada um deles correspondendo a uma das iterações apresentadas na Figura 3. A 1ª iteração envolveu a participação de estudantes dos cursos de computação, tanto com deficiência visual quanto sem. Foram empregados questionários e grupos focais como métodos de avaliação. Os procedimentos adotados neste Estágio de Validação, assim como os resultados obtidos, são apresentados no Capítulo 6.

Na 2ª iteração do Estágio de Validação, também recorreu-se a parecer especializado para validar o conjunto de Diretrizes. Nessa fase, no entanto, participaram três programadores cegos, um analista de acessibilidade e um professor de curso de computação envolvido no desenvolvimento de um ambiente de programação. Para esta etapa da validação, foram utilizadas as técnicas de questionários e entrevista. Todas as etapas e resultados obtidos neste Estágio de Validação são apresentados no Capítulo 7.

### **3.2.6 Estágio de Refinamento**

Com base nos resultados obtidos no Estágio de Validação, as diretrizes foram refinadas, ajustadas e aprimoradas. Na 1ª iteração do Estágio de Refinamento, identificou-se a necessidade de excluir uma das diretrizes e incluir outra, que abordaria tópicos que não haviam sido contemplados na versão preliminar. Além disso, durante esta etapa da pesquisa, foram fornecidos subsídios que levaram ao direcionamento exclusivo das diretrizes para estudantes cegos, deixando de abordar todos os tipos de deficiência visual. Essas mudanças culminaram na elaboração da 1ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos. Esse processo é descrito em detalhes no Capítulo 7.

A avaliação da 1ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos não resultou na adição ou remoção completa de nenhuma das diretrizes apresentadas. Em vez disso, os participantes ressaltaram alguns aspectos pontuais que precisariam passar por revisão e adequação. Esses ajustes e melhorias resultaram na 2ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos. O Capítulo 7 descreve a condução desta execução do Estágio de Refinamento e o Capítulo 8 apresenta a 2ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos na íntegra.

## **3.3 Considerações Parciais**

Este Capítulo apresentou as abordagens e classificações desta pesquisa, bem como os métodos utilizados para atingir os objetivos propostos para esta Tese de Doutorado. Todas as etapas executadas para a obtenção do conjunto final de Diretrizes de Acessibilidade em IDEs para Estudantes Cegos são descritos em detalhes nos

próximos Capítulos.

É importante destacar que todos os dados foram coletados com consentimento dos participantes, que concordaram em ter suas respostas utilizadas desde que se garantisse o seu completo anonimato. Para garantir a conformidade com aspectos éticos, todas as etapas da pesquisa que envolvem a participação de seres humanos foram submetidas e aprovadas pelo Comitê de Ética em Pesquisa do Instituto Federal Farroupilha (RS), conforme Pareceres Consubstanciados n.º 54297421.9.0000.5574 (Apêndice A) e n.º 68361023.1.0000.5574 (Apêndice B).

Todos os participantes formalizaram seu consentimento ao assinar o Termo de Consentimento Livre e Esclarecido (TCLE) e, quando aplicável, o Termo para Uso da Imagem e/ou Som da Voz (TUIV) antes de iniciar os estudos conduzidos.

Conforme ilustrado na Figura 3, o processo de desenvolvimento gerou três versões das Diretrizes de Acessibilidade em IDEs:

1. Versão Preliminar das Diretrizes de Acessibilidade em IDEs para Estudantes com Deficiência visual, avaliada por estudantes de computação. Esta versão está disponível em: <https://elianazen.github.io/DiretrizesAcessibilidadeIDE/recomendacoesPreliminares.html>
2. 1ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos. Versão atualizada com base nos resultados da avaliação com estudantes de computação e direcionada para estudantes cegos. Disponível em: <https://elianazen.github.io/DiretrizesAcessibilidadeIDE/1VersaoDiretrizes.html>
3. 2ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos. Refinada após avaliação com programadores cegos, analista de acessibilidade e desenvolvedor de IDE, apresentada no Capítulo 8 e disponível em: <https://elianazen.github.io/DiretrizesAcessibilidadeIDE/2VersaoDiretrizes.html>

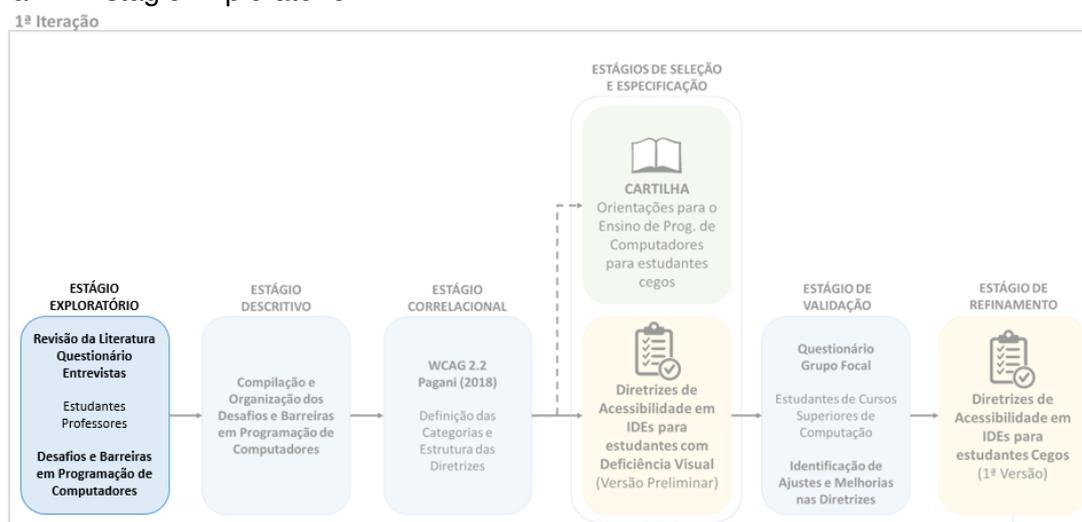
Cada versão das Diretrizes de Acessibilidade em IDEs para estudantes Cegos será abordada ao longo dos Capítulos que compõem esta Tese de Doutorado.

## 4 DESAFIOS E BARREIRAS DE INTERAÇÃO

A forte dependências de elementos gráficos nos cursos da área de Computação implica em barreiras e desafios para pessoas com deficiência visual, que podem se manifestar tanto durante o processo de ensino-aprendizagem quanto na execução de atividades profissionais e na utilização das tecnologias necessárias para programar (Rajaselvi et al., 2021).

Este Capítulo delinea os procedimentos adotados para identificar essas barreiras e desafios, correspondendo ao Estágio Exploratório definido na metodologia de Quiñones; Rusu; Rusu (2018), conforme evidenciado na Figura 4.

Figura 4 – Estágio Exploratório



Os resultados foram obtidos por meio de revisão da literatura *ad-hoc*, entrevistas e questionários. A análise dos resultados ocorreu mediante uma abordagem qualitativa, utilizando a técnica de Análise de Conteúdo proposta por Bardin (2015).

Esta etapa da pesquisa foi aprovada pelo Parecer Consubstanciado n.º 54297421.9.0000.5574 emitido pelo Comitê de Ética em Pesquisa com Seres Humanos no Instituto Federal Farroupilha (IFFar), disponível no Apêndice A.

## 4.1 Revisão da Literatura

A revisão da literatura conduzida nesta fase da pesquisa foi realizada de forma *ad hoc*, sendo guiada pela necessidade de explorar de maneira específica e direcionada as informações relevantes para os objetivos do estudo. Foi utilizada a metodologia *Backward Snowballing*, onde as referências citadas por um artigo são utilizadas para encontrar trabalhos relacionados.

O ponto de partida escolhido foi o estudo publicado por Mountapmbeme; Okafor; Ludi (2022), previamente detalhado na Subseção 2. Tal escolha se justifica pela abordagem de aspectos altamente similares aos objetivos desta análise e pela abrangência significativa da Revisão Sistemática da Literatura (RSL) realizada pelos autores, que analisaram 70 artigos publicados no período entre 2000 e 2020.

### 4.1.1 Caracterização do Estudo

Para a análise proposta neste trabalho, o processo inicial de revisão da literatura envolveu a leitura dos títulos, resumos e palavras-chave dos 70 artigos citados por Mountapmbeme; Okafor; Ludi (2022), culminando na seleção de 18 artigos para análise mais aprofundada: Albusays; Ludi (2016); Albusays; Ludi; Huenerfauth (2017); Albusays (2020); Alotaibi; Al-khalifa; Alsaeed (2020); Armaly; Rodeghero; Mcmillan (2018); Baker; Milne; Ladner (2015); Milne; Ladner (2018); Huff et al. (2020); Ludi (2015); Ludi; Spencer (2017); Mealin; Murphy-hill (2012); Mountapmbeme; Ludi (2020); Potluri et al. (2018); Schanzer; Bahram; Krishnamurthi (2019); Smith et al. (2003); Stefik et al. (2009); Stefik; Hundhausen; Patterson (2011); e, Utreras; Pontelli (2020).

Em seguida, procedeu-se à leitura completa dos 18 trabalhos selecionados. Nessa fase, foram priorizados os estudos que exploravam desafios e obstáculos enfrentados por pessoas com deficiência visual ao interagir com IDEs baseados em texto, especialmente quando utilizados em conjunto com leitores de tela. Artigos que abordavam IDEs desenvolvidos para programação em blocos ou híbridos, bem como aqueles que faziam uso de outros recursos de Tecnologia Assistiva, foram deliberadamente excluídos do escopo da análise.

A leitura completa dos 18 artigos inicialmente selecionados permitiu verificar que apenas 11 deles efetivamente abordavam uma ou mais obstáculos enfrentadas por pessoas com deficiência visual ao utilizar leitores de tela em IDEs baseadas em texto. Portanto, após a exclusão desses sete trabalhos, os artigos selecionados para integrar o corpo de referência desta etapa da pesquisa foram os seguintes: Albusays; Ludi (2016), Albusays; Ludi; Huenerfauth (2017), Alotaibi; Al-khalifa; Alsaeed (2020), Armaly; Rodeghero; Mcmillan (2018), Baker; Milne; Ladner (2015), Huff et al. (2020), Mealin; Murphy-hill (2012), Potluri et al. (2018), Schanzer; Bahram; Krishnamurthi

(2019), Smith et al. (2003) e Stefik et al. (2009).

Posteriormente, utilizando novamente a metodologia *Backward Snowballing*, foram examinados os artigos referenciados pelos 11 trabalhos selecionados, considerando inicialmente título, resumo e palavras-chave. Essa estratégia possibilitou identificar mais um estudo que discute as dificuldades enfrentadas por pessoas com deficiência visual na área de programação: Siegfried (2006).

Além deste estudo, também foi incorporado ao conjunto de artigos que seriam analisados o trabalho de Baker; Bennett; Ladner (2019), descrito no Capítulo 2. Embora não explore exaustivamente o assunto, o artigo aborda alguns desafios relacionados à interação com IDEs enfrentados por estudantes. É importante ressaltar que esse trabalho não foi mencionado em nenhum dos 11 artigos inicialmente selecionados.

Dado que a RSL publicada por Mountapmbeme; Okafor; Ludi (2022) abrange artigos publicados até 2020, realizou-se uma nova pesquisa utilizando a ferramenta Google Scholar<sup>1</sup>, a fim de verificar se haviam sido publicados novos trabalhos entre os anos de 2021 e 2023 que abordassem a temática em questão. Uma combinação de termos relevantes identificados a partir das palavras-chave presentes nos artigos analisados foi utilizada para formar a *string* de busca utilizada:

*(Computer Programming OR Computer development) AND (Integrated Development Environment OR IDE) AND (Visual Impairment OR blind OR blindness)*

Como resultado, foram identificados mais três estudos: Kearney-volpe; Hurst (2021), Pandey et al. (2022) e Sampath; Merrick; Macvean (2021). Novamente, recorreu-se à metodologia *Backward Snowballing* para examinar os artigos referenciados por estes trabalhos selecionados, resultando na inclusão de dois outros estudos que abordavam temas relacionados aos objetivos da revisão da literatura: (1) Petrausch; Loitsch (2017), citado por Pandey et al. (2022); e, (2) Hutchinson; Metatla (2018), citado por Pandey et al. (2022) e Sampath; Merrick; Macvean (2021).

Ao todo, portanto, foram analisados 18 trabalhos, abrangendo diferentes perspectivas e contribuições: Albusays; Ludi (2016), Albusays; Ludi; Huenerfauth (2017), Alo-taibi; Al-khalifa; Alsaeed (2020), Armaly; Rodeghero; Mcmillan (2018), Baker; Milne; Ladner (2015), Baker; Bennett; Ladner (2019), Huff et al. (2020), Mealin; Murphy-hill (2012), Potluri et al. (2018), Schanzer; Bahram; Krishnamurthi (2019), Smith et al. (2003), Stefik et al. (2009), Siegfried (2006), Kearney-volpe; Hurst (2021), Pandey et al. (2022), Sampath; Merrick; Macvean (2021), Hutchinson; Metatla (2018) e Petrausch; Loitsch (2017).

---

<sup>1</sup><https://scholar.google.com/>

### 4.1.2 Resultados

As informações derivadas da revisão da literatura foram submetidas a uma análise qualitativa utilizando a técnica de Análise de Conteúdo proposta por Bardin (2015). A definição das Categorias e Códigos envolveu uma abordagem tanto dedutiva quanto indutiva. A abordagem dedutiva foi empregada quando as citações estavam diretamente relacionadas à Categoria "Tarefas e Desafios de Programação" e seus respectivos Códigos, predefinidos por Mountapmbeme; Okafor; Ludi (2022): (1) Compreensão do Código; (2) Depuração de Código; (3) Navegação no Código; (4) Edição de Código; e, (5) *Skimming* de Código. Já a abordagem indutiva foi utilizada para classificar elementos que não se encaixavam nas categorias preestabelecidas, gerando os seguintes Códigos adicionais: (6) Compreensão da Saída; (7) Sobrecarga Auditiva; e, (8) Leitura de código-fonte.

Os resultados obtidos nesta etapa da pesquisa comprovam que a maioria dos desafios de interação enfrentados por estudantes com deficiência visual ao programar e interagir com IDEs está associada ao fato de que os leitores de tela navegam pela interface gráfica e pelo código-fonte de forma linear, obrigando o usuário a percorrer o código uma linha de cada vez, em sequência (Albusays; Ludi; Huenerfauth, 2017; Mountapmbeme; Okafor; Ludi, 2022). Esse modo de interação dificulta o aprendizado e a realização de atividades relacionadas à programação de computadores por estudantes com deficiência visual (Albusays; Ludi, 2016), resultando em vários problemas de interação, o quais são apresentados na Tabela 1. Nesta Tabela, cada desafio é identificado por um código formado por letras e números. As letras indicam o método utilizado para o levantamento da informação (RL, indicando que a barreira foi identificada na Revisão da Literatura), enquanto os números servem para identificar o desafio dentro da lista obtida.

Tabela 1 – Desafios e Barreiras identificados na Revisão da Literatura

<b>Compreensão do Código</b>	
<b>RL01</b>	Explorar a estrutura de um programa, já que o leitor de telas verbaliza os rótulos dos componentes gráficos apresentados por um explorador de pacotes, mas não consegue transmitir o seu significado lógico.
<b>RL02</b>	Distinguir o relacionamento entre classes e subclasses, especialmente quando existem várias subclasses que herdam todas ou algumas características da classe principal.
<b>Depuração de Código</b>	
<b>RL03</b>	Os mecanismos disponíveis para informar o programador sobre erros de sintaxe durante a digitação do código não são acessíveis aos leitores de tela.

*Continua na próxima página...*

- RL04 Recursos visuais que exibem informações sobre como corrigir erros no código não são acessíveis aos leitores de tela.
- RL05 Não há formas rápidas de consultar valores de variáveis e pontos de interrupção.

---

### Navegação no código

---

- RL06 Alguns ícones e outros elementos visuais não possuem rótulos claros o suficiente para transmitir o seu significado.
- RL07 A movimentação pelo código-fonte usando as teclas de seta não é suficiente devido ao seu *layout* e a sua estrutura.
- RL08 Localizar informações específicas em uma base de código utilizando a funcionalidade de pesquisa por meio de palavras-chave ou alguns outros recursos de navegação fornecidos pelos IDEs pode ser demorado e frustrante, pois uma palavra-chave pode aparecer em vários locais na mesma base de código.
- RL09 Retornar rapidamente a uma linha específica ao revisar instruções ou quando é necessário trabalhar com múltiplos arquivos de código-fonte simultaneamente.
- RL10 Compreender o nível do escopo enquanto se navega por estruturas profundamente aninhadas, bem como detectar o início e o fim de um bloco de código.
- RL11 Interagir com as interfaces dos IDEs, fortemente baseadas em estímulos visuais, utilizando estruturas de menus encapsuladas, caixas de diálogo, ícones e outros elementos gráficos.
- RL12 Acessar a numeração das linhas nos editores de código-fonte.
- RL13 Acessar recursos que visivelmente disponíveis na interface gráfica, mas que podem ficar ocultos dentro de vários níveis de hierarquia de navegação e são difíceis de acessar quando se utiliza leitores de tela.
- RL14 Identificar novos recursos adicionados às novas versões dos IDEs, bem como verificar se as funcionalidades já existentes sofreram modificações.
- RL15 Consultar os atalhos para as funcionalidades do IDE.

---

### Edição de Código

---

- RL16 Acessar alguns recursos disponíveis visualmente, como o alinhamento de caracteres especiais ou recuos que marcam o início e o fim do aninhamento.

*Continua na próxima página...*

**RL17** Realizar a leitura de todos os espaços em branco utilizados em linguagens de programação que usam esse tipo de recuo para delimitar blocos de código.

**RL18** Acessar recursos como preenchimento automático (que auxiliam o programador a recordar a nomenclatura de classes, métodos, atributos, dentre outros), e que são exibidos por meio de menu *pop-up* contendo previsões de digitação que podem ser realizadas.

**RL19** Ignorar ou navegar até o final de comentários inseridos no código-fonte.

**RL20** Identificar a ausência, excesso ou posicionamento inadequado de caracteres, operadores e parênteses no código.

### **Skimming de Código**

**RL21** Acessar o recurso de dobra de código, dificultando a identificação de áreas ocultas no código-fonte.

**RL22** Obter uma visão geral da estrutura do código sem precisar ler um documento inteiro.

### **Compreensão da saída**

**RL23** Compreender e verificar se a saída gráfica gerada por um código-fonte atende às especificações desejadas.

### **Sobrecarga Auditiva**

**RL24** Experimentar estresse decorrente de sobrecarga auditiva, ao receber muitas informações transmitidas no canal de áudio em uma interface.

### **Leitura de Código-fonte**

**RL25** Ao analisar um código-fonte, o leitor de tela divide o programa em uma sequência de caracteres, palavras e linhas, lendo cada item de maneira isolada, o que pode dificultar a compreensão para o ouvinte.

É importante destacar que as barreiras e desafios identificados na revisão da literatura não se limitam aos estudantes de cursos de computação. Dado que tais obstáculos afetam tanto estudantes quanto programadores profissionais com deficiência visual, optou-se por conduzir um estudo específico para compreender se havia outras dificuldades enfrentadas pelos estudantes que não tivessem sido identificadas nos estudos publicados. Os resultados detalhados desta etapa da pesquisa são apresentados na próxima seção.

## 4.2 Entrevistas e Questionários

Com o objetivo de aprofundar a compreensão dos desafios enfrentados por estudantes com deficiência visual nas disciplinas de programação de computadores e contribuir para a melhoria da acessibilidade em IDEs, esta etapa da pesquisa se concentrou na exploração desse contexto específico. Para isso, foram conduzidas entrevistas com estudantes e egressos de Cursos Superiores de Computação com deficiência visual, enquanto questionários foram aplicados aos professores que ministraram disciplinas de programação de computadores para esse público.

As questões das entrevistas e do questionário foram elaboradas a partir de uma abordagem dedutiva, em que, segundo Guerin; Jayatilaka; Ranasinghe (2015), assume-se que a fundamentação teórica fornece informações suficientes para gerar o conjunto inicial de questões. Tanto os estudantes quanto os professores foram recrutados a partir de listas de e-mails, indicação de professores, contatos dos pesquisadores e amostragem de bola de neve (*Snowball sampling*<sup>2</sup>).

### 4.2.1 Caracterização do Estudo

Para os professores, a técnica de pesquisa escolhida foi o questionário, contendo perguntas abertas e fechadas. Optou-se por utilizá-lo porque permite a coleta de dados de um número maior de pessoas, inclusive aquelas geograficamente dispersas (Barbosa; Silva, 2010). A participação dos docentes requereria que estivessem atualmente lecionando ou tivessem lecionado disciplinas relacionadas à programação de computadores para estudantes com deficiência visual. Antes de responder às perguntas, os professores precisavam concordar com o Termo de Consentimento Livre e Esclarecido (TCLE), disponibilizado digitalmente. As questões do questionário buscavam identificar o perfil dos docentes, bem como as dificuldades e obstáculos enfrentados pelos estudantes com deficiência visual durante as disciplinas de programação de computadores. O questionário enviado aos docentes, assim como o TCLE, estão disponíveis no Anexo A.

No caso dos estudantes e egressos, optou-se por adotar a técnica de entrevista, uma conversa guiada por um roteiro de perguntas ou tópicos, na qual um entrevistador busca coletar informações de um entrevistado para obter uma visão profunda e abrangente dos tópicos investigados (Barbosa; Silva, 2010). As entrevistas foram conduzidas virtualmente, utilizando a ferramenta *Google Meet*<sup>3</sup> ou por chamada de vídeo via *WhatsApp*<sup>4</sup>, de acordo com a preferência de cada entrevistado. Todas as entrevistas foram gravadas em áudio/vídeo e iniciaram com a leitura do Termo de Consentimento

---

<sup>2</sup>Técnica de amostragem onde o pesquisador pede que os participantes indiquem outras pessoas que possuam as características desejadas para a pesquisa (Gil et al., 2002)

<sup>3</sup><https://meet.google.com/>

<sup>4</sup><https://web.whatsapp.com/>

Livre e Esclarecido (TCLE) e do Termo de Uso de Imagem e Voz (TUIV), disponíveis nos Anexos B e C, respectivamente. Após a leitura, os participantes tiveram a oportunidade de fazer perguntas e fornecer o consentimento verbal para participar da pesquisa. As perguntas da entrevista abrangiam aspectos relacionados à identificação do perfil de cada participante, além de tópicos relacionados à experiência deles nas disciplinas de programação de computadores. O roteiro adotado está disponível no Anexo D.

#### 4.2.2 Análise dos Dados

A análise dos dados foi conduzida empregando uma abordagem qualitativa por meio da técnica de análise de conteúdo, seguindo as cinco fases propostas por Bardin (2015): (1) organização da análise; (2) codificação; (3) categorização; (4) tratamento dos resultados; e, (5) inferência e interpretação dos resultados.

Durante a fase de organização da análise, tornou-se evidente que as informações coletadas apresentavam a possibilidade de serem examinadas a partir de duas perspectivas distintas, quais sejam:

1. **Desafios e barreiras educacionais**, relacionado ao processo de aprendizagem nas disciplinas de programação de computadores; e,
2. **Desafios e barreiras de interação** com IDEs e outras ferramentas utilizadas para programar.

Essas duas abordagens propiciaram uma análise mais abrangente e detalhada, possibilitando uma compreensão mais completa do conjunto de dados. Após essa organização, avançou-se para as etapas de Codificação e Categorização.

A análise dos dados relacionados à perspectiva de **Desafios e Barreiras Educacionais** se baseou nas Categorias e Códigos adotados por Baker; Bennett; Ladner (2019), descritos no Capítulo 2. Entretanto, foi utilizado exclusivamente a Categoria "**Acomodações**", visto que aborda diretamente a maneira como os estudantes com deficiência visual acessam as informações durante as aulas. As Categorias "Tecnologias" e "Impactos sociais" não foram aplicados devido à falta de relevância direta em relação aos resultados específicos que estavam sendo investigados. Os dados foram organizados com base nas três subcategorias de "Acomodações" especificadas pelos autores: (1) Aulas/Palestras (*Lectures*); (2) Materiais (*Materials*); e, Tarefas/Atividades (*Assignments*).

Em relação à perspectiva de **Desafios e Barreiras de Interação**, a análise realizada teve como base as Categorias e Códigos apresentados por Mountapmbeme; Okafor; Ludi (2022). Da mesma forma, apenas a Categoria "**Tarefas e Desafios de Programação**" foi utilizada na análise, por abordar temas diretamente vinculados aos resultados obtidos nesta fase da pesquisa. Esta Categoria engloba os seguintes

subcategorias (Mountapmbeme; Okafor; Ludi, 2022): (1) Compreensão do código; (2) Navegação no código; (3) Depuração de código; (4) Edição de código; e, (5) *Skimming* de código.

Após a definição das Categorias e Códigos a serem adotados, avançou-se para a fase de análise das respostas provenientes das entrevistas e questionários. Entretanto, durante a execução desta etapa, observou-se que alguns apontamentos remetiam a desafios que não haviam sido contemplados nos estudos conduzidos por Baker; Bennett; Ladner (2019) ou por Mountapmbeme; Okafor; Ludi (2022). Tais apontamentos referiam-se a desafios específicos relacionados à compreensão da saída gráfica gerada por um código-fonte. Diante disso, para a análise realizada neste trabalho, foi incorporado o Código "Compreensão da saída" à Categoria "Tarefas e Desafios de Programação", com o intuito de registrar as respostas coletadas nas entrevistas e questionários relacionadas a essa temática específica. A Tabela 2 apresenta o conjunto final de Categorias e Códigos adotados para a análise realizada nesta etapa da pesquisa.

Tabela 2 – Categorias e Códigos adotados na Análise de Conteúdo

<b>Perspectiva</b>	<b>Categoria</b>	<b>Códigos</b>
Desafios e Barreiras Educacionais	Acomodações	Aulas/Palestras Materiais Tarefas/Atividades
Desafios e Barreiras de Interação	Tarefas e Desafios de Programação	Compreensão do código Navegação no código Depuração de código Edição de código <i>Skimming</i> de código Compreensão da Saída

### 4.2.3 Resultados

Essa seção apresenta uma síntese das informações resultantes da Análise de Conteúdo aplicada aos dados obtidos por meio de questionários e entrevistas. A organização segue as Perspectivas, Categorias e Códigos detalhados na Tabela 2.

No total, a pesquisa contou com 18 participantes, sendo 12 docentes de diferentes instituições de ensino brasileiras e 6 estudantes e/ou egressos de Cursos de Computação com deficiência visual. Dentre os estudantes, havia 1 mulher e 5 homens, com idade entre 22 e 51 anos. Um dos participantes possui daltonismo, enquanto os outros 5 são cegos.

#### 4.2.3.1 *Desafios e Barreiras Educacionais*

Quando questionados sobre as dificuldades enfrentadas durante a aprendizagem de programação de computadores, os estudantes destacaram desafios associados à acessibilidade dos recursos e materiais de apoio utilizados nos cursos. Segundo eles, muitos livros didáticos e apostilas não são totalmente acessíveis aos leitores de tela. Isso acaba prejudicando o acompanhamento das aulas e realização das atividades solicitadas pelos professores.

Em se tratando especificamente de programação de computadores, uma das dificuldades apontadas pelos estudantes está relacionada à falta de um padrão estabelecido para leitores de tela verbalizarem o código-fonte. Isso pode resultar na omissão ou transmissão inadequada de informações importantes. Além disso, a leitura do código-fonte realizada pelos leitores de tela nem sempre é compreensível, pois não é feita a interpretação de caracteres não alfanuméricos e não são identificadas as partes relevantes do código-fonte que devem ser lidas ou ignoradas, prejudicando a comunicação efetiva dessas informações.

Alguns estudantes acreditam que os professores podem não estar plenamente cientes de suas necessidades e preferências. A falta de compreensão sobre os obstáculos enfrentados por esses estudantes pode resultar na escolha inadequada de estratégias e ferramentas de ensino. Conseqüentemente, pode impedir a criação de um ambiente de aprendizagem propício (Rajaselvi et al., 2021), além de contribuir para a evasão dos programas de Computação e impossibilitar a produção de soluções para tornar o cenário do ensino mais acessível e inclusivo (Alves et al., 2022).

A Tabela 3 apresenta uma síntese dos resultados obtidos nesta etapa da Análise de Conteúdo, organizados a partir dos Códigos pré-estabelecidos na Tabela 2. Os dados obtidos são apresentados conforme a padronização já adotada neste Capítulo, identificando cada desafio com um código formado por letras e números. As letras indicam a perspectiva associada (DE, indicando que a barreira está relacionada a Desafios e Barreiras Educacionais), enquanto os números servem para identificar o desafio dentro da lista obtida.

Tabela 3 – Desafios e Barreiras Educacionais - Categoria: Acomodações

---

**Código: Aulas/Palestras**

---

**DE01** Ao utilizarem videoaulas ou ferramentas de compartilhamento de tela, quem está falando não verbaliza todas as informações apresentadas visualmente de maneira completa e adequada. O estudante também encontra dificuldades porque o leitor de telas não consegue verbalizar o conteúdo visual exibido, visto que as informações transmitidas como uma imagem.

*Continua na próxima página...*

**DE02** Em algumas ocasiões, os professores falam muito rápido, dificultando o acompanhamento das atividades ou a compreensão das explicações fornecidas.

---

**Código: Materiais**

---

**DE03** Há poucos livros, apostilas e outros materiais disponíveis no formato de audiolivros. A vantagem desses recursos em relação à leitura de materiais digitais realizada pelos leitores de tela é que a narrativa do texto é gravada e disponibilizada em um formato que facilita a compreensão do conteúdo pelo ouvinte.

**DE04** No intuito de fornecer uma alternativa acessível aos materiais impressos, por vezes os professores optam por escaneá-los. Contudo, essa prática é muitas vezes realizada de forma inadequada, resultando na disponibilização dos recursos em formato de imagem, o que os torna inacessíveis aos leitores de tela.

**DE05** Os Ambientes Virtuais de Aprendizagem (AVAs)<sup>5</sup> ou as plataformas específicas para compartilhamento de arquivos e outros recursos educacionais utilizadas pelas instituições de ensino podem dispor de acessibilidade inadequada ou insuficiente, dificultando o acesso aos materiais e atividades disponibilizados pelo professor;

**DE06** Em algumas ocasiões, a voz do leitor de tela pode se tornar cansativa, sendo necessário alterá-la.

---

**Código: Tarefas/Atividades**

---

**DE07** Por não encontrarem uma alternativa acessível para avaliar os estudantes com deficiência visual, muitas vezes as avaliações realizadas pelos professores são conduzidas de forma oral ou em formatos alternativos, utilizando instrumentos diferentes daqueles adotados para o restante da turma.

**DE08** As avaliações também podem ocorrer em horários e locais diferentes dos demais estudantes, além de necessitar do auxílio de um monitor para garantir a acessibilidade.

**DE09** Estudantes com deficiência visual podem precisar utilizar ferramentas diferentes dos demais colegas para realização das atividades, pois os recursos utilizados pelo restante da turma não proporcionam a acessibilidade necessária.

**DE10** Estudantes com deficiência visual podem necessitar de enunciados das tarefas e atividades com um nível maior de detalhamento para facilitar a compreensão do que está sendo solicitado.

*Continua na próxima página...*

---

<sup>5</sup>Ferramentas digitais interdisciplinares usadas para fornecer recursos pedagógicos (Beda et al., 2022)

**DE11** Os estudantes não se sentem confiantes para executar tarefas que exigem a construção de interfaces gráficas e muitas vezes precisam solicitar ajuda do professor ou dos colegas para verificar se a saída gerada atende às especificações desejadas.

---

Os resultados obtidos ressaltam a importância de conhecer as necessidades e preferências dos estudantes com deficiência visual para aprimorar a escolha das estratégias, recursos e ferramentas adotados pelos professores no ensino de programação de computadores. Além disso, os professores de programação, professores de sala de recursos, educadores especiais e outros profissionais precisam estar familiarizados com as tecnologias existentes e as possibilidades de configuração dos recursos de Tecnologia Assistiva que podem auxiliar na interação dos estudantes. Torna-se essencial, portanto, promover a conscientização entre os educadores e fornecer o treinamento adequado para criar um ambiente de aprendizagem mais inclusivo.

#### *4.2.3.2 Desafios e Barreiras de Interação*

Os participantes expressaram diversas observações, tanto nos questionários (docentes) quanto nas entrevistas (estudantes e egressos), indicando os desafios enfrentados ao utilizar os IDEs nas disciplinas de programação de computadores.

A análise das respostas confirma que os IDEs não oferecem completamente os mecanismos de acessibilidade necessários para interação utilizando teclado e leitor de tela. Além disso, ocorre também o desconhecimento por parte de estudantes e professores em relação aos recursos de acessibilidade oferecidos, tanto do ponto de vista da Tecnologia Assistiva quanto do IDE.

Frequentemente, as barreiras enfrentadas pelos estudantes com deficiência visual resultam na necessidade de um tempo maior em comparação aos colegas para realizar as atividades. Isso ocorre porque os estudantes podem depender do auxílio do professor, dos colegas ou de outras pessoas para encontrar uma ferramenta adequada para realizar as atividades ou para aprender a utilizar os recursos necessários para programar. Essas barreiras impactam negativamente no comprometimento e dedicação dos estudantes, pois podem deixá-los frustrados e demandar mais esforço cognitivo.

A Tabela 4 apresenta uma síntese dos dados identificados neste estudo. Nesta Tabela, novamente os desafios são identificados por um código formado por letras e números. As letras indicam a perspectiva associada (DI, indicando que a barreira está relacionada a Desafios e Barreiras de Interação), enquanto os números servem para identificar o desafio dentro da lista obtida.

Tabela 4 – Desafios e Barreiras de Interação - Categoria: Tarefas e Desafios de Programação

<b>Compreensão do Código</b>	
-	Não foram feitos apontamentos relativos à Compreensão do Código.
<b>Navegação no código</b>	
<b>DI01</b>	As interfaces dos IDEs são complexas e constituídas de muitos elementos visuais, o que limita a acessibilidade para leitores de tela, dificultando ou até mesmo impedindo o acesso a alguns recursos.
<b>DI02</b>	É difícil lidar com múltiplos arquivos de código-fonte simultaneamente.
<b>DI03</b>	Para pessoas com daltonismo, pode ser difícil distinguir a imagem do ícone da imagem do ponteiro do mouse (quando o mouse está posicionado sobre o ícone).
<b>DI04</b>	A maioria dos IDEs permite configurar o tamanho da fonte utilizada para a escrita do código-fonte, mas não permite configurar o tamanho dos outros elementos da interface gráfica. Para isso, é necessário utilizar recursos disponibilizados pelo sistema operacional, como configuração de exibição ou lupa.
<b>DI05</b>	Não existe uma forma de consultar os atalhos associados às funcionalidades do ambiente na própria ferramenta.
<b>DI06</b>	Pode ocorrer de tanto a IDE quanto o leitor de telas utilizam um mesmo atalho para acessar funcionalidades diferentes, causando sobreposição.
<b>DI07</b>	Há casos em que diferentes IDEs utilizam atalhos diferentes para acessar a mesma funcionalidade.
<b>DI08</b>	É necessário memorizar os diferentes atalhos associados a cada funcionalidade do IDE.
<b>Depuração de Código</b>	
<b>DI09</b>	Não existe um mecanismo acessível para informar a ocorrência de erros de sintaxe, realizada visualmente através da utilização de cores.
<b>DI10</b>	É necessário percorrer todo o código para encontrar um erro e, então, corrigi-lo.
<b>DI11</b>	Os estudantes desconhecem os recursos de depuração disponibilizados pelo IDE.
<b>Edição de Código</b>	
<b>DI12</b>	Não existe um mecanismo acessível para identificar indentação de código e, conseqüentemente, reconhecer facilmente o nível de escopo e detectar o início e o fim de um bloco de código.
<b>DI13</b>	O recurso autocompletar não é acessível aos leitores de tela, implicando na memorização da nomenclatura das classes, métodos e atributos, ou outras informações de um arquivo de código-fonte.

*Continua na próxima página...*

**DI14** As palavras-reservadas da maioria das linguagens de programação são escritas em língua inglesa e, normalmente, são pronunciadas de maneira incorreta pelos leitores de tela, que estão configurados para pronunciar o conteúdo na língua da interface em uso (geralmente em língua portuguesa), causando dificuldades de entendimento.

---

### **Skimming de Código**

---

- Não foram feitos apontamentos relativos à *Skimming* de Código.

---

### **Compreensão da saída**

---

**DI15** É difícil identificar os elementos que compõem a saída gráfica do sistema.

---

É importante destacar que nenhum dos participantes (estudantes, egressos ou professores) fez qualquer comentário que mencionasse desafios associados à "Compreensão de Código" e "*Skimming* de Código", o que pode indicar que nem estudantes nem docentes lembraram ou perceberam esses desafios, ou talvez, não os consideraram significativos o suficiente para mencioná-los. Isso pode sugerir que os participantes não tenham consciência desses tipos específicos de barreiras ou que tenham priorizado outros obstáculos, identificando-os como mais relevantes sobre sua experiência com IDEs e programação de computadores.

Além disso, nenhum dos docentes que respondeu ao questionário identificou quaisquer dificuldades encontradas pelos estudantes em relação à "Depuração de Código". Apenas estudantes apontaram desafios e barreiras relacionadas a esse aspecto específico. Essa observação sugere que os professores podem não ter percebido, durante as aulas, as dificuldades enfrentadas pelos estudantes nessa área, reforçando a necessidade de uma formação adequada para os docentes que lecionam para estudantes com deficiência visual.

## **4.3 Considerações Parciais**

Assegurar a acessibilidade dos materiais e atividades ao longo da formação profissional, assim como dos recursos e tecnologias essenciais para realizar tarefas de programação é essencial para a preparação adequada dos estudantes com deficiência visual. O treinamento adequado e a familiarização adquirida durante esses cursos podem exercer influência significativa no desenvolvimento de suas habilidades, na participação plena em atividades acadêmicas e profissionais e, por conseguinte, no seu sucesso futuro.

Neste estágio da pesquisa, buscou-se compreender esses desafios de aprendizagem de programação e de interação com IDEs enfrentados pelos estudantes com deficiência visual, reconhecendo a importância de desenvolver recursos que facilitem

um maior acesso e compreensão dessas ferramentas e promover uma experiência mais inclusiva e produtiva para esses estudantes.

Os resultados apresentados neste Capítulo resultaram em três publicações, que contém mais detalhes a respeito dos procedimentos adotados e dos resultados obtidos neste Estágio da pesquisa:

1. ZEN, Eliana; DA COSTA, Vinícius Kruger; TAVARES, Tatiana Aires. Experiências Educacionais em Disciplinas de Programação de Computadores: uma Análise Qualitativa na Perspectiva dos Estudantes com Deficiência Visual. In: Anais do XXXIV Simpósio Brasileiro de Informática na Educação. SBC, 2023. p. 960-971. <https://sol.sbc.org.br/index.php/sbie/article/view/26726>.
2. ZEN, Eliana; DA COSTA, Vinicius Kruger; TAVARES, Tatiana Aires. *Understanding the Accessibility Barriers Faced by Learners with Visual Impairments in Computer Programming*. In: *Proceedings of the XXII Brazilian Symposium on Human Factors in Computing Systems*. 2023. p. 1-11. <https://dl.acm.org/doi/10.1145/3638067.3638076>.
3. ZEN, Eliana; TAVARES, Tatiana Aires; DA COSTA, Vinícius Kruger. Desafios e Percepções sobre Acessibilidade em Ambientes de Desenvolvimento Integrado. *Revista Novas Tecnologias na Educação*, v. 21, n. 2, p. 244-253, 2023. <https://seer.ufrgs.br/renote/article/view/137744>.

## **5 ESPECIFICAÇÃO DAS DIRETRIZES**

Este Capítulo detalha o processo de elaboração a versão preliminar das Diretrizes de Acessibilidade em IDEs para Estudantes com Deficiência Visual. Tais diretrizes são resultantes da convergência dos resultados obtidos a partir de revisão da literatura e estudo qualitativo realizado com professores, estudantes e egressos de cursos de Computação, apresentados no Capítulo 4.

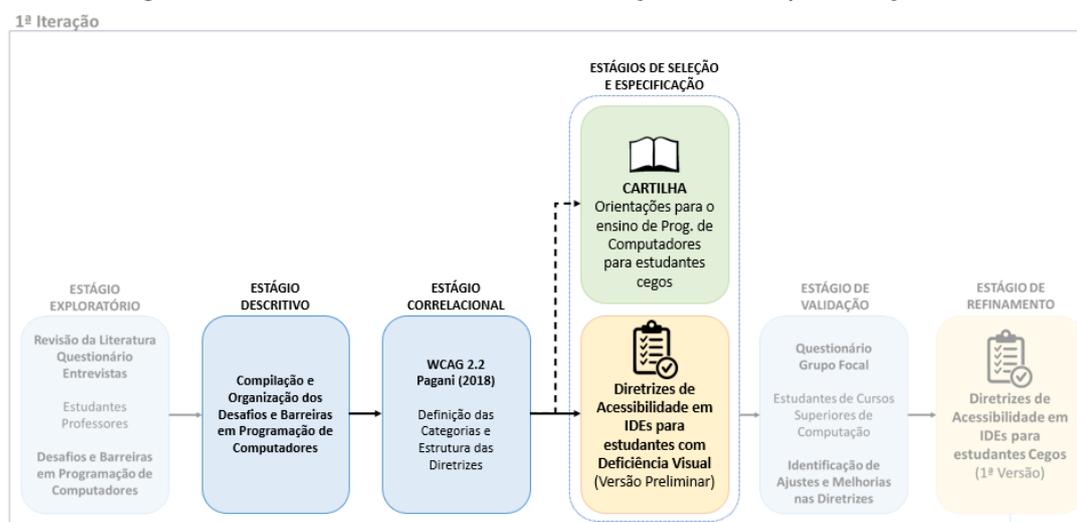
Além disso, é relevante ressaltar que, devido à abordagem de características distintas, os dados coletados no Estágio Exploratório também proporcionaram informações sobre os desafios e as barreiras educacionais enfrentadas no aprendizado de programação de computadores. Essas informações contribuíram para a elaboração de uma Cartilha de orientações voltada para o ensino de programação de computadores a estudantes cegos. A concepção dessa Cartilha surgiu da constatação de que os professores muitas vezes não têm um entendimento claro e aprofundado das necessidades e preferências desses estudantes.

O processo de elaboração das Diretrizes de Acessibilidade em IDEs para Estudantes com Deficiência Visual descrito neste Capítulo corresponde aos Estágios Descritivo, Correlacional, de Seleção e de Especificação da metodologia de Quiñones; Rusu; Rusu (2018), conforme ilustrado na Figura 5.

### **5.1 Versão Preliminar das Diretrizes**

O processo de elaboração das diretrizes teve início com a compilação dos resultados obtidos no Estágio Exploratório (Capítulo 4), derivados de revisão da literatura e estudo qualitativo envolvendo estudantes e professores. No entanto, uma vez que os dados obtidos no estudo qualitativo foram classificados em duas perspectivas de análise distintas (Desafios e Barreiras Educacionais e Desafios e Barreiras de Interação), para a elaboração da versão preliminar do conjunto de Diretrizes de Acessibilidade em IDEs para Estudantes com Deficiência Visual, apenas os resultados relacionados à perspectiva de Desafios e Barreiras de Interação foram considerados.

Figura 5 – Estágios Descritivo, Correlacional, de Seleção e de Especificação



### 5.1.1 Categorização das Diretrizes

Quiñones; Rusu; Rusu (2018) recomendam que as Diretrizes sejam organizadas em categorias para reduzir a complexidade das informações coletadas. Cada categoria pode agrupar recomendações que avaliam determinados aspectos específicos. Neste sentido, para o propósito deste estudo, foram definidas oito categorias, seguindo a categorização adotada na análise dos dados obtidos no estudo qualitativo descrito no Capítulo 4, a saber:

- Categoria 1. Compreensão do código;
- Categoria 2. Depuração de código;
- Categoria 3. Navegação no código;
- Categoria 4. Edição de código;
- Categoria 5. *Skimming* de código;
- Categoria 6. Compreensão da Saída;
- Categoria 7. Sobrecarga auditiva; e,
- Categoria 8. Leitura do código-fonte;

As categorias estabelecidas agrupam um conjunto de barreiras ou limitações encontradas por estudantes com deficiência visual ao interagir com IDEs. Cada categoria abrange uma ou mais diretrizes que contém recomendações para orientar o desenvolvimento de IDEs acessíveis para pessoas com deficiência visual.

A Tabela 5 apresenta uma síntese das barreiras e desafios de interação identificados, organizados com base no conjunto de categorias definidas, juntamente com

o estudo correspondente que possibilitou sua identificação: (DI) Desafios e Barreiras de Interação; e, (RL) Revisão da Literatura. Cada desafio recebeu um novo código, composto por letras e números, criado para permitir que pudessem ser referenciados posteriormente nesta Tese.

Tabela 5 – Dificuldades de Interação com IDEs por Pessoas com Deficiência Visual

<b>Compreensão do Código</b>		DI	RL
DES01	Explorar a estrutura de um programa, transmitindo o significado lógico dos itens listados no explorador de pacotes.		RL01
DES02	Distinguir o relacionamento entre classes e subclasses em uma base de código.		RL02
<b>Depuração de código</b>		DI	RL
DES03	Identificar erros de sintaxe durante a digitação.	DI09	RL03
DES04	Encontrar erros no código.	DI10	
DES05	Corrigir erros no código.	DI10	RL04
DES06	Obter informações sobre variáveis e pontos de interrupção.		RL05
<b>Navegação no código</b>		DI	RL
DES07	Interagir com as interfaces gráficas dos IDEs utilizando leitores de tela e acessar recursos ficam ocultos em diferentes níveis de hierarquia de navegação.	DI01	RL11 RL13
DES08	Manusear múltiplos arquivos de código-fonte.	DI02	
DES09	Permitir configurar a exibição dos elementos da interface gráfica para atender às necessidades de pessoas com daltonismo.	DI03	
DES10	Possibilitar a configuração da ampliação dos elementos na interface gráfica.	DI04	
DES11	Compreender o significado dos rótulos de ícones e outros elementos visuais.		RL06
DES12	Movimentar o cursor do teclado pelo código-fonte utilizando as teclas de seta.		RL07
DES13	Localizar informações específicas em uma base de código.		RL08
DES14	Mover o cursor do teclado para uma posição de foco anterior ou linha específica no código.		RL09
DES15	Compreender o nível de escopo de uma linha de código.		RL10
DES16	Consultar o número das linhas no editor de código-fonte.		RL12
DES17	Reconhecer alterações em novas versões dos IDEs.		RL14
DES18	Consultar atalhos para as funcionalidades do IDE.	DI05	RL15
DES19	Utilização de atalhos idênticos para funções distintas.	DI06	
DES20	Utilização de atalhos distintos para funções idênticas.	DI07	

*Continua na próxima página...*

<b>DES21</b>	Lembrar os atalhos para as funcionalidades.	<b>DI08</b>	
<b>Edição de código</b>		DI	RL
<b>DES22</b>	Identificar níveis de indentação de uma linha de código.	<b>DI12</b>	<b>RL16</b>
<b>DES23</b>	Identificar o início e o fim de um bloco de código.		<b>RL16</b>
<b>DES24</b>	Utilizar o recurso autocompletar.	<b>DI13</b>	<b>RL18</b>
<b>DES25</b>	Utilizar recuos de espaço em branco para delimitar blocos de código.		<b>RL17</b>
<b>DES26</b>	Ignorar ou navegar até final de comentários inseridos no código-fonte.		<b>RL19</b>
<b>DES27</b>	Identificar a ausência, excesso ou posicionamento inadequado de caracteres não-alfanuméricos no código.		<b>RL20</b>
<b>Skimming de código</b>		DI	RL
<b>DES28</b>	Reconhecer e utilizar o recurso de dobra de código.		<b>RL22</b>
<b>DES29</b>	Obter uma visão geral do código-fonte.		<b>RL22</b>
<b>Compreensão da Saída</b>		DI	RL
<b>DES30</b>	Verificar a saída gráfica gerada por um código-fonte.	<b>DI15</b>	<b>RL23</b>
<b>Sobrecarga Auditiva</b>		DI	RL
<b>DES31</b>	Experimentar estresse ao receber muitas informações transmitidas pelo canal de áudio.		<b>RL24</b>
<b>Leitura de Código-Fonte</b>		DI	RL
<b>DES32</b>	Reconhecer a pronúncia das palavras-reservadas das linguagens de programação.	<b>DI14</b>	
<b>DES33</b>	Ler o código-fonte de maneira que faça sentido para o ouvinte		<b>RL25</b>

Após a categorização dos desafios e barreiras de interação, deu-se início à definição da estrutura da versão preliminar das Diretrizes de Acessibilidade em IDEs para estudantes com deficiência visual.

### 5.1.2 Estrutura das Recomendações Preliminares

As Diretrizes de Acessibilidade em IDEs para estudantes com deficiência visual foram estruturadas tendo como base a organização adotada pelo *Web Content Accessibility Guidelines 2.2*, (WCAG 2.2, 2023), e por Britto; Pizzolato (2018), seguindo o seguinte formato:

- **Descrição da Diretriz:** apresenta um resumo da recomendação;
- **Crterios de sucesso:** declarações testáveis que visam determinar objetivamente se o conteúdo satisfaz a diretriz;

- **Por que fazer?** Descreve a importância da orientação para reduzir barreiras de interação para pessoa com deficiência visual e quais as dificuldades de interação que ela pode auxiliar a mitigar;
- **Como fazer?** Contém sugestões de implementação da recomendação, elaboradas a partir das soluções já implementadas extraídas dos trabalhos obtidos na literatura; e,
- **Fonte:** referências bibliográficas utilizadas como base para a elaboração das diretrizes e para sugerir como implementá-las.

Após definir a estrutura que cada recomendação adotaria, passou-se para a definição do conjunto de diretrizes e do conteúdo que elas abrangeriam, baseado nas categorias e na estrutura estabelecidas.

### 5.1.3 Recomendações Preliminares

A fim de estabelecer o conjunto preliminar de Diretrizes de Acessibilidade em IDEs para Estudantes com Deficiência Visual, cada desafio apresentado na Tabela 5 foi analisado. O objetivo foi identificar quais desafios compartilhavam características semelhantes, permitindo que fossem agrupados de forma coerente para formar o conjunto de diretrizes.

No total, a versão preliminar contou com 21 diretrizes contendo recomendações que podem ser aplicadas durante o processo de concepção, desenvolvimento e avaliação dessas aplicações. Os resultados desta fase da análise estão detalhados na Tabela 6, que apresenta as categorias e suas respectivas diretrizes, juntamente com o agrupamento dos desafios utilizados como base para sua formação.

Tabela 6 – Categorias e Diretrizes Preliminares

<b>Categoria: Compreensão do código</b>	
Diretriz 1.1 Semântica	DES01 DES02
<b>Categoria: Depuração de código</b>	
Diretriz 2.1 Informações sobre erros	DES03 DES04 DES05 DES27
Diretriz 2.2 Variáveis e constantes	DES06
Diretriz 2.3 Marcadores e Pontos de interrupção	DES06
<b>Categoria: Navegação no código</b>	
Diretriz 3.1 Rótulos	DES11
Diretriz 3.2 Ampliação e Contraste	DES09 DES10
Diretriz 3.3 Movimentação pelo código	DES08 DES12 DES13 DES14
Diretriz 3.4 Contexto e nível de escopo	DES15 DES22 DES23
Diretriz 3.5 Estratégias de navegação	DES07

*Continua na próxima página...*

Diretriz 3.6 Numeração de linhas	DES16
Diretriz 3.7 Atalhos	DES18 DES19 DES20 DES21
Diretriz 3.8 Ajuda e Documentação	DES17
<b>Categoria: Edição de código</b>	
Diretriz 4.1 Espaçamento	DES25
Diretriz 4.2 Autocompletar	DES24
Diretriz 4.3 Comentários	DES26
<b>Categoria: <i>Skimming</i> de código</b>	
Diretriz 5.1 Dobra de Código	DES28
Diretriz 5.2 Visão geral do código	DES29
<b>Categoria: Compreensão da Saída</b>	
Diretriz 6.1 Validação da interface gráfica	DES30
<b>Categoria: Sobrecarga auditiva</b>	
Diretriz 7.1 Alertas sonoros	DES31
<b>Categoria: Leitura de código-fonte</b>	
Diretriz 8.1 Idioma	DES32
Diretriz 8.2 Leitura contextual	DES33

É possível observar que algumas diretrizes englobam a união de mais de um desafio. Por exemplo, DES18, DES19, DES20 E DES21 abordam barreiras relacionadas à utilização de atalhos e, portanto, foram agrupados em uma única diretriz que trata desse tipo de obstáculo. Houve, ainda, casos em que um único desafio serviu de base para a formulação de mais de uma diretriz, como o desafio DES06, desmembrado para formar duas novas diretrizes, visto que abordava aspectos diferentes de programação: Diretriz 2.2 Variáveis e constantes e Diretriz 2.3 Marcadores e Pontos de interrupção.

Por fim, um dos desafios precisou ser associado a uma diretriz classificada em uma categoria diferente da atribuída previamente: DES27. Durante a elaboração das diretrizes, constatou-se que a barreira "Identificar a ausência, excesso ou posicionamento inadequado de caracteres não-alfanuméricos no código" estava diretamente relacionada à Diretriz 2.1 (Informações sobre Erros), pois abordava obstáculos relativos à identificação de problemas de sintaxe no código.

Estabelecido o conjunto preliminar de Diretrizes, avançou-se para a etapa de redação do conteúdo de cada uma delas. O texto completo das diretrizes preliminares pode ser acessado pelo *link*: <https://elianazen.github.io/DiretrizesAcessibilidadeIDE/recomendacoesPreliminares.html>.

## 5.2 Cartilha

Durante o Estágio Exploratório, foram identificados desafios e barreiras educacionais diretamente ligados aos obstáculos enfrentados pelos estudantes com deficiência visual durante o aprendizado de programação de computadores, categorizados sob a perspectiva Desafios e Barreiras Educacionais (Subseção 4.2.3.1). Observou-se que a maioria desses obstáculos estava relacionada à falta de compreensão por parte dos professores sobre as necessidades e preferências desses alunos. Essa falta de clareza pode influenciar na escolha inadequada dos métodos, recursos e tecnologias utilizados para o ensino dessas disciplinas.

Diante deste contexto, decidiu-se por elaborar uma Cartilha de orientações para o ensino de Programação de Computadores voltada professores que lecionam para estudantes cegos. Essa decisão foi tomada considerando que esses alunos geralmente dependem principalmente de leitores de tela como um recurso de Tecnologia Assistiva para acessar e interagir com computadores. Nesta pesquisa, os leitores de tela foram examinados em detalhes, o que permitiu identificar as principais limitações enfrentadas por esses estudantes durante as aulas, assim como algumas orientações específicas para atender às necessidades dos estudantes cegos no contexto do ensino de programação de computadores.

Nesta Cartilha são apresentadas informações sobre os principais desafios enfrentados por esses estudantes e fornecidas orientações para auxiliar na elaboração e na seleção de materiais, tarefas e estratégias de avaliação, além de uma lista de ferramentas que podem aprimorar a acessibilidade e auxiliar esses estudantes durante o processo de aprendizado. A ideia é disponibilizá-la sob a licença "*Creative Commons* Atribuição – Uso não comercial – Sem derivações 4.0 Internacional" (CC BY-NC 4.0 DEED). Sob este tipo de licença, qualquer pessoa que tenha acesso ao seu conteúdo pode copiá-lo e compartilhá-lo, desde que atribua crédito aos autores, não realize alterações e não utilize a obra para fins comerciais.

A cartilha foi submetida a uma avaliação por meio de um grupo focal composto por profissionais do Instituto Federal Farroupilha, Campus São Vicente do Sul (IFFar-SVS), incluindo duas pedagogas, duas professoras de Educação Especial e três professores de Programação de Computadores (Figura 6). Esses profissionais foram selecionados para analisar o material e coletar *feedbacks* a partir de diferentes perspectivas. Enquanto pedagogas e professoras de Educação Especial auxiliam os estudantes com necessidades especiais e orientam os professores sobre as melhores práticas para atendê-los, os professores de programação de computadores têm um papel central no ensino e na formação técnica desses estudantes.

A escolha da amostra se deu por conveniência, que ocorre quando os elementos são selecionados por estarem mais disponíveis para participar do estudo (Gil et al.,

2002). Esses profissionais foram convidados porque eram colegas de trabalho da pesquisadora que já haviam auxiliado estudantes com deficiência visual anteriormente. O grupo focal ocorreu no mês de dezembro de 2023 e seguiu o roteiro disponível no Anexo F.

Figura 6 – Grupo Focal com Profissionais do IFFar-SVS



Os participantes receberam a Cartilha com antecedência e, posteriormente, o grupo se reuniu com a pesquisadora para discutir sobre a estrutura, clareza, abrangência e precisão das informações nela contidas, avaliando também sua utilidade no contexto educacional. Todos os participantes concordaram com o Termo de Consentimento Livre e Esclarecido e com o Termo para Uso da Imagem e/ou Som da Voz.

Em geral, o grupo avaliou a Cartilha de maneira positiva, enfatizando sua utilidade e importância para o contexto educacional. Segundo os participantes, a Cartilha oferece uma visão inicial das necessidades dos estudantes cegos e sugere soluções viáveis, fornecendo orientações para que os professores possam se preparar adequadamente para receber os alunos.

Os professores de programação destacaram uma das orientações presentes na cartilha, que aborda a necessidade de dialogar com os alunos cegos antes de iniciar as aulas e durante todo o processo educativo. De acordo com eles, avaliar o domínio do braile, as habilidades de computação e a familiaridade com o teclado e leitor de telas é essencial para que o professor possa tomar decisões mais apropriadas em relação às estratégias de ensino e às tecnologias a serem adotadas.

Uma das pedagogas afirmou que a cartilha auxiliaria na divulgação das informações a respeito das necessidades de estudantes cegos, tanto para os profissionais envolvidos na sua formação educacional quanto para os próprios estudantes.

Algumas melhorias pontuais para aprimoramento do conteúdo foram elencadas. Por exemplo, uma das professoras de Educação Especial ressaltou a importância de disponibilizá-la em um formato acessível para todas as pessoas, independente de terem deficiência ou não. Ela ressaltava também, que para outros profissionais que não conhecem programação de computadores e utilizarão a cartilha, seria interessante ter um glossário de termos técnicos.

Os professores de programação destacaram as dificuldades encontradas ao ensinar disciplinas que normalmente utilizam componentes visuais para explicar o conteúdo, como matrizes, vetores ou estruturas de dados. Isso também se aplica a disciplinas que envolvem a construção de elementos gráficos. Segundo eles, seria interessante complementar a cartilha com uma seção que apresentasse informações sobre "casos de sucesso", ou seja, estratégias e metodologias já adotadas por outros professores e que obtiveram resultados positivos.

No geral, todos os participantes destacaram que a cartilha proporcionará informações que ajudarão a aprimorar o entendimento dos professores sobre as necessidades dos alunos cegos, fornecendo orientações sobre a seleção de materiais e estratégias, bem como sobre como conduzir o processo de ensino para auxiliar os estudantes no aprendizado de programação de computadores. Eles também sugeriram que a incorporação das sugestões tornaria o material ainda mais útil e interessante.

A Cartilha produzida pode ser acessada em [https://elianazen.github.io/DiretrizesAcessibilidadeIDE/Cartilha\\_Final.pdf](https://elianazen.github.io/DiretrizesAcessibilidadeIDE/Cartilha_Final.pdf) OU <https://elianazen.github.io/DiretrizesAcessibilidadeIDE/cartilhafinal/cartilhaFinal.html>. É importante ressaltar que, além da incorporação das melhorias citadas, o documento ainda precisa passar pelo processo de diagramação e ilustração antes de sua publicação oficial. As imagens inseridas até o momento são todas protegidas pela licença *Creative Commons*, mas possuem caráter meramente ilustrativo. A versão final da cartilha demanda uma análise mais detalhada em relação à sua apresentação, requerendo a colaboração de outros profissionais especializados nesta área.

### 5.3 Considerações Parciais

A versão preliminar das diretrizes elaborada nesta etapa da pesquisa foi submetida à validação por estudantes de cursos superiores da área de Computação. Os detalhes relativos a esta etapa da pesquisa, assim como os resultados obtidos, são apresentados no Capítulo 6.

É importante ressaltar que a metodologia proposta por Quiñones; Rusu; Rusu (2018) recomenda manter o número de diretrizes relativamente pequeno, idealmente entre 10 e 16. Segundo os autores, essa sugestão decorre da dificuldade prática de aplicar uma grande quantidade de heurísticas. Entretanto, neste trabalho, foram elaboradas 21 diretrizes. Os pesquisadores ponderaram que a complexidade e as particularidades envolvidas na programação de computadores e no uso de IDEs demandavam um número maior de diretrizes para serem adequadamente abordadas e organizadas.

Em se tratando da cartilha produzida, esta foi avaliada por um grupo de profissionais que atua diretamente no auxílio e ensino de estudantes cegos, incluindo pe-

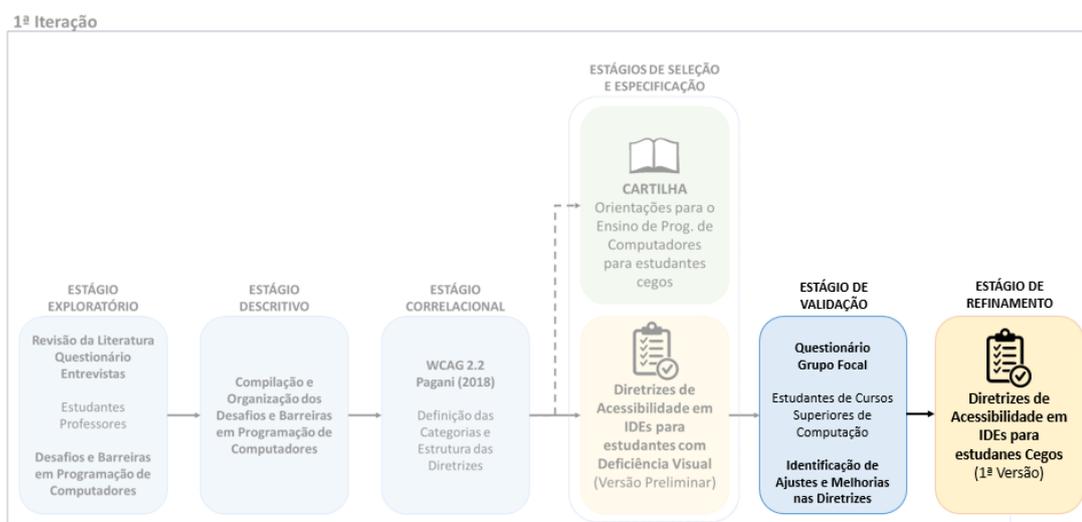
dagogos e professores de educação especial e programação de computadores. É importante ressaltar que, devido ao prazo estabelecido para a conclusão desta tese, as sugestões de melhorias e ajustes na cartilha ainda não puderam ser incorporadas ao documento. a prioridade foi dada ao tema central da Tese de Doutorado, que é a elaboração de Diretrizes de Acessibilidade em IDEs para Estudantes Cegos. No entanto, há a intenção de adaptar o material para incorporar essas mudanças.

## 6 VALIDAÇÃO E REFINAMENTO DAS DIRETRIZES (1ª ITERAÇÃO)

Diretrizes podem apresentar contradições e incoerências, sendo necessário verificar sua solidez e confiabilidade (Nicolle; Abascal, 2001). Portanto, o Estágio de Validação apresenta-se como uma fase crítica no processo de elaboração de diretrizes, e três tipos de avaliação os mais recomendados para esta etapa (Quiñones; Rusu; Rusu, 2018): (1) Avaliação heurística; (2) Avaliação por especialistas; e, (3) Teste do usuário.

Neste contexto, este capítulo pretende descrever o protocolo adotado no primeiro ciclo de execução dos Estágios de Validação e Refinamento (Figura 7) do conjunto preliminar de Diretrizes elaboradas, bem como apresentar os resultados obtidos nessas etapas.

Figura 7 – Estágios de Validação e Refinamento (1ª iteração)



É importante salientar que esta etapa da pesquisa foi aprovada pelo Parecer Consubstanciado n.º 68361023.1.0000.5574 emitido pelo Comitê de Ética em Pesquisa com Seres Humanos no Instituto Federal Farroupilha (IFFar), disponível no Apêndice B.

## 6.1 Caracterização do Estudo

Nesta fase da pesquisa, buscou-se validar a versão preliminar das Diretrizes propostas na Tese de Doutorado utilizando o método de avaliação por especialistas, que contou com a participação de estudantes de cursos superiores de Computação.

Para etapa da avaliação, foram convidados a participar os estudantes com deficiência visual que já haviam se envolvido nas etapas anteriores desta pesquisa, descritas no Capítulo 4. Três estudantes aceitaram o convite e receberam as informações necessárias para realizar a avaliação. Contudo, apenas um deles a concluiu.

A pesquisa também contou com a participação de estudantes matriculados no 4º e 6º semestre do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal Farroupilha Campus São Vicente do Sul (RS). A escolha por esse grupo de participantes ocorreu por conveniência (Gil et al., 2002), visto que a pesquisadora já havia ministrado a disciplina de Interação Humano-Computador para os estudantes do 6º semestre e estava finalizando a mesma disciplina com os estudantes do 4º semestre. Portanto, esses participantes já possuíam conhecimento dos principais conceitos e métodos de avaliação relacionados à Interação Humano-Computador e à acessibilidade de interfaces. Além disso, todos os estudantes já dominavam diversos conceitos e técnicas de programação e sabiam utilizar diferentes IDEs. Ademais, esses estudantes recebem formação que os capacita a contribuir no desenvolvimento dessas ferramentas no futuro.

As técnicas selecionadas para a coleta de dados foram: (1) questionário, que permite adquirir informações de maneira padronizada e ampla, permitindo a obtenção de dados sobre as percepções e experiências dos participantes (Gil et al., 2002); e, (2) grupo focal, uma vez que viabiliza a obtenção de diversas perspectivas e aprofunda a compreensão sobre um tema específico através de discussões em grupo mediadas por um moderador (Kind, 2004).

No total, 20 estudantes participaram da avaliação, sendo 19 do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas e 1 do Curso de Licenciatura em Computação. Este último já havia participado do estudo qualitativo discutido no Capítulo 4.

## 6.2 1ª Etapa de Validação: Questionário

Dentre os 20 estudantes que aceitaram o convite para participar desta etapa da pesquisa e responderam ao questionário, apenas o estudante da Licenciatura em Computação afirmou ter cursado menos de 50% das disciplinas do curso até o momento do preenchimento do questionário. Dentre os demais, 7 haviam cursado aproximadamente 50% das disciplinas e 10 afirmaram ter cursado mais de 50% das disciplinas do curso.

O questionário foi segmentado em diversas seções, começando pelo Termo de Consentimento Livre e Esclarecido, o qual os estudantes precisavam aceitar para acessar as perguntas. Na sequência, foram incluídas perguntas destinadas a identificar o perfil de cada participante, indagando se possuíam deficiência visual, o curso em que estavam matriculados e o percentual de disciplinas já concluídas até aquele momento.

Após essa fase introdutória, foram fornecidos detalhes sobre a organização das categorias e diretrizes. Posteriormente, foram apresentados os textos de cada diretriz, seguidos por perguntas destinadas à avaliação das mesmas. Para cada uma das 21 diretrizes, os participantes foram convidados a responder duas perguntas:

1. Uma questão que utilizava escala *Likert*<sup>1</sup> de 5 pontos. Nessa pergunta, os participantes avaliaram a consistência, clareza, utilidade e relevância de cada Diretriz, atribuindo valores correspondentes a cada uma das seguintes questões:
  - A recomendação é apresentada de forma clara e compreensível?
  - O texto da recomendação esclarece a forma como ela deve ser aplicada?
  - Os detalhes foram suficientes para a compreensão da recomendação?
  - A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?
  - As sugestões de implementação apresentadas em "Como fazer?" são úteis?
  - Há algum termo técnico ou jargão que precisa de mais explicação?
2. Uma questão aberta, permitindo que o participante expressasse sua opinião sobre a respectiva diretriz e fornecesse sugestões de melhoria, caso julgasse necessário e pertinente.

Por fim, o questionário incluía três perguntas que buscavam avaliar a percepção dos participantes em relação à estrutura geral e organização das Diretrizes:

1. Uma questão que também utilizava uma escala *Likert* de 5 pontos, na qual o participante deveria avaliar cada uma das perguntas a seguir:
  - As recomendações contribuem para a tomada de boas decisões de design?
  - As recomendações permitem obter um maior entendimento sobre deficiência visual?
  - As recomendações contribuirão para soluções mais inclusivas?

---

<sup>1</sup> Escala de resposta em que os participantes especificam seu nível de concordância com uma afirmação (Gil et al., 2002).

- As recomendações estão clara e consistentemente organizadas?
  - As recomendações são práticas e aplicáveis na realidade de desenvolvimento de IDEs?
  - O texto mantém uma abordagem coerente em toda a sua extensão?
  - Há alguma área específica que não foi considerada e que deveria ser incluída?
2. Uma questão aberta solicitava que o participante compartilhasse sugestões para inclusão, exclusão ou melhoria na organização, ou na redação das recomendações; e,
  3. Uma questão que perguntava se o participante teria interesse em participar de fases posteriores da pesquisa.

Todas as questões que utilizavam o formato de resposta em escala *Likert* de 5 pontos tinham valores variando de 1 a 5, representando: (1) Discordo Totalmente; (2) Discordo Parcialmente; (3) Não concordo, nem discordo; (4) Concordo parcialmente; e (5) Concordo totalmente.

Para evitar eventuais problemas de entendimento e na organização do questionário, antes de disponibilizá-lo aos estudantes, foi realizado um teste piloto com o professor de programação de computadores do Instituto Federal Farroupilha Campus São Vicente do Sul (IFFar-SVS). Algumas sugestões de ajustes foram feitas e as mudanças consideradas pertinentes foram executadas.

O questionário enviado aos estudantes está disponível para acesso em <https://forms.gle/wi45rpNnHBMZZq257>

### 6.2.1 Resultados

Os resultados obtidos são apresentados empregando duas abordagens distintas. Inicialmente, as perguntas estruturadas com escala *Likert* foram analisadas com o intuito de medir o nível de concordância ou discordância dos respondentes em relação às afirmações propostas. Em seguida, as respostas às questões abertas do questionário foram analisadas para capturar percepções e experiências dos participantes. Acredita-se que essas duas formas de análise sejam complementares e permitam uma compreensão mais abrangente das opiniões dos participantes em relação ao conteúdo das diretrizes avaliadas.

Para embasar as descobertas desta fase da pesquisa, foram selecionadas algumas citações dos participantes no intuito de ilustrar ou evidenciar a opinião específica de um determinado estudante sobre o tema em questão.

### 6.2.1.1 *Respostas das perguntas que utilizaram escala Likert*

Após analisar as respostas dos estudantes sobre a clareza, compreensibilidade, utilidade e relevância das diretrizes, pode-se inferir que, em geral, as recomendações foram bem recebidas pelos estudantes, que concordaram (total ou parcialmente) com a maioria das diretrizes. Houve casos específicos, como nas Diretrizes 1.1 e 2.3, em que cerca de 30% dos respondentes indicaram concordância parcial com todas as afirmações. Entretanto, poucas dessas respostas foram justificadas pelos participantes, dificultando o entendimento do que precisaria ser alterado, mas sugere a necessidade de ajustes para garantir uma melhor compreensão e aplicação dessas Diretrizes.

Outro ponto relevante é a observação de que algumas diretrizes receberam uma proporção significativa de respostas "concordo parcialmente" em relação à pertinência e especificidade para melhorar a acessibilidade para pessoas com deficiência visual e quanto à utilidade das sugestões de implementação apresentadas em "Como fazer?", apontando a necessidade de revisão destas recomendações. No entanto, é importante ressaltar que as sugestões de implementação apresentadas no tópico "Como fazer?" foram obtidas a partir da revisão da literatura ou foram sugestões dos próprios usuários que participaram do estudo qualitativo, e não limitam ou restringem a implementação a apenas essas formas de abordagem.

Por fim, ao serem questionados se havia algum termo técnico ou jargão que precisava de mais explicação em cada uma das Diretrizes, alguns estudantes indicaram que sim, selecionando a opção "concordo totalmente" ou "concordo parcialmente". Entretanto, apenas um participante mencionou a necessidade de maiores explicações, especificamente para o termo "alertas sonoros". Entre os demais, nenhum participante identificou o termo técnico ou jargão que necessitasse de mais explicações.

Acredita-se que as respostas a essa indagação possam não ter sido devidamente ponderadas, pois, em contraste com as demais questões, que solicitavam *feedbacks* positivos, esta inquiria sobre aspectos negativos, e as respostas eram inversamente interpretadas. Em outras palavras, caso não se vislumbrasse a necessidade de esclarecimentos adicionais, os participantes deveriam optar pela opção "discordo totalmente" ou "discordo parcialmente". Nas demais questões, ao concordarem com o enunciado, os usuários deveriam selecionar "concordo totalmente" ou "concordo parcialmente". A falta de respostas positivas pode indicar uma possível subavaliação dessa questão, devido ao seu caráter inverso em relação às demais perguntas.

Os gráficos gerados a partir das respostas dos participantes para cada uma das afirmações que avaliavam as diretrizes individualmente estão disponíveis no Anexo H.

### 6.2.1.2 *Respostas das perguntas abertas*

Nesta seção serão apresentados os resultados obtidos nas respostas às perguntas abertas, que visavam aprofundar a compreensão das percepções e experiências dos

participantes em relação às diretrizes propostas. Estas perguntas não eram obrigatórias, e, por esse motivo, poucos participantes forneceram comentários que justificassem suas respostas para as perguntas que utilizavam escala *Likert* ou que pudessem contribuir para a estrutura, organização ou escrita das Diretrizes.

No intuito de manter o anonimato dos participantes, optou-se por se referir a eles de forma genérica. Neste sentido, para a análise das respostas obtidas, os participantes serão identificados pela letra "E" seguida de um número, começando em E1 e indo até E20.

Adicionalmente, com o propósito de formalizar e organizar os ajustes e correções necessários, cada tópico será designado por um código composto por letras e números. Esse código será empregado como uma referência específica para identificar os itens que requerem revisão ou modificação futura.

Ao avaliar a **Diretriz 1.1 (Semântica)**, um dos participantes (E10) mencionou que o termo "alertas sonoros" não estava muito claro, e sugeriu que essa terminologia fosse explicada com mais detalhes no texto das Diretrizes **EC01**. Outro estudante, E13, expressou dúvidas quanto à utilização desses "alertas sonoros" para garantir a acessibilidade dos IDEs **EC02**. O estudante fez o seguinte questionamento:

E13: "(...) é mencionado o uso de alertas sonoros para transmitir informações. Poderia ser implementado diferentes alertas sonoros para cada tipo de informação?"

Outro exemplo de dúvidas relacionados à utilização de "Alertas Sonoros" surgiu durante a avaliação da **Diretriz 2.1 (Informações sobre erros)**, mais especificamente no **Critério de sucesso 2.1.1** (Os erros de sintaxe são informados em tempo real.). O mesmo estudante, E13, enfatizou que:

E13: "Muitas vezes os erros são detectados antes mesmo do usuário terminar de digitar o código, sendo assim, se toda vez que um erro for detectado disparar um alerta sonoro, pode ser irritante dependendo de quantas vezes o alerta é disparado (...)."

A utilização de "alertas sonoros" é abordada por uma Diretriz específica no documento: **Diretriz 7.1 (Alertas sonoros)**. Contudo, essa orientação é encontrada em uma seção mais avançada do texto das Diretrizes. As dúvidas manifestadas pelos participantes evidenciam que, até alcançarem essa seção do documento, os usuários podem encontrar diversas incertezas em relação a esse tipo de recurso.

Dois participantes (E13 e E16) sugeriram que o texto do **Critério de sucesso 2.2.1** (É possível obter informações como nome, valor, tipo, qualificador e escopo de variáveis e constantes), da **Diretriz 2.2 (Variáveis e constantes)**, fosse revisado. Segundo E16, seria interessante permitir que o usuário modifique o nome, o valor ou o tipo das variáveis a partir de uma lista de variáveis declaradas fornecida pelo sistema. Isso

eliminar a necessidade de localizar e alterar essas informações em todas as ocorrências de uma variável no código. Dessa forma, ao efetuar a alteração na lista, o software ajustaria automaticamente as informações em todo o código **EC03**.

Em relação à **Diretriz 3.8 (Ajuda e Documentação)**, mais especificamente no **Critério de Sucesso 3.8.4** (A cada nova versão do IDE, o usuário é informado sobre funcionalidades adicionadas e/ou removidas), um participante (E10) sugeriu que, durante a instalação das novas versões do IDE, o usuário seja informado sobre as atualizações realizadas e como essas mudanças impactam na acessibilidade do sistema **EC04**. Além disso, o sistema poderia oferecer a opção ao usuário de escolher quais elementos deseja atualizar, assim como quais configurações ou recursos de acessibilidade da versão anterior ele gostaria de manter **EC05**. A justificativa para essa sugestão é que, ao longo do tempo, o usuário pode automatizar algumas tarefas, deixando de depender de recursos de acessibilidade que antes eram essenciais, conforme for aprimorando o conhecimento a respeito da ferramenta. Alterações inesperadas ou a necessidade de configurar todos os recursos de acessibilidade na nova versão podem ocasionar transtornos e demandar uma adaptação que consome tempo e esforço desnecessários.

Outro participante (E14) observou que o item "Descrição" da **Diretriz 5.1 (Dobra de Código)** não abrangia todos os tópicos abordados no texto da diretriz e sugeriu que o texto fosse revisado para incluir todas essas características **EC06**.

Em se tratando da **Diretriz 5.2 (Visão geral do código)**, mais especificamente em relação ao **Critério de sucesso 5.2.2** (É possível obter uma lista de todos os arquivos de código-fonte de um projeto de software) e ao **Critério de sucesso 5.2.4** (É possível navegar entre os arquivos de código-fonte em um projeto de software), o estudante E13 destaca que um projeto de software pode conter diversos tipos de arquivos, não se limitando apenas aos arquivos de código-fonte **EC07**. Dentre os outros tipos de arquivos pode-se citar: arquivos binários (executáveis, bibliotecas compartilhadas, arquivos de imagem, entre outros); arquivos de configuração (armazenam propriedades e parâmetros do projeto, por exemplo); arquivos de documentação (manuais, documentação técnica, diagramas, etc.); entre outros.

No geral, a avaliação individual das diretrizes mostrou-se positiva, destacando alguns itens específicos que requerem revisão. Todas as observações e comentários fornecidos pelos participantes foram analisados, e as alterações pertinentes foram incorporadas ao documento final das Diretrizes para assegurar que estas estejam o mais alinhadas possível com as necessidades e perspectivas dos estudantes com deficiência visual. O processo de revisão das sugestões e alteração do texto das diretrizes foi realizado em duas etapas, descritas nas Seções 6.4 e 7.5.

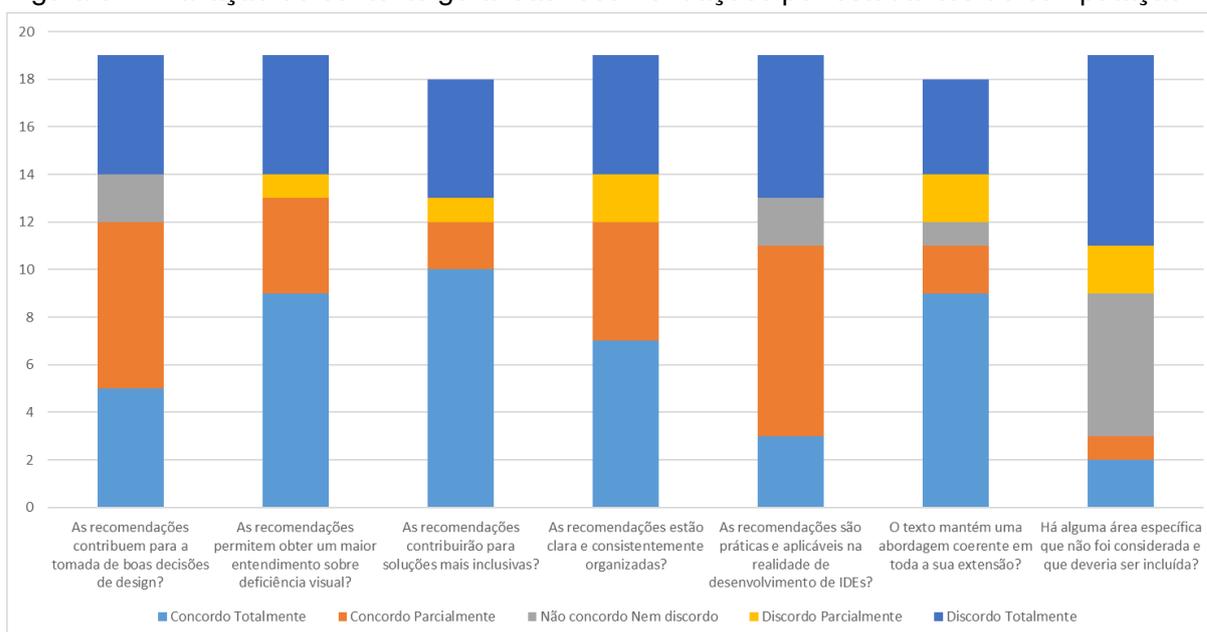
### 6.2.1.3 Avaliação das estrutura e organização das Diretrizes

A avaliação do contexto geral das recomendações preliminares pelos estudantes de computação apresentou resultados bastante positivos. Cerca de 63% dos participantes concordam (parcial ou totalmente) que as recomendações contribuem para boas decisões de design, enquanto 68% concordam (parcial ou totalmente) que facilitam a compreensão sobre deficiência visual.

Adicionalmente, 67% dos participantes acredita que as diretrizes podem promover o desenvolvimento de soluções mais inclusivas. Por outro lado, apenas 58% dos participantes consideram-nas práticas para a promover a acessibilidade dos IDEs, somente metade deles (50%) concorda com a coerência do texto em sua totalidade e 53% identificam que algumas áreas não foram consideradas e que deveriam ser incluídas nas recomendações. No entanto, somente um estudante (E17) contribuiu com informações para embasar essa resposta, expressando preocupação com a quantidade de atalhos necessários para cobrir todas as funcionalidades do IDE.

A Figura 8 apresenta um apanhado geral das respostas dos estudantes de computação quanto ao contexto geral das recomendações.

Figura 8 – Avaliação do contexto geral das recomendações por estudantes de computação



Embora a análise tenha revelado predominantemente resultados favoráveis, é imprescindível reconhecer que persistem áreas passíveis de aprimoramento na estrutura e organização das recomendações, demonstrando a necessidade contínua de revisão e refinamento do documento.

Além disso, diante da escassez de detalhes fornecidos nas respostas dos estudantes, optou-se por realizar um grupo focal para explorar mais profundamente suas

percepções e experiências a respeito do conteúdo das diretrizes. O objetivo foi obter informações mais detalhadas que pudessem contribuir para o seu aprimoramento.

### 6.3 2ª Etapa de Validação: Grupo Focal

Concluída a avaliação por meio de questionário, todos os 19 estudantes do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas foram convidados a participar de um Grupo Focal para discutir o conteúdo e a organização das Diretrizes. Treze estudantes do 4º semestre (E4, E5, E8, E9, E10, E11, E14, E15, E16, E17, E18, E19 e E20), matriculados na Disciplina de Interação Humano-Computador, concordaram em participar desta etapa da pesquisa.

O Grupo Focal ocorreu no mês de dezembro de 2023 em um Laboratório de Informática do Instituto Federal Farroupilha Campus São Vicente do Sul e seguiu o roteiro especificado no Anexo E. A atividade iniciou com a leitura do Termo de Consentimento Livre e Esclarecido (TCLE) e o Termo de Consentimento para Uso da Imagem e/ou som da Voz (TUIV), disponíveis nos Anexos L e C, respectivamente.

A atividade foi organizada em dois momentos distintos, começando com uma tarefa prática seguida por uma discussão sobre as percepções dos participantes em relação à execução da tarefa e sua relevância em relação ao conjunto de Diretrizes.

A primeira tarefa consistia na realização de um exercício prático de programação. Para isso, a turma foi dividida em duplas ou trios, e cada grupo recebeu a tarefa de criar uma classe em Linguagem Java para exibir todos os números pares entre 1 e 10. O exercício foi elaborado com um nível de dificuldade mínimo, levando em consideração que os estudantes já possuíam domínio dos diferentes aspectos relacionados à lógica exigida para a resolução do problema, assim como da linguagem de programação e do IDE necessários para a realização da atividade.

Para realizar a tarefa, era exigida a utilização do IDE VSCode<sup>2</sup>, ferramenta conhecida e dominada por todos os estudantes, juntamente com o leitor de telas NVDA<sup>3</sup>. Ambos os softwares foram instalados previamente nos computadores.

Os estudantes tiveram que escolher um membro de cada grupo para desempenhar o papel de "programador", responsável por elaborar a lógica de resolução do exercício e implementá-lo em linguagem Java. Após essa escolha, os estudantes foram informados de que o "programador" não teria acesso ao monitor enquanto estivesse escrevendo o código do programa. Recomendou-se que esse estudante utilizasse a técnica *think aloud*<sup>4</sup> para compartilhar com os colegas de grupo todas as suas decisões, reflexões e as dificuldades encontradas durante a realização da atividade.

---

<sup>2</sup><https://code.visualstudio.com/>

<sup>3</sup><https://www.nvaccess.org/download/>

<sup>4</sup>Quando é solicitado que o participante relate em voz alta o que ele está pensando e fazendo durante a interação (Barbosa; Silva, 2010).

Figura 9 – Organização dos grupos de atividade prática do Grupo Focal



Os demais participantes deveriam observar a execução da atividade pelo "programador" e fazer anotações de tudo que fosse dito por ele, assim como registrar o tempo dedicado para a tarefa, se foi concluída com êxito e a quantidade de erros cometidos (se houvesse). Além disso, foi solicitado que esses estudantes não interferissem ou auxiliassem o "programador" na realização da tarefa, atuando apenas de observadores. A Figura 9 exemplifica a organização dos grupos para a execução da atividade.

No intuito de auxiliar na realização da atividade, os estudantes receberam fone de ouvido e uma lista contendo os atalhos de teclado necessários para a utilização do leitor de telas e do IDE VSCode.

Para iniciar a atividade prática, os estudantes contavam com um arquivo contendo a estrutura básica da classe Java salvo em seus computadores (Figura 10). Portanto, era preciso que o "programador" inserisse apenas a lógica necessária para a resolução do problema solicitado, sem o auxílio do monitor.

Figura 10 – Estrutura básica da classe Java necessária para a atividade prática do Grupo Focal

```
public class ExibePares {  
  
}
```

Ao final da tarefa, o código que implementa a funcionalidade solicitada deveria ser semelhante à representação apresentada na Figura 11. É possível reconhecer que poucas linhas de código seriam suficientes para resolver o exercício.

Figura 11 – Exemplo de resolução para o problema solicitado na atividade prática do Grupo Focal

```
public class ExibePares {  
    public static void main(String[] args) {  
        for (int i = 0; i < 10; i++) {  
            if (i % 2 == 0)  
                System.out.println(i + " é um número par!");  
        }  
    }  
}
```

A atividade prática foi executada duas vezes, alternando o papel de "programador" entre os diferentes estudantes de cada grupo. Ou seja, em cada rodada, um estudante diferente do grupo assumiria o papel de "programador", enquanto os demais atuariam como observadores. Para cada rodada, cada "programador" teve 15 minutos para realizar a atividade. No entanto, nenhum dos grupos conseguiu concluir a tarefa.

Concluído o prazo estabelecido, procedeu-se à discussão das percepções dos estudantes em relação às dificuldades encontradas para concluir a atividade prática e das diretrizes previamente avaliadas por eles. Essa segunda etapa do grupo focal ocorreu imediatamente após o término da tarefa prática, em uma sala de aula próxima ao Laboratório de Informática onde a atividade prática foi realizada. Os estudantes foram organizados em classes formando um semicírculo. Essa atividade foi registrada em áudio e vídeo para posterior transcrição e análise das intervenções verbais.

### 6.3.1 Resultados

As contribuições fornecidas pelos estudantes que participaram do Grupo Focal possibilitaram uma compreensão mais aprofundada das percepções desse grupo de participantes em relação à versão preliminar das diretrizes propostas, complementando os resultados obtidos por meio dos questionários.

Seguindo a organização adotada neste Capítulo, no intuito de formalizar e organizar os ajustes e correções necessários no documento que contém as diretrizes, cada tópico será designado por um código composto por letras e números. Tal código será utilizado para referenciar os tópicos específicos que requerem revisão ou modificação futura.

A segunda etapa do Grupo Focal iniciou com a pesquisadora questionando a respeito das dificuldades encontradas pelos estudantes durante a execução da atividade prática. Inicialmente, todos os estudantes relataram problemas para aprender a utilizar o leitor de telas e interagir com o IDE sem o auxílio do monitor. Essa dificuldade já era esperada, visto que nenhum deles havia interagido com o computador de maneira semelhante anteriormente.

Além disso, outro ponto destacado pelos participantes estava relacionado à compreensão das mensagens de erro fornecidas pelo IDE. A pronúncia dessas mensagens realizada pelo leitor de telas não era clara e não facilitou a identificação e correção de problemas. Os estudantes sugeriram que essas mensagens fossem aprimoradas para fornecer informações mais relevantes, auxiliando de maneira mais eficaz o usuário. Algumas falas dos participantes confirmam essa informação:

E16: *"Eu acho que o erro é muito mal explicado, ele é muito, podia ser muito mais simples."*

E9: *"(...) ele lê, ele fala o que o IDE mostra no erro. Ele não, meio que filtra, ele só lê o erro. (...), e ele me falou também que um monte de código para, para esse mesmo erro, que nem enxergando eu sei o que significa."*

No entanto, quando questionados se essas barreiras já haviam sido previstas no texto das Diretrizes, os participantes reconheceram que o **Critério de Sucesso 2.1.4** (As mensagens de erro contêm informações sobre a causa do erro, sua localização no código-fonte e como corrigi-lo), da **Diretriz 2.1 (Informações sobre erros)**, já abordava esse tipo de desafio. Segundo os estudantes, não seria necessário alterar o Critério de Sucesso, tampouco a Diretriz, pois contemplava completamente o problema identificado.

Outro tópico ressaltado por alguns estudantes (E9, E11, E16, E17 e E18) está relacionado à **Diretriz 4.2 (Autocompletar)**, mais especificamente à utilização de caracteres não alfanuméricos, especialmente aqueles empregados para indicar o início e fim de uma instrução ou bloco de código, expressões condicionais e parâmetros de funções (tais como chaves, colchetes, parênteses, etc.). Os estudantes informaram que, quando inseriam uma chave, parêntese ou colchete, não eram notificados sobre a inserção automática do caractere de fechamento correspondente, realizada pelo **GF01**, como exemplificado nos trechos a seguir:

E11: *"(...) e se tu coloca uma chave, não sabe se fecha no final."*

E16: *"Eu não sabia se, se tipo... não fazia ideia, não sabia se se eu desse o ENTER no 'abre chaves' ele ia fechar embaixo. (...) Tanto que eu peguei, eu mesmo fechei sozinho, então ficou um monte de chaves sobrando. "*

E9: *"(...) sobrou muita chave. E parênteses também."*

Ainda em relação à **Diretriz 4.2 (Autocompletar)**, segundo os estudantes E9 e E18, o recurso realizava a leitura das sugestões automaticamente, sem que o usuário solicitasse, o que atrapalhava a compreensão e causava confusão **GF02**.

Alguns estudantes (E9 e E16) também manifestaram dúvidas sobre o que estava sendo lido pelo leitor de telas em um determinado momento, pois não havia indicava se a informação que estava sendo lida trata-se de um comentário, do conteúdo do

recurso autocompletar, ou de uma instrução de código-fonte **GF03**.

Quando o usuário não tem certeza do que está sendo lido pelo leitor de telas, pode encontrar dificuldade em compreender a informação, conforme destacado pelos estudantes:

E16: *"Ah, era bom eu saber o que que eu tô fazendo. Tipo, (...) se ele tá me lendo o que eu escrevi, se ele tá lendo o que eu comentei, se ele tá lendo uma sugestão, se ele tá lendo o que eu apaguei."*

E9: *"Muitas das vezes eu sabia que ele tava, muitas das vezes eu sabia que ele tava lendo uma, uma sugestão porque eu não escrevia aquilo, eu sabia que eu não tinha escrito aquilo, e ele pegou e começou a ler. Só que eu não sabia, claro, que era uma sugestão. "*

Alguns participantes (E4, E14, E16 e E18) também expressaram dúvidas em relação à construção de interfaces gráficas por estudantes cegos. Segundo eles, além da **Diretriz 6.1 (Validação da interface gráfica)**, que recomenda que o IDE deveria realizar validação automatizada da interface gráfica gerada a partir de um código-fonte, seria importante que o IDE fornecesse recursos para auxiliar esses usuários nas tarefas que envolvem escrita de código-fonte para construção da interface gráfica **GF04**. Isso poderia ser feito fornecendo informações que colaborassem para o processo de desenvolvimento, reduzindo a necessidade de auxílio de outras pessoas, como professores ou colegas.

E16 destacou a importância de algumas Diretrizes para usuários cegos, como a **Diretriz 3.6 (Numeração de linhas)**, fundamental para facilitar o entendimento da localização do cursor do teclado em um arquivo de código-fonte. Da mesma forma, o estudante E9 ressaltou a relevância da **Diretriz 8.2 (Leitura contextual)**, dada a significativa dificuldade na compreensão do código lido pelo leitor de telas:

E9: *"(...) uma coisa que eu achei, que eu acho muito importante é a questão da interpretação do código. Porque... muito do leitor de tela ele lê exatamente o que tá ali e ele não, por exemplo, tá ele poderia tanto me dar exatamente o que eu escrevi como me explicar o que que eu escrevi (...)."*

Outros apontamentos estavam relacionados à leitura realizada pelo leitor de telas (E14, E15 e E18), mas não estão restritos exclusivamente à leitura da interface dos IDEs, uma vez que esses problemas também ocorrem em outros softwares. A primeira observação, por exemplo, está relacionada à ação de **apagar** uma letra, palavra ou trecho de código, pois, segundo os estudantes, o leitor de telas não informa o que foi apagado:

E15: *"Uma dificuldade que a gente encontrou é que na hora que nós tentamos apagar alguma coisa, e a gente não sabia o que que tava apagando. Não dizia: 'você apagou tipo, com ponto e vírgula'".*

E18: "(...) porque quando tu tá escrevendo ele, ele fala que tá escrevendo, só que quando tu apaga, não sabe o que tu apagou."

E14: "Para mim no caso ele falou... só o que ficou. Então, eu meio que fiquei perdido (...)."

Além disso, E15 e E16 relataram dificuldades em **identificar** se os caracteres lidos pelo leitor de telas estavam em **letra maiúscula ou minúscula**. Essa informação é crucial para o programador, considerando que a maioria das linguagens de programação seja *Case sensitive*, ou seja, faz distinção entre maiúsculas e minúsculas. Outras dificuldades foram identificadas quando o usuário utilizou atalhos do teclado para selecionar um trecho de código ou utilizar os recursos para **copiar/recortar e colar** partes do código. Esses desafios surgem porque, conforme relatado pelos estudantes E14 e E18, o leitor de telas não informa o que foi selecionado, copiado/recortado e colado.

Apesar de os estudantes terem enfrentado esses tipos de obstáculos durante a realização da atividade prática, essas dificuldades podem ser atribuídas ao fato de os estudantes estarem utilizando um leitor de tela pela primeira vez. Esses desafios poderiam ser encontrados durante a interação com qualquer software que exigisse a eliminação de uma letra ou texto, não estando exclusivamente relacionada à interação com IDEs.

Alguns leitores de tela oferecem ao usuário a capacidade de identificar quando uma letra está em maiúscula ou minúscula. No Narrador (Microsoft, 2024), por exemplo, essa identificação é feita por meio da pronúncia da palavra "cap" antes da letra em questão, ou pela leitura da palavra/letra em um tom mais alto em comparação com as demais.

Em relação ao uso das teclas de atalho relacionadas às operações da área de transferência (copiar, recortar, colar e selecionar tudo), a maioria dos leitores de tela não informa quando essas operações são efetivamente executadas pelo IDE, caso do NVDA. No entanto, existem complementos disponíveis que podem ser instalados e têm a capacidade de anunciar a funcionalidade executada sempre que o usuário empregar as combinações de teclas "CTRL + X", "CTRL + C", "CTRL + V" ou "CTRL + A". Problemas podem surgir se o IDE utilizar conjuntos de teclas diferentes para essas funcionalidades.

Por fim, E14 e E16 também relataram que, ao digitar o código-fonte necessário para resolver o problema proposto, formavam uma imagem mental do código a ser escrito. De acordo com eles, essa abordagem prejudicou a execução da atividade, pois não conseguiam prestar atenção total no que estava sendo pronunciado pelo leitor de telas. Além disso, os estudantes afirmaram que o barulho da sala interferiu na tarefa de acompanhar o que o leitor de telas estava pronunciando.

A atividade prática conduzida no Grupo Focal foi, conforme relatado por todos os participantes, de significativa importância para que pudessem compreender, mesmo

que de maneira limitada, as barreiras e desafios enfrentados por estudantes cegos em programação. Segundo E18, um desafio potencial para um estudante cego poderia surgir, por exemplo, ao realizar um teste de mesa, prática comum na programação. Nesse processo, o código é executado manualmente para verificar a lógica, identificar erros e garantir o correto funcionamento do programa. Quando não há ferramentas acessíveis que auxiliem na execução desta tarefa, pode ser difícil acompanhar o fluxo do programa ou rastrear as mudanças nas variáveis, aspectos essenciais durante a realização de testes de mesa. Salienta-se que alguns desses aspectos já foram abordados nas **Diretriz 2.2 (Variáveis e Constantes)**.

## 6.4 Refinamento das Diretrizes

Esta seção tem como objetivo fornecer uma visão do panorama resultante das investigações descritas ao longo deste Capítulo e apresentar como esses resultados influenciaram na customização do texto da versão preliminar das Diretrizes de Acessibilidade em IDEs para Estudantes com Deficiência Visual, resultando na 1ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos<sup>5</sup>.

É importante ressaltar que nem todas as sugestões de melhoria coletadas foram implementadas nesta etapa. Optou-se por realizar apenas alterações que envolvessem a inclusão ou exclusão total de alguma diretriz. As sugestões que requeriam modificações no conteúdo das diretrizes que permaneceram no documento foram adiadas para serem realizadas em uma etapa posterior, descritas na Seção 7.5. Essa decisão foi tomada com base na necessidade de uma avaliação mais detalhada dessa versão do documento antes de efetuar modificações adicionais, levando em consideração que outros perfis de avaliadores podem ter percepções diferentes. Entende-se que é importante contrapor essas opiniões para garantir que as diretrizes atendam da melhor forma possível às necessidades dos usuários com deficiência visual.

Uma das principais constatações da etapa da pesquisa descrita neste Capítulo está relacionada ao fato de não terem sido coletadas informações suficientes para abordar diretrizes voltadas para usuários com baixa visão de forma adequada. Isso se deve à limitada participação de estudantes com esse perfil nas etapas que compuseram o Estágio Exploratório (Capítulo 4). Além disso, a pesquisa concentrou-se principalmente nos desafios e barreiras relacionados à utilização de leitores de tela, comumente utilizados por estudantes cegos. Os recursos de Tecnologia Assistiva (TA) utilizados especificamente por estudantes com baixa visão não foram analisados com a profundidade necessária.

Diante dessa constatação, optou-se por direcionar a 1ª versão das Diretrizes de Acessibilidade para atender às necessidades e preferências do grupo específico de

---

<sup>5</sup><https://elianazen.github.io/DiretrizesAcessibilidadeIDE/1VersaoDiretrizes.html>

estudantes cegos, especialmente relacionadas aos desafios encontrados quando se utiliza leitores de tela.

Nesse contexto, a primeira modificação realizada no documento contendo as recomendações foi a alteração do título, que passou a se chamar "**Diretrizes de Acessibilidade em Ambientes de Desenvolvimento Integrado (IDEs) para Estudantes Cegos**". Além disso, o texto que compõe o corpo do documento também necessitou de ajustes, substituindo-se todas as ocorrências do termo "**estudantes com deficiência visual**" por "**estudantes cegos**".

Diante desse mesmo contexto, a **Categoria 3 (Navegação no Código)**, que inicialmente continha 8 diretrizes na versão preliminar, teve a **Diretriz 3.2 (Ampliação e contraste)** suprimida. Essa decisão foi tomada porque essa diretriz estava diretamente relacionada às necessidades de estudantes com baixa visão. Conseqüentemente, a **Categoria 3 (Navegação no Código)** passou a contar apenas com 7 diretrizes, conforme evidenciado na Tabela 7.

Tabela 7 – Reestruturação das Diretrizes da Categoria 3 (Navegação no Código)

<b>Versão Preliminar</b>	<b>1ª Versão das Diretrizes</b>
Diretriz 3.1 Rótulos	Diretriz 3.1 Rótulos.
Diretriz 3.2 Ampliação e contraste	Diretriz 3.2 Movimentação pelo código
Diretriz 3.3 Movimentação pelo código.	Diretriz 3.3 Contexto e nível de escopo
Diretriz 3.4 Contexto e nível de escopo	Diretriz 3.4 Estratégias de navegação
Diretriz 3.5 Estratégias de navegação	Diretriz 3.5 Numeração de linhas
Diretriz 3.6 Numeração de linhas	Diretriz 3.6 Atalhos
Diretriz 3.7 Atalhos	Diretriz 3.7 Ajuda e Documentação
Diretriz 3.8 Ajuda e Documentação	

Outra modificação realizada ocorreu na **Categoria 6 (Compreensão da Saída)**, visando atender à sugestão dos participantes de auxiliar estudantes cegos na construção da interface gráfica **GF04**. A **Diretriz 6.1** foi renomeada para "**Construção de Interface Gráfica**", e passou a abordar recomendações que visam auxiliar a construção de interfaces que utilizam componentes visuais por estudantes cegos. Como resultado, a **Diretriz 6.1 (Validação da interface gráfica)** das recomendações preliminares passou a ser numerada como **Diretriz 6.2** na 1ª versão das diretrizes.

Para adequar-se a todas essas mudanças e englobar todos os tópicos abrangidos nas **Diretrizes 6.1 e 6.2**, a **Categoria 6**, na versão preliminar denominada "**Compreensão da Saída**", teve o título alterado para "**Interface Gráfica de Usuário**", resultando em uma descrição mais abrangente das orientações incorporadas nesta Categoria **GF04**. Dessa forma, a Categoria 6 ficou organizada da seguinte maneira:

- Diretriz 6.1 Construção de Interface Gráfica. Fornecer mecanismos para facilitar a construção de interfaces que utilizam componentes visuais.

- Diretriz 6.2 Validação da interface gráfica. Realizar validação automatizada da interface gráfica gerada a partir de um código-fonte.

Para a 1ª versão das Diretrizes, portanto, não foram efetuadas mudanças relacionadas aos itens que propunham modificações nos textos das diretrizes que permaneceriam no documento: **EC01**, **EC02**, **EC03**, **EC04**, **EC05**, **EC06**, **EC07**, **GF01**, **GF02** e **GF03**. Ressalta-se que todas as sugestões apresentadas foram analisadas juntamente com aquelas obtidas durante a segunda iteração de validação das Diretrizes, descrita no Capítulo 7.

## 6.5 Considerações Parciais

Esta seção descreveu o primeiro ciclo de execução dos Estágios de Validação e Refinamento do conjunto de diretrizes preliminares, contando com a participação de estudantes de cursos superiores na área de Computação.

Os resultados obtidos conduziram ao direcionamento do conteúdo das diretrizes exclusivamente para estudantes cegos, uma vez que as características, necessidades e preferências de estudantes com outros tipos de deficiência visual não foram estudadas com a profundidade necessária para englobá-los nessa versão das diretrizes.

As contribuições obtidas nesta etapa foram parcialmente implementadas, pois optou-se por analisar um conjunto dessas sugestões juntamente com os resultados obtidos da segunda iteração de avaliação, conforme descrito no Capítulo 7. Somente foram efetuadas alterações que exigiam a inserção ou eliminação de diretrizes para esta 1ª Versão.

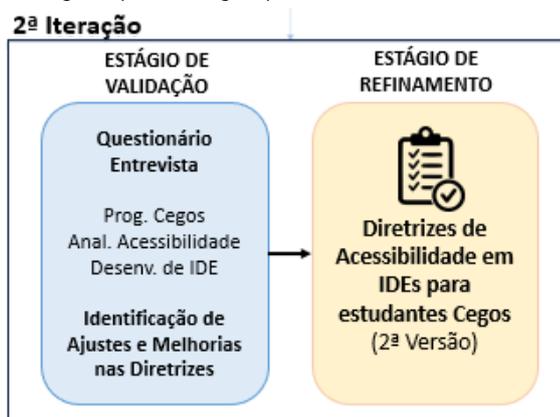
A 1ª Versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos pode ser acessada através do link <https://elianazen.github.io/DiretrizesAcessibilidadeIDE/1VersaoDiretrizes.html>.

## 7 VALIDAÇÃO E REFINAMENTO DAS DIRETRIZES (2ª ITERAÇÃO)

A avaliação da 1ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos foi conduzida com a participação de especialistas no domínio da aplicação. Esses especialistas contribuíram com suas opiniões sobre o texto das Diretrizes, fundamentadas em seu conhecimento e experiência nesta área específica.

Como ilustrado na Figura 12, neste Capítulo são delineados os procedimentos empregados na segunda etapa de execução dos Estágios de Validação e Refinamento, juntamente com os resultados alcançados, culminando com a 2ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos, que será apresentada no Capítulo 8.

Figura 12 – Estágio de Validação (2ª Iteração)



Esta etapa de validação foi aprovada pelo Parecer Consubstanciado n.º 68361023.1.0000.5574 emitido pelo Comitê de Ética em Pesquisa com Seres Humanos no Instituto Federal Farroupilha (IFFar), disponível no Apêndice B.

### 7.1 Caracterização do Estudo

Nesta fase da pesquisa, buscou-se validar a 1ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos utilizando o método de avaliação por es-

pecialistas no domínio da aplicação. O grupo de participantes incluiu dois egressos e um estudante do Curso de Computação, todos com deficiência visual, sendo que este último atua profissionalmente como programador há mais de 2 anos. Além disso, participaram um analista de acessibilidade que trabalha na Google do Brasil<sup>1</sup>, que também é cego, e um professor de curso superior de Computação que está envolvido no desenvolvimento de uma IDE brasileira chamada Portugol Studio<sup>2</sup> (Noschang et al., 2014).

O recrutamento desse grupo de participantes foi conduzido por meio de contatos dos pesquisadores, listas de e-mails específicas e chamadas à participação divulgadas em formulários de contato de empresas relacionadas ao desenvolvimento de IDEs. Além disso, a divulgação foi feita nos grupos do Facebook<sup>3</sup> e páginas Instagram<sup>4</sup> dessas empresas, bem como no grupo de discussão "Cegos Programadores<sup>5</sup>", que reúne brasileiros com deficiência visual com interesse na área de tecnologia.

O convite enviado incluía informações sobre a pesquisadora responsável, os objetivos do estudo e os procedimentos que seriam adotados para a avaliação das diretrizes. Na solicitação de participação, foi inserido um *link* para um questionário, solicitando que o participante indicasse seu interesse em participar da pesquisa. O convite enviado por e-mail está disponível no Anexo G, e o questionário de manifestação de interesse pode ser acessado através do endereço: <https://forms.gle/Zm7KHpZbrNDRNwpN7>. O mesmo texto foi utilizado em todas as chamadas à participação enviadas e, quando necessário, tanto os e-mails quanto o questionário de manifestação de interesse eram traduzidos para língua inglesa.

Após a manifestação de interesse, o texto correspondente à 1ª Versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos foi enviado por e-mail em formato PDF, juntamente com os links para a página *Web* que continha o texto das diretrizes (<https://elianazen.github.io/DiretrizesAcessibilidadeIDE/>) e para o questionário de avaliação, acessível em <https://forms.gle/ydQEGZdeYcfrwEo9>.

O questionário disponibilizado nesta segunda etapa da validação das diretrizes seguiu o mesmo formato do questionário apresentado na Seção 6.2 do Capítulo 6, disponibilizado aos estudantes de computação. Entretanto, nesta versão o texto completo das Diretrizes não foi apresentado, visto que o arquivo PDF, bem como o *link* para acesso à página *Web* contendo as Diretrizes já haviam sido enviados por e-mail antecipadamente. Para responder às perguntas do questionário, primeiramente, era necessário concordar como TCLE apresentado.

---

<sup>1</sup><https://www.google.com/>

<sup>2</sup><https://univali-lite.github.io/Portugol-Studio/>

<sup>3</sup><https://www.facebook.com/>

<sup>4</sup><https://www.instagram.com/>

<sup>5</sup>[cegos\\_programadores@googlegroups.com](mailto:cegos_programadores@googlegroups.com)

Um dos participantes, que atua como analista de acessibilidade, optou por realizar uma entrevista para discutir o texto das diretrizes em vez de responder ao questionário enviado. A entrevista foi conduzida no mês de dezembro de 2023 e teve uma duração aproximada de 30 minutos. A entrevista foi registrada em áudio e vídeo e sua transcrição completa foi realizada para análise posterior. É importante destacar que o participante concordou previamente com o (TCLE) e o (TUIV), que foram enviados previamente por e-mail em arquivos individuais.

Os resultados apresentados neste Capítulo foram estruturados de acordo com o perfil dos participantes: programadores cegos, analista de acessibilidade e professor de computação envolvido no desenvolvimento de IDE, no intuito de analisar as diferentes perspectivas e contribuições específicas de cada grupo.

Novamente, para garantir o anonimato dos participantes, quando necessário, eles serão identificados por uma letra (P) seguida de um número, e algumas citações poderão ser utilizadas para exemplificar a percepção ou opinião manifestada por algum deles.

## 7.2 Validação por Programadores Cegos

A partir da divulgação do convite no grupo de discussão Cegos Programadores, cinco pessoas inscritas manifestaram interesse em participar da pesquisa. No entanto, apesar de várias tentativas de contato, lamentavelmente, nenhum delas retornou o questionário de avaliação enviado.

Também foram convidados a participar desta fase da pesquisa três indivíduos que já haviam se envolvido em etapas anteriores do estudo (Capítulo 4), todos cegos. Os três aceitaram o convite. Entre eles, dois (P1 e P2) são egressos de Curso Técnico em Informática e Cursos Superiores de Computação, sendo um formado em Tecnologia em Análise e Desenvolvimento de Sistemas e o outro em Licenciatura em Computação. O terceiro participante (P3) é estudante de Curso de Bacharelado em Ciência da Computação e atua profissionalmente com programação há pelo mais de dois anos. Os outros dois participantes não possuem experiência profissional na área.

Esta etapa da avaliação teve como objetivo avaliar se as recomendações abrangiam as necessidades de acessibilidade vivenciadas por esses usuários ao utilizarem IDEs e a percepção dos mesmos em relação ao conjunto de recomendações.

### 7.2.1 Resultados

As perguntas que empregavam a escala *Likert* foram respondidas por todos os participantes. No entanto, apenas P3, que atua profissionalmente com programação de computadores, preencheu as questões abertas que solicitavam que compartilhassem suas opiniões sobre o texto de cada diretriz e sugerissem melhorias que julgassem

necessárias e pertinentes.

P1 e P2 concordam plenamente com a grande maioria das afirmações relacionadas às 21 diretrizes apresentadas. P3, por outro lado, expressou algumas incertezas e declarou concordar apenas parcialmente com algumas das afirmações ligadas às diretrizes. Além disso, esse participante também ofereceu diversos comentários fundamentados em suas experiências com o uso de IDEs para realizar atividades de programação, os quais serão delineados a seguir.

Por exemplo, P3 demonstrou concordância parcial em relação à pertinência e especificidade da **Diretriz 1.1 (Semântica)** para melhorar a acessibilidade para pessoas com deficiência visual, mais especificamente em relação ao **Critério de sucesso 1.1.4** (É possível obter informações contextuais de cada função ou método de um arquivo de código-fonte). O participante recomendou que o termo "finalidade" fosse substituído **PC01**, pois segundo ele:

P3: "(...) dá a entender que o IDE deveria interpretar o código para determinar a 'finalidade' daquele método. Isso pode ser uma tarefa bastante custosa para o IDE, no entanto muitos programadores acabam escrevendo uma documentação do método que pode ser lida nessa fase, sem necessidade de interpretar o método para descobrir sua finalidade."

Em se tratando da **Diretriz 2.1 (Informações sobre erros)**, embora tenha concordado totalmente com todas as afirmações, P3 sugeriu que fosse fornecida uma forma acessível de informar ao usuário cego sobre o sucesso ou falha de um teste unitário em um programa **PC02**. Habitualmente, os IDEs utilizam cores para indicar se um teste unitário foi bem-sucedido ou não. Quando essas informações não são apresentadas em um formato acessível para leitores de tela, a informação visual associada à cor pode ser perdida, impedindo que o programador cego saiba se o teste unitário foi bem-sucedido.

Quanto à **Diretriz 2.2 (Variáveis e constantes)**, P3 sugeriu que o IDE pudesse informar o usuário sobre os modificadores de acesso<sup>6</sup> das variáveis, utilizando sons para indicar se foram declaradas como *public*, *private* ou *protected* **PC03**. Essa abordagem sonora poderia aprimorar a interpretação do código-fonte, proporcionando uma compreensão mais eficaz e precisa.

Na **Diretriz 2.3 (Marcadores e Pontos de interrupção)**, o participante sugeriu a inclusão de um novo Critério de Sucesso. Segundo ele, além de permitir a identificação, listagem e inserção de marcadores ou pontos de interrupção no código, é importante que o IDE notifique o usuário quando a execução do programa for interrompida devido à presença de um ponto de interrupção **PC04**. Essa notificação poderia ser implementada por meio de configurações de som ou fala, indicando o motivo da inter-

<sup>6</sup>Palavra-chave que define a visibilidade de um atributo, método ou classe.

rupção e a linha específica do código **PC05**. Adicionalmente, seria benéfico disponibilizar a opção de movimentar o cursor do editor para a linha correspondente **PC06**, facilitando ainda mais a identificação e resolução do problema pelo programador.

Além disso, o participante também sugere que o IDE informe ao usuário qual método está sendo executado naquele ponto do código em que o ponto de interrupção está localizado **PC07**, possibilitando que o usuário interaja com a pilha de chamadas de métodos<sup>7</sup> **PC08**.

Em relação à **Diretriz 3.3 (Contexto e nível de escopo)**, foi sugerido que quando o usuário inserisse um parêntese de abertura ou fechamento de instrução em um arquivo, o leitor de tela tocasse para informar o nível de escopo **PC09**. Segundo o participante, isso importante porque:

P3: "(...) o usuário não precisaria ficar contando mentalmente os parênteses na hora de fechar em instruções (...). O bip dos parênteses opcionalmente poderia ser configurável com um tom diferente para não confundir com o bip do recuo de linhas."

O participante expressou discordância parcial quanto à clareza de aplicação das estratégias de navegação e à suficiência dos detalhes contidos na **Diretriz 3.4 (Estratégias de Navegação)**, principalmente especificamente em relação ao **Critério de sucesso 3.4.1** (É possível escolher entre interagir com uma interface gráfica ou uma interface simplificada, com um layout baseado em texto e menus organizados em níveis). Segundo ele, a sugestão de criar uma versão textual da IDE não se mostra tão necessária, considerando que seria um esforço considerável adicional de desenvolvimento para replicar todas as funcionalidades da versão gráfica na versão em modo *console* **PC10**.

Já em relação à **Diretriz 3.7 (Ajuda e Documentação)**, E3 sugeriu que fosse realizada uma revisão no texto do **Critério de Sucesso 3.7.4** (A cada nova versão do IDE, o usuário é informado sobre funcionalidades adicionadas e/ou removidas). De acordo com ele, ao lançar uma versão do IDE, seria importante notificar o usuário a respeito dos recursos de acessibilidade que foram adicionados ou removidos naquela versão **PC11**. Isso se justifica pelo fato de que as opções de acessibilidade podem ser configuradas de forma opcional, e fornecer ao usuário a oportunidade de conhecer e ajustar essas configurações após a atualização da IDE poderia facilitar seu uso posterior.

Outro ponto em que P3 demonstrou concordância parcial está relacionado à **Diretriz 4.1 (Espaçamento)**, especificamente em relação à pertinência e especificidade de suas recomendações para melhorar a acessibilidade para pessoas cegas. O participante afirma que opção de pronunciar todos os espaços utilizados para indicar

<sup>7</sup>Do inglês *call stack*, refere-se a uma lista ordenada utilizada rastrear o fluxo de execução de métodos ou funções em um programa.

identificação em uma linha, de uma única vez, pode ser bastante útil. No entanto, ele ressalta a preocupação de que isso poderia gerar confusão quando o usuário tenta remover um espaço, pois ele pode interpretar erroneamente que isso resultaria na remoção da quantidade total de espaços no início da linha. É importante destacar que essa observação não sugere a eliminação da diretriz em questão, mas sim uma reflexão sobre a forma como ela será implementada no IDE, visando evitar possíveis mal-entendidos por parte dos usuários.

Na **Diretriz 4.2 (Autocompletar)**, o participante propôs que, durante a digitação de uma instrução de código-fonte, a cada letra inserida pelo usuário, o leitor de tela deveria repetir o conteúdo do recurso de autocompletar, sempre que o foco do cursor do teclado não se altere durante a digitação **PC12**.

P3 também discordou parcialmente do texto apresentado na **Diretriz 8.1 (Idioma)**, ressaltando a necessidade de uma análise mais aprofundada do **Critério de Sucesso 8.1.1** (A pronúncia do código-fonte é realizada em língua inglesa). Ele observou que nem todos os programadores escrevem o código-fonte exclusivamente em língua inglesa. Por exemplo, a nomenclatura de variáveis, constantes, métodos e classes, bem como os comentários, podem ser escritos na língua nativa do usuário. O participante ponderou que, embora a pronúncia correta das palavras reservadas seja um ponto a favor da troca de idioma, isso pode prejudicar a leitura de trechos que estejam em língua portuguesa entre as instruções e comandos específicos das linguagens de programação. O participante acredita que seria mais interessante se o IDE permitisse que o usuário determine o que deve ser pronunciado em língua inglesa e o que deve ser pronunciado em outra língua, como a língua da interface em uso **PC13**.

Além disso, o participante aponta que estudantes iniciantes enfrentam um desafio adicional relacionado à pronúncia do código-fonte. Isso ocorre devido à dificuldade que os estudantes cegos têm em memorizar a grafia correta das palavras reservadas, uma vez que a pronúncia muitas vezes difere da escrita. Essa discrepância entre a pronúncia e a escrita pode gerar confusão para o estudante, tornando o processo de aprendizagem mais complexo.

Ainda em relação à **Diretriz 8.1 (Idioma)**, agora em relação ao **Critério de sucesso 8.1.3** (As mensagens de erro podem ser traduzidas para a língua da interface em uso), o participante afirma que:

P3: "(...) *as mensagens de erro serem traduzidas por um sistema de tradução automática também pode ser um problema, pois pode traduzir palavras que não deve e prejudicar mais ainda a compreensão do usuário.* **PC14**"

Por fim, a pergunta sobre a necessidade de explicação adicional de termos técnicos ou jargões revelou uma disparidade nas respostas dos participantes. No entanto, nenhum deles identificou termos técnicos que necessitassem de maiores explicações,

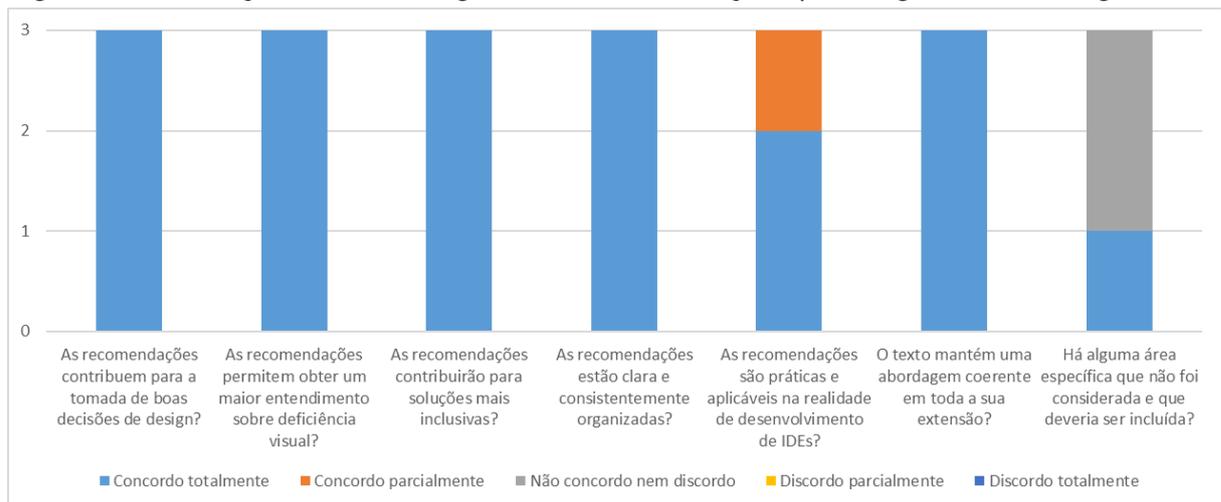
o que pode sugerir uma falta de atenção ou uma compreensão inadequada das perguntas.

Ao contrário de P1 e P2, que concordaram plenamente com todas as afirmações apresentadas para cada Diretriz, as dúvidas expressas por P3 destacam áreas que precisam ser revisadas e aprimoradas. Especificamente, suas discordâncias parciais em relação à clareza de aplicação, pertinência e utilidade de algumas diretrizes indicam pontos que merecem uma revisão cuidadosa para garantir sua eficácia e utilidade para os usuários-alvo.

Os gráficos que mostram as respostas dos participantes para cada uma das afirmações individuais relacionadas às 21 diretrizes podem ser visualizados no Anexo I.

Ao avaliarem a estrutura e organização geral das Diretrizes, os participantes fizeram uma avaliação bastante positiva, como pode ser visualizado na Figura 13.

Figura 13 – Avaliação do contexto geral das recomendações por Programadores Cegos



Especificamente em relação à alternativa que afirmava "As recomendações são práticas e aplicáveis na realidade de desenvolvimento de IDEs?", P3 selecionou a opção "Concordo parcialmente", justificando que embora as recomendações sejam bastante claras e precisas sobre o que precisa ser feito, ele acredita que seria necessário estabelecer diversas parcerias entre programadores de IDEs, empresas responsáveis pelas APIs (*Application Programming Interfaces*) de acessibilidade e criadores de leitores de tela para que fossem implementadas. Segundo ele, apesar de parecerem simples de implementar, na prática, podem demandar uma revisão significativa e melhorias na forma como a indústria aborda a acessibilidade de IDEs para leitores de tela e programas em geral. Ele acredita que essa transformação seria extremamente positiva e que a implementação dessas recomendações representaria um avanço significativo na acessibilidade dos IDEs, mas reconhece que pode demandar um esforço considerável em termos de desenvolvimento.

Como sugestão de melhoria do documento que contém as diretrizes, esse mesmo participante sugeriu que fossem abordadas, adicionalmente, questões relacionadas à acessibilidade das ferramentas de controle de versão **PC15**. Essas abordagens poderiam incluir características como permitir que o programador identifique as alterações feitas em um arquivo modificado no *Git*<sup>8</sup> (sistema de controle de versões distribuído), possibilitando a identificação do arquivo alterado, das linhas modificadas e do responsável pela modificação.

### 7.3 Validação por Analista de Acessibilidade

Esta etapa da avaliação contou com a participação de um profissional que atua como professor de programação em cursos profissionalizantes e como Analista de Acessibilidade na Google do Brasil, desenvolvendo e testando soluções acessíveis. Esse participante é cego e foi recrutado através do Grupo de Discussão "Cegos Programadores", e será identificado no texto como P4.

Assim como os demais participantes, P4 recebeu por e-mail o endereço da página HTML e o arquivo PDF contendo o texto das diretrizes, além do *link* para o questionário de validação. No entanto, ele expressou preferência pela realização de uma entrevista para discutir suas percepções em relação a essas recomendações.

Uma entrevista semiestruturada foi conduzida em dezembro de 2023 por meio de uma chamada de vídeo no *WhatsApp*, com duração aproximada de 30 minutos.

#### 7.3.1 Resultados

No início da entrevista, o participante expressou uma avaliação geral positiva das diretrizes, destacando sua clareza e utilidade. No entanto, fez algumas observações pontuais sobre aspectos do documento que poderiam ser aprimorados, os quais serão abordados ao longo desta seção.

A primeira observação do participante está relacionada à **Diretriz 2.1 (Informações sobre erros)**. Ele sugere a inclusão de um tópico no item "**Como fazer?**" recomendando que o IDE utilize recursos de Inteligência Artificial (IA) para auxiliar o usuário a identificar erros ou oferecer sugestões de melhoria e correção do código **AA01**. Por exemplo:

P4: "(...) quando eu quero resolver um problema de código ele [IDE] pesquisa no ChatGPT<sup>9</sup>, e o chat, e aí ele recebe as informações do chatGPT e me dá uma informação mais clara de um erro de código, por exemplo, ou como resolver um código, por exemplo (...)."

---

<sup>8</sup><https://git-scm.com/>

<sup>9</sup><https://chat.openai.com/>

Outro apontamento feito pelo participante diz respeito à **Diretriz 3.6 - Atalhos**. Ele enfatiza que, em sua experiência profissional já utilizou diversos IDEs para programar. Segundo ele, em muitas deles, os atalhos para as mesmas funcionalidades variam, o que dificulta a memorização. **AA02**. Ele destaca a necessidade de:

P4: "(...) padronização de atalho de teclado, (...) todos os atalhos tem que ser os mesmos para todas as IDEs neh."

Além disso, o participante ressaltou a importância de fornecer controle para que o usuário possa ativar ou desativar os recursos de acessibilidade oferecidos pelo IDE, enfatizando que, embora os recursos de acessibilidade precisem estar disponíveis em um IDE, não significa necessariamente que o usuário irá utilizá-las o tempo todo. Essa preferência pode variar de um usuário para outro.

P4 sugere considerar a possibilidade de incluir como diretriz a opção de ativar ou desativar os recursos de acessibilidade conforme as necessidades e preferências de cada usuário **AA03**. Isso permitiria que cada usuário personalizasse a configuração da IDE de acordo com suas necessidades específicas, adaptando-a conforme suas preferências e mudanças nas exigências. Conforme mencionado pelo participante, as ferramentas devem estar prontamente disponíveis para serem ativadas quando o usuário necessitar, mas também devem oferecer a opção de serem desativadas, caso o usuário opte por não utilizá-las.

Por fim, o participante destacou a importância de o IDE reconhecer as configurações e preferências do usuário durante o uso, possibilitando a ativação ou desativação dos recursos de acessibilidade de acordo com esse perfil **AA04**. A funcionalidade se mostra especialmente útil em ambientes onde múltiplas pessoas compartilham o mesmo computador, permitindo uma experiência mais personalizada e adaptada às necessidades individuais de cada usuário. Isso é particularmente relevante para estudantes de computação, que frequentemente compartilham computadores em laboratórios de informática durante as aulas de programação.

Este participante também enfatizou a importância de incorporar diretrizes que detalhem características de acessibilidade essenciais para tornar as ferramentas de controle de versionamento de código (como *Git*) mais acessíveis aos programadores cegos **AA05**. Essa preocupação já foi mencionada anteriormente por P3, reforçando a importância de considerar as necessidades específicas dos programadores cegos ao colaborar com equipes de desenvolvimento.

Quanto à estrutura e organização geral das Diretrizes, o participante afirmou que o texto é claro e compreensível. Além disso, ele destacou que as Diretrizes estão organizadas em um formato que seria de grande auxílio para aqueles que pretendem implementá-las na prática ou verificar, por meio de inspeção, se um determinado IDE atende às suas recomendações.

## 7.4 Validação por Desenvolvedor de IDE

Uma das formas de validação das diretrizes planejadas para a segunda iteração previa a participação de pessoas envolvidas no desenvolvimento de IDEs. Foram realizadas diversas tentativas para identificar e contatar esses especialistas. No entanto, o processo de recrutamento de participantes com esse perfil foi bastante desafiador.

Foram enviados convites de participação por e-mail, listas de discussão, páginas do Instagram e comunidades do Facebook, além do preenchimento de formulários de contato para várias empresas envolvidas no desenvolvimento dos seguintes IDEs: *Android Studio*<sup>10</sup>, *ATOM*<sup>11</sup>, *Brackets*<sup>12</sup>, *Bluefish*<sup>13</sup>, *BlueJ*<sup>14</sup>, *CodeAnywhere*<sup>15</sup>, *Codeshare.io*<sup>16</sup>, *CodeSpace*<sup>17</sup>, *coffeeCup*<sup>18</sup>, *DevC++*<sup>19</sup>, *Eclipse*<sup>20</sup>, *IntelliJ*<sup>21</sup>, *NetBeans*<sup>22</sup>, *Sublime*<sup>23</sup>, *Scratch*<sup>24</sup>, *TextMate*<sup>25</sup>, *UltraEdit*<sup>26</sup>, *Vim*<sup>27</sup>, *VisualStudio*<sup>28</sup>, *VSCode*<sup>29</sup>, *XCode*<sup>30</sup>, *Portugol Studio*<sup>31</sup> e *VisuAlg*<sup>32</sup>.

Apesar da diversidade de tentativas de recrutamento desses profissionais, foram recebidas apenas três respostas, provenientes das empresas *Oracle* (responsável pelo desenvolvimento do *NetBeans*), *Scratch* e *Portugol Studio*. A *Oracle* informou apenas que não possui uma equipe de desenvolvedores localizada no Brasil e não indicou outros profissionais que pudessem contribuir na pesquisa. A *Scratch* comprometeu-se a encaminhar o convite para suas equipes de desenvolvimento, contudo, não obtivemos manifestação de interesse por parte desses profissionais. No caso do *Portugol Studio*, obtivemos a resposta positiva de um dos professores envolvidos no seu desenvolvimento, que prontamente respondeu ao questionário de avaliação.

<sup>10</sup><https://developer.android.com/studio?hl=pt-br>

<sup>11</sup><https://www.accesstomemory.org/pt-br/community/support/>

<sup>12</sup><https://desenvolvimentoaberto.org/about/contato/>

<sup>13</sup><https://sourceforge.net/projects/bluefish/lists/bluefish-community>

<sup>14</sup>[bluej-support@bluej.org](mailto:bluej-support@bluej.org)

<sup>15</sup><https://codeanywhere.com/contact>

<sup>16</sup><https://codeshare.io/>

<sup>17</sup><https://github.com/features/discussions>

<sup>18</sup>[helpme@coffeecup.com](mailto:helpme@coffeecup.com)

<sup>19</sup>[webmaster@bloodshed.net](mailto:webmaster@bloodshed.net)

<sup>20</sup>[brazil-ec@eclipse.org](mailto:brazil-ec@eclipse.org)

<sup>21</sup>[jbcc@jetbrains.com](mailto:jbcc@jetbrains.com)

<sup>22</sup>[dev@netbeans.incubator.apache.org](mailto:dev@netbeans.incubator.apache.org)

<sup>23</sup><https://forum.sublimetext.com/c/ideas-and-feature-requests>

<sup>24</sup><https://scratch.mit.edu/contact-us/>

<sup>25</sup><https://macromates.com/support>

<sup>26</sup>[renewals@ultraedit.com](mailto:renewals@ultraedit.com)

<sup>27</sup>[vim-dev@vim.org](mailto:vim-dev@vim.org)

<sup>28</sup><https://visualstudio.microsoft.com/pt-br/>

<sup>29</sup><https://code.visualstudio.com/>

<sup>30</sup><https://developer.apple.com/xcode/>

<sup>31</sup><https://univali-lite.github.io/Portugol-Studio/>

<sup>32</sup><https://visualg3.com.br/>

Esta última etapa de validação das Diretrizes envolveu, portanto, a participação de um professor de Computação que também atua no desenvolvimento do IDE Portugol Studio (P5). O Portugol Studio é uma ferramenta desenvolvida pela Universidade do Vale do Itajaí (Univali), em Santa Catarina (SC), com o propósito de facilitar o ensino de programação. Possui uma sintaxe amigável, fundamentada nas linguagens C e PHP, e é direcionado ao ensino de programação para falantes da língua portuguesa.

#### 7.4.1 Resultados

Esta seção aborda as contribuições fornecidas pelo participante P5 durante a avaliação das diretrizes, que trouxe à tona questões importantes, destacando áreas específicas que requerem revisão e aprimoramento.

Em geral, as diretrizes receberam avaliações positivas, indicando um alto nível de concordância do participante com as afirmações apresentadas. No entanto, ficou claro que algumas Diretrizes requerem revisão de certos critérios específicos, abordados a seguir.

Em relação à **Diretriz 1.1 (Semântica)**, P5 sugeriu que o texto do **Critério de sucesso 1.1.1** (É possível navegar pelo explorador de pacotes utilizando as teclas de seta) fosse revisado. De acordo com ele, não está claro por que a compreensão do código deve estar vinculada a um "explorador de pacotes", já que nem todos os IDEs são organizados dessa forma **DS01**. O PortugolStudio, os IDEs de Python e os IDEs online de C e JavaScript, por exemplo, não utilizam pacotes. Geralmente, são IDEs como esses que são empregados no ensino para estudantes iniciantes em programação.

Além disso, o participante expressou preocupação com falta de clareza do termo "informações contextuais" do **Critério de sucesso 1.1.4** (É possível obter informações contextuais de cada função ou método de um arquivo de código-fonte). Ele justifica que não são especificadas as ações necessárias para auxiliar uma pessoa cega a obter as informações contextuais de cada função ou método em um arquivo de código-fonte. O participante sugeriu a substituição do termo "informações contextuais" por outro que proporcione maior clareza ao Critério de Sucesso 1.1.4 **DS02**.

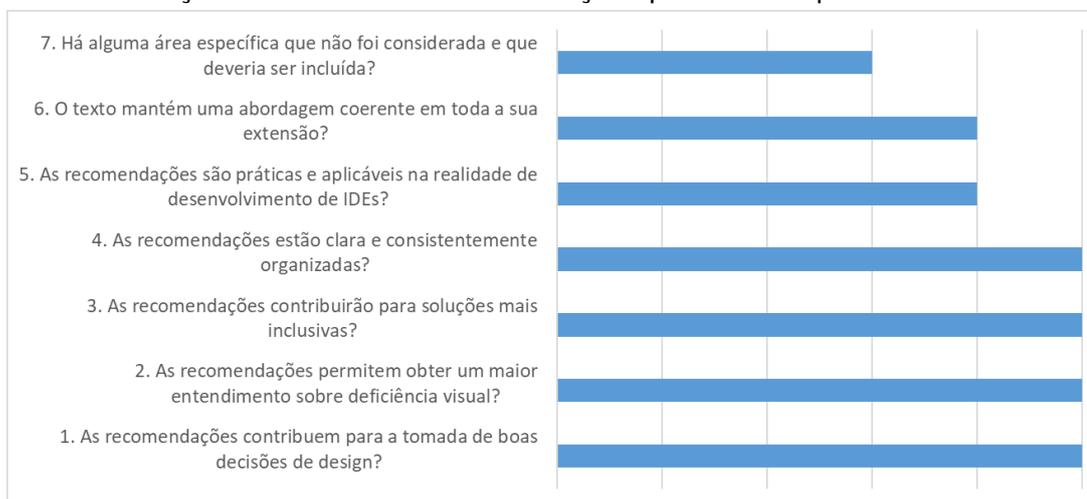
A análise do participante também abordou questões relacionadas à **Diretriz 2.1 (Informações sobre erros)**, mais precisamente ao **Critério de sucesso 2.1.4** (As mensagens de erro contêm informações a respeito da causa do erro, sua localização no código-fonte e como corrigi-lo). P5 destacou a necessidade de uma distinção clara entre o IDE e o Compilador, especialmente no que diz respeito às mensagens de erro **DS03**. Ele ressalta que as mensagens de erros são geradas pelo compilador que, em geral, não proporciona muita flexibilidade para aprimorar a qualidade das mensagens. A melhoria nesse aspecto só é possível quando o desenvolvedor da IDE também é responsável pela criação do compilador, como ocorre no caso do Portugol Studio.

O participante também fez uma observação relacionada ao **Critério de sucesso 4.2.3** (O usuário consegue solicitar explicitamente a leitura do conteúdo exibido no recurso autocompletar) da **Diretriz 4.2 (Autocompletar)**, enfatizando a importância do controle do usuário sobre esse recurso, podendo ativá-lo ou desativá-lo conforme sua preferência **DS04**. Além disso, ele destaca a importância de o usuário ser informado pelo leitor de tela sobre o que foi inserido automaticamente no código pelo recurso autocompletar **DS05**.

Em resumo, os resultados refletem uma visão geral positiva das Diretrizes, identificando áreas específicas que podem requerer atenção adicional durante a revisão para garantir que atendam efetivamente às necessidades dos usuários-alvo. As respostas individuais atribuídas pelo participante para cada Diretriz podem ser consultadas no Anexo J.

Quando questionado sobre a estrutura e organização das diretrizes, as respostas do participante indicam que as recomendações foram consideradas claras, consistentemente organizadas e aplicáveis na realidade do desenvolvimento de IDEs. Segundo o participante, alguns pontos específicos do texto precisam de alguns ajustes, mas em geral mantém uma abordagem coerente. Uma síntese dessa avaliação pode ser visualizada na Figura 14.

Figura 14 – Avaliação Individual das Recomendações preliminares por Desenvolvedor de IDE



O participante ressaltou a relevância e importância do trabalho realizado, destacando que as recomendações estão bem elaboradas e cuidadosamente desenvolvidas, sendo derivadas de experiências e dificuldades reais enfrentadas por usuários cegos.

## 7.5 Refinamento das Diretrizes

Esta seção apresenta os resultados obtidos nas duas etapas de validação das diretrizes (Capítulos 6 e 7) e descreve como esses resultados moldaram a adaptação do texto da 2ª versão Diretrizes de Acessibilidade em IDEs para Estudantes com Deficiência Visual, detalhada no Capítulo 8.

Os resultados alcançados na primeira e segunda etapas de validação foram empregados para identificar áreas que necessitavam de aprimoramento e ajustes, visando aprimorar sua eficácia e aplicabilidade. O objetivo principal desta etapa foi garantir uma melhor adequação do documento às necessidades e experiências dos estudantes cegos ao interagir com IDEs.

Para facilitar a análise das sugestões obtidas dos diferentes perfis de participantes, elas foram agrupadas por categorias e diretrizes, proporcionando uma visão mais estruturada das áreas de melhoria identificadas. A análise permitiu verificar que algumas das sugestões recebidas pelos participantes já estavam contempladas no texto das Diretrizes. Nestes casos, os tópicos relevantes foram revisados para proporcionar uma redação mais clara e precisa do texto. Além disso, algumas sugestões não foram implementadas e, nesses casos, a justificativa para explicar por que a sugestão não foi adotada é fornecida.

É importante ressaltar que os resultados apresentados serão identificados de acordo com as convenções adotadas ao longo desta Tese para representar cada grupo de participantes do estudo: estudantes de computação **EC**; programadores cegos **PC**; analista de acessibilidade **AA**; e, professor que atua no desenvolvimento de IDE **DS**.

As modificações realizadas (nos casos de aprovação) ou a justificativa (nos casos de rejeição) para cada sugestão de alteração serão apresentadas a seguir. Ressalta-se que as Diretrizes 3.1, 3.2, 3.5, 4.1, 6.2 e 8.2 não receberam sugestões de alteração e, portanto, não serão discutidas nesta seção.

### Diretriz 1.1 (Semântica):

**EC01** É necessário esclarecer a terminologia "alertas sonoros".

- Adição da seção "Glossário" no final do Documento, inserindo um texto que descreve o termo "Alertas Sonoros"; e,
- Revisão do texto do item "Como fazer?", visando aprimorar a escrita das sugestões de implementação da Diretriz.

**PC01** Alterar o termo "Finalidade" no Critério de sucesso 1.1.4 (É possível obter informações contextuais de cada função ou método de um arquivo de código-fonte).

- O termo "Finalidade" foi substituído pela palavra "Descrição".

**DS01** Revisar o termo "explorador de pacotes", já que nem todos os IDEs são organizados dessa forma.

- O termo "explorador de pacotes" foi substituído por "gerenciador de arquivos".

**DS02** Substituir o termo "informações contextuais" por outro que proporcione maior clareza ao Critério de Sucesso 1.1.4.

- O termo "informações contextuais" foi substituído por "informações detalhadas".

### **Diretriz 2.1 (Informações sobre erros):**

**PC02** Informar o usuário sobre o sucesso ou falha de um teste unitário em um programa.

- Adição de Critério de sucesso contendo o texto: "O usuário é informado sobre o sucesso ou falha na execução de testes unitários".

**AA01** Sugerir a utilização Inteligência Artificial (IA) para auxiliar na correção de um erro no código.

- Inclusão de sugestão no item "Como fazer?" contendo o texto: "Utilizar técnicas de Inteligência Artificial (IA) para ajudar o usuário na identificação de erros, bem como para oferecer sugestões de aprimoramento e correção do código".

**DS03** Rever a aplicabilidade do Critério de sucesso 2.1.4 (As mensagens de erro contêm informações a respeito da causa do erro, sua localização no código-fonte e como corrigi-lo), já que as mensagens de erros são geradas pelo compilador e não pelo IDE.

- Exclusão do Critério de Sucesso 2.1.4 e de todo o texto que fizesse menção ao conteúdo das mensagens de erro.
- Exclusão do Critério de sucesso 8.2.2 (As mensagens de erro são claras e facilmente compreendidas) da Diretriz 8.2.

### **Diretriz 2.2 (Variáveis e constantes):**

**EC02** Possibilitar a modificação do nome, valor ou tipo das variáveis declaradas no código-fonte por meio de uma lista fornecida pelo sistema. Ao efetuar a alteração na lista, o IDE realizaria automaticamente os ajustes em todo o código.

- Adição de Critério de sucesso contendo o texto: "É possível obter informações como nome, valor, tipo, modificador de acesso e escopo de variáveis e constantes declaradas em uma base de código".
- Adição de Critério de sucesso contendo o texto: "É possível alterar o nome, o valor, o tipo ou o modificador de acesso de uma variável uma única vez e solicitar que o sistema atualize automaticamente todas as ocorrências dessa variável no código-fonte".

**PC03** Notificar o usuário sobre o tipo de modificador de acesso definido para cada atributo e/ou variável.

- Essa funcionalidade já estava prevista no Critério de sucesso 2.2.1 (É possível obter informações como nome, valor, tipo, qualificador e escopo de variáveis e constantes declaradas em uma base de código). Para tornar mais claro o texto deste Critério de sucesso, o termo "qualificador" foi substituído por "modificador de acesso";
- Da mesma forma, na Diretriz 1.1, o termo "qualificador" foi substituído por "modificador de acesso"; e,
- O significado do termo "modificador de acesso" foi descrito na seção "Glossário", juntamente com os possíveis tipos de modificadores de acesso: *public*, *private* e *protected*.

### **Diretriz 2.3 (Marcadores e Pontos de interrupção):**

**PC04** Notificar o usuário quando o programa parar de executar devido à presença de um ponto de interrupção.

- Adição de Critério de sucesso contendo o texto: "O sistema notifica o usuário quando a execução for pausada devido à existência de um ponto de interrupção no código".

**PC05** Informar o motivo da presença do ponto de interrupção e a linha específica do código onde está localizado.

- Este item não implicou em alterações no texto desta Diretriz porque a sugestão já estava contemplada nos tópicos do item "Como fazer?".

**PC06** Permitir que o cursor do teclado seja movido para a linha correspondente ao ponto de interrupção.

- Alteração no segundo tópico do item "Como fazer?": "Disponibilizar atalhos de teclado para listar os pontos de interrupção em um arquivo de código-fonte. Ao selecionar um ponto de interrupção, o usuário pode acessar informações, como a localização no código, escopo e descrição. Além disso, o IDE deve permitir o deslocamento do cursor do teclado para a linha de código correspondente ao ponto de interrupção."

**PC07** Ao identificar um ponto de interrupção, informar O usuário qual método está sendo executado no trecho de código correspondente.

- Este item não implicou em alterações no texto desta Diretriz porque a sugestão já estava contemplada nos tópicos do item "Como fazer?".

**PC08** Possibilitar a interação do usuário com a pilha de chamadas de métodos.

- Está sugestão não resultou em alterações no conteúdo da Diretriz porque se constatou que a maioria dos IDEs não oferece um acesso direto à pilha de chamadas de métodos durante a execução do programa. Assim, não se trata apenas de garantir a acessibilidade a essa funcionalidade, uma vez que ela simplesmente não está disponível. Além disso, há um entendimento de que essa funcionalidade é especialmente útil para usuários com um nível mais avançado de habilidades em programação, enquanto sua utilidade para estudantes pode ser restrita.

### **Diretriz 3.3 (Contexto e nível de escopo):**

**PC09** Utilizar alertas sonoros para informar o nível de escopo quando o usuário insere um parêntese de abertura ou fechamento de instrução em um arquivo.

- Adição de Critério de sucesso contendo o texto: "O sistema oferece informações sobre a instrução correspondente sempre que o usuário insere um parêntese, chave ou colchete de fechamento no código-fonte".

### **Diretriz 3.4 (Estratégias de navegação):**

**PC10** Rever a pertinência do Critério de sucesso 3.4.1 (É possível escolher entre interagir com uma interface gráfica ou uma interface simplificada, com um *layout* baseado em texto e menus organizados em níveis).

- O Critério de Sucesso 3.4.1 foi mantido, pois, apesar da potencial dificuldade de implementação, possibilitar que estudantes cegos escolham a forma de interação desejada - seja por meio de uma interface gráfica ou simplificada - pode contribuir para uma experiência de uso mais eficiente e agradável.

**Diretriz 3.6 (Atalhos):**

**AA02** Ressaltar a necessidade de padronização dos atalhos de teclado para funcionalidades iguais em diferentes IDEs.

- Adição de Critério de sucesso contendo o texto: "Os atalhos para as funcionalidades do IDE seguem as convenções e padrões adotados por outros IDEs, softwares e Sistemas Operacionais".

**Diretriz 3.7 (Ajuda e Documentação):**

**EC03** **PC11** No momento da instalação de novas versões do IDE, informar o usuário sobre o que foi atualizado, incluindo os recursos adicionados ou removidos, e como essas atualizações interferem na acessibilidade do sistema.

- Adição de Critério de Sucesso contendo o texto: "Ao instalar uma nova versão do IDE, o usuário é informado sobre funcionalidades adicionadas e/ou removidas".
- Adição de Critério de Sucesso contendo o texto: "Ao instalar uma nova versão do IDE, o usuário é informado sobre os recursos de acessibilidade adicionados e/ou removidos".

**EC04** No momento da instalação das novas versões do IDE, permitir que o usuário escolha quais elementos gostaria de atualizar e quais recursos e configurações de acessibilidade da versão anterior ele gostaria de manter.

- Adição de Critério de Sucesso contendo o texto: "Ao instalar uma nova versão do IDE, o usuário pode optar por manter ou alterar as configurações de acessibilidade utilizadas na versão anterior".

**Diretriz 4.2 (Autocompletar):**

**GF01** Notificar quando o caractere de término de uma instrução ou bloco de código, expressão condicional ou lista de parâmetros de funções (tais como chaves, colchetes e parênteses) forem inseridos automaticamente, após o usuário inserir o caractere de início.

- Adição de Critério de Sucesso contendo o texto: "O usuário é informado quando o IDE adiciona automaticamente um caractere de fechamento de instrução ou bloco de código.

**GF02** Notificar a existência do recurso autocompletar e solicitar uma ação explícita para que seja realizada a leitura de seu conteúdo.

- Este tópico já estava contemplado no Critério de sucesso 4.2.3 (O usuário consegue solicitar explicitamente a leitura do conteúdo exibido no recurso autocompletar) e, portanto, não implicou em alterações no texto da Diretriz.

**GF03** Destacar o tipo de informação que estiver sendo lida pelo leitor de telas. Por exemplo: comentário, conteúdo do recurso autocompletar, dentre outros.

- Esta sugestão não acarretou alterações no texto da Diretriz, visto que os Critérios de Sucesso 4.2.1 e 4.2.3 já especificam que o sistema deve informar sobre a existência do recurso autocompletar e solicitar uma ação explícita do usuário para que a leitura do conteúdo seja realizada.

**PC12** Adicionar um Critério de Sucesso que especifique que o recurso de autocompletar deve apresentar sugestões com base na entrada do usuário sempre que o foco do teclado permanecer na instrução que estiver sendo inserida.

- Adição de Critério de sucesso contendo o texto: "O recurso de autocompletar exibe sugestões com base na entrada do usuário, mantendo o foco do leitor de tela no conteúdo disponibilizado por este recurso durante a digitação".

**DS04** Permitir que o recurso autocompletar seja ativado ou desativado, conforme as preferências do usuário.

- Adição de Critério de sucesso contendo o texto: "O usuário pode ativar ou desativar o recurso autocompletar a qualquer momento".

**DS05** Informar ao usuário sobre instruções forem inseridas automaticamente no código pelo recurso de autocompletar.

- Alteração do Critério de Sucesso que continha o texto "O usuário é notificado quando o IDE adiciona automaticamente um caractere de fechamento de instrução ou bloco de código", passando a ser formulado da seguinte maneira: "O usuário é notificado quando o IDE adiciona um caractere de fechamento de instrução ou bloco de código, ou quando insere uma instrução sugerida pelo recurso autocompletar".
- Adição de tópico ao item "Como fazer?" contendo o texto: "3. Utilizar alertas sonoros para notificar usuário quando o IDE adicionar automaticamente um caractere de fechamento de instrução ou bloco de código (por exemplo: chaves, parênteses ou colchetes) após o usuário digitar o caractere de início correspondente".

**Diretriz 4.3 (Comentários):**

**GF03** Destacar o tipo de informação que estiver sendo lida pelo leitor de telas. Por exemplo: comentário, conteúdo do recurso autocompletar, dentre outros.

- Esta sugestão não acarretou alterações no texto da Diretriz, visto que os Critérios de Sucesso 4.3.1 e 4.3.2 já especificam que o sistema deve informar sobre a existência de um comentário na posição de foco do cursor do teclado e solicitar uma ação explícita do usuário para que a leitura de seu conteúdo seja realizada.

**Diretriz 5.1 (Dobra de Código):**

**EC05** Alterar o texto do item "Descrição" para contemplar todos os tópicos abordados pela Diretriz.

- O texto do item "Descrição" alterado para: "Permitir ao usuário ocultar ou expandir seções de código, bem como identificar trechos de código ocultos".

**Diretriz 5.2 (Visão geral do código):**

**EC06** Alterar o texto dos Critérios de sucesso 5.2.2 e 5.2.4 para contemplar todos os tipos de arquivo que podem compor uma base de código.

- O texto do Critério de sucesso 5.2.2 foi alterado para: "É possível obter uma lista de todos os arquivos que compõem um projeto de software".
- O texto do Critério de sucesso 5.2.4 foi alterado para: "É possível navegar entre os arquivos que compõem um projeto de software".

**Diretriz 6.1:**

**GF04** Oferecer recursos que auxiliem o usuário na elaboração de código-fonte para a construção de interfaces gráficas.

- Esta alteração já foi realizada na 1ª iteração do Estágio de Refinamento, descrita no Capítulo 6. Portanto, já está contida no texto da 1ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos.

**Diretriz 7.1 (Alertas sonoros):**

**EC07** Informar ao leitor sobre a existência de uma Diretriz específica que trata da utilização de "Alertas Sonoros".

- Inserção do tópico "IMPORTANTE" em todas as Diretrizes que sugerem a utilização de "Alertas Sonoros", informando que a referida Diretriz deve ser consistente com as recomendações apresentadas na Diretriz 7.1 (Alertas sonoros). A alteração foi realizada nas Diretrizes 2.1, 2.3, 3.3, 3.4, 4.2, 4.3, 5.1 e 8.2.

### Diretriz 8.1 (Idioma)

**PC13** Permitir que o usuário configure como quer que o código-fonte seja pronunciado, especificando o que deve ser pronunciado em língua inglesa e o que deve ser pronunciado na língua da interface em uso.

- Divisão do Critério de sucesso 8.1.1 (A pronúncia do código-fonte é realizada em língua inglesa) em dois Critérios de sucesso: Critério de sucesso 8.1.1 (A pronúncia do código-fonte é realizada em língua inglesa) e Critério de sucesso 8.1.2 (A pronúncia dos rótulos e demais componentes da interface é realizada na língua da interface em uso).
- Alteração da numeração do Critério de sucesso 8.1.2 (A pronúncia dos rótulos e demais componentes da interface é realizada na língua da interface em uso), passando a ser 8.1.3.
- Alteração da numeração do Critério de sucesso 8.1.3 (As mensagens de erro podem ser traduzidas para a língua da interface em uso), passando a ser 8.1.4.
- Alteração do texto do Critério de sucesso 8.1.3, passando a ser formulado da seguinte maneira: "O usuário pode configurar a língua que será utilizada para pronúncia da nomenclatura atribuída a variáveis, classes e métodos/funções, bem como dos rótulos e demais componentes da interface."

**PC14** Rever Critério de sucesso 8.1.3 (agora 8.1.4), pois a tradução automática das mensagens de erro deve ser implementada de maneira que não prejudique a compreensão do usuário.

- O Critério de sucesso 8.1.3 (As mensagens de erro podem ser traduzidas para a língua da interface em uso) não foi alterado, mas o texto item "Como fazer?" passou por ajuste para adequar a estas sugestões.
- O segundo tópico do item "Como fazer?" teve seu texto alterado para: "Fornecer atalhos que possibilitem a tradução das mensagens de erro para o idioma da interface em uso, abrangendo apenas o conteúdo textual dessas mensagens e estando em conformidade com as configurações de pronúncia especificadas pelo usuário".

Além das sugestões de correção e adaptação no conteúdo individual de cada Diretriz, que foram apresentadas até o momento, alguns participantes propuseram a inclusão de novos tópicos. Esses tópicos abordam temas que não foram previstos inicialmente pelos pesquisadores:

**PC15 AA05** Incorporar ao documento novas Diretrizes detalhando características de acessibilidade essenciais para tornar as ferramentas de controle de versionamento de código mais acessíveis aos estudantes cegos.

- Dada a natureza desta Tese de Doutorado, centrada na formulação de Diretrizes voltadas para o uso de IDEs por estudantes, as ferramentas de controle de versionamento não serão tratadas neste conjunto de orientações. Essa escolha é fundamentada na complexidade desses sistemas, assim como em outras funcionalidades e recursos mais avançados oferecidos pelos IDEs, os quais requerem uma investigação mais detalhada. O principal foco desta tese é garantir a acessibilidade das tarefas básicas fundamentais para o aprendizado de programação.

**AA03** Incluir uma diretriz que determine que o usuário deve ser capaz de ativar ou desativar qualquer recurso de acessibilidade do IDE, conforme suas necessidade.

- Adição da seção "Outras Recomendações", após o texto das Diretrizes, contendo uma recomendação para que o sistema permita que o usuário ative ou desative qualquer recurso de acessibilidade oferecido pelo IDE, a qualquer momento.

**AA04** Automatizar a ativação ou desativação dos recursos de acessibilidade no IDE, de acordo com as preferências e configurações individuais de cada usuário.

- Complementação da seção "Outras Recomendações", após o texto das Diretrizes, para conter um texto que especifique que o sistema deve ativar ou desativar automaticamente os recursos de acessibilidade, a partir das informações salvas no perfil do usuário que estiver logado no momento.

Após a implementação das modificações destacadas, verificou-se a necessidade de revisar todo o texto do documento, visando assegurar maior coesão e a consistência do texto, além de torná-lo mais compreensível e eficaz na comunicação das ideias propostas. Cada diretriz foi examinada para assegurar que as alterações realizadas estivessem alinhadas com o conteúdo geral, contribuindo para sua qualidade e clareza. Nesse processo, foram identificadas algumas alterações adicionais seriam necessárias.

A primeira necessidade de alteração estava relacionada ao fato de que algumas diretrizes estavam intrinsecamente ligadas às recomendações presentes em outras diretrizes ou abordavam tópicos especificados por elas. Essas alterações incluíram:

- Adição ou alteração do tópico "IMPORTANTE" em todas as Diretrizes que fazem menção à utilização de "Atalhos", informando que as orientações presentes na referida Diretriz precisariam ser consistentes com as recomendações presentes na Diretriz 3.6 (Atalhos). Este tópico foi adicionado às Diretrizes 1.1, 2.1, 2.2, 2.3, 3.2, 3.3, 3.4, 3.5, 3.7, 4.2, 4.3, 5.1, 5.2, 8.1 e 8.2.
- Alteração do tópico "IMPORTANTE" na Diretriz 1.1 (Semântica) informando que as orientações apresentadas também deveriam ser consistentes com as recomendações presentes na Diretriz 5.2 (Visão Geral de Código).
- Alteração do tópico "IMPORTANTE" na Diretriz 3.2 (Movimentação pelo Código) informando que as orientações apresentadas também deveriam ser consistentes com as recomendações presentes na Diretriz 1.1 (Semântica).
- Alteração do tópico "IMPORTANTE" na Diretriz 3.4 (Estratégias de Navegação) informando que as orientações apresentadas também deveriam ser consistentes com as recomendações presentes na Diretriz 3.1 (Rótulos). Esta mudança ocorreu pós se verificar que a necessidade de exclusão do Critério de sucesso 3.4.5 (Todos os recursos e funcionalidades do IDE podem ser lidos pelos leitores de tela), visto que Critério de sucesso em questão já estava sendo abordado pela Diretriz 3.1 (Rótulos).
- Alteração do tópico "IMPORTANTE" na Diretriz 5.2 (Visão geral de código) informando que as orientações apresentadas também deveriam ser consistentes com as recomendações presentes na Diretriz 1.1 (Semântica).
- Alteração do tópico "IMPORTANTE" na Diretriz 8.1 (Idioma) informando que as orientações apresentadas também deveriam ser consistentes com as recomendações presentes na Diretriz 2.1 (Informações sobre erros).
- Alteração do título da Diretriz 2.3 (Marcadores e Pontos de interrupção) para Diretriz 2.3 (Pontos de interrupção). A mudança ocorreu para evitar problemas de compreensão do usuário, visto que a Diretriz 3.2 (Movimentação pelo código) já faz referência ao termo "marcador", mas com outra finalidade.

Por fim, uma revisão completa do documento foi realizada, eliminando ambiguidades e fazendo ajustes gramaticais essenciais para melhorar sua clareza e coesão.

## 7.6 Considerações Parciais

A variedade de perspectivas trazida pelo grupo de estudantes de computação (Capítulo 6) e pelos especialistas que participaram dos ciclos de validação das diretrizes foi essencial para adequá-las para que abordassem as necessidades dos estudantes cegos no contexto de aprendizagem de programação de computadores e utilização dos recursos básicos oferecidos pelos IDEs.

Foram identificados diversos pontos que exigiram ajustes e adequações para melhor atender às demandas desses estudantes. Tais ajustes foram realizados levando em consideração não apenas a clareza e a precisão das recomendações, mas também a sua efetividade prática e sua capacidade de promover uma experiência de programação mais inclusiva para os estudantes cegos.

As recomendações contidas na 2ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos são apresentadas no Capítulo 8. O documento completo pode ser acessado em <https://elianazen.github.io/DiretrizesAcessibilidadeIDE/2VersaoDiretrizes.html>.

## 8 DIRETRIZES DE ACESSIBILIDADE EM IDES PARA ESTUDANTES CEGOS

Este Capítulo introduz a 2ª versão das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos<sup>1</sup>, resultante dos ajustes e modificações detalhados no Capítulo 7. A Tabela 8 apresenta as 21 diretrizes distribuídas nas oito categorias previamente estabelecidas no Capítulo 5.

O documento que inicia com uma introdução que contextualiza a motivação por trás da concepção das recomendações e destaca que as diretrizes são destinadas a IDEs que oferecem suporte à programação baseada em texto, com foco particular na interação de usuários cegos que dependem de leitores de tela como recurso de TA. Também são detalhadas a estrutura e a organização do documento, visando tornar sua compreensão mais acessível ao leitor.

Tabela 8 – Resumo de categorias e diretrizes

<b>Categoria</b>	<b>Diretrizes</b>
1. Compreensão do Código	Diretriz 1.1 Semântica
2. Depuração de Código	Diretriz 2.1 Informações sobre erros
	Diretriz 2.2 Variáveis e constantes
	Diretriz 2.3 Pontos de interrupção
3. Navegação no Código	Diretriz 3.1 Rótulos
	Diretriz 3.2 Movimentação pelo código
	Diretriz 3.3 Contexto e nível de escopo
	Diretriz 3.4 Estratégias de navegação
	Diretriz 3.5 Numeração de linhas
	Diretriz 3.6 Atalhos
	Diretriz 3.7 Ajuda e Documentação

*Continua na próxima página...*

<sup>1</sup><https://elianazen.github.io/DiretrizesAcessibilidadeIDE/2VersaoDiretrizes.html>

4. Edição de Código	Diretriz 4.1 Espaçamento
	Diretriz 4.2 Autocompletar
	Diretriz 4.3 Comentários
5. <i>Skimming</i> de Código	Diretriz 5.1 Dobra de Código
	Diretriz 5.2 Visão geral do código
6. Interface Gráfica de Usuário	Diretriz 6.1 Construção de Interface Gráfica
	Diretriz 6.2 Validação da interface gráfica
7. Sobrecarga Auditiva	Diretriz 7.1 Alertas sonoros
8. Leitura de Código-fonte	Diretriz 8.1 Idioma
	Diretriz 8.2 Leitura contextual

Nas seções a seguir, são apresentadas as diretrizes, abordando sua descrição, os critérios de sucesso especificados para atendê-las, a motivação por trás de sua elaboração, sugestões de implementação e outras informações relevantes.

## 8.1 CATEGORIA 1: Compreensão do Código

A categoria **Compreensão do Código** contém uma única diretriz (Semântica) que oferece recomendações relacionadas aos desafios enfrentados por estudantes cegos ao tentarem entender a estrutura e a lógica adotada para a implementação de um projeto de software.

Tabela 9 – Diretriz 1.1 Semântica

### Diretriz 1.1 Semântica

#### Descrição

Comunicar o significado lógico dos elementos que constituem uma base de código.

#### Crterios de Sucesso

Crterio de sucesso 1.1.1: É possvel navegar pelo gerenciador de arquivos de um projeto de software utilizando as teclas de seta.

Crterio de sucesso 1.1.2: A ordem de navegao pelo gerenciador de arquivos reflete a estrutura lgica do projeto de software.

Crterio de sucesso 1.1.3: Ao navegar pelo gerenciador de arquivos de um projeto de software, so fornecidas informaes sobre o significado lgico de cada no dentro da hierarquia, bem como o relacionamento desse no com seus irmos ou pais.

Crterio de sucesso 1.1.4: É possvel obter informaes detalhadas de cada funo/mtodo em um arquivo de cdigo-fonte:

- **Descrio:** detalha a funcionalidade implementada pela funo/mtodo;

*Continua na prxima pgina...*

- Modificador de acesso: quando aplicável, determina o tipo de visibilidade da função/método;
- Tipo de retorno: informa o tipo de dado retornado após a execução da função/método; e,
- Parâmetros: atributos de entrada que podem ser recebidos por uma função/método.

Critério de sucesso 1.1.5: É possível mover o cursor do teclado para o nó principal do gerenciador de arquivos de um projeto de software, a qualquer momento e independentemente do nó que esteja selecionado pelo cursor do teclado.

Critério de sucesso 1.1.6: É possível pesquisar por um nó específico do gerenciador de arquivos de um projeto de software.

Critério de sucesso 1.1.7: É possível obter informações sobre a localização de um nó específico, em relação à estrutura geral do gerenciador de arquivos de um projeto de software.

Critério de sucesso 1.1.8: É possível obter informações sobre a distância do nó selecionado em relação ao nó raiz do gerenciador de arquivos de um projeto de software.

---

### **Por que fazer?**

A exploração da estrutura de um projeto de software utilizando as teclas de seta para subir ou descer no gerenciador de arquivos é desafiadora para estudantes cegos, pois o leitor de telas verbaliza os rótulos dos componentes gráficos, mas não transmite nenhuma informação sobre o significado lógico e o relacionamento de um nó com seus irmãos ou pais. Em outras palavras, os leitores de tela não são inerentemente capazes de comunicar a funcionalidade das visualizações hierárquicas. Além disso, estudantes cegos também podem ter dificuldade para distinguir o relacionamento entre classes e subclasses em uma base de código, especialmente quando existem várias subclasses que herdam todas ou algumas das características da classe principal.

---

### **Como fazer?**

1. Organizar a estrutura de um projeto de software e do código-fonte utilizando uma árvore de navegação hierárquica. Ao posicionar o cursor do teclado em um nó da árvore, o usuário pode utilizar atalhos para solicitar informações, tais como: (1) descrição do nó e seu relacionamento hierárquico desse nó com os demais nós; e, (2) descrição do papel do nó atual em relação à estrutura global do projeto de software.
2. A navegação pelos nós da árvore deve seguir a lógica da hierarquia da árvore e não a ordem na qual os elementos são listados dentro da hierarquia da árvore.

*Continua na próxima página...*

3. Fornecer atalhos de teclado para: (1) consultar localização de um nó específico; (2) consultar a distância de um nó em relação à raiz da árvore; (3) verificar quais são os nós primos de um nó selecionado; (4) consultar a densidade de uma subárvore dentro da hierarquia; (5) pesquisar por um nó específico na árvore hierárquica; (6) mover o cursor do teclado para o nó principal da árvore hierárquica; (7) mover o cursor para os nós ascendentes, irmãos e descendentes de um nó específico; e, (8) alterar o contexto quando o foco do teclado for direcionado para um novo ponto de ancoragem.

---

**IMPORTANTE:** As orientações presentes na **Diretriz 1.1 (Semântica)** precisam ser consistentes com as recomendações presentes na **Diretriz 5.2 (Visão Geral de Código)** e na **Diretriz 3.6 (Atalhos)**.

---

**Fonte:**

Hutchinson; Metatla (2018)

Mealin; Murphy-hill (2012)

Schanzer; Bahram; Krishnamurthi (2019)

Smith et al. (2003)

---

## 8.2 CATEGORIA 2: Depuração de Código

A Categoria 2, **Depuração de Código**, inclui três Diretrizes destinadas a auxiliar os estudantes cegos nas atividades relacionadas à identificação de erros no código, compreensão das causas dos erros e sua correção.

Tabela 10 – Diretriz 2.1 Informações sobre erros

---

### **Diretriz 2.1 Informações sobre erros**

---

**Descrição**

Informar o usuário sobre a existência de erros no código.

---

**Critérios de Sucesso**

Critério de sucesso 2.1.1: Os erros de sintaxe são informados em tempo real.

Critério de sucesso 2.1.2: Em caso de falha na compilação do código-fonte, uma lista dos erros de sintaxe detectados é exibida.

Critério de sucesso 2.1.3: O usuário é informado quando não forem encontrados erros de sintaxe após a compilação de um código-fonte.

Critério de sucesso 2.1.4: O usuário é informado sobre o sucesso ou falha na execução de testes unitários.

---

*Continua na próxima página...*

---

**Por que fazer?**

Os IDEs fornecem a maioria das informações sobre erros por meio de dicas visuais, como cores que destacam a sintaxe ou ícones que exibem informações sobre como corrigir o erro ao passar o mouse sobre eles. Esses recursos não estão disponíveis de maneira proativa para estudantes cegos, que podem enfrentar desafios ao identificar, por exemplo, a ausência, o excesso ou o posicionamento inadequado de caracteres, operadores e parênteses, uma vez que não existem mecanismos de suporte em tempo real alternativos para o reconhecimento de erros durante a escrita de código. Além disso, os IDEs também usam cores para indicar o sucesso ou falha de testes unitários. Se essas informações não forem apresentadas em um formato acessível para leitores de tela, a compreensão da informação pode ser prejudicada.

---

**Como fazer?**

1. Utilizar alertas sonoros em tempo real para informar o usuário sobre a presença de erros de sintaxe no código-fonte.
  2. Fornecer atalhos para listar os erros de sintaxe identificados após falha na compilação do código-fonte. Deve ser possível navegar até a linha que contém o erro pressionando a tecla ENTER.
  3. Utilizar alertas sonoros para informar quando não for encontrado nenhum erro de sintaxe após a compilação de um código-fonte.
  4. Utilizar técnicas de Inteligência Artificial (IA) para ajudar o usuário na identificação de erros, bem como para oferecer sugestões de aprimoramento e correção do código.
- 

**IMPORTANTE:** As orientações presentes na **Diretriz 2.1 (Informações sobre erros)** precisam ser consistentes com as recomendações presentes na **Diretriz 3.6 (Atalhos)** e na **Diretriz 7.1 (Alertas sonoros)**.

---

**Fonte:**

Albusays; Ludi (2016)

Petrausch; Loitsch (2017)

Potluri et al. (2018)

Sampath; Merrick; Macvean (2021)

Zen; Costa; Tavares (2023)

---

Tabela 11 – Diretriz 2.2 Variáveis e constantes

---

**Diretriz 2.2 Variáveis e constantes****Descrição**

Fornecer informações sobre variáveis e constantes declaradas no código-fonte.

---

*Continua na próxima página...*

---

### **Critérios de Sucesso**

Critério de sucesso 2.2.1: É possível obter uma lista de todas as variáveis e constantes declaradas em uma base de código.

Critério de sucesso 2.2.2: É possível obter informações como nome, valor, tipo, modificador de acesso e escopo de variáveis e constantes declaradas em uma base de código.

Critério de sucesso 2.2.3: É possível alterar o nome, o valor, o tipo ou o modificador de acesso de uma variável uma única vez e solicitar que o sistema atualize automaticamente todas as ocorrências dessa variável no código-fonte.

---

### **Por que fazer?**

Informações sobre nomes, valores, tipos e modificadores de acesso de variáveis não estão facilmente disponíveis para estudantes cegos. É necessário percorrer o código manualmente para consultar essas informações. Entretanto, usuários de leitores de tela não conseguem examinar o código sem mover o cursor também. Para encontrar algo como a grafia do identificador de uma variável, é necessário mover o cursor até o local onde a variável está declarada para que o leitor de tela verbalize a informação.

---

### **Como fazer?**

1. Fornecer uma lista resumida dos nomes, valores, tipos, modificadores de acesso e escopo de variáveis e constantes declaradas em uma base de código, acessível por meio de atalhos de teclado.
  2. Fornecer atalhos de teclado para obter informações sobre valor, tipo, modificador de acesso e escopo de uma variável ou constante que estiver atualmente selecionada pelo cursor do teclado.
  3. Permitir que o usuário modifique o nome, valor, tipo ou modificador de acesso de uma variável ou constante e solicite que o IDE atualize automaticamente o código-fonte em todas as instâncias que essa variável ou constante estiver sendo utilizada.
- 

### **IMPORTANTE:**

As orientações presentes na **Diretriz 2.2 (Variáveis e Constantes)** precisam ser consistentes com as recomendações presentes na **Diretriz 3.6 (Atalhos)**.

---

### **Fonte:**

Mealin; Murphy-hill (2012)

Potluri et al. (2018)

---

Tabela 12 – Diretriz 2.3 Pontos de interrupção

---

## **Diretriz 2.3 Pontos de interrupção**

---

### **Descrição**

Fornecer recursos para reconhecimento, consulta e utilização de pontos de interrupção.

---

### **Critérios de Sucesso**

Critério de sucesso 2.3.1: É possível identificar pontos de interrupção inseridos em um arquivo de código-fonte.

Critério de sucesso 2.3.2: É possível listar todos os pontos de interrupção inseridos em um arquivo de código-fonte.

Critério de sucesso 2.3.3: É possível inserir pontos de interrupção em um arquivo de código-fonte.

Critério de sucesso 2.3.4: O sistema notifica o usuário quando a execução é pausada devido à presença de um ponto de interrupção no código.

---

### **Por que fazer?**

Informações sobre pontos de interrupção não estão facilmente disponíveis para usuários de leitores de tela, impondo desafios para a identificação e utilização desses recursos por estudantes cegos.

---

### **Como fazer?**

1. Utilizar alertas sonoros para informar sobre pontos de interrupção inseridos no código-fonte, quando o cursor estiver posicionado sobre a linha correspondente.
  2. Disponibilizar atalhos de teclado para listar os pontos de interrupção em um arquivo de código-fonte. Ao selecionar um ponto de interrupção, o usuário pode acessar informações, como a localização no código, escopo e descrição. Além disso, o IDE deve permitir o deslocamento do cursor do teclado para a linha de código correspondente ao ponto de interrupção.
  3. Permitir que o usuário utilize atalhos para inserir pontos de interrupção em locais específicos no código-fonte e adicione descrições personalizadas para documentar o motivo pelo qual o ponto de interrupção foi definido.
- 

**IMPORTANTE:** As orientações presentes na **Diretriz 2.3 (Marcadores e Pontos de Interrupção)** precisam ser consistentes com as recomendações presentes na **Diretriz 3.6 (Atalhos)** e na **Diretriz 7.1 (Alertas sonoros)**.

---

### **Fonte:**

Potluri et al. (2018)

---

### 8.3 CATEGORIA 3: Navegação no Código

A categoria **Navegação no Código** contém sete diretrizes que abordam recomendações destinadas a auxiliar estudantes cegos a navegar pelo código-fonte e pela interface do IDE, localizar trechos de código ou funcionalidades especificadas do ambiente, além de identificar o contexto ou o nível de escopo de uma determinada linha, ou bloco de código.

Tabela 13 – Diretriz 3.1 Rótulos

---

#### Diretriz 3.1 Rótulos

---

##### Descrição

Fornecer descrições textuais suficientemente claras para todo conteúdo não textual.

##### Crterios de Sucesso

Crterio de sucesso 3.1.1: Todo conteúdo não textual exibido ao usuário possui uma descrição textual alternativa clara para transmitir o seu significado, bem como o seu destino ou funcionalidade associada.

##### Por que fazer?

Para usuários cegos, é crucial fornecer texto alternativo significativo para descrever elementos não textuais presentes na interface gráfrica, como *links* e ícones. Caso contrário, embora a presença destes elementos possa ser percebida pelo usuário, a compreensão de sua funcionalidade pode ser prejudicada.

##### Como fazer?

1. Fornecer descrições de texto alternativas para conteúdo não textual (imagens, gráficos e outros elementos visuais), visíveis ou não, permitindo que os usuários com deficiência visual tenham acesso a todas as informações por meio de leitores de tela.

##### Fonte:

ISO 9241-171 (2018)

Petrausch; Loitsch (2017)

Potluri et al. (2018)

WCAG 2.2 (2023)

---

Tabela 14 – Diretriz 3.2 Movimentação pelo código

---

#### Diretriz 3.2 Movimentação pelo código

---

##### Descrição

Oferecer recursos alternativos para navegar pela base de código.

*Continua na próxima página...*

---

**Critérios de Sucesso**

Critério de sucesso 3.2.1: É possível deslocar o cursor do teclado para o início, o meio ou o fim de um arquivo de código-fonte.

Critério de sucesso 3.2.2: É possível mover o cursor do teclado de volta para a última linha do código-fonte que tenha sido editada.

Critério de sucesso 3.2.3: É possível localizar e mover o cursor do teclado para posições específicas de um arquivo de código-fonte.

Critério de sucesso 3.2.4: É possível marcar linhas de código ou trechos de código em um arquivo de código-fonte para consultas futuras.

---

**Por que fazer?**

Para se movimentar no código, IDEs limitam os usuários a apenas algumas ações:

- Usar as teclas de seta para percorrer cada linha do código-fonte. Pode ser difícil e demorado, por exemplo, retornar rapidamente para uma linha de código específica ao revisar instruções do código-fonte, pois os usuários de leitores de tela não conseguem examinar o código sem mover o cursor também. Essa limitação pode acarretar inserção, exclusão ou alteração de código em locais incorretos, ou não intencionais.
  - Usar o gerenciador de arquivos para navegar até um método ou arquivo de código-fonte. Depois que o arquivo ou método é selecionado, é necessário navegar pelo código de forma sequencial, linha por linha.
  - Usar um mecanismo de pesquisa. É menos eficiente e mais difícil para um usuário cego localizar informações específicas em uma base de código. A realização de busca utilizando palavras-chave pode ser demorada e frustrante, já que uma palavra-chave pode aparecer em vários locais na mesma base de código.
- 

**Como fazer?**

1. Poderia ser utilizado um recurso de árvore de navegação hierárquica que possibilite navegar entre as funções e métodos declarados no arquivo de código atual.
2. Atalhos de teclado poderiam ser utilizados para deslocar o cursor do teclado para o início, para o meio ou para o fim de um arquivo de código-fonte, bem como para uma linha de código específica ou para a última linha de código que tenha sido editada.

*Continua na próxima página...*

3. Utilizar marcadores ou *tags* em locais específicos do código-fonte para marcar uma instrução de código e retornar a ela posteriormente para modificações adicionais. É recomendado utilizar o recurso de marcador para saltar para uma instrução de código específica, ao invés de um número de linha específico, porque a numeração das linhas de código pode ser alterada quando uma linha ou trecho de código é inserido ou excluído. Além disso, seria interessante que se pudesse classificar esses marcadores como "públicos" ou "privados". Marcadores privados só seriam acessíveis às pessoas que os adicionaram, enquanto marcadores públicos poderiam ser acessados por todos os envolvidos em um projeto.

---

**IMPORTANTE:** As orientações presentes na **Diretriz 3.2 (Movimentação pelo Código)** precisam ser consistentes com as recomendações presentes na **Diretriz 1.1 (Semântica)** e na **Diretriz 3.6 (Atalhos)**.

---

**Fonte:**

Albusays; Ludi (2016)  
 Albusays; Ludi; Huenerfauth (2017)  
 Alotaibi; Al-khalifa; Alsaeed (2020)  
 Baker; Bennett; Ladner (2019)  
 Mealin; Murphy-hill (2012)  
 Potluri et al. (2018)

---

Tabela 15 – Diretriz 3.3 Contexto e nível de escopo

---

### **Diretriz 3.3 Contexto e nível de escopo**

---

**Descrição**

Fornecer informações sobre contexto e nível de escopo de uma determina linha ou trecho de código.

---

**Critérios de Sucesso**

Critério de sucesso 3.3.1: É possível obter informações a respeito do nível de escopo de uma variável, constante ou instrução de fluxo de controle inserida no código-fonte.

Critério de sucesso 3.3.2: É possível obter informações a respeito do início e do fim de uma instrução de fluxo de controle ou bloco de código.

Critério de Sucesso 3.3.3: É possível verificar o nível de profundidade do aninhamento de uma instrução de fluxo de controle ou bloco de código.

Critério de sucesso 3.3.4: O sistema oferece informações sobre a instrução correspondente sempre que o usuário insere um parêntese, chave ou colchete de fechamento no código-fonte.

---

*Continua na próxima página...*

---

**Por que fazer?**

Pode ser difícil para estudantes cegos compreender o nível de escopo enquanto navegam por estruturas profundamente aninhadas, bem como detectar o início e o fim de um bloco de código. Os recursos visuais utilizados pelos IDEs para auxiliar nesse processo, como o alinhamento de caracteres especiais ou recuos que marcam o início e o fim de bloco de código, não são facilmente acessíveis aos leitores de telas.

---

**Como fazer?**

1. Utilizar uma estrutura semelhante aos níveis de cabeçalho de um arquivo HTML para identificar informações estruturais como classes, métodos e instruções de fluxo de controle. O nível do cabeçalho identificaria o nível de escopo de uma determinada linha de código e sua localização relativa no código.
2. Utilizar diferentes alertas sonoras para representar o início ou o fim de uma instrução de fluxo de controle ou bloco de código.
3. Disponibilizar recurso de indicador de nível de aninhamento e escopo, que leria em voz alta a localização atual do cursor quando uma combinação especial de teclas de atalho fosse pressionada.

---

**IMPORTANTE:** As orientações presentes na **Diretriz 3.3 (Contexto e Nível de Escopo)** precisam ser consistentes com as recomendações presentes na **Diretriz 3.6 (Atalhos)** e na **Diretriz 7.1 (Alertas sonoros)**.

---

**Fonte:**

Albusays; Ludi; Huenerfauth (2017)  
Armaly; Rodeghero; Mcmillan (2018)  
Baker; Milne; Ladner (2015)  
Baker; Bennett; Ladner (2019)  
Huff et al. (2020)  
Hutchinson; Metatla (2018)  
Zen; Costa; Tavares (2023)

---

Tabela 16 – Diretriz 3.4 Estratégias de navegação

---

**Diretriz 3.4 Estratégias de navegação**

---

**Descrição**

Fornecer diferentes possibilidades de organização e navegação pelo conteúdo.

---

**Critérios de Sucesso**

Critério de sucesso 3.4.1: É possível escolher entre interagir com uma interface gráfica ou uma interface simplificada, com um *layout* baseado em texto e menus organizados em níveis.

*Continua na próxima página...*

Critério de sucesso 3.4.2: É possível monitorar processos executando em segundo plano.

Critério de sucesso 3.4.3: É possível determinar a localização do usuário dentro da hierarquia de menus e funcionalidades disponibilizadas pelo sistema.

Critério de sucesso 3.4.4: Todos os recursos e funcionalidades do IDE podem ser acessados e utilizados por meio do teclado.

Critério de sucesso 3.4.5: O usuário é notificado quando o foco do cursor do teclado é movido para o *console* do IDE ou para a interface gráfica gerada após a compilação do código-fonte.

---

### **Por que fazer?**

As interfaces dos IDEs concentram-se quase que exclusivamente em estímulos visuais, utilizando estruturas de menus encapsuladas, caixas de diálogo, ícones e outros elementos gráficos e são difíceis de interagir por meio de leitores de tela. O acesso a essas informações utilizando leitor de tela depende do foco do cursor, ou seja, o usuário precisa buscar ativamente informações em vários componentes do IDE e definir explicitamente o foco no painel apropriado, sem a possibilidade de analisar as ferramentas periféricas. Além disso, alguns recursos que estão abertamente visíveis na interface gráfica do IDE, podem ficar ocultos dentro de vários níveis de hierarquia de navegação e são difíceis de acessar quando se utiliza leitores de tela. Em certas situações, os estudantes cegos podem não ter conhecimento da presença de um painel contendo as informações que procura.

---

### **Como fazer?**

1. Fornecer atalhos de teclado que permitam escolher entre interagir com uma interface gráfica ou uma interface simplificada, com um *layout* baseado em texto e menus organizados em níveis.
2. Criar uma estrutura de conteúdo lógica e bem-organizada, facilitando a navegação e a compreensão do conteúdo por estudantes cegos.
3. Utilizar alertas sonoros para informar a finalização ou o resultado de algum processo que estiver executando em segundo plano.
4. Utilizar alertas sonoros para indicar a mudança do foco do cursor da IDE para o console ou interface gráfica gerada após a compilação do código-fonte

---

**IMPORTANTE:** As orientações presentes na **Diretriz 3.4 (Estratégias de Navegação)** precisam ser consistentes com as recomendações presentes na **Diretriz 3.1 (Rótulos)**, na **Diretriz 3.6 (Atalhos)** e na **Diretriz 7.1 (Alertas sonoros)**.

---

### **Fonte:**

Mealin; Murphy-hill (2012)

*Continua na próxima página...*

Potluri et al. (2018)  
 (Stefik et al., 2009)  
 WCAG 2.2 (2023)  
 Zen; Costa; Tavares (2023)

---

Tabela 17 – Diretriz 3.5 Numeração de linhas

---

## **Diretriz 3.5 Numeração de linhas**

---

### **Descrição**

Informar a numeração das linhas do código-fonte.

### **Critérios de Sucesso**

Critério de sucesso 3.5.1: É possível obter informação sobre o número da linha de código em que o cursor do teclado está posicionado.

Critério de sucesso 3.5.2: É possível obter informação sobre o número da linha inicial e a linha final de um trecho de código.

### **Por que fazer?**

Os números de linha, no editor de código-fonte dos IDEs, facilitam a navegação, a depuração, a colaboração e a manutenção do código-fonte, mas não são acessíveis aos leitores de tela.

### **Como fazer?**

1. Fornecer atalhos de teclado para consultar o número de uma linha específica de código, bem como para consultar o número da linha inicial e o número da linha final de um determinado trecho de código.

**IMPORTANTE:** As orientações presentes na **Diretriz 3.5 (Numeração de Linhas)** precisam ser consistentes com as recomendações presentes na **Diretriz 3.6 (Atalhos)**.

### **Fonte:**

Albusays; Ludi; Huenerfauth (2017)

---

Tabela 18 – Diretriz 3.6 Atalhos

---

## **Diretriz 3.6 Atalhos**

---

### **Descrição**

Fornecer funções e comandos facilmente operáveis por meio de teclado.

### **Critérios de Sucesso**

Critério de sucesso 3.6.1: São disponibilizados atalhos de teclado para todas as funcionalidades do IDE.

*Continua na próxima página...*

Critério de sucesso 3.6.2: São disponibilizados atalhos de teclado para digitação automatizada de instruções das linguagens de programação.

Critério de sucesso 3.6.3: É possível consultar todos os atalhos de teclado configurados no IDE.

Critério de sucesso 3.6.4: É possível configurar novos atalhos de teclado e alterar os atalhos já cadastrados no IDE.

Critério de sucesso 3.6.5: O usuário pode criar uma lista de atalhos de teclado favoritos para facilitar consultas futuras.

Critério de sucesso 3.6.6: Os atalhos de teclado para as funcionalidades do IDE seguem as convenções e padrões adotados por outros IDEs, softwares e Sistemas Operacionais.

---

### **Por que fazer?**

Determinar os equivalentes em teclado (atalhos) para as funcionalidades do IDE pode ser complexo para estudantes cegos, pois não existe uma forma acessível de consultar essas informações na própria ferramenta, tornando necessário solicitar auxílio de professores e colegas ou realizar uma busca externa. Além disso, pode ser difícil memorizar todos os atalhos necessários para acessar as funcionalidades do IDE. A falta de padronização também representa um obstáculo, pois em algumas situações tanto o IDE quanto o leitor de tela utilizam os mesmos atalhos para funções diferentes ou IDEs diferentes utilizam atalhos distintos para acessar a mesma funcionalidade. Essa falta de padronização pode confundir o usuário.

---

### **Como fazer?**

1. Disponibilizar um guia de referência rápida para os atalhos utilizados no IDE.
2. Informar os atalhos correspondentes a uma determinada funcionalidade quando o cursor do teclado estiver posicionado sobre o seu respectivo ícone ou *link*.
3. Oferecer meios para que o usuário possa personalizar os atalhos, de acordo com suas preferências

---

### **Fonte:**

Albusays; Ludi; Huenerfauth (2017)

Baker; Bennett; Ladner (2019)

Petrausch; Loitsch (2017)

Potluri et al. (2018)

Zen; Costa; Tavares (2023)

---

Tabela 19 – Diretriz 3.7 Ajuda e Documentação

---

**Diretriz 3.7 Ajuda e Documentação**

---

**Descrição**

Fornecer diferentes formas de ajuda e documentação ao usuário: dicas, tutoriais, recomendações ou exemplos.

---

**CrITÉrios de Sucesso**

CrITÉrio de Sucesso 3.7.1: Os tutoriais e instruções pōem ser facilmente acessados.

CrITÉrio de Sucesso 3.7.2: É fornecida uma descriçāo resumida de cada funcionalidade do sistema.

CrITÉrio de Sucesso 3.7.3: Toda documentaçāo do IDE é acessível aos leitores de tela.

CrITÉrio de Sucesso 3.7.4: Ao instalar uma nova versāo do IDE, o usuáριο é informado sobre funcionalidades adicionadas e/ou removidas.

CrITÉrio de Sucesso 3.7.5: Ao instalar uma nova versāo do IDE, o usuáριο é informado sobre os recursos de acessibilidade adicionados e/ou removidos.

CrITÉrio de Sucesso 3.7.6: Ao instalar uma nova versāo do IDE, o usuáριο pode optar por manter ou alterar as configuraçōes de acessibilidade utilizadas na versāo anterior.

---

**Por que fazer?**

Os tutoriais e instruções devem ser acessíveis para os estudantes cegos, uma vez que documentaçōes detalhadas das funcionalidades podem aprimorar a eficiēncia e facilitar o uso dessas ferramentas. Alēm disso, é importante observar que cada nova versāo de um IDE pode introduzir novos recursos, e as funcionalidades existentes podem ser modificadas. Muitas dessas alteraçōes sāo representadas visualmente, mas nāo hā uma abordagem estruturada para informar os usuários cegos sobre essas mudançās.

---

**Como fazer?**

1. É necessáριο projetar tutoriais especificamente voltados para as necessidades das pessoas cegas.
  2. Cada funcionalidade do sistema deve vir acompanhada de uma descriçāo concisa de sua finalidade e instruções sobre como utilizá-la. Essas informaçōes devem estar acessíveis por meio de atalhos de teclado.
  3. No momento da instalaçāo de uma nova versāo de um IDE, o usuáριο deveria ser informado sobre a inclusāo de novos recursos e sobre as modificaçōes realizadas nas funcionalidades jā existentes.
  4. O processo de instalaçāo de uma nova versāo do IDE deveria consultar o usuáριο sobre a preferēncia de manter as configuraçōes de acessibilidade da versāo anterior. Em caso afirmativo, o sistema deveria informar e aplicar essas configuraçōes automaticamente.
- 

*Continua na prōxima página...*

---

**IMPORTANTE:** A orientação presente na **Diretriz 3.7 (Ajuda e Documentação)** precisam ser consistentes com as recomendações presentes na **Diretriz 3.6 (Atalhos)**.

---

**Fonte:**

Albusays; Ludi; Huenerfauth (2017)

Pandey et al. (2022)

---

## 8.4 CATEGORIA 4: Edição de Código

Na categoria **Edição de Código** são fornecidas três diretrizes contendo recomendações para auxiliar estudantes cegos durante a escrita do código-fonte. Estas diretrizes abrangem a identificação de espaçamento utilizado para indentação, o reconhecimento e aproveitamento de recursos como o autocompletar, a inserção automática de caracteres não-alfanuméricos realizada pelos IDEs e a utilização de comentários de código-fonte.

Tabela 20 – Diretriz 4.1 Espaçamento

---

### Diretriz 4.1 Espaçamento

---

**Descrição**

Informar a quantidade de espaços a serem adicionados ou removidos em linguagens de programação que utilizam essa técnica para indicar indentação.

---

**Critérios de Sucesso**

Critério de sucesso 4.1.1: Em linguagens que utilizam espaços em branco para indicar indentação, esses espaços são verbalizados como uma lista completa de espaços em branco, indicando o comprimento total do recuo.

---

**Por que fazer?**

Algumas linguagens de programação usam recuo de espaço em branco para delimitar blocos de código, ao invés de usar palavras-chave ou chaves. Nessas linguagens, um aumento no recuo pode indicar um bloco de código novo e mais profundo, e uma diminuição no recuo indica o fim de um bloco de código. O leitor de tela geralmente interpreta todos os espaços em branco como uma sequência de caracteres “espaço” individuais, em vez de um único recuo de comprimento específico.

---

**Como fazer?**

1. Efetuar a contagem da quantidade de espaços em branco empregados para especificar um recuo de código e informar o comprimento total deste recuo.
- 

*Continua na próxima página...*

---

**Fonte:**

Albusays; Ludi; Huenerfauth (2017)

Potluri et al. (2018)

---

Tabela 21 – Diretriz 4.2 Autocompletar

---

**Diretriz 4.2 Autocompletar**

---

**Descrição**

Notificar a presença e realizar a leitura das informações exibidas pelo recurso de autocompletar.

---

**Critérios de Sucesso**

Critério de sucesso 4.2.1: O usuário é informado a respeito da existência do recurso autocompletar a partir da posição de foco do teclado.

Critério de sucesso 4.2.2: O conteúdo do recurso autocompletar é acessível aos leitores de tela.

Critério de sucesso 4.2.3: O usuário consegue solicitar explicitamente a leitura do conteúdo exibido no recurso autocompletar.

Critério de sucesso 4.2.4: O recurso de autocompletar exibe sugestões com base na entrada do usuário, mantendo o foco do leitor de tela no conteúdo disponibilizado por este recurso durante a digitação.

Critério de sucesso 4.2.5: O usuário é notificado quando o IDE adiciona um caractere de fechamento de instrução ou bloco de código (como chaves, parênteses ou colchetes), ou quando insere uma instrução sugerida pelo recurso autocompletar.

Critério de sucesso 4.2.6: O usuário pode ativar ou desativar o recurso autocompletar a qualquer momento.

---

**Por que fazer?**

Recursos como preenchimento automático, que auxiliam o programador a recordar a nomenclatura de classes, métodos e atributos, são exibidos por meio de um menu *pop-up* contendo previsões de digitação que podem ser realizadas. Esses recursos não são acessíveis aos leitores de tela, exceto se o usuário o configurar explicitamente para executar essa funcionalidade. Além disso, a maioria dos IDEs insere automaticamente caracteres que delimitam o fim de instruções ou blocos de código quando se insere o caractere de abertura (por exemplo: chaves, colchetes, parênteses ou outros caracteres de fechamento correspondentes). Ao informar os estudantes sobre essa inserção automática, evita-se a inserção inadvertida de caracteres desnecessários e possíveis problemas de sintaxe no código.

---

*Continua na próxima página...*

---

**Como fazer?**

1. Utilizar alertas sonoros para notificar o estudante cego sobre a presença do recurso autocompletar, na posição de foco do teclado.
2. Fornecer atalhos de teclado para que o usuário possa solicitar que seja realizada a leitura do conteúdo do recurso autocompletar.
3. Utilizar alertas sonoros para notificar usuário quando o IDE adicionar automaticamente um caractere de fechamento de instrução ou bloco de código após o usuário digitar o caractere de início correspondente.

---

**IMPORTANTE:** As orientações presentes na **Diretriz 4.2 (Autocompletar)** precisam ser consistentes com as recomendações presentes na **Diretriz 3.6 (Atalhos)** e na **Diretriz 7.1 (Alertas sonoros)**.

---

**Fonte:**

Albusays; Ludi; Huenerfauth (2017)

Zen; Costa; Tavares (2023)

---

Tabela 22 – Diretriz 4.3 Comentários

---

**Diretriz 4.3 Comentários****Descrição**

Notificar a existência de comentários no código-fonte.

---

**Critérios de Sucesso**

Critério de sucesso 4.3.1: O usuário é informado sobre a existência de comentário na posição atual de foco do teclado.

Critério de sucesso 4.3.2: O usuário pode optar entre realizar a leitura ou ignorar um comentário no código.

Critério de sucesso 4.3.3: O usuário pode classificar os comentários.

---

**Por que fazer?**

Comentários são utilizados para tornar um código-fonte mais legível, facilitar tarefas de manutenção e depuração, documentar o funcionamento de um determinado trecho de código, auxiliar a superar barreiras de navegação, localizar erros ou *bugs* no código, ou para destacar instruções de código que requerem revisão adicional. Entretanto, enquanto desenvolvedores com visão podem ignorar grandes comentários de código (como documentação e licenças), usuários de leitores de tela podem ter dificuldades para navegar até o final desses comentários.

---

**Como fazer?**

1. Utilizar alertas sonoros para informar o usuário sobre a existência de comentários em determinados trechos de código.

*Continua na próxima página...*

2. Fornecer atalhos de teclado que possibilitem que o usuário, explicitamente, solicite que seja feita a leitura do comentário.
3. Permitir que o usuário classifique os comentários de código conforme a sua função. Exemplos de classificação: tornar o código legível ou descrever uma funcionalidade; localizar erros que precisam de correção; ou, destacar declarações de código que necessitam de revisões futuras.
4. Fornecer meios de classificar comentários de código, tais como: comentários públicos, que podem ser visualizados por qualquer membro da equipe de desenvolvimento; ou, comentários privados, acessíveis apenas para o usuário que declarou o comentário.

---

**IMPORTANTE:** As orientações presentes na **Diretriz 4.3 (Comentários)** precisam ser consistentes com as recomendações presentes na **Diretriz 3.6 (Atalhos)** e **Diretriz 7.1 (Alertas Sonoros)**.

---

**Fonte:**

Albusays; Ludi; Huenerfauth (2017)  
Potluri et al. (2018)

---

## 8.5 CATEGORIA 5: *Skimming* de Código

As diretrizes contidas na categoria ***Skimming* de Código** visam auxiliar estudantes cegos a obter uma visão geral dos diferentes arquivos que compõem um projeto de software e utilizar recursos que permitem recolher ou expandir trechos de código.

Tabela 23 – Diretriz 5.1 Dobra de Código

---

### **Diretriz 5.1 Dobra de Código**

---

**Descrição**

Permitir ao usuário ocultar ou expandir seções de código, bem como identificar trechos de código ocultos.

**Critérios de Sucesso**

Critério de sucesso 5.1.1: O usuário é informado quando um trecho de código está dobrado (oculto) e pode ser expandido.

Critério de sucesso 5.1.2: O usuário pode expandir um trecho de código que estiver dobrado (oculto).

Critério de sucesso 5.1.3: O usuário pode ocultar (dobrar) um trecho de código.

*Continua na próxima página...*

---

**Por que fazer?**

O recurso de dobra de código permite ocultar e exibir seletivamente seções de um arquivo de código-fonte, permitindo que os programadores tenham uma visão melhor das instruções de código no entorno da região colapsada. Esse recurso não é acessível aos leitores de tela, e os estudantes cegos podem não ser capazes de identificar áreas ocultas no código ou não ter clareza sobre o uso desse recurso.

---

**Como fazer?**

1. Utilizar alertas sonoros para notificar o usuário sobre a presença de um código oculto quando o cursor do teclado estiver posicionado sobre a linha correspondente.
2. Fornecer atalhos de teclado para que o usuário possa dobrar (ocultar) ou expandir um trecho de código.

---

**IMPORTANTE:** As orientações presentes na **Diretriz 5.1 (Dobra de Código)** precisam ser consistentes com as recomendações presentes na **Diretriz 3.6 (Atalhos)** e na **Diretriz 7.1 (Alertas sonoros)**.

---

**Fonte:**

Petrausch; Loitsch (2017)

Potluri et al. (2018)

---

Tabela 24 – Diretriz 5.2 Visão geral do código

---

**Diretriz 5.2 Visão geral do código**

---

**Descrição**

Fornecer visão geral da base de código.

---

**Critérios de Sucesso**

Critério de sucesso 5.2.1: É possível obter a quantidade total de linhas de um arquivo de código-fonte.

Critério de sucesso 5.2.2: É possível obter uma lista de todos os arquivos que compõem um projeto de software.

Critério de sucesso 5.2.3: É possível navegar entre os arquivos que compõem um projeto de software.

Critério de sucesso 5.2.4: É possível obter uma lista de todas as funções e métodos de um arquivo de código-fonte.

Critério de sucesso 5.2.5: É possível navegar entre as funções e métodos de um arquivo de código-fonte.

---

*Continua na próxima página...*

---

**Por que fazer?**

Estudantes cegos têm dificuldade para obter uma visão geral de alto nível de seu código porque os leitores de tela os restringem a ler o código linha por linha, em sequência. Ao contrário de usuários com visão, que podem obter uma visão geral da estrutura do código rolando rapidamente para cima e para baixo em uma página, os leitores de tela forçam os usuários cegos a ler um documento inteiro.

---

**Como fazer?**

1. Organizar a estrutura de um projeto de software e do código-fonte em uma árvore de navegação hierárquica, listando todos os arquivos de código-fonte de um projeto, bem como todas as funções e métodos existentes em um arquivo de código-fonte. Deve ser possível navegar até o componente de código desejado pressionando a tecla ENTER.
2. No caso de código-fonte hospedado na internet, pode-se usar *tags* HTML para determinar informações estruturais, onde o tipo de *tag* de título (nível de cabeçalho) anexada a um elemento depende de sua localização relativa no código-fonte.
3. Fornecer atalhos do teclado para que o usuário possa consultar a quantidade de linhas de um arquivo de código-fonte.

---

**IMPORTANTE:** As orientações presentes na **Diretriz 5.2 (Visão geral de código)** precisam ser consistentes com as recomendações presentes na **Diretriz 1.1 (Semântica)** e na **Diretriz 3.6 (Atalhos)**.

---

**Fonte:**

Mealin; Murphy-hill (2012)

Potluri et al. (2018)

---

## 8.6 CATEGORIA 6: Interface Gráfica de Usuário

A categoria **Interface Gráfica de Usuário** contém recomendações para auxiliar estudantes cegos a elaborar códigos-fonte para a implementação de interfaces que contenham elementos visuais e a testá-las para verificar se atendem aos principais princípios de desenvolvimento de sistemas interativos e às recomendações básicas de acessibilidade.

Tabela 25 – Diretriz 6.1 Construção de Interface Gráfica

---

**Diretriz 6.1 Construção de Interface Gráfica**

---

**Descrição**

Fornecer recursos para auxiliar a construção de interfaces que utilizam componentes visuais.

---

**Critérios de Sucesso**

Critério de sucesso 6.1.1: O usuário pode determinar a posição dos componentes gráficos na tela.

Critério de sucesso 6.1.2: As dimensões dos componentes gráficos são atualizadas automaticamente, a partir de parâmetros especificados pelo usuário.

---

**Por que fazer?**

Usuários com visão podem utilizar ferramentas automatizadas fornecidas pelos IDEs para arrastar e soltar os componentes de uma interface gráfica e criar um *layout* rapidamente. Como as interações do mouse não são acessíveis para estudantes cegos, normalmente eles precisam escrever todo o código para construir uma interface gráfica, necessitando de um tempo maior para realizar essa tarefa.

---

**Como fazer?**

1. Dividir a tela em três linhas (superior, intermediária e inferior) e três colunas (esquerda, central e direita), oferecendo ao usuário a opção de escolher uma localização aproximada para posicionar os componentes gráficos nas diferentes áreas da tela, eliminando a necessidade de medições ou uso do método de tentativa e erro.
  2. Disponibilizar gerenciadores de *layout*: ferramentas que agrupam e organizam automaticamente os componentes da interface gráfica de acordo com restrições especificadas pelo estudante.
  3. Oferecer ao estudante a capacidade de estabelecer critérios temáticos para a interface e disponibilizar paletas de cores que permitam que ele defina as combinações de cores que seriam adotadas pelos elementos adicionados à interface.
- 

**Fonte:**

Kearney-volpe; Hurst (2021)

Pandey et al. (2022)

Siegfried (2006)

Zen; Costa; Tavares (2023)

---

Tabela 26 – Diretriz 6.2 Validação da interface gráfica

---

## **Diretriz 6.2 Validação da interface gráfica**

---

### **Descrição**

Realizar validação automatizada da interface gráfica gerada a partir de um código-fonte.

---

### **Critérios de Sucesso**

Critério de sucesso 6.2.1: É possível verificar se a interface gráfica gerada atende aos critérios de acessibilidade especificados pelo WCAG 2.2 (2023)

Critério de sucesso 6.2.2: É possível consultar informações espaciais como localização, altura e largura dos elementos da interface gráfica.

Critério de sucesso 6.2.3: É possível verificar a consistência relacionada ao espaçamento entre os elementos, bem como à utilização de fontes e cores.

---

### **Por que fazer?**

Estudantes cegos têm dificuldades para compreender e verificar se a saída gráfica gerada um código-fonte atende às necessidades e especificações desejadas. Essa dificuldade pode forçá-los a solicitar auxílio de colegas e professores ou procurar soluções alternativas. Além disso, a realização de tarefas que envolvem o desenvolvimento de interfaces de usuário faz com que muitos estudantes cegos sintam que não são capazes de realizá-las com a mesma eficiência ou confiança que os estudantes com visão. Por consequência, muitos estudantes cegos optam por concentrar seus esforços no aspecto lógico do programa.

---

### **Como fazer?**

1. Oferecer validadores de acessibilidade automatizados para verificar se a interface gráfica segue as recomendações estabelecidas na versão mais atual do WCAG 2.2 (2023)
  2. Fornecer informações espaciais como localização, altura e largura dos elementos da interface gráfica utilizando unidades de medida selecionadas pelo usuário (píxeis, centímetros ou milímetros, por exemplo).
  3. Verificar a consistência de espaçamento horizontal e vertical entre elementos no mesmo nível.
  4. Verificar se há variações significativas nas fontes utilizadas e se são aplicadas no máximo três tipos de fontes diferentes em uma página.
  5. Verificar se elementos gráficos de mesmo tipo compartilham as mesmas cores.
  6. Emitir uma lista de violações de projeto encontradas que seja acessível aos leitores de tela.
- 

### **Fonte:**

Baker; Bennett; Ladner (2019)

Huff et al. (2020)

*Continua na próxima página...*

Mealin; Murphy-hill (2012)  
 Pandey et al. (2022)  
 Petrausch; Loitsch (2017)  
 Potluri; He; Chen; Froehlich; Mankoff (2019)  
 Siegfried (2006)  
 Zen; Costa; Tavares (2023)

---

## 8.7 CATEGORIA 7: Sobrecarga Auditiva

No intuito de evitar estresse provocado pelo excesso de informações transmitidas por meio de áudio, a categoria **Sobrecarga Auditiva** contém um diretriz que traz orientações para a utilização desse tipo de recurso.

Tabela 27 – Diretriz 7.1 Alertas sonoros

---

### Diretriz 7.1 Alertas sonoros

---

#### Descrição

Permitir a personalização das configurações dos alertas sonoros.

---

#### Crterios de Sucesso

Crterio de sucesso 7.1.1: É possvel configurar o volume dos alertas sonoros

Crterio de sucesso 7.1.2: É possvel ativar ou desativar os alertas sonoros.

Crterio de sucesso 7.1.3: É possvel alterar o tipo de alerta sonoro associado a uma determinada funcionalidade.

Crterio de sucesso 7.1.4: É dada preferncia para alertas sonoros no verbais ao invs de alertas sonoros de fala.

---

#### Por que fazer?

Estudantes cegos podem experimentar estresse decorrente de sobrecarga auditiva ao receber muitas informaes transmitidas pelo canal de áudio de uma interface. Isso acontece porque alm da leitura dos elementos da interface e cdigo-fonte, os IDEs tambm utilizam alertas sonoros para repassar outras informaes aos usurios.

---

#### Como fazer?

1. A utilizao de tons curtos, claros e distinguveis é prefervel para oferecer *feedback* adicional, como erros e avisos. Especialmente quando é necessrio que esses alertas sejam emitidos ao mesmo tempo que é realizada a leitura de cdigo-fonte, comentrios ou entradas.
2. Deve-se implementar mecanismos que permitam ao usurio escolher entre os diferentes tipos de alertas sonoros, bem como o tipo de som que ser emitido para cada funcionalidade.

*Continua na prxima pgina...*

3. Deve-se implementar mecanismos que permitam ao usuário controlar o volume do áudio dos alertas sonoros emitidos, independentemente do nível global de volume do sistema operacional.
4. Deve-se implementar mecanismos que permitam ao usuário habilitar ou desabilitar os alertas sonoros associados a alguma funcionalidade.

---

**Fonte:**

Albusays; Ludi; Huenerfauth (2017)

Petrausch; Loitsch (2017)

Zen; Costa; Tavares (2023)

---

## 8.8 CATEGORIA 8: Leitura de Código-fonte

A categoria **Leitura de Código-fonte** especifica duas diretrizes que abordam recomendações para facilitar a compreensão do código-fonte e das mensagens de erro por parte dos estudantes cegos.

Tabela 28 – Diretriz 8.1 Idioma

---

**Diretriz 8.1 Idioma**

---

**Descrição**

Alternar automaticamente entre os idiomas pronunciados pelo leitor de telas.

**Critérios de Sucesso**

Critério de sucesso 8.1.1: A pronúncia das palavras-reservadas das linguagens de programação é realizada em língua inglesa.

Critério de sucesso 8.1.2: Os rótulos e demais componentes da interface são pronunciados no idioma da interface em uso.

Critério de sucesso 8.1.3: O usuário pode configurar o idioma que será utilizada para a pronúncia da nomenclatura atribuída a variáveis, classes e métodos/funções.

Critério de sucesso 8.1.4: As mensagens de erro podem ser traduzidas para a língua da interface em uso.

---

*Continua na próxima página...*

---

**Por que fazer?**

A pronúncia de palavras-reservadas das linguagens de programação e das mensagens de erro (em geral, em língua inglesa), realizada pelos leitores de tela, nem sempre é correta, o que pode atrapalhar o entendimento do conteúdo transmitido. Isso ocorre porque, normalmente, o leitor de telas está configurado para leitura na língua da interface em uso (no caso de estudantes brasileiros, a língua da interface em uso geralmente é a língua portuguesa) e não consegue realizar uma leitura do código-fonte de maneira que faça sentido para o ouvinte.

---

**Como fazer?**

1. Identificar, automaticamente, a posição de foco do cursor do teclado (código-fonte, *console*, mensagens de erro ou componentes da interface) e alternar entre os idiomas pronunciados pelo leitor de telas, de acordo com as configurações de pronúncia especificadas pelo usuário.
2. Fornecer atalhos que possibilitem a tradução das mensagens de erro para o idioma da interface em uso, abrangendo apenas o conteúdo textual dessas mensagens e estando em conformidade com as configurações de pronúncia especificadas pelo usuário.

---

**IMPORTANTE: IMPORTANTE:** As orientações presentes na **Diretriz 8.2 (Leitura contextual)** precisam ser consistentes com as recomendações presentes na **Diretriz 3.6 (Atalhos)**.

---

**Fonte:**

Sampath; Merrick; Macvean (2021)

Zen; Costa; Tavares (2023)

---

Tabela 29 – Diretriz 8.2 Leitura contextual

---

**Diretriz 8.2 Leitura contextual**

---

**Descrição**

Ler o código-fonte de maneira que faça sentido para o ouvinte.

---

**Critérios de Sucesso**

Critério de sucesso 8.2.1: A leitura do código-fonte fornece informações que facilitam a compreensão das instruções.

---

*Continua na próxima página...*

---

**Por que fazer?**

Ao realizar a leitura de um código-fonte, o leitor de telas verbaliza um toque de cada vez, como se o código fosse um texto, e o programa é dividido em nada mais do que uma série de caracteres, palavras e linhas. Como a sintaxe correta é imperativa na ciência da computação, até mesmo erros intermitentes podem confundir os estudantes. Além disso, as mensagens de erro podem conter muitos jargões (por exemplo, expressões regulares, siglas específicas do domínio etc.) que podem ser difíceis de serem verbalizadas por um leitor de tela, o que pode dificultar a compreensão dessas mensagens.

---

**Como fazer?**

1. Ao efetuar a leitura do código-fonte, o leitor de telas pode utilizar alertas sonoros para sinalizar o início ou o término de um bloco de código (geralmente representado por chaves), notificar a declaração ou invocação de um método (normalmente envolvendo parênteses) ou indicar a declaração e/ou acesso aos valores armazenados em um vetor (comumente realizado através do uso de colchetes). Esses caracteres não alfanuméricos podem ser omitidos quando o usuário optar por ler o código-fonte e serão pronunciados ou destacados quando ocorrer um erro de sintaxe em tempo real ou quando o usuário solicitar uma lista dos erros de sintaxe identificados no código-fonte.

---

**IMPORTANTE:** As orientações presentes na **Diretriz 8.1 (Idioma)** precisam ser consistentes com as recomendações presentes na **Diretriz 2.1 (Informações sobre erros)**, na **Diretriz 3.6 (Atalhos)** e na **Diretriz 7.1 (Alertas Sonoros)**.

---

**Fonte:**

Armaly; Rodeghero; Mcmillan (2018)  
Baker; Bennett; Ladner (2019)  
Sampath; Merrick; Macvean (2021)  
Schanzer; Bahram; Krishnamurthi (2019)  
Zen; Costa; Tavares (2023)

---

## 8.9 Outras Recomendações

Além deste conjunto de diretrizes, uma seção intitulada "Outras Recomendações" foi adicionada ao documento. Essa seção ressalta a importância de permitir que os recursos de acessibilidade sejam configurados conforme as preferências individuais de cada usuário, recomendando que o sistema ofereça a possibilidade de ativar ou desativar os recursos de acessibilidade oferecidos, a qualquer momento. Além disso, também sugere-se que o IDE possa automaticamente ativar ou desativar os recursos

de acessibilidade com base no perfil do usuário que estiver utilizando-o.

## 8.10 Considerações Parciais

No decorrer deste capítulo, foram apresentadas a 2ª versão das Diretrizes de Acessibilidade em IDEs para estudantes cegos, consolidando assim os resultados de todo o processo de validação e refinamento descritos nos Capítulos 6 e 7.

A versão final do documento, disponível em <https://elianazen.github.io/DiretrizesAcessibilidadeIDE/2VersaoDiretrizes.html>, contém não apenas as recomendações apresentadas neste Capítulo, mas também as seguintes seções:

- **Introdução.** Fornece uma visão geral do propósito e do contexto das diretrizes, explicando por que são importantes e como podem beneficiar os estudantes cegos;
- **Estrutura e Organização das Diretrizes.** Explora a organização e estrutura das diretrizes, delineando as categorias estabelecidas para agrupá-las e as informações contidas em cada uma delas;
- **Glossário.** Explica alguns termos técnicos específicos que podem não ser familiares para todos os leitores; e,
- **Referências Bibliográficas.** Lista todas as fontes bibliográficas utilizadas na elaboração das diretrizes.

O Capítulo 9, por fim, sintetiza as descobertas e lições aprendidas ao longo deste estudo, além de destacar as limitações da pesquisa e possíveis direcionamentos para trabalhos futuros.

## 9 CONCLUSÕES

Nos últimos anos, esforços significativos em diversas áreas têm sido empreendidos para promover a inclusão e a igualdade de acesso aos sistemas interativos por pessoas com deficiência. Dentre esses esforços, destacam-se aqueles ligados ao desenvolvimento de recursos de Tecnologia Assistiva e à elaboração de normas e diretrizes voltadas ao projeto de sistemas acessíveis. Entretanto, embora muito progresso tenha sido feito, ainda há desafios significativos a serem enfrentados para garantir que os sistemas interativos sejam verdadeiramente acessíveis para todas as pessoas.

No contexto educacional, é comum que muitos estudantes com deficiência ainda se deparem com desafios consideráveis, que dificultam tanto o ingresso quanto a conclusão dos estudos. Essas dificuldades podem se manifestar em todas as fases de sua formação, desde o acesso aos materiais didáticos até a execução de atividades e as interações com professores e colegas.

Especificamente nos cursos de Computação, especialmente no caso de estudantes cegos, tais obstáculos podem restringir ou até mesmo impedir o aprendizado de disciplinas fundamentais, como programação de computadores. Tais limitações podem repercutir substancialmente no processo de aprendizado e no desenvolvimento profissional desses alunos, visto que essas disciplinas permeiam os cursos de Computação e são fundamentais para a resolução de problemas, o desenvolvimento de software e a criação de soluções tecnológicas.

Compreender as necessidades e desafios enfrentados pelos estudantes cegos nas disciplinas de programação de computadores contribui para fomentar discussões e avançar em pesquisas que visam promover a inclusão desses estudantes nos cursos de computação e, conseqüentemente, no mercado de trabalho de tecnologia. A ampliação e a valorização da participação de pessoas com deficiência nos cursos de Computação não apenas promove a diversidade e a igualdade de oportunidades, mas também impulsiona o avanço da Ciência e da Tecnologia, bem como a inovação e o progresso na criação de um mundo digital mais inclusivo e acessível para todos.

No intuito de investigar este cenário e propor soluções para o problema, esta pesquisa foi norteada pelo objetivo: "*Estabelecer um conjunto de Diretrizes de ACESSI-*

*bilidade para apoiar o desenvolvimento de Ambientes de Desenvolvimento Integrado acessíveis a estudantes cegos".*

A condução do estudo se baseou nos objetivos específicos definidos na Seção 1.2 e buscou responder à seguinte questão de pesquisa:

***"Quais estratégias devem ser adotadas para promover a acessibilidade em Ambientes de Desenvolvimento Integrado para estudantes com deficiência visual?"***

É possível afirmar que todos os objetivos específicos estabelecidos foram atingidos, visto que:

- Foram identificados os recursos de Tecnologia Assistiva (TA) utilizados por pessoas com deficiência visual para interagir com sistemas digitais, bem como as principais barreiras e limitações encontradas por eles durante a interação (**OE1**). Os resultados obtidos são apresentados na Seção 2.2.
- As principais barreiras e limitações enfrentadas por estudantes cegos ao aprender a programar e executar tarefas relacionadas à programação de computadores (**OE2**), bem como ao interagir com IDEs (**OE3**), são detalhadas no Capítulo 4;
- Um conjunto preliminar de recomendações para aprimorar a acessibilidade de IDEs para estudantes com deficiência visual (**OE4**) foi elaborado. Os detalhes são apresentados no Capítulo 5;
- As recomendações foram validadas com diferentes especialistas do domínio da aplicação: estudantes de computação, programadores cegos, analista de acessibilidade e desenvolvedor de IDE (**OE5**). Os procedimentos e resultados são descritos nos Capítulos 6 e 7;
- O conjunto de diretrizes foi refinado e resultou na elaboração das Diretrizes de Acessibilidade em Ambientes de Desenvolvimento Integrado (IDEs) para estudantes cegos (**OE6**), apresentadas no Capítulo 8.

Os resultados obtidos em cada etapa desta pesquisa, juntamente com as avaliações conduzidas com especialistas no domínio da aplicação, demonstraram que o objetivo desta Tese de Doutorado foi alcançado.

A seguir, são detalhadas as principais contribuições da pesquisa realizada, bem como restrições enfrentadas ao longo de todo o processo. Além disso, são esboçadas novas perspectivas para trabalhos futuros, visando prosseguir com a análise do tema em questão.

## 9.1 Principais Contribuições

A principal contribuição desta Tese consiste no conjunto de Diretrizes de Acessibilidade em IDEs para estudantes cegos. Essas diretrizes foram elaboradas com o intuito de fornecer orientações claras e práticas para desenvolvedores e pesquisadores que atuam na implementação dessas ferramentas.

A pesquisa ofereceu a oportunidade de examinar as barreiras e desafios enfrentados por estudantes cegos no uso de IDEs e na compreensão de conceitos de programação, abordando uma variedade de perspectivas que incluíram desde estudantes até profissionais e professores. Essa variedade de opiniões e percepções propiciou a obtenção de pontos de vista diversificados.

Foi observado que ainda há uma necessidade significativa de esforços para tornar os IDEs acessíveis às pessoas cegas que dependem de leitores de tela para interação. Além disso, embora diversas normas, recomendações e diretrizes tenham sido publicadas nos últimos anos com o intuito de proporcionar mais acessibilidade dos sistemas interativos, estes documentos especificam recomendações genéricas para construção de sistemas acessíveis, que não abordam aspectos particulares dos IDEs. As limitações relacionadas ao avanço nessa área podem estar associadas ao fato de que a maioria dos pesquisadores, projetistas e desenvolvedores envolvidos diretamente na implementação desses ambientes possuem visão normal, o que pode resultar em uma compreensão limitada das necessidades desses usuários.

Outra constatação que merece destaque é a importância de promover a conscientização entre os professores e fornecer o treinamento adequado para criar um ambiente de aprendizagem inclusivo e eficaz para os estudantes cegos. Compreender as necessidades e preferências desses estudantes é fundamental para selecionar estratégias, recursos e ferramentas mais adequados para o ensino de Programação de Computadores.

Destaca-se também a importância de oferecer formação adequada a todos os profissionais envolvidos na educação desses estudantes. Essa formação poderia oportunizar a aquisição de conhecimento a respeito das tecnologias utilizadas para programar e as possibilidades de configuração dos recursos de Tecnologia Assistiva para que possam efetivamente auxiliá-los. Por sua vez, os estudantes precisam de métodos acessíveis para encontrar recursos que podem ser desconhecidos para eles e que, muitas vezes, nem mesmo sabem que necessitam.

Por outro lado, como o aprendizado de programação é um processo gradual, ocorrendo em etapas, é possível que a percepção da necessidade de recursos de acessibilidade também se desenvolva gradualmente. Os estudantes podem perceber a necessidade de um ou mais recursos à medida que avançam em seu aprendizado. Além disso, alguns recursos podem deixar de ser úteis depois que se adquire mais

experiência com a ferramenta. Portanto, é fundamental que eles possam ser ativados ou desativados conforme as necessidades e preferências de cada usuário.

Por fim, acredita-se que as recomendações contidas no conjunto de diretrizes elaboradas não se limitam a auxiliar apenas os estudantes cegos. Algumas delas têm potencial para contribuir na melhora da interação e na realização de atividades relacionadas à programação de computadores por parte de outros estudantes e até mesmo de desenvolvedores profissionais.

Como resultado do desenvolvimento da pesquisa apresentada nesta Tese de Doutorado, foram publicados os seguintes estudos:

- Artigo apresentado e publicado no *XVIII Brazilian Symposium on Information Systems: Assistive technology to assist the visually impaired in the use of ICTs: A systematic literature review* (2022).
- Artigo publicado no *iSys - Brazilian Journal of Information Systems: Tecnologia Assistiva para auxiliar a interação entre pessoas com deficiência visual e sistemas computacionais: Um Mapeamento Sistemático da Literatura* (2023).
- Artigo apresentado e publicado nos Anais do XXII Simpósio Brasileiro sobre Fatores Humanos em Sistemas Computacionais (IHC): *Understanding the Accessibility Barriers Faced by Learners with Visual Impairments in Computer Programming* (2023).
- Artigo apresentado e publicado nos Anais do *Workshop* de Teses e Dissertações do XXII Simpósio Brasileiro sobre Fatores Humanos em Sistemas Computacionais (WTD-IHC): *Estratégias de Acessibilidade em IDEs para Pessoas com Deficiência Visual* (2023).
- Artigo apresentado e publicado nos Anais do Simpósio Brasileiro de Informática na Educação (SBIE): *Experiências Educacionais em Disciplinas de Programação de Computadores: uma Análise Qualitativa na Perspectiva dos Estudantes com Deficiência Visual* (2023).
- Artigo apresentado e publicado nos Anais Estendidos do XII Congresso Brasileiro de Informática na Educação (STUDX-CBIE): *Estratégias de Acessibilidade em IDEs para Estudantes com Deficiência Visual* (2023).
- Artigo apresentado e publicado nos Anais do Encontro Anual de Tecnologia da Informação (EATI): *Normas e Diretrizes de Acessibilidade para Sistemas Digitais: um Levantamento das Principais Recomendações para Usuários com Deficiência Visual* (2023).

- Artigo apresentado e publicado nos Anais do Encontro de Pós-Graduação (EN-POS) da 9ª Semana Integrada UFPEL (SIIPE): Estratégias de Acessibilidade em IDEs para estudantes com Deficiência Visual (2023).
- Artigo publicado na Revista Novas Tecnologias na Educação (RENOTE): Desafios e Percepções sobre Acessibilidade em Ambientes de Desenvolvimento Integrado (2023).
- Submissão de versão estendida do artigo *Understanding the Accessibility Barriers Faced by Learners with Visual Impairments in Computer Programming* no *SBC Journal on Interactive Systems (JIS)* (2024).

## 9.2 Limitações da Pesquisa

Durante os quatro anos de duração desta pesquisa, várias limitações surgiram. A primeira restrição está associada ao fato de que os dois primeiros anos foram conduzidos em um cenário marcado pela pandemia de COVID-19. A necessidade de distanciamento social imposta neste período tornou inviável o deslocamento e o contato presencial. Essa situação ressaltou a importância da adaptação e da flexibilidade nas abordagens de pesquisa, demonstrando a necessidade de explorar métodos remotos e virtuais para a coleta de dados e a interação com os participantes.

Também foram encontradas limitações relativas ao tempo disponível para condução do estudo e à dificuldade de recrutamento de participantes, ressaltando a necessidade de conduzir novas avaliações. Seria valioso contar com a participação de um maior número de especialistas no domínio da aplicação, bem como de profissionais especializados na realização de avaliações heurísticas e testes com usuários. Isso garantiria maior clareza e a consistência do texto das diretrizes.

Além disso, as barreiras e desafios identificados foram coletados com a participação de um número pequeno de estudantes, professores e profissionais que atuam na área de desenvolvimento de software (com e sem deficiência visual) e, ainda, com base em uma revisão da literatura *ad-hoc*. Seria importante realizar entrevistas ou questionários com um número maior de pessoas com esses perfis e outros profissionais envolvidos no projeto e implementação de IDEs.

## 9.3 Direcionamento para Trabalhos Futuros

Algumas perspectivas de pesquisas futuras podem ser vislumbradas para dar continuidade ao estudo conduzido. Novos estudos qualitativos poderiam ampliar a compreensão dos desafios e barreiras enfrentados por estudantes cegos ao utilizarem IDEs, fornecendo novos requisitos de acessibilidade e também de usabilidade relacionados ao projeto desses sistemas.

A execução de etapas de validação adicionais, contando com a participação de especialistas em avaliação heurística e desenvolvedores de IDEs, também contribuiria para aprimorar o conteúdo das diretrizes.

Melhorias na apresentação e disponibilização tanto das Diretrizes quanto da Cartilha também se fazem necessárias, uma vez que atualmente elas estão disponibilizadas por meio de um documento estático hospedado no GitHub<sup>1</sup>. A criação de um *website* dedicado poderia proporcionar melhor acesso aos professores, analistas, desenvolvedores e pesquisadores interessados.

Especificamente em relação à cartilha, a realização de uma Revisão Sistemática da Literatura possibilitaria a identificação das experiências de outros educadores no ensino de programação de computadores para estudantes cegos. Os resultados poderiam ser aproveitados para criar uma seção que oferecesse exemplos práticos e estratégias úteis. Também seria interessante criar uma forma de permitir que outras pessoas possam contribuir com a atualização constante do conteúdo da cartilha, compartilhando experiências e recursos que possam ser adotados por professores e outros profissionais para aprimorar o processo de ensino-aprendizagem.

Também seria interessante verificar que tipos de alertas sonoros são mais adequados para comunicar quais tipos de informação e se as recomendações relacionadas à pronúncia do código-fonte impactariam de maneira positiva na aprendizagem dos estudantes.

Por fim, novos estudos que abordem a acessibilidade de funcionalidades e recursos especializados oferecidos pelos IDEs, como as ferramentas de controle de versionamento, são necessários e importantes para garantir a acessibilidade.

---

<sup>1</sup><https://github.com/>

## REFERÊNCIAS

ABASCAL, J.; NICOLLE, C. Moving towards inclusive design guidelines for socially and ethically aware HCI. **Interacting with computers**, [S.l.], v.17, n.5, p.484–505, 2005.

ABREU, S. et al. Usability evaluation of a resource to read mathematical formulae in a screen reader for people with visual disabilities. In: BRAZILIAN SYMPOSIUM ON HUMAN FACTORS IN COMPUTING SYSTEMS, 18., 2019. **Proceedings...** [S.l.: s.n.], 2019. p.1–11.

ALBUSAYS, K. L. **The Role of Sonification as a Code Navigation Aid**: Improving Programming Structure Readability and Understandability For Non-Visual Users. [S.l.]: Rochester Institute of Technology, 2020.

ALBUSAYS, K.; LUDI, S. Eliciting programming challenges faced by developers with visual impairments: exploratory study. In: INTERNATIONAL WORKSHOP ON COOPERATIVE AND HUMAN ASPECTS OF SOFTWARE ENGINEERING, 9., 2016. **Proceedings...** [S.l.: s.n.], 2016. p.82–85.

ALBUSAYS, K.; LUDI, S.; HUENERFAUTH, M. Interviews and observation of blind software developers at work to understand code navigation challenges. In: INTERNATIONAL ACM SIGACCESS CONFERENCE ON COMPUTERS AND ACCESSIBILITY, 19., 2017. **Proceedings...** [S.l.: s.n.], 2017. p.91–100.

ALOTAIBI, H.; AL-KHALIFA, H. S.; ALSAEED, D. Teaching Programming to Students with Vision Impairment: Impact of Tactile Teaching Strategies on Student's Achievements and Perceptions. **Sustainability 12, no. 13: 5320**, [S.l.], n.13: 5320, 2020.

ALVES, E. A.; CARDOSO, P. C. F.; FREIRE, A. P. Automatically Generated Summaries as In-Page Web Navigation Accelerators for Blind Users. In: BRAZILIAN SYMPOSIUM ON HUMAN FACTORS IN COMPUTING SYSTEMS, 17., 2018. **Proceedings...** [S.l.: s.n.], 2018. p.1–8.

ALVES, L. F. et al. Estudantes com Deficiência Visual em Computação: participação,

perspectivas e desafios enfrentados. In: II SIMPÓSIO BRASILEIRO DE EDUCAÇÃO EM COMPUTAÇÃO, 2022. **Anais...** [S.l.: s.n.], 2022. p.67–76.

APPLE, C. B. L. **Visão para todos os pontos de vista**. (Accessed on 06/14/2022), <https://www.apple.com/br/accessibility/vision/>.

ARGYROPOULOS, V.; PAVELI, A.; NIKOLARAIZI, M. The role of DAISY digital talking books in the education of individuals with blindness: A pilot study. **Education and Information Technologies**, [S.l.], v.24, n.1, p.693–709, 2019.

ARMALY, A.; RODEGHERO, P.; MCMILLAN, C. Audiohighlight: Code skimming for blind programmers. In: IEEE INTERNATIONAL CONFERENCE ON SOFTWARE MAINTENANCE AND EVOLUTION (ICSME), 2018., 2018. **Anais...** [S.l.: s.n.], 2018. p.206–216.

AYDIN, A. S. et al. Sail: Saliency-driven injection of aria landmarks. In: INTERNATIONAL CONFERENCE ON INTELLIGENT USER INTERFACES, 25., 2020. **Proceedings...** [S.l.: s.n.], 2020. p.111–115.

BAKER, C. M.; BENNETT, C. L.; LADNER, R. E. Educational experiences of blind programmers. In: ACM TECHNICAL SYMPOSIUM ON COMPUTER SCIENCE EDUCATION, 50., 2019. **Proceedings...** [S.l.: s.n.], 2019. p.759–765.

BAKER, C. M.; MILNE, L. R.; LADNER, R. E. Structjumper: A tool to help blind programmers navigate and understand the structure of code. In: ANNUAL ACM CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS, 33., 2015. **Proceedings...** [S.l.: s.n.], 2015. p.3043–3052.

BARBOSA, S.; SILVA, B. **Interação humano-computador**. [S.l.]: Elsevier Brasil, 2010.

BARDIN, L. **Análise de Conteúdo**. [S.l.]: Edições 70, 2015.

BEDA, J. S. et al. Acessibilidade sob Desenho Universal para Ambientes Virtuais de Aprendizagem: um Mapeamento de Critérios e Desafios. In: XXXIII SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 2022. **Anais...** [S.l.: s.n.], 2022. p.680–693.

BERSCH, R. Introdução à tecnologia assistiva. **Porto Alegre: CEDI**, [S.l.], v.21, 2008.

BRASIL. DECRETO Nº 5.296, DE 2 DE DEZEMBRO DE 2004. **Diretrizes Curriculares Nacionais para a Educação Profissional Técnica de Nível Médio**. Ministério da Educação, Brasília, DF, [S.l.], 2004.

BRASIL, C. C. Lei Nº 13.146, de 6 de julho 2015. **Institui a lei brasileira de inclusão da pessoa com deficiência (estatuto da pessoa com deficiência)**. Brasília, [S.l.], 2015.

BRASIL, P. d. R. Lei Nº 9.394, de 20 de dezembro de 1996. **Lei de Diretrizes e Bases da Educação**, [S.l.], 1996. (Accessed on 06/01/2023).

BRITTO, T. C. P.; PIZZOLATO, E. B. GAIA: uma proposta de um guia de recomendações de acessibilidade de interfaces Web com foco em aspectos do Autismo. **Revista Brasileira de Informática na Educação**, [S.l.], v.26, n.2, 2018.

BURTON, M. J. et al. The Lancet global health Commission on global eye health: vision beyond 2020. **The Lancet Global Health**, [S.l.], v.9, n.4, p.e489–e551, 2021.

CAT, C. d. A. T. Tecnologia assistiva. **Brasília: Corde**, [S.l.], 2009.

DIAS, J. d. L.; DIAS, M. d. L. O. Os leitores de tela como ferramenta de acessibilidade e inclusão da pessoa com Deficiência Visual. **Brazilian Journal of Development**, [S.l.], v.5, n.12, p.28869–28878, 2019.

EPI, E. P. I. **EF EPI | O Índice de Proficiência em Inglês da EF | EF Brasil**. (Accessed on 02/12/2024), <https://www.ef.com.br/epi/>.

FREEDOMSCIENTIFIC. **JAWS® – Job Access With Speech**. (Accessed on 06/14/2022), <https://www.freedomscientific.com/products/software/jaws/>.

GERALDO, R. J. **Um auxílio à navegação acessível na web para usuários cegos**. 2016. Tese (Doutorado em Ciência da Computação) — São Carlos (SP): Teses de Doutorado em Ciências da Computação e Matemática Computacional da Universidade de São Paulo.

GERHARDT, T. E.; SILVEIRA, D. T. Métodos de pesquisa. Universidade Aberta do Brasil–UAB/UFRGS. **Porto Alegre: Editora da UFRGS**, [S.l.], p.120, 2009.

GIBBS, G. **Análise de dados qualitativos**: coleção pesquisa qualitativa. [S.l.]: Bookman Editora, 2009.

GIL, A. C. et al. **Como elaborar projetos de pesquisa**. [S.l.]: Atlas São Paulo, 2002. v.4.

GNOMEHELP. **Orca Screen Reader**. (Accessed on 06/14/2022), [https://help.gnome.org/users/orca/stable/index.html.pt\\_BR](https://help.gnome.org/users/orca/stable/index.html.pt_BR).

GOMES, M. et al. Um estudo sobre erros em programação-Reconhecendo as dificuldades de programadores iniciantes. In: WORKSHOPS DO CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 2015. **Anais...** [S.l.: s.n.], 2015. v.4, n.1, p.1398.

GOOGLE. **TalkBack - Ajuda do Acessibilidade no Android**. (Accessed on 06/14/2022), [https://support.google.com/accessibility/android/topic/3529932?hl=pt-BR&ref\\_topic=9078845](https://support.google.com/accessibility/android/topic/3529932?hl=pt-BR&ref_topic=9078845).

GUERIN, C.; JAYATILAKA, A.; RANASINGHE, D. Why start a higher degree by research? An exploratory factor analysis of motivations to undertake doctoral studies. **Higher Education Research & Development**, [S.l.], v.34, n.1, p.89–104, 2015.

GUIMARÃES, Í. J. B. et al. Acessibilidade em websites de comércio eletrônico: avaliação através da interação com usuários cegos. , [S.l.], 2016.

Hadwen-Bennett, A.; SENTANCE, S.; MORRISON, C. Making programming accessible to learners with visual impairments: a literature review. **International Journal of Computer Science Education in Schools**, [S.l.], v.2, n.2, p.3–13, 2018.

HERMANS, F.; ALDEWERELD, M. Programming is writing is programming. In: COMPANION TO THE FIRST INTERNATIONAL CONFERENCE ON THE ART, SCIENCE AND ENGINEERING OF PROGRAMMING, 2017. **Anais...** [S.l.: s.n.], 2017. p.1–8.

HUFF, E. W. et al. Examining the work experience of programmers with visual impairments. In: 2020 IEEE INTERNATIONAL CONFERENCE ON SOFTWARE MAINTENANCE AND EVOLUTION (ICSME), 2020. **Anais...** [S.l.: s.n.], 2020. p.707–711.

HUTCHINSON, J.; METATLA, O. An initial investigation into non-visual code structure overview through speech, non-speech and spearcons. In: EXTENDED ABSTRACTS OF THE 2018 CHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS, 2018. **Anais...** [S.l.: s.n.], 2018. p.1–6.

INEP. **Censo da Educação Superior 2021**. [S.l.]: Ministério da Educação, Diretoria de Estatísticas Educacionais, Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira, 2022. (Accessed on 05/08/2023).

INTERVOX. **Projeto DOSVOX**. (Accessed on 06/14/2022), <http://intervox.nce.ufrj.br/dosvox/>.

ISMAILI, J. et al. Mobile learning as alternative to assistive technology devices for special needs students. **Education and Information Technologies**, [S.l.], v.22, n.3, p.883–899, 2017.

ISO 9241-171. ISO/IEC 9241. Ergonomia da interação humano-sistema. PArte 171: Orientações sobre acessibilidade de software. , [S.l.], 2018.

JARIWALA, A.; MARGHITU, D.; CHAPMAN, R. MyA+ Math: Teaching Math to Students with Vision Impairment. In: INTERNATIONAL CONFERENCE ON HUMAN-COMPUTER INTERACTION, 2020. **Anais...** [S.l.: s.n.], 2020. p.200–211.

KEARNEY-VOLPE, C.; HURST, A. Accessible web development: Opportunities to improve the education and practice of web development with a screen reader. **ACM Transactions on Accessible Computing (TACCESS)**, [S.l.], v.14, n.2, p.1–32, 2021.

KHAN, A.; KHUSRO, S. Blind-friendly user interfaces—a pilot study on improving the accessibility of touchscreen interfaces. **Multimedia Tools and Applications**, [S.l.], v.78, n.13, p.17495–17519, 2019.

KHAN, A. et al. TetraMail: a usable email client for blind people. **Universal Access in the Information Society**, [S.l.], v.19, n.1, p.113–132, 2020.

KHURANA, R. et al. Nonvisual interaction techniques at the keyboard surface. In: CHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS, 2018., 2018. **Proceedings...** [S.l.: s.n.], 2018. p.1–12.

KIND, L. Notas para o trabalho com a técnica de grupos focais. **Psicologia em revista**, [S.l.], v.10, n.15, p.124–138, 2004.

LAHTINEN, E.; ALA-MUTKA, K.; JÄRVINEN, H.-M. A study of the difficulties of novice programmers. **Acm sigcse bulletin**, [S.l.], v.37, n.3, p.14–18, 2005.

LEONARDIS, D.; CLAUDIO, L.; FRISOLI, A. A survey on innovative refreshable braille display technologies. In: INTERNATIONAL CONFERENCE ON APPLIED HUMAN FACTORS AND ERGONOMICS, 2017. **Anais...** [S.l.: s.n.], 2017. p.488–498.

LEPORINI, B.; PATERNÒ, F. Increasing usability when interacting through screen readers. **Universal access in the information society**, [S.l.], v.3, n.1, p.57–70, 2004.

LUDI, S. Position paper: Towards making block-based programming accessible for blind users. In: IEEE BLOCKS AND BEYOND WORKSHOP (BLOCKS AND BEYOND), 2015., 2015. **Anais...** [S.l.: s.n.], 2015. p.67–69.

LUDI, S.; SPENCER, M. Design Considerations to Increase Block-based Language Accessibility for Blind Programmers Via Blockly. **J. Vis. Lang. Sentient Syst.**, [S.l.], v.3, n.1, p.119–124, 2017.

MATTHEISS, E. et al. User-centred design with visually impaired pupils: A case study of a game editor for orientation and mobility training. **International Journal of Child-Computer Interaction**, [S.l.], v.11, p.12–18, 2017.

MEALIN, S.; MURPHY-HILL, E. An exploratory study of blind software developers. In: 2012 IEEE SYMPOSIUM ON VISUAL LANGUAGES AND HUMAN-CENTRIC COMPUTING (VL/HCC), 2012. **Anais...** [S.l.: s.n.], 2012. p.71–74.

MENDONÇA, A. et al. **Alunos cegos e com baixa visão. Orientações curriculares**. [S.l.]: DGIDC/DSEEASE, 2008.

MICROPOWER. **Virtual Vision - Acessibilidade para pessoas com deficiência visual**. (Accessed on 06/14/2022), <https://micropowerglobal.com/solucoes/virtual-vision/>.

MICROSOFT. **Guia completo do Narrador - Suporte da Microsoft**. (Accessed on 02/10/2024), [https://support.microsoft.com/pt-br/windows/guia-completo-do-narrador-e4397a0d-ef4f-b386-d8ae-c172f109bdb1#WindowsVersion=Windows\\_10](https://support.microsoft.com/pt-br/windows/guia-completo-do-narrador-e4397a0d-ef4f-b386-d8ae-c172f109bdb1#WindowsVersion=Windows_10).

MILNE, L. R.; LADNER, R. E. Blocks4All: overcoming accessibility barriers to blocks programming for children with visual impairments. In: CHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS, 2018., 2018. **Proceedings...** [S.l.: s.n.], 2018. p.1–10.

MOUNTAPMBEME, A.; LUDI, S. Investigating challenges faced by learners with visual impairments using block-based programming/hybrid environments. In: INTERNATIONAL ACM SIGACCESS CONFERENCE ON COMPUTERS AND ACCESSIBILITY, 22., 2020. **Proceedings...** [S.l.: s.n.], 2020. p.1–4.

MOUNTAPMBEME, A.; OKAFOR, O.; LUDI, S. Addressing Accessibility Barriers in Programming for People with Visual Impairments: A Literature Review. **ACM Transactions on Accessible Computing (TACCESS)**, [S.l.], v.15, n.1, p.1–26, 2022.

MUNIANDY, M.; SULAIMAN, S. Touch sensation as part of multimedia design elements to improve computer accessibility for the blind users. In: INTERNATIONAL CONFERENCE ON RESEARCH AND INNOVATION IN INFORMATION SYSTEMS (ICRIIS), 2017., 2017. **Anais...** [S.l.: s.n.], 2017. p.1–6.

MURILLO-MORALES, T.; MIESENBERGER, K. AUDiaL: A Natural Language Interface to Make Statistical Charts Accessible to Blind Persons. In: INTERNATIONAL CONFERENCE ON COMPUTERS HELPING PEOPLE WITH SPECIAL NEEDS, 2020. **Anais...** [S.l.: s.n.], 2020. p.373–384.

MURPHY, G. C. Beyond integrated development environments: adding context to software development. In: IEEE/ACM 41ST INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING: NEW IDEAS AND EMERGING RESULTS (ICSE-NIER), 2019., 2019. **Anais...** [S.l.: s.n.], 2019. p.73–76.

MWPT. **Número de sites brasileiros aprovados em todos os testes de acessibilidade mantém crescimento, mas ainda é menos de 1% do total - WPT.** (Accessed on 02/13/2024), <https://mwpt.com.br/numero-de-sites-brasileiros-aprovados-em-todos-os-testes-de-acessibilidade-mantem->

NASR, E.; ALSAGGAF, W.; SINNARI, D. Developing Usability Guidelines for mHealth Applications (UGmHA). **Multimodal Technologies and Interaction**, [S.l.], v.7, n.3, p.26, 2023.

NICOLLE, C.; ABASCAL, J. **Inclusive design guidelines for HCI.** [S.l.]: CRC Press, 2001.

NOSCHANG, L. F. et al. Portugol studio: Uma ide para iniciantes em programação. In: XXII WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO, 2014. **Anais...** [S.l.: s.n.], 2014. p.1–10.

NVDA. **NVDA, the free and open source Screen Reader for Microsoft Windows.** (Accessed on 06/14/2022), <https://github.com/nvaccess/nvda>.

ONU, O. d. N. U. **Objetivos de Desenvolvimento Sustentável.** (Accessed on 01/02/2024), <https://brasil.un.org/pt-br/sdgs>.

PACIELLO, M. **Web accessibility for people with disabilities.** [S.l.]: Crc Press, 2000.

PANDEY, M. et al. Accessibility of UI Frameworks and Libraries for Programmers with Visual Impairments. In: IEEE SYMPOSIUM ON VISUAL LANGUAGES AND HUMAN-CENTRIC COMPUTING (VL/HCC), 2022., 2022. **Anais...** [S.l.: s.n.], 2022. p.1–10.

PETRAUSCH, V.; LOITSCH, C. Accessibility analysis of the eclipse ide for users with visual impairment. In: **Harnessing the Power of Technology to Improve Lives.** [S.l.]: IOS Press, 2017. p.922–929.

PNAD. **Pesquisa Nacional por Amostra de Domicílios Contínua.** (Accessed on 02/12/2024), <https://www.ibge.gov.br/estatisticas/sociais/trabalho/17270-pnad-continua.html>.

PORTE, M. d. S.; TRINDADE, J. D. R. Barreiras tecnológicas: um fator limitador na acessibilidade das pessoas com deficiência. **Texto Livre**, [S.l.], v.14, 2022.

POTLURI, V.; HE, L.; CHEN, C.; FROEHLICH, J. E.; MANKOFF, J. A multi-modal approach for blind and visually impaired developers to edit webpage designs. In: INTERNATIONAL ACM SIGACCESS CONFERENCE ON COMPUTERS AND ACCESSIBILITY, 21., 2019. **Proceedings...** [S.l.: s.n.], 2019. p.612–614.

POTLURI, V. et al. Codetalk: Improving programming environment accessibility for visually impaired developers. In: OF THE 2018 CHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS, 2018. **Proceedings...** [S.l.: s.n.], 2018. p.1–11.

QUIÑONES, D.; RUSU, C.; RUSU, V. A methodology to develop usability/user experience heuristics. **Computer standards & interfaces**, [S.l.], v.59, p.109–129, 2018.

RAJASELVI, M. et al. A Survey of Programming Editors for the Visually Impaired. **Accessed: Aug**, [S.l.], v.12, 2021.

SALES, I. H.; TORRES, J. P. Inclusão de estudantes com deficiência visual em uma Universidade Federal Mineira. **Revista Educação Especial**, [S.l.], p.e20–1, 2022.

SAMPATH, H.; MERRICK, A.; MACVEAN, A. Accessibility of command line interfaces. In: CHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS, 2021., 2021. **Proceedings...** [S.l.: s.n.], 2021. p.1–10.

SÁNCHEZ, J.; AGUAYO, F. APL: Audio programming language for blind learners. In: INTERNATIONAL CONFERENCE ON COMPUTERS FOR HANDICAPPED PERSONS, 2006. **Anais...** [S.l.: s.n.], 2006. p.1334–1341.

SCHANZER, E.; BAHRAM, S.; KRISHNAMURTHI, S. Accessible AST-based programming for visually-impaired programmers. In: ACM TECHNICAL SYMPOSIUM ON COMPUTER SCIENCE EDUCATION, 50., 2019. **Proceedings...** [S.l.: s.n.], 2019. p.773–779.

SHARMA, M. R. A Short Communication on Computer Programming Languages in Modern Era. , [S.l.], 2020.

SIEGFRIED, R. M. Visual programming and the blind: the challenge and the opportunity. **ACM SIGCSE Bulletin**, [S.l.], v.38, n.1, p.275–278, 2006.

SILVA, C. F. da; FERREIRA, S. B. L.; RAMOS, J. F. M. WhatsApp accessibility from the perspective of visually impaired people. In: BRAZILIAN SYMPOSIUM ON HUMAN FACTORS IN COMPUTING SYSTEMS, 15., 2016. **Proceedings...** [S.l.: s.n.], 2016. p.1–10.

SMITH, A. C. et al. Nonvisual tool for navigating hierarchical structures. **ACM SIGACCESS Accessibility and Computing**, [S.l.], n.77-78, p.133–139, 2003.

STEFIK, A.; HUNDHAUSEN, C.; PATTERSON, R. An empirical investigation into the design of auditory cues to enhance computer program comprehension. **International Journal of Human-Computer Studies**, [S.l.], v.69, n.12, p.820–838, 2011.

STEFIK, A. et al. Sodbeans. In: IEEE 17TH INTERNATIONAL CONFERENCE ON PROGRAM COMPREHENSION, 2009., 2009. **Anais...** [S.l.: s.n.], 2009. p.293–294.

THIOLLENT, M. **Metodologia da pesquisa-ação**. [S.l.]: Cortez editora, 2022.

TORRES, J. P.; COSTA, C. S. L. d.; LOURENÇO, G. F. Substituição sensorial visuo-tátil e visuo-auditiva em pessoas com deficiência visual: uma revisão sistemática. **Revista Brasileira de Educação Especial**, [S.l.], v.22, p.605–618, 2016.

UTRERAS, E.; PONTELLI, E. Accessibility of block-based introductory programming languages and a tangible programming tool prototype. In: INTERNATIONAL CONFERENCE ON COMPUTERS HELPING PEOPLE WITH SPECIAL NEEDS, 2020. **Anais...** [S.l.: s.n.], 2020. p.27–34.

VANUKURU, R. Accessible Spatial Audio Interfaces: A Pilot Study into Screen Readers with Concurrent Speech. In: EXTENDED ABSTRACTS OF THE 2020 CHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS, 2020. **Anais...** [S.l.: s.n.], 2020. p.1–6.

VELÁZQUEZ, R.; SÁNCHEZ, C. N.; PISSALOUX, E. E. Visual impairment simulator based on the Hadamard product. **Electronic Notes in Theoretical Computer Science**, [S.l.], v.329, p.169–179, 2016.

WCAG 2.2, W. W. W. C. Web Content Accessibility Guidelines (WCAG) 2.2. , [S.l.], 2023. (Accessed on 01/03/2023).

WHO, Q. H. O. **Disability**. (Accessed on 02/12/2024), [https://www.who.int/health-topics/disability#tab=tab\\_1](https://www.who.int/health-topics/disability#tab=tab_1).

WHO, W. H. O. **Up to 45 million blind people globally - and growing**. (Accessed on 01/02/2024), <https://www.who.int/news/item/09-10-2003-up-to-45-million-blind-people-globally---and-growing>.

ZEN, E.; COSTA, V. K. da; TAVARES, T. A. Understanding the Accessibility Barriers Faced by Learners with Visual Impairments in Computer Programming. In: XXII SIMPÓSIO BRASILEIRO SOBRE FATORES HUMANOS EM SISTEMAS COMPUTACIONAIS, 2023. **Anais...** [S.l.: s.n.], 2023.

ZEN, E. et al. Assistive Technology to Assist the Visually Impaired in the Use of ICTs: A Systematic Literature Review. In: XVIII BRAZILIAN SYMPOSIUM ON INFORMATION SYSTEMS, 2022. **Anais...** [S.l.: s.n.], 2022. p.1–8.

ZEN, E. et al. Tecnologia Assistiva para auxiliar a interação entre pessoas com deficiência visual e sistemas computacionais: Um Mapeamento Sistemático da Literatura. **iSys - Brazilian Journal of Information Systems**, [S.l.], v.16, n.1, p.6:1–6:27, May 2023.

## **Apêndices**

APÊNDICE A – Parecer Consubstanciado n.º 54297421.9.0000.5574

**PARECER CONSUBSTANCIADO DO CEP**

**DADOS DO PROJETO DE PESQUISA**

**Título da Pesquisa:** Levantamento de dificuldades de utilização de ferramentas de desenvolvimento de software por estudantes deficientes visuais nas disciplinas relacionadas à Programação de Computadores.

**Pesquisador:** ELIANA ZEN

**Área Temática:**

**Versão:** 3

**CAAE:** 54297421.9.0000.5574

**Instituição Proponente:** INSTITUTO FEDERAL DE EDUCACAO, CIENCIA E TECNOLOGIA

**Patrocinador Principal:** Financiamento Próprio

**DADOS DO PARECER**

**Número do Parecer:** 5.250.149

**Apresentação do Projeto:**

O projeto investigará as dificuldades e soluções nos processos de ensino e aprendizagem de programação para estudantes e egressos dos cursos técnicos e superiores que possuem deficiência visual. Pretende-se compreender os vieses dos estudantes/egressos Cursos do Eixo Informação e Comunicação do Instituto Federal Farroupilha e de professores de todo o Brasil que já tenham lecionado ou estejam lecionando disciplinas relacionadas a programação de computadores para estudantes deficientes visuais.

**Objetivo da Pesquisa:**

O objetivo primário deste projeto é identificar as principais dificuldades encontradas por estudantes deficientes visuais nas disciplinas de programação de computadores, em cursos técnicos e superiores, no uso de tecnologia assistiva e ferramentas voltadas ao desenvolvimento de software.

**Avaliação dos Riscos e Benefícios:**

Descreve adequadamente os riscos e benefícios.

**Comentários e Considerações sobre a Pesquisa:**

Pesquisa adequadamente fundamentada.

**Endereço:** Alameda Santiago do Chile, 195

**Bairro:** Nossa Sra. das Dores

**CEP:** 97.050-685

**UF:** RS

**Município:** SANTA MARIA

**Telefone:** (55)3218-9800

**E-mail:** cep@iffarroupilha.edu.br

Continuação do Parecer: 5.250.149

**Considerações sobre os Termos de apresentação obrigatória:**

Todos os termos devem ser paginados;

**Conclusões ou Pendências e Lista de Inadequações:**

- Pagar todos os termos de consentimento, assentimento, confidencialidade anexados;

**Considerações Finais a critério do CEP:**

O CEP acata o parecer do(a) relator(a).

**Este parecer foi elaborado baseado nos documentos abaixo relacionados:**

Tipo Documento	Arquivo	Postagem	Autor	Situação
Informações Básicas do Projeto	PB_INFORMAÇÕES_BÁSICAS_DO_PROJETO_1871720.pdf	18/01/2022 09:42:02		Aceito
Projeto Detalhado / Brochura Investigador	ProjetoPesquisa.pdf	18/01/2022 09:41:19	ELIANA ZEN	Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	5_TCLE_PaisResponsaveis_Estudantes Menores18anos.pdf	07/01/2022 09:30:19	ELIANA ZEN	Aceito
Outros	2_TermoConfidencialidade.pdf	22/12/2021 10:26:20	ELIANA ZEN	Aceito
Outros	QuestionarioDocentes.pdf	22/12/2021 10:25:50	ELIANA ZEN	Aceito
Outros	ROTEIRO_Entrevista_Estudantes.pdf	22/12/2021 10:23:12	ELIANA ZEN	Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	4_TCLE_Estudantes_Maiores18anos.pdf	22/12/2021 10:22:13	ELIANA ZEN	Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	6_Termo_Uso_Imagem_Voz.pdf	22/12/2021 10:22:02	ELIANA ZEN	Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	3_TCLE_Docentes.pdf	22/12/2021 10:21:49	ELIANA ZEN	Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	1_TermoAssentimento_Estudantes_Menores18anos.pdf	22/12/2021 10:21:34	ELIANA ZEN	Aceito
Folha de Rosto	FolhaDeRosto.pdf	09/12/2021	ELIANA ZEN	Aceito

**Endereço:** Alameda Santiago do Chile, 195

**Bairro:** Nossa Sra. das Dores

**CEP:** 97.050-685

**UF:** RS

**Município:** SANTA MARIA

**Telefone:** (55)3218-9800

**E-mail:** cep@iffarroupilha.edu.br

INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E  
TECNOLOGIA FARROUPILHA



Continuação do Parecer: 5.250.149

Folha de Rosto	FolhaDeRosto.pdf	09:30:43	ELIANA ZEN	Aceito
Outros	CartaAnuencia.pdf	06/12/2021 09:49:19	ELIANA ZEN	Aceito

**Situação do Parecer:**

Aprovado

**Necessita Apreciação da CONEP:**

Não

SANTA MARIA, 18 de Fevereiro de 2022

---

**Assinado por:**  
**RACHEL DOS SANTOS MARQUES**  
**(Coordenador(a))**

**Endereço:** Alameda Santiago do Chile, 195

**Bairro:** Nossa Sra. das Dores

**CEP:** 97.050-685

**UF:** RS **Município:** SANTA MARIA

**Telefone:** (55)3218-9800

**E-mail:** cep@iffarroupilha.edu.br

APÊNDICE B – Parecer Consubstanciado n.º 68361023.1.0000.5574

**PARECER CONSUBSTANCIADO DO CEP**

**DADOS DO PROJETO DE PESQUISA**

**Título da Pesquisa:** ELABORAÇÃO DE DIRETRIZES DE ACESSIBILIDADE PARA AMBIENTES DE DESENVOLVIMENTO DE SOFTWARE VOLTADAS A USUÁRIOS COM DEFICIÊNCIA VISUAL

**Pesquisador:** ELIANA ZEN

**Área Temática:**

**Versão:** 2

**CAAE:** 68361023.1.0000.5574

**Instituição Proponente:** INSTITUTO FEDERAL DE EDUCACAO, CIENCIA E TECNOLOGIA

**Patrocinador Principal:** Financiamento Próprio

**DADOS DO PARECER**

**Número do Parecer:** 6.072.734

**Apresentação do Projeto:**

Este projeto tem como objetivo principal desenvolver um conjunto de diretrizes de acessibilidade para garantir que pessoas cegas tenham acesso a todos os recursos disponibilizados pelas ferramentas e ambientes de desenvolvimento de software. A pesquisa será realizada em parceria entre o Instituto Federal Farroupilha e o Programa de Pós-Graduação em Computação da Universidade Federal de Pelotas.

A hipótese é que as limitações das ferramentas de desenvolvimento de software em conjunto com tecnologias assistivas podem comprometer o acesso completo desses recursos por pessoas com deficiência visual. A elaboração de diretrizes específicas para essas ferramentas e ambientes de desenvolvimento pode contribuir para uma maior acessibilidade.

**Objetivo da Pesquisa:**

Objetivo Primário:

Partindo do pressuposto de que as atuais diretrizes de acessibilidade não são suficientes para abordar as peculiaridades inerentes às ferramentas e ambientes de desenvolvimento de software, especialmente no que diz respeito às barreiras de acessibilidade impostas aos usuários cegos, o objetivo principal deste projeto de pesquisa é elaborar um conjunto de diretrizes de acessibilidade para garantir que pessoas cegas tenham acesso a todos os recursos disponibilizados pelas

**Endereço:** Alameda Santiago do Chile, 195

**Bairro:** Nossa Sra. das Dores

**UF:** RS

**Município:** SANTA MARIA

**CEP:** 97.050-685

**Telefone:** (55)3218-9800

**E-mail:** cep@iffarroupilha.edu.br

Continuação do Parecer: 6.072.734

ferramentas e ambientes de desenvolvimento de software.

Objetivo Secundário:

Para atender o objetivo principal, foram estabelecidos os seguintes objetivos específicos: • Realizar levantamento e análise de Normas e Diretrizes de Acessibilidade existentes no cenário nacional e internacional; • Analisar a acessibilidade das IDEs mais utilizados por estudantes com deficiência visual de cursos da área de Computação (levantamento já realizado em 2022); • Elaborar elementos que contemplem as Diretrizes de Acessibilidade para ferramentas e ambientes de desenvolvimento de software para usuários cegos; • Implementar um plug-in para o leitor de telas NVDA (NonVisual Desktop Access) que implemente as Diretrizes propostas; • Validar as Diretrizes de Acessibilidade por meio da realização de testes de acessibilidade com usuários cegos. • Refinar as Diretrizes de Acessibilidade a partir dos resultados obtidos nos testes de acessibilidade.

#### **Avaliação dos Riscos e Benefícios:**

Riscos:

Os riscos relacionados a esta pesquisa são mínimos, tendo em vista que não haverá exposição dos participantes a procedimentos, materiais ou substâncias que possam causar algum prejuízo à sua integridade física. Deste modo, os riscos para os participantes desta pesquisa se referem às possibilidades remotas de cansaço (ocasionado pela permanência em frente ao computador), desconforto ou constrangimento (ao realizar as tarefas que comporão o teste de acessibilidade ou responder ao questionário de avaliação da satisfação). O teor das tarefas do teste de acessibilidade e das perguntas do questionário apresentam riscos mínimos, considerados inerentes à vida diária. Além disso, a pesquisa não pretende causar danos morais ou riscos à saúde física, mental, social ou espiritual dos participantes. Os riscos podem ser minimizados, uma vez que o(s) participante(s) poderá escolher o momento mais adequado para a sua permanência em frente ao computador, reduzindo o cansaço, e interromper o teste ou o preenchimento do questionário. Da mesma forma, o participante poderá solicitar uma pausa ou mesmo a interrupção do teste e do preenchimento do questionário a qualquer momento. Além disso, a minimização do desconforto e constrangimento, bem como a manutenção da privacidade serão garantidas, visto que os dados permanecerão em total sigilo e garantia de anonimato, sendo apenas o pesquisador o seu conhecedor. Porém, se reconhece que ao expressar o que está sendo experienciado, os sujeitos podem mobilizar emoções. Quanto a este risco, a pesquisadora se responsabiliza pelo atendimento de saúde, encaminhando a pessoa para o atendimento disponível mais próximo, ou aquele de preferência do participante. Nenhum dos procedimentos a serem utilizados nesta pesquisa oferece riscos físicos e

**Endereço:** Alameda Santiago do Chile, 195

**Bairro:** Nossa Sra. das Dores

**CEP:** 97.050-685

**UF:** RS

**Município:** SANTA MARIA

**Telefone:** (55)3218-9800

**E-mail:** cep@iffarroupilha.edu.br

Continuação do Parecer: 6.072.734

à dignidade dos participantes, bem como estes procedimentos adotados obedecem aos Critérios de Ética em Pesquisa com Seres Humanos conforme Resolução n.º 466/2012 e as normas aplicáveis a pesquisas em Ciências Humanas e Sociais conforme Resolução n.º 510/2016 do Conselho Nacional de Saúde.

**Benefícios:**

Como benefícios indiretos para os participantes do estudo, destaca-se a possibilidade de expressarem, compartilharem e serem ouvidos atentamente em relação às principais barreiras encontradas na utilização de ferramentas e ambientes de desenvolvimento de software por estudantes cegos. Os resultados devem permitir que se trace alguns caminhos possíveis para contribuir na melhoria dos processos e recursos disponíveis.

**Comentários e Considerações sobre a Pesquisa:**

Protocolo Original - versão 2

Número de participantes: 10

Data de início do contato com participante: 01/11/2023

Data de fim do contato com participante: 30/11/2023

**Considerações sobre os Termos de apresentação obrigatória:**

Vide campo "Conclusões ou Pendências e Lista de Inadequações".

**Conclusões ou Pendências e Lista de Inadequações:**

Após análise do protocolo de pesquisa, não foram encontrados óbices éticos. Abaixo seguem as pendências que havia e foram sanadas.

1. No documento "ProjetoPesquisa.pdf"

É responsabilidade do pesquisador responsável "manter os dados da pesquisa em arquivo, físico ou digital, sob sua guarda e responsabilidade, por um período mínimo de 5 (cinco) anos após o término da pesquisa". Se faz necessário, portanto, explicitar de que forma esse armazenamento será realizado, e por quanto tempo. (Resolução CNS nº 510/2016). Essa informação deve constar também no TCLE.

- PENDÊNCIA SANADA.

2. Nos documentos "TCLE\_EstudantesEgressosMaiores18anos.pdf"; "TCLE\_PaisResponsáveis.pdf" e "TermoAssentimento\_EstudanteEgressos\_Menores18anos.pdf":

Atentar para a terminologia: de acordo com a regulamentação do sistema CEP/CONEP, o termo

**Endereço:** Alameda Santiago do Chile, 195

**Bairro:** Nossa Sra. das Dores

**UF:** RS

**Município:** SANTA MARIA

**Telefone:** (55)3218-9800

**CEP:** 97.050-685

**E-mail:** cep@iffarroupilha.edu.br

INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E  
TECNOLOGIA FARROUPILHA -  
IFFAR



Continuação do Parecer: 6.072.734

adequado é "participante de pesquisa", e não participante, sujeito, voluntário ou assemelhado.

- PENDÊNCIA SANADA.

3. Nos termos de consentimento/assentimento deve ser explicitado como se dará a garantia do anonimato dos participantes e do sigilo de suas informações.

- PENDÊNCIA SANADA.

4. Os documentos "TCLE\_EstudantesEgressosMaiores18anos.pdf" e

"TermoAssentimento\_EstudanteEgressos\_Menores18anos.pdf" são apresentados de forma pouco adequada ao público-alvo. Solicita-se adequação em conformidade com o Capítulo III da resolução CNS nº 510/2016.

- PENDÊNCIA SANADA.

**Considerações Finais a critério do CEP:**

O CEP acata o parecer do(a) relator(a).

Orientações importantes, conforme a Norma Operacional CNS nº 001/2013:

1) Alterações no projeto aprovado, devem ser apresentadas ao CEP na forma de Emenda ou Extensão. Havendo modificações importantes de objetivos e métodos, deve ser apresentado novo protocolo de pesquisa.

2) Ao final da pesquisa cabe ao (à) pesquisador(a) responsável a apresentação do relatório final ao CEP, no formato de Notificação. Na página do CEP no portal do IFFAR constam orientações e modelo para a apresentação do relatório.

Obs: Orientações sobre a submissão de emendas, extensões ou notificações estão disponíveis no Manual do Pesquisador da Plataforma Brasil. Um material informativo adicional está disponível na página do CEP IFFAR: <https://www.iffarroupilha.edu.br/comit%C3%AA-de-%C3%A9tica-em-pesquisa-2>

**Este parecer foi elaborado baseado nos documentos abaixo relacionados:**

Tipo Documento	Arquivo	Postagem	Autor	Situação
Informações Básicas do Projeto	PB_INFORMAÇÕES_BÁSICAS_DO_PROJETO_2093225.pdf	08/05/2023 14:23:59		Aceito

**Endereço:** Alameda Santiago do Chile, 195

**Bairro:** Nossa Sra. das Dores

**CEP:** 97.050-685

**UF:** RS

**Município:** SANTA MARIA

**Telefone:** (55)3218-9800

**E-mail:** cep@iffarroupilha.edu.br

INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E  
TECNOLOGIA FARROUPILHA -  
IFFAR



Continuação do Parecer: 6.072.734

Outros	Resposta_Avaliacao_CEP_Versao2.docx	08/05/2023 14:23:40	ELIANA ZEN	Aceito
Parecer Anterior	PB_PARECER_CONSUBSTANCIADO_CEP_6018306.pdf	02/05/2023 13:31:50	ELIANA ZEN	Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	TermoConfidencialidade.pdf	02/05/2023 13:31:16	ELIANA ZEN	Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	TermoAssentimento_EstudantesEgressos_Menores18anos.pdf	02/05/2023 13:30:55	ELIANA ZEN	Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	TCLE_PaisResponsaveis.pdf	02/05/2023 13:30:43	ELIANA ZEN	Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	TCLE_EstudantesEgressosMajores18anos.pdf	02/05/2023 13:30:33	ELIANA ZEN	Aceito
Projeto Detalhado / Brochura Investigador	ProjetoPesquisa.pdf	02/05/2023 13:30:19	ELIANA ZEN	Aceito
Declaração de Instituição e Infraestrutura	CartaAuencialFFarPRPPGI.pdf	22/03/2023 13:03:20	ELIANA ZEN	Aceito
Folha de Rosto	FolhaDeRosto.pdf	28/02/2023 14:20:50	ELIANA ZEN	Aceito

**Situação do Parecer:**

Aprovado

**Necessita Apreciação da CONEP:**

Não

SANTA MARIA, 22 de Maio de 2023

---

**Assinado por:  
Talitha Comaru  
(Coordenador(a))**

**Endereço:** Alameda Santiago do Chile, 195

**Bairro:** Nossa Sra. das Dores

**CEP:** 97.050-685

**UF:** RS

**Município:** SANTA MARIA

**Telefone:** (55)3218-9800

**E-mail:** cep@iffarroupilha.edu.br

## **Anexos**

ANEXO A – Questionário enviado aos docentes de Programação de Computadores

# PESQUISA: Percepção do docente de disciplinas relacionadas à Programação de Computadores a respeito da utilização de ferramentas de desenvolvimento de software por estudantes cegos

Este formulário refere-se a um instrumento de coleta de dados integrante da pesquisa de Doutorado em Computação pela Universidade Federal de Pelotas (UFPel), realizado pela aluna Eliana Zen sob orientação da professora Dra. Tatiana Aires Tavares.

\* Indica uma pergunta obrigatória

1. E-mail \*

---

## TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Prezado(a) docente!

Você está sendo convidado(a) a participar da pesquisa denominada "Levantamento de dificuldades de utilização de ferramentas de desenvolvimento de software por estudantes com deficiência visual nas disciplinas relacionadas à Programação de Computadores". O objetivo é identificar as principais barreiras percebidas pelo professor para os estudantes cegos dos cursos técnicos e superiores nas disciplinas relacionadas à Programação de Computadores, com relação à Tecnologia Assistiva, soluções alternativas e ambientes para desenvolvimento de software. Justifica-se a importância desta pesquisa pelo fato de que na área de Computação, estudantes cegos encontram dificuldades que podem não ser sanadas com a utilização de recursos de Tecnologia Assistiva disponíveis atualmente. Além disso, se para desenvolvedores de software a utilização de ferramentas ou ambientes de desenvolvimento de software pode representar um desafio, para estudantes cegos, que estão aprendendo os fundamentos teóricos e práticos relacionados à programação de computadores, o desafio pode ser ainda maior.

Na realização do presente estudo, é necessário que você responda a um questionário on-line com questões abertas e fechadas, as quais buscam levantar o seu conhecimento e entendimento sobre a temática da pesquisa.

O teor das perguntas apresenta riscos mínimos, considerados inerentes à vida diária. Também, não pretende causar danos morais ou riscos à sua saúde física, mental, social ou espiritual. Você poderá, a qualquer momento, recusar-se a responder qualquer pergunta ou desistir de participar e retirar seu consentimento. Sua recusa não trará nenhum prejuízo em sua relação com a pesquisadora ou com a instituição em que trabalha. A pesquisadora estará à disposição para esclarecimento de dúvidas referentes a este estudo.

O benefício relacionado à sua participação na pesquisa será o de contribuir para o aumento do conhecimento científico para a área de acessibilidade, principalmente para cegos. A sua participação na pesquisa não acarretará prejuízo financeiro, bem como não será remunerada. Você tem direito à remuneração e à indenização diante de eventuais danos decorrentes da pesquisa, bem como de não responder qualquer questão, sem necessidade de explicação ou justificativa para tal, podendo também retirar o seu consentimento a qualquer momento, sem qualquer prejuízo ou sanção. Ainda, você tem o direito de ter acesso aos resultados da pesquisa.

Este projeto foi aprovado pelo Comitê de Ética em Pesquisa do Instituto Federal Farroupilha (IFFar), conforme Parecer Consubstanciado n.º 5.250.149. Tudo o que for produzido nos questionários será guardado pela pesquisadora responsável e após o processamento dos dados, será destruído.

Declaro que fui informado(a) dos objetivos da pesquisa acima mencionados, de maneira clara e detalhada, e esclareci minhas dúvidas. Sei que em qualquer momento poderei solicitar novas informações e modificar minha decisão se assim desejar. A pesquisadora certificou-me de que minha identidade será preservada. Em caso de dúvidas poderei chamar a pesquisadora responsável, Eliana Zen, pelo e-mail [eliana.zen@inf.ufpel.edu.br](mailto:eliana.zen@inf.ufpel.edu.br) e/ou telefone (55) 996033120. Declaro que concordo em participar deste estudo, que recebi uma cópia deste TCLE e que me foi dada a oportunidade de ler e esclarecer as minhas dúvidas.

2. Declaro que li e estou de acordo com o Termo de Consentimento Livre e Esclarecido? \*

Marcar apenas uma oval.

Sim

Não

3. Nome completo: \*

---

## Formação Acadêmica e Experiência Profissional

4. 1. Formação Acadêmica (Graduação): \*

Marque todas que se aplicam.

Bacharelado em Ciência da Computação

Bacharelado em Engenharia da Computação

Bacharelado em Engenharia de Software

Bacharelado em Informática

Bacharelado em Sistemas de Informação

Licenciatura em Computação

Tecnologia em Análise e Desenvolvimento de Sistemas

Tecnologia em Sistemas para Internet

Outro: \_\_\_\_\_

5. 2. Tempo de experiência em docência:

Marcar apenas uma oval.

- Menos de 1 ano
- Entre 1 e 5 anos
- Entre 5 e 10 anos
- Mais de 10 anos

6. 3. Instituição(ões) que atua/atuou com estudante(s) cego(s):

\_\_\_\_\_

7. 4. Tipo de Instituição:

Marque todas que se aplicam.

- Pública
- Privada
- Outro: \_\_\_\_\_

8. 5. Quantidade de estudantes cegos para os quais você já lecionou disciplina relacionada à Programação de Computadores:

Marcar apenas uma oval.

- 1 estudante
- 2 estudantes
- 3 estudantes
- Outro: \_\_\_\_\_

#### Recursos utilizados e Disciplinas ministradas

9. 6. Quais recursos de Tecnologia Assistiva foram utilizadas pelo(s) estudante(s) cego(s) nas disciplinas de Programação de Computadores?

Marque todas que se aplicam.

- Leitor de Tela
- Display Braille
- Não foi utilizada Tecnologia Assistiva
- Outro: \_\_\_\_\_

10. 7. Esses recursos já eram conhecidos e/ou utilizadas pelo(s) estudante(s) quando a disciplina iniciou?

Marcar apenas uma oval.

- Sim
- Não
- Em parte

11. 8. Caso o(s) estudante(s) tenha(m) utilizado "leitores de tela", indique quais foram utilizados:

Marque todas que se aplicam.

- JAWS
- DosVox
- VirtualVision
- NVDA
- Orca
- Não foi utilizado leitor de tela
- Outro: \_\_\_\_\_

12. 9. Caso o(s) estudante(s) tenha(m) utilizado "leitores de tela", foi identificada alguma limitação na(s) ferramenta(s)?

Marcar apenas uma oval.

- Sim  
 Não  
 Não foi utilizado leitor de tela

13. 10. Caso tenha selecionado a opção "Sim" na questão anterior, quais limitações foram identificadas?

---

---

---

---

---

14. 11. Há quanto tempo você lecionou para este(s) estudante(s)?

Marque todas que se aplicam.

- Há menos de 5 anos  
 Entre 5 e 10 anos atrás  
 Há mais de 10 anos

15. 12. Quanto à(s) forma(s) de oferta da disciplina que você lecionou, selecione a(s) alternativa(s) apropriada(s):

Marque todas que se aplicam.

- Ensino Presencial  
 Ensino Remoto  
 Ensino à Distância  
 Ensino Híbrido  
 Outro: \_\_\_\_\_

16. 13. Em qual nível de ensino você atua e/ou atuou lecionando disciplina relacionada à programação de computadores para estudante(s) cego(s)?

Marque todas que se aplicam.

- Educação de Jovens e Adultos  
 Curso Técnico  
 Curso Superior (Bacharelado, Licenciatura ou Curso de Tecnologia)  
 Pós-Graduação (Especialização, Mestrado ou Doutorado)  
 Outro: \_\_\_\_\_

17. 14. O(s) estudante(s) concluiu/concluíram o curso?

Marcar apenas uma oval.

- Sim  
 Não  
 Parte dos estudantes concluiu, parte não (no caso de mais de um estudante)  
 Desconheço essa informação  
 Outro: \_\_\_\_\_

18. 15. Quais conteúdos foram abordados nas disciplinas de programação de computadores lecionadas para estudante(s) cego(s)?

Marque todas que se aplicam.

- Introdução a Algoritmos
- Estruturas de Dados
- Lógica de Programação
- Programação Estruturada
- Programação Orientada a Objetos
- Programação para Dispositivos Móveis
- Programação Visual (front-end)
- Programação para Web
- Outro: \_\_\_\_\_

19. 16. Quais linguagens de programação foram utilizadas nas disciplinas de programação de computadores lecionadas para estudante(s) cego(s)?

Marque todas que se aplicam.

- C
- C++
- C#
- Delphi
- Java
- Pascal
- PHP
- Português Estruturado
- Python
- VisualBasic
- Outro: \_\_\_\_\_

20. 17. Quais ferramentas de desenvolvimento e/ou IDEs foram utilizadas por estudante(s) cego(s) nas disciplinas de Programação de Computadores?

Marque todas que se aplicam.

- Bloco de Notas
- Dev-C++
- Eclipse
- FreePascal
- Lazarus
- NetBeans
- NotePad++
- VisuAlg
- VisualStudio
- Outro: \_\_\_\_\_

21. 18. As ferramentas escolhidas eram acessíveis ao(s) estudante(s)?

Marcar apenas uma oval.

- Sim
- Não
- Em parte

22. 19. Caso tenha selecionado a opção "Não" ou "Em parte" na questão anterior, descreva as limitações enfrentadas pelos estudantes para interagir com a ferramenta:

---

---

---

---

---

23. 20. Foi necessário trocar a ferramenta escolhida?

Marcar apenas uma oval.

- Sim
- Não

24. 21. Caso tenha selecionado a opção "Sim" na questão anterior, descreva quais ferramentas foram utilizadas e por que foi necessário fazer a troca:

---

---

---

---

---

25. 22. Na sua opinião, o(s) estudante(s) cegos(s) encontraram barreiras de interação com relação à Tecnologia Assistiva e/ou Ambiente de Desenvolvimento de Software nas disciplinas de Programação de Computadores?

Marcar apenas uma oval.

- Sim
- Não
- Em parte

26. 23. Caso tenha selecionado a opção "Sim" ou "Em parte" na questão anterior, analise as alternativas listadas a seguir e indique a frequência que os estudantes encontravam barreiras de interação relacionadas aos tópicos listados:

Marcar apenas uma oval por linha.

	Nunca	Raramente	Às vezes	Muitas vezes	Sempre
<b>Utilizar/acessar funcionalidades oferecidas pelas ferramentas utilizadas para escrita do código-fonte do programa</b>	<input type="radio"/>				
<b>Utilizar a tecnologia assistiva disponível</b>	<input type="radio"/>				
<b>Utilizar o computador, de modo geral</b>	<input type="radio"/>				

27. 24. Em relação à aprendizagem dos conteúdos relacionados à Programação de Computadores, analise as alternativas listadas a seguir e indique a frequência que os estudantes encontravam dificuldades relacionadas aos tópicos listados:

Marcar apenas uma oval por linha.

	Nunca	Raramente	Às vezes	Muitas vezes	Sempre
<b>Propor a lógica para a resolução do programa</b>	<input type="radio"/>				
<b>Utilizar a sintaxe correta</b>	<input type="radio"/>				
<b>Identificar erros no código</b>	<input type="radio"/>				
<b>Compilar e executar o programa</b>	<input type="radio"/>				
<b>Localizar uma parte específica do código</b>	<input type="radio"/>				
<b>Identificar o início e o fim de uma estrutura (repetição, condicional, função, procedimento, método, etc.)</b>	<input type="radio"/>				
<b>Entender a estrutura de uma parte específica do código</b>	<input type="radio"/>				

28. 25. Se você identificou alguma dificuldade/barreira que não tenha sido listada nas questões anteriores, por favor, especifique-a:

---

---

---

---

---

29. 26. Na sua opinião, quais foram as capacidades/habilidades em que os estudantes cegos se destacaram disciplinas de Programação de Computadores?

---

---

---

---

---

30. 27. Você fez uso de outros recursos e/ou soluções alternativas para auxiliar no processo de ensino-aprendizagem do(s) estudante(s) cego(s) nas disciplinas relacionadas à programação de computadores (tais como materiais táteis, objetos de aprendizagem, etc.)?

Marcar apenas uma oval.

- Sim  
 Não

31. 28. Caso tenha selecionado a opção "Sim" na questão anterior, informe quais recursos e/ou soluções alternativas foram utilizados:

---

---

---

---

---

32. 29. Você participou da elaboração dos recursos de solução alternativa utilizados?

*Marcar apenas uma oval.*

- Sim, os recursos foram elaborados e produzidos por mim
- Sim, os recursos foram elaborados e produzidos por mim com participação do Setor de Ações Inclusivas ou outro(s) profissional(is) da Instituição.
- Não, os recursos foram elaborados e produzidos pelo Setor de Ações Inclusivas ou outro(s) profissional(is) da Instituição.
- Outro: \_\_\_\_\_

33. 30. Em relação às metodologias e estratégias adotadas para o ensino de programação para estudante(s) cego(s), quais foram as maiores dificuldades que você encontrou?

---

---

---

---

---

34. 31. Em relação às metodologias e estratégias adotadas para o ensino de programação para estudante(s) cego(s), você se surpreendeu positivamente em algum momento? Explique:

---

---

---

---

---

35. 32. O(s) estudante(s) recebiam apoio/ajuda extraclasse?

*Marque todas que se aplicam.*

- Sim, atendimento individualizado do docente da disciplina
- Sim, auxílio de monitor da disciplina
- Sim, Atendimento Educacional Especializado (AEE)
- Não, o(s) estudante(s) não recebiam auxílio extraclasse
- Outro: \_\_\_\_\_

36. 33. Sinta-se à vontade para fazer alguma observação ou comentário que considere relevante sobre a sua experiência no ensino de disciplinas de Programação de Computadores e das ferramentas e ambientes de desenvolvimento de software para estudante(s) cego(s):

---

---

---

---

---



ANEXO B – Termo de Consentimento Livre e Esclarecido (TCLE) enviado a  
Estudantes e Egressos

## TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

(Estudantes maiores de 18 anos)

Você está sendo convidado(a) a participar da pesquisa **denominada** “Levantamento de dificuldades de utilização de ferramentas de desenvolvimento de software por estudantes com deficiência visual nas disciplinas relacionadas à Programação de Computadores”. O objetivo é identificar as principais dificuldades encontradas por estudantes com deficiência visual dos cursos técnicos e superiores nas disciplinas relacionadas à Programação de Computadores, com relação aos recursos de Tecnologia Assistiva, soluções alternativas e ambientes de desenvolvimento de software. Justifica-se a importância desta pesquisa pelo fato de que na área de Computação, estudantes com deficiência visual encontram muitas dificuldades, que podem não ser sanadas com a utilização dos recursos de Tecnologia Assistiva disponíveis atualmente. Além disso, se para desenvolvedores de software com deficiência visual a utilização de ferramentas modernas pode representar um desafio, para estudantes com deficiência visual, que estão aprendendo seus fundamentos teóricos e práticos relacionados à programação de computadores, além das ferramentas necessárias para o desenvolvimento de software, o desafio pode ser ainda maior. Dessa forma, pretende-se analisar quais são os principais recursos de Tecnologia Assistiva, soluções alternativas e ambientes para desenvolvimento de software utilizados nas disciplinas relacionadas à Programação de Computadores nos Cursos do Eixo Informação e Comunicação do Instituto Federal Farroupilha, no intuito de identificar as principais limitações e dificuldades encontradas pelos estudantes.

Na realização do presente estudo, você **participará de uma entrevista on-line** com duração de, no máximo, 1 hora. Esta entrevista compreende questões relacionadas a informações pessoais, deficiência, Tecnologia Assistiva e soluções alternativas utilizadas, bem como dificuldades encontradas nas disciplinas que envolvem conteúdos relacionados à Programação de Computadores. A sua participação é voluntária e, em momento algum, esta pesquisa servirá para sua identificação e/ou julgamento das informações prestadas.

O teor das perguntas apresenta **riscos mínimos**, considerados inerentes à vida diária. Também, **não pretende causar danos** morais ou riscos à sua saúde física, mental, social ou espiritual. Entretanto, caso se sinta desconfortável, a pesquisa será imediatamente interrompida e você receberá suporte inicial da própria pesquisadora, que se responsabiliza para viabilizar atendimento profissional, se necessário. Você poderá, a qualquer momento, recusar-se a responder qualquer pergunta ou desistir de participar e retirar seu consentimento. A pesquisadora estará à disposição para esclarecimento de dúvidas referentes a este estudo.

O benefício relacionado à sua participação na pesquisa será o de contribuir para o aumento do conhecimento científico para a área de acessibilidade, principalmente para deficientes visuais.

A sua participação na pesquisa **não acarretará prejuízo financeiro**, bem como não será remunerada. Você tem direito à **remuneração** e à **indenização** diante de eventuais danos decorrentes da pesquisa, bem como de não responder qualquer questão, sem necessidade de explicação ou justificativa para tal, podendo também **retirar o seu consentimento** a qualquer momento, sem qualquer prejuízo ou sanção. Ainda, você tem o direito de ter **acesso aos resultados** da pesquisa.

Este projeto foi aprovado pelo Comitê de Ética em Pesquisa do Instituto Federal Farroupilha (IFFar). Tudo o que for produzido nas entrevistas será guardado pela pesquisadora responsável e após o processamento dos dados, será destruído.

Eu, [NOME DO ESTUDANTE], declaro que fui informado(a) dos objetivos da pesquisa acima mencionados, de maneira clara e detalhada, e esclareci minhas dúvidas. Sei que em qualquer momento **poderei solicitar novas informações e modificar minha decisão**, se assim desejar. A pesquisadora certificou-me de que **minha identidade será preservada**. Em caso de dúvidas **poderei chamar a pesquisadora responsável**, Eliana Zen, pelo e-mail [eliana.zen@inf.ufpel.edu.br](mailto:eliana.zen@inf.ufpel.edu.br) e/ou telefone (55)XXXXXXXX. Declaro que concordo em participar deste estudo, que recebi uma cópia deste TCLE e que me foi dada a oportunidade de ler e esclarecer as minhas dúvidas.

São Vicente do Sul, \_\_\_/\_\_\_/\_\_\_

---

Assinatura do participante

---

Eliana Zen

Assinatura da pesquisadora responsável

ANEXO C – Termo para Uso da Imagem e/ou Som da Voz (TUIV) enviado a  
Estudantes e Egressos

## TERMO PARA USO DE IMAGEM E/OU SOM DA VOZ

Eu, [NOME DO PARTICIPANTE], permito que a pesquisadora **Eliana Zen** obtenha fotografia, filmagem ou gravação de voz de minha pessoa para fins da pesquisa científica/educacional intitulada “Levantamento de dificuldades de utilização de ferramentas de desenvolvimento de software por estudantes com deficiência visual nas disciplinas relacionadas à Programação de Computadores”.

Concordo que o material e as informações obtidas relacionadas a minha pessoa possam ser publicados em aulas, congressos, eventos científicos, palestras ou periódicos científicos. Porém, minha pessoa não deve ser identificada, tanto quanto possível, por nome ou qualquer outra forma.

As fotografias, vídeos e gravações ficarão sob a propriedade da pesquisadora responsável pelo estudo e sob sua guarda.

Santa Maria - RS, \_\_\_/\_\_\_/\_\_\_

---

Assinatura do participante ou responsável legal

---

Eliana Zen  
Assinatura da pesquisadora responsável

## ANEXO D – Roteiro de entrevistas realizadas com estudantes com deficiência visual

## ROTEIRO DE ENTREVISTA SEMIESTRUTURADA

**Público-alvo: Estudantes com deficiência visual**

### 1. Informações pessoais:

- a. Idade
  - b. Sexo
  - c. Tipo de deficiência visual
  - d. Você é estudante e/ou egresso de quais cursos do Eixo Informação e Comunicação?
  - e. Instituição de Ensino
  - f. Ano de conclusão do(s) curso(s)
  - g. Você atua ou atuou profissionalmente em atividades relacionadas aos processos de desenvolvimento de software?
2. Você utilizou e/ou utiliza alguma **Tecnologia Assistiva** para auxiliar nas disciplinas relacionadas à programação de computadores? Quais?
  3. Você utilizou e/ou utiliza algum **ambiente e/ou ferramenta de desenvolvimento de software** (IDE) nas disciplinas relacionadas à programação de computadores? Quais?
  4. Você utilizou e/ou utiliza alguma **solução alternativa** nas disciplinas relacionadas à programação de computadores? Quais?
  5. A seguir são listadas algumas dificuldades comuns encontradas pelos estudantes nas disciplinas relacionadas à programação de computadores. Quais dessas **dificuldades** você encontrou?
  6. Quais tarefas de programação são mais desafiadoras ou exigem mais esforço?

ANEXO E – Roteiro de Grupo Focal com Estudantes da Disciplina de Interação  
Humano Computador

# ROTEIRO DE GRUPO FOCAL

Um Grupo Focal é uma técnica de coleta de dados especialmente indicada para pesquisas qualitativas [KIND, 2004], envolvendo a reunião de pessoas em uma discussão ou entrevista coletiva [BARBOSA & SILVA, 2010]. O objetivo é obter diferentes pontos de vista e coletar dados sobre um determinado assunto. Os dados coletados consideram o processo do grupo, reconhecendo-o como mais do que apenas a soma das opiniões, sentimentos e pontos de vista individuais envolvidos [KIND, 2004].

## **PARTICIPANTES**

Estudantes da Disciplina de Interação Humano Computador do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal Farroupilha Campus São Vicente do Sul (RS).

## **DESCRIÇÃO:**

Este grupo focal visa reunir ideias e opiniões sobre a percepção de estudantes da Disciplina de Interação Humano Computador sobre as recomendações preliminares de acessibilidade em IDEs para estudantes cegos. A dinâmica do grupo focal seguirá a seguinte estrutura:

### **1. INTRODUÇÃO**

Tempo estimado: 10 min

#### **Atividades previstas:**

- 1.1 Breve explicação para os participantes sobre: os objetivos da pesquisa e do grupo focal; a relevância do tema e o que será feito com os dados obtidos; os critérios metodológicos do grupo focal, bem como as regras, duração e a importância da contribuição de todos.
- 1.2 É assegurado aos participantes que não existem opiniões corretas, que opiniões contrárias serão bem-vindas, e que não há interesse em nenhuma opinião em particular.
- 1.3 Leitura e explicação do Termo de Consentimento Livre e Esclarecido (TCLE) e do Termo para Uso da Imagem e/ou Som da Voz (TUIV).
- 1.4 Apresentação da equipe, atribuições de cada membro e explanação sobre os aspectos éticos da discussão e da gravação.

### **2. ATIVIDADE PRÁTICA**

Tempo estimado: 15 min

#### **Atividades previstas:**

- 2.1 Os participantes recebem uma lista contendo todos os atalhos necessários para a realização da atividade prática.

- 2.2 A turma é dividida em grupos de 2 ou 3 integrantes. Cada grupo recebe uma descrição de exercício que deve ser implementado em linguagem Java.
- 2.3 O moderador solicita que um estudante de cada grupo abra o IDE VSCode, digite, compile e execute o código-fonte referente à atividade solicitada. Este estudante não poderá visualizar o que está sendo exibido no monitor em nenhum momento durante a realização da atividade. Além disso, é recomendado que o participante que estiver digitando o código utilize a técnica *think aloud*<sup>1</sup> para compartilhar com os colegas de grupo tudo o que estiver pensando durante a realização da atividade. Os outros participantes devem tomar notas das dúvidas encontradas pelo participante que está digitando, o tempo para realizar cada tarefa, se a tarefa foi realizada com êxito, a quantidade de erros (se houver).

### 3. DISCUSSÃO

Tempo estimado: 20 min

O moderador reúne todos os estudantes e inicia a discussão a respeito das tarefas realizadas

1. Quais as dificuldades encontradas?
2. O que foi mais difícil?
3. A tarefa foi finalizada com sucesso?
4. Todas as dificuldades encontradas são abordadas pelas diretrizes? Em caso negativo, quais são as não abordadas?

### 4. ENCERRAMENTO

Tempo: 5 min

Encerramento do Grupo Focal.

### REFERÊNCIAS

KIND, Luciana. Notas para o trabalho com a técnica de grupos focais. **Psicologia em revista**, v. 10, n. 15, p. 124-138, 2004.

BARBOSA, Simone; SILVA, Bruno. **Interação humano-computador**. Elsevier Brasil, 2010.

---

<sup>1</sup> Quando é solicitado que o participante relate em voz alta o que ele está pensando e fazendo durante a interação [BARBOSA & SILVA, 2010].

ANEXO F – Roteiro de Grupo Focal com Pedagogas, Professoras de Educação Especial e Professores de Programação de Computadores

# ROTEIRO DE GRUPO FOCAL

Um Grupo Focal é uma técnica de coleta de dados especialmente indicada para pesquisas qualitativas [KIND, 2004], envolvendo a reunião de pessoas em uma discussão ou entrevista coletiva [BARBOSA & SILVA, 2010]. O objetivo é obter diferentes pontos de vista e coletar dados sobre um determinado assunto. Os dados coletados consideram o processo do grupo, reconhecendo-o como mais do que apenas a soma das opiniões, sentimentos e pontos de vista individuais envolvidos [KIND, 2004].

## **PARTICIPANTES**

Pedagogas, professoras de Educação Especial e Professores de Programação de Computadores do Instituto Federal Farroupilha Campus São Vicente do Sul (RS).

## **DESCRIÇÃO:**

Uma vez que a Cartilha tem como propósito oferecer orientações a professores que ensinam Programação de Computadores a estudantes cegos, este grupo focal visa reunir ideias e opiniões sobre a necessidade de realização de ajustes e sobre a utilidade desse material para os diversos profissionais que podem utilizá-la para planejar suas aulas e atividades ou para aqueles que precisam fornecer orientação para esses professores.

A dinâmica do grupo focal seguir a seguinte estrutura:

### **1. INTRODUÇÃO**

Tempo estimado: 10 min

#### **Atividades previstas:**

- 1.1 Breve explicação para os participantes sobre: os objetivos da pesquisa e do grupo focal; a relevância do tema e o que será feito com os dados obtidos; os critérios metodológicos do grupo focal, bem como as regras, duração e a importância da contribuição de todos.
- 1.2 É assegurado aos participantes que não existem opiniões corretas, que opiniões contrárias serão bem-vindas, e que não há interesse em nenhuma opinião em particular.
- 1.3 Leitura e explicação do Termo de Consentimento Livre e Esclarecido (TCLE) e do Termo para Uso da Imagem e/ou Som da Voz (TUIV).
- 1.4 Apresentação da equipe, atribuições de cada membro e explanação sobre os aspectos éticos da discussão e da gravação.

### **2. APRESENTAÇÃO DA CARTILHA**

Tempo estimado: 20 min

- 2.1 Explicar o que motivou a produção da cartilha, sua organização;
- 2.2 Ressaltar que o documento ainda precisa passar pelo processo de diagramação e ilustração antes de sua publicação oficial;

2.3 Deixar claro que as imagens inseridas até o momento são todas protegidas pela licença *Creative Commons*, mas possuem caráter meramente ilustrativo e que a versão final da Cartilha demanda uma análise mais detalhada em relação à sua apresentação, requerendo a colaboração de outros profissionais especializados nesta área.

### **3. DISCUSSÃO**

Tempo estimado: 30min

Discussão a respeito do conteúdo da Cartilha:

- 3.1 Pedir para os participantes compartilharem suas impressões sobre o conteúdo, a linguagem, entre outros aspectos da cartilha?
- 3.2 Explorar se a cartilha é clara e compreensível ou se há partes confusas ou difíceis de entender.
- 3.3 Avaliar se a cartilha aborda questões importantes e se atende às necessidades do público-alvo.
- 3.4 Investigar se os participantes acreditam que a cartilha é útil e se oferece informações práticas.
- 3.5 Identificar as áreas que precisam de melhorias ou ajustes.

### **4. ENCERRAMENTO**

Tempo: 10 min

Agradecer a participação de todos e explicar como o feedback será utilizado para melhorar a cartilha.

### **REFERÊNCIAS**

KIND, Luciana. Notas para o trabalho com a técnica de grupos focais. **Psicologia em revista**, v. 10, n. 15, p. 124-138, 2004.

BARBOSA, Simone; SILVA, Bruno. **Interação humano-computador**. Elsevier Brasil, 2010.

ANEXO G – E-mail convite enviado para Programadores Cegos e Desenvolvedores de IDE

## E-MAIL CONVITE

**Assunto:** Convite para participação em pesquisa de Doutorado sobre Acessibilidade em IDEs

**Texto do e-mail:**

Prezado(a),

Me chamo Eliana Zen, sou professora de Computação no Instituto Federal Farroupilha (RS, Brasil) e atualmente estou cursando o Doutorado em Computação na Universidade Federal de Pelotas (UFPEL - RS, Brasil), com foco na área de Interação Humano-Computador.

Minha proposta de tese de Doutorado concentra-se na elaboração de Diretrizes de Acessibilidade em Ambientes de Desenvolvimento Integrado (IDEs) para estudantes com deficiência visual. Para isso, desenvolvemos um conjunto preliminar de recomendações destinadas a assegurar a acessibilidade dos IDEs para esse público-alvo. A importância desta pesquisa se fundamenta no fato de que na área de Computação, estudantes com deficiência visual frequentemente enfrentam desafios para interagir com as tecnologias disponíveis. Esses desafios podem não ser sanados com a utilização de recursos de Tecnologia Assistiva disponíveis atualmente, o que pode impactar na desmotivação e redução do engajamento dos estudantes e, em alguns casos, causar evasão dos cursos de Computação.

Optamos por concentrar nossa atenção em estudantes de cursos da área de Computação brasileiros porque etapas anteriores da pesquisa revelaram que esses alunos possuem preferências e necessidades específicas.

Para a avaliação, elaboramos um questionário online que inclui um Termo de Consentimento Livre e Esclarecido, ao qual os participantes precisam concordar para participar da pesquisa. O questionário contém perguntas que visam avaliar cada diretriz individualmente e questões que buscam coletar a percepção dos avaliadores quanto à estrutura e organização geral das Diretrizes. Ressalto que a pesquisa foi aprovada por Comitê de Ética em Pesquisa com Seres Humanos.

Gostaria de explorar a possibilidade de divulgar um convite para que **[PROGRAMADORES CEGOS/DESENVOLVEDORES DE IDE]** possam avaliar as recomendações propostas. Para manifestar interesse em participar da pesquisa basta preencher o formulário disponível em <https://forms.gle/P4XJ3qCwRD4mtZ6P7> ou enviar e-mail para [eliana.zen@inf.ufpel.edu.br](mailto:eliana.zen@inf.ufpel.edu.br). Após a manifestação de interesse enviaremos os links e documentos necessários para realizar a avaliação.

Estou à disposição para fornecer mais esclarecimentos, se necessário.

Atenciosamente,

ANEXO H – Respostas para as perguntas que utilizavam escala *Likert* ao questionário de Avaliação das Diretrizes de Acessibilidade em IDEs para Estudantes com Deficiência Visual enviado a Estudantes de Computação

Figura 15 – Pergunta: "A recomendação é apresentada de forma clara e compreensível?"

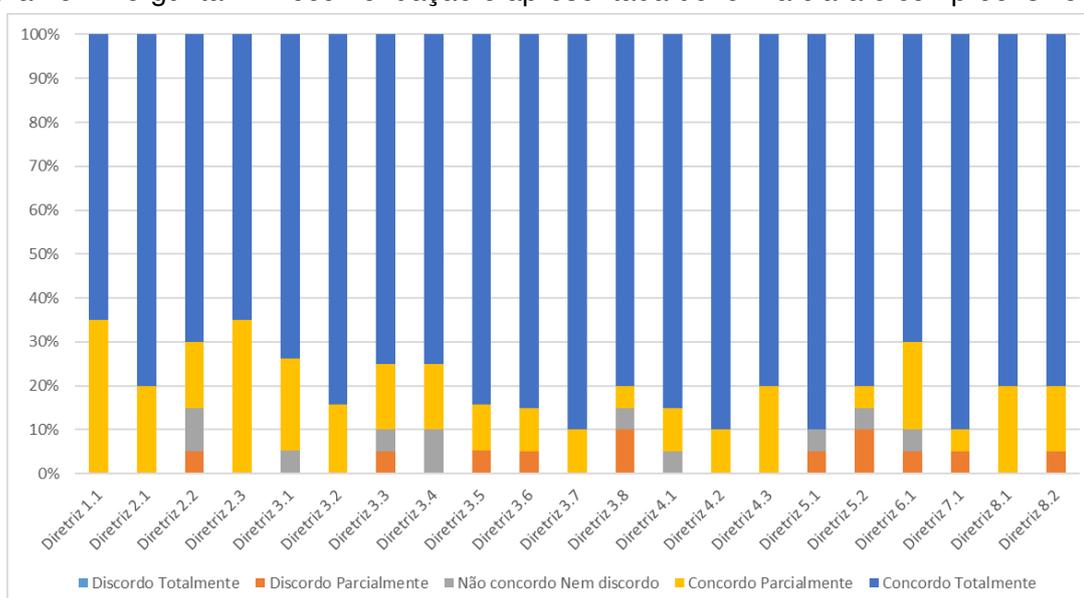


Figura 16 – Pergunta: "O texto da recomendação esclarece a forma como ela deve ser aplicada?"

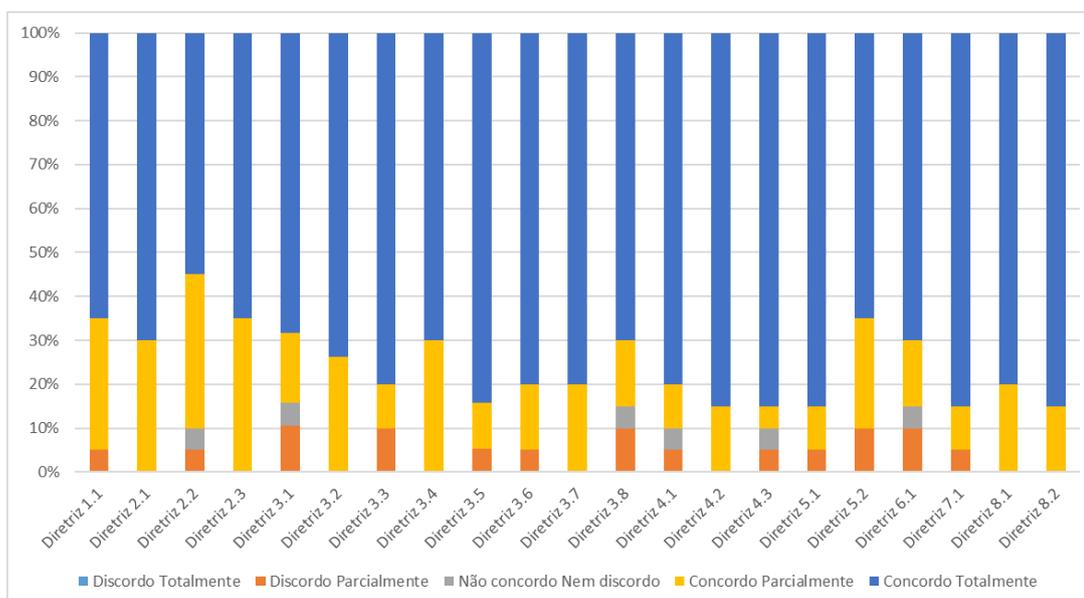


Figura 17 – Pergunta: "Os detalhes foram suficientes para a compreensão da recomendação?"

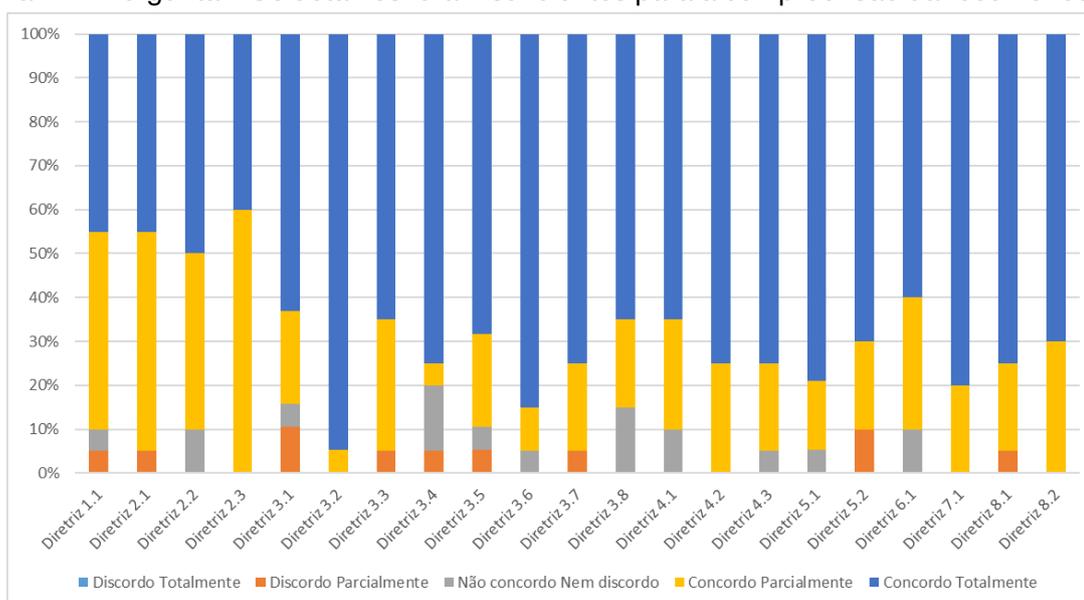


Figura 18 – Pergunta: "A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?"

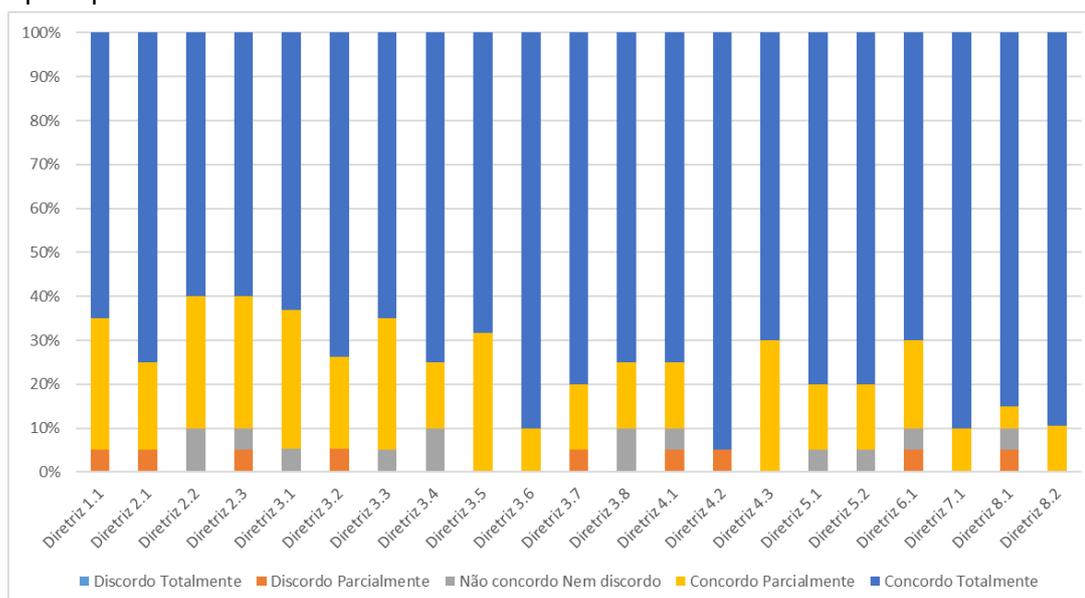


Figura 19 – Pergunta: "As sugestões de implementação apresentadas em "Como fazer?" são úteis?"

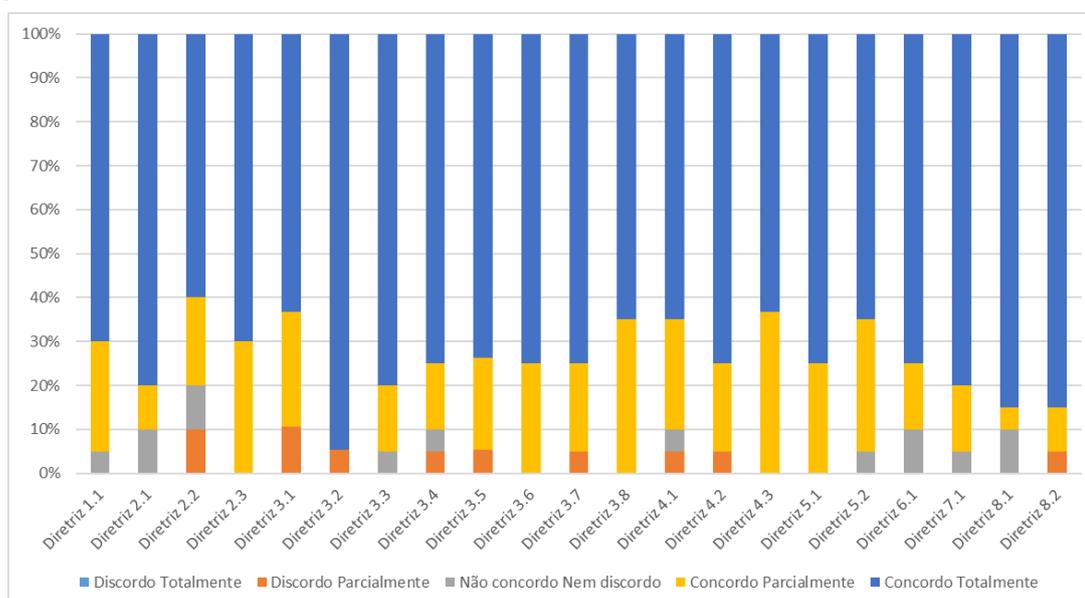
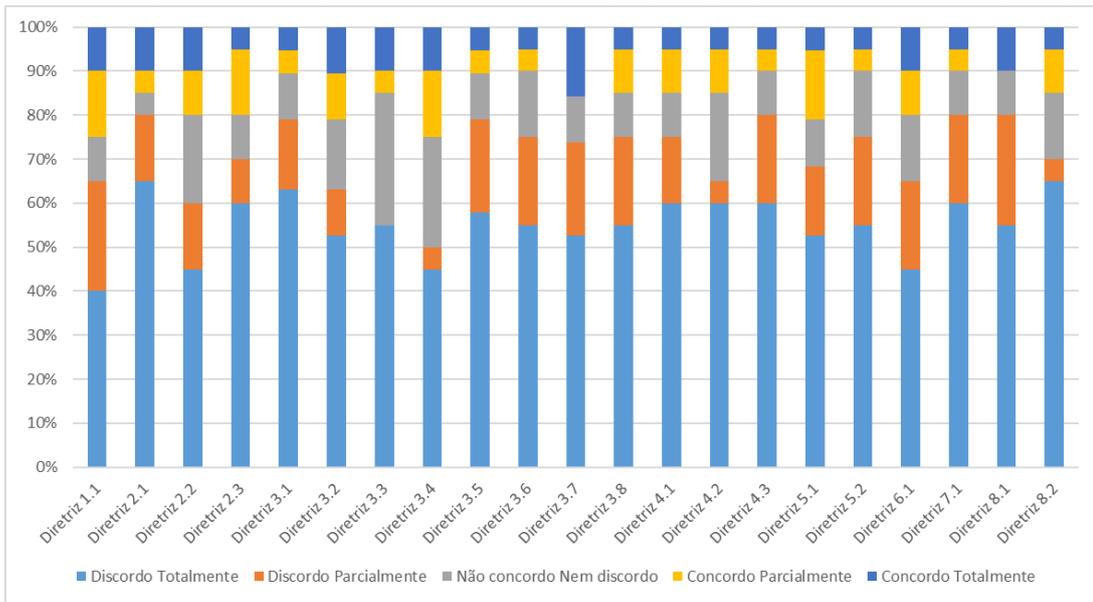


Figura 20 – Pergunta: "Há algum termo técnico ou jargão que precisa de mais explicação?"



ANEXO I – Respostas para as perguntas que utilizavam escala *Likert* ao questionário de Avaliação das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos enviado a Programadores Cegos

Figura 21 – Pergunta: "A recomendação é apresentada de forma clara e compreensível?"

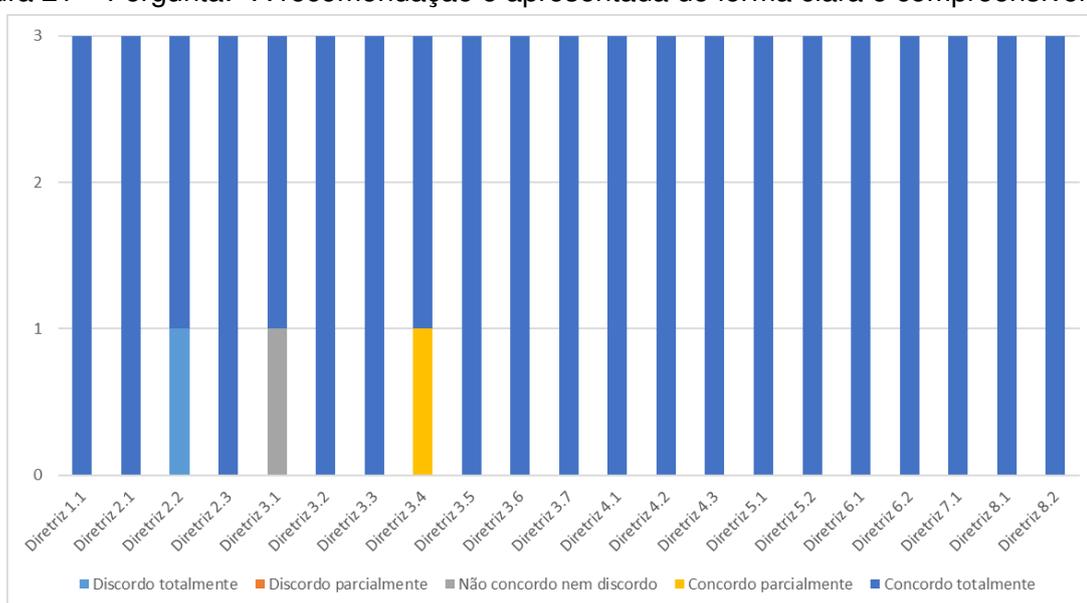


Figura 22 – Pergunta: "O texto da recomendação esclarece a forma como ela deve ser aplicada?"



Figura 23 – Pergunta: "Os detalhes foram suficientes para a compreensão da recomendação?"

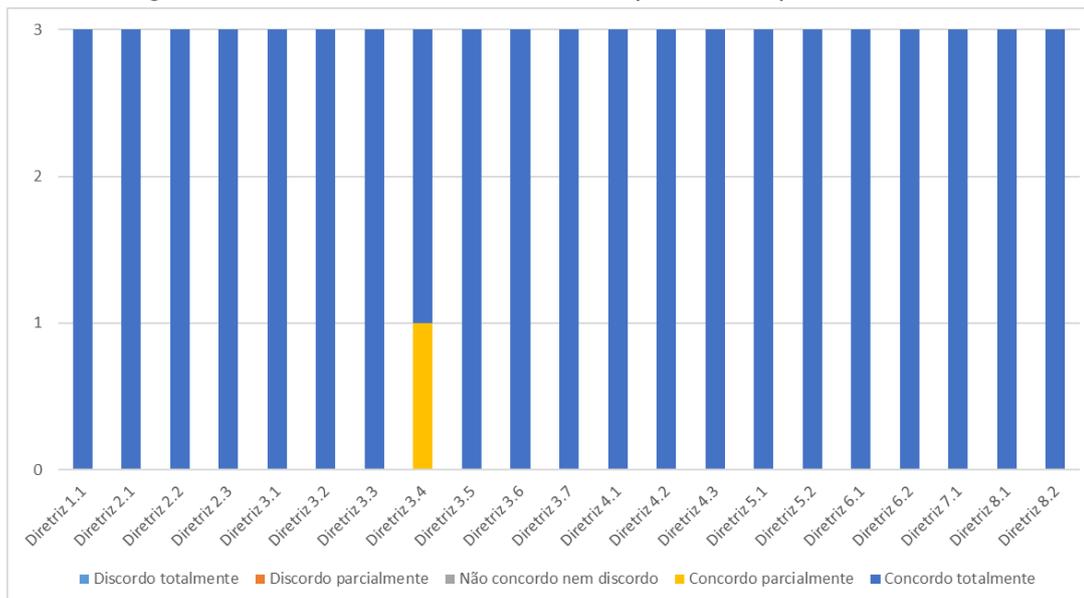


Figura 24 – Pergunta: "A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?"

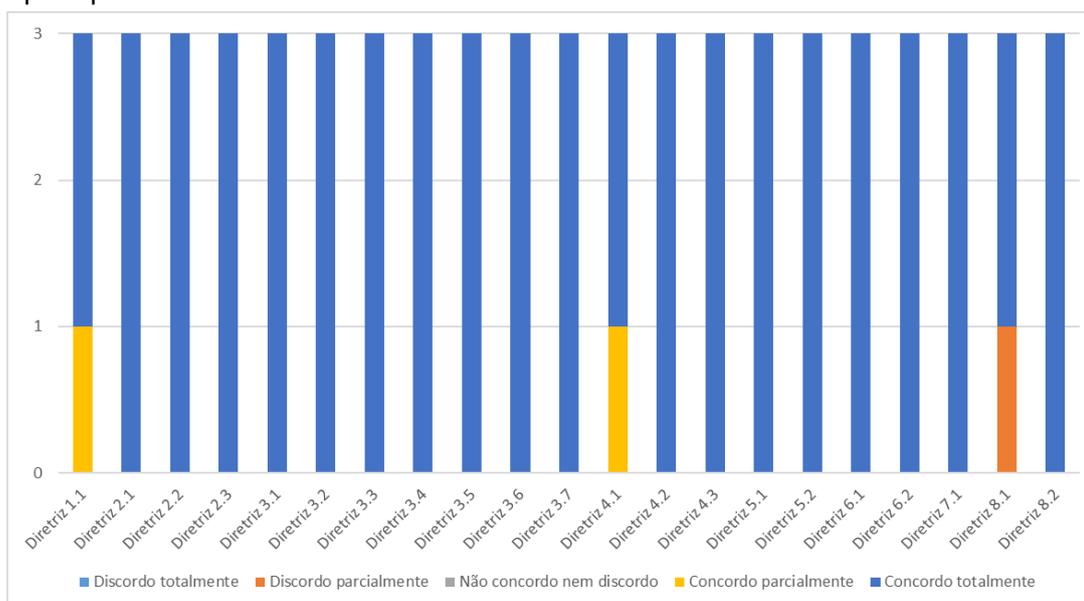
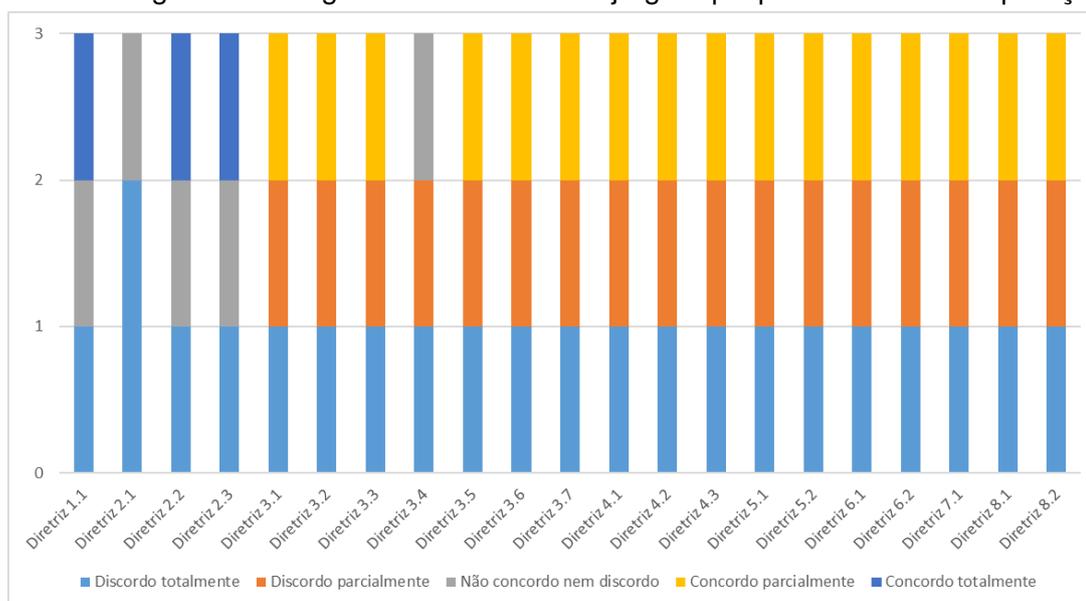


Figura 25 – Pergunta: "As sugestões de implementação apresentadas em "Como fazer?" são úteis?"



Figura 26 – Pergunta: "Há algum termo técnico ou jargão que precisa de mais explicação?"



ANEXO J – Respostas para as perguntas que utilizavam escala *Likert* ao questionário de Avaliação das Diretrizes de Acessibilidade em IDEs para Estudantes Cegos enviado ao Desenvolvedor de IDE

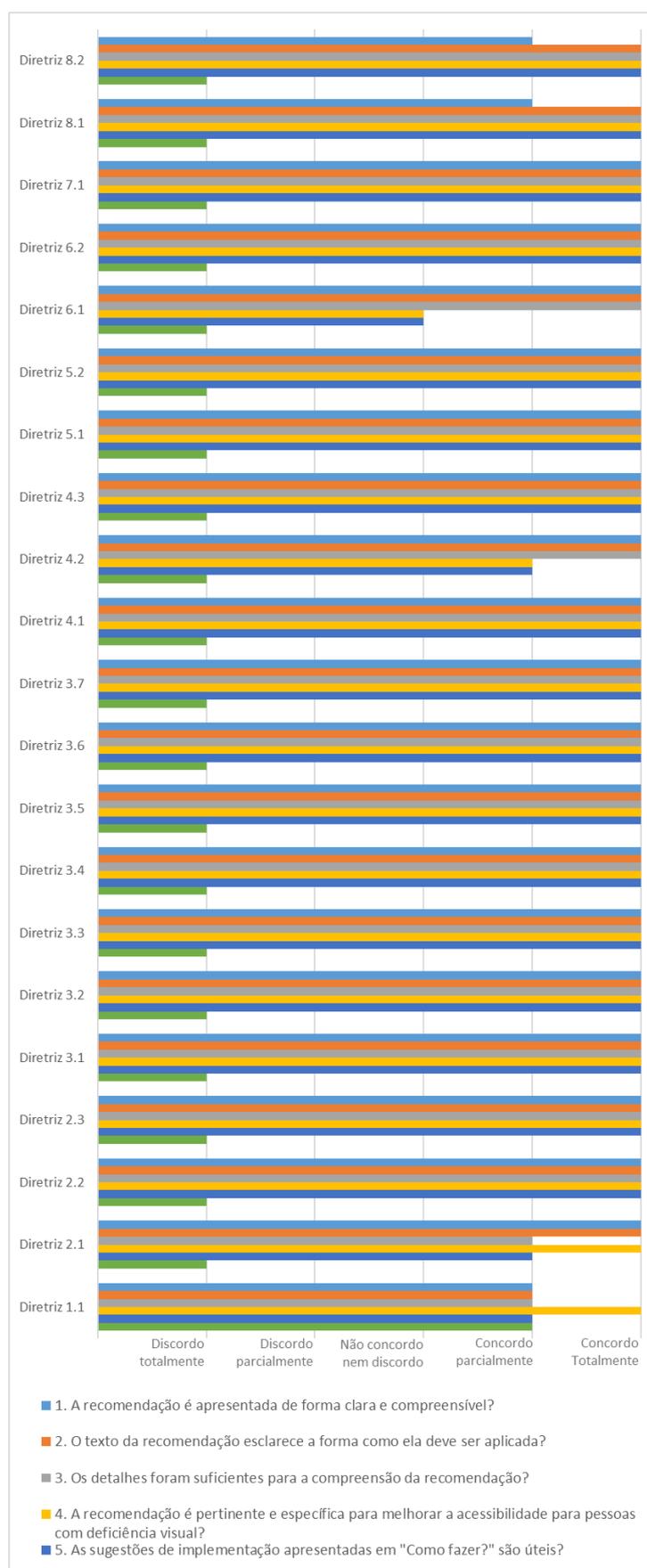


Figura 27 – Avaliação individual das Diretrizes realizada por Desenvolvedor de IDE

ANEXO K – Questionário de Avaliação das Diretrizes de Acessibilidade em IDEs  
para Pessoas com Deficiência Visual

# Avaliação: Diretrizes de Acessibilidade em Ambientes de Desenvolvimento Integrado para Pessoas com Deficiência Visual

Este formulário refere-se a um instrumento de coleta de dados integrante da pesquisa de Doutorado em Computação pela Universidade Federal de Pelotas (UFPel), realizado pela aluna Eliana Zen sob orientação da professora Dra. Tatiana Aires Tavares.

\* Indica uma pergunta obrigatória

## 1. E-mail \*

---

### TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Prezado(a) senhor(a),

Você está sendo convidado(a) a participar da pesquisa denominada "**ELABORAÇÃO DE DIRETRIZES DE ACESSIBILIDADE PARA AMBIENTES DE DESENVOLVIMENTO DE SOFTWARE VOLTADAS A USUÁRIOS COM DEFICIÊNCIA VISUAL**".

O objetivo é avaliar um conjunto de recomendações preliminares para garantir a acessibilidade dos Ambientes de Desenvolvimento de Integrado (*Integrated Development Environments* - IDEs) para pessoas com deficiência visual. Justifica-se a importância desta pesquisa pelo fato de que na área de Computação, estudantes com deficiência visual encontram dificuldades que podem não ser sanadas com a utilização de recursos de Tecnologia Assistiva disponíveis atualmente. Além disso, os IDEs, essenciais para atividades de desenvolvimento de software, possuem características altamente visuais. Esses aspectos, combinados com as particularidades das linguagens de programação, representam obstáculos significativos para estudantes com deficiência visual que buscam obter formação nessa área.

Na realização do presente estudo, é necessário que você responda a um questionário contendo questões abertas e fechadas, as quais buscam verificar se as recomendações propostas são claras e objetivas, bem como se você teria alguma sugestão de alteração ou melhoria.

O teor das perguntas apresenta riscos mínimos, considerados inerentes à vida diária. Também, não pretende causar danos morais ou riscos à sua saúde física, mental, social ou espiritual. Você poderá, a qualquer momento, recusar-se a responder qualquer pergunta ou desistir de participar e retirar seu consentimento. A pesquisadora estará à disposição para esclarecimento de dúvidas referentes a este estudo.

O benefício relacionado à sua participação na pesquisa será o de contribuir para o aumento do conhecimento científico para a área de acessibilidade, principalmente para pessoas com deficiência visual. A sua participação na pesquisa não acarretará prejuízo financeiro, bem como não será remunerada e você tem o direito de ter acesso aos resultados da pesquisa.

Este projeto foi aprovado pelo Comitê de Ética em Pesquisa do Instituto Federal Farroupilha (IFFar). Tudo o que for produzido nos questionários será guardado pela pesquisadora responsável e após o processamento dos dados, será destruído.

Declaro que fui informado(a) dos objetivos da pesquisa acima mencionados, de maneira clara e detalhada, e esclareci minhas dúvidas. Sei que em qualquer momento poderei solicitar novas informações e modificar minha decisão se assim desejar. A pesquisadora certificou-me de que minha identidade será preservada. Em caso de dúvidas poderei chamar a pesquisadora responsável, Eliana Zen, pelo e-mail [eliana.zen@inf.ufpel.edu.br](mailto:eliana.zen@inf.ufpel.edu.br) e/ou telefone (55) 996033120. Declaro que concordo em participar deste estudo, que recebi uma cópia deste TCLE e que me foi dada a oportunidade de ler e esclarecer as minhas dúvidas.

## 2. Declaro que li e concordo com o Termo de Consentimento Livre e Esclarecido? \*

Marcar apenas uma oval.

Sim

Não

## 3. Nome Completo:

---

### Organização do Questionário

Este questionário está dividido em seções. Cada seção apresenta o **resumo das Diretrizes** propostas, as quais são descritas em detalhes no documento "**Diretrizes de Acessibilidade em Ambientes de Desenvolvimento Integrado para Pessoas com Deficiência Visual**", que pode ser acessado através deste link:

Para cada Diretriz, o participante é convidado a responder duas perguntas: **uma questão que utiliza escala Likert**; e, **uma questão aberta**, na qual o participante pode expressar sua opinião a respeito da respectiva Diretriz, além de oferecer sugestões de melhoria, se julgar necessário e pertinente. No final do questionário, são apresentadas mais três perguntas: duas para avaliar a estrutura e organização das Diretrizes e uma que questiona se o participante tem interesse em participar de fases posteriores da pesquisa.

**Desde já agradecemos o seu tempo e contribuição para esta pesquisa. Suas opiniões são extremamente valiosas!**

[Pular para a pergunta 4](#)

## Categoria 1: Compreensão do Código

Relacionada às barreiras encontradas por pessoas com deficiência visual para ler e resumir um código-fonte utilizando leitores de tela.

### Diretriz 1.1 Semântica

Comunicar o significado lógico (semântica) dos elementos que constituem uma base de código.

#### 4. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

#### 5. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

## Categoria 2: Depuração de Código

Compreende as barreiras para identificar falhas no software, compreender a origem da falha e determinar a melhor maneira de removê-la ou corrigi-la.

### Diretriz 2.1 Informações sobre erros

Informar

o usuário sobre a existência de erros no código.

6. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

7. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

## Diretriz 2.2 Variáveis e constantes

Fornecer informações sobre variáveis e constantes declaradas no código-fonte.

8. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

9. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

## Diretriz 2.3 Marcadores e Pontos de interrupção

Permitir a utilização de marcadores ou pontos de interrupção em um arquivo de código-fonte.

10. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

11. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Categoria 3: Navegação no Código

Abrange questões relacionadas à leitura sequencial, linha por linha, realizada pelos leitores de tela, necessidade de procurar informações no código sem perder a posição de foco do cursor, navegação no ambiente de programação e localização dos recursos disponibilizados pela ferramenta de desenvolvimento.

#### Diretriz 3.1 Rótulos

Fornecer

descrições textuais suficientemente claras para todo conteúdo não textual.

12. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

13. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Diretriz 3.2 Movimentação pelo código

Oferecer

recursos alternativos para navegar pela base de código.

14. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

15. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Diretriz 3.3 Contexto e nível de escopo

Informar

o usuário a respeito do contexto e do nível de escopo de uma determinada linha de código ou trecho de código.

16. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

17. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Diretriz 3.4 Estratégias de navegação

Fornecer diferentes formatos de organização e navegação pelo conteúdo sem perder informação.

18. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

19. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Diretriz 3.5 Numeração de linhas

Informar

o número de uma determinada linha de código ou o intervalo numérico de um trecho de código.

20. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

21. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Diretriz 3.6 Atalhos

Fornecer funções e comandos facilmente operáveis por meio de teclado e leitor de tela.

22. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

23. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Diretriz 3.7 Ajuda e Documentação

Fornecer

diferentes formas de ajuda e documentação ao usuário: dicas, tutoriais, recomendações ou exemplos.

24. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

25. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

#### Categoria 4: Edição de Código

Abrange as barreiras encontradas durante a escrita do código-fonte, tais como: uso de indentação para indicar nível de escopo; utilização de cores diferentes para realçar a sintaxe do código; escrita correta dos comandos; uso extensivo de caracteres não alfanuméricos; e, complexidade da sintaxe das linguagens de programação.

#### Diretriz 4.1 Espaçamento

Informar a quantidade total de espaços em branco quando se utilizar linguagens de programação que utilizam esses mecanismos para indicar indentação.

26. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

27. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

#### Diretriz 4.2 Autocompletar

Realizar

a leitura das informações exibidas pelo recurso autocompletar.

28. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

29. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

#### Diretriz 4.3 Comentários

Fornecer informações sobre comentários no código-fonte.

30. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

31. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Categoria 5: *Skimming* de Código

Aborda questões envolvidas no processo de obter uma visão geral de alto nível do código e identificar recursos como dobra de código.

#### Diretriz 5.1 Dobra de Código

Informar quando um trecho de código está dobrado (oculto) e pode ser expandido.

32. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

33. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Diretriz 5.2 Visão geral do código

Fornecer

visão geral mais ampla da base de código.

34. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

35. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

## Categoria 6: Interface Gráfica de Usuário

Compreende as dificuldades encontradas na realização de tarefas de desenvolvimento que envolvem a construção de algum componente visual.

### Diretriz 6.1: Construção de Interface Gráfica

Fornecer mecanismos para facilitar a construção de interfaces que utilizam componentes visuais.

36. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

37. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Diretriz 6.2 Validação da interface gráfica

Realizar validação automatizada da interface gráfica gerada a partir de um código-fonte.

38. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

39. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Categoria 7: Sobrecarga Auditiva

Relacionada ao excesso de informações transmitidas via canal de áudio em uma interface.

#### Diretriz 7.1 Alertas sonoros

Possibilitar personalizar as configurações dos alertas sonoros utilizados.

40. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

41. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Categoria 8: Leitura de Código-fonte

Relacionada à pronúncia das palavras-reservadas das linguagens de programação e das mensagens de erro.

#### Diretriz 8.1 Idioma

Alternar automaticamente entre as línguas faladas pelo leitor de telas.

42. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

43. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

### Diretriz 8.2 Leitura contextual

Ler o código-fonte de maneira que faça sentido para o ouvinte.

44. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
A recomendação é apresentada de forma clara e compreensível?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto da recomendação esclarece a forma como ela deve ser aplicada?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Os detalhes foram suficientes para a compreensão da recomendação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A recomendação é pertinente e específica para melhorar a acessibilidade para pessoas com deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As sugestões de implementação apresentadas em "Como fazer?" são úteis?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há algum termo técnico ou jargão que precisa de mais explicação?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

45. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na escrita desta recomendação, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

#### Avaliação da Estrutura e Organização das Diretrizes

46. Por favor, avalie cada afirmação a seguir com base em sua experiência ou opinião: \*

Marcar apenas uma oval por linha.

	Concordo totalmente	Concordo parcialmente	Não concordo nem discordo	Discordo parcialmente	Discordo totalmente
As recomendações contribuem para a tomada de boas decisões de design?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As recomendações permitem obter um maior entendimento sobre deficiência visual?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As recomendações contribuirão para soluções mais inclusivas?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As recomendações estão clara e consistentemente organizadas?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
As recomendações são práticas e aplicáveis na realidade de desenvolvimento de IDEs?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
O texto mantém uma abordagem coerente em toda a sua extensão?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Há alguma área específica que não foi considerada e que deveria ser incluída?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

47. Caso tenha alguma sugestão para inclusão, exclusão ou melhoria na organização, ou na escrita das recomendações, por favor, utilize este espaço para compartilhá-la conosco.

---

---

---

---

---

48. Você estaria interessado em participar de fases subsequentes deste estudo, que incluiriam a validação de IDEs utilizando a versão final dessas Diretrizes? \*

Marcar apenas uma oval.

- Sim  
 Não

---

Este conteúdo não foi criado nem aprovado pelo Google.

Google Formulários

ANEXO L – Termo de Consentimento Livre e Esclarecido (TCLE)  
Grupo Focal com Estudantes de Computação

## TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Você está sendo convidado(a) a participar da pesquisa denominada “**ELABORAÇÃO DE DIRETRIZES DE ACESSIBILIDADE PARA AMBIENTES DE DESENVOLVIMENTO DE SOFTWARE VOLTADAS A USUÁRIOS COM DEFICIÊNCIA VISUAL**”.

O objetivo é elaborar um conjunto de Diretrizes de Acessibilidade para garantir que pessoas com deficiência visual tenham acesso a todos os recursos disponibilizados pelas ferramentas e ambientes de desenvolvimento de software. Justifica-se a importância desta pesquisa pelo fato de que, na área de Computação, estudantes com deficiência visual encontram muitas dificuldades, que podem não ser sanadas com a utilização dos recursos de Tecnologia Assistiva disponíveis atualmente. Além disso, se para desenvolvedores de software cegos a utilização de ferramentas modernas pode representar um desafio, para estudantes com deficiência visual, que estão aprendendo os fundamentos teóricos e práticos relacionados à programação de computadores, além das ferramentas necessárias para o desenvolvimento de software, o desafio pode ser ainda maior. Dessa forma, pressupõe-se que a utilização de tais diretrizes contribua para a implementação de ferramentas e ambientes de desenvolvimento de software com maior acessibilidade para este grupo de usuários.

Na realização do presente estudo, você **participará de um grupo focal**, contendo atividades práticas de escrita de código-fonte utilizando uma IDE e questões que visam obter informações a respeito da sua opinião sobre as dificuldades encontradas na realização dessas atividades e sobre a aplicabilidade das Diretrizes de acessibilidade que você recebeu previamente. O grupo focal terá duração de, no máximo, 2 horas. A sua participação é voluntária e, em momento algum, esta pesquisa servirá para sua identificação e/ou julgamento das informações prestadas.

O teor das tarefas e das perguntas do questionário apresenta **riscos mínimos**, considerados inerentes à vida diária. Também, **não pretende causar danos** morais ou riscos à sua saúde física, mental, social ou espiritual. Entretanto, caso sinta-se desconfortável, a pesquisa será imediatamente interrompida e você receberá suporte inicial da própria pesquisadora, que se responsabiliza por viabilizar atendimento profissional, se necessário. Você poderá, a qualquer momento, recusar-se a responder qualquer pergunta ou desistir de participar da pesquisa e retirar seu consentimento. A pesquisadora estará à disposição para esclarecimento de dúvidas referentes a este estudo.

O benefício relacionado à sua participação na pesquisa será o de contribuir para o aumento do conhecimento científico para a área de acessibilidade, principalmente para pessoas com deficiência visual.

A sua participação na pesquisa **não acarretará prejuízo financeiro**, bem como não será remunerada. Você tem direito à remuneração e à **indenização** diante de eventuais danos decorrentes da pesquisa, bem como de não responder qualquer questão, sem necessidade de explicação ou justificativa para tal, podendo também **retirar o seu consentimento** a qualquer momento, sem qualquer prejuízo ou sanção. Ainda, você tem o direito de ter **acesso aos resultados** da pesquisa.

Os dados coletados física e eletronicamente serão armazenados e processados sob responsabilidade da pesquisadora responsável pelo estudo. A pesquisadora se responsabiliza por mantê-los seguros, em formato impresso e digital (armazenados em um HD externo de propriedade da pesquisadora), em sua residência, na Rua Pinheiro Machado, 3069, na cidade de Santa Maria/RS. Ademais, após cinco anos de armazenamento dos dados, eles serão descartados de forma adequada pela pesquisadora.

Este projeto foi aprovado pelo Comitê de Ética em Pesquisa (CEP) do Instituto Federal Farroupilha (IFFar), responsável pela avaliação dos questionamentos éticos e legais relacionados ao desenvolvimento das pesquisas com envolvimento de seres humanos no âmbito IFFar, localizado na Reitoria do IFFar (Alameda Santiago do Chile, 195 - Santa Maria, RS), e-mail [cep@iffarroupilha.edu.br](mailto:cep@iffarroupilha.edu.br), telefone (55) 32189800. Tudo o que for produzido no grupo focal será guardado pela pesquisadora responsável e, após o processamento dos dados, será destruído.

Eu, \_\_\_\_\_, declaro que fui informado (a) dos objetivos da pesquisa, de maneira clara e detalhada, e esclareci minhas dúvidas. Sei que em qualquer momento **poderei solicitar novas informações e modificar minha decisão**, se assim desejar, bem como terei acesso ao registro de consentimento sempre que solicitar. A pesquisadora certificou-me de que minha **identidade será preservada**. Em caso de dúvidas **poderei chamar a pesquisadora responsável**, Eliana Zen, pelo e-mail [eliana.zen@inf.ufpel.edu.br](mailto:eliana.zen@inf.ufpel.edu.br) e/ou telefone (55) 996033120. Da mesma forma, afirmo que é de meu conhecimento que todos os dados a meu respeito serão mantidos sob sigilo e responsabilidade da pesquisadora responsável. Diante do exposto, declaro que concordo em participar deste estudo, assinando este TCLE em duas vias, ficando de posse de uma delas. Declaro também que me foi dada a oportunidade de ler e esclarecer as minhas dúvidas.

São Vicente do Sul - RS, \_\_\_/\_\_\_/\_\_\_\_\_

---

Assinatura do participante da pesquisa

---

Eliana Zen  
(Pesquisadora responsável)