

**INVESTIGAÇÃO SOBRE A APLICABILIDADE
DOS MÉTODOS DE AVALIAÇÃO DE
COMUNICABILIDADE AO DOMÍNIO
EDUCACIONAL**

ERICA RODRIGUES DE OLIVEIRA

INVESTIGAÇÃO SOBRE A APLICABILIDADE
DOS MÉTODOS DE AVALIAÇÃO DE
COMUNICABILIDADE AO DOMÍNIO
EDUCACIONAL

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

ORIENTADORA: RAQUEL OLIVEIRA PRATES

Belo Horizonte - MG

Junho de 2010

© 2010, Erica Rodrigues de Oliveira.
Todos os direitos reservados.

Oliveira, Erica Rodrigues de
O48i Investigação sobre a Aplicabilidade dos Métodos de
Avaliação de Comunicabilidade ao Domínio Educacional
/ Erica Rodrigues de Oliveira. — Belo Horizonte - MG,
2010

xx, 206 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de
Minas Gerais

Orientador: Raquel Oliveira Prates

1. Interfaces de usuário (Sistema de computador) -
Tese. Interação homem-máquina - Tese. Informática na
educação - Tese. I. Título.

CDU 519.6*75(043)



UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Investigação sobre a aplicabilidade dos métodos de avaliação de comunicabilidade ao domínio educacional

ERICA RODRIGUES DE OLIVEIRA

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROFA. RAQUEL OLIVEIRA PRATES - Orientadora
Departamento de Ciência da Computação - UFMG

DRA. CARLA FARIA LEITÃO
Pesquisadora Associada do Grupo de Pesquisa em Engenharia
Semiótica - SERG/DI/PUC-RJ

PROF. CLARINDO ISAIAS PEREIRA DA SILVA E PÁDUA
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 07 de junho de 2010.

Agradecimentos

Primeiramente a Deus pela oportunidade da existência, da saúde e do estudo.

À minha mãe, pelo amor incondicional, pela presença constante, pela dedicação sem limites e por sempre acreditar em mim e me motivar na busca pelos meus sonhos. Ao meu pai, que se faz presente com seu exemplo de determinação e perseverança. Ao meu irmão pela amizade, pelo carinho, por sempre confiar em mim e em meu potencial.

Ao Wemerson, pela compreensão e carinho. À minha filha Yasmin, que ainda em meu ventre, me fez companhia na escrita deste texto e, diariamente, vem me ensinando o significado do amor incondicional.

À minha amiga e orientadora, professora Raquel Prates, que se mostrou, além de uma excelente profissional, uma mulher ímpar. Muito obrigada por todo o apoio dado no direcionamento deste trabalho e, principalmente, pelo carinho, amizade, compreensão e paciência.

Aos membros da banca, Carla Leitão e Clarindo de Pádua, que muito enriqueceram o trabalho com suas contribuições.

À professora Maria Augusta Nelson, pelo incentivo à pesquisa e a ingressar na pós-graduação do DCC/UFMG.

Aos meus amigos e familiares, pela presença, apoio e participação em todas as etapas da minha vida.

Aos membros do PENSi, pelo companheirismo e troca de experiência. Aos colegas Luiz Luz e Gisleide Aidano, que muito contribuíram para o desenvolvimento deste trabalho.

Ao meu amigo Luiz Cantoni, pela amizade e pelo grande incentivo dado desde o início do curso de computação. Ao colega Sérgio Dias, por seu exemplo de esforço e dedicação.

Aos participantes desta pesquisa, pela gentil e voluntária contribuição. À FAPEMIG, pelo apoio financeiro.

Finalmente, agradeço à Túlia, Sheila e Renata, funcionárias do DCC/UFMG, por serem tão prestativas e mostrarem interesse em ajudar os alunos sempre que necessário.

Resumo

Cada vez mais professores têm feito uso da tecnologia como ferramenta para facilitar o processo de ensino e aprendizagem. O uso crescente desta tecnologia de apoio ao ensino presencial traz novos desafios para a concepção e avaliação de ambientes educacionais, especialmente para o design de interfaces. Dentre estes desafios, está a necessidade de avaliar a aprendizagem por parte do aluno, bem como a adequação deste sistema para uso no ensino presencial. Além disso, é necessário envolver aluno, professor e projetista no design e avaliação deste domínio. Nesse contexto, o objetivo deste trabalho é investigar a aplicabilidade dos métodos de avaliação propostos pela teoria da Engenharia Semiótica, como forma de avaliar a comunicabilidade de sistemas educacionais, considerando as três dimensões: aluno, professor e projetista. Por meio do levantamento bibliográfico de trabalhos que realizaram a avaliação de sistemas educacionais, verificou-se que a maioria destes trabalhos enfatiza os problemas de interação e muitos não consideram aspectos relacionados ao conteúdo, ao modelo pedagógico envolvido e ao processo de aprendizagem do aluno. Neste trabalho, considerou-se a aprendizagem do aluno como parte da avaliação. Para avaliação com o aluno, utilizou-se o Método de Avaliação de Comunicabilidade (MAC), enquanto para a avaliação com o projetista, o Método de Inspeção Semiótica (MIS) foi utilizado. Como nenhum destes métodos permitia a análise da comunicabilidade sob a perspectiva do professor, o Método de Inspeção Semiótica Intermediado (MISI) foi proposto. Ao utilizar estes métodos na avaliação dos sistemas educacionais inspecionados, identificou-se, além de problemas relacionados à interação, problemas relacionados ao conteúdo e à aprendizagem dos alunos. Dentre os resultados obtidos, destacam-se a importância de envolver diferentes usuários na avaliação do domínio educacional para garantir uma melhor comunicabilidade do sistema, assim como é necessário considerar aspectos do conteúdo, do modelo pedagógico e do processo de aprendizagem envolvidos na interação do aluno com o sistema.

Abstract

More and more teachers have made use of technology as a tool to facilitate the process of teaching and learning. The increasing use of this technology in the classroom brings new challenges for the design and evaluation of educational environments, especially for the design of interfaces. Among these challenges is the need to assess learning by students, as well as the suitability of this system for use in the classroom. Furthermore, it is necessary to involve the student, teacher and designer in the design and evaluation of this area. In this context, our objective is to investigate the applicability of evaluation methods proposed by the Semiotic Engineering theory in evaluating the communicability of educational systems, considering the involvement of students, teachers and designer. In a bibliographical survey of studies regarding assessment of educational systems, it was found that most of these studies emphasize the problems of interaction and many do not consider student learning. In this work, we considered the student's learning as part of the assessment. For evaluation with students, the communicability evaluation method (CEM) was considered, while for the evaluation with the designer, the semiotic inspection method (SIM) was investigated. As none of these methods allow for collecting data on the teacher's perspective or experience, the semiotic inspection intermediated method (SIIM) was proposed. By applying these methods in the evaluation of three different educational systems it was possible not only to identify interaction problems in these systems, but also collect indicators on their support to student learning. Among the results, we highlight the importance of involving different stakeholders in evaluating educational systems well as to consider educational aspects in the evaluation.

Lista de Figuras

3.1	Exemplo de signo estático	29
3.2	Exemplo de Signo Dinâmico	29
3.3	Bipide: exemplo de signo metalinguístico tooltip	30
3.4	Visão Geral do Método de Inspeção Semiótica [de Souza et al., 2010]	34
3.5	Passos do MAC [Mattos, 2010]	39
4.1	Tela principal do VCalc	47
4.2	ProfesSort: tela principal	48
4.3	Bipide: Tela de Simulação	50
4.4	VCalc: Objetivo Educacional	53
4.5	VCalc: Janela para Selecionar Nova Área	53
4.6	VCalc: Opção de zoom	54
4.7	ProfesSort: Níveis de Erro	62
4.8	ProfesSort: Histórico da Execução do Algoritmo	62
4.9	ProfesSort: Ausência de Indicação de Funcionalidade do Signo	63
4.10	ProfesSort: Ações a tomar do método HeapSort	63
4.11	Bipide: Telas de Ajuda	68
4.12	Bipide: Tela Principal	70
4.13	Bipide: Erro de Sintaxe	71
5.1	ProfesSort: etiqueta Socorro, referente ao conteúdo!	83
6.1	Método de Explicitação do Discurso Subjacente - MEDS	99
6.2	Visão Geral do MISI	100
6.3	Método de Inspeção Semiótica Intermediado - MISI	102
6.4	VCalc: Movimentação gráfica	107
6.5	ProfesSort: Opção de Código	116
7.1	Tabela de Classificação dos Métodos da Engenharia Semiótica	128

Lista de Tabelas

3.1	Descrição das expressões para etiquetagem do MAC - Parte A	41
3.2	Descrição das expressões para etiquetagem do MAC - Parte B	42
3.3	Descrição das expressões para etiquetagem do MAC - Parte C	43

Sumário

Agradecimentos	vii
Resumo	ix
Abstract	xi
Lista de Figuras	xiii
Lista de Tabelas	xv
1 Introdução	1
2 Revisão sobre Métodos de Avaliação de Ambientes Educacionais	5
2.1 Avaliações Envolvendo Alunos	6
2.1.1 Novos Métodos	6
2.1.2 Métodos Adaptados	9
2.1.3 Métodos Existentes	10
2.2 Avaliações Envolvendo Professores	13
2.2.1 Novos Métodos	13
2.2.2 Métodos Adaptados	14
2.2.3 Métodos Existentes	15
2.3 Avaliações Envolvendo Especialistas	16
2.3.1 Novos Métodos	16
2.3.2 Métodos Existentes	18
2.4 Classificação de avaliação de ambientes educacionais	20
3 A Engenharia Semiótica e seus Métodos de Avaliação	27
3.1 Teoria da Engenharia Semiótica	27
3.2 Método de Inspeção Semiótica	32
3.2.1 Passos do MIS	33

3.3	Método de Avaliação de Comunicabilidade	36
3.3.1	Passos do MAC	36
4	Análise do Método de Inspeção Semiótica Para o Domínio Educa-	
	cional	45
4.1	Metodologia	45
4.2	VCalc	46
4.3	ProfesSort	47
4.4	Bipide	49
4.5	Avaliação Com a Participação do Especialista	50
4.5.1	Aplicação do MIS no VCalc	51
4.5.2	Aplicação do MIS no ProfesSort	58
4.5.3	Aplicação do MIS no Bipide	66
4.6	Aplicabilidade do MIS no Domínio Educacional	73
5	Análise do Método de Avaliação de Comunicabilidade para o	
	Domínio Educacional	75
5.1	Aplicação do MAC no VCalc	76
5.1.1	Preparação e Execução do Teste	76
5.1.2	Análise dos Dados	78
5.2	Aplicação do MAC no ProfesSort	80
5.2.1	Preparação e Execução do Teste	80
5.2.2	Análise dos Dados	81
5.3	Aplicação do MAC no Bipide	87
5.3.1	Preparação e Execução do Teste	87
5.3.2	Análise dos Dados	89
5.4	Discussão	91
5.5	Considerações sobre o uso MAC para o Domínio Educacional	93
6	Método de Inspeção Semiótica Intermediado	97
6.1	Visão Geral	98
6.2	Descrição dos Passos	100
6.2.1	Passos do MISI	101
6.3	Estudos de Caso	106
6.3.1	VCalc	106
6.3.2	ProfesSort	113
6.4	Discussão da Aplicação do MISI	122

7 Considerações Finais	125
7.1 Contribuições	125
7.2 Trabalhos Futuros	128
Anexo A Material de Inspeção do MIS no VCalc	131
Anexo B Material de Aplicação do MAC no VCalc	133
Anexo C MAC: Relatório da Etiquetagem do VCalc	139
Anexo D Material de Aplicação do MAC no ProfesSort	149
Anexo E MAC: Relatório da Etiquetagem do ProfesSort	159
Anexo F Material de Aplicação do MAC no Bipide	165
Anexo G MAC: Relatório da Etiquetagem do Bipide	175
Anexo H Roteiro do MISI - Sistema VCalc	183
Anexo I Roteiro do MISI - Sistema ProfesSort	189
Referências Bibliográficas	199

Capítulo 1

Introdução

O uso adequado de tecnologias digitais na prática didático-pedagógica vem trazendo diversos benefícios às instituições de ensino e à sociedade como um todo. Cada vez mais professores têm utilizado *software* para facilitar o processo de ensino e aprendizagem. Este uso crescente de tecnologia de apoio ao ensino presencial já aponta para os novos desafios e especificidades que ambientes educacionais trazem para as etapas de projeto e avaliação de *software*, em particular para suas interfaces [Quintana et al., 2001]. Estas questões específicas muitas vezes requerem que o projetista da interface desconsidere ou vá contra diretrizes de Interação Humano-Computador (IHC). Por exemplo, o sistema pode deixar o aluno errar, em oposição a prevenir erros, para que ele entenda melhor um conceito ou método [Lewis et al., 1998].

Do ponto de vista da avaliação, um dos desafios que o domínio educacional traz é que, além da qualidade da interação a ser avaliada, deve-se avaliar também o aprendizado por parte do aluno, e a adequação do sistema para apoio ao ensino presencial, dentro e fora da sala de aula. Assim, alguns pesquisadores na área de IHC argumentam que a simples extensão para o domínio educacional de métodos de avaliação de interação consolidados pode não ser suficiente e novos métodos que considerem os objetivos educacionais precisariam ser propostos [Jones et al., 1999; Quintana et al., 2001; Squires & Preece, 1999; Tselios et al., 2008].

Grande parte dos métodos consolidados hoje em IHC têm sua origem em conhecimento empírico (e.g. testes de usabilidade e avaliação heurística). No entanto, vários pesquisadores de IHC têm alertado para a importância de se ter uma teoria para a área que forneça uma visão integrada da área de IHC e fundamente pesquisa e métodos que contribuam para o seu avanço [Carroll, 2003; de Souza, 2005; Greenberg & Buxton, 2008]. Assim, neste trabalho olhamos para o desafio de avaliação de ambientes educacionais sob a lente da teoria da Engenharia Semiótica [de Souza, 2005; Prates &

Barbosa, 2007; de Souza & Leitão, 2009].

A Engenharia Semiótica (EngSem) [de Souza, 2005; Prates & Barbosa, 2007; de Souza & Leitão, 2009] é uma teoria de IHC que entende a interface como uma mensagem sendo transmitida pelo projetista ao usuário. À medida que o usuário interage com o sistema ele entende a quem o sistema se destina, que problemas resolve e por que, e como interagir com ele. Neste quadro teórico, a qualidade de uso focada é a comunicabilidade, a propriedade de um sistema de transmitir ao usuário os princípios de interação e decisões que o guiaram [de Souza, 2005; Prates & Barbosa, 2007; de Souza & Leitão, 2009]. Atualmente existem dois métodos que permitem a avaliação da propriedade da comunicabilidade: Método de Avaliação de Comunicabilidade (MAC) [Prates et al., 2000] e Método de Inspeção Semiótica (MIS) [de Souza et al., 2006; de Souza & Leitão, 2009; de Souza et al., 2010].

O MAC permite a avaliação da comunicabilidade do sistema a partir da interação usuários-sistema [de Souza, 2005; Prates & Barbosa, 2007; de Souza & Leitão, 2009]. O MIS, por sua vez, é um método de inspeção em que a avaliação é feita por especialistas em Engenharia Semiótica [de Souza, 2005; de Souza et al., 2006; de Souza & Leitão, 2009; de Souza et al., 2010].

A motivação para realização deste trabalho residiu no crescimento do uso de sistemas educacionais de apoio ao ensino presencial, como forma de apoiar a aprendizagem dos alunos e os novos desafios que o domínio educacional trazem para a área de IHC. Ao avaliar este domínio, é importante considerar, além de aspectos relativos à interação, aspectos da aprendizagem dos alunos. Neste contexto, estender os tradicionais métodos de avaliação de interfaces pode não ser suficiente e novos métodos que consideram os objetivos educacionais precisariam ser propostos [Jones et al., 1999; Squires & Preece, 1999].

A EngSem é uma teoria que tem como foco a comunicação feita do projetista para o usuário, em tempo de interação [de Souza, 2005]. O conteúdo dessa comunicação do projetista para o usuário transcende os temas relacionados aos objetivos dos usuários, seus planos e suas ações, permitindo que se considere outros elementos envolvidos na interação (e.g. a visão do projetista sobre a interface). No entanto, no domínio educacional além da avaliação por especialistas e usuários (i.e. alunos), pode ser necessário envolver o professor na avaliação [Pardo et al., 2006] para que ele faça uma apreciação da adequação do sistema para o seu processo de ensino e disciplina. Neste caso, nenhum dos dois métodos fundamentados na EngSem é adequado.

Sob a lente da Engenharia Semiótica, propõe-se, neste trabalho, fazer uso científico dos métodos de avaliação para gerar conhecimento sobre esses métodos. A avaliação da interface de sistemas educacionais de apoio ao ensino presencial foi feita con-

siderando três diferentes participantes: projetista, aluno e professor. Assim, analisou-se a aplicabilidade do MIS e MAC para o contexto educacional.

A aplicação do MIS em ambientes educacionais demonstrou que este método identifica, além de problemas de interação, problemas específicos do sistema educacional inspecionado. Isto contribui para a área de IHC à medida que demonstra que um método de avaliação de sistemas interativos pode ser aplicado a qualquer domínio, sem a necessidade de fazer adaptações.

A aplicação do MAC no domínio educacional apontou para a necessidade de distinguir etiquetas de interação e conteúdo, de forma a melhorar a qualidade da comunicação projetista-aluno. Assim, em relação à interface, conforme esperado, o MAC permitiu identificar rupturas de interação vivenciadas pelos alunos durante os testes. Em relação à aprendizagem verificou-se que o MAC pode ser utilizado pelo professor, no acompanhamento do aprendizado dos alunos.

Para a avaliação sob a perspectiva do professor propôs-se um novo método denominado Método de Inspeção Semiótica Intermediada - MISI [de Oliveira et al., 2008]. O MISI permite fazer a avaliação da comunicabilidade de um sistema educacional sob o ponto de vista do professor, por meio da aplicação do MIS [de Souza et al., 2006] utilizado como base para definir o roteiro de uma interação e entrevista guiada, com base no Método de Explicitação de Discurso Subjacente (MEDS) [Nicolaci-da Costa et al., 2004].

A apreciação dos métodos da EngSem para o domínio educacional traz benefícios para este domínio, à medida que permitem identificar questões envolvidas no processo de comunicação.

Este trabalho traz contribuições para duas áreas distintas: interação humano-computador e informática e educação. Para a área de informática e educação o trabalho contribui à medida que apresenta novos métodos ou métodos adaptados capazes de avaliar questões de interesse em um ambiente educacional, sob a ótica do especialista, aluno e professor. Para a área de IHC e, em especial, teoria da EngSem, a contribuição se dá pela proposta ou combinação métodos que permitem a avaliação da comunicabilidade adaptadas para o domínio educacional.

A seguir é apresentada a organização deste trabalho. No capítulo 2, é apresentada uma revisão de trabalhos sobre avaliação de ambientes educacionais, visando mostrar o que os demais pesquisadores têm feito em termos de avaliação no domínio educacional, considerando as perspectivas de projetista, aluno e professor. Ainda neste capítulo, elaboramos uma tabela para classificar os trabalhos sobre avaliação realizados.

No capítulo 3 é apresentada a fundamentação teórica, onde é feita uma breve introdução sobre a teoria da Engenharia Semiótica, seus principais conceitos e os seus

métodos de avaliação. O capítulo 4 apresenta a metodologia utilizada, os sistemas educacionais utilizados nos estudos de caso e detalha o resultado da aplicação do MIS nos sistemas avaliados. O capítulo 5 trata da aplicabilidade do MAC no domínio educacional e apresenta as considerações sobre o uso deste método no domínio educacional.

Em seguida, o capítulo 6 apresenta o MISI [de Oliveira et al., 2008], um novo método de avaliação que permite ao professor avaliar a comunicabilidade de um sistema interativo. Também são apresentados os estudos de caso realizados para validar este método. Finalmente, no capítulo 7, são feitas as considerações finais deste trabalho e delineados os trabalhos futuros.

Capítulo 2

Revisão sobre Métodos de Avaliação de Ambientes Educacionais

Este capítulo consiste em apresentar uma revisão sobre métodos de avaliação de interfaces existentes para o domínio educacional, considerando três dimensões: aluno, professor e especialista.

Para realização da avaliação sobre trabalhos voltados para ambientes educacionais, foi feita uma revisão bibliográfica em que se buscou métodos de avaliação de interfaces considerando as três dimensões, aluno, professor e especialista. Nesta busca, foram identificadas propostas de novos métodos, adaptações de métodos existentes, bem como métodos descritos em estudos de caso, onde o foco era avaliação de ambientes educacionais tanto para o apoio presencial, em sala de aula e fora dela, quanto para educação a distância.

As seções 2.1, 2.2 e 2.3 apresentam os trabalhos considerando, respectivamente, os envolvidos aluno, professor e especialista. Além disso, os trabalhos avaliados foram classificados como:

- *Proposta de novo método*: refere-se aos trabalhos que propuseram um novo método de avaliação para avaliar a qualidade da interface de sistemas educacionais;
- *Proposta de novo critério e/ou adaptação de método*: refere-se aos trabalhos que adaptaram ou propuseram novos critérios de avaliação para os tradicionais métodos de avaliação de interface;

- *Aplicação de métodos existentes*: refere-se a estudos de casos realizados onde utilizou-se os métodos de avaliação de interface da forma como eles são propostos, com a finalidade de avaliar o sistema definido.

Após a apresentação destes métodos, uma tabela foi elaborada (ver seção 2.4) com as características consideradas como relevantes para o domínio educacional.

2.1 Avaliações Envolvendo Alunos

A avaliação de um sistema é uma etapa importante do processo de desenvolvimento e avaliar a interface de sistemas educativos é uma tarefa complexa, pois deve-se considerar, dentre outros aspectos, o aprendizado de um conteúdo [Lanzilotti et al., 2006; Quintana et al., 2002]. Com o crescente uso do computador como uma ferramenta de apoio à aprendizagem, vários pesquisadores têm proposto novos métodos ou adaptações para avaliação de ambientes educacionais [Squires & Preece, 1999]. Nesta seção focamos os trabalhos que envolvem necessariamente a perspectiva do aluno, permitindo que se tenha uma apreciação do sistema sob seu ponto de vista.

A seguir, são apresentados os trabalhos analisados e a avaliação proposta/realizada. Esses métodos foram classificados como: (i) novos métodos, (ii) métodos adaptados e (iii) métodos existentes.

2.1.1 Novos Métodos

Matera et al. [2002] propuseram um novo método de avaliação denominado Avaliação Sistemática de Usabilidade (SUE) que combina avaliação por inspeção com avaliação com alunos. O SUE propõe um conjunto de tarefas abstratas (AT) que guiam as atividades do especialista, descrevendo quais objetos da aplicação devem ser analisados e quais ações realizar durante a inspeção. Assim, o especialista identifica pontos críticos para serem avaliados com os alunos.

A avaliação com os alunos tem objetivos de curto e longo prazo. No curto prazo o objetivo é validar o projeto revisado com base no resultado de inspeção, observando o impacto de algumas situações críticas em um contexto real de uso e com usuários reais, produzindo *feedback* de avaliação do sistema a ser avaliado. No longo prazo, o objetivo é identificar quais são as soluções de concepção mais adequada, e melhorar o quadro conceitual da avaliação, por meio da definição de novos atributos de usabilidade, como *eficiência*, que refere-se às funções previstas pelo aplicativo e *aprendizagem*, que refere-

se aos recursos do aplicativo que permite aos usuários inexperientes utilizá-lo e atingir um nível máximo de desempenho, bem como novas tarefas abstratas.

Zaharias [2006] propôs um novo método para avaliar a usabilidade de sistemas de educação a distância e argumenta sobre a dificuldade de avaliar a usabilidade neste tipo de sistemas. Segundo ele, avaliar a usabilidade deste tipo de sistema não é uma tarefa trivial, pois o aumento da diversidade de alunos, os avanços tecnológicos e as mudanças nas tarefas de aprendizagem apresentam desafios significativos que dificultam a avaliação. O autor afirma ainda que os tradicionais métodos de usabilidade como eficiência, eficácia e satisfação não são suficientes para avaliação do domínio educacional. O autor argumenta ainda sobre a necessidade de avaliar o estado afetivo do aluno enquanto ele interage com o sistema. Segundo ele, tem-se argumentado que este efeito é o combustível que o aluno traz para o ambiente de aprendizagem, associado às causas da aprendizagem. Assim, a necessidade de melhorar as prioridades internas do aluno é descrita como motivação do aluno para aprender. A motivação para aprender é proposta como uma âncora para o desenvolvimento de novas medidas de usabilidade para sistemas de educação a distância. Nesse trabalho, o autor propôs um questionário baseado no teste de usabilidade para aplicações *e-learning*. Este método coloca o aluno como um aprendiz e integra aspectos cognitivos (percepção da usabilidade dos usuários) e afetivos (motivação dos usuários para aprender).

O método proposto consiste em (i) elaborar o questionário, (ii) definir o ponto de amostragem, (iii) realizar o teste piloto, (iv) gerar uma nova versão do questionário. Na elaboração do questionário, utilizou-se um questionário psicométrico para medir a motivação do aluno e sua percepção sobre a usabilidade do sistema. O questionário continha questões sobre navegabilidade, consistência, *design* visual, interatividade, recursos e conteúdo, *feedback*, guia para o aluno, estratégias de aprendizagem e uso de mídias. A partir destes parâmetros para sistemas de educação a distância, foi proposta uma nova dimensão para avaliação de usabilidade: motivação para aprender.

A partir da execução do teste piloto, o questionário foi disponibilizado na internet para avaliação dos alunos com experiência em sistemas de educação a distância. Os dados obtidos foram analisados e alguns ajustes foram feitos, incluindo a reformulação de alguns itens. Este processo serviu para validar o questionário, que totalizou 49 itens, sendo 39 para medição da usabilidade e 10 para medição da motivação para aprender. As respostas basearam-se na escala *likert*, indo de 1 a 5, onde 5 equivalia a concordo plenamente. Segundo o autor, a proposta deste questionário baseado no método de avaliação de usabilidade estende a prática corrente de medida de usabilidade e propõe a motivação para aprender como um novo tipo de medida de usabilidade.

Quintana et al. [2002] mostraram a importância da utilização de *scaffolds*¹ como suporte ao aprendizado dos alunos em sistemas educacionais por meio da avaliação com alunos. *Scaffolds* são ferramentas, estratégias e recursos que apóiam alunos em dificuldades específicas de conteúdo, de forma a atingir um nível mais elevado de compreensão [Quintana & Fishman, 2006; Brush & Saye, 2001].

O estudo foi realizado em duas etapas usando um sistema que tratava da investigação da qualidade do ar e permitia aos alunos planejar, coletar, visualizar e modelar dados. Inicialmente, verificou-se como os alunos utilizavam o sistema. A partir desta observação inicial e com o auxílio dos pesquisadores que usavam o sistema, incorporou-se, na ferramenta um conjunto de *scaffolds*. A inclusão destes *scaffolds* tinha como objetivo verificar como tais apoios fornecidos auxiliariam os alunos na execução das atividades realizadas no sistema. A elaboração de uma interface de *software* que atenda às necessidades dos usuários consiste em:

- analisar o trabalho prático para identificar todos os aspectos que serão necessários no desenvolvimento da ferramenta (como os alunos realizam tal atividade sem o *software*);
- analisar as necessidades dos alunos: identificar áreas no trabalho prático onde o aluno tem dificuldade,
- determinar a estratégia de *scaffold* que vai suportar a aprendizagem de determinado conteúdo pelos alunos, além de especificar o caminho em que essas estratégias serão fisicamente implementadas;
- avaliar a eficiência do *scaffold*, ou seja, como os alunos usam os *scaffolds* para apoiar a aprendizagem.

Naquele trabalho, participaram das avaliações 6 alunos, sendo 5 do sexo masculino e 1 do sexo feminino. Esses alunos utilizaram o sistema em sala de aula, de quatro a cinco horas por semana, num período de 9 semanas para realizar investigações científicas sobre a qualidade do ar. Inicialmente, os alunos trabalharam em pares para realizar as investigações sobre a qualidade do ar da cidade de Michigan. Para realização desta atividade, os alunos utilizavam o sistema Symphony para planejar, coletar e visualizar dados, construir e analisar modelos, e escrever conclusões suportadas por evidências obtidas em atividades anteriores. As questões respondidas eram abertas, permitindo

¹Em inglês, *scaffold* significa andaime, apoio. No contexto educacional, representa o apoio ao aprendizado descrito nesta definição. Neste trabalho, foi usado o termo em inglês pelo fato dos autores não identificarem um termo em português adequado para traduzi-lo.

aos alunos dar diferentes respostas a partir de sua própria investigação. Em seguida, os professores revisaram e criticaram as conclusões feitas pelos alunos, e os alunos podiam refinar a investigação para então passar à próxima questão.

As avaliações foram conduzidas por projetistas e professores a partir de um roteiro e toda a interação dos alunos foi gravada, para análise posterior. Os autores então propuseram um conjunto de critérios de usabilidade específicos para avaliar sistemas com *scaffolds*, a saber: gerenciamento de tempo, acessibilidade, uso, eficiência, acurácia, progressão e reflexão sobre a tarefa executada. Os vídeos gerados no teste com os usuários foram analisados segundo estes critérios. Assim, os autores buscaram descrever como os alunos usaram os *scaffolds* e como melhoraram nas tarefas a partir deste uso. Os resultados indicaram alto índice de utilização destes apoios, pelos alunos, no auxílio à execução das tarefas, em especial os *scaffolds* gráficos.

2.1.2 Métodos Adaptados

Masemola & De Villiers [2006] adaptaram o teste de usabilidade para aplicá-lo em um ambiente de ensino a distância para aprendizagem matemática. Neste trabalho, os autores identificaram aspectos específicos de avaliação para sistemas de ensino a distância que auxiliam na geração de um *framework* de avaliação para esse tipo de domínio. Para realização de teste com alunos, os autores usaram o teste de usabilidade e o *think aloud*, seguido de entrevista pós-teste. Participaram do teste 6 (seis) usuários, sendo quatro do sexo masculino e dois do sexo feminino, na faixa etária de 17 a 30 anos. Destes alunos, dois deles tinham experiência avançada com computador, enquanto dois tinham experiência mínima e dois possuíam experiência média. Em relação à metodologia utilizada, os participantes eram informados sobre todo o procedimento da avaliação e que poderiam parar a qualquer momento, fazer perguntas ou pedir ajuda.

O *framework* proposto não continha características específicas para avaliar o contexto educacional. No entanto, na análise, foram considerados fatores específicos ao domínio - de forma diferente que seriam em outros domínios (e.g. o tempo menor nem sempre ser bom, nem sempre é desejável minimizar erros). A análise obteve três tipos de indicadores relacionados à aprendizagem, à utilidade do sistema e sobre a interface. Com relação à aprendizagem, verificou-se que nas situações em que o usuário errava, o sistema oferecia *feedback*, de forma a auxiliar o aluno a acertar. Percebeu-se que alguns alunos reparavam seus erros enquanto outros selecionavam a resposta errada de modo a obter a resposta correta mais rápida. Sobre a utilidade do sistema, os dados observados como sucesso na execução das tarefas (completou ou não a tarefa), a lista de problemas de usabilidade identificados e o número de interrupções do avaliador in-

dicam que o sistema apresentou poucos problemas. Em relação à interface, por meio da baixa taxa de erros encontrados na interface pelos alunos, foi observada uma elevada taxa de conclusão das tarefas e a pontuação satisfatória no teste. Para medir a satisfação dos alunos, realizou-se uma entrevista pós-teste. No geral, o sistema mostrou-se satisfatório para os alunos.

Ardito et al. [2006] ressaltam que um dos maiores desafios para os projetistas da área de IHC é desenvolver ferramentas capazes de envolver alunos iniciantes e apoiar a sua aprendizagem. Os sistemas educacionais devem considerar as diferentes formas de aprendizado dos alunos e assegurar que a interação do aluno seja natural e intuitiva. Isso pode exigir a revisão de paradigmas de interação tradicionais para fornecer maior flexibilidade e adaptabilidade, adequando-se às peculiaridades deste domínio. Neste sentido, deve-se haver uma sinergia entre o processo de aprendizagem e a interação do aluno com o sistema.

Como forma de avaliar a usabilidade e a aprendizagem dos alunos em sistemas de ensino a distância, Ardito et al. [2006] combinaram o método de inspeção (SUE), proposto por Matera et al. [2002] e o teste de usabilidade. Por meio desta combinação, os autores definiram um conjunto de critérios e diretrizes para avaliar ambientes de ensino a distância. Os autores descreveram a metodologia de avaliação que explora esses critérios e diretrizes para conduzir as atividades dos avaliadores. Os critérios avaliados foram: (i) análise das necessidades da aprendizagem; (ii) definição dos objetivos da aprendizagem; (iii) organização do conteúdo didático; (iv) seleção da metodologia de ensino apropriada; (v) avaliação da aprendizagem. Além desses critérios, os autores também avaliaram a acessibilidade dos sistemas de ensino a distância.

A inspeção foi realizada utilizando-se a metodologia SUE [Matera et al., 2002], que visa definir um quadro geral de avaliação de usabilidade por avaliadores. O SUE combina atividades de inspeção e teste com usuário, de forma a obter uma avaliação mais confiável, por considerar diferentes dimensões, interação e questões específicas, que referem-se à proposta da aplicação. A avaliação foi conduzida por especialistas que inspecionaram o sistema selecionado a partir da definição das tarefas, utilizando os critérios e diretrizes definidos anteriormente. Em seguida, realizou-se teste com os usuários para uma avaliação mais ampla.

2.1.3 Métodos Existentes

Com o objetivo de identificar problemas de usabilidade e interatividade, e suas possíveis consequências para o domínio educacional, Freitas & Dutra [2009] realizaram um estudo de caso que tinha como objetivo investigar, por meio de teste com usuário baseado

em questionários, a opinião deles com relação aos ambientes virtuais de aprendizagem, Moodle e WebAula. Neste trabalho, os autores argumentam que, apesar da usabilidade não ser a questão mais importante em um ambiente educacional, ela pode determinar ou ser o estímulo inicial e continuado para o aluno na utilização do sistema. A avaliação realizada nos sistemas inspecionados encontrou problemas relacionados à usabilidade como: navegação, *links*, nome de categorias, *pop-ups* e procura. Entretanto, não foram considerados aspectos relacionados ao conteúdo do site, uma vez que o conteúdo dos cursos disponibilizados nos sistemas eram elaborados por equipes diferentes. Os cursos oferecidos pela WebAula eram elaborados por equipes multidisciplinares, enquanto os cursos oferecidos pelo Moodle tinham o conteúdo elaborado pelos professores das disciplinas.

Brush & Saye [2001] a fim de demonstrar os benefícios do uso de *scaffolds* no domínio educacional, realizaram um estudo de caso para apresentar os tipos de *scaffolds* embutidos em sistemas educacionais, como eles foram usados pelos alunos e quais apoios foram mais usados por eles. Os *scaffolds* embutidos no sistema avaliado foram conceitual (ajuda o aluno a determinar o que considerar na resolução de um problema) e metacognitivo (associa-se ao gerenciamento da aprendizagem individual do aluno).

Participaram do estudo 1 professor e 36 estudantes, divididos em grupos de 4 alunos. Na análise dos dados, utilizou-se o *log* do sistema, a fim de verificar quais apoios os alunos usaram. Para consolidar os resultados obtidos na análise dos *logs*, realizou-se uma entrevista com um membro de cada grupo. As perguntas referiam-se especificamente à percepção dos alunos quanto à eficácia dos apoios incluídos no sistema. O resultado do estudo indicou que alguns dos *scaffolds* demonstram-se mais eficientes que outros, devido à maior utilização pelos alunos. Observou-se que alguns *scaffolds* contribuíram para a realização das tarefas propostas, enquanto outros indicaram uma dificuldade por parte do aluno em integrar as informações fornecidas pelo sistema com o conhecimento prévio que ele possuía (ou não).

Iahad et al. [2004] apresentaram um estudo de caso feito a partir da avaliação de um teste *online* de múltipla escolha, num curso de *e-commerce*, por meio do paradigma de avaliação centrado no usuário. Este modelo consiste nas seguintes etapas: formular sentenças de intenção de resultados da aprendizagem, desenvolver/selecionar medidas de avaliação, criar experiências considerando o usuário, discutir e usar o resultado da avaliação para melhorar a aprendizagem. Esta avaliação baseou-se em critérios de funcionalidade e usabilidade. Baseado neste modelo, os usuários respondiam às questões do teste. O *feedback* consistia em fornecer ao usuário, apresentando *links* que explicavam a solução correta, para situações em que o aluno selecionava a resposta errada. Se acertava, o sistema emitia mensagens para motivá-lo. Após a realização do teste,

os usuários eram convidados a participar de uma avaliação do sistema. Porém, como não era obrigatório, apenas 47, dos 115 alunos que fizeram o teste, participaram da avaliação. A partir dessa avaliação foi possível obter indicadores de aprendizagem e interface. Em termos de aprendizagem, alguns alunos perceberam em que parte do conteúdo precisavam se dedicar mais. Sobre a interface, sugeriram alterações, principalmente com relação ao conteúdo (e.g. acrescentar mais explicações sobre a alternativa correta).

Rentróia-bonito et al. [2008] integraram um método empírico de avaliação de usabilidade testado com instrutores. Neste artigo, os autores enfatizam que uma condição básica para que sistemas de apoio ao aprendizado sejam usáveis é que o aluno seja capaz de concentrar-se no conteúdo, e não no sistema, o que demanda uma integração da avaliação centrada no usuário. Assim, as dimensões técnica e pedagógica, são desafios para a equipe de desenvolvimento. A proposta de avaliação consistiu no desenvolvimento de um protótipo do sistema para apoiar o processo de aprendizagem e avaliação baseada na experiência do usuário. Utilizou-se o método empírico de avaliação de usabilidade combinando avaliação técnica e teste com usuário, por meio do estudo de *logs* e indicadores de desempenho. Os alunos fizeram a avaliação de usabilidade a partir do preenchimento de um questionário. O resultado da pesquisa indicou que em sistemas de ensino a distância é mais adequado ir ao encontro das necessidades do processo instrucional e apoiar o comportamento e ação dos aprendizes. Com isso, a comunicação entre os envolvidos é crucial para uma aprendizagem efetiva.

Os trabalhos citados ressaltam a importância de se envolver o aluno no processo de avaliação de um sistema educacional, além de apontar para a necessidade de avaliar a aprendizagem do aluno envolvida no uso deste sistema [Zaharias, 2006; Lanzilotti et al., 2006; Iahad et al., 2004]. Nesse caso, permitir ao aluno errar pode ser importante para o seu processo de aprendizagem [Quintana et al., 2002] e analisar a eficiência com que um aluno conclui uma tarefa pode não ser uma medida adequada para este domínio [Masemola & De Villiers, 2006]. É importante que as interfaces educacionais façam sentido rapidamente, uma vez que o usuário poderá não fazer uso do sistema por um longo período de tempo. Elas também devem permitir ao aluno se concentrar na aprendizagem do conteúdo, ao invés de focar em aprender como usar o sistema [Zaharias et al., 2001]. Com isso, os tradicionais métodos de avaliação da qualidade de interface podem não ser suficientes e novos métodos (ou a adaptação dos métodos existentes) faz-se necessário para considerar as especificidades do domínio educacional [Jones et al., 1999; Squires & Preece, 1999; Quintana et al., 2001; Ardito et al., 2004; Hornbæk, 2006; Zaharias, 2006; Greenberg & Buxton, 2008].

2.2 Avaliações Envolvendo Professores

Diversos estudos têm sido feitos propondo novos métodos ou adaptações para ambientes educacionais [Squires & Preece, 1999]. Os métodos normalmente envolvem a equipe de *design* (projetistas ou avaliadores), alunos ou professores. Nesta seção, focamos nos trabalhos que envolvem o professor, permitindo que se tenha uma apreciação do sistema sob seu ponto de vista. Também foi feita uma classificação dos trabalhos que propuseram novos critérios de avaliação a partir de métodos de avaliação de interface existentes, ou adaptaram estes métodos para avaliar sistemas educacionais.

2.2.1 Novos Métodos

Visando integrar aspectos de usabilidade e aprendizagem, Squires & Preece [1999] propuseram um método de avaliação por inspeção para professores, por meio da adaptação das heurísticas, propostas por Molich & Nielsen [1990], para uma perspectiva de aprendizagem sócio-construtivista. Neste artigo, os autores sugerem uma abordagem que provê um modelo de avaliação preditiva para professores. Segundo os autores, a avaliação heurística é apenas uma parte da avaliação de usabilidade. Assim, o uso inadequado da avaliação heurística pode fazer com que o professor não cubra parte da avaliação, pois é necessário que o professor avalie o sistema por meio do seu conhecimento prévio, de como ele o apresentará aos alunos e de como os alunos poderão aprender com ele.

O modelo proposto por Squires & Preece [1999] consiste em analisar um sistema educacional considerando:

- necessidade de uma correspondência entre os modelos do projetista e do aluno;
- fidelidade da navegação;
- necessidade de considerar os níveis adequados de controle do aluno, o que resulta numa análise de controle do aprendiz e de responsabilidade compartilhada;
- prevenção de erros cognitivos;
- representação simbólica e compreensível para os alunos;
- apoio a aprendizagem;
- necessidade de estratégias para o reconhecimento de erros cognitivos;
- correspondência com o currículo escolar adotado.

Assim, este modelo oferece um apoio ao professor para analisar características importantes de serem consideradas num sistema educacional.

Outra proposta para apoiar o professor na avaliação de um sistema educacional foi a criação de um ponto de encontro através do ambiente Comunidade de Análise de *Software* Educativo (CASE) para uma comunidade de profissionais brasileiros - educadores e projetistas - interessados na análise e *design* de sistemas educativos [Lyra et al., 2003]. O ambiente permite ao professor avaliar um sistema educacional de interesse sob diferentes perspectivas: aprendizagem do aluno e currículo, aspectos técnicos e pedagógicos e prática docente. Todo usuário do sistema CASE pode cadastrar sistemas educativos e fazer avaliação de qualquer sistema cadastrado. As avaliações são realizadas por meio do preenchimento de um questionário, e disponibilizadas no ambiente CASE, para que outros usuários (professores, especialistas e visitantes) tenham acesso. A ferramenta procura atender a duas necessidades educacionais: (1) auxiliar o professor na seleção de sistemas educacionais e (2) fornecer ao projetista as análises feitas por educadores, a partir da prática efetiva em contextos de uso.

2.2.2 Métodos Adaptados

Squires & Preece [1996] apontam para a dificuldade que professores têm ao selecionar um sistema para uso e falam da necessidade de apoiá-los na escolha de um bom sistema educacional. Neste trabalho, os autores realizaram um estudo de caso baseado em inspeção de usabilidade em sistemas educacionais. Segundo os autores, a forma mais conhecida de realizar a avaliação preditiva é por meio de *checklists*. Entretanto, a análise realizada apontou as falhas da abordagem baseada em um *checklist*, específico para usabilidade, para integrar aspectos relacionados à usabilidade e questões educacionais. Assim, um modelo de avaliação que procura integrar estas características (usabilidade e questões educacionais) ajuda os professores a se concentrar em questões referentes ao aprendizado dos alunos e na usabilidade do sistema avaliado.

Neste trabalho, os autores utilizaram o modelo Jigsaw [Squires, 1994]. Este modelo consiste em integrar aspectos de usabilidade e questões educacionais. Segundo os autores, o uso de um sistema educacional é dependente de contexto, ou seja, não é possível avaliar uma aplicação educacional preditivamente sem referencia a um cenário educacional. Para isso, o modelo Jigsaw [Squires, 1994] consiste em três níveis. No nível 1, as tarefas de aprendizagem são pensadas como conceitos de aprendizagem individual que precisam ser específicas para um determinado tema ou que estão relacionados com a área geral de estudo. O nível 2 consiste em integrar conceitos específico e geral (relacionamento entre sistema e aplicação). Enquanto no nível 3 ocorre a integração

entre a aprendizagem e a tarefa operacional. Esses níveis levam a uma avaliação crítica sobre o quão bem as funções previstas no sistema e a usabilidade do seu *design* de interação corresponde às necessidades dos alunos ao usá-lo.

De Villiers [2004] fez uma revisão das heurísticas propostas por Squires & Preece [1999]. Nesta revisão, identificou-se a necessidade de duas novas categorias: categorias distintas, exclusivas para o ambiente avaliado; e capacidade do sistema para envolver os alunos e prender sua atenção. Além da criação destas novas categorias, o estudo identificou problemas relacionados à interação e à aprendizagem dos alunos.

2.2.3 Métodos Existentes

Pardo et al. [2006] por meio de uma avaliação de sistema envolvendo professor, especialista e aluno, mostrou a importância da participação do professor como facilitador em avaliação centrada no usuário, já que o professor tem melhores condições de entender o comportamento e as percepções dos alunos que o especialista. Para uma familiarização com os sistemas avaliados, os professores os utilizaram por uma semana. Realizaram-se oito avaliações que continham uma lista de tarefas executadas por alunos. Em cada avaliação, o professor ou o especialista atuava como facilitador. Apesar de utilizarem um roteiro, ambos faziam perguntas de acordo sua percepção. As percepções foram classificadas em: usabilidade, aprendizagem, ambas e outras. Essas avaliações foram gravadas e analisadas posteriormente pelo professor e especialista. A análise baseou-se no cruzamento das percepções do professor e especialista, de forma a comparar o comportamento deles durante a avaliação, bem como avaliar os sistemas. Pela análise concluiu-se que professores são melhores facilitadores, segundo o ponto de vista do aluno, na avaliação de ambientes educacionais.

Os trabalhos citados, direta ou indiretamente, consideram que várias decisões relativas a estratégias de ensino e planos pedagógicos cabem ao professor, e não ao projetista [Andres, 2005; Jones et al., 1999]. A importância do professor na avaliação de um sistema educacional é percebida nos esforços de adaptar os métodos para serem utilizados pelos professores. No entanto, a avaliação de sistemas educacionais requer habilidades que o professor geralmente não possui [Pardo et al., 2006] e preparação para considerar aspectos tecnológicos [Jones et al., 1999]. Portanto, para contar com a contribuição de todos os especialistas, é importante considerar as diferentes perspectivas - professor, aluno e projetista, de forma conjunta ou não, para que os aspectos relevantes relacionados à avaliação do sistema possam ser considerados.

2.3 Avaliações Envolvendo Especialistas

Grande parte dos tradicionais métodos de avaliação de interface são baseados na inspeção de especialistas em IHC. Esses métodos têm, sobretudo, um carácter formativo, uma vez que podem ser aplicadas por especialistas simulando o uso típico esperado do sistema, durante os estágios iniciais de um ciclo de *design* iterativo.

Estes métodos são classificados como analíticos (ou inspeção)² e a avaliação é realizada com o objetivo de identificar problemas relacionados à qualidade de uso em uma interface existente, e analisar estes problemas.

Os métodos mais conhecidos deste grupo são: avaliação heurística [Nielsen & Molich, 1990], que envolve especialistas em usabilidade que tem como objetivo julgar os elementos presentes na interface, de acordo um conjunto de heurísticas baseado nos princípios de usabilidade. Percurso cognitivo [Wharton et al., 1994] que simula em detalhes o processo de realização de tarefas específicas, e visa determinar se os objetivos simulados pelo usuário podem levar a operações subseqüentes necessárias. Outras técnicas envolvendo especialistas são métodos de inspeção, de normas e uso de diretrizes que fornecem *checklists*, com uma referência para avaliar características da interface do software. Nesta seção, são apresentados alguns trabalhos realizados no domínio educacional utilizando estes métodos.

2.3.1 Novos Métodos

Lanzilotti et al. [2006] propuseram um novo método de avaliação específico para sistemas de educação a distância, denominado *e-Learning Systematic Evaluation* (eLSE). Este método é composto por duas etapas: inicialmente, é feita uma inspeção por avaliadores experientes e, em seguida, são realizados testes com os alunos, como forma de validar o resultado obtido pelos especialistas.

Na primeira etapa, os avaliadores mais experientes (geralmente especialistas em IHC) elaboram as tarefas abstratas (AT) que descrevem as atividades a serem melhoradas durante a inspeção e identificam quais objetos e ações da aplicação devem ser analisados. A elaboração dessas tarefas é baseada em um *framework* que foca na qualidade de sistemas interativos para sistemas de educação à distância e visa prover o que é desejável para uma boa qualidade do sistema. Com as tarefas elaboradas, os avaliadores realizam a inspeção do sistema. O resultado de cada inspeção é comparado com o resultado da avaliação dos demais avaliadores. Quando há discrepâncias entre

²Métodos de avaliação analíticos são aqueles nos quais os avaliadores inspecionam ou examinam aspectos de uma interface de usuário relacionados a qualidade de uso [Prates & Barbosa, 2003].

o resultado da avaliação por inspeção, realizam-se testes com os alunos para validar os problemas encontrados pelos avaliadores.

Para validar a metodologia eLSE, selecionaram-se 73 (setenta e três) alunos de IHC, divididos em três grupos. Cada grupo realizou uma avaliação diferente: avaliação heurística, *think aloud* e eLSE. Para validar o resultado obtido pelos 73 alunos especialistas, recrutaram-se 25 estudantes de cursos de computação para a aplicação de teste com usuário. Os resultados indicaram que a AT encontrou mais problemas que a avaliação heurística e o método *think aloud*. O resultado também revelou que as diferentes técnicas abordadas encontraram diferentes tipos de problemas. Os testes com estudantes e a inspeção heurística destacaram os problemas comuns a todos os sistemas interativos, enquanto a inspeção AT encontrou os problemas específicos do sistema de educação a distância. Com isso, esta pesquisa aponta para a necessidade e benefícios de se ter métodos que levam em consideração as especificidades do domínio.

Triacca et al. [2004] propuseram um novo método de inspeção guiado por cenário (quando os avaliadores verificam a viabilidade das tarefas), que se baseia nos conceitos de perfil de aluno, modelo de aluno, cenário e atributos de usabilidade, com avaliação guiada por heurística (verifica a conformidade do sistema com um conjunto de princípios de usabilidade). Este método visa conduzir a validação centrada no aluno, podendo antecipar e justificar, analiticamente, as falhas de usabilidade, além de fornecer indicações para o *redesign* da interface inspecionada. Os autores argumentam que sem a compreensão clara da necessidade dos alunos e seus objetivos, é difícil executar a avaliação de usabilidade que visa fornecer resultados úteis e em profundidade. Com isso, o conhecimento do domínio e do contexto de uso da aplicação são fundamentais para auxiliar o avaliador na inspeção.

Neste contexto, o primeiro passo a ser feito é a identificação dos perfis de alunos do sistema e seus objetivos. Além disso, os alunos têm objetivos genéricos que gostariam ou que deveriam realizar por meio do sistema. Assim, os objetivos estão relacionadas com o conjunto dos objetivos gerais de aprendizagem do curso. Portanto, é possível associar um ou mais objetivos de alto nível para cada perfil de estudante, criando assim os elementos essenciais de um cenário de aluno. A partir da elaboração dos cenários, os avaliadores definem as tarefas a serem realizadas. Cabe a eles identificar os objetivos e as tarefas críticas que são importantes, conforme sua experiência de ensino. Finalmente, o resultado da definição de cenário é um conjunto estruturado de tarefas associado a cada perfil de aluno. Durante a execução das tarefas os avaliadores avaliam a eficácia e eficiência para concluir uma tarefa, e também são apoiados por heurísticas específicas para orientar a inspeção. Com isso, eles se concentram em aspectos como o conteúdo abordado, a eficácia da navegação e a clareza dos *links*.

2.3.2 Métodos Existentes

Tergan [1998] argumenta que a utilização de *checklists* para avaliação de sistemas educacionais são adequadas e flexíveis por atender a diferentes propostas e intenção dos avaliadores. *Checklists* para avaliação de software consistem de uma lista de itens estruturadas intuitivamente de acordo com algumas categorias (e.g. características técnicas, fundamentos pedagógicos). Entretanto o seu foco reside em questões técnicas, em vez de questões educacionais. Tergan [1998] fez uma revisão sobre avaliação de sistemas educacionais baseada em *checklists* e verificou que, quando baseado em critérios bem definidos, este tipo de avaliação pode ser um importante instrumento para projetistas no processo de avaliação formativa de um produto.

Ssemugabi & de Villiers [2007] argumentam que a usabilidade de ambientes educacionais apresentam necessidades particulares, como aspectos pedagógicos e sobre a aprendizagem. Com isso, a avaliação deste tipo de ambiente requer critérios diferentes. Com o objetivo de analisar os resultados obtidos com dois métodos de avaliação de usabilidade, avaliação heurística (feito por especialistas) e teste de usabilidade (feito por usuários finais), os autores realizaram um estudo de caso em que aplicou-se ambos os métodos, em um mesmo sistema de *e-learning*.

Inicialmente, Ssemugabi & de Villiers [2007] definiram a questão de pesquisa, geraram um conjunto de heurísticas específicas para sistemas de *e-learning* baseado nos trabalhos de Nielsen & Molich [1990]; Squires & Preece [1999] (e outros), fizeram avaliação com usuários finais por meio de questionário e entrevistas, realizaram a avaliação heurística e analisaram os dados para responder à questão de pesquisa. Esta questão de pesquisa tinha como finalidade comparar os problemas de usabilidade identificados pela avaliação heurística com os problemas identificados pelos usuários finais.

O conjunto de heurísticas gerados consistia de questões relacionadas à usabilidade, critérios específicos para *websites* educacionais e aspectos sobre a aprendizagem do aluno. Numa análise quantitativa dos resultados, verificou-se que os especialistas identificaram setenta e sete por cento dos problemas enquanto os usuários finais, setenta e três por cento. Entretanto, cinquenta e um por cento dos problemas foram identificados por ambos os grupos. Assim, o resultado desta pesquisa enfatiza a importância da combinação de diferentes métodos de avaliação de usabilidade para o domínio educacional.

Tselios et al. [2008] argumentam que critérios tradicionais para avaliar a usabilidade de sistemas educacionais não são apropriados para este domínio. Os autores citam como exemplo a ocorrência do erro, que para o processo de aprendizagem leva o aluno à experimentar, assim os erros oferecem oportunidades de aprendizagem signi-

ficativa. Portanto, para o contexto educacional, errar não necessariamente diminui o valor pedagógico do sistema. No domínio educacional é desejável que os sistemas sejam benéficos para os alunos, desde que contribuam para o processo de aprendizagem, e não simplesmente suporte a execução de tarefas.

Tselios et al. [2008] inicialmente classifica os sistemas educacionais em três classes de aprendizagem. (i) Primário, procuram envolver o aluno por meio de conteúdos e, eventualmente, proporciona percursos interessantes através do material. (ii) Secundário, a aprendizagem ocorre através da execução de atividades. (iii) Terciário, compreensão ocorre após um processo de reflexão constante com ênfase na cooperação e no diálogo. Em seguida, o autor contrasta estas classes com os possíveis métodos de avaliação que podem ser utilizados em cada uma delas, ou a combinação dos métodos.

Os autores ressaltam ainda que a combinação dos métodos de avaliação pode ser uma alternativa para enfrentar o desafio da dificuldade de se avaliar sistemas educacionais. Em um dos estudos realizados neste trabalho, verificou-se, por exemplo, que a avaliação heurística indicou falhas importantes na interface do sistema, tais como a terminologia utilizada, questões de adaptação ao procedimento de seleção de objetos e a qualidade do *feedback*. Entretanto, a avaliação dos aspectos dinâmicos da interação, como suporte ao usuário e fluidez na navegação, não puderam ser avaliados de forma confiável, sem a participação ativa dos estudantes. Assim, o objetivo dos métodos propostos é identificar as limitações do *design* de interação em ambientes de aprendizagem, demonstrando que a sua adaptação para o domínio educacional não deteriorará as possibilidades de aprendizagem oferecidas pelo sistema. Embora a avaliação de usabilidade não certifique o resultado educacional de tais ambientes, uma interface intuitiva é uma condição obrigatória para uma boa experiência de aprendizado.

Os trabalhos apresentados demonstram a variedade de métodos existentes que permitem a avaliação do especialista em IHC. No entanto, na maioria dos trabalhos tem-se utilizado uma combinação desses métodos para avaliar o domínio educacional. Essa combinação tem demonstrado que diferentes técnicas são capazes de revelar problemas diferentes, que não seriam descobertos apenas com a aplicação de um único método. Assim, os diferentes ambientes de aprendizagem impõem a necessidade de diferentes abordagens para a avaliação da interação com o aluno, o professor e o projetista.

2.4 Classificação de avaliação de ambientes educacionais

A partir do levantamento dos trabalhos que tratam da avaliação de sistemas educacionais, foi proposto uma tabela de classificação para o domínio educacional. Para elaborar as categorias presentes nesta tabela, considerou-se o trabalho feito por Pinelle & Gutwin [2000], com adaptações para o domínio educacional.

A classificação proposta por Pinelle & Gutwin [2000] tinha como objetivo analisar os trabalhos de avaliação realizados para sistemas colaborativos. Nesta tabela, os autores classificaram os trabalhos quanto ao tipo de avaliação, suas principais características, técnicas para coleta de dados e foco da avaliação. Para classificação dos artigos analisados neste trabalho consideramos as categorias listadas a seguir.

- *Envolvidos*: refere-se às pessoas que estão envolvidas no processo de avaliação e podem ser: aluno, professor ou projetista/especialista. Em alguns trabalhos, a avaliação era feita por mais de um envolvido (e.g. aluno e especialista).
- *Método novo ou estudo de caso*: refere-se ao método de avaliação utilizado para envolver os *stakeholders*. Considerou-se a proposta de novo método os trabalhos que apresentaram, a partir de métodos existentes ou não, uma nova proposta de avaliação para o domínio educacional. Em alguns trabalhos era apresentado um estudo de caso com os tradicionais métodos de avaliação de interface (e.g. teste de usabilidade e avaliação heurística).
- *Tipo de avaliação*: nos trabalhos analisados considerou-se o tipo de avaliação feita e estes foram classificados como qualitativa ou quantitativa, e formativa ou somativa.
- *Coleta de dados*: refere-se à forma como os dados foram coletados (e.g. inspeção por tarefas ou teste com usuário).
- *Tipo de sistema*: refere-se ao tipo de sistema educacional que o método se propôs a avaliar, a saber: (i) sistemas de ensino a distância, (ii) sistemas educacionais e (iii) ambiente virtual de aprendizagem - AVA (e.g. Moodle).
- *Ambiente para realização da avaliação*: refere-se ao local onde a avaliação foi realizada, a saber (i) ambiente controlado³ ou (ii) avaliação por inspeção⁴.

³ambiente em que o usuário poderá se concentrar no sistema, sem correr o risco de ser interrompido ou ter sua atenção desviada das atividades do teste [Prates & Barbosa, 2003]

⁴utilizada para identificar problemas de usabilidade e analisar estes problemas visando fazer re-

- *Características avaliadas*: refere-se aos critérios analisados na avaliação. Eles foram classificados como (i) usabilidade, qualidade de uso relacionada à facilidade e eficiência de aprendizado e de uso, bem como satisfação do usuário [Nielsen, 1994]; (ii) comunicabilidade, propriedade de um sistema de transmitir ao usuário os princípios de interação e decisões que o guiaram [Prates et al., 2000] e (iii) aprendizagem.
- *Indicadores*: refere-se aos indicadores gerados pela avaliação. Podem ser: (i) interface, (ii) aprendizado e (iii) utilidade do sistema.

Apresentar uma forma de classificação e classificar alguns dos trabalhos existentes sobre avaliação de sistemas educacionais traz uma importante contribuição para a área de IHC e Informática e Educação. Essa classificação permite que os demais pesquisadores interessados em avaliar um sistema educacional tenham uma visão geral do tipo de avaliação que podem utilizar e dos critérios que desejam considerar.

Os métodos propostos procuram envolver educadores e/ou alunos e perspectivas educacionais, mas não se baseiam em uma teoria de IHC. Neste trabalho, será considerada a teoria da Engenharia Semiótica, como forma de fundamentar a pesquisa na área de IHC. Baseado nesta teoria, foi proposto um método que permite a avaliação de sistemas educacionais sob o ponto de vista do professor, denominado Método de Inspeção Semiótica Intermediado (MIS Intermediado), conforme pode-se ver no capítulo 6.1.

A tabela com o resultado da classificação dos trabalhos avaliados é apresentada a seguir.

Tabela de avaliação para o domínio educacional

Referência	Método de origem	Tipo de avaliação	Tipo de sistema	Características avaliadas	Participantes	Coleta de dados	Ambiente para realização do teste	Foco dos resultados
Lanzilotti et al. [2006]	Novo: eLSE (e-Learning Systematic Evaluation)	Somativa e quantitativa	Específico para ensino a distância	Usabilidade	Aluno	Alunos realizam o método <i>think aloud</i>	Ambiente controlado	Problemas de interação
					Especialista	Inspeção por meio de tarefas	Inspeção	Problemas específicos do sistema de ensino a distância
Matera et al. [2002]	Novo: SUE (Avaliação Sistemática de Usabilidade)	Somativa e quantitativa	Sistema educacional	Usabilidade	Aluno	Aluno faz o teste de usabilidade	Ambiente controlado	Interface
					Especialista	Especialista faz a inspeção por meio de tarefas	Inspeção	Aprendizagem
Zaharias [2006]	Novo: Questionário baseado na avaliação de usabilidade, foco na motivação para aprender.	Somativa e quantitativa	Sistemas de ensino a distância	Usabilidade	Aluno	Questionário baseado no teste de usabilidade, visando integrar aspectos cognitivos e afetivos.	Teste online	Interface, estratégias de aprendizagem e motivação para aprender
Masemola & De Villiers [2006]	Adaptação: teste de usabilidade	Somativa e quantitativa	Ensino a distância para aprendizagem matemática	Usabilidade e aprendizagem	Aluno	Teste de usabilidade com <i>think aloud</i> e entrevista pós-teste	Ambiente controlado	Aprendizagem, utilidade do sistema e sobre a interface.

Ardito et al. [2006]	Combinação de método de inspeção - usando o SUE [Matera et al. [2002]] e teste com aluno	Somativa, quantitativa e qualitativa	Sistemas de ensino a distância	Usabilidade	Aluno	Teste de usabilidade	Ambiente controlado	Usabilidade
				Aprendizagem	Especialista	Inspeção usando o SUE	Inspeção	Aprendizagem
Quintana et al. [2002]	Novo: critérios para avaliar sistemas com <i>scaffolds</i>	Somativa e qualitativa	Sistema de apoio ao aprendizado com <i>scaffolds</i>	Usabilidade e aprendizagem	Aluno	Critérios de usabilidade definidos pelos autores (gerenciamento de tempo, acessibilidade, uso, eficiência, acurácia, progressão e reflexão sobre a tarefa executada)	Ambiente controlado	Usabilidade
				Uso dos <i>scaffolds</i> pelos alunos e aprendizagem	Professor ou projetista		Ambiente controlado	Aprendizagem e <i>scaffolds</i> mais usados
Freitas e Dutra [2009]	Estudo de caso	Somativa e qualitativa	Ambientes Virtuais de Aprendizagem (AVA)	Usabilidade	Aluno	Teste com usuário baseado em questionários	Pesquisa de campo	Usabilidade
Brush e Saye [2001]	Estudo de caso	Somativa e qualitativa	Sistemas educacionais	Uso dos <i>scaffolds</i> pelos alunos	Aluno	Log do sistema e entrevista	Ambiente controlado	Aprendizagem e uso de <i>scaffolds</i>
					Professor			Aprendizagem e <i>scaffolds</i> mais usados
Iahad et al. [2004]	Estudo de caso	Somativa, qualitativa e quantitativa	Sistemas de ensino a distância	Usabilidade e aprendizagem	Aluno	Questionário, baseado em critérios de funcionalidade e usabilidade,	Teste Online	Usabilidade e aprendizagem

						por meio de avaliação formativa e somativa.		
Squires & Preece [1996]	Adaptação: baseada no modelo Jigsaw [Squires 1994]	Somativa e qualitativa	Sistemas de apoio ao aprendizado	Usabilidade e aspectos educacionais	Especialista ou professor	Avaliação preditiva	Inspeção	Usabilidade e questões educacionais
Squires & Preece [1999]	Novo: Método de avaliação por inspeção	Somativa (análise preditiva) e qualitativa	Sistemas de apoio ao aprendizado	Usabilidade e aspectos educacionais	Professor	Adaptação das heurísticas para o domínio educacional	Inspeção	Usabilidade e aprendizagem
De Villiers [2004]	Adaptação: baseada nas heurísticas de Squires & Preece [1999]	Formativa e qualitativa	Sistema educacional	Usabilidade e aspectos educacionais	Professor	Heurísticas propostas por Squires & Preece [1999]	Inspeção	Usabilidade e aprendizagem
Lyra et al. [2003]	Novo: Ambiente para avaliação de sistemas educacionais baseado em questionários	Somativa e qualitativa	Sistema educacional	Aprendizagem, aspectos pedagógicos	Professor	Questionário	Teste on-line	Auxiliar o professor na seleção de sistemas educacionais
				Aspectos técnicos	Projetista			Fornecer ao designer as análises feitas por educadores, a partir da prática efetiva em contextos de uso
Pardo et al. [2006]	Estudo de caso	Somativa e qualitativa	Sistema educacional	Usabilidade e aprendizagem	Professor	Observação do usuário	Ambiente controlado	Usabilidade e aprendizagem
					Aluno	Teste com usuário		

					Projetista	Observação do usuário		
Rentróia-bonito et al. [2008]	Combinação: avaliação técnica e teste com usuário	Formativa e qualitativa	Sistemas de ensino a distância	Usabilidade e aprendizagem	Professor	Log do sistema e indicadores de desempenho	Ambiente controlado	Usabilidade e indicadores de desempenho
					Aluno	Avaliação de usabilidade através de questionário		Usabilidade
Tergan [1998]	Estudo de caso	Formativa e quantitativa	Sistema educacional	Questões técnicas	Especialista	Checklists	Inspeção	Usabilidade
Triacca et al. [2004]	Novo: Método guiado por inspeção – Mile (Milano Lugano Evaluation Method)	Somativa e qualitativa	Sistema educacional	Usabilidade e aprendizagem	Especialista	Inspeção guiada por cenário	Inspeção	Conteúdo abordado, a eficácia da navegação e a clareza dos links.
De Souza e Sedig [2001]	Estudo de caso	Somativa e qualitativa	Classificação de Peirce para analisar a interface de um software educacional	Sistema educacional	especialista	inspeção	Signos usados em representação visual em interfaces de manipulação direta	Interação (manipulação direta)
Ssemugabi e De Villiers [2007]	Estudo de caso	Somativa e quantitativa	Sistemas de ensino a distância	Usabilidade e aprendizagem	Especialista	Avaliação heurística	Inspeção	Usabilidade, aprendizagem e aspectos específicos para web sites educacionais
					Aluno	Teste com usuário	Ambiente controlado	

Capítulo 3

A Engenharia Semiótica e seus Métodos de Avaliação

Este capítulo apresenta a teoria da Engenharia Semiótica [de Souza, 2005], que fundamenta este trabalho, e seus métodos de avaliação da comunicabilidade de um sistema. A seção 3.1 apresenta os principais conceitos desta teoria e, em seguida, a seção 3.2 apresenta o Método de Inspeção Semiótica e, finalmente, a seção 3.3 aborda o Método de Avaliação de Comunicabilidade.

3.1 Teoria da Engenharia Semiótica

A Engenharia Semiótica (EngSem) [de Souza, 2005] é uma teoria explicativa de Interação Humano-Computador (IHC), fundamentada na Semiótica – uma disciplina que estuda os fenômenos de significação e comunicação. A EngSem oferece explicações sobre os fenômenos envolvidos no projeto, uso e avaliação de um sistema interativo. Seu foco é no processo de comunicação entre projetista (designer) e usuário, sendo feito por meio da interface de um sistema.

Para a EngSem, a interface de um sistema é entendida como uma comunicação do projetista para o usuário, que tem por objetivo comunicar ao usuário a visão do projetista sobre a quem ela se destina; que problemas ela pode resolver e como interagir com ela [de Souza, 2005; Prates & Barbosa, 2007; de Souza & Leitão, 2009], conforme pode-se observar na figura 3.1. O usuário compreende esta mensagem à medida que interage com o sistema. Pela teoria da EngSem, a interface do sistema é vista como o preposto do projetista, isto é, o sistema transmite a mensagem do projetista ao usuário. Esta mensagem pode ser parafraseada como mostrado na figura 3.1 (b).

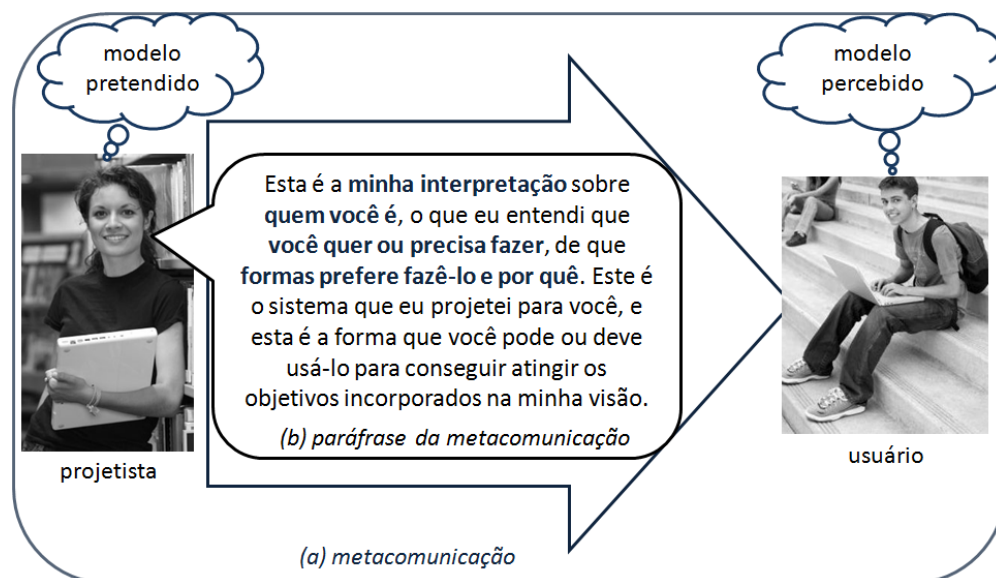


Figura 3.1. Visão Geral da Engenharia Semiótica

Assim, a interface é percebida como um artefato de metacomunicação indireto e unidirecional [de Souza, 2005]. Um artefato de metacomunicação é aquele cuja comunicação é relativa à própria comunicação, no caso a mensagem do projetista para o usuário é sobre a comunicação do usuário com o sistema. A metacomunicação é indireta porque o usuário deve compreendê-la à medida que interage com a interface do sistema e não com o designer, que é o emissor da mensagem. É unidirecional pois o usuário, em tempo de interação, não pode dar continuidade àquela comunicação com o projetista.

No que se refere à ambientes educacionais a metamensagem, idealmente, não é enviada apenas pelo projetista, mas também por um educador [Prates, 2004]. Cabe ao educador as decisões referentes ao objetivo e o método de ensino adotados na ferramenta. O objetivo e método são definidos em função do perfil do aluno, ou seja, sua idade, seu conhecimento prévio, as condições em que deverá utilizar o sistema, dentre outros. Ao projetista cabe definir como será a interação do aluno com o sistema para conseguir atingir seu objetivo.

As perguntas a serem respondidas para a reconstrução da metamensagem são: (i) quem é você, (ii) o que você quer ou precisa fazer, (iii) de que formas prefere fazê-lo e (iv) como interagir com o sistema para isso. No caso do domínio educacional, o educador é responsável por definir a resposta às perguntas (i), (ii) e (iii) da Engenharia Semiótica para a reconstrução da metamensagem, que pode ser reescrita da seguinte forma: (i) *Qual o perfil do aluno a quem a atividade se destina?*, (ii) *O que o aluno pode aprender?* (iii), *Que atividades o permitem atingir este aprendizado?*. O projetista continua sendo

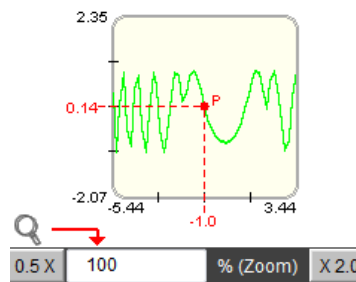


Figura 3.2. Exemplo de signo estático

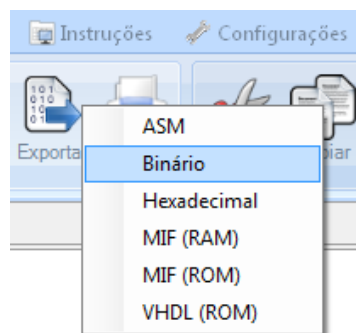


Figura 3.3. Exemplo de Signo Dinâmico

responsável por responder à pergunta *(iv) Como interagir com o sistema para isso?*.

Para compor a mensagem sendo enviada, o projetista faz uso de signos, onde signo é tudo aquilo que significa algo para alguém [Peirce et al., 1932]. A EngSem classifica os signos de um sistema computacional em três diferentes níveis: *signos estáticos*, *dinâmicos* e *metalinguísticos* [de Souza et al., 2006; de Souza & Leitão, 2009; de Souza et al., 2010].

Os *signos estáticos* são signos em que o contexto da interpretação é limitada aos elementos que estão presentes na interface em um único momento no tempo. Eles expressam o estado do sistema e podem ser percebidos ao se olhar para a interface do sistema (e.g. texto de um botão, botão de fechar). A figura 3.1, por exemplo, mostra o recurso de *zoom* e a parte gráfica da função, no sistema VCalc. Os *signos dinâmicos* são aqueles relativos a aspectos temporais e causais da interface. Estes signos surgem com a interação e devem ser interpretados com referência a ela, assim eles expressam o comportamento do sistema quando o usuário interage com ele (e.g. ação iniciada após clicar em um botão). A figura 3.2 ilustra um exemplo de signo dinâmico, uma vez que mostra o menu que é aberto quando o usuário clica no botão *Exportar*.

Os signos estáticos e dinâmicos estão diretamente relacionados [de Souza & Leitão, 2009]. Signos estáticos estimulam o usuário a interagir com o sistema. Os

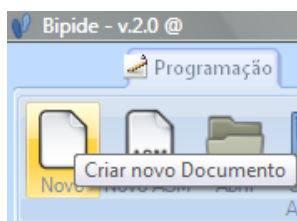


Figura 3.4. Bipide: exemplo de signo metalinguístico tooltip

signos dinâmicos confirmam, ou não, o que foi antecipado pelo usuário por meio dos signos estáticos.

O significado dos signos estáticos e dinâmicos é explicitamente informado ou explicado por signos de outra classe, os *signos metalinguísticos*. Estes signos referem a outros signos presentes na interface estática, dinâmica, ou mesmo metalinguísticos. Geralmente esta classe de signo vem em forma de ajuda ou mensagens de erro, avisos, dicas, dentre outros. A figura 4.4 ilustra um signo metalinguístico referente ao *help* do sistema VCalc, enquanto a figura 3.3 apresenta um signo *tooltip*, com informação sobre a criação de um novo documento no sistema Bipide. Por meio dos signos metalinguísticos, o projetista comunica explicitamente aos usuários os significados codificados no sistema e como eles podem ser usados. Assim, esta é a comunicação mais direta do projetista para o usuário sobre o sistema ou elementos específicos.

Para avaliar a qualidade da comunicação projetista-usuário, a EngSem define a propriedade de comunicabilidade. A comunicabilidade é definida como a propriedade de um sistema transmitir ao usuário, de forma eficaz e eficiente, as intenções e princípios de interação que guiaram seu design [Prates et al., 2000]. À medida que o usuário interage com o sistema, podem ocorrer rupturas de comunicação que podem dificultar ou mesmo impedir a meta-comunicação do usuário com o sistema. Quanto mais rupturas ocorrerem durante a interação usuário-sistema, mais baixa será a comunicabilidade da interface. Atualmente, há dois métodos para se avaliar a comunicabilidade de uma interface: o Método de Inspeção Semiótica (MIS) [de Souza et al., 2006] e o Método de Avaliação de Comunicabilidade (MAC) [Prates et al., 2000].

O Método de Inspeção Semiótica [de Souza et al., 2006; Prates & Barbosa, 2007; de Souza & Leitão, 2009; de Souza et al., 2010] é um método antecipativo, ou seja, método em que um especialista percorre a interface e identifica (i.e. antecipa) potenciais rupturas de comunicação que poderiam surgir na interação usuário-sistema e reconstrói a metamensagem. Como o foco é na comunicação sendo feita através do sistema (projetista-usuário) e com o sistema (sistema-usuário), e não em princípios de interação, a princípio um especialista em IHC poderia utilizá-lo para avaliar sistemas

baseados em diferentes tecnologias ou domínios [Mattos et al., 2009; Bento et al., 2009].

O Método de Avaliação de Comunicabilidade [Prates et al., 2000; de Souza, 2005; Prates & Barbosa, 2007; de Souza & Leitão, 2009] é um método que envolve a observação de usuários em um ambiente controlado (e.g. laboratório de testes) por um especialista. Durante o experimento, o especialista grava a interação do usuário com o sistema e, posteriormente, revê a gravação.

Tanto o MIS quanto o MAC são métodos qualitativos e interpretativos [Denzin & Lincoln, 2000]. Ser qualitativo significa que o método é exploratório, o que permite uma postura aberta do avaliador com o objetivo de se aprofundar nas questões de interesse. Essa exploração gera novos conhecimentos, ao invés de confirmar hipóteses prévias e conhecidas. São interpretativos, pois a qualidade dos resultados depende da capacidade do avaliador de interpretar os dados coletados e a reconstrução de processos de comunicação é sempre uma interpretação.

A partir do ano de 2000, a pesquisa sobre a teoria da EngSem evoluiu e, conseqüentemente, o conceito de comunicabilidade foi sutilmente alterado. A definição original de comunicabilidade citada anteriormente se refere principalmente às condições operacionais. Como forma de explicitar os interlocutores envolvidos no processo de comunicação, de Souza [2005, p.113] adaptou o conceito de comunicabilidade, como pode-se verificar a seguir:

“Comunicabilidade pode ser definida tecnicamente como a capacidade do preposto do designer [o sistema] para alcançar a metacomunicação completa, transmitindo aos usuários a essência da mensagem original do designer. [...] Comunicabilidade aplica-se a códigos interpretativos e expressivos utilizados pelo preposto do designer para gerar e interpretar mensagens durante a interação com os usuários.” de Souza [2005, p.114] (tradução da autora)¹

No contexto de IHC, os métodos propostos pela teoria da EngSem podem ser utilizados para duas finalidades: aplicação técnica e científica [de Souza & Leitão, 2009; de Souza et al., 2010]. A aplicação técnica tem como finalidade aplicar o conhecimento

¹Communicability can thus be more technically defined as the designer’s deputy capacity to achieve full metacommunication, conveying to users the gist of the original designer’s message. This message, as seen in previous chapters, will tell the users what is the designer’s understanding of who they are, what they want or need to do, in which preferred ways, and why. It will also tell them about the system that the designer has built for them based on what he knows, and how they can or should use it in order to fulfill a range of purposes that fall within the original designer’s vision. Communicability applies to both interpretive and expressive codes that the designer’s deputy handles for generating and interpreting messages during situated interaction with users.

e métodos da teoria para aumentar a qualidade do *design* de IHC de uma aplicação específica, de modo a melhorar também a qualidade da interação e da experiência do usuário, gerando produtos de maior qualidade. Já a aplicação científica tem a finalidade de construir novos conhecimentos para aprimorar o *design* de IHC e a experiência do usuário, por meio da identificação de novos problemas e desafios na área, bem como mapear e analisar as estratégias e soluções de design, dentre outros.

Neste trabalho como o objetivo era obter indicadores sobre a melhoria da qualidade da interação de sistemas educacionais, foi realizada uma investigação sobre a aplicabilidade dos métodos propostos pela teoria da EngSem a este domínio, considerando-se o uso científico dos métodos para gerar conhecimento científico sobre eles. A seguir, cada um destes métodos é apresentado em detalhes.

3.2 Método de Inspeção Semiótica

O MIS [de Souza et al., 2006] é um método de inspeção em que um especialista em IHC e EngSem analisa a metacomunicação do projetista para o usuário sendo transmitida por meio do sistema. Este método tem como objetivo identificar potenciais rupturas de comunicação que poderiam surgir na interação usuário-sistema e gerar a reconstrução da metamensagem sendo enviada pelo projetista [de Souza et al., 2006]. O MIS tem como foco a emissão da metamensagem do projetista para o usuário. Por meio da exploração e análise dos signos presentes na interface o especialista (avaliador) reconstrói a metamensagem. A sua aplicação pode contribuir na melhoria da qualidade de um determinado sistema e ainda expandir o conhecimento técnico do avaliador.

Assim como os tradicionais métodos inspeção de IHC, o MIS não requer a observação de usuários interagindo com o sistema. Ele deve ser aplicado por avaliadores especialistas em EngSem e os permite antecipar algumas consequências possíveis que escolhas de *design* podem trazer quando os usuários interagem com o sistema.

Sob a visão da EngSem, um avaliador gera um conjunto de possíveis interpretantes sobre a metacomunicação, logo não é necessário ter mais de um avaliador, uma vez que “todos os possíveis caminhos interpretativos que possam ser gerados por um ou mais avaliadores são plausíveis” [Prates & Barbosa, 2007]. No entanto, ele pode ser aplicado por mais de um avaliador. Neste caso, o MIS não visa replicar os resultados gerados por cada avaliador, mas buscar a consistência entre esses resultados e ainda enriquecer o relatório gerado com visões distintas. A seguir, os passos para aplicação do MIS são detalhados.

3.2.1 Passos do MIS

Como outros métodos de inspeção, o MIS requer uma preparação inicial que precede a aplicação dos demais passos. A preparação é composta pelos passos seguintes. Inicialmente, o avaliador define o objetivo e o escopo da inspeção. Esta definição pode levar em consideração partes do sistema a ser avaliado que representem as principais funções que os usuários terão que executar ou ainda partes do sistema que a equipe de *design* defina como relevante. Em seguida, o avaliador faz uma inspeção informal do sistema, com o objetivo de identificar quem são os principais usuários do sistema e as principais tarefas que este usuário pode realizar no sistema. Para realizar esta inspeção, o avaliador recorre a informações sobre o sistema como website, sistema de ajuda, dentre outros. Uma vez identificados os usuários e as principais tarefas que estes usuários podem realizar no sistema, o avaliador elabora os cenários de inspeção² necessários para a análise da comunicabilidade.

Finalizada a etapa de preparação, inicia-se a aplicação dos demais passos do MIS. A análise consiste no especialista inspecionar o sistema considerando três diferentes níveis de signos utilizados pelo projetista: metalingüísticos, estáticos e dinâmicos [de Souza et al., 2006]. O MIS é composto por 5 (cinco) passos, conforme apresentado na figura 3.4.

1. Inspeção dos signos metalingüísticos;
2. Inspeção dos signos estáticos;
3. Inspeção dos signos dinâmicos;
4. Contraste e comparação entre as metagensagens identificadas anteriormente; e
5. Apreciação da qualidade da metacomunicação.

A análise metalingüística consiste na inspeção aprofundada dos signos metalingüísticos como ajuda, site, etc., com o objetivo de identificar potenciais problemas na comunicação. À medida que analisa estes signos, o especialista reconstrói a metagensagem enviada pelo projetista ao usuário por meio dos signos metalingüísticos. Esta mensagem é a comunicação mais direta do projetista para o usuário, porque nestes signos o projetista fala diretamente para o usuário e em linguagem natural. No entanto, nem sempre ele explicita o que se quer saber na metagensagem. Muitos sistemas de

²Cenário é uma narrativa textual plausível contextualizada e detalhada que descreve determinada situação de uso da aplicação, envolvendo usuários, processos e dados reais ou potenciais [Carroll et al., 1994]

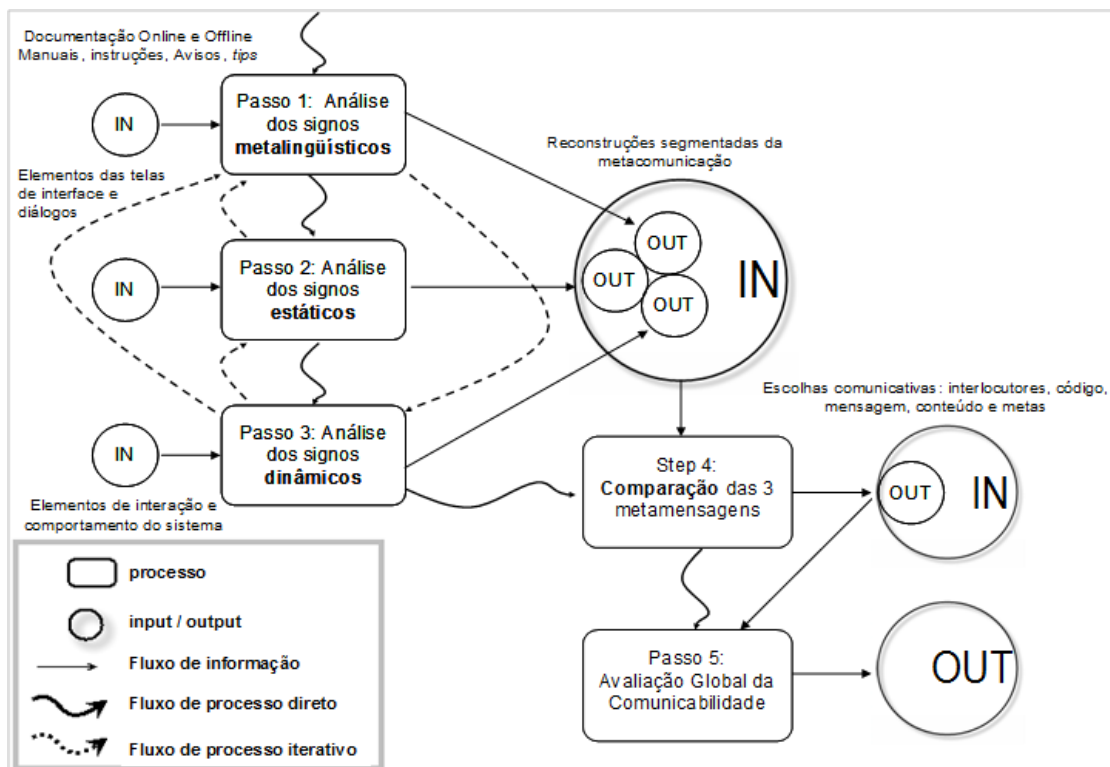


Figura 3.5. Visão Geral do Método de Inspeção Semiótica [de Souza et al., 2010]

ajuda são apenas um “Como fazer” isso ou aquilo. Nesta etapa, o especialista deve utilizar a paráfrase da metamensagem do projetista como *template* a ser preenchido, conforme apresentado a seguir.

“Esta é a minha interpretação sobre quem você é, o que eu entendi que você quer ou precisa fazer, de que formas prefere fazê-lo e por quê. Este é, portanto, o sistema que eu projetei para você, e esta é a forma que você pode ou deve usá-lo para conseguir atingir os objetivos incorporados na minha visão.”

Na análise dos signos estáticos o especialista inspeciona os signos estáticos presentes na interface do sistema também visando encontrar potenciais problemas na comunicação projetista-usuário. Baseado somente no que estes signos comunicam, o avaliador faz a reconstrução da metacomunicação projetista-usuário utilizando o mesmo *template* do passo 1 (um). Vale ressaltar que as metamensagens geradas em cada passo podem ser inconsistentes, complementares ou até mesmo parecidas. Entretanto, é importante manter a segmentação da análise em cada passo.

Na inspeção dos signos dinâmicos analisa-se a interação do usuário com o sistema. Os signos dinâmicos representam o comportamento do sistema e exercem um impor-

tante papel na comunicação projetista-usuário, pois por meio deles, o usuário confirma (ou não) o significado percebido na análise dos signos estáticos. Então, a partir dos potenciais problemas encontrados, o especialista reconstrói novamente a metamensagem baseado somente no que estes signos comunicam.

No passo 4, baseado no preenchimento dos 3 (três) *templates*, o especialista compara o conteúdo de cada um deles, com o objetivo de destacar as consistências e inconsistências entre as mensagens geradas nos passos anteriores e apresenta as classes de signos identificadas na inspeção. A partir desta análise, ele explora a possibilidade de o usuário atribuir significados diferentes a um mesmo signo.

No passo 5, o especialista gera um relatório contendo uma breve descrição do método, os critérios definidos para selecionar a parte do sistema avaliado, os signos relevantes e os problemas de comunicabilidade encontrados. Finalmente, a partir das consistências e inconsistências encontradas nos 3 (três) *templates* preenchidos e comparados entre si, o especialista incorpora e integra seus conteúdos, gerando assim uma metamensagem unificada.

Embora a aplicação científica do MIS não seja o foco do trabalho, vale ressaltar que esta aplicação consistiria em mais um passo, além dos mencionados. O último passo da aplicação científica do MIS é a triangulação, que consiste em conferir a validade dos resultados da inspeção [de Souza & Leitão, 2009; de Souza et al., 2010]. Validar, neste caso, refere-se à busca de diferentes interpretações sobre a mesma questão de pesquisa, ou seja, é uma forma de não replicar as conclusões e considerações geradas em outros contextos. A triangulação é a comparação dos resultados obtidos no método com os obtidos em outros contextos e pode ser feita em fontes endógena ou exógena [de Souza et al., 2010]. Fontes endógenas são artefatos computacionais que compartilham o mesmo modelo de domínio, mas considerados em outros contextos (i.e. mesmo sistema ou sistemas do mesmo tipo). Já as fontes exógenas são artefatos computacionais de diferentes domínios, mas que compartilham algumas características de *design* (i.e. criar conta de email e criar grupo de discussão)

O argumento é que o MIS faz a inspeção da comunicação sendo feita por meio da interface, sem no entanto restringi-la a critérios específicos de interação ou tecnologia. Cabe ao avaliador, considerando o domínio da aplicação, a tecnologia utilizada, o sistema e o cenário fornecido, identificar potenciais problemas na comunicação projetista-usuário. Alguns trabalhos avaliaram a aplicabilidade do MIS para diferentes contextos: interação humano-robô [Bento et al., 2009], sistemas colaborativos [Moura et al., 2008; Mattos et al., 2009] e sistemas educacionais [Castro & Fuks, 2009]. No trabalho de Castro & Fuks [2009], foi avaliado um ambiente de gerência de conteúdo, o Moodle. Vale ressaltar que esta avaliação não considerou aspectos sobre a aprendizagem dos

alunos, mas sim, os recursos oferecidos pelo sistema para apoiar a aprendizagem.

3.3 Método de Avaliação de Comunicabilidade

O Método de Avaliação de Comunicabilidade (MAC) [Prates et al., 2000; de Souza, 2005] é um método de investigação da comunicabilidade que, diferente do MIS, envolve a participação de usuários em um ambiente controlado (i.e. ambiente em que o usuário pode se concentrar no sistema, sem ser interrompido ou ter sua atenção desviada das atividades do teste). Este método tem como objetivo identificar e antecipar algumas potenciais consequências de determinadas escolhas de *design* por meio da interpretação do avaliador a respeito da experiência real dos usuários [Prates et al., 2000].

O MAC tem como foco a recepção da metagemagem pelo usuário. Assim, a metagemagem é reconstruída a partir da experiência do usuário com o sistema, baseada nas potenciais rupturas de comunicação vivenciadas por ele durante a interação [Prates et al., 2000; Prates & Barbosa, 2007; de Souza, 2005; de Souza & Leitão, 2009]. O avaliador é o responsável por observar este processo de interação e, a partir de sua interpretação, identifica as rupturas de comunicação vivenciadas pelo usuário por meio de um conjunto de expressões denominadas etiquetas de comunicabilidade. A partir das rupturas identificadas, o avaliador analisa os problemas de comunicação identificados e faz a reconstrução da metagemagem apontando os seus problemas. A seguir, os passos para aplicação do MAC são detalhados.

3.3.1 Passos do MAC

O MAC é realizado por meio de 3 (três) passos: preparação do teste, coleta de dados e análise dos dados [Prates et al., 2000; de Souza, 2005; Prates & Barbosa, 2007].

3.3.1.1 Preparação do teste

Esta etapa é dividida em 5 (cinco) sub-etapas [de Souza & Leitão, 2009].

1. Inicialmente, o avaliador, por meio de uma inspeção informal do sistema, determina o *objetivo do avaliação*. Esta definição pode ser feita a partir da identificação das principais atividades e metas do sistema e/ou ainda, pontos problemáticos da interface.
2. Em seguida, o avaliador deve *elaborar as tarefas* a serem executadas pelos usuários. A definição das tarefas a serem executadas podem ser definidas a partir

dos objetivos definidos no passo anterior. Para cada tarefa é necessário gerar um cenário (ver seção 3.2.1) para contextualizar uma situação de uso real, de forma que o participante sintá-se motivado a realizá-la.

3. A *seleção dos participantes* deve ser feita com base no perfil de usuários a que o sistema se destina. É recomendado que haja de 5 (cinco) a 8 (oito) participantes da avaliação [Prates & Barbosa, 2003].
4. *Aspectos éticos*: tendo em vista a existência da Resolução 196/96, o Conselho Nacional de Saúde [CNS, 1996]³ regulamenta pesquisas científicas brasileiras envolvendo pessoas, e se aplica à avaliação de sistemas interativos com usuários [Leitão, 2003]. Segundo esta resolução, deve-se considerar, dentre outras coisas, o caráter voluntário de participação do usuário e o seu anonimato, assim como o direito de interromper a participação no teste a qualquer momento.
5. Finalmente, faz-se a *geração do material impresso* para avaliação que inclui: questionário pré-teste, para coletar o perfil do participante; termo de consentimento sobre cuidados éticos; roteiro para as entrevistas pré e pós-teste, para coletar a opinião dos usuários sobre o sistema; formulário de acompanhamento do teste, para que o avaliador faça as anotações que julgar necessárias durante o teste; e se necessário, textos explicativos sobre o sistema a ser avaliado.

3.3.1.2 Coleta de dados

Finalizada a preparação, o próximo passo é a aplicação dos testes. Esta etapa deve ser feita em um ambiente controlado, para que o avaliador acompanhe a interação dos usuários com o sistema e faça as anotações necessárias. O avaliador recebe o participante nesta sala e explica como será a aplicação do teste, esclarecendo as possíveis dúvidas do participante. Em seguida, o avaliador dá as tarefas que foram planejadas para avaliação do sistema para serem executadas pelo participante. Somente nesta etapa o participante faz a interação com o sistema avaliado. O avaliador deve ressaltar que toda a interação do usuário com o sistema será gravada para análise posterior. Idealmente, dois avaliadores participam desta etapa, sendo um responsável por conduzir o teste e dar as orientações necessárias ao participante, enquanto o outro fica na sala de observação acompanhando a interação do participante por meio de um monitor-clone.

³[CNS, 1996] Conselho Nacional de Saúde. Resolução número 196/96 sobre pesquisas envolvendo seres humanos. 1996. Disponível online em <http://conselho.saude.gov.br/docs/Reso196.doc>. Acesso em maio 2010.

Durante a observação, os avaliadores fazem anotações que consideram relevantes ou que geraram dúvidas durante a interação do participante com o sistema.

3.3.1.3 Análise dos dados

Esta etapa é dividida em 3 (três) passos: etiquetagem, interpretação e perfil semiótico, conforme pode-se observar na figura 3.5.

1. *Etiquetagem*: neste passo, o avaliador vê o filme gerado e analisa a interação do usuário com o objetivo de identificar rupturas de comunicação. A análise do comportamento do participante é classificada associando-se uma ou mais de treze expressões de comunicabilidade, denominadas etiquetas [Prates et al., 2000]. Cada etiqueta remete a um tipo de problema de comunicação vivenciado pelo usuário. Assim, a presença de etiquetas denota a existência de problemas de comunicabilidade; sua ausência, a falta de evidência de tais problemas. As etiquetas têm por objetivo serem expressões naturais, plausíveis de serem manifestadas pelo usuário durante a interação (e.g. “Cadê?” ocorre quando o usuário busca na interface a operação que deseja executar). Na tabela 3.3 é apresentada a lista completa das etiquetas e seus principais sintomas [Prates et al., 2000; Prates & Barbosa, 2007].

O MAC categoriza uma ruptura de interação em três níveis: *operacional*, *tático* e *estratégico* [Prates et al., 2000; de Souza, 2005; de Souza & Leitão, 2009]. A ruptura de nível operacional refere-se a problemas locais, referentes a uma ação individual executada pelo usuário. Enquanto que no nível tático a ruptura é gerada por uma seqüência de ações que o usuário executou para se atingir um objetivo. Já no nível estratégico a ruptura refere-se à formulação do problema e sua solução. Rupturas neste nível podem ser fatais para a adoção da tecnologia, uma vez que eles podem apontar equívocos fundamentais na concepção da metamensagem, como quem são os usuários, o que eles querem ou precisam fazer, como e porquê.

2. *Interpretação*: este passo requer que o avaliador identifique os principais problemas da interface com base nas rupturas identificadas e nas relações entre elas. A interpretação dos resultados dependerá da sua experiência e conhecimento em EngSem. As anotações feitas durante o teste e as entrevistas pós-teste com os participantes contribuem para eliminar possíveis ambiguidades e enriquecer a interpretação do processo de metacomunicação. Por isso, a entrevista, apesar de



Figura 3.6. Passos do MAC [Mattos, 2010]

não ser obrigatória, é muito recomendada para sanar possíveis dúvidas que surgiram durante a etiquetagem. A interpretação das etiquetas permite ao avaliador obter indicações sobre as causas dos problemas identificados e, em muitos casos, também oferece indicadores sobre uma possível solução.

3. *Perfil semiótico*: esta etapa finaliza a aplicação do MAC, que consiste na caracterização profunda da comunicabilidade da aplicação avaliada. A elaboração do perfil semiótico consiste na reconstrução da mensagem enviada do projetista ao usuário por meio das seguintes perguntas: “Quem é o usuário?”; “O que o usuário quer ou precisa fazer?”; “De que formas prefere fazê-lo e por quê?”; “Como os usuários podem ou devem usá-lo?”; “Qual é a minha visão de design?”. Para responder a estas perguntas, o avaliador faz uso do *template* apresentado na seção 3.1. Enquanto preenche este *template*, o avaliador identifica os principais resultados relacionados ao objetivo da avaliação e apresenta também os potenciais problemas percebidos.

No domínio educacional, um estudo de caso anterior [Prates et al., 2004] aponta para a necessidade de adaptações do MAC para considerar as especificidades deste domínio. O estudo de caso feito por Prates et al. [2004] apresentou ambigüidades de sintoma, uma vez que ações que, originalmente, seriam consideradas sintomas de uma ruptura na comunicação, no domínio do estudo de caso representavam aspectos sendo considerados pelo participante ao raciocinar sobre o próximo passo do conteúdo sobre o aprendido. Outro problema encontrado refere-se a uma

situação em que o aluno tinha dificuldade em compreender o assunto abordado pelo sistema, o que não caracteriza uma ruptura da comunicação sobre a interface, mas sobre o conteúdo. Esse problema mostrou a importância de ter o educador e especialista fazendo a etiquetagem, pois somente o educador foi capaz de identificar que não era um problema de interface, e sim de conteúdo.

No próximo capítulo será apresentada a análise da aplicabilidade dos métodos propostos pela EngSem para o domínio educacional. Inicialmente, são apresentados os sistemas inspecionados e, em seguida, a aplicação dos métodos MIS e MAC, com a participação do especialista em EngSem e do aluno, respectivamente. Finalmente, são discutidos a aplicabilidade destes métodos para o contexto educacional.

Tabela 3.1: Descrição das expressões para etiquetagem do MAC - Parte A

ETIQUETA	DESCRIÇÃO
<i>Cadê?</i>	Ocorre quando o usuário sabe a operação que deseja executar mas não a encontra de imediato na interface. O principal sintoma desta ruptura é a procura pela operação na interface, inspecionando vários elementos de interface sem ativá-los (e.g. abrindo e fechando menus e submenus ou passando o cursor de mouse sobre botões para ver a dica associada).
<i>Ué, o que houve?</i>	Identificado quando o usuário não percebe a resposta dada pelo sistema a uma ação sua (e.g. a resposta é muito sutil, ou mesmo inexistente) ou não é capaz de entendê-la. Os sintomas típicos incluem repetir a ação ou buscar uma forma alternativa de alcançar o resultado esperado.
<i>E agora?</i>	O usuário não sabe o que fazer e procura descobrir qual é o seu próximo passo. Os sintomas incluem vagar com o cursor do mouse sobre a tela e iniciar um caminho aleatório de interação.
<i>Epa!</i>	O usuário realiza uma ação indesejada e, ao perceber isto, imediatamente desfaz a ação. Os sintomas incluem o acionamento imediato do <i>Undo</i> ou o cancelamento de um quadro de diálogo aberto indevidamente. Observe-se que este 2º sintoma poderia ser visto também como parte de uma busca (“Cadê?”). O “Epa!” se diferencia por ser uma única ação e não parte de uma seqüência maior.
<i>Assim não dá.</i>	O usuário realiza uma seqüência de ações e acredita estar seguindo por um caminho improdutivo, interrompendo-o e cancelando-o. Os sintomas incluem o acionamento de <i>Undo</i> repetidas vezes, a interrupção de um caminho guiado pelo sistema ou ainda o cancelamento quadros de diálogos relacionados. A diferença entre o “Assim não dá.” e o “Epa!” é que o primeiro envolve várias passos, enquanto o “Epa!” envolve apenas um.
<i>Onde estou?</i>	O usuário tenta efetuar operações que não são apropriadas para o contexto em que se encontra, mas o seriam para outros contextos do sistema, indicando uma confusão em relação ao contexto com o qual está interagindo (e.g. tenta editar um elemento da interface disponível apenas para visualização). Um sintoma típico é desfazer a ação incorreta e mudar em seguida para o contexto desejado.

Tabela 3.2: Descrição das expressões para etiquetagem do MAC - Parte B

ETIQUETA	DESCRIÇÃO
<i>O que é isto?</i>	Ocorre quando o usuário não sabe o que significa um elemento de interface. O principal sintoma consiste em deixar o cursor do mouse sobre o elemento por alguns instantes, à espera de uma dica. Outro sintoma é quando o usuário abre menus e submenus ou quadros de diálogos para ver a que se referem. Observe que este sintoma pode acontecer também para a expressão “Cadê?”. A diferença está na intenção do usuário. Quando o usuário está procurando algo então este mesmo sintoma seria um “Cadê?”, se está explorando a interface, então seria “O que é isto?”.
<i>Por que não funciona?</i>	A operação efetuada não produz o resultado esperado, e o usuário não entende o por quê (ou não se conforma com o fato). O sintoma é quando o usuário executa uma ação (ou seqüência de ações), percebe que não obteve o resultado desejado e então repete sua ação na tentativa de identificar a causa de não ter atingido o efeito esperado e corrigi-la.
<i>Socorro!</i>	O usuário não consegue realizar sua tarefa através da exploração da interface e recorre a signos de meta-comunicação para conseguir entender e dar continuidade à sua tarefa. O sintoma é recorrer aos sistemas de ajuda (e.g. agentes automáticos ao qual se pode fazer uma pergunta, ou funções de ajuda que apresentam explicações a elementos da interface em um contexto), documentação (eletrônica ou impressa), ou mesmo pedir explicação a outra pessoa.
<i>Vai de outro jeito.</i>	O usuário não consegue realizar a tarefa da forma prevista como preferencial pelo projetista, e resolve seguir outro caminho, geralmente mais longo ou complicado. Cabe ao avaliador determinar, se possível junto ao projetista, qual é a forma preferencial de execução da tarefa. Normalmente, as formas mais salientes na interface são as consideradas preferenciais. O sintoma é a tentativa frustrada de executar uma ação utilizando a forma preferencial, seguida da adoção de uma solução alternativa, ou mesmo a direta da solução alternativa, sem dar sinais de conhecimento da existência da forma preferencial.
<i>Não, obrigado.</i>	O usuário conhece a solução preferencial do designer, mas opta explicitamente por uma outra forma de interação. O sintoma é o usuário utilizar a ação preferencial (ou demonstrar conhecê-la) e depois utilizar uma ou mais formas alternativas para se alcançar o mesmo resultado.

Tabela 3.3: Descrição das expressões para etiquetagem do MAC - Parte C

ETIQUETA	DESCRIÇÃO
<i>Para mim está bom.</i>	O usuário acha equivocadamente que concluiu uma tarefa com sucesso. O sintoma típico é encerrar a tarefa e indicar na entrevista ou no questionário pós-teste que a tarefa foi realizada com sucesso. O observador, no entanto, sabe que se trata de um engano, provavelmente causado por uma falha de resposta do sistema ou modo de visualização inadequado para a tarefa atual.
<i>Desisto.</i>	O usuário não consegue fazer a tarefa e desiste. O sintoma é a interrupção prematura da tarefa. A causa pode ser falta de conhecimento, tempo, paciência, informação necessária, etc.

Capítulo 4

Análise do Método de Inspeção Semiótica Para o Domínio Educativo

Neste capítulo será apresentada a aplicação do Método de Inspeção Semiótica (MIS), proposto pela EngSem, em três sistemas de apoio ao aprendizado, que têm como objetivo auxiliar e/ou facilitar a compreensão dos alunos sobre um determinado conteúdo, através do uso de recursos computacionais. Os sistemas avaliados foram *VCalc* - que facilita a investigação de taxa de variação de funções; *ProfesSort* [de Castro & Prates, 2009] - que auxilia os alunos na consolidação da aprendizagem de métodos de ordenação; e *Bipide* [Vieira et al., 2009] - que permite ao aluno criar, simular e executar códigos na linguagem portugal.

Na seção 4.1 é descrita a metodologia utilizada neste trabalho, nas seções 4.2, 4.3 e 4.4 são apresentados os sistemas avaliados, enquanto a seção 4.5 apresenta a análise realizada com a aplicação do MIS e sua aplicabilidade para domínio educacional (seção 4.6). Assim, o objetivo deste capítulo é fazer uma apreciação da aplicabilidade do Método de Inspeção Semiótica no domínio educacional, considerando a participação do especialista em EngSem. A seguir é descrita a metodologia utilizada para condução deste trabalho.

4.1 Metodologia

A metodologia utilizada ao longo deste trabalho consiste nas seguintes etapas:

1. *Levantamento bibliográfico*: consistiu no levantamento bibliográfico de alguns

métodos propostos para avaliação de ambientes educacionais, conforme apresentado no capítulo 2.

2. *Aplicação do MIS*: aplicação com especialista em IHC e EngSem do Método de Inspeção Semiótica. O argumento é de que sua aplicação é direta, independente de tecnologia [de Souza et al., 2006], pois permite avaliar a comunicabilidade de um sistema interativo qualquer (ver capítulo 4).
3. *Aplicação do MAC*: aplicação com alunos do Método de Avaliação de Comunicabilidade. A aplicação deste método teve como objetivo coletar indicadores sobre a aplicabilidade do MAC, contrastar com Prates et al. [2004] e identificar possíveis considerações a serem feitas para o domínio educacional (ver capítulo 5).
4. *Proposta do Método de Inspeção Semiótica Intermediado (MISI) e estudos de caso*: nesta etapa, foi proposto um novo método que permite ao professor avaliar a comunicabilidade de um sistema educacional. Para validar este método, foram realizados dois estudos de caso (ver capítulo 6).

Para uma melhor apreciação e entendimento das avaliações realizadas, as próximas seções apresentam os três sistemas educacionais utilizados neste trabalho: VCalc, ProfesSort e Bipide. Para cada sistema, é feita uma descrição detalhada, incluindo as principais características de cada um deles.

4.2 VCalc

Disciplinas como a matemática lidam com conteúdos abstratos e de difícil representação e manipulação. Assim, o uso de sistemas interativos apresenta-se como grande potencial de aprendizagem neste domínio, por permitir a criação de objetos que podem ser manipulados concretamente na tela do computador [Gomes et al., 2002; Gravina & Santarosa, 1998].

Considerando o potencial dos sistemas no contexto educacional e a necessidade de avaliá-los, utilizou-se, neste estudo de caso o aplicativo VCalc¹, desenvolvido no Departamento de Matemática (DMAT) da UFMG para ser utilizado nas aulas de Cálculo I. Este aplicativo tem como objetivo facilitar a investigação da taxa de variação de funções em um determinado intervalo escolhido pelo usuário, mostrando tanto a parte algébrica quanto a gráfica. Na figura 4.1 é apresentada a tela principal do VCalc.

¹Disponível em <http://www.mat.ufmg.br/gepemnt/vcalc>. Acesso em 16 mai. 2010.

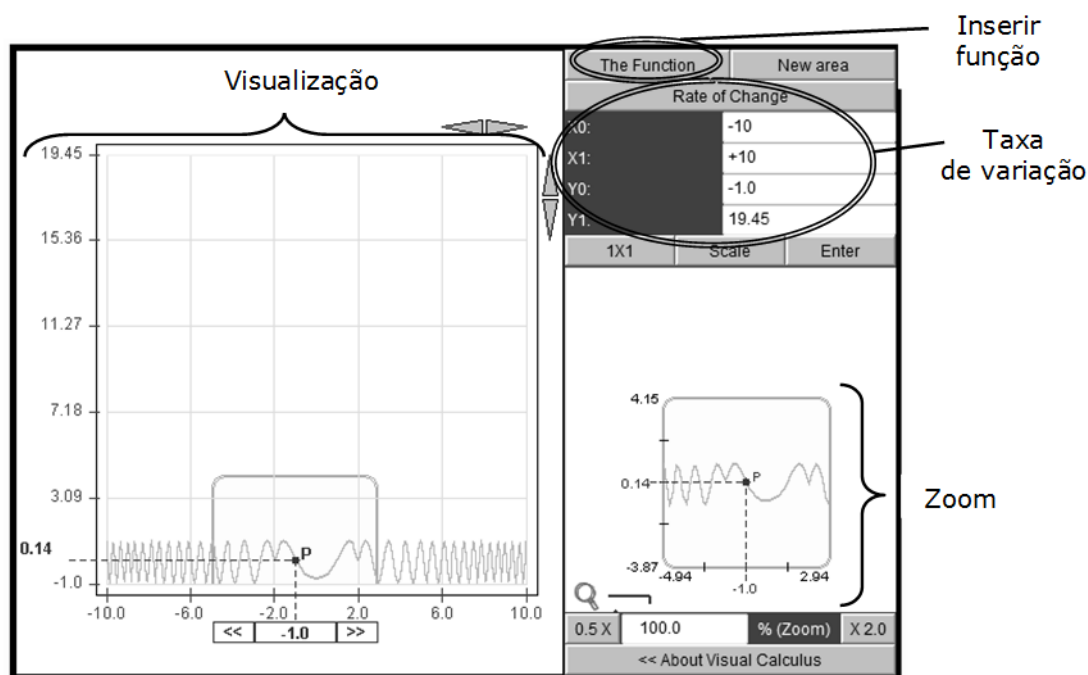


Figura 4.1. Tela principal do VCalc

Esta ferramenta é útil para a exploração gráfica e para a análise do comportamento local de funções. O VCalc não tem por objetivo ensinar funções matemáticas, mas consolidar o aprendizado feito em sala de aula. Esta ferramenta é constituída por um *applet* em que o usuário informa a função que deseja analisar e define a região eixo x e y , para manipulação. Por meio do recurso de *zoom*, o usuário pode visualizar, com maior precisão, o que ocorre com o gráfico à medida que aumenta ou diminui o *zoom*. Como esta visualização gráfica é difícil de ser representada com lápis e papel, este aplicativo preenche esta lacuna ao permitir a manipulação e visualização gráfica da função.

O VCalc foi escolhido pela sua simplicidade, pelo fato de ser utilizado por professores do DMAT da UFMG para apoiar o ensino de derivada e devido à formação na área de matemática de um dos pesquisadores que participou da avaliação deste aplicativo, e de sua experiência em um grupo de pesquisa que estuda o ensino de matemática com o uso de novas tecnologias.

4.3 ProfesSort

O ensino de algoritmos e métodos de ordenação são considerados fundamentais em cursos da área de computação. O principal objetivo desta aprendizagem é propiciar o

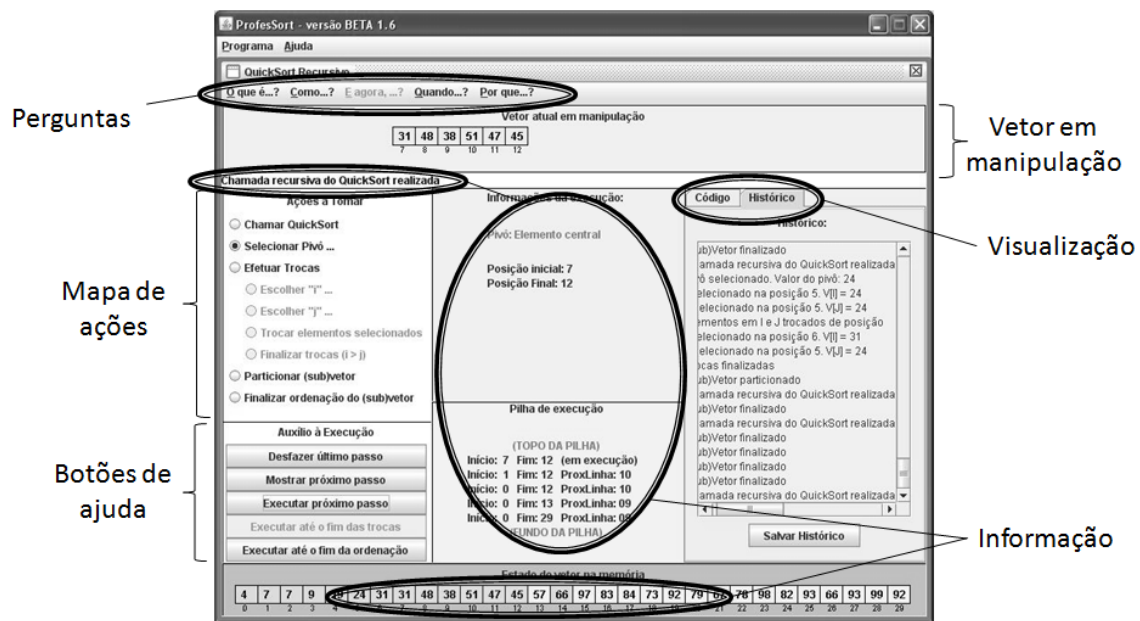


Figura 4.2. ProfesSort: tela principal

desenvolvimento da lógica de programação nos alunos, que será amplamente utilizada durante o curso [Hostins & Raabe, 2007]. Entretanto, a aprendizagem de conceitos de programação é considerada desafiadora para muitos estudantes iniciantes, por possuir alto índice de problemas de aprendizagem, desistências e reprovações [Mota et al., 2008].

Com o objetivo de auxiliar os alunos na aprendizagem desses algoritmos, foi desenvolvido no Departamento de Ciência da Computação (DCC) da UFMG a ferramenta ProfesSort [de Castro & Prates, 2009], para ser utilizada na disciplina de Algoritmos e Estruturas de Dados II (AEDs II).

Por meio da execução passo a passo dos algoritmos, os alunos têm a oportunidade de compreender melhor e exercitar alguns dos algoritmos de ordenação estudados em sala, como *heapsort*, *quicksort*, *seleção*, *inserção* [Cormen et al., 2001; Ziviani, 2004]. O ProfesSort utiliza-se de *scaffolds* para facilitar ou consolidar o aprendizado dos alunos que o utilizarem.

Por meio dos *scaffolds* o ProfesSort oferece aos alunos: (i) perguntas contextuais sobre conceitos e passos do método de ordenação selecionado; (ii) mensagens de erro graduais que não apresentam a solução diretamente, encorajando o aluno a raciocinar sobre o erro e encontrar a solução; (iii) visão geral de todos os passos a serem executados pelo método de ordenação selecionado, permitindo ao aluno identificar o passo a ser executado a cada momento; (iv) botões de auxílio que executam um ou mais passos automaticamente ou mostram ao usuário a próxima ação a ser executada. A figura 4.2

apresenta a tela principal do ProfesSort e seus principais elementos.

O ProfesSort tem como objetivo educacional não apenas permitir ao aluno automatizar a execução dos algoritmos de ordenação, mas também possibilitar seu entendimento. Assim, o sistema permite ao aluno tirar suas dúvidas sobre o método de ordenação durante a execução do algoritmo, além de permitir um melhor entendimento do que significam os passos envolvidos em cada método de ordenação, como executar cada passo e o que fazer em cada momento. A escolha do ProfesSort deve-se ao fato dele ser uma ferramenta indicada pelos professores de AEDS II do DCC/UFMG para consolidar a aprendizagem dos algoritmos de ordenação.

4.4 Bipide

Disciplinas como Arquitetura e Organização de Computadores são fundamentais na formação de alunos dos cursos de computação, pois permitem ao aluno estabelecer relações dos conceitos de lógica com aspectos concretos do *hardware*, o que reduz o nível de abstração envolvido nesta aprendizagem [Borges & Silva, 2006]. Entretanto, os modelos de processadores tipicamente utilizados são muito abstratos e não permitem estabelecer essas relações [Morandi & et al., 2006].

Com a finalidade de reduzir esta abstração foi criado, no Laboratório de Sistemas Embarcados e Distribuídos da Universidade do Vale do Itajaí, em Santa Catarina, o sistema denominado *Basic Instruction-set Processor Integrated Development Environment* (Bipide). O Bipide [Vieira et al., 2009] tem como objetivo possibilitar o uso dos processadores BIP na apresentação de conceitos das disciplinas de Algoritmos e Programação. O Bipide [Vieira et al., 2009] é um ambiente de desenvolvimento que permite ao aluno desenvolver, executar e simular programas em linguagem Portugol², relacionando-os à arquitetura dos processadores BIP. Este ambiente é composto por três módulos principais [Vieira et al., 2009], conforme pode-se observar na figura 4.3.

1. *Programação*, neste módulo o usuário pode escrever e compilar programas na linguagem Portugol;
2. *Simulação*, onde são simulados os programas criados pelo usuário; e
3. *Ajuda*, na interface deste módulo são apresentadas informações sobre as principais funcionalidades do sistema e sobre a arquitetura e organização dos processadores BIP.

²Portugol é uma pseudolinguagem utilizada na descrição de algoritmos.

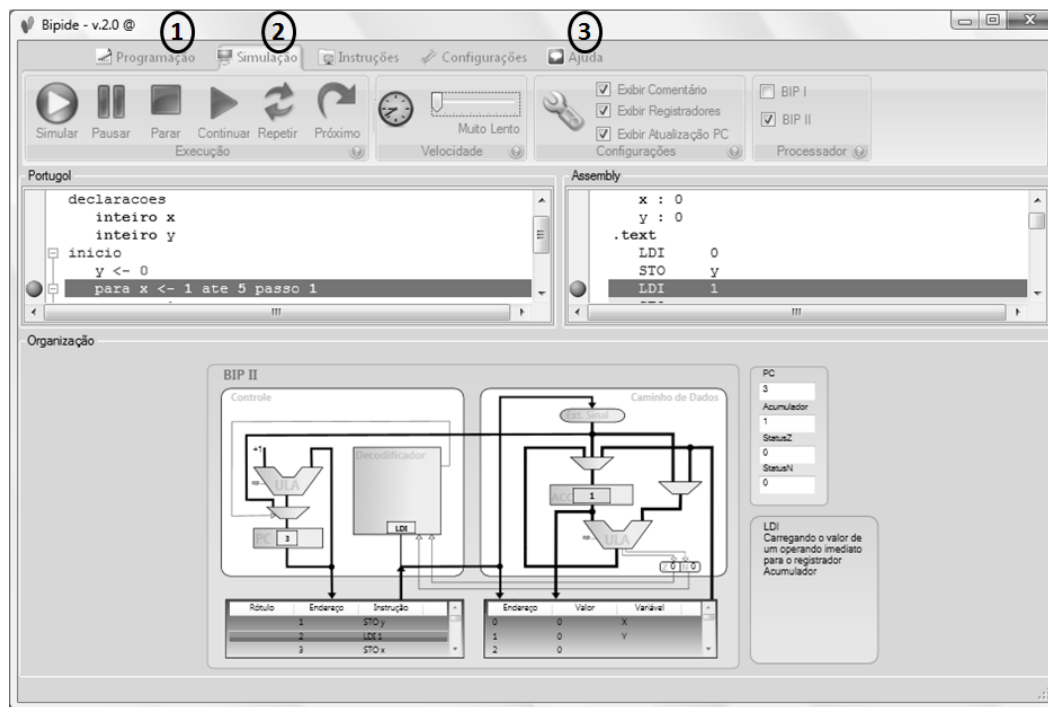


Figura 4.3. Bipide: Tela de Simulação

O Bipide foi escolhido por ser um ambiente interativo com interface bem elaborada, fácil de usar e, principalmente, por simular o funcionamento do *hardware* ao processar instruções *assembly*. Este sistema não era utilizado no DCC/UFMG, no entanto, vislumbrou-se a possibilidade de realização de testes com alunos da disciplina Organização de Computadores, do curso de computação desta instituição.

4.5 Avaliação Com a Participação do Especialista

Conforme apresentado na seção 3.2, o MIS [de Souza et al., 2006; de Souza & Leitão, 2009; de Souza et al., 2010] é um método de inspeção que se concentra na emissão da mensagem sendo enviada do projetista para o usuário. Assim, o MIS é a visão do especialista em IHC. Como o foco do MIS é na comunicação, e não em princípios de interação, o argumento é que sua aplicação seja independente de tecnologia, pois permite avaliar a comunicabilidade de um sistema interativo qualquer [de Souza et al., 2006].

Para avaliar a aplicabilidade do MIS e sua independência de tecnologia para o domínio educacional, realizaram-se três avaliações nos sistemas educacionais de apoio à aprendizagem apresentados nas três últimas seções, VCalc, ProfesSort e Bipide. A avaliação do VCalc contou com a participação de três especialistas em IHC e EngSem,

sendo que um deles era também especialista em matemática. Já a avaliação do ProfSort e do Bipide contou com a participação de um especialista em IHC e EngSem. A aplicação do MIS nesses estudos de caso seguiu a metodologia apresentada na seção 3.2.1 e contou com a elaboração de diferentes cenários, específicos para cada sistema, mas com o mesmo foco, analisar a metacomunicação do projetista para o usuário em sistemas de apoio à aprendizagem.

Nas próximas seções, serão mostrados os principais resultados da aplicação do MIS para cada um dos sistemas avaliados.

4.5.1 Aplicação do MIS no VCalc

A análise do MIS no VCalc foi feita no 2º semestre de 2007 e contou com a participação de três avaliadores. Dois deles eram alunos da pós-graduação, sendo que um deles tinha formação em matemática computacional, ambos cursavam a disciplina de IHC e aplicavam o MIS pela primeira vez, enquanto o outro avaliador havia cursado a disciplina de IHC, em nível de graduação, e aplicava o MIS pela segunda vez.

Para a **preparação**, inicialmente definiu-se o objetivo da pesquisa: *analisar a metacomunicação do projetista para o usuário do sistema de apoio à aprendizagem VCalc*. Para elaboração do material a ser utilizado durante a avaliação, realizou-se uma inspeção informal do sistema. Esta inspeção foi feita por meio da leitura do *help* do sistema VCalc e uma análise superficial de sua interface, com o objetivo de identificar os usuários do sistema e as principais atividades que estes usuários poderiam realizar neste aplicativo. Como o VCalc é um sistema pequeno, simples e com poucas funcionalidades, toda a sua interface foi avaliada, desde a inserção de uma função à verificação de como esta função se comporta no gráfico, utilizando também as taxas de variação. O material para inspeção do VCalc utilizando o MIS encontra-se no anexo A.

Nesta inspeção inicial ficou claro o objetivo do aplicativo e a identificação de um potencial usuário do sistema: alunos da disciplina de Cálculo que desejam verificar a taxa de variação de função. A partir desta inspeção, gerou-se o seguinte cenário:

No Departamento de Matemática da UFMG vários professores têm utilizado aplicativos como complemento às aulas de matemática. Em particular, o Grupo de Estudos e Pesquisa em Educação Matemática e Novas Tecnologias (GEPENNT) têm desenvolvido aplicativos voltados para o ensino de Cálculo Diferencial e Integral e testado esses novos recursos nos processos de ensino e aprendizagem (aulas e oficinas) elaborados pelo grupo. Porém,

até o momento, não foi feita nenhuma avaliação de comunicabilidade nestes aplicativos e, por isso, o grupo solicitou os serviços de especialistas na área de IHC e EngSem.

Em um primeiro momento foi pedido a avaliação do aplicativo VCalc. O VCalc tem sido utilizado no ensino de taxa de variação e de derivada, em aulas de caráter investigativo. Nestas aulas os alunos, diante do software, realizam atividades de investigação de conceitos da matemática, por meio de manipulações gráficas, zoom e outros recursos. Uma das tarefas que eles fazem com frequência é plotar a função $x \cdot \cos(x)$ e mostrar, utilizando o applet VCalc, qual será, aproximadamente, a derivada desta função no ponto $x = -3.43$.

A seguir, são apresentados os passos do MIS utilizados para a realização desta inspeção e os principais problemas encontrados em cada etapa.

1. **Análise dos Signos Metalinguísticos:** nesta etapa os avaliadores realizaram uma inspeção em profundidade da ajuda *online* do VCalc, que na época estava contida no endereço eletrônico da própria ferramenta. Esta análise permitiu identificar elementos relativos à metacomunicação importantes para a comunicação projetista-usuário.

A documentação *online* do aplicativo VCalc é curta e apresenta brevemente seu objetivo, as ações disponíveis para atingi-lo e aspectos operacionais³ do sistema. Na documentação, são apresentadas as principais funcionalidades deste aplicativo, além das operações e funções suportadas por ele, como (i) inserir uma nova equação e definir o intervalo dos eixos x e y ; (ii) verificar a taxa de variação da função inserida e (iii) visualizar, graficamente, o comportamento desta função conforme o intervalo fornecido. Considerando o fato de ser um aplicativo voltado para o domínio educacional, teve-se a preocupação de deixar claro nos signos metalinguísticos o objetivo educacional do aplicativo, conforme pode ser observado na figura 4.4.

Um outro problema encontrado se refere à ausência de mensagens de interação, ou informativas, que dificultam o uso desta ferramenta pelos alunos. Se um aluno, ao verificar o valor da derivada num ponto do gráfico, digitar uma vírgula no lugar de um ponto (i.e. 3,43 ao invés de 3.43) faz o sistema travar. A ausência de mensagens de interação podem gerar rupturas de interação que desestimulam

³Aspectos operacionais são formas de como usar o sistema e como funcionam os signos presentes na interface.

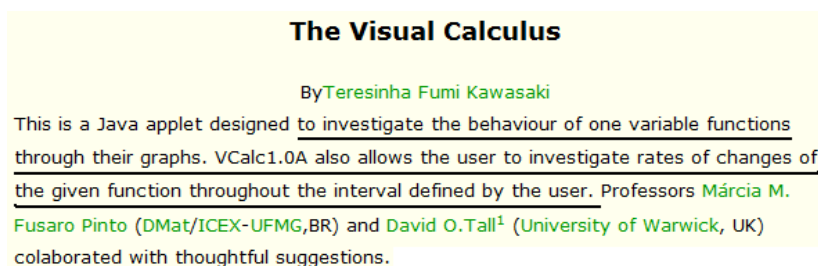


Figura 4.4. VCalc: Objetivo Educacional



Figura 4.5. VCalc: Janela para Selecionar Nova Área

o uso do sistema pelos alunos. Como o VCalc não tem o objetivo de apoiar o aprendizado de forma ativa, mas apenas através da visualização das funções, a ausência de mensagens sobre a aprendizagem pode dificultar esta visualização.

De maneira geral, a descrição sobre o VCalc é concisa e suficiente para explicar o propósito do aplicativo. Entretanto, a apresentação não explicita sobre o perfil de aluno para o qual o sistema é destinado. Por meio das funcionalidades apresentadas no sistema de ajuda, pressupõe-se que seja para um aluno que deseja investigar o comportamento gráfico de funções e a taxa de variação de uma função em um determinado intervalo. Além disso, não há indicações sobre os benefícios de se utilizar este aplicativo para fins educacionais, com relação ao ganho obtido pelo aluno para ampliar o seu conhecimento.

- 2. Análise dos Signos Estáticos:** uma observação inicial feita pelos avaliadores é que o aplicativo combina instruções em inglês e português, conforme pode-se observar na figura 4.5. Esta combinação pode dificultar o uso do sistema por alunos que não têm domínio do inglês, além de ser inconsistente em termos de qualidade de interface.

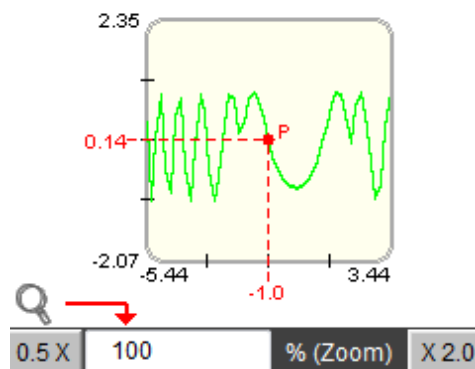


Figura 4.6. VCalc: Opção de zoom

Também verificou-se que a representação dos signos utilizados pode dificultar a comunicação do projetista com o usuário, pelo fato dos signos não lhe serem familiares. Por exemplo, os signos usados para aumentar ou reduzir a área (modificar o zoom), conforme pode-se observar na figura 4.1, diferem muito dos signos convencionais. Apesar de ter a imagem de uma lupa, este signo não é clicável e os signos que aumentam ou reduzem a área gráfica são representados por 0.5X e X2.0, conforme mostrado na figura 4.6. Apenas observando estes signos não é possível saber quando está se ampliando ou reduzindo o *zoom*. Além disso, a *affordance*⁴ para representação dos signos *The Function*, *New Area* e *Rate of Change* não lembra a ideia de botões, o que sugere que os dados podem ser alterados a partir de um clique.

Um importante signo estático identificado no VCalc é o ponto *P*, indicado pela cor vermelha na área central da tela principal. Apesar de ser um elemento importante para o aluno observar o deslocamento do gráfico, sua manipulação é direta e com isso, não há indicações de como ele pode ser manipulado.

Para a reconstrução da metamensagem baseada nos signos estáticos, pode-se concluir que a tela principal do VCalc (ver figura 4.1) comunica que o usuário tem domínio dos conceitos matemáticos envolvidos, tais como taxa de variação, restringindo assim o perfil do aluno que pode utilizá-lo. Além disso, é esperado que ele conheça algumas palavras no idioma inglês, pois algumas telas mesclam opções em português e inglês. Observou-se também que o aluno deseja investigar graficamente, por meio de manipulação direta, o comportamento local de funções, uma vez que parte da tela principal trata-se de uma área gráfica.

⁴Termo utilizado para se referir ao atributo de um objeto que permite às pessoas saberem como utilizá-lo. [Preece et al., 2004]

3. **Análise dos Signos Dinâmicos:** dentre os resultados identificados nesta etapa, destacam-se alguns problemas encontrados na comunicação projetista-usuário.

Nas categorias alterar e visualizar os resultados no gráfico, há inconsistência lógica entre estes parâmetros. Para alguns parâmetros, uma janela se abre (e.g. signo a função); para outros, a edição é feita na interface principal (e.g. signos X0, X1); e para outros é indicado que o resultado apareça substituindo o próprio gráfico (e.g. signos 1x1, Esc).

Apesar do clique sobre o ponto P (ponto vermelho) para movimentá-lo estar bem comunicado, ao utilizar esse recurso para movimentação desse ponto, o aluno poderá ter problemas. Uma vez que clica-se neste signo, é disparada uma animação e a única forma de pará-la é clicando na área central deste signo. Entretanto, não há indicação de que a manipulação deve ser feita desta forma, ou seja, o aluno precisa interagir para descobrir o funcionamento deste signo. Caso o aluno precise definir com precisão e rapidez um ponto específico da função no gráfico, ele terá dificuldades em realizar esta ação por meio deste botão, pois a única forma de interagir com este signo é por meio de manipulação direta.

Além disso, a funcionalidade de alguns signos da interface é errônea. O tradicional botão “*fechar*” não funciona da forma esperada, pois tem *bugs*, o que obriga o usuário a procurar outros meios para fechar uma janela que foi aberta. Apesar dos *bugs* não serem intencionais, eles podem gerar rupturas para o usuário. Outro signo com problema é a opção *help* que é apresentada quando se clica em algumas telas. O aluno que recorre a este recurso não obtém qualquer *feedback* do sistema. A ação de clicar neste signo não tem nenhum efeito, é como se ele estivesse desabilitado.

De forma geral, os problemas identificados nesta etapa se referem a rupturas de interação. Essas várias rupturas que podem ser geradas desestimulam o aluno a utilizar o VCalc, podendo levá-lo a desistir do apoio educacional que ele oferece.

Em termos de metacomunicação, concluiu-se que o usuário alvo do VCalc são alunos que precisam entender e investigar o comportamento local de funções e a sua taxa de variação graficamente. A principal forma de utilizar o sistema é por meio da manipulação direta. Além disso, é importante que o aluno tenha curiosidade para descobrir como manipular alguns elementos da interface.

4. **Comparação das três metamensagens:** a partir da reconstrução das metamensagens geradas nos três passos anteriores (metalinguístico, estático e dinâmico), verificou-se que, para alguns signos, a interpretação foi divergente.

Os signos referentes à alteração de parâmetros (*The Function*, *New Area*, *Esc*, etc.) não estão presentes nos signos metalinguísticos. Com isso, o usuário identificará a funcionalidade destes signos somente a partir da interação com o sistema.

As funcionalidades do VCalc identificadas na inspeção dos signos metalinguísticos se confirmou nos demais passos. Do ponto de vista educacional, isso é muito importante, por evidenciar, desde o *help*, qual é o objetivo educacional do sistema e que conteúdo ele aborda.

Nesta análise foram identificadas as seguintes classes de signos:

- *Acesso e alteração de parâmetros em janelas, na forma de botões*: estes signos referem-se à *The Function*, *New Area* e *Rate of Change*. Ao selecionar uma destas opções, uma nova janela é aberta para que o usuário execute uma nova ação. Outros signos que também se encaixam nesta classe são os campos para alterar os parâmetros da função.
- *Alteração direta dos parâmetros*: estes signos referem-se às opções *X0*, *X1*, *Y0*, *Y1* e *zoom*. Nestes signos, o usuário faz a manipulação direta com o sistema.
- *Comandos e definições de alterações*: referem-se às opções de confirmar, cancelar e ajuda.
- *Visualização*: referem-se aos signos que permitem a visualização gráfica de funções (i.e. a área que contém o ponto vermelho *P*) e também os signos de movimentação (similar à barra de rolagem) da janela de visualização.

5. **Avaliação Global da Comunicabilidade**: considerando o objetivo desta avaliação como sendo a *análise da metacomunicação do projetista para o usuário do VCalc*, foi possível verificar algumas questões relacionadas à comunicabilidade deste aplicativo.

O material com informações sobre o sistema e a forma como ele é apresentado permitiram identificar o perfil do usuário do sistema: alunos que precisam entender e investigar o comportamento local de funções e a sua taxa de variação. Além disso, os avaliadores concluíram que o VCalc pretende ser um mecanismo para facilitar o estudo e consolidação do aprendizado de funções de forma gráfica.

Também identificou-se a necessidade de conhecimento prévio por partes dos alunos sobre o conteúdo abordado pelo VCalc. O aluno que entende os conceitos envolvidos no aplicativo pode sentir-se apto a utilizá-lo. Caso ele não

entenda os conceitos, ele terá dificuldades em utilizar o VCalc, ou ainda subutilizar o sistema, por não ter o domínio desejado. O VCalc não fornece formas de guiar o aluno no uso do sistema para obter o aprendizado desejado. O aluno é o responsável por interpretar, de forma autônoma, os gráficos mostrados e associar significado a eles.

Outro ponto que recebeu uma atenção especial refere-se à manipulação direta. As interfaces que utilizam-se do recurso de manipulação direta normalmente não têm uma forma clara de comunicar que tipo de manipulação é válida, apenas por tentativa e erro. No caso do VCalc, poderiam ter signos redundantes na interface para auxiliar neste entendimento (e.g. através do menu, signos estáticos; ou help, signos metalinguísticos). Esta é uma importante questão a ser repensada para o *redesign* da interface do VCalc.

Do ponto de vista educacional, um dos avaliadores que possuía maior conhecimento sobre o domínio matemático identificou uma questão relevante com relação ao recurso de *zoom*. Na matemática, ao dar um *zoom* em uma função, permite ao aluno perceber não só o fato de que a imagem se aproximou, mas pode-se perceber também que uma curva se transformou em uma reta. Esta abstração na matemática é difícil de ser ilustrada utilizando apenas um quadro negro e giz. Por isso, os recursos computacionais oferecidos pelo VCalc podem auxiliar os alunos a entender este tipo de abstração.

Apesar de alguns signos presentes no VCalc gerarem dúvidas devido aos potenciais problemas de comunicação encontrados, verificou-se pontos positivos sobre esta ferramenta. Considerando que o VCalc é uma ferramenta de apoio à aprendizagem, constatou-se que este aplicativo contém poucas funcionalidades e a sua utilização é bem específica, o que facilita a sua compreensão por parte dos alunos. Com pouco tempo de utilização, os alunos conseguem realizar praticamente todas as funções que o sistema apresenta e estarão aptos a apreciar o conteúdo, ao invés de focarem a atenção na aprendizagem do aplicativo.

Por fim, é apresentada a metamensagem consolidada sobre o VCalc.

O VCalc é um aplicativo feito para você aluno, que precisa entender e investigar o comportamento local de funções e a sua taxa de variação. Ele vai ajudá-lo a verificar taxas de variação em funções, conforme um intervalo definido pelo usuário. Como você precisa entender melhor o comportamento da função, você poderá fazer o uso do zoom, e verificar, local ou globalmente, o seu comportamento. Entretanto, os botões de

zoom não são muito claros, por não evidenciarem qual botão aumenta ou diminui o zoom. E, como você precisa entender a taxa de variação de uma função em um ponto, você poderá observar a inclinação da reta secante que passa por dois pontos informados por você, que fazem parte da curva. A interface é exatamente o eixo x e y , porém alguns botões relativos à interação não são muito claros quanto à sua funcionalidade e, quando utilizados, podem reagir de forma diferente do esperado por você. No entanto, à medida que você usar o aplicativo, você dominará esta e outras opções. O VCalc também não conta com caixas de diálogo explicativas, assim se você cometer algum erro, terá que descobrir sozinho como resolver tal problema. Além disso, você é o responsável por interpretar os gráficos mostrados e associar significado a eles. Como o VCalc tem poucas funcionalidades e a sua utilização é bem específica, você terá facilidade em compreender o sistema.

4.5.2 Aplicação do MIS no ProfesSort

A aplicação do MIS no ProfesSort contou com a participação de um avaliador, aluno da pós-graduação que cursava a disciplina Tópicos Especiais em Engenharia Semiótica no DCC/UFMG, no 2º semestre de 2008. Este avaliador aplicava o método pela segunda vez.

Para a fase de *preparação*, inicialmente definiu-se o objetivo da avaliação *analisar a metacomunicação do projetista para o usuário do sistema de apoio à aprendizagem de métodos de ordenação, ProfesSort*. Em seguida, realizou-se uma inspeção informal do sistema, com a finalidade de elaborar o material para a avaliação. Como o ProfesSort não tinha *help*, foi disponibilizado um documento, a partir de publicações do sistema, que explicava as suas principais funcionalidades. Este documento foi aprovado pelos autores da ferramenta.

Assim, a inspeção foi feita por meio deste documento gerado e também de uma análise preliminar da interface do ProfesSort, com o objetivo de conhecer um pouco sobre a utilização do sistema e também permitir a elaboração das tarefas e a geração dos cenários para aplicação técnica do MIS. Como apresentado na seção 4.3, esta ferramenta possibilita a ordenação com 5 (cinco) algoritmos de ordenação distintos. Como ficaria inviável realizar a inspeção de todos os métodos, selecionou-se dois métodos mais complexos e de difícil entendimento por parte dos alunos, *quick sort* e *heap sort* [Cormen et al., 2001; Ziviani, 2004].

A análise inicial revelou que o ProfesSort pretende auxiliar os alunos no entendi-

mento sobre os passos dos métodos de ordenação e sua execução. Para isso, ele oferece formas de auxiliar o aluno neste aprendizado através da possibilidade de fazer perguntas sobre o método ou solicitar a informação sobre o passo seguinte. Para a elaboração do cenário utilizou-se como base um professor que poderia usar o ProfesSort em sala de aula como ferramenta de apoio à aprendizagem dos alunos em cursos de computação. O cenário gerado é descrito a seguir.

Yasmin é a nova professora da disciplina de Algoritmos e Estrutura de Dados II (AEDs II) do Departamento de Computação. Como ela gosta muito de utilizar o computador como ferramenta para apoiar a aprendizagem, ela resolveu conversar com outros professores do departamento para saber se eles utilizam algum sistema desse tipo. Uma colega do departamento indicou um sistema que ela usa em sala com seus alunos, um sistema denominado ProfesSort. Esse sistema auxilia os alunos da disciplina de AEDs II na consolidação da aprendizagem dos métodos de ordenação de algoritmos. Yasmin então resolve explorar os recursos do ProfesSort e se certificar que ele pode auxiliá-lo no aprendizado de seus alunos. Se a inspeção corresponder às suas expectativas, ela utilizará o ProfesSort para apoiar a aprendizagem dos seus novos alunos.

A seguir, são detalhados os passos do MIS usados na inspeção do ProfesSort, bem como os principais resultados obtidos em cada etapa.

1. **Análise dos Signos Metalinguísticos:** como o ProfesSort não tinha uma documentação que explicasse suas principais funcionalidades, foi necessário gerar um documento com tais informações. Este documento encontra-se no Anexo I. Além deste documento, o ProfesSort tem como signos metalinguísticos as mensagens de erro e as respostas às dúvidas dos alunos.

As mensagens de erro indicam a tentativa de que o próprio aluno possa identificar o erro para prosseguir ou refletir sobre suas ações e tentar identificar sozinho o problema. Caso não consiga, ele recebe uma explicação completa para lhe auxiliar na resolução do erro. As dúvidas são explicações sobre os métodos de ordenação, e não sobre o sistema, que também auxiliam o aluno a esclarecer a dúvida. O conteúdo destes signos metalinguísticos é a fala do professor embutida no sistema e a forma como são disponibilizados durante a interação referem-se à solução do método de ordenação escolhido.

Por meio do documento gerado com informações sobre o ProfesSort, o avaliador realizou uma análise aprofundada desta documentação. A análise dos signos

metalinguísticos permitiu que se identificasse elementos relacionadas à metacomunicação, como (i) o objetivo educacional do ProfesSort (auxiliar os alunos no aprendizado dos algoritmos de ordenação); (ii) as principais funcionalidades do sistema (executar passo a passo os algoritmos de ordenação, e observar e controlar a execução do algoritmo); e (iii) os apoios oferecidos durante a execução dos algoritmos (perguntas guia, mensagens de erro, botões de auxílio à execução e visão geral dos algoritmos).

Apesar de ter sido possível identificar estas informações para a reconstrução da metamensagem, é importante ressaltar a necessidade de se elaborar um *help* ou documentação que seja acoplada ao sistema ProfesSort.

2. **Análise dos Signos Estáticos:** esta análise possibilitou identificar elementos relevantes sobre a comunicação do projetista para o usuário.

Neste passo verificou-se que os menus possuem uma estrutura lógica e com opções habilitadas e desabilitadas, indicando as possibilidades de uso da ferramenta. Percebeu-se também uma ausência de comentários na opção *código*, importante para o aprendizado dos alunos. A existência de comentários no código é um recurso útil para algoritmos, por permitir ao aluno entender o que acontece em cada linha do programa.

Neste passo foram identificadas algumas classes de signos, como (i) perguntas, (ii) botões de ajuda, (iii) visualização e (iv) informação sobre estado atual, conforme pode-se observar na figura 4.2.

- *Perguntas:* apresentam diversas perguntas pré-determinadas que o aluno pode fazer sobre o método e são organizadas de acordo com o contexto em que se encontra.
- *Botões de ajuda:* são botões que permitem uma solicitação de ajuda sobre o método de ordenação escolhido e sendo executado pelo aluno.
- *Visualização:* permite ao aluno visualizar todos os passos do algoritmo que foram executados, bem como uma descrição de alto nível do método de ordenação escolhido.
- *Informação sobre estado atual:* mostra a situação atual do vetor que está sendo manipulado.

Acredita-se que estas classes são apoios que têm como objetivo auxiliar o aluno durante a execução dos algoritmos, para evitar que ele faça uso de outros recursos

adicionais. As perguntas e botões permitem a criação da expectativa do que virá, porém na análise dos signos estáticos ainda não é possível afirmar com precisão que tipo de apoio é oferecido.

Para a reconstrução da metamsagem foi possível concluir que a tela do ProfesSort comunica que o usuário tem um conhecimento prévio sobre métodos de ordenação e pode aprofundar no entendimento destes métodos utilizando o sistema. Como o ProfesSort possui recursos para auxiliar o aluno no aprendizado dos métodos de ordenação, vislumbrou-se a possibilidade de uso desta ferramenta por alunos, para consolidar a aprendizagem vista em sala, e por professores, para ter nesta ferramenta um recurso extra para ensinar os alunos a abstração envolvida nos métodos de ordenação. Concluiu-se também que o aluno, ao utilizar este sistema, deseja consolidar o aprendizado sobre métodos de ordenação e pode inserir um vetor dado pelo professor ou então solicitar que o próprio sistema crie o vetor a ser ordenado, bastando para isso informar o tamanho do vetor a ser ordenado.

3. **Análise dos Signos Dinâmicos:** nesta etapa foram identificados problemas relacionados à interação e também as seguintes classes de signo.

- *Mapa de ações:* estas opções são utilizadas pelos usuários para selecionar o próximo passo do algoritmo de ordenação selecionado.
- *Botões de ajuda:* os elementos desta classe auxiliam o usuário na execução passo a passo do algoritmo de ordenação selecionado. À medida que o usuário interage com a ferramenta, ele recebe dicas sobre a resposta correta, ou seja, qual é o próximo passo, ao invés de receber a resposta diretamente. Este recurso leva o usuário a refletir sobre o aprendizado do método.
- *Informação sobre estado atual:* esta opção mostra aos usuários o vetor que está sendo manipulado. Ele é atualizado à medida que o usuário executa os passos do algoritmo de ordenação selecionado.
- *Visualização:* esta classe apresenta ao usuário opções de visualização sobre a execução do método de ordenação: código, histórico e recursão. A opção de visualização de *Código* apresenta o código do método de ordenação selecionado na linguagem de programação C, enquanto a opção *Histórico* apresenta os passos executados desde o início da execução do algoritmo. A opção *recursão* é uma opção exclusiva do algoritmo *quick sort recursivo*, nela são listadas as chamadas recursivas feitas pelo método.

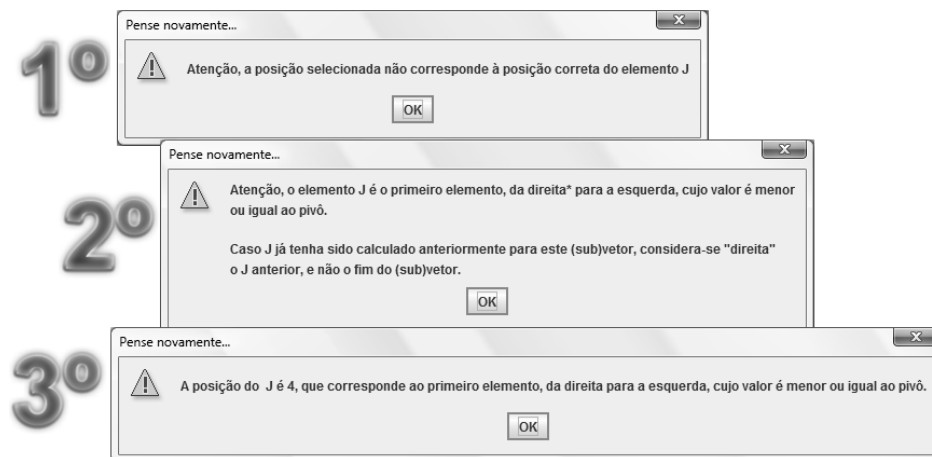


Figura 4.7. ProfesSort: Níveis de Erro

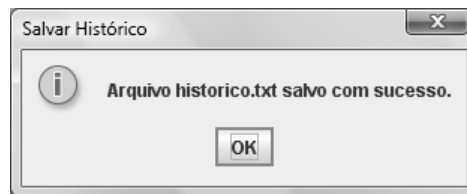


Figura 4.8. ProfesSort: Histórico da Execução do Algoritmo

- *Informação:* foram identificadas três diferentes classes desse tipo e referem-se aos passos para execução do método de ordenação. A primeira classe exibe uma informação que ajudará o aluno a tomar a decisão sobre o que deverá fazer no próximo passo da execução do algoritmo. Na segunda, ela exibe informações sobre a execução do algoritmo, permitindo ao usuário visualizar o valor de cada variável. A terceira só é exibida na tela conforme o método de ordenação escolhido (*quick sort recursivo*, por exemplo) e fica na parte inferior da tela, exibindo o estado do vetor na memória.
- *Níveis de erro:* esta classe oferece mensagens de erro graduais em três níveis, conforme pode-se observar na figura 4.7. Com isso, o ProfesSort não apresenta a solução diretamente, encorajando o aluno a raciocinar sobre o erro e encontrar a solução.

Os principais problemas de interação identificados referem-se a:

- *Mensagem incompleta:* ao salvar o histórico da execução do algoritmo, o aluno precisa descobrir sozinho onde o histórico foi salvo, pois a mensagem não exibe esta informação (ver figura 4.8).

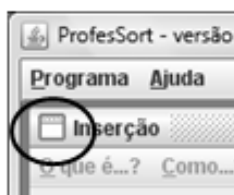


Figura 4.9. ProfesSort: Ausência de Indicação de Funcionalidade do Signo

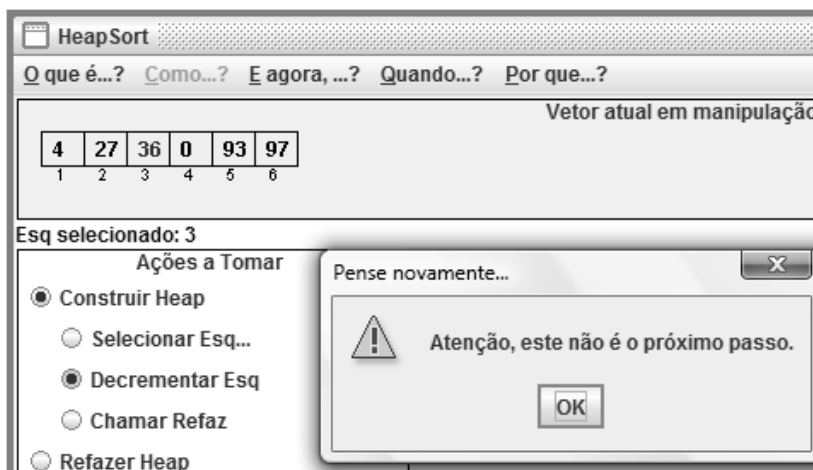


Figura 4.10. ProfesSort: Ações a tomar do método HeapSort

- *Signo não indica sua funcionalidade:* o signo ao lado do nome do algoritmo selecionado (ver figura 4.9) não indica sua funcionalidade. Ao clicar duas vezes sobre ele, a janela atual do algoritmo em execução é fechada, levando o aluno à tela de seleção de algoritmo. Essa funcionalidade só é perceptível quando se clica sobre o signo e não está claro que há um comportamento associado.
- *Sequência da opção Ações a Tomar:* ao executar o método *heap sort*, a disposição das opções *ações a tomar* dá a falsa ideia de que os passos são sequenciais. Como nas demais opções deste recurso os passos obedecem à sequência apresentada, este problema de interação pode induzir o aluno a errar, ou ainda gerar um problema de conteúdo. Ao iniciar a construção do *heap*, o segundo passo é *chamar refaz* e não *decrementar esq*, conforme listado na sequência da figura 4.10.

Com base na análise dos signos dinâmicos, pôde-se concluir que o ProfesSort é um sistema para consolidação do aprendizado de métodos de ordenação que requer um conhecimento prévio por parte do usuário. Como o conteúdo abordado por ele é bem específico, o usuário alvo deste sistema são alunos que aprenderam

ou estão aprendendo os métodos de ordenação e desejam utilizar o sistema para consolidar esta aprendizagem. Também identificou-se a possibilidade de uso desta ferramenta por professores, em sala de aula, como recurso computacional para auxiliá-lo no ensino destes métodos de ordenação. Para auxiliar os alunos no aprendizado dos métodos, o ProfesSort permite a ele executar os algoritmos passo a passo, de forma que ele compreenda bem cada passo. Em caso de dúvidas, o aluno pode recorrer às opções de ajuda oferecidas pelo sistema para saná-las.

4. **Comparação das Três Metamensagens:** como cada nível de signo tem capacidade de expressão diferente, a análise segmentada da metamensagem permitiu-nos identificar o que o projetista tem a dizer sobre cada conjunto de signos. Considerando que o foco desta avaliação foi *analisar a metacomunicação do projetista para o usuário do sistema de apoio à aprendizagem de métodos de ordenação, ProfesSort*, percebeu-se que a metamensagem identificada nos passos 1 (metalingüísticos), 2 (estáticos) e 3 (dinâmicos) em geral, foi consistente.

Nas três análises foi possível identificar o público alvo do sistema, alunos que estavam aprendendo métodos de ordenação, e o seu objetivo educacional, auxiliar os alunos no aprendizado desses métodos. A análise dos signos metalingüísticos e dinâmicos revelou a possibilidade de uso do ProfesSort por professores, como recurso computacional para facilitar a compreensão dos conceitos abstratos envolvidos nos métodos de ordenação pelos alunos. Para um bom uso do sistema, é necessário que os alunos tenham um conhecimento prévio sobre métodos de ordenação e à medida que tiver dúvidas, o ProfesSort o auxilia através dos apoios inseridos nele.

Estes apoios têm como objetivo facilitar a consolidação da aprendizagem dos alunos sobre métodos de ordenação. Com isso, os alunos podem recorrer ao sistema para esclarecer uma dúvida que surgiu durante a interação e que ele talvez nem pensava que poderia ter. Na análise dos signos dinâmicos a sequência das ações a tomar na execução do método *quick sort* merece atenção especial. Apesar de ser um problema de interação, ele pode induzir o aluno ao erro, o que não é desejável em qualquer sistema interativo, especialmente em sistemas educacionais.

5. **Avaliação Global da Comunicabilidade:** com a inspeção do sistema ProfesSort, em todos os passos da aplicação técnica do MIS, não foram encontrados problemas graves que possam dificultar ou impedir sua utilização pelo usuário.

Pela interface apresentada pelo sistema, é possível identificar o seu uso como ferramenta de apoio à aprendizagem de algoritmos de ordenação. Porém, como o sistema não possui um *help* explicando seu funcionamento, o aluno, sem o auxílio do professor, pode ter dificuldades iniciais no uso do ProfesSort. À medida que for interagindo com o sistema, essas dificuldades poderão ser sanadas.

Um recurso identificado como importante para apoiar o aluno na aprendizagem refere-se às opções de apoio oferecidas pelo sistema. Por mais que o aluno que utiliza o sistema esteja aprendendo ou já saiba utilizar os algoritmos de ordenação, ele pode ter dúvidas sobre alguns detalhes específicos de cada algoritmo. Ao invés de recorrer às anotações de aula ou à *sites* de busca, o próprio sistema exibe a resposta a algumas dúvidas que o aluno poderá ter.

Outro recurso importante é o fato de não apresentar, de imediato, a solução para o problema, o que mostra a preocupação com o processo de aprendizagem do aluno pelo professor. Nas mensagens graduais apresentadas ao aluno, à medida que ele erra e no menu de perguntas, percebe-se a preocupação com a parte educacional como parte da mensagem relativa à aprendizagem.

Como resultado desta inspeção, gerou-se a seguinte metamensagem consolidada:

Caro aluno, o ProfesSort é um sistema desenvolvido na linguagem de programação Java que foi criado para você, aluno da disciplina de Algoritmos e Estrutura de Dados e que já aprendeu ou está aprendendo os algoritmos de ordenação. Ele vai ajudá-lo na consolidação do aprendizado de algoritmos de ordenação como seleção, inserção, heap sort e quick sort. Logo na tela inicial, você pode escolher qual algoritmo deseja executar. Você também pode selecionar o tamanho do vetor que utilizará e optar que o próprio sistema faça o preenchimento automático do vetor, ou se preferir, você também pode inserir os elementos que desejar, porém os valores devem estar no intervalo de 0 a 99. Essa opção pode ser muito útil para situações em que o seu professor passar alguma atividade sobre os algoritmos de ordenação, pois você pode conferir se a sua execução está correta. O ProfesSort também permite ao aluno executar o algoritmo passo a passo. Para isso, é necessário que o aluno defina cada passo a ser executado, de forma a consolidar o aprendizado visto em sala. Com isso, o ProfesSort lhe permite compreender melhor e exercitar alguns dos métodos de ordenação estudados em sala, além de permitir a revisão e fixação desses métodos. Ele também lhe permite

observar e controlar a execução do algoritmo indicando, por exemplo, qual é o próximo passo que deve ser executado, quais elementos devem ser escolhidos para troca, dentre outros. Com isso, o ProfesSort garante que você está executando o algoritmo da forma correta, por não deixar você passar para o próximo passo se tiver errado o passo atual. O ProfesSort também apresenta quatro formas de apoio para facilitar ou consolidar o seu aprendizado: (1) perguntas que você pode fazer ao programa quando tiver dúvidas; (2) mensagens de erro graduais, que lhe incentivarão a raciocinar, e não apenas receber a resposta correta sem compreendê-la; (3) visão geral de todos os passos envolvidos no algoritmo, de forma que você possa pensar, ao fim de um passo, qual a próxima ação a ser executada; (4) botões de auxílio que executam um ou mais passos, para que você possa simplesmente acompanhar a execução quando desejar. Na opção histórico, você pode visualizar os passos selecionados durante a execução do algoritmo de ordenação escolhido. Além disso, você pode ainda salvar este histórico no formato txt, para analisar posteriormente. Esse histórico é salvo na mesma pasta onde encontra-se o executável do ProfesSort e todos os históricos ficam armazenados no mesmo documento. Se você salvar muitos históricos e desejar ter cada histórico em um arquivo independente, você terá que fazer isso manualmente.

4.5.3 Aplicação do MIS no Bipide

A aplicação técnica do MIS no sistema Bipide [Vieira et al., 2009] ocorreu no 2º semestre de 2009 e foi feita por um avaliador que então aplicava o MIS pela terceira vez.

Na *preparação* definiu-se o objetivo da investigação: *analisar a metacomunicação do projetista para o usuário para comunicar o funcionamento do hardware ao executar um programa.* A leitura da ajuda e do artigo sobre o Bipide [Vieira et al., 2009], juntamente com a análise informal da interface, permitiram a realização da inspeção do sistema.

Como o Bipide é um sistema com muitas funcionalidades, optou-se por inspecionar a parte da simulação de processadores, que aborda o conteúdo da disciplina de Arquitetura e Organização de Computadores (AOC) e também a parte que envolve conceitos de lógica programação. Uma vez que o usuário, para entender o funcionamento do simulador, precisa inserir um código com instruções lógicas.

A análise inicial demonstrou que o Bipide é um ambiente de apoio ao aprendizado

que possibilita o desenvolvimento, execução e simulação de programas em linguagem Portugol, envolvendo o conceito de AOC. A partir desta inspeção, gerou-se o seguinte cenário:

Fábio é professor do departamento de computação. Ele está preparando o material para sua disciplina Arquitetura e Organização de Computadores (AOC), que começará dentro de alguns dias. Um colega do departamento de computação comentou que ele utilizou o Bipide na sua disciplina do semestre anterior como ferramenta de apoio ao aprendizado dos alunos e a experiência foi muito positiva. Seu colega lhe disse que o sistema Bipide é uma ferramenta gratuita que ajuda os alunos a desenvolver, executar e simular algoritmos na linguagem Portugol. Com isso, os alunos podem simular um código, para entender como o processador interpreta este código. Fábio então resolve explorar os recursos do Bipide para verificar como essa ferramenta pode contribuir para a aprendizagem de seus alunos. Caso o sistema corresponda às suas expectativas, ele utilizará o Bipide como ferramenta de apoio à aprendizagem dos seus alunos de AOC.

A seguir são detalhados os passos da aplicação técnica do MIS, no sistema Bipide, e os resultados obtidos em cada passo.

1. **Análise dos Signos Metalinguísticos:** a documentação *online* do Bipide é bem detalhada, o que facilitou a reconstrução da metamsagem neste nível de signo.

Nesta inspeção, verificou-se que a ajuda apresenta informações didáticas a respeito da arquitetura de computadores, além de explicar o funcionamento dos principais signos da interface, conforme pode-se observar na figura 4.11.

Assim, a comunicação do projetista neste primeiro nível de signo é bem completa e se divide em dois tipos: conteúdo de arquitetura e organização de computadores e as funcionalidades do sistema. O conteúdo subdividi-se em:

- *Arquitetura e organização de computadores*, com informações sobre a teoria envolvida.
- *Processadores*, com informações sobre os processadores BIP.
- *Portugol*, com informações sobre a estrutura geral de um programa na linguagem Portugol.

A opção funcionalidades também subdividi-se em:

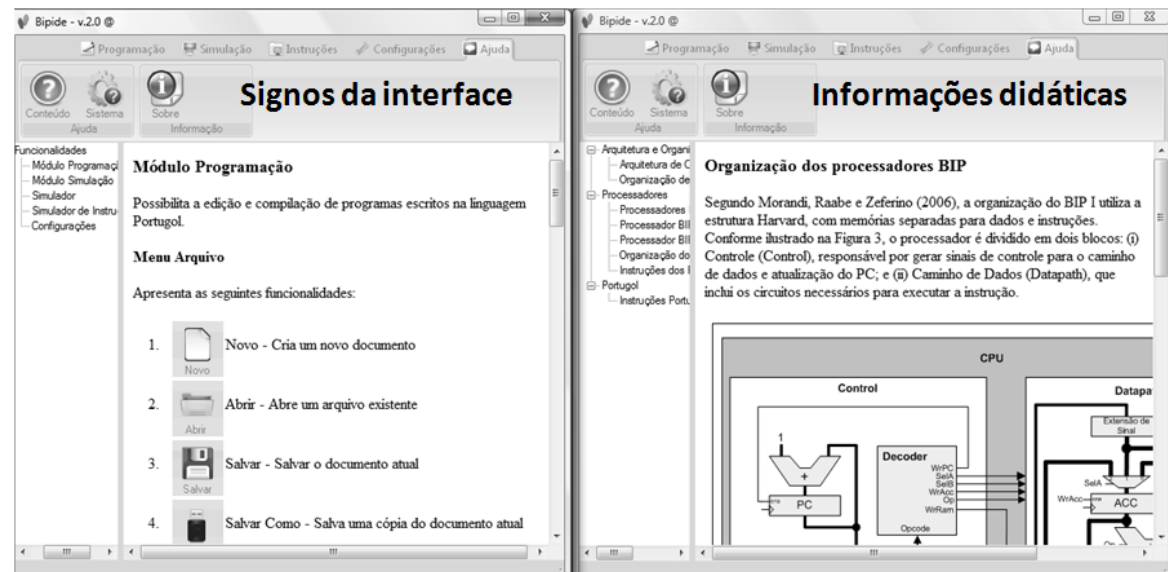


Figura 4.11. Bipide: Telas de Ajuda

- *Programação*: possibilita a edição e compilação de programas escritos na linguagem Portugol. Também apresenta, de forma detalhada, a imagem e o significado de cada signo presente na interface.
- *Simulação*: possibilita a execução e simulação de programas Portugol sobre a arquitetura dos processadores BIP I e BIP II.
- *Simulador*: exhibe a simulação do programa nas arquiteturas dos processador BIP.
- *Configurações*: apresenta opções de configuração do ambiente.

Neste primeiro passo, para reconstruir a metamsagem, foi possível identificar o objetivo do Bipide, auxiliar os alunos no desenvolvimento, execução e simulação de programas em linguagem Portugol. Embora não esteja explícito a quem o sistema se destina, como os projetistas falam de simulação de algoritmos, organização de computadores e da linguagem Portugol, pode-se entender que o sistema se destina a alunos iniciantes nos cursos de computação. Com isso, ele pode ser útil tanto para alunos que estão aprendendo lógica de programação quanto para alunos que estão aprendendo sobre o funcionamento de um processador.

Através dos signos metalinguísticos foi possível perceber que o foco do Bipide não é apenas no uso do sistema, mas também disponibilizar informações sobre o conteúdo que o sistema pretende ensinar/consolidar. Então entende-se que houve a participação do discurso educador e do projetista, de forma integrada, no desenvolvimento desta ferramenta.

2. **Análise dos Signos Estáticos:** nesta etapa, identificou-se as seguintes classes de signos: *(i) arquivo*, com opções de criar, abrir, salvar e imprimir um documento; *(ii) editar*, com opções de recortar, colar, copiar, selecionar, procurar, desfazer e refazer, escrever códigos em Portugol; *(iii) compilar e simular*, permite compilar e simular o programa na linguagem Portugol; *(iv) configurar*, permite configurar a exibição de elementos para simulação.

Outra classe verificada foi denominada Linguagem de Programação, conforme pode-se verificar na figura 4.3, no canto superior esquerdo. Nesta parte da tela é exibido o programa em linguagem Portugol que está sendo simulado e neste editor, a linha do programa que está sendo simulada é destacada em vermelho. A classe *Assembly* também pode ser vista nesta mesma figura, porém do lado superior direito da interface. Nesta janela é exibido o código *assembly* gerado pelo compilador e a linha do programa que está sendo simulada aparece destacada, também na cor vermelha.

Nesta análise, percebeu-se que o projetista fez uso do sistema de significação utilizado em outros sistemas interativos. Com isso, os menus fazem um bom mapeamento das funções básicas executadas pela ferramenta, como é o caso das classes arquivo e editar. A possibilidade de embutir na interface signos familiares aos quais o usuário está habituado faz com ele tenha uma facilidade, mesmo que inicial, de identificar o que estes signos fazem.

A análise dos signos estáticos permitiu identificar os usuários do sistema, alunos do curso de computação. Isto foi possível devido aos signos da interface que remetem à compilação e simulação de um algoritmo, termos utilizados por alunos da área de computação. Os signos sobre simulação comunicam ao aluno a possibilidade dele interagir com o sistema simulando a execução de um algoritmo. Assim, pode-se entender que o objetivo educacional do Bipide é proporcionar ao aluno um recurso de apoio ao aprendizado para simulação e execução de algoritmos nas linguagens Portugol e Assembly.

3. **Análise dos Signos Dinâmicos:** esta análise possibilitou a identificação de uma nova classe de signos e também problemas relacionados ao conteúdo e à interação.

A nova classe identificada, apesar de surgir durante a interação (análise dos signos dinâmicos), faz parte dos signos metalinguísticos e refere-se às mensagens, conforme pode-se observar na parte inferior da figura 4.12. Nesta parte do sistema são apresentadas as mensagens de sucesso (ou erro) ocorridas durante a

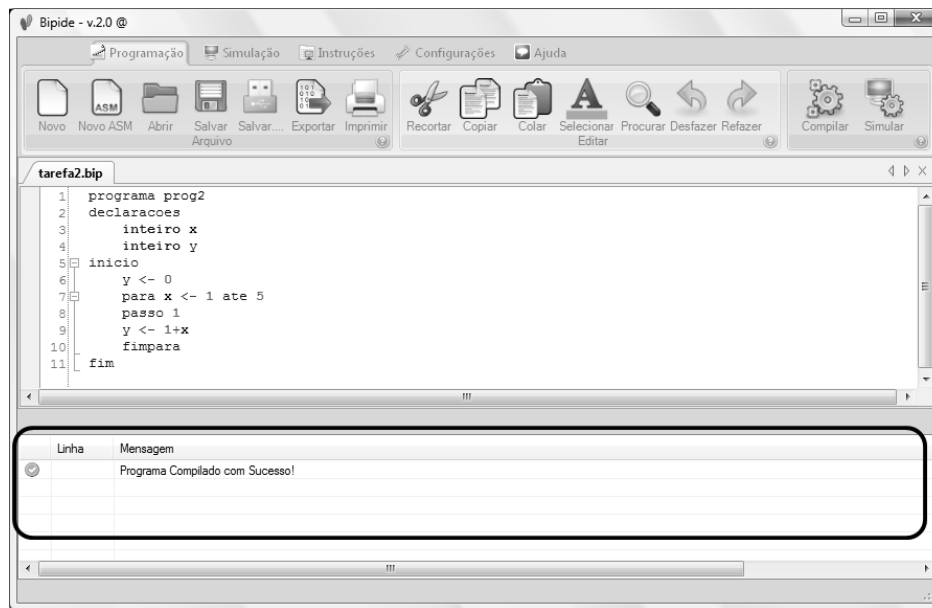


Figura 4.12. Bipide: Tela Principal

compilação do programa e mensagens de alerta, pertinentes ao programa. Nela também há a indicação da linha do programa onde se encontra o erro.

Com relação à interação, alguns problemas foram identificados. Ao clicar na opção *procurar*, o aluno é surpreendido, uma vez que, diferente de toda a interface, a tela aberta ao clicar neste signo está em inglês. Esta é uma inconsistência que possui baixo custo para a interação.

Outro problema encontrado refere-se à *Simulação* do código. Nesta tela, há muita informação e o aluno pode ficar perdido, sem saber em qual informação ele deve se concentrar (ver figura 4.3). O recurso de velocidade, mesmo na opção *muito lento*, ainda é rápido. Para o aluno que deseja entender como o processador interpreta o código e qual o caminho de dados percorrido pela instrução, seria útil inserir opções com pouca velocidade, para que ele possa acompanhar melhor o que está acontecendo.

Nesta análise, concluiu-se que o Bipide é um sistema voltado para alunos dos cursos de computação, que tem conhecimento sobre lógica de programação e desejam utilizar o sistema para executar códigos em Portugol e verificar o código *assembly* correspondente.

4. **Comparação das Três Metamensagens:** a metamensagem identificada nos passos anteriores (metalinguístico, estático e dinâmico) foi pouco divergente. Acredita-se que isso se deve ao fato da comunicação presente na ajuda do Bipide

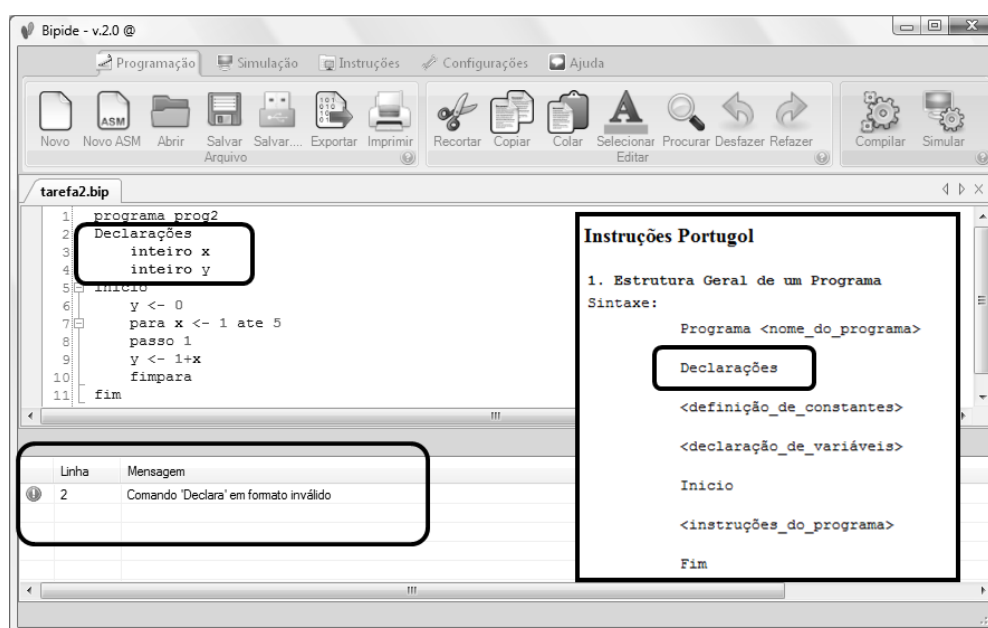


Figura 4.13. Bipide: Erro de Sintaxe

ser muito completa e os signos estáticos serem bem representados, além de fazerem uso de signos conhecidos pelos usuários de sistemas interativos.

No entanto, ao contrastar signos metalinguísticos e dinâmicos, constatou-se um problema de conteúdo referente à ajuda. Como o avaliador não tinha conhecimento sobre a linguagem Portugol, ele recorreu à ajuda para conhecer a sintaxe e escrever um pequeno código para testar as demais funcionalidades do Bipide. Entretanto, algumas instruções listadas na ajuda sobre a sintaxe da linguagem Portugol dificultaram o seu uso inicial. No exemplo da estrutura geral de um programa escrito em Portugol, a palavra *Declarações* estava escrito desta forma. Porém, ao escrever *Declarações* desta maneira na tela do editor, o programa dava erro e a mensagem não fornecia apoio suficiente para que o erro fosse corrigido, conforme pode-se verificar na figura 4.13. Após várias tentativas de correção, descobriu-se que a forma correta de declarar era *declaracoes*.

Do ponto de vista educacional, este problema pode ter consequências graves, caso o aluno não perceba que precisa retirar os caracteres especiais e a letra maiúscula para resolver o problema. Se o aluno for paciente e estiver interessado em usar o sistema, ele fará várias tentativas, mas caso ele não esteja disposto a dedicar um certo tempo para resolver este problema, ele pode desistir de usar a ferramenta.

Para reconstrução da metamensagem, o projetista do Bipide considerou que o aluno deve programar para ver o simulador funcionando, ou seja, ele tem a ex-

pectativa de que o usuário saiba programar. Apesar dos signos metalinguísticos serem bem completos, não há programas prontos para o aluno focar somente na simulação do funcionamento do computador. Uma vez que o aluno saiba programar, ele pode utilizar os recursos de simular para verificar quais as instruções *assembly* são necessárias para executar uma instrução na linguagem Portugol.

5. **Avaliação Global da Comunicabilidade:** considerando que o objetivo da inspeção era analisar a metacomunicação do projetista-usuário para comunicar o funcionamento do *hardware* ao executar um programa, verificou-se questões relacionadas a esta comunicação do Bipide.

Pela inspeção, constatou-se que o Bipide pode ser útil tanto para alunos de computação que estão aprendendo lógica de programação quanto por alunos que estão aprendendo sobre o funcionamento de um processador. Além disso, ele também pode ser utilizado pelo professor em sala de aula, para mostrar ao aluno, através do recurso de simulação, a forma como o processador interpreta um código.

De forma geral, a inspeção permitiu identificar com clareza os objetivos propostos pelo sistema e os recursos utilizados para comunicar este objetivo ao usuário. Assim, tem-se a seguinte metamensagem consolidada:

Caro usuário, o Bipide é um sistema gratuito que eu fiz para você, aluno de sistemas de apoio à aprendizagem e professor que faz uso de sistemas interativos em suas aulas. Eu entendi que você deseja aprender/ensinar ou está aprendendo/ensinando lógica de programação e o funcionamento de um processador. Para isso, eu criei o Bipide, um sistema que utiliza a linguagem Portugol, que permite a você criar pequenos programas em português estruturado e ainda simulá-los para entender como o processador interpreta este código criado. Para entender o processamento feito pelo processador, o Bipide permite a você executar o código passo a passo na opção de Simulação. Na Simulação, você acompanha os valores assumidos pelas variáveis do seu algoritmo, bem como a instrução assembly gerada por ele. Como é muita informação para você processar ao mesmo tempo, você pode fazer uso do recurso velocidade, que o permite reduzir ou acelerar a velocidade da simulação. A qualquer momento, você também pode pausar ou parar a execução da simulação. Caso você tenha utilizado uma sintaxe errada, a opção Mensagem o ajudará a identificar esses possíveis erros em seus algoritmos, bem como apresentar dicas de como modificá-lo para que

ele execute corretamente. Para acessar a ajuda, você pode fazer uso do menu ou ainda pressionar a tecla F1. Caso você precise, no menu Ajuda você encontrará informações relativas à ferramenta Bipide, à linguagem Portugol e ao conteúdo sobre arquitetura e organização de computadores, que poderão ser úteis no uso desta ferramenta.

4.6 Aplicabilidade do MIS no Domínio Educacional

Um importante resultado obtido com a aplicação do MIS nos estudos de caso refere-se à possibilidade deste método identificar, não somente problemas de interação, mas reconstrução de aspectos da metamensagem relacionados à aprendizagem, e.g. modelo de aprendizagem privilegiada.

Alguns problemas de interação identificados foram: (i) VCalc: ausência de mensagens de interação ou informativas e combinação de instruções em dois idiomas; (ii) ProfesSort: sequência de passos do algoritmo *heap sort*, mensagem incompleta para salvar o histórico; (iii) Bipide: simulação de código com muita informação. Com relação aos aspectos educacionais, destacam-se: o recurso de *zoom* do VCalc, que permite ver a curva virar reta localmente; os apoios à aprendizagem dos métodos oferecidos pelo ProfesSort e a sintaxe das instruções para Portugol no Bipide.

Na seção 3.2.1 argumentou-se que a aplicabilidade do MIS no domínio educacional é direta, ou seja, independe da tecnologia, por permitir a avaliação da comunicabilidade de qualquer sistema interativo. Assim, a hipótese é que o MIS faz a inspeção da comunicação sendo feita por meio da interface, sem restringi-la a critérios específicos de interação ou tecnologia. Por meio dos estudos de caso realizados, esta hipótese pôde ser confirmada para o contexto educacional. Como na aplicação do MIS o avaliador tem liberdade para explorar de forma aprofundada o sistema inspecionado, a visão da comunicabilidade neste método é mais ampla, pois o avaliador pode focar nas estratégias de comunicação utilizadas pelo projetista e assim reconstruir a metamensagem em sua totalidade, levando em consideração a questão de pesquisa que guia a inspeção [de Souza & Leitão, 2009].

Na aplicação dos estudos de caso foi possível identificar o objetivo educacional do sistema inspecionado e esta informação é importante para a comunicabilidade do sistema, uma vez que contribui para a reconstrução da metamensagem, na visão do especialista, à medida que responde à pergunta “*Para que serve o sistema?*”. A resposta a esta pergunta indica que os objetivos educacionais do sistema foram identificados pelo avaliador.

Outra observação relevante refere-se à importância do avaliador ter experiência no domínio educacional inspecionado. Dado que o avaliador possui conhecimento sobre o conteúdo abordado pelo sistema, ele aumenta as possibilidades de identificar problemas relacionados ao conteúdo, o que melhora também a qualidade da avaliação da comunicabilidade realizada. Na inspeção do VCalc, por exemplo, alguns problemas de conteúdo só foram identificados pelo avaliador que era especialista no domínio. Na inspeção dos demais sistemas, ProfesSort e Bipide, o avaliador com formação em computação e conhecimento mais aprofundado do conteúdo sendo trabalhado no sistema, encontrou tanto problemas de interação quanto de conteúdo. Para situações em que o avaliador não tem domínio do conteúdo abordado pelo sistema, ou ainda se o domínio for muito específico (e.g. sistemas médicos) pode limitar o tipo de análise que o avaliador é capaz de fazer. Neste caso, seria necessário contar com a contribuição ou consultoria de um especialista do domínio.

Estes resultados trazem importantes contribuições para as áreas de IHC, EngSem e Informática e Educação. Para a área de IHC contribui à medida que demonstra que um método de avaliação de sistemas interativos pode ser aplicado a qualquer domínio, sem a necessidade de fazer adaptações. Além disso, contribui para a valorização dos métodos de IHC baseados em teoria [Greenberg & Buxton, 2008; Carroll, 2003]. Para a EngSem, contribui para a avaliação do próprio MIS e o seu uso em diferentes domínios, gerando dados que apóiam a confirmação da hipótese levantada por de Souza et al. [2006]. Para a área de Informática e Educação, o método traz contribuições ao levantar questões sobre interação e aprendizagem.

Assim, a aplicação do MIS permitiu que se identificasse tanto rupturas de interação, conforme era previsto, quanto rupturas relacionadas ao conteúdo, que poderiam dificultar ou inviabilizar o uso dos sistemas educacionais avaliados.

Capítulo 5

Análise do Método de Avaliação de Comunicabilidade para o Domínio Educacional

O MAC [Prates et al., 2000; Prates & Barbosa, 2007; de Souza & Leitão, 2009] é um método que permite a avaliação da comunicabilidade do sistema a partir da interação usuários-sistema. Conforme visto na seção 3.3 o foco é na recepção da metamensagem pelo usuário, ou no caso de ambientes educacionais, pelo aluno.

Prates et al. [2004] fizeram a avaliação de um ambiente de apoio ao aprendizado denominado Ampliar e apontaram três importantes diferenças que surgiram na análise: (1) ambiguidade de sintomas, (2) novos sintomas e (3) alterações na interpretação da ruptura observada (ver seção 3.3.1, página 35). Assim, neste trabalho nosso objetivo era realizar novas avaliações em ambientes educacionais para aprofundar a investigação iniciada em Prates et al. [2004]. A ideia consistia em coletar indicadores a partir da avaliação de outros ambientes e contrastá-los com as considerações levantadas no trabalho de Prates et al. [2004]. Dado que alguns tratamentos diferenciados haviam sido identificados em Prates et al. [2004], outro aspecto a ser considerado era se seria necessário, ou possível, adaptar o MAC para o domínio educacional.

No trabalho feito por Prates et al. [2004], observou-se que algumas rupturas não eram relativas à interação, mas ao conteúdo sendo tratado pelo sistema educacional. Nesta direção, em um trabalho de iniciação científica executado em 2007, aplicou-se o MAC ao sistema ProfesSort (seção 4.3). Nesse trabalho percebeu-se que, ora as etiquetas se referenciavam ao conteúdo, ora à interação, e era possível fazer esta distinção no passo de etiquetagem. Assim, com o objetivo de investigar mais a fundo esta possibilidade, decidimos por durante o passo de etiquetagem fazer a distinção entre

rupturas de interação e conteúdo. Desta forma a cada ruptura identificada associou-se: *etiqueta*, *tempo*, *tipo* e *explicação*. A *etiqueta* é a expressão que categoriza a ruptura ocorrida; *tempo* é o momento da interação (por meio do vídeo) em que a ruptura ocorreu; *tipo* refere-se à ruptura de interação ou conteúdo e *explicação* é a descrição da ruptura.

A aplicação do MAC nos três sistemas avaliados, VCalc, ProfesSort e Bipide, foi realizada no laboratório de teste, localizado no DCC da UFMG e contém toda a infra-estrutura necessária para a realização deste tipo de avaliação. Este laboratório é composto por duas salas, separadas por um vidro espelhado. Uma destas salas é onde o usuário realiza o teste e a outra sala é de observação para os avaliadores. A sala de teste é composta por um computador, um microfone e uma câmera de vídeo. O computador é instrumentado com um *software* de captura de interação. A câmera pode ser utilizada para registrar a expressão facial e fala do usuário durante o teste. Na sala de observação há dois televisores, sendo que um deles é ligado à câmera de vídeo e o outro, ao computador da sala de teste. Através deste televisor, o avaliador observa toda a interação do usuário com o sistema inspecionado¹.

A aplicação técnica do MAC nos sistemas avaliados permitiu a identificação de considerações específicas para o domínio educacional, podendo auxiliar não apenas na identificação de aspectos relacionados à interação, mas também à aprendizagem. Nas seções seguintes os resultados obtidos com a aplicação do MAC nos sistemas avaliados serão detalhados.

5.1 Aplicação do MAC no VCalc

A seguir são detalhados os passos para avaliação do VCalc utilizando o MAC.

5.1.1 Preparação e Execução do Teste

Neste passo foi definido o *objetivo da avaliação*, analisar a metacomunicação do projetista para o usuário no sistema de apoio ao aprendizado, VCalc. Para avaliação do VCalc com o MAC, utilizou-se os resultados obtidos na análise do MIS para elaborar o material a ser utilizado no MAC. Assim como no MIS, optou-se por avaliar toda a interface do VCalc, por ser um aplicativo simples e com poucas funcionalidades. Ainda nesta etapa elaborou-se o termo de consentimento, o roteiro das entrevistas pré

¹O MAC não requer a gravação de vídeo dos usuários, apenas a de sua interação. Então as câmeras de vídeo foram utilizadas durante o teste para observação dos usuários, mas seu conteúdo não foi gravado.

e pós-teste, as tarefas, o formulário de acompanhamento do teste (ver anexo B) e o seguinte cenário:

Você é aluno do curso de Cálculo I e atualmente estuda os conceitos de derivada. Para facilitar o entendimento de alguns conceitos, o professor resolveu dar aula utilizando o computador, como forma de apoiar a sua aprendizagem. O professor indicou o endereço eletrônico e, ao abri-lo, foi executado o aplicativo VCalc. Utilizando este applet o professor solicitou que você construísse o gráfico de uma dada equação, com o objetivo de identificar a taxa de variação e o comportamento dessa função no gráfico.

A avaliação do VCalc foi conduzida por dois alunos da pós-graduação do DCC/UFMG, como parte do trabalho da disciplina de IHC oferecida para a pós-graduação, no segundo semestre de 2007, e era a primeira vez que esses alunos aplicavam o MAC. Um destes avaliadores tem formação em Matemática Computacional e havia trabalhado no desenvolvimento de sistemas de apoio ao aprendizado voltado para a área matemática.

Participaram da avaliação 6 (seis) alunos que cursavam a disciplina de Cálculo I na época da avaliação. Os alunos tinham idade entre 19 (dezenove) e 23 (vinte e três) anos, sendo quatro do sexo masculino e dois do sexo feminino. Os alunos eram dos cursos de: Engenharia Civil (2); Matemática Computacional (1); Sistema de Informação (2) e Medicina (1)².

Como os alunos cursavam a disciplina de Cálculo I e estavam na etapa de consolidação de conhecimentos sobre derivada, eles se mostraram como um perfil representativo de usuários-alvo para o qual o VCalc foi desenvolvido. Os alunos foram recrutados voluntariamente e demonstraram interesse em participar da avaliação de comunicabilidade do VCalc.

Cada participante, ao chegar ao laboratório de teste, era recepcionado pelos avaliadores e recebia orientações gerais sobre o teste. Nestas orientações, o participante era informado sobre a observação e registro de toda a interação com o sistema de captura de tela. Em seguida, ele era convidado a ler e assinar o termo de consentimento e o questionário pré teste. Além disso, uma breve descrição sobre o sistema VCalc lhe era apresentada e, após o aluno confirmar o entendimento sobre a ferramenta, ele recebia o cenário de inspeção e a tarefa que deveria resolver utilizando o VCalc.

²Cálculo I não é disciplina obrigatória do curso de Medicina, mas o aluno que se interessar pode fazer como disciplina eletiva (que era o caso).

No caso do VCalc, assim como nos demais testes realizados nos outros sistemas, o filme da interação foi complementado com anotações dos observadores durante o teste, e uma entrevista pós-teste com cada aluno foi realizada.

5.1.2 Análise dos Dados

De posse dos vídeos da interação, os avaliadores realizaram a etiquetagem à luz do conjunto de expressões (etiquetas) de comunicabilidade, conforme apresentado na seção 3.3. A etiquetagem completa dos vídeos gerados pelos participantes no VCalc encontra-se no anexo C.

Vale ressaltar que das seis avaliações realizadas, foi necessário desconsiderar uma delas, pois o aplicativo VCalc travou enquanto o participante realizava a tarefa. Este travamento fez com o aluno reiniciasse a tarefa. No entanto, como já havia interagido com o sistema as rupturas observadas na sua primeira interação não foram repetidas pelo aluno. Assim, somente cinco avaliações foram consideradas para análise dos dados.

Ao iniciar a análise, os avaliadores se depararam com algumas situações de ruptura de interação que deveriam ter um tratamento diferenciado do proposto no método original. A seguir, são apresentados os resultados da análise observada na aplicação do MAC no sistema VCalc.

Durante o teste de dois participantes (P2 e P4) verificou-se que, em alguns momentos, esses alunos vagavam com o cursor do mouse sobre a tela, clicando nos botões de forma aleatória. Conforme as etiquetas, esta ruptura normalmente seria classificada como um “E agora?”, indicando que o usuário não sabia como prosseguir com a utilização do aplicativo. Porém, o avaliador com formação em matemática desconfiou que a ruptura de fato representava uma dificuldade que os alunos estavam tendo com o conteúdo abordado pelo VCalc e este argumento, no caso do P2, foi confirmado pelo próprio participante. Na entrevista pós-teste, ele foi questionado sobre essa atitude (vagar com o cursor do mouse pela interface do sistema), e o P2 disse que estava perdido e não sabia como prosseguir em relação ao conteúdo abordado pelo sistema. No caso do P4, ele não foi questionado sobre essa questão na entrevista.

Entretanto, neste caso, a etiqueta “E agora?” refere-se à comunicação sobre derivada, conteúdo abordado pelo VCalc. Assim, estas rupturas do P2 e P4 foram classificadas como “E agora?” de conteúdo pelos avaliadores, isto é, embora os sintomas tenham sido os mesmos, foi feita uma qualificação desta ruptura como sendo relativa ao conteúdo. É importante ressaltar que esta diferenciação entre ruptura de conteúdo e interação levou em consideração, não apenas as ações representativas do sintoma da etiqueta “E Agora?”, mas também um conjunto maior de ações realizadas

pelos usuários, que incluem também outras rupturas e que permitiram a contextualização da etiqueta. Desta forma, pode-se considerar que a qualificação da etiqueta como sendo de conteúdo de fato foi associada no passo da etiquetagem.

O P4, após várias tentativas, desiste de concluir as tarefas solicitadas, apesar de responder no questionário pré-teste que se lembrava dos conceitos que estava aprendendo na disciplina de Cálculo I. E esta ruptura foi então classificada como “*Desisto*”. No MAC, esta ruptura é considerada falha completa, uma vez que o usuário não conseguiu interagir com a interface do sistema para resolver uma tarefa a que a ferramenta se propõe.

No caso do domínio educacional, o “*Desisto*”, embora seja uma falha, pode ter consequências positivas se relacionada ao conteúdo. Afinal, pode indicar ao aluno que ele não tem o domínio do conteúdo esperado ou que acreditava possuir. O próprio P4, durante a interação, disse “*Eu não estou entendendo nada, estou perdido.*”. Então, neste caso, a interação foi útil por mostrar ao aluno que ele precisa melhorar seu conhecimento sobre o assunto abordado pelo VCalc. Assim, pode-se dizer que o VCalc permitiu ao aluno identificar o conhecimento esperado que ele tivesse relativo à taxa de variação de função.

O P3 terminou a tarefa de forma errada, apesar de acreditar que o resultado da derivada apresentado estava correto. De acordo com o MAC, esta ruptura é associada à etiqueta “*Para mim, está bom*”. Novamente, para o MAC, esta falha é considerada grave, pois o usuário não percebe que está equivocado. Neste caso, esta ruptura não é diferente da análise original do MAC. No entanto, no domínio educacional ela pode ter consequências graves ao aprendizado do aluno, pois pode levá-lo a crer que domina o conteúdo abordado pelo sistema. Pelo fato do VCalc ser um aplicativo voltado para a visualização gráfica, ele não avalia a atividade do aluno em relação à sua corretude. Com isso, o aluno pode fazer um uso errôneo do sistema, contribuindo até para um falso entendimento sobre o conteúdo abordado pelo sistema.

Finalizada a aplicação do MAC no aplicativo VCalc, tem-se o seguinte perfil semiótico:

Caro usuário, o VCalc é uma ferramenta que eu criei para você, aluno da disciplina de Cálculo I e que precisa consolidar sua aprendizagem sobre taxa de variação de função. O VCalc, além de ajudá-lo na compreensão de taxa de variação de função por meio de recursos gráficos, também lhe permitirá investigar sobre o comportamento desta função em um intervalo definido por você. Assim você poderá analisar graficamente o comportamento de uma função estudada em sala de aula. Para que você faça um

bom uso deste sistema, é necessário que você domine o conceito de derivada de função, bem como taxa de variação. Caso você tenha dúvidas sobre o conceito de derivada, é recomendado voltar ao material de aula indicado pelo seu professor e só então você poderá retornar ao VCalc, pois ele não oferece este tipo de auxílio. Com o VCalc você poderá inserir funções vistas em sala de aula e determinar os eixos, x e y , do gráfico onde a função será apresentada. É recomendado que você confira as funções inseridas para análise, para que não gere visualizações errôneas, pois o sistema não será capaz de lhe auxiliar nesta identificação. Eu também disponibilizei a ferramenta zoom, que o permite investigar cautelosamente o comportamento do gráfico à medida que a derivada se aproxima do ponto 0. Você pode ainda aumentar ou diminuir o delta x e visualizar no gráfico o que ocorre com a reta secante. Para movimentar o ponto vermelho você pode fazer uso do botão que se encontra na parte inferior da tela. Para parar este botão, basta clicar na parte central, senão uma animação é disparada continuamente.

A etiquetagem e interpretação dos dados apontou para a necessidade de diferenciar problemas de interação e conteúdo. Além disso, a interpretação evidenciou a necessidade da presença de um especialista no domínio como forma de reduzir as ambiguidades que possam surgir na avaliação do domínio educacional, conforme pôde-se observar nos problemas identificados do P2.

5.2 Aplicação do MAC no ProfesSort

Nas seções seguintes é feito o detalhamento da inspeção do MAC no sistema de apoio ao aprendizado ProfesSort.

5.2.1 Preparação e Execução do Teste

Neste passo definiu-se o *objetivo da avaliação*, analisar a metacomunicação do projetista para o usuário do sistema de apoio à aprendizagem de métodos de ordenação, ProfesSort. Participara da aplicação do MAC duas alunas de iniciação científica (IC), graduandas em Sistemas de Informação. Foi realizada uma inspeção informal do ProfesSort para gerar o cenário da avaliação. Após a avaliação informal do sistema, optou-se por avaliar somente o método de ordenação *quick sort*, considerado de difícil entendimento pelos alunos da disciplina de AEDS II. Nesta etapa também foram elaborados (ver

anexo D): roteiro de observação do teste, roteiro das entrevistas pré e pós-teste, termo de consentimento, tarefas, formulário de acompanhamento do teste e este cenário:

Você é aluno do curso de AEDs II e está estudando o método de ordenação Quicksort, que se caracteriza por ser um método rápido e eficiente. Seu professor lhe apresentou o programa ProfesSort, a fim de auxiliá-lo no aprendizado do Quicksort. Como o professor falou que você poderia usar o ProfesSort na resolução dos exercícios, você resolve utilizá-lo para garantir que não vai cometer nenhum erro e também para tirar eventuais dúvidas que apareçam durante o exercício. No primeiro exercício, o professor forneceu um vetor com 7 (sete) posições para ser ordenado, e solicitou que você mostrasse, passo a passo, como ordená-lo usando o quicksort.

As tarefas da avaliação consistiam no aluno ordenar dois vetores distintos utilizando o método *quick sort* no sistema ProfesSort (ver anexo D).

Com o material para aplicação do teste pronto, realizou-se o teste piloto, como forma de apreciar a qualidade do material gerado. O teste piloto consistiu em executar a avaliação planejada, com um aluno de perfil similar aos demais participantes. O objetivo deste teste era verificar se o participante era capaz de entender o material gerado, se as tarefas estavam condizentes com o objetivo proposto da avaliação e se o tempo gasto na execução do teste estava de acordo o previsto [Prates & Barbosa, 2007]. É válido ressaltar que os dados coletados no teste piloto foram desconsiderados para análise, uma vez que eles foram usados apenas para avaliação do material gerado.

A coleta dos dados seguiu os mesmos procedimentos apresentados na seção 5.1.1. Toda a interação foi gravada por um sistema de captura de tela e a entrevista gravada em áudio, para análise posterior. Participaram da avaliação 4 (quatro) alunos, todos do sexo masculino com idade entre 18 (dezoito) e 22 (vinte e dois) anos. Três destes alunos cursavam o curso de Engenharia de Controle e Automação e um cursava Engenharia Elétrica. Todos estavam cursando a disciplina da AEDs II na época da avaliação.

5.2.2 Análise dos Dados

Como a aplicação e análise dos testes com o ProfesSort havia sido feita por duas alunas de IC, a avaliadora, aluna de pós graduação, fez uma nova etiquetagem dos vídeos gerados, com a finalidade de comparar os resultados encontrados na etiquetagem feita anteriormente. Munida dos vídeos da interação e das anotações feitas, a avaliadora realizou a etiquetagem baseada nas etiquetas de comunicabilidade, classificadas como *etiqueta*, *tempo*, *tipo* e *explicação*. Ainda nesta etapa, a avaliadora interagiu com a

outra avaliadora que participou do teste, para resolver pontos que apenas pela gravação estavam ambíguos. A etiquetagem completa dos vídeos gerados pelos participantes no teste do ProfesSort encontra-se no anexo E.

Para a análise dos dados foi necessário descartar o teste de um dos participantes, pois a opção de gravar a interação do usuário com o mouse não foi selecionada, portanto impossibilitou a análise do vídeo. Assim, neste estudo de caso, foram considerados três testes dos participantes. A seguir são detalhadas as interpretações desta etiquetagem.

1. Como o ProfesSort conta com o uso de *scaffolds*, que auxiliam o aluno na execução passo a passo do algoritmo de ordenação, sempre que tinha uma dúvida, o aluno recorria aos apoios oferecidos por ele. Todos os participantes da avaliação do ProfesSort, por exemplo, utilizaram com frequência os botões disponibilizados na área *Auxílio à execução*. Em termos de interação, os sintomas referem-se à etiqueta *Socorro!*, ou seja, é um pedido explícito à metacomunicação. Porém, neste caso, é um pedido para entrar em metacomunicação, só que não com o projetista, mas com o professor. Assim, a dúvida não é em relação à interação, mas claramente em relação ao conteúdo.

Outra situação observada refere-se às inúmeras vezes que o participante 4 (P4) recorreu às opções de ajuda fornecidas pelo ProfesSort, principalmente *Auxílio à execução* e *Menu de perguntas*. Como pode-se observar na figura 5.1, P4 recorreu ao menu de perguntas para saber *o que fazer quando não havia mais elementos a trocar e i e j ainda não haviam se cruzado*. Esta ruptura foi categorizada como *Socorro!*, referente ao método de ordenação, e não à interação. Esta ação de selecionar elementos para troca já havia sido executada anteriormente por este participante, então acreditou-se que ele sabia como interagir. Porém, ele não sabia como prosseguir para realizar a troca em questão. Além desta ruptura, percebeu-se que este participante recorreu constantemente às opções de ajuda oferecidas, principalmente no momento de fazer as trocas dos elementos *i* e *j*.

Quando o aluno clicava na opção *Mostrar próximo passo* considerou-se como um pedido de ajuda (*Socorro!*). Neste caso, a diferença de como o *Socorro!* é interpretado no método original poderia ter uma exploração dos *scaffolds* e tentativas (e.g. estou tentando entender, mas não consigo *versus* nem tentei e quero ver o que fazer), a fim de verificar, na etapa da interpretação, a utilidade do sistema para o aprendizado. A interpretação desta etiqueta teria que ser feita em relação à sequência de rupturas geradas pelo aluno, podendo-se apontar para o esforço e/ou interesse do aluno de aprender.

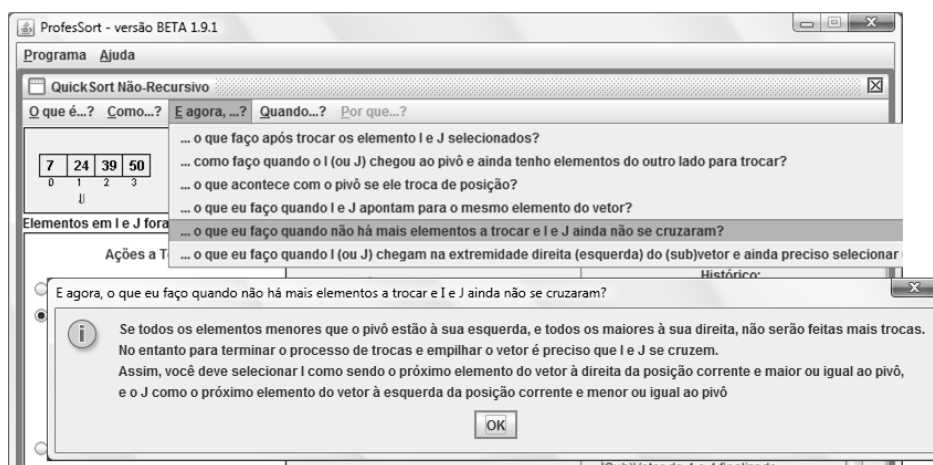


Figura 5.1. ProfesSort: etiqueta Socorro, referente ao conteúdo!

A ruptura *Socorro!* é uma falha temporária e indica que o usuário não conseguiu realizar a tarefa por meio da exploração da interface. Do ponto de vista da comunicabilidade, isto é um problema, pois o usuário não conseguiu entender a comunicação pretendida pelo projetista. No entanto, para o domínio educacional esta ruptura pode ser considerada como positiva, pois o aluno pode identificar explicitamente os pontos que ainda não estão claros para ele e tirar suas dúvidas. Além disso, pode gerar indicadores sobre o aprendizado para o avaliador ou professor (caso este esteja envolvido na avaliação ou receba seus resultados) as dúvidas daquele aluno, ou olhando globalmente - as dúvidas mais comuns dos alunos sobre aquele conteúdo em específico.

No passo de interpretação do MAC pode-se identificar tanto (i) pontos em que os alunos têm mais dúvidas, a partir das rupturas e pedidos de ajuda que surgem, quanto (ii) analisar o impacto do sistema no aprendizado, por meio da comparação da execução de várias tarefas no sistema, observando se caem (ou não) as rupturas relativas ao conteúdo no decorrer do uso do sistema.

Um outro ponto importante a ser discutido sobre o uso do MAC no domínio educacional é a questão da inserção (ou não) de *scaffolds* no sistema. Os sistemas que utilizam *scaffolds*, como ProfesSort e Ampliar, permitem identificar com mais facilidade as dúvidas relativas ao conteúdo por parte do aluno, pois neste tipo de sistema, ele explicita suas dúvidas. No caso de sistemas que não possuem *scaffolds*, caberia ao avaliador, como aconteceu na avaliação do VCalc, identificar estas dúvidas (e.g. Para mim está bom... e Desisto, Epa!) a partir da realização da tarefa pelo aluno.

2. Em outra situação, o participante 2 (P2) ficou vagando por alguns instantes com o cursor do mouse sobre duas opções do recurso *Ações a tomar*, trocar elementos selecionados e finalizar trocas. Estes sintomas caracterizam a ruptura como sendo um *E agora?*. No entanto, como o participante vagava com o cursor entre duas opções e já havia interagido com estes elementos de interface, as avaliadoras identificaram que ele estava em dúvida sobre qual seria o próximo passo do método de ordenação (e não da interação). O P2 então recorre à opção *Mostrar próximo passo*, confirmando a interpretação das avaliadoras.

No VCalc, conforme discutido na seção 5.1.2, também se qualificou uma ruptura como *E agora?* relativa ao conteúdo. No entanto, esta qualificação sobre o conteúdo só pôde ser confirmada na interpretação. Por isso, esta situação é diferente do que houve no VCalc, pois no caso do ProfesSort, na etiquetagem já foi possível fazer esta qualificação, considerando-se que o aluno havia interagido com o signo anteriormente.

Em outra situação, o P2 parece ter ficado perdido, sem saber qual próximo passo deveria tomar para prosseguir com a ordenação. Ele ficou vagando com o cursor do mouse sobre a tela. Quando esta situação foi percebida, ele já havia interagido com os signos presentes na tela, então P2 sabia interagir, mas ficou na dúvida sobre qual opção selecionar (escolher elemento i ou j). Esta ruptura então foi classificada como um *E agora?* sobre o conteúdo, ou seja, ele sabia como interagir, mas não sabia qual elemento selecionar. Neste caso, na etiquetagem foi possível identificar essa ruptura, mesmo sem a explicitação da dúvida através dos *scaffolds* pelo aluno.

O fato de o ProfesSort oferecer apoio à aprendizagem por meio dos *scaffolds*, facilita a identificação de problemas de conteúdo, pois uma vez que o aluno usa um determinado signo da interface, ele sabe como o signo funciona. Então ele não teria, a princípio, problemas para interagir com este signo. Se ele fica vagando com o cursor do mouse sobre dois signos (e.g. troca de i e j), indica que ele está com dificuldade em identificar qual o próximo passo a ser executado, o que pode ser caracterizado como uma ruptura de conteúdo e não de interação.

3. Numa análise geral do P4, verificou-se que, somente na tarefa 1, este aluno teve um total de 27 (vinte e sete) rupturas, sendo 23 (vinte e três) rupturas relativas ao método *quick sort*, ou seja, rupturas de conteúdo. Destas 23, 7 rupturas foram de interação com o *Menu de perguntas*, 8 rupturas ocorreram com a opção *Mostrar próximo passo* e as demais foram rupturas variadas, mas relativas ao

conteúdo. Na tarefa 2, houve uma melhora pouco significativa, pois ele teve um total de 22 (vinte e duas) rupturas, sendo 20 relativas a problemas de conteúdo e 2 relativas à interação. Com base na interação do aluno apenas, poderia se concluir que talvez o sistema não tenha auxiliado muito seu aprendizado. No entanto, na entrevista pós-teste, P4 disse que não se recordava das definições das variáveis i e j e que gostou muito da ajuda oferecida pelo ProfesSort. Com base nisso, pode-se concluir que talvez o aluno não tivesse o conhecimento mínimo esperado sobre o método para que se pudesse consolidá-lo. O alto número de rupturas relacionadas a solicitação explícita de ajuda sobre o conteúdo podem indicar uma tentativa do aluno de aprender o método (ou as partes que não estavam claras) através do sistema. Assim, esta interação do aluno pode ser um indicador positivo sobre o uso do sistema no aprendizado.

4. Nas situações em que o aluno clicava na opção *Executar próximo passo* pode-se fazer duas observações distintas. Uma seria quando o aluno não estiver interessado em aprender (e.g. não estou conseguindo fazer, então faça logo por mim) ou pode estar relacionada ao fato do aluno já saber fazer aquele passo do método (e.g. já sei fazer, então vamos passar logo para frente).

O signo *Executar próximo passo* é uma opção apresentada claramente na tela do ProfesSort e é uma comunicação intencional do projetista e, em termos de interação, usar este recurso não é uma ruptura na comunicação. No entanto, se o aluno utiliza o recurso como forma de evitar as atividades de aprendizado propostas, neste caso seria uma ruptura em relação à parte educacional (ou seja, de conteúdo). O P4 teve uma grande quantidade de rupturas de conteúdo. Por exemplo, sua interação teve vários *Epas!* que apontavam dúvidas sobre qual elemento do vetor era necessário selecionar para realizar a troca. Além disso, ele usou a opção *Executar próximo passo* nas duas tarefas com frequência. Assim, pode-se concluir que o aluno estava tendo dificuldades em fazer a ordenação e recorreu ao elemento da interface para resolver por ele. Neste caso, esta ruptura seria etiquetada com um “*Não, obrigado.*” relativo ao conteúdo. O aluno entendeu as atividades propostas para o seu aprendizado (em que ele deveria indicar os passos e executá-los), no entanto preferiu não engajar no aprendizado e, para resolver a tarefa, utilizou outro recurso disponibilizado que não envolve aprendizado. Vale ressaltar que, normalmente, isto poderia não ser representativo de um baixo interesse do aluno (e.g. “Não estou interessado, faça por mim!”), mas em algumas situações, poderia representar a tentativa de aprender pelo sistema (e.g. “Não sei fazer, me mostre como é.”). No entanto, como esta distinção é

muito difícil de fazer e o sistema não tem o objetivo de ensinar, consideramos nestas situações o *Executar próximo passo* uma ruptura de conteúdo.

Já o P5, ao realizar a segunda tarefa, recorreu três vezes à opção *Executar próximo passo* quando estava prestes a concluir a ordenação do vetor. Como na primeira tarefa ele não usou este recurso nenhuma vez, acredita-se que ele havia entendido o funcionamento do método e não estava disposto a fazer a troca dos elementos do vetor passo a passo. Neste caso, pode-se concluir que ele já havia entendido como o método funcionava, e queria finalizar logo as trocas para (talvez) focar em outra etapa do método.

A situação do P4 é uma ruptura, enquanto do P5 não. Esta situação é diferente da etiquetagem do MAC original, pois nela não temos casos onde um sintoma é observado e ora é ruptura, ora não. Só é possível fazer esta distinção durante a interpretação.

5. Outro ponto que surgiu na análise refere-se à possibilidade de ter marcadores sobre o aprendizado do aluno. Na etapa de interpretação do MAC, um dos aspectos que se propõe que o avaliador leve em consideração é se existem padrões de sequências de rupturas. No caso de ambientes educacionais foi relevante observar alguns padrões de mais alto nível que poderiam ser indicadores interessantes de aprendizado. Em particular, nos referimos aos padrões em que o aluno comete a ruptura de conteúdo, uma ou mais vezes, para executar determinado passo e, a partir de um dado momento, consegue completá-lo sem rupturas. Este é claramente um indicador de que o sistema apoiou o aluno no seu aprendizado. O P5, por exemplo, na tarefa 1, no momento de finalizar as trocas dos elementos i e j, tinha dificuldades e várias vezes recorria aos apoios oferecidos pelo ProfesSort. Já na tarefa 2, ele não errou no momento desta finalização. Na entrevista pós-teste, este aluno confirmou a observação feita pelos avaliadores: “...com a utilização do ProfesSort ficou mais claro para mim a forma como o algoritmo realmente funciona.”.

Ter um marcador positivo para o aprendizado é relevante, uma vez que permite ao aluno explicitar que não tem conhecimento sobre determinado ponto abordado pelo sistema. Os marcadores positivos podem oferecer *feedback* sobre o sistema educacional e suas possibilidades de apoio. Para o aprendizado, isto seria um recurso importante, pois permitiria ao avaliador (ou professor) identificar dúvidas específicas dos alunos e assim reforçar este conteúdo em sala de aula.

5.3 Aplicação do MAC no Bipide

Como a aplicação do MAC no Bipide ocorreu após a avaliação do VCalc e do ProfesSort, realizou-se este estudo com a finalidade de verificar a aplicabilidade do MAC em um sistema de visualização e se surgiria alguma nova situação específica para este tipo de sistema. A seguir é feito o detalhamento da inspeção do MAC no sistema de apoio ao aprendizado Bipide.

5.3.1 Preparação e Execução do Teste

A preparação consistiu em definir o *objetivo da avaliação*, analisar a metacomunicação do projetista para o usuário para comunicar o funcionamento do *hardware* ao executar um programa. Para a definição das partes do sistema a serem inspecionadas, considerou-se simulação de processadores e a parte que envolve os conceitos de lógica programação, isto é, as mesmas partes selecionadas para a aplicação do MIS (ver seção 4.5.3).

Nesta etapa também elaborou-se todo o material para aplicação do MAC (roteiros - de observação do teste e das entrevistas pré e pós-teste, termo de consentimento, tarefas, formulário de acompanhamento do teste, uma tabela com os principais comandos do Bipide), conforme pode-se observar no anexo F. O seguinte cenário também foi gerado:

Você é aluno da disciplina de Arquitetura e Organização de Computadores (AOC). Você tem conhecimento em lógica de programação e está estudando sobre o funcionamento do hardware ao executar um programa. Seu professor lhe apresentou o programa Bipide, a fim de auxiliá-lo no aprendizado do processo que o hardware faz para executar um programa. Como o professor falou que você poderia usar o Bipide para escrever códigos na linguagem Portugol e simular o funcionamento do hardware ao executar este código, você resolve usá-lo para tirar eventuais dúvidas que apareçam e assim consolidar seu aprendizado.

O Bipide é um programa que tem como foco a visualização de um código na linguagem Portugol. Conforme apresentado na seção 5.3.1, a primeira tarefa consistia no aluno criar um programa usando a linguagem Portugol. Como é desejável que as tarefas sejam independentes, para situações em que o aluno não consegue (ou desiste de) realizar uma tarefa, ele possa realizar a outra. Assim, a segunda tarefa do Bipide

consistia no aluno simular um código em Portugol e verificar como o *hardware* processa este código. Este código foi dado na tarefa (ver anexo F) e o aluno precisava apenas digitá-lo, justamente pensando na situação em que o aluno tivesse dificuldade de realizar a primeira tarefa.

Para avaliar o material gerado para aplicação dos testes no sistema Bipide, realizou-se um teste piloto. O participante convidado, assim como os demais usuários, era aluno do curso de Sistemas de Informação do DCC/UFMG e na época cursava a disciplina AOC. Vale ressaltar que este participante foi convidado por ter um perfil similar aos demais alunos que fariam o teste e também por ele conhecer o MAC e já ter participado da aplicação deste método em outras avaliações do domínio educacional. Com isso, após a execução do teste ela discutiu com a avaliadora o material, contribuindo para que mudanças significativas fossem feitas no material gerado.

Dentre as principais mudanças ocorridas, destacam-se a elaboração de questões específicas sobre a parte de simulação de algoritmo do Bipide. Essas questões foram inseridas como parte da entrevista pós-teste para complementar a etiquetagem e podem ser vistas no anexo F. Como a parte de simulação de algoritmo no Bipide, alvo desta avaliação, é basicamente uma visualização e não requer interação do aluno com o sistema (exceto para alteração da velocidade da simulação) não se esperava que surtissem rupturas nesta parte. Isto se deve ao fato dos conceitos de AOC abordados pelo sistema serem apenas visuais, ou seja, uma simulação apresentava como o *hardware* se comportava para armazenar as instruções do código na linguagem Portugol. Ainda assim, queríamos avaliar se questões sobre o aprendizado (i.e. rupturas sobre o conteúdo) aconteciam, mesmo que o aluno não tivesse a oportunidade de expressá-la através do sistema. Como veremos na análise da avaliação feita, de fato as questões atingiram seu objetivo.

Finalizado o material de aplicação do MAC, realizou-se a avaliação do sistema Bipide. A avaliação desta ferramenta foi conduzida, no segundo semestre de 2009, pela autora que aplicava o MAC pela terceira vez. Participaram da avaliação 7 (alunos) alunos, sendo 6 (seis) do sexo masculino e 1 (um) do sexo feminino, na faixa etária de 20 a 29 anos. Dos 6 alunos, somente um deles não tinha experiência com sistemas de apoio ao aprendizado.

Dos sete participantes, dois tiveram o teste invalidado para a análise. O teste do P1 foi descartado pelo fato dele, por problema de horário, ter feito as tarefas em dias distintos, o que poderia impactar na realização da segunda tarefa. O teste do P7 foi desconsiderado, pois ele, no momento de salvar o vídeo da segunda tarefa, sobrescreveu o vídeo da tarefa 1. Como ele já havia interagido com o sistema, não lhe foi permitido fazer a tarefa 1 novamente.

5.3.2 Análise dos Dados

Finalizada a execução dos testes, o avaliador fez a etiquetagem dos vídeos dos cinco participantes. A etiquetagem completa dos vídeos gerados pelos participantes no teste do Bipide encontra-se no anexo G. A avaliação do Bipide apontou uma situação nova, o que enriqueceu a nossa discussão sobre a aplicabilidade do MAC para o domínio educacional. A seguir, são listados os problemas identificados nesta análise.

A etiquetagem do MAC no sistema Bipide permitiu distinguir rupturas relacionadas à interação e ao conteúdo. No entanto, ao iniciar a etiquetagem, a ocorrência de uma sequência de rupturas na primeira tarefa, que consistia em escrever um código na linguagem Portugal, chamou a atenção da avaliadora. Dos cinco participantes, quatro tiveram um número considerável de rupturas *Epa!*, *Porque não funciona?* e *Socorro!* (em média 13 rupturas por participante). Esta sequência de rupturas se deve a um erro de sintaxe presente na ajuda do Bipide, anteriormente observado com a aplicação do MIS (ver seção 4.5.3).

As linguagens de programação comumente ensinadas nas disciplinas de programação do DCC/UFMG são Pascal e C [Ziviani, 2004]. Por isso, os alunos que participaram da avaliação não tinham conhecimento da linguagem Portugal. No entanto, para realizar a primeira tarefa, eles precisavam conhecer a sintaxe desta linguagem. Com a finalidade de auxiliar os alunos na escrita deste código, foi elaborada uma tabela com os principais comandos e sintaxe da linguagem Portugal. Além disso, na ajuda do Bipide há informações sobre a sintaxe desta linguagem.

Todos os participantes, ao iniciarem a primeira tarefa, recorriam imediatamente à ajuda, para saber como é a sintaxe do Portugal. Esta consulta ao signos metalinguísticos foi etiquetada como *Socorro!* para a interação, pois os alunos tinham conhecimento da lógica de programação, mas não sabiam como escrever o código usando Portugal. Apoiados pelo *help* eles criavam o código solicitado corretamente, do ponto de vista da lógica de programação. Entretanto, ao simular este código, o Bipide apresentava um erro de sintaxe, conforme apresentado na figura 4.13. Após esta mensagem de erro, o aluno tentava corrigir o código e com isso, gerava novas rupturas de interação do tipo *Epa!*, seguidas de *Porque não funciona?*.

A ruptura etiquetada como *Socorro!* revelou um aspecto importante a ser considerado em relação à comunicabilidade de um sistema de apoio ao aprendizado: mensagens de erro pouco informativas. A carência de mensagens informativas de qualidade neste domínio podem contribuir para um uso ineficiente dos sistemas educacionais ou mesmo impedir o seu uso pelo aluno. Esta situação foi observada na avaliação do Bipide, onde dois participantes, após várias tentativas sem sucesso, desistiram de con-

cluir a primeira tarefa. Na entrevista pós-teste, estes alunos foram questionados sobre o apoio oferecido pela ajuda e pelas mensagens. Ambos disseram que estes recursos não os apoiaram conforme esperado. Um deles disse: ...”*A ajuda e as mensagens não eram tão específicas, poderiam ser mais elaboradas, com exemplo de código pronto. Pensando bem, a ajuda me confundiu.*”. Assim, fornecer signos metalinguísticos de qualidade podem auxiliar a aprendizagem por parte do aluno, além de favorecer uma boa comunicabilidade do projetista com o aluno.

Nas duas situações em que os alunos desistiram de concluir a tarefa eles tiveram dificuldade de entender a metamensagem do projetista (ou educador) sendo comunicada através da interface do sistema. Neste caso, ficou claro que eles não entenderam como poderiam interagir com o sistema para escrever o código na linguagem Portugol. Do ponto de vista da EngSem isso é um problema grave, pois houve uma falha na comunicação do projetista para o aluno.

Um problema de interação identificado refere-se à parte de simulação. Nesta tela há muita informação e o aluno tem dificuldades de acompanhar plenamente as informações apresentadas. Todos os alunos, ao serem questionados sobre esta opção, relataram esta dificuldade. Para auxiliar este acompanhamento, o projetista inseriu a opção de *velocidade*, que permite controlar a velocidade da simulação. Entretanto, mesmo na opção *muito lento*, é difícil identificar tudo que está sendo mostrado na interface. O P3 chegou a comentar: “*Se fosse mais lento, seria mais didático.*”.

Outra questão identificada refere-se à parte de simulação (figura 4.3), na segunda tarefa, que simula o código em Portugol. Como esta tarefa era apenas de visualização, não foram identificadas rupturas de interação. Mas para certificar que os alunos entenderam a simulação, perguntas específicas desta parte foram feitas na entrevista pós-teste, conforme pode-se observar no anexo F. Nestas perguntas, somente dois participantes (P3 e P6) responderam corretamente todas as questões feitas, demonstrando domínio do conteúdo. Os demais tiveram dificuldade em responder algumas questões, e outras, não souberam responder.

Diferente dos outros sistemas avaliados (e.g. ProfesSort), na parte de simulação do Bipide o aluno não pode expressar para o sistema dúvidas que possam surgir durante a interação, ou mesmo demonstrar o seu entendimento sobre o conteúdo abordado por ele. Com isso, a avaliação do quanto o Bipide pode contribuir com a aprendizagem do aluno precisa ser feita fora do sistema. Para sistemas de visualização e simulação, como o Bipide, tem-se então uma limitação na aplicação do MAC, por não haver interação do usuário com o sistema. Como neste tipo de sistema o aluno não precisa interagir, não é possível identificar rupturas de interação, uma vez que elas não existem. Entretanto, o uso da entrevista pós-teste permitiu que se obtivesse uma visão dos alunos sobre

os aspectos educacionais, relacionados à disciplina de AOC, abordados pelo Bipide. Apesar de não ter sido possível identificar aspectos relacionados à comunicabilidade do Bipide na parte de simulação, a entrevista mostrou-se como uma importante ferramenta para coletar informações sobre a aprendizagem dos alunos.

Finalizada a análise dos dados, gerou-se o seguinte perfil semiótico:

Caro usuário, o Bipide é uma ferramenta que eu criei para você, aluno da disciplina de Arquitetura e Organização de Computadores. Eu entendi que você está aprendendo lógica de programação e quer consolidar sua aprendizagem sobre o funcionamento de um processador. Para isso, eu criei o Bipide, que utiliza a linguagem Portugol e permite a você escrever códigos simples nesta linguagem e simulá-los para entender como o processador interpreta este código. Caso você não saiba Portugol, você pode utilizar a ajuda para escrever seus primeiros códigos. Na ajuda você encontra os principais comandos da linguagem portugol, informações sobre arquitetura e organização de computadores e a sintaxe que pode ser utilizada para escrever seus programas. No entanto, fique atento ao declarar suas variáveis, pois o Bipide é sensitive case e as mensagens informativas podem não ser suficientes para te auxiliar a resolver o erro apresentado. Ao concluir o código em Portugol, você pode ainda simular sua execução utilizando o recurso de Simulação. Na Simulação é possível acompanhar os valores assumidos pelas variáveis do seu algoritmo e as instruções em assembly geradas por ele. A Simulação contém muita informação, então você pode usar o recurso de velocidade, para tentar acompanhar todos os passos da simulação. Você pode ainda fazer uso dos recursos de pausar, parar ou repetir a simulação.

5.4 Discussão

Um estudo preliminar feito por Prates et al. [2004] apontou para considerações específicas a serem feitas na etapa de análise do MAC para o domínio educacional, e mostrou a necessidade de se fazer uma investigação mais aprofundada sobre o tema. Um dos pontos levantados pelo trabalho refere-se à possibilidade de distinguir as rupturas relativas à interação, daquelas relativas ao conteúdo sendo ensinado/aprendido. Para investigar este ponto, nos estudos de caso realizados, buscou-se distinguir as rupturas de interação das rupturas de conteúdo. No VCalc, em determinadas situações, teve-se dificuldade de distinguir se o problema referia-se à interação ou ao conteúdo. Esta dificuldade residia no conhecimento limitado dos avaliadores sobre o conteúdo

sendo tratado. Entretanto, o avaliador que tinha maior conhecimento sobre este conteúdo conseguiu diferenciar algumas destas rupturas, enquanto outras puderam ser distinguidas na entrevista pós-teste. No ProfesSort, o fato dos avaliadores terem conhecimento sobre o domínio da aplicação e também pelos *scaffolds* oferecidos, foi mais fácil diferenciar as rupturas ocorridas durante os testes dos participantes. À medida que o aluno recorria aos apoios, ele explicitava sua dúvida, permitindo aos avaliadores diferenciar as rupturas de interação e conteúdo. No Bipide, o fato do avaliador também conhecer o domínio facilitou a distinção das rupturas ocorridas e algumas dúvidas que surgiram, foram esclarecidas na entrevista pós-teste.

Estas análises ratificam as observações feitas por Prates et al. [2004], sobre a necessidade de se ter conhecimento do domínio para se identificar algumas rupturas de conteúdo, ou ainda diferenciá-las das de interação. A análise ficará mais precisa se tiver um avaliador que entenda do domínio ou talvez baste consultar o professor em alguns pontos, caso o avaliador consiga identificar se a dúvida refere-se à interação ou ao domínio. No caso do Ampliar [Prates et al., 2004] o envolvimento do professor na avaliação permitiu retirar as ambiguidades existentes na etiquetagem. Neste trabalho, como sempre tinha um avaliador relacionado com o domínio, as rupturas ocorridas foram distinguidas entre interação e conteúdo.

A aplicação do MAC no domínio educacional permitiu a obtenção de resultados interessantes, tanto do ponto de vista da interação, quanto de questões relacionadas à aprendizagem. Com relação à interface, conforme esperado, ele permitiu a identificação de rupturas de interação vivenciadas pelos alunos durante os testes. Com relação à aprendizagem, verificou-se que o MAC pode ser utilizado também para informar o professor, permitindo a identificação dos pontos de maiores dúvidas dos alunos e impacto do sistema no aprendizado.

Em sistemas educacionais específicos, como os que fazem uso de *scaffolds*, pode ser necessário o tratamento de questões específicas. No caso do ProfesSort, o *Não, obrigado!* referente ao conteúdo, é uma etiqueta possível de ser utilizada, relacionada com o botão de solicitação do sistema para fazer o passo pelo aluno. Prates et al. [2004] relatam também na seção *Novos Sintomas* questões específicas sobre o Ampliar. No caso do Ampliar, apesar de não fazer parte do objetivo do ensino do método, poderia ter um botão para execução de próximo passo. Embora ambos sistemas sejam baseados no mesmo modelo de apoio, as questões identificadas são específicas de cada um, ou seja, não relacionado apenas com o tipo de sistema, mas ao seu domínio.

Também identificou-se uma limitação sobre a aplicação do MAC em sistemas de visualização. Como o objetivo do MAC é analisar a metacomunicação sob o ponto de vista do usuário, ele precisa interagir com o sistema para perceber a comunicação

sendo feita através da interface. Em sistemas de visualização (ou simulação) se não for necessário o usuário fornecer alguma informação, ele não vai interagir com o sistema, apenas observará o que está acontecendo na interface. Neste caso, a utilização do MAC não permite avaliar a comunicabilidade ou identificar rupturas neste tipo de sistema interativo, e o uso de outros métodos (e.g. entrevistas) poderia trazer resultados mais significativos para este domínio.

Os resultados dos estudos realizados demonstraram que o MAC se aplica ao domínio educacional ao permitir a análise da comunicabilidade sob o ponto de vista do aluno e também pelo fato da metamensagem incluir aspectos sobre interação e questões educacionais. Além disso, foi interessante verificar que não apenas consegue apontar ruptura sobre as diferentes partes da metamensagem, mas muitas vezes (nos estudos de caso), até separadamente, sabe identificar sobre que parte ocorre a ruptura. Com relação às rupturas, observou-se que (i) a análise das rupturas de conteúdo geram indicadores sobre o aprendizado do aluno, ou seja, as rupturas frequentes apontam para dúvidas comuns; (ii) a análise das rupturas de um aluno em relação a outros pode ser um indicativo sobre seu aprendizado em relação à turma; e (iii) a mudança nas rupturas (e.g. sua redução) gera indicadores do impacto do sistema no aprendizado dos alunos.

Por fim, é importante enfatizar que não é objetivo deste trabalho argumentar o quanto o aluno aprendeu (ou não) com o sistema, mas demonstrar que estes indicadores podem servir para o professor verificar a aprendizagem do aluno. Assim, o MAC pode auxiliar *stakeholders* interessados em verificar questões relativas ao aprendizado.

5.5 Considerações sobre o uso MAC para o Domínio Educacional

O MAC permite avaliar a comunicabilidade de um sistema interativo a partir da interação usuário-sistema [Prates et al., 2000; de Souza & Leitão, 2009; de Souza et al., 2010]. No entanto, no domínio educacional é importante avaliar a interação do aluno com o conteúdo durante seu processo de ensino e aprendizado. Para isso, pode-se ter métodos distintos para avaliar cada um desses aspectos. Como o MAC analisa a comunicabilidade do projetista, e no caso do domínio educacional a metamensagem tem aspectos sobre interação e aprendizagem, o MAC consegue também gerar indicadores sobre esses dois aspectos.

Assim, a análise dos resultados obtidos nos estudos de caso da aplicação do MAC nos sistemas de apoio ao aprendizado, VCalc, ProfesSort e Bipide, vem aprofundar

a investigação iniciada em Prates et al. [2004]. Através desta análise foi possível (i) ratificar alguns pontos, como novos sintomas; (ii) identificar novos pontos, como as interpretações distintas para algumas etiquetas e (iii) enriquecer outros pontos, como a análise do impacto de sistemas educacionais na aprendizagem do aluno.

1. *Etiquetagem*

Em Prates et al. [2004] levantou-se a necessidade de se investigar a possibilidade de criação de um novo conjunto de etiquetas específico para o domínio educacional. Assim, o propósito da nossa investigação era verificar que tipo de etiquetas diferenciadas surgiam e se elas se aplicavam a mais de um contexto de ambiente educacional, com o objetivo de analisar se seria o caso de propor um novo conjunto de etiquetas específico para este domínio. No entanto, pelos estudos de caso realizados, descobrimos que as rupturas de conteúdo podem ser descritas pelas mesmas etiquetas e sem alterar os seus sintomas, porém se referem ao conteúdo. A ruptura de conteúdo pode ser um problema na emissão (enunciação ruim) ou na recepção (não aprendeu). Assim, não é necessário outro conjunto de etiquetas. A adaptação necessária é pequena e se refere à qualificação da etiqueta em relação à interação ou ao conteúdo.

Para realizar a etiquetagem de sistemas educacionais, o avaliador, com conhecimento em EngSem, deve considerar ao aplicar o MAC: *etiqueta*, *tempo*, *tipo* e *explicação*. Sendo *etiqueta* a expressão que categoriza a ruptura; *tempo* refere-se ao momento da interação em que a ruptura ocorreu, através do vídeo gerado pelo participante; *tipo* é o qualificador da etiqueta e pode se referir a uma ruptura de interação ou conteúdo; e *explicação* é a causa dessa ruptura. Ao classificar a etiquetagem desta forma, o avaliador terá uma visão geral sobre os principais problemas vivenciados por cada participante do teste.

Como o conjunto de etiquetas proposto pelo MAC não precisou ser adaptado, os sintomas também permanecem os mesmos. Para fazer a diferenciação entre ruptura de interação e conteúdo é necessário observar:

- *Etiquetas diretas*: a atribuição de algumas etiquetas às rupturas vivenciadas pelo aluno é feita de forma direta, conforme o que foi observado na interação do aluno com o sistema. Como a situação da etiqueta *Socorro!*, que se refere explicitamente a uma ruptura de conteúdo em que o aluno pede ajuda ao sistema.
- *Dependência do contexto*: neste caso, a distinção das rupturas de interação e conteúdo na etiquetagem dependem de outras ações observadas durante a

interação do aluno com o sistema. O caso da etiqueta *E agora?* apresentado no ProfesSort ilustra esta dependência, ou seja, o aluno vagava com o cursor por dois elementos da tela que ele já havia interagido. Então esta ruptura foi classificada como *E agora?* referente ao conteúdo.

- *Dependência de conhecimento do domínio:* para retirar a ambiguidade de algumas rupturas é necessário que o avaliador tenha conhecimento do domínio abordado pelo sistema. No caso do VCalc, por exemplo, um dos avaliadores desconfiou que um participante estava com dificuldade sobre o conteúdo tratado pelo sistema, e não como interagir com o VCalc. Esta desconfiança pôde ser confirmada na entrevista pós-teste.

A distinção entre etiquetas de interação e conteúdo, muitas vezes, vai depender do sistema avaliado. Na aplicação do estudo de caso do ProfesSort, como ele permite ao aluno explicitar suas dúvidas através dos *scaffolds*, no passo de etiquetagem, a distinção das rupturas de interação e conteúdo pode ser direta. Em sistemas de apoio ao aprendizado que não fazem uso de *scaffolds* a retirada de ambiguidades geralmente é possível com a participação de um especialista no domínio ou ainda na entrevista pós-teste. Com isso, cabe ao avaliador qualificar a ruptura conforme o sistema avaliado.

Uma situação diferente surgiu na etiquetagem do ProfesSort e refere-se à etiqueta *Não, obrigado!* de conteúdo. No caso do ProfesSort, por exemplo, o aluno entende que o sistema requer que ele faça os passos para consolidar a aprendizagem sobre o método. No entanto, quando apesar de não saber bem o método, ele resolve utilizar o botão *Executar próximo passo*, ele está recusando a proposta educacional disponibilizada pelo sistema. Esta situação representa uma ruptura do conteúdo associado ao *Não, obrigado!*. Porém, se o aluno já entendeu o método e faz uso do botão para chegar mais rapidamente a outro passo do método, o uso do botão não configuraria uma ruptura. Assim, apesar do sintoma ser o mesmo, ele pode ter interpretações radicalmente diferentes. Muitas vezes, como ocorreu no ProfesSort, o avaliador saberá diferenciá-los a partir das demais ações do aluno no sistema.

2. Interpretação

Em sistemas educacionais, algumas rupturas identificadas na etiquetagem podem não ser facilmente qualificadas como sendo rupturas de interação ou conteúdo, pois geram dúvidas no avaliador. Com isso, na etapa da interpretação, através de uma análise mais ampla, geralmente é possível fazer esta distinção.

A etiqueta *Desisto!* no domínio educacional pode ter consequências positivas quando relacionada ao conteúdo, ao indicar ao aluno que ele não tem o conhecimento esperado para utilizar o sistema. A ruptura *Para mim está bom!* é uma falha grave que, para o domínio educacional, pode ter consequências mais graves ainda. Esta gravidade se refere ao fato do aluno acreditar que domina o conteúdo abordado pelo sistema.

Esta é a situação das etiquetas *Para mim está bom!* e *Desisto!*. Para qualificar estas etiquetas como sendo de interação ou conteúdo, é necessário fazer uma interpretação do conjunto de rupturas gerado pelo aluno, pois para determinadas situações pode se referir a um problema de conteúdo (o aluno não tem o conhecimento esperado do conteúdo abordado pelo sistema) ou de interação (o aluno não entendeu a metacomunicação do projetista).

Outro ponto relevante para a interpretação é a possibilidade de, a partir da análise (e.g. frequência ou contexto) do conjunto de etiquetas, obter indicadores da parte educacional. Através destes dados é possível identificar se o sistema educacional auxilia na aprendizagem do aluno e ainda qual o impacto deste sistema neste aprendizado. Este indicador pode ser identificado, por exemplo, através da redução do número de etiquetas nas tarefas realizadas pelos alunos. Assim, estes indicadores podem ser utilizados pelos alunos, à medida que têm no sistemas uma forma de identificar o seu domínio sobre determinado conteúdo, e pelos professores, à medida que podem identificar problemas ou dúvidas mais frequentes dos alunos.

3. *Perfil Semiótico*

Para geração do perfil semiótico, a principal consideração a ser feita refere-se à identificação do objetivo educacional, utilizando o mesmo *template* proposto pela EngSem, na reconstrução da metamensagem. Assim, na geração do perfil semiótico é possível identificar a fala explícita do educador, relacionado à parte educacional, com o projetista, relacionado à interação. Com isso, há uma combinação dos emissores da metamensagem reconstruída, considerando a participação do projetista e do educador.

Capítulo 6

Método de Inspeção Semiótica Intermediado

Dentre os métodos de avaliação propostos pela teoria da EngSem, não havia um método que permitisse a avaliação de comunicabilidade de um sistema com a participação do professor. Como parte da pesquisa deste trabalho, foi proposto um novo método denominado Método de Inspeção Semiótica Intermediado (MISI). A seção 6.1 apresenta uma visão geral do método, que então é detalhado na seção 6.2. Na seção 6.3 são apresentadas duas aplicações do MISI no domínio educacional.

Conforme apresentado na seção 3.2, o MIS é um método de inspeção e deve ser realizado por um avaliador que seja especialista em EngSem, tendo como foco a emissão da metamensagem [de Souza et al., 2006]. O capítulo 4 demonstrou que com a aplicação do MIS, o especialista pode identificar aspectos educacionais sendo comunicados, mas em algumas situações o conhecimento sobre o domínio pode ser necessário ter a participação de um especialista no domínio.

O MAC (ver seção 3.3) é um método que permite a avaliação da comunicabilidade do sistema a partir da interação usuário-sistema e tem como foco a recepção da metamensagem pelos usuários (no caso do domínio educacional, os alunos) [Prates et al., 2000]. Entretanto, no domínio educacional além da avaliação por especialistas e usuários (i.e. alunos), pode ser necessário envolver o professor na avaliação, para que ele faça uma apreciação da adequação do sistema para o seu processo de ensino e disciplina [Squires & Preece, 1999; Pardo et al., 2006]. Neste caso, nenhum dos dois métodos fundamentados na EngSem é adequado.

6.1 Visão Geral

O MISI tem como objetivo permitir a análise da qualidade da recepção da solução proposta pela equipe de *design* ao professor, em outras palavras, a visão do professor sobre o sistema. Às vezes, a metamensagem é para o aluno, mas pode ser interessante o professor avaliar o quanto esta metamensagem se adequa aos seus objetivos didáticos ou metodologia. Apesar de o professor não ser um usuário direto de sistemas de apoio ao aprendizado, ele é, muitas vezes, um usuário indireto. Isto porque, em muitas situações, ele é o responsável por definir como o aluno fará uso do sistema tanto dentro quanto fora da sala de aula. Assim, é fundamental para o sucesso de ambientes educacionais obter a visão do professor sobre a qualidade do sistema.

Por ser baseado na teoria da Engenharia Semiótica, o MISI tem como foco a comunicabilidade, a propriedade de um sistema de transmitir ao usuário os princípios de interação e decisões que o guiaram [Prates et al., 2000; de Souza, 2005; de Souza & Leitão, 2009]. Mesmo que existam outros métodos [Squires & Preece, 1999; De Villiers, 2004; Pardo et al., 2006] que permitam a um professor avaliar a qualidade de um sistema educacional, o seu foco não é na comunicabilidade. O foco do MISI é obter a visão do professor sobre a comunicabilidade de um ambiente de apoio ao aprendizado. Esta visão é importante por permitir ao professor entender, por meio da interface, para que o sistema serve, a que aluno ele se destina, quais as vantagens de utilizá-lo, como ele funciona e quais são os princípios gerais que definem as possibilidades de interação com ele. Uma vez que o professor consegue identificar estas questões ele terá melhores condições de avaliar se este sistema está (ou não) adequado às necessidades dos seus alunos e à sua metodologia de ensino.

Para obter a análise da visão do professor, o MISI [de Oliveira et al., 2008] combina passos do MIS [de Souza et al., 2006], para definir o roteiro da interação, com alguns passos do Método de Explicitação de Discurso Subjacente (MEDS) [Nicolaci-da Costa et al., 2004], um método de entrevista semi-estruturado. Por ser baseado no MIS e no MEDS, ambos os métodos qualitativos e interpretativos, o MISI também é qualitativo e interpretativo. Desta forma, ele permite não apenas a análise da visão do professor da solução proposta, mas também a identificação de quais critérios são relevantes para o professor na análise do sistema educacional.

O MEDS [Nicolaci-da Costa et al., 2004] é um método qualitativo de pesquisa originalmente utilizado nas ciências humanas e sociais. Em IHC, o seu principal objetivo é tornar visível aspectos da configuração interna humana, como preferências, dificuldades, desejos, dentre outros; importantes para o desenvolvimento de sistemas interativos. Assim, sua maior utilidade para a área de IHC é a possibilidade de captar o

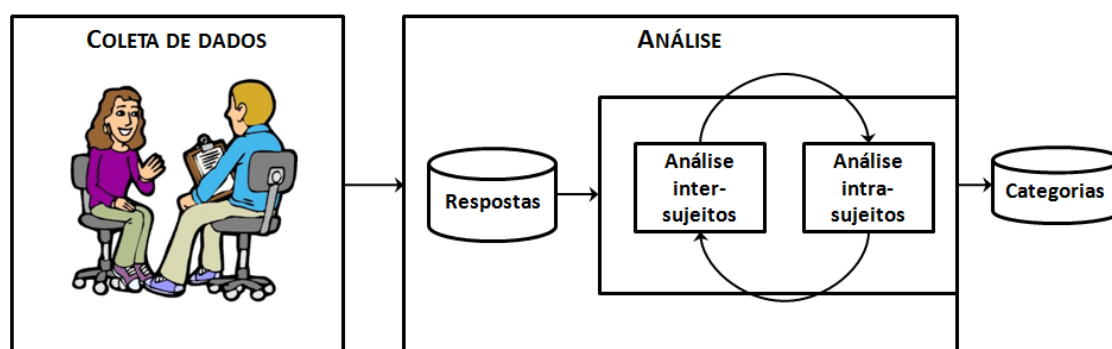


Figura 6.1. Método de Explicitação do Discurso Subjacente - MEDS

que não é perceptível por outros métodos. O MEDS prevê um roteiro semi-estruturado de perguntas abertas, ou seja, não é necessário ter perguntas prontas, mas sim um roteiro. O objetivo do roteiro é garantir que os mesmos pontos serão explorados com todos os entrevistados. O avaliador pode aprofundar mais ou menos, de acordo com o que é dito pelo entrevistado.

No MIS, cada inspeção é única, mas no MEDS a recorrência é importante. Diante desta amostragem, no MISI recomendamos que a inspeção conte com, no mínimo, 3 (três) participantes. Esse número visa garantir uma análise aprofundada na visão de diferentes participantes, sobre um mesmo sistema. Apesar dos participantes possuírem perfis similares (e.g. lecionar a mesma disciplina), cada um deles tem uma maneira diferente de lecionar determinado conteúdo e da mesma forma, podem ter uma visão diferente sobre um determinado sistema de apoio ao aprendizado. Assim, essas visões distintas enriquecem a qualidade da inspeção.

Em resumo, o MEDS [Nicolaci-da Costa et al., 2004] consiste em delinear os objetivos da entrevista e, em seguida, recrutar os participantes, visando a homogeneidade entre eles. Em seguida, elabora-se o roteiro com as perguntas abertas e inicia-se a coleta dos dados, que deve ser gravada para análise posterior. As entrevistas devem ser transcritas, para garantir a veracidade do que foi dito pelo entrevistado. A seguir, faz a análise da transcrição em duas etapas. A primeira etapa consiste em analisar cada entrevistada separadamente (intra-sujeito) e identificar os pontos de maior importância por entrevistado. Na segunda etapa a análise é feita com os dados de todos os entrevistados (inter-sujeito), com o objetivo de encontrar pontos em comum entre eles. Finalmente, interpreta-se os resultados obtidos por cada entrevista, individualmente e pelo grupo, gerando-se as categorias relevantes. A figura 6.1 apresenta uma visão geral sobre o método.

O objetivo do MISI é permitir a reconstrução da metagemagem e uma análise da comunicabilidade pelo avaliador com o apoio/colaboração do professor. Como seria um

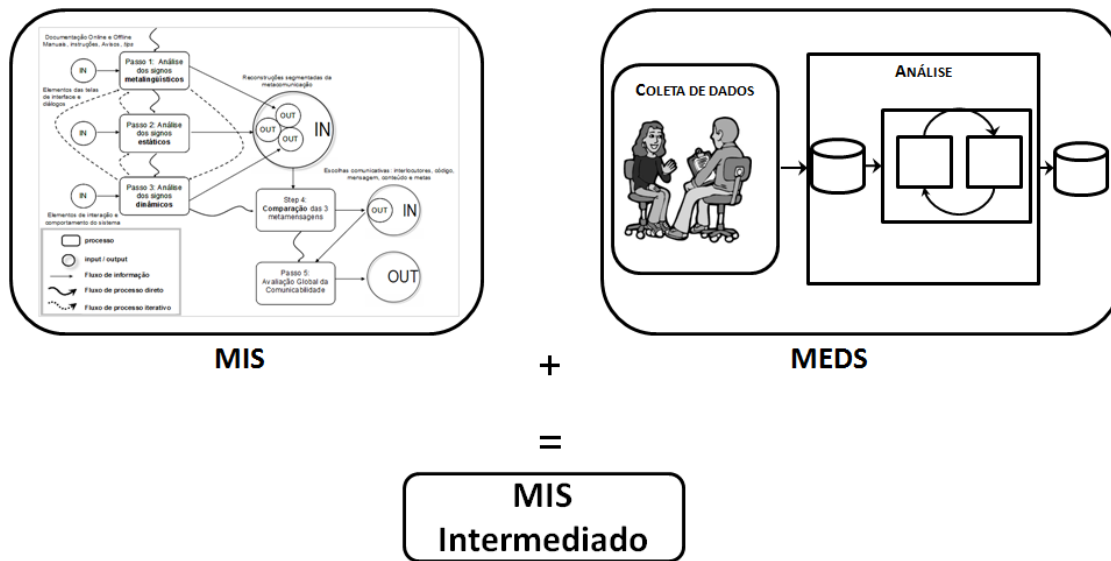


Figura 6.2. Visão Geral do MISI

custo muito alto ensinar o MIS para o professor e os conceitos de EngSem, o objetivo é um especialista, por meio de uma entrevista baseada na interação com o sistema (ou que guia a interação), obter a visão do professor a partir de uma entrevista. Assim, decidiu-se por criar o MISI, um método que é baseado no MIS e no MEDS, conforme pode-se observar na figura 6.2.

O MISI combina o MIS [de Souza et al., 2006] e o MEDS [Nicolaci-da Costa et al., 2004] de forma que através do exame do professor dos diferentes níveis de signos do sistema e entrevistas semi-estruturadas relativas à sua visão o avaliador coleta dados para reconstruir a visão do professor sobre a comunicabilidade do sistema. No MISI, ao longo de todo o processo, o avaliador utiliza-se de um roteiro que garante a inspeção de todo o sistema. Através de um roteiro a inspeção em cada nível de signos (metalinguístico, estático e dinâmico) é realizada, indiretamente, pelos especialistas do domínio. O especialista do domínio sem ter conhecimento do MIS ou EngSem, gera os dados para a inspeção semiótica do avaliador sob o ponto de vista do especialista. Na próxima seção apresentamos detalhadamente o MISI.

6.2 Descrição dos Passos

O MISI consiste em guiar a interação do professor com o sistema por meio de um roteiro baseado no MIS, e à medida que ele interage, entrevistá-lo sobre a sua percepção da metamensagem do sistema, ou seja, a quem se destina, que problemas resolve, como e por quê. Desta forma, é possível coletar indicadores da comunicabilidade do sistema

com a participação do professor. Assim, o método requer tanto que o roteiro inclua a definição dos passos de interação, quanto as questões a serem exploradas na entrevista.

Os passos da interação são baseados nos passos do MIS, ou seja, a primeira parte da interação requer que o participante leia o material que apresenta os signos metalinguísticos. Este material pode ser apresentado em forma eletrônica ou impressa. Em seguida, é feita a entrevista para a identificação da metamensagem relativa a esta parte. Feito isso, dá-se início à interação com o sistema.

Para cada tela com a qual o participante irá interagir, primeiro o avaliador, por meio da entrevista, deve explorar com ele a sua percepção sobre os signos estáticos, a seguir ele interage com o sistema, e o avaliador explora os signos dinâmicos. Nota-se que esta etapa difere do método original do MIS, no qual no passo 2 (dois) o avaliador inspeciona todos os signos estáticos, para só então inspecionar os dinâmicos no passo 3 (três). O objetivo ao se juntar estes 2 (dois) passos é permitir a avaliação do participante (que não é especialista em IHC ou EngSem) dos signos estáticos no contexto da interação, e o contraste entre signos dinâmicos e estáticos pelo próprio participante, por meio da entrevista.

Após a identificação dos significados atribuídos aos signos estáticos e dinâmicos, o avaliador, por meio de perguntas, revê com o participante a metamensagem identificada. As perguntas tem como objetivo verificar se a percepção do professor sobre os signos identificados mudou. Com isso, é possível observar se foram apontadas mudanças em relação à metamensagem reconstruída a partir da inspeção dos signos de metacomunicação. Após a interação, o avaliador explora, por meio da entrevista, a experiência do usuário levando em consideração outros critérios que sejam relevantes para o contexto ou uso do sistema.

Para a realização do MISI, é necessário que o avaliador seja especialista em EngSem, para que a avaliação seja conduzida de forma a obter informações relevantes que, posteriormente, serão analisadas quanto às questões de comunicabilidade. Assim, como o MIS e MEDS, o MISI pode ser realizado com apenas um avaliador. Entretanto, o aumento do número de avaliadores pode permitir uma análise mais rica dos dados coletados.

A figura 6.3 apresenta uma visão geral do MIS Intermediado e na seção 6.2.1 são detalhados os passos necessários para a aplicação desse método.

6.2.1 Passos do MISI

Na preparação faz-se o roteiro de perguntas e de interação do usuário que guiará a coleta de dados a ser feita. A análise é feita com base na transcrição das entrevistas,

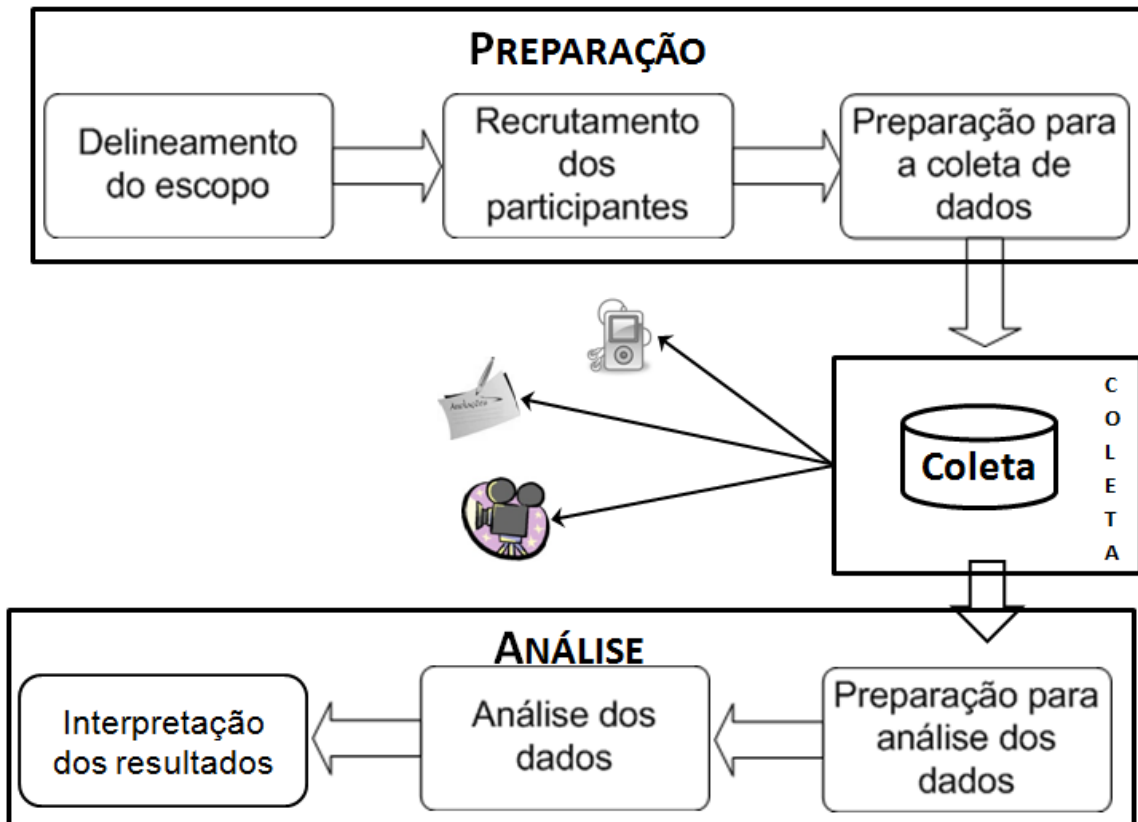


Figura 6.3. Método de Inspeção Semiótica Intermediado - MISI

mas tem por objetivo a reconstrução da metacomunicação sob o ponto de vista do professor. A seguir apresentamos cada passo do método detalhadamente.

1. **Delineamento do escopo:** consiste em delimitar a parte do sistema a ser inspecionada e os cenários a serem considerados. O cenário é utilizado a fim de definir o objetivo do usuário a ser considerado na avaliação, definindo assim o foco da avaliação. No MISI, nem sempre o cenário será necessário, uma vez que o usuário indireto (e.g. educador) estará considerando a sua experiência e necessidade de uso do sistema a ser inspecionado. Entretanto, caso se deseje focar ainda mais a visão do participante, a geração do cenário é recomendada.
2. **Recrutamento dos participantes:** após a definição do escopo, deve-se selecionar os participantes para entrevista. É importante ressaltar que os participantes devem ser um grupo homogêneo em relação às características que se deseja analisar. No caso do contexto educacional, pode ser em relação à lecionar a mesma disciplina.
3. **Preparação para a coleta de dados:** nesta etapa, deve-se elaborar todo o

material a ser utilizado durante a inspeção do sistema, que inclui tanto os passos da interação, quanto questões de entrevista.

O roteiro inicia-se com a apresentação dos objetivos da pesquisa e do método de inspeção para o participante. Em seguida, faz-se o levantamento do perfil detalhado do participante. Nesta etapa deve-se incluir questões sobre a formação acadêmica do participante, conhecimento sobre o conteúdo abordado pelo sistema e tempo que leciona/trabalha com o conteúdo ou disciplina relacionado ao sistema a ser inspecionado. No caso do domínio educacional, pode-se incluir, além destas questões, perguntas sobre metodologia de ensino, recursos que utiliza para consolidação/aprendizagem dos alunos e visão sobre sistemas de apoio ao aprendizado. Tais questões para o domínio educacional podem indicar a experiência que o professor tem em lecionar determinado conteúdo e ainda os recursos de que ele faz uso para apoiar e/ou consolidar a aprendizagem dos alunos.

Identificado o perfil do participante, parte-se para a próxima etapa, que consiste na definição da inspeção do sistema pelo professor, a começar pela análise dos signos metalinguísticos. Como esta etapa requer que o participante leia o material que apresenta os signos de metacomunicação, na preparação deve-se identificar o material a ser utilizado e prepará-lo para apresentação ao participante (e.g. impressão da parte relevante do sistema de ajuda) de acordo com o escopo definido. Além disso, deve-se ter também o roteiro da entrevista para se identificar a percepção do participante sobre a metamensagem inicial do sistema incluindo as seguintes questões: *(i) a quem se destina o sistema, (ii) que problemas é capaz de resolver, (iii) como e (iv) por que.*

A seguir devem ser determinadas as telas a serem percorridas e as tarefas a serem executadas pelo participante durante a inspeção de signos estáticos e dinâmicos. Essa definição é de responsabilidade do avaliador (especialista). Como ocorre no MIS, o especialista realiza uma inspeção preliminar do sistema e procura identificar os principais usuários do sistema, bem como as principais tarefas que este usuário pode realizar. Conforme o conhecimento do especialista, pode-se também identificar tarefas mais problemáticas ou de difícil compreensão por parte dos alunos.

Em seguida, os pontos a serem questionados para a inspeção em cada nível também devem ser definidos. Por exemplo, questões sobre o que entendeu dos signos estáticos apresentados em uma tela (e.g. "O que você acha que este botão faz?") e signos dinâmicos (e.g. "Agora que você interagiu com esta opção, ela se comporta conforme esperado? Por que (não)?").

Finalmente, o roteiro deve conter também questões associadas à experiência de uso do sistema e também relacionadas com o objetivo da avaliação. Por exemplo, "*Você acha que este sistema poderia ser útil para o aprendizado de seus alunos?*" e "*Qual sua opinião sobre as possibilidades de apoio à aprendizagem oferecidas por este sistema?*".

Vale ressaltar que, embora tenha-se usado como exemplo as perguntas finais que seriam feitas pelo avaliador, seguindo o MEDS [Nicolaci-da Costa et al., 2004], o roteiro não deve conter perguntas prontas, mas os pontos a serem questionados e explorados. Por exemplo, no roteiro se teria o ponto "*Entendimento sobre o signo A*" e "*Se gostaria de usar o sistema em sua disciplina. Em caso afirmativo, como. Em caso negativo, por que não?*".

Nesta etapa também se prepara o *Termo de consentimento* e realiza-se o *teste piloto*.

4. **Coleta de dados:** esta etapa inicia-se com a apresentação dos objetivos da pesquisa. Em seguida, o participante é convidado a ler e assinar o termo de consentimento [CNS, 1996]. A realização da entrevista deve ser feita em um local onde o participante não seja interrompido para que ele possa se concentrar no sistema e não ter sua atenção desviada das atividades do teste. Toda a entrevista deve ser gravada para transcrição posterior, inclusive a interação. A gravação do áudio pode não ser suficiente, uma vez que o participante pode apontar para elementos da interface e não expressá-los verbalmente (i.e. este botão se comportou de maneira inesperada). Com isso, a gravação da interação auxilia na identificação destas possíveis ambiguidades que podem ocorrer durante a inspeção. Assim, é recomendado que se grave o áudio e a interação (se sincronizados) ou o vídeo (deve ser capaz de distinguir elementos apontados na tela).
5. **Preparação para análise dos dados:** consiste na transcrição de todas as entrevistas. Deve-se ter o cuidado de incluir na transcrição pausas mais longas na fala ou hesitação, e também referências dêiticas ao sistema (e.g. apontou para determinado elemento do sistema).
6. **Análise dos dados:** conforme proposto pelo MEDS, esta etapa é feita em duas etapas, intra-sujeito e inter-sujeito. A partir da transcrição das entrevistas, o avaliador faz a análise *intra-sujeito*, com o objetivo de identificar categorias recorrentes de cada análise, por participante. Em seguida, o avaliador faz a análise *inter-sujeito*, gerando a reconstrução da metamensagem a partir da percepção dos participantes. O avaliador também procura identificar e registrar questões

levantadas para cada categoria, possíveis conflitos de opinião e potenciais inconsistências do sistema em relação às categorias de signos. Estes problemas podem estar relacionados a um dos níveis (e.g. ao olhar os signos estáticos de uma tela, o participante poderia comentar "Não faço idéia para que serve este botão"), ou mesmo à inconsistência entre os níveis (e.g. ao interagir com um botão, o participante poderia comentar "Ué, achei que este botão faria X, mas agora estou vendo que na verdade ele faz Y."). Assim, esta análise tem dois focos (1) identificação das categorias relevantes ao uso do sistema pelos participantes e (2) a metamensagem com base na sua visão.

(1) *Identificação das categorias relevantes ao uso do sistema pelo participante:* a partir da transcrição do áudio de cada entrevista, o avaliador categoriza as principais características apontadas pelos especialistas durante a inspeção. Estas categorias estão diretamente relacionadas ao roteiro gerado para o acompanhamento das entrevistas e aos fatores considerados como relevantes para cada participante. De forma geral, estas categorias podem se referir ao perfil do participante e também aos problemas de interação encontrados. No caso do domínio educacional, além das categorias citadas anteriormente, outras de cunho educacional podem surgir, por exemplos referentes à metodologia de ensino adotada pelo professor, estilo de aula, objetivo educacional do sistema inspecionado, dentre outros.

(2) *Reconstrução da metamensagem com base na visão dos participantes:* nesta etapa o avaliador, por meio da análise dos resultados, interpreta os dados, com o objetivo de identificar a visão que o usuário indireto teve da metamensagem do projetista, e problemas identificados nesta metamensagem. Ele então reconstrói o *template* de metamensagem para cada participante com base nas suas respostas à entrevista.

7. Interpretação dos resultados: a partir da análise feita no passo anterior, o avaliador faz a interpretação dos dados, identificando qual a visão que o usuário indireto teve da metamensagem do projetista, e problemas identificados nesta metamensagem. Além disso, faz-se uma análise das categorias que surgiram e da caracterização da solução proposta do projetista de acordo com elas.

Na seção 6.3 são apresentados os estudos de caso em que aplicou-se o método proposto para obtenção da apreciação da visão do professor sobre dois sistemas de apoio ao aprendizado.

6.3 Estudos de Caso

6.3.1 VCalc

A primeira aplicação do MISI foi feita no aplicativo VCalc¹. Conforme apresentado na seção 4.2, o VCalc é um aplicativo desenvolvido no Departamento de Matemática (DMAT) da UFMG para ser utilizado nas aulas de Cálculo I. Este aplicativo tem como objetivo facilitar a investigação da taxa de variação de funções em um determinado intervalo escolhido pelo usuário, mostrando tanto a parte algébrica quanto a gráfica.

A seguir é descrita a aplicação do MISI no aplicativo VCalc e os resultados obtidos com a avaliação. O roteiro completo usado nesta inspeção, incluindo termo de consentimento e os pontos a serem explorados com cada participante, encontra-se no Apêndice I. A seguir são descritas as etapas da aplicação do MISI no aplicativo VCalc.

1. **Delineamento do escopo:** por se tratar de um sistema pequeno com 4 (quatro) telas e 12 (doze) funções, optou-se por realizar a inspeção de todas as funcionalidades do VCalc. Considerando que todos os entrevistados tinham ampla experiência (mais de 12 anos) em ministrar a disciplina de Cálculo I, não foi necessário gerar cenário para a inspeção.
2. **Perfil dos participantes:** participaram deste estudo de caso três professores do DMAT da UFMG. Todos lecionam ou já lecionaram a disciplina de Cálculo I e nunca tinham utilizado o sistema, fator importante que foi considerado, para evitar que o conhecimento prévio sobre o sistema impactasse nos resultados.

O fato do participante conhecer o sistema poderia afetar a avaliação, uma vez que ele já teria uma visão pré concebida sobre o sistema. Isso poderia, por exemplo, fazer com que ele não identificasse problemas que o sistema possui, por ser um usuário que está habituado com o aplicativo. Caso houvesse somente participantes que já conhecessem o sistema, seria necessário fazer um levantamento sobre o uso que eles fazem do sistema para então elaborar o roteiro considerando possibilidades de uso distintas e/ou complementares das conhecidas pelos participantes.

Os professores tinham experiência em lecionar a disciplina, pois tinham lecionado por 12, 14 e 30 anos, o que demonstra a vasta experiência dos participantes. Com relação à formação, dois dos participantes possuem pós doutorado, enquanto o outro, doutorado em matemática.

¹Disponível em <http://www.mat.ufmg.br/gepemnt/vcalc>

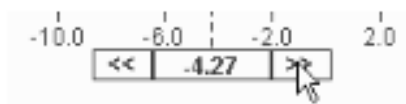


Figura 6.4. VCalc: Movimentação gráfica

3. **Preparação para a coleta de dados:** para guiar a entrevista, preparou-se um roteiro com pontos a serem abordados com todos os entrevistados. O roteiro utilizado e o termo de consentimento estão disponíveis no anexo I. A seguir são listadas as etapas de aplicação do MISI no aplicativo VCalc contidas no roteiro.

- *Identificar o perfil do entrevistado:* formação, tempo que leciona Cálculo I, método de ensino utilizado para aprendizagem dos alunos e utilização de sistemas de apoio ao aprendizado.
- *Inspeção do sistema usando o MISI:* este passo subdivide-se em duas etapas: passo 1 e passo 2.

Passo 1: o participante era convidado a ler a documentação eletrônica do sistema² e, a partir desta leitura, era convidado a responder as questões para descrever sua visão sobre o sistema para que, posteriormente, fosse possível reconstruir a metamensagem (Quem é o usuário deste sistema? O que o sistema faz? Como ele pode ser utilizado? Qual seu objetivo educacional?).

Passo 2: como o VCalc é um aplicativo pequeno, foi possível inspecioná-lo por completo. Essa inspeção consistiu nos pontos: mudança de função, taxa de variação, *zoom* e mudança de intervalos. Para cada ponto de interação: obteve-se a visão do professor sobre signos estáticos, ou seja, sobre o significado dos signos visíveis na tela. Em seguida, realizou-se a inspeção dos signos dinâmicos. O professor interagiu com o sistema enquanto era questionado se o comportamento dos signos presentes na interface correspondiam às suas expectativas. Nesta etapa, a expectativa de alguns dos participantes sobre o comportamentos dos signos dinâmicos foi diferente do esperado ao analisar os signos estáticos. Um exemplo é fala de um participante sobre o signo apresentado na figura 6.4 (movimentação gráfica). O participante comentou na análise estática, “...se eu clicar aqui [no signo de movimentação gráfica], ele vai mudando. Ou talvez eu possa mudar arrastando-o.”. Na análise dos signos dinâmicos, o professor surpreso, disse, “É uma animação, não era o que eu pensava.”.

²Disponível em <http://www.mat.ufmg.br/gepemnt/vcalc/> Acesso em maio 2010.

O professor também era questionado sobre uma possível interpretação que os alunos teriam sobre os signos da interface. Um participante experiente em lecionar Cálculo I e que nunca havia usado sistemas de apoio ao aprendizado com seus alunos, quando questionado sobre a interpretação dos alunos a respeito do signo *New Area* (ver figura 4.1), disse “...meus alunos vão entender melhor que eu, pois estão acostumados a lidar com computador diariamente.”

Como os professores que participaram da avaliação têm experiência em lecionar o conteúdo abordado pelo VCalc, acreditava-se que eles conseguiriam identificar os principais pontos de dúvida que os alunos poderiam ter ao utilizar o aplicativo. Esta observação pôde se confirmar, conforme será apresentado na análise dos dados. Por fim, novamente as questões para reconstrução da metamensagem eram retomadas, como forma de verificar a consistência entre as mensagens geradas nos passos 1 e 2.

- *Potencial de uso do sistema*: nesta etapa, o participante era questionado sobre as possibilidades de uso do VCalc, dentro ou fora da sala de aula. Ainda buscou-se verificar se o sistema estava alinhado com sua metodologia de ensino, qual seria o custo para incluir a ferramenta na disciplina e ainda se recomendaria o sistema para colegas.

4. **Coleta de dados**: para o registro da entrevista utilizou-se a gravação em áudio e vídeo. Cada avaliação contou com a participação de dois especialistas, em IHC e EngSem, e um professor. Enquanto um avaliador utilizava o roteiro para interagir com o professor, o outro fazia anotações sobre as informações relevantes. Por ser a primeira aplicação do método, optou-se por dois avaliadores, apesar do método requerer apenas um avaliador, uma vez que toda a entrevista é registrada em áudio e vídeo.

A coleta dos dados foi realizada na sala de teste do DCC/UFMG. Esta sala é composta por duas câmeras filmadoras, que permitem a gravação das expressões faciais e da interação do participante da avaliação; um computador; um microfone para auxiliar na qualidade da gravação do áudio e dois televisores, sendo um conectado à filmadora que grava a interação do participante com o sistema e outro conectado ao computador.

Inicialmente o participante obtinha informações sobre a avaliação a ser realizada, como objetivo da avaliação e o método a ser usado. Ele também era orientado a

ler e assinar o termo de consentimento. Em seguida, a inspeção do sistema era iniciada, seguindo os passos do roteiro.

Primeiramente, o professor era convidado à responder questões relacionadas ao seu perfil, questões estas que foram feitas nesta parte inicial também como forma de deixar o participante mais à vontade. O passo seguinte do roteiro consistia no professor ler a documentação do sistema e responder às questões que permitissem aos avaliadores reconstruir a metamensagem sob sua visão. A seguir, por meio do roteiro, o professor era convidado a identificar os signos estáticos presentes na interface do aplicativo e expressar o seu entendimento sobre eles. Então o professor relatava sua expectativa sobre tais signos. Ao fim da interpretação dos signos estáticos, o professor era convidado a interagir com o sistema, explorando os signos dinâmicos. Durante a interação, eles reinterpretavam os signos interpretados anteriormente. Caso o aplicativo se comportasse de uma maneira inesperada pelo professor, eles reconheciam a situação e externalizavam o novo entendimento adquirido em relação àqueles signos. Entretanto, se o comportamento do sistema correspondia às interpretações feitas anteriormente, eles relatavam com satisfação a confirmação de suas expectativas. Além da visão do professor, o roteiro também continha pontos que permitiam explorar a expectativa do professor em relação à visão do aluno sobre o sistema.

Finalizada a inspeção dos signos estáticos e dinâmicos, o professor novamente era convidado a responder algumas perguntas que o conduziam a relatar novamente sua visão sobre a metamensagem do designer. Finalmente, o professor era entrevistado sobre sua experiência com o sistema, sua avaliação desta ferramenta e sobre a possibilidade de uso dela em suas aulas.

5. **Preparação para a análise dos dados:** a partir da filmagem realizada, transcreveu-se toda a entrevista realizada, seguindo os passos do MEDS.
6. **Análise dos dados:** conforme descrito na seção 6.2.1, utilizou-se a análise inter e intra-sujeito, bem como a identificação das categorias relevantes e a reconstrução da metamensagem.
7. **Interpretação dos Resultados:** a aplicação do MISI no aplicativo VCalc permitiu identificar categorias relevantes para o professor na consideração de um sistema educacional. Também foram identificados problemas relacionados tanto à interação, quanto ao conteúdo. Finalmente foi possível reconstruir e fazer uma apreciação da metamensagem sendo enviada pelo projetista. A seguir são descritos cada um dos tipos de indicadores obtidos.

Categorias Relevantes ao Uso do Sistema

As categorias relevantes para o domínio educacional sob o ponto de vista do professor estão diretamente relacionadas aos pontos abordados no roteiro. São elas:

- *Adequação ao processo de ensino*: consiste em verificar se a ferramenta avaliada está em conformidade com os métodos de ensino utilizados pelo professor. Por meio da análise, foi possível concluir que, para um bom uso do sistema, o VCalc precisaria do apoio do professor para orientar o aluno para que ele seja completamente efetivo. Eles enfatizaram que sem o auxílio do professor, o uso do sistema pelos alunos pode ficar comprometido.
- *Interface*: refere-se aos aspectos visuais do sistema e a representação utilizada por ele. Percebeu-se uma preocupação por parte dos professores em relação à representação utilizada no VCalc. Segundo os participantes, o sistema é fácil de usar e o aluno tem facilidade em entendê-lo. Um professor argumentou: “...é um aplicativo interessante e eu acho que o que ele tem de muito positivo é essa facilidade de uso.”.
- *Conteúdo abordado*: consiste em identificar quais tópicos da disciplina são abordados pelo sistema. Os professores identificaram quais tópicos da disciplina de Cálculo I são abordados pelo sistema. Segundo um dos participantes “...o sistema pode ser bem utilizado em curso de Cálculo I, exatamente no estudo interativo gráfico do conceito de derivada.”. Houve também verificação sobre o comportamento do sistema para os casos especiais - para ver se tratava devidamente. Um dos participantes analisou especificamente alguns casos especiais que julgou de interesse: “...Eu estou dizendo isto pelo seguinte, estes são os exemplos de coisas que não têm derivadas. É um exemplo onde você pode mostrar que você tem que usar o limite para calcular, aí fica fácil ver que o limite é zero...”.
- *Logística para uso*: refere-se a equipamentos, local e presença de aluno-monitor para que seja possível usar o sistema em sala de aula. Segundo os professores, para fazer uso do VCalc em suas aulas, eles precisaram trabalhar com turmas menores (visto que as turmas de cálculo do DCC/UFMG têm, em média, 50 alunos). Além disso seria necessário ter um computador por aluno e os professores precisariam contar com o auxílio de um aluno-monitor.

Um outro tipo de problema identificado refere-se aos problemas de interação

identificados pelos professores. Eles foram capazes de identificar problemas tanto em relação à expressão, quanto ao conteúdo dos signos apresentados.

Com relação à expressão, os problemas identificados foram sobre a representação de signos estáticos em termos matemáticos, se representavam bem (ou não) o conceito associado. Um professor comentou que não chamaria o botão *rate of change* (ver figura 4.1) desta forma: "...eu não chamaria de taxa de variação, eu chamaria mais de variação média..." e explicou a razão para isto "eu vou daqui para lá em um certo tempo e não me interessa qual caminho eu percorri, eu só sei que percorri uma distância em um certo tempo.". Com relação ao conteúdo um professor analisou o comportamento do sistema para casos especiais em que a função não tem derivada. Ao perceber que o sistema não os tratava adequadamente, considerou-o um problema grave e explicou por que. "...O sistema cometeu um erro, pois não tem derivada no ponto zero. Como eu calculo sempre da esquerda para a direita, a taxa de variação é um."

Em relação à interação, o estudo permitiu identificar aspectos da interface que poderiam ser melhorados, (i.e. o signo estáticos 1x1 que não levou à interpretação desejada). Além disso, o caso especial que o sistema não trata precisa ser resolvido para evitar que o aluno tenha uma interpretação errônea sobre o conteúdo abordado pelo sistema.

Com a aplicação do estudo de caso também foi possível perceber que o professor, sem o auxílio de um especialista em IHC e EngSem, teria dificuldades de realizar a inspeção, da mesma forma que um especialista em IHC e EngSem teria dificuldades em diferenciar problemas de interação e conteúdo. Ao combinar o MIS e o MEDS, o MISI permite uma sinergia entre as habilidades e competências do especialista em EngSem e do especialista do domínio. Desta forma, o método permitiu que se obtivessem indicadores tanto sobre aspectos educacionais relevantes para o educador, quanto sobre a comunicabilidade do sistema. Esse indicador reafirma a necessidade de ter equipes multidisciplinares no desenvolvimento de sistemas de apoio à aprendizagem, à medida que maximiza a possibilidade de se ter um sistema educacional que envolva conceitos importantes, na visão do professor, e uma interface de qualidade, sob a visão do especialista em IHC e EngSem.

Com relação à reconstrução da metagemagem, houve uma consistência da mensagem identificada nos passos 1 (inspeção dos signos de metacomunicação) e 2 (inspeção dos signos estáticos e dinâmicos). No entanto, foi possível perceber que, após a interação dos professores com o sistema, a visão do sistema ficava

mais aprofundada e precisa. Por exemplo, após a leitura da documentação todos identificaram que o sistema se destinava a alunos de Cálculo I para ensino de taxa de variação e derivadas fazendo uso de gráficos. Após a interação os professores foram capazes de identificar aspectos específicos do conteúdo (e.g. estudo local da função) a serem abordados ou mesmo outros conceitos (e.g. conceito de escala) e também fizeram considerações sobre seu uso em sala de aula (necessidade de preparação de atividades e disponibilidade de laboratórios). Da mesma forma, alguns vislumbraram um uso ainda mais amplo do sistema por outros usuários, como por exemplo, professores que poderiam fazer uso do sistema para preparação de aulas.

Reconstrução da metagem na perspectiva do participante

A metagem consolidada do VCalc sob o ponto de vista do projetista e do educador é:

Caro aluno, o VCalc é um sistema que foi desenvolvido para você, [Quem você é?] aluno de Cálculo I que já aprendeu ou está aprendendo agora o conceito de taxa de variação e de derivada. Ele vai ajudá-lo no aprendizado deste assunto, mais especificamente no [O que você precisa aprender?] entendimento da taxa de variação de uma função em um determinado intervalo. Para isso, ele [Que atividades te permitem atingir este aprendizado?] permite a visualização gráfica da função em um determinado intervalo, da reta que passa por dois pontos sob a função e que indica a taxa de variação ou variação média e outras informações relacionadas com a função, com os pontos que estão sob a função e sobre a reta que passa por estes dois pontos. Para utilizar o sistema [Como você pode interagir com o sistema?] você deve informar a função, o intervalo, a coordenada no eixo x dos dois pontos e variar estas informações para fazer um estudo da taxa de variação. Fique atento à taxa de variação apresentada, pois ela sempre é calculada da esquerda para a direita. Quando precisar deslocar o gráfico, você precisará de um cuidado especial para pará-lo. Por abordar somente o estudo sobre a taxa de variação, este software, apesar de tratar apenas de uma pequena parte do conteúdo de Cálculo I, é simples, permitindo que você foque nos conceitos, e não no aprendizado do sistema propriamente dito. O sistema não traz embutido atividades educacionais, o que requer que o

professor prepare atividades dirigidas ou que orientem o aluno no uso do software para o aprendizado dos conceitos.

6.3.2 ProfesSort

A segunda aplicação do MISI foi feita no sistema ProfesSort [de Castro & Prates, 2009]. O ProfesSort é uma ferramenta desenvolvida no DCC/UFMG (ver seção 4.3). Esta ferramenta foi desenvolvida para ser utilizada na disciplina de algoritmos e estruturas de dados II (AEDs II), com o objetivo de auxiliar os alunos no aprendizado de algoritmos de ordenação.

A segunda aplicação do MISI teve como objetivo coletar mais indicadores sobre a aplicabilidade do método a sistemas educacionais. As hipóteses eram que, com a aplicação do MISI em outro aplicativo, o método permitisse a identificação de problemas relacionados à comunicação projetista-usuário, sob a perspectiva do professor. A seguir, é detalhado a aplicação do método no sistema ProfesSort.

1. **Delineamento do escopo:** como o ProfesSort possui 5 (cinco) métodos de ordenação de algoritmos (*quick sort*, *recursivo e não recursivo*, *heap sort*, *seleção e inserção*) e alguns são semelhantes em termos de lógica, optou-se por avaliar apenas 2 (dois) destes métodos: *quick sort recursivo* e *heapsort*. Esta escolha deve-se ao fato destes métodos serem de difícil entendimento por parte dos alunos e os professores, geralmente, precisam dedicar um tempo maior para explicá-los. Nesta avaliação também não foi necessário a geração de cenários para a inspeção do sistema devido à experiência que os professores tinham com o conteúdo abordado pelo sistema.
2. **Perfil dos participantes:** participaram desta avaliação 4 (quatro) professores do DCC/UFMG, sendo 2 (dois) doutores e 2 (dois) pós doutores, todos na área de computação. Três desses professores, no período da avaliação (maio/2009), lecionavam a disciplina de Algoritmos e Estrutura de Dados II (AEDS II) no DCC/UFMG. Com relação à experiência em lecionar a disciplina de AEDS II, o perfil dos participantes foi bem heterogêneo. Um professor lecionava a disciplina há 17 (dezessete anos), enquanto outros dois lecionavam há um ano, embora tivessem sólido conhecimento sobre os métodos de ordenação abordados pelo sistema.

O outro professor tinha experiência em lecionar a disciplina Algoritmos e Estrutura de Dados I (AEDS I), porém foi convidado a participar da avaliação por seu conhecimento na área de educação de informática. Além disso, esse participante

estava bem a par do conteúdo da disciplina, embora não o estivesse ensinando-o a alunos. Apesar deste participante ter um perfil diferente dos demais (não leciona a disciplina de AEDS II), a sua participação na inspeção foi muito rica, por ser muito detalhista e se preocupar com questões metodológicas. Como ele conhecia os métodos de ordenação abordados pelo ProfesSort, não foi necessário fazer nenhuma adaptação para a aplicação do MISI.

Estes professores têm o hábito de ensinar aos alunos os métodos de ordenação bolha, inserção, seleção, *heap sort*, *quick sort*, *merge sort* e *shell sort* [Cormen et al., 2001; Ziviani, 2004].

Dos quatro professores que participaram da inspeção, somente um deles conhecia o sistema ProfesSort e o indicava para que seus alunos fizessem uso dele. Inicialmente, considerou-se a possibilidade de descartar a inspeção feita por este professor, por acreditarmos que o seu conhecimento sobre a ferramenta pudesse impactar no resultado da avaliação. Do ponto de vista da comunicabilidade do sistema, a inspeção sofreu pouco impacto, pelo fato dele conhecer o funcionamento de alguns signos presentes na interface. Entretanto, este conhecimento prévio não foi empecilho para que ele identificasse problemas na interface do sistema e a sua inspeção agregasse valor à avaliação do ProfesSort.

3. **Preparação para a coleta de dados:** assim como a aplicação do VCalc, neste estudo de caso também preparou-se um roteiro com os pontos a serem abordados com todos os entrevistados. Este roteiro consistia de 3 (três) pontos gerais, detalhados a seguir. O roteiro completo incluindo o termo de consentimento e a documentação gerada sobre os signos metalinguísticos pode ser encontrado no anexo I.

- *Perfil do entrevistado:* inclui formação do participante, tempo de ensino em AEDS II, métodos de ordenação de algoritmos ensinados em sala, estilo de aula e recursos que utiliza para aprendizado e consolidação da aprendizagem dos alunos, visão sobre o uso de ferramentas de apoio ao aprendizado e quais ferramentas utiliza.
- *Inspeção do sistema usando o MISI:* para o ProfesSort, o participante foi convidado a realizar duas inspeções: uma para o método de ordenação *quick sort* recursivo e outra para o *heap sort*. Entretanto, o professor era convidado a falar sobre a especificidade de cada método, conforme a sua apresentação na interface do sistema. Para a inspeção dos signos metalinguísticos foi

gerado um documento que explicava as principais funcionalidades do ProfesSort, aprovado pelos autores do sistema. Este documento foi feito a partir de publicações sobre o ProfesSort.

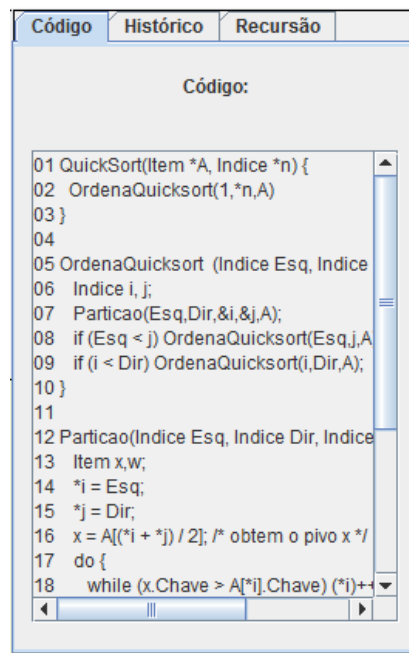
A inspeção do ProfesSort nos três níveis de signo (metalinguístico, estático e dinâmico) tinha como objetivo verificar a opinião do professor sobre a comunicação que os signos metalinguístico, estático e dinâmico faziam por meio da interface, e verificar a consistência da metamensagem gerada.

- *Potencial de uso do sistema*: o participante era questionado sobre a utilidade do ProfesSort para a disciplina de AEDS 2. Ele também falava sobre a adequação da ferramenta para uso em suas aulas e se era adequada aos seus atuais métodos de ensino. Além disso, o professor opinava sobre as possibilidades de apoio à aprendizagem oferecidas pelo sistema.

4. **Coleta de dados**: cada inspeção contou com a participação de um professor e de dois especialistas, em IHC e EngSem, sendo que um deles aplicava o MISI pela primeira vez. Este avaliador era responsável por fazer anotações que julgasse pertinentes, enquanto o outro, por meio do roteiro, dava seguimento à aplicação do MISI. A entrevista foi realizada no laboratório de teste, do DCC/UFMG, e toda a inspeção foi filmada e o áudio, gravado. O uso destes dois meios de registro visava assegurar que tanto o áudio quanto referências dêiticas ao sistema (e.g. apontou para determinado elemento do sistema) pudessem ser identificadas.

Assim como no estudo de caso do VCalc, o participante recebia instruções sobre o objetivo da pesquisa e, em seguida, era convidado a ler e assinar o termo de consentimento (ver anexo I). Seguindo os passos do roteiro, um dos avaliadores conduzia a entrevista. De início, fazia-se perguntas para coletar o seu perfil. Após a coleta do perfil, iniciava-se a inspeção do sistema por meio da leitura de um texto informativo sobre o ProfesSort (ver anexo I). A partir desta leitura, o professor respondia questões sobre o que o sistema faz, seu público alvo, como o sistema poderia ser usado por este público e qual seu objetivo educacional. A resposta a estas questões permitiriam ao avaliador reconstruir a metamensagem baseado nos signos metalinguísticos.

A seguir, iniciava-se a inspeção dos signos estáticos e dinâmicos, conforme visto na seção 6.3.1 (coleta de dados). Inicialmente, o professor era convidado a observar a tela do sistema e, em seguida, descrevia, em voz alta, qual a sua percepção sobre os signos presentes naquela interface. Nesta etapa, a percepção sobre alguns



```

Código
-----
Código:
01 QuickSort(Item *A, Indice *n) {
02  OrdenaQuicksort(1,*n,A)
03 }
04
05 OrdenaQuicksort (Indice Esq, Indice
06  Indice i, j;
07  Particao(Esq,Dir,&i,&j,A);
08  if (Esq < j) OrdenaQuicksort(Esq,j,A)
09  if (i < Dir) OrdenaQuicksort(i,Dir,A);
10 }
11
12 Particao(Indice Esq, Indice Dir, Indice
13  Item x,w;
14  *i = Esq;
15  *j = Dir;
16  x = A[( *i + *j ) / 2]; /* obtem o pivo x */
17  do {
18    while (x.Chave > A[*j].Chave) (*j)++

```

Figura 6.5. ProfesSort: Opção de Código

signos divergiu entre a análise estática e dinâmica, como o signo *código* (ver figura 6.5).

Um participante disse “*Eu achava que ele [opção código] mostrava o trecho do programa que estava sendo executado, como se fosse um depurador. Mas depois que cliquei nele, percebi que ele mostra o código todo.*”. Um professor, ao ser questionado se houve alguma mudança na interpretação dos signos após a interação, disse: “*...várias coisas foram diferentes. Algumas coisas eu fiquei surpreso, mas outras eu realmente não esperava. Por exemplo, o fato de eu clicar na opção **escolher i** e ter que selecionar o *i* no vetor. Era esperado que eu particionasse o subvetor manualmente, não imaginei que o sistema faria isso para mim. A pilha também está andando antes da interface. Só depois que eu faço a chamada recursiva é que ele deveria mostrar o elemento na pilha. Ele está dando pilha futura.* (ver figura 4.2).

Houve divergência na opinião dos professores sobre alguns signos da interface. Dentre eles, destacam-se a opção *Níveis de Ajuda* (ver figura 4.2). Alguns gostaram da opção “*Estas opções evitam que o aluno faça o algoritmo passo, usando a manipulação direta do vetor.*”. Outros disseram “*Para o aluno que quer aprender, é importante. Para o aluno que não quer, ele vai errar até aparecer a resposta.*”. O professor também era questionado sobre uma possível percepção que os alunos poderiam ter sobre tais signos.

As tarefas consistiam em cada professor ordenar um vetor com 12 (doze) posições usando os métodos *quick sort* recursivo e, posteriormente, o método *heap sort*, também com 12 posições. Por ser tratar de tarefas repetitivas a partir de um dado momento, o professor poderia fazer a ordenação parcial destes vetores ou ainda fazia uso dos recursos automáticos do ProfesSort para concluir a ordenação. Durante a ordenação, ele era convidado a explorar todos os recursos, inclusive os *scaffolds*, oferecidos pelo ProfesSort. A inspeção dos signos estático e dinâmico ocorria em paralelo, ou seja, primeiro o professor observava os signos e dizia qual era sua percepção sobre eles e, em seguida, interagiu, para verificar se sua percepção se confirmava ou não.

5. **Preparação para a análise dos dados:** seguindo os passos do MEDS, todo o vídeo ou o áudio das entrevistas foi transcrito, como forma de garantir a precisão do que foi dito por cada participante. No caso do MISI é importante conter os signos dêiticos (e.g. apontou para o elemento de interface).
6. **Análise dos dados:** foi feita a análise inter e intra-participantes da transcrição conforme descrito na seção 6.2.1.
7. **Interpretação dos resultados:** assim como a primeira aplicação do MISI no aplicativo VCalc, esta segunda avaliação também encontrou resultados referentes às categorias relevantes ao uso do sistema e a reconstrução da metamensagem sob a visão do educador foi feita.

Algumas categorias que surgiram neste estudo de caso são similares às que foram identificadas na inspeção do VCalc, porém novas categorias também apareceram. De certa forma, isso era esperado, pois apesar de serem sistemas distintos, ambos são sistemas de apoio ao aprendizado. A seguir são listadas as evidências que permitiram chegar a estas categorias.

Categorias Relevantes ao Uso do Sistema

- *Objetivo educacional:* consiste em verificar o propósito do sistema para o domínio educacional. Os professores afirmaram que o ProfesSort tem como objetivo auxiliar os alunos, com conhecimento prévio de métodos de ordenação, a consolidar a aprendizagem sobre algoritmos. Um professor enfatizou que o ProfesSort tem como foco “...ajudar na tarefa de consolidação do que foi ensinado em sala dos algoritmos de ordenação”. Segundo outro participante, “... a primeira coisa que o professor deve fazer é ensinar o algoritmo para depois usar a ferramenta”.

- *Potencial de uso da ferramenta*: refere-se à forma como o sistema poderia ser usado pelos alunos para atingir o objetivo educacional proposto. Em relação a esta categoria, a opinião dos professores foi divergente, alguns recomendam o uso extra-classe, enquanto outros defendem o uso do sistema em sala de aula. Por outro lado, todos concordaram que, primeiramente, seria necessário que o aluno tivesse uma aula teórica sobre os métodos de ordenação para depois fazer uso do sistema.

O uso extra-classe foi defendido por um professor que já conhecia o sistema e tem o hábito de recomendá-lo a seus alunos. Segundo ele, o aluno, para fazer bom uso do ProfesSort, precisaria fazer isto sozinho, de preferência em casa. Ele afirmou que, em sala de aula, o aluno faz um uso induzido da ferramenta ao ouvir a opinião de colegas sobre os próximos passos a fazer na execução de um algoritmo e, conseqüentemente, ele não tem oportunidade de refletir sobre o seu aprendizado ou sobre a própria ordenação em si.

A utilização do ProfesSort em sala de aula, na opinião de três professores, poderia complementar a aula teórica, como um recurso didático no ensino de algoritmos. Por se tratar de um recurso visual diferente, poderia despertar o interesse dos alunos pelo conteúdo abordado. Um professor vislumbrou a possibilidade do aluno, inicialmente, visualizar o método de ordenação na aula teórica e depois executá-lo no ProfesSort. Em seguida, por meio de uma atividade, o professor poderia pedir ao aluno para escrever um código que executa exatamente os passos que o aluno seguiu quando estava usando o sistema.

- *Interface*: refere-se aos aspectos visuais do sistema e a representação por ele utilizada. Os professores identificaram problemas relacionados à representação utilizada no ProfesSort que poderiam gerar um entendimento errado por parte dos alunos.

Conforme a análise de dois professores, a representação do recurso *ações a tomar* do ProfesSort (ver figura 4.2) não é adequada, pois pode levar o aluno a pensar que o conteúdo apresentado nesta parte do sistema é uma sequência. Segundo um professor, o aluno, ao observar as opções disponíveis no recurso *ações a tomar*, pode pensar que precisará executar todos os passos para chegar à opção finalizar trocas. Outro professor argumentou que o diálogo com a interface, em algumas situações, é confuso “...tem horas que estas opções são sequencias, tem horas que não. Tem momento que você clica e o ele (sistema) espera que você vai fazer a ação, tem momento que você clica

e ele faz a ação”.

Outro recurso apontado como problemático por três professores é a opção *informações da execução e estado da pilha*. Segundo eles, há muita informação na tela, o que pode atrapalhar o aluno, por não saber onde deve concentrar sua atenção.

- *Apoio ao aprendizado*: refere-se aos recursos inseridos no sistema com a finalidade de auxiliar o aluno na aprendizagem. Os principais pontos abordados nesta categoria referem-se à possibilidades de consolidação do aprendizado oferecidas por sistemas de apoio ao aprendizado e os recursos do ProfesSort que possibilitam esta consolidação.

Todos os participantes elucidaram a importância de ferramentas como o ProfesSort no apoio à aprendizagem. Segundo um professor, “...o aspecto visual é interessante para o aluno lembrar como a coisa é feita”. Nesta direção também, outro professor comentou “...tem alunos que entendem [o conteúdo] só com a aula teórica, outros alunos têm mais dificuldade e são mais visuais, precisam ver acontecendo para entrar na cabeça deles”.

Segundo um professor, os recursos de apoio à aprendizagem oferecidos pelo ProfesSort evitam que o aluno desista de usar a ferramenta. Com relação à opção *níveis de ajuda*, dois professores até brincaram, dizendo que o sistema ajuda demais ao dar a resposta correta ao aluno. Eles afirmaram que para o aluno que quer consolidar a aprendizagem, o sistema será muito útil, pois ele obriga o aluno a pensar um pouco. Para o aluno que não está preocupado em aprender, ele vai continuar errando até vir a resposta. Um professor argumentou ainda que os recursos de apoio do ProfesSort são importantes para os alunos que não gostam (ou têm vergonha) de perguntar durante a aula. Se este aluno quer aprender, ele pode executar o sistema em casa e tentar entender sozinho, com os apoios oferecidos pelo sistema.

Os professores também identificaram problemas relacionados à interface e ao conteúdo, que podem ser utilizados para o *redesign* do ProfesSort.

Com relação aos problemas de interface, dois professores apontaram que o nome das variáveis utilizadas pressupõe que o aluno decore o algoritmo visto em sala, o que segundo eles, não é adequado para a aprendizagem. Outro problema identificado refere-se à necessidade de realçar o pivô no método de ordenação *quick sort* recursivo. Segundo dois professores, este elemento poderia estar mais destacado, pelo fato do aluno fazer confusão do pivô com a posição dele no vetor.

Em relação ao conteúdo, alguns pontos foram ressaltados. Dentre eles, destacamos o recurso código e o pivô. Por ser tratar de uma ferramenta para consolidar a aprendizagem, o código do algoritmo deveria estar escrito corretamente, mas não está. *“Está faltando um ponto e vírgula no código”*, afirmou um professor. Além disso, sugeriu-se inserir comentários no código *“...em termos de aprendizagem, acrescentar os comentários é interessante, porque aqui [método quicksort recursivo] o comentário obtém pivô diz tudo. Aluno também tem a maior dificuldade de entender que tem uma coisa dentro da outra sendo executada ao mesmo tempo”*. Daí a necessidade de ter mais comentários no código.

Reconstrução da metamsagem com base na visão do participante

Dos quatro professores que participaram da avaliação do ProfesSort, um deles discordou explicitamente dos objetivos educacionais do sistema, ao enfatizar que o ProfesSort poderia ser utilizado como um exercício de fixação e não para consolidar aprendizagem, conforme os demais professores vislumbraram. Este professor entendeu que o objetivo do ProfesSort era fixar os métodos de ordenação, mas ele gostaria de ter um sistema que explorasse mais o conhecimento tácito (ou estratégico) dos alunos, que o operacional (operacional no sentido do ProfesSort ensinar os alunos o funcionamento de cada método).

Apesar desta discordância, houve consistência entre as metamsagens identificadas nos passos 1 e 2. No entanto, à medida que o professor interagiu com o sistema, sua percepção do ProfesSort tornava-se mais minuciosa. Por exemplo, os participantes, ao lerem a documentação sobre o ProfesSort, identificaram que o público alvo do sistema era alunos. Após a interação, eles complementaram: alunos de graduação que estão aprendendo algoritmos de ordenação e que já teriam tido aulas sobre os métodos.

A metamsagem consolidada do ProfesSort, na visão do educador é:

*Caro aluno, o ProfesSort é um sistema que eu projetei para você, **aluno de graduação que já aprendeu ou está aprendendo algoritmos de ordenação**. O ProfesSort vai auxiliá-lo na consolidação da aprendizagem dos métodos **quick sort (recursivo e não recursivo), heap sort, seleção e inserção**, mas para isso é necessário que antes você tenha tido uma aula sobre os métodos. Para isso, ele permite a você **executar os algoritmos passo a passo**, para que você tenha a oportunidade de compreender melhor e exercitar alguns dos algoritmos de ordenação estudados em sala, além*

de permitir a revisão e fixação do que você viu em sala aula. Para auxiliar você no aprendizado, o ProfesSort apresenta apoios para consolidar o seu aprendizado: (1) perguntas que você pode fazer ao programa quando tiver dúvidas; (2) mensagens de erro graduais, que lhe incentivarão a raciocinar; (3) visão geral de todos os passos envolvidos no algoritmo; (4) botões de auxílio que executam um ou mais passos, para que você possa simplesmente acompanhar a execução quando desejar. Você pode visualizar o código do método de ordenação que está sendo executado, mas atenção: o código contém alguns erros de sintaxe, então ele poderá ser útil para uma descrição de alto nível do algoritmo. Este sistema também oferece a opção de histórico, que é uma espécie de um log detalhado dos passos que foram executados. Assim, se você se perder, é possível visualizar os passos que foram feitos anteriormente. Você pode fazer uso deste sistema tanto em sala de aula quanto em atividades extra-classe. Se você tiver dúvidas sobre a execução dos métodos, você pode utilizar os recursos de apoio oferecidos pelo sistema para esclarecê-las. Você pode também utilizá-lo para realizar exercícios indicados pelo seu professor, informando os números a serem ordenados. No entanto, se você quiser aprender mais sobre o efeito global da ordenação ou ver o método escolhido em alto nível, este sistema não será adequado, pois ele não abstrai o baixo nível..

6.3.2.1 Outras possibilidades de uso do ProfesSort

Na análise dos dados, além das categorias verificadas como relevantes para os professores, o MISI permitiu que os professores identificassem aspectos do aprendizado que não são cobertos pelo sistema, mas que poderiam ser de interesse para a avaliação e estão em linha com a metamensagem do sistema. De forma geral, os pontos observados referem-se (i) ao potencial de uso do ProfesSort, à medida que se insira o código do método de ordenação em outras linguagens; (ii) possibilidade do aluno desfazer mais de um passo.

Os professores vislumbraram possibilidades de melhoria da ferramenta como a inserção da opção *código* em várias linguagens, pois com isso estende-se o potencial de uso do ProfesSort. Outro recurso apontado como benefício à aprendizagem refere-se à

possibilidade de permitir ao aluno desfazer mais de um passo, “...isso daria liberdade ao aluno ver a execução do algoritmo da maneira que ele quiser”, frisou um professor.

Sobre as possibilidades de apoio à aprendizagem oferecidas pelo ProfesSort, os professores disseram que são importantes, em especial para os alunos que não gostam de perguntar durante a aula. Com isso, os participantes falaram da importância que tem uma ferramenta como o ProfesSort, ou seja, se o aluno quer aprender e é tímido, ele pode executar esta ferramenta em casa e tentar entender sozinho. Sobre os níveis de ajuda, eles disseram que obriga o aluno a pensar sobre a sua ação, embora acreditem que a maioria dos alunos vai continuar testando até aparecer a resposta.

O professor que trabalha com educação de informática enfatizou a importância de se ter a presença de um professor experiente no desenvolvimento de sistemas educacionais. Segundo ele, “numa interface deste tipo é interessante fornecer uma informação a mais para evitar ou tentar forçar que ele (o aluno) erre, para mostrar: “olha, está errado por causa disso””.

6.4 Discussão da Aplicação do MISI

A diversidade de fatores envolvidos no processo de ensino e aprendizagem indica a necessidade de avaliações amplas, que possibilitem uma análise da qualidade do *software*, conforme o contexto em que ele será utilizado. Análise esta que requer o envolvimento do professor no processo de avaliação. Assim, neste capítulo apresentou-se a proposta de um novo método, o Método de Inspeção Semiótica Intermediado - MISI (neste caso, um educador), baseado no MIS e no MEDS, para analisar a visão do professor sobre a solução proposta pelos projetistas.

Por ter como base o MIS, o MIS Intermediado permite que se analise a qualidade da comunicação sendo feita através do sistema dos projetistas pelos educadores. Embora os educadores não sejam o público-alvo, esta comunicação é fundamental, pois muitas vezes cabe a eles a adoção do sistema em sala de aula, e mesmo o apoio ao uso deste fora de sala. Desta forma, esta comunicação tem um papel fundamental no sucesso do sistema. A estrutura de entrevista e análise das informações baseadas no MEDS possibilitou uma maior naturalidade na captura das impressões dos envolvidos, assim como uma maior flexibilidade e uma exploração mais aprofundada neste processo. Além disso, o MEDS permite uma análise mais rica e aprofundada dos dados. O MISI, ao combinar o MIS e MEDS, permitiu que se obtivessem indicadores tanto sobre aspectos educacionais relevantes para o educador, quanto sobre a comunicabilidade do sistema.

A avaliação feita utilizando os métodos MIS e MAC (apresentadas no capítulo 4), deixa claro que nenhum deles, individualmente, seria capaz de gerar os resultados obtidos com a aplicação do MISI, conforme pôde-se observar ao longo deste trabalho. O MIS foca na inspeção da comunicabilidade e deve ser feito por um especialista em EngSem - adotando a perspectiva do projetista-, que normalmente não tem o conhecimento do domínio necessário para levantar as questões identificadas pelos professores. O MAC tem como foco a apreciação do entendimento dos alunos, e permitiu identificar suas rupturas de comunicação, mas não questões educacionais relativas à sua adequação ao processo de ensino e aprendizagem. Desta forma, o MISI foi capaz de permitir que um novo conjunto de indicadores fosse obtido a partir da sua aplicação.

Embora a proposta e avaliação do MISI tenha sido feita para o domínio educacional, os passos do método foram propostos para serem independentes de domínio. Apenas o roteiro da entrevista foi elaborado para os estudos de caso considerando não apenas o domínio, mas os sistemas avaliados. Como este roteiro deve ser desenvolvido para cada situação ou contexto de aplicação do método, acreditamos que ele poderia ser adaptado para outros contextos. A aplicação do MISI é interessante em sistemas onde se tem um usuário indireto (*stakeholder*), como: sistemas voltados para um perfil específico, mas que interessa a outro (e.g. sistema para atividade do técnico, poderia interessar ao gerente avaliá-lo ou pode-se pensar de forma mais geral alguns em que o cliente é diferente do usuário - gerente contrata desenvolvimento de sistema para técnicos).

Nos estudos de caso realizados, o MISI permitiu não apenas uma apreciação sobre sua adequação ao uso em sala de aula, mas gerou indicadores sobre aspectos relevantes para os educadores na análise daquele sistema, e a visão de como o entendem, tanto sobre o uso que eles podem fazer do sistema, quanto sobre os benefícios que ele pode trazer aos alunos. Assim, o MISI permite a um usuário indireto avaliar a comunicabilidade de um sistema.

Capítulo 7

Considerações Finais

7.1 Contribuições

A diversidade de fatores envolvidos no processo de ensino e aprendizagem indicam a necessidade de avaliações amplas, que possibilitem uma análise da qualidade do sistema, conforme o contexto em que ele será utilizado. Esta análise requer o envolvimento do aluno, professor e especialista no processo de avaliação. Assim, este trabalho investigou a aplicabilidade dos métodos de avaliação propostos pela teoria da Engenharia Semiótica, MIS e MAC, no contexto educacional. O MIS possibilita a avaliação com foco na visão do especialista, ou seja, aspectos relativos ao projeto do sistema. O MAC permite a avaliação da comunicabilidade com a participação do aluno, o usuário final do sistema. Como nenhum dos métodos era adequado para se obter a visão do professor (quando este não é usuário, mas um *stakeholder* relevante, foi proposto um novo método, o Método de Inspeção Semiótica Intermediado (MISI). O MISI foi baseado no MIS e no MEDS e tem por objetivo analisar a visão do professor sobre a solução proposta pelos projetistas.

Neste trabalho, apresentamos uma revisão sobre os métodos existentes de avaliação para sistemas educacionais. Como não havia nenhuma informação organizada sobre o que já existe sobre métodos de avaliação para este domínio, uma classificação dos trabalhos lidos foi proposta, a fim de (i) auxiliar pesquisadores que desejam avaliar um sistema educacional e não sabem por onde começar e (ii) apresentar, de forma organizada, aos pesquisadores da área de IHC e Informática e Educação uma possível classificação sobre os métodos existentes e os critérios de avaliação considerados, a fim de evitar que estes pesquisadores tenham que fazer um levantamento bibliográfico aprofundado sobre os trabalhos que tratam de avaliação em ambientes educacionais.

O MIS é um método que permite uma visão ampla do especialista em EngSem

sobre a comunicabilidade do sistema interativo inspecionado. A aplicabilidade do MIS no domínio educacional apóia a hipótese de que sua aplicação é direta e independe de tecnologia, ao permitir avaliar a comunicabilidade de um sistema interativo qualquer. Além disso, o MIS encontrou pontos específicos, relacionados ao sistema educacional inspecionado (e.g. o objetivo educacional). Também observamos que a experiência do avaliador sobre o conteúdo sendo abordado no sistema inspecionado pode contribuir para enriquecer os resultados da sua inspeção, à medida que este conhecimento favorece a identificação de problemas relacionados ao conteúdo e contribui para a melhoria da qualidade da comunicabilidade do sistema.

O MAC tem como foco a recepção da metamsagem pelo usuário. Para o domínio educacional, além da interação usuário-sistema, é importante avaliar a aprendizagem do aluno. Como o MAC analisa a comunicabilidade do projetista para o usuário, em tempo de interação, e no caso do domínio educacional a metamsagem tem aspectos sobre interação e aprendizagem, o MAC consegue também gerar indicadores sobre esses dois aspectos. A aplicabilidade do MAC no domínio educacional ressaltou a necessidade de se fazer considerações específicas para a aplicação deste método ao domínio educacional. Essas considerações se referem à qualificação de rupturas, distinguindo aquelas que são de conteúdo das que são de interação, e à importância do avaliador ter conhecimento do domínio abordado no sistema. Este conhecimento aprofundado permite que a análise seja mais precisa, melhorando a qualidade da comunicabilidade projetista-usuário. Na etiquetagem de sistemas educacionais específicos (e.g. sistemas com *scaffolds*) pode ser necessário tratar questões específicas. A etapa de interpretação pode auxiliar na distinção da qualificação de algumas etiquetas (e.g. *Não, obrigado!*). Além disso, ainda na interpretação, foi possível obter informações sobre indicadores da parte educacional, fato importante observado sobre a aplicação do MAC no domínio educacional. Com isso, além de obter informações sobre a interação, ele permite coletar indicadores sobre o aprendizado do aluno. Para a geração do perfil semiótico, as principais considerações a serem feitas referem-se à identificação do objetivo educacional do sistema avaliado.

A aplicação do MAC no Bipide apontou uma limitação deste método para um tipo específico de sistema educacional, aqueles que trabalham com visualização e simulação. Para o MAC analisar a recepção da metamsagem pelo usuário, é necessário que ele interaja com o sistema para que se identifique a comunicação. Em sistemas de visualização/simulação o aluno não interage com o *software*, ele geralmente fica apenas observando o que lhe é apresentado na interface. Assim, para este tipo específico de sistema, o MAC não encontra rupturas de interação.

Embora a limitação identificada se refira à aplicação do MAC no domínio educa-

cional, para sistemas de simulação em geral poder-se-ia ter o mesmo problema. Assim, os resultados da aplicação do MAC no domínio educacional foram positivos, em especial para a EngSem, por oferecer um método que pode ser aplicado em qualquer domínio, sem necessidade de maiores considerações. As considerações a serem feitas são pequenas no método e tem um baixo custo para o avaliador fazê-las, entretanto elas são relevantes e possibilitam uma avaliação adequada para o domínio educacional, ao permitir a geração de indicadores sobre a interação e sobre a parte educacional.

Diferente do MIS, que permite a avaliação da comunicabilidade na visão de um especialista em EngSem, ao utilizar o MAC o especialista é guiado pelo que os usuários fazem durante a interação com o sistema. Assim, a análise se refere às evidências coletadas durante a interação do usuário com o sistema, isto é, as falhas na recepção de metacomunicação. Com isso, o MAC se concentra em uma classe particular de fenômenos observáveis no sistema de interação do usuário, ou seja, que se referem a falhas de comunicação, o que restringe o âmbito da análise ainda mais [de Souza & Leitão, 2009]. Entretanto, é válido considerar a experiência real do usuário ao interagir com um sistema, que permite ao avaliador explorar aspectos da metacomunicação diferentes daqueles que teriam ocorrido durante a inspeção com o MIS, por exemplo. Ambos os métodos, MIS e MAC, podem ser complementares ou podem ser selecionados para fins específicos, conforme o que se deseja analisar.

Por ter como base o MIS, o MISI permite que se analise a qualidade da comunicação sendo feita através do sistema dos projetistas para os educadores. Embora os educadores não sejam o público-alvo, esta comunicação é fundamental, uma vez que cabe a eles, muitas vezes, a adoção do sistema em sala de aula, e mesmo o apoio ao uso deste fora de sala [Squires & Preece, 1999]. Desta forma, esta comunicação tem um papel fundamental no sucesso do sistema. A estrutura de entrevista e análise das informações baseadas no MEDS possibilitou uma maior naturalidade na captura das impressões dos envolvidos, assim como uma maior flexibilidade e uma exploração mais aprofundada neste processo. Além disso, o MEDS permite uma análise mais rica e aprofundada dos dados.

Ao combinar os dois métodos, o MISI permite uma sinergia entre as habilidades e competências do especialista em EngSem e do especialista do domínio. Desta forma, o método permitiu que se obtivessem indicadores tanto sobre aspectos educacionais relevantes para o educador, quanto sobre a comunicabilidade do sistema. o MISI foi proposto para o domínio educacional - específico para a avaliação do professor, mas como o roteiro e passos de interação são definidos no contexto da avaliação, acreditamos que ele se aplicaria para avaliações com *stakeholders* em outros contextos (que não o educacional). Além disso, dado que a avaliação do Bipide mostrou que o MAC não

Referência	Método de origem	Tipo de avaliação	Tipo de sistema	Características avaliadas	Participantes	Coleta de dados	Ambiente para realização do teste	Foco dos resultados
Prates et al. 2000	MAC	Qualitativa e Interpretativa	Qualquer tipo	Comunicabilidade	Aluno	Teste com usuário	Ambiente controlado	Rupturas de interação e conteúdo
De Souza et al. 2006	MIS	Qualitativa e Interpretativa	Qualquer tipo	Comunicabilidade	Especialista	Inspeção	Inspeção	Comunicabilidade
De Oliveira et al. 2008	MISI	Qualitativa e Interpretativa	Envolvem usuários indiretos	Comunicabilidade	Usuário indireto (Professor)	Entrevista baseada na interação OU Entrevista com etapa de interação	Ambiente controlado	Comunicabilidade e aspectos educacionais

Figura 7.1. Tabela de Classificação dos Métodos da Engenharia Semiótica

era muito adequado, pela baixa interação do aluno com sistema, talvez o MISI seja adequado também para situações como esta.

Ao considerar a classificação dos métodos de avaliação proposta no capítulo 2, é importante classificar os métodos da EngSem para que eles também possam ser incluídos na tabela de classificação, conforme pode-se observar na figura 7.1. Assim, temos pontos comuns para os três métodos: a avaliação é qualitativa e interpretativa, a característica avaliada é a comunicabilidade. Os pontos específicos consistem em:

- *MAC*: tem o aluno como participante, permite a avaliação de qualquer tipo de sistema, independente do domínio, a coleta de dados se dá em ambiente controlado, e o foco dos resultados é nas rupturas de interação e conteúdo.
- *MIS*: tem o especialista em EngSem como participante, permite a avaliação de qualquer tipo de sistema, a coleta de dados é por inspeção e o foco dos resultados é na comunicabilidade.
- *MISI*: tem como foco um usuário indireto, no caso do domínio educacional, o professor. Permite a avaliação de um sistema que tenha usuários indiretos, a coleta de dados é baseada em uma entrevista semi-estruturada e o foco dos resultados é na comunicabilidade sob o ponto de vista do usuário indireto e também nos aspectos educacionais.

7.2 Trabalhos Futuros

Foi feita uma busca ampla para se identificar artigos sobre avaliação de ambientes educacionais ou que apresentassem estudos de caso sobre este domínio. A partir deste

levantamento, foi realizada, no capítulo 2, o esboço de uma classificação desses métodos de avaliação de sistemas educacionais. Esta classificação demonstrou a variedade de métodos e critérios que têm sido propostos e/ou adaptados para avaliar o domínio educacional. Porém, talvez um levantamento maior da área, não apenas buscando artigos diretamente relacionados à avaliação, pudesse ser interessante para ampliar a visão dos trabalhos que existem atualmente na área. Para isso, seria interessante uma varredura intencional nos artigos que apresentam além de discussão sobre avaliação, sobre projeto e implementação, nos quais discussões sobre a avaliação pode, estar sendo feitas no contexto do artigo.

Na investigação sobre a aplicabilidade dos métodos de EngSem fizemos aplicações técnicas dos métodos. Aqui apresentamos apenas os problemas identificados relevantes para discutir a aplicabilidade destes métodos - e não detalhamos um relatório sobre a qualidade da interação de cada sistema. Sobre isso, seria interessante fazer um relatório mais detalhado, relativo ao que foi encontrado com cada método sobre os sistemas. Como os mesmos sistemas foram avaliados com diferentes métodos, contrastar os indicadores obtidos com cada um. Isso permitiria uma discussão mais aprofundada do custo *versus* benefício de cada um deles.

Não fez parte do escopo da dissertação fazer uma aplicação científica dos métodos propostos pela Engenharia Semiótica. No entanto, acredita-se que eles poderiam ser utilizados para geração de conhecimento específico a IHC em domínios educacionais (assim, como já o são para IHC). Além disso, o MISI também poderia ser útil - se não na geração de novo conhecimento, para triangulação. A proposta do MISI foi baseada no domínio educacional. No entanto, poderia-se realizar estudos de caso em outros domínios (e.g. sistemas que tenham usuários indiretos) a fim de verificar a possibilidade de aplicação deste método, bem como comparar os seus resultados com outros métodos de avaliação de comunicabilidade.

Embora tenhamos feito estudos de casos em diferentes domínios, não cobrimos a variedade de propósitos de sistemas educacionais. Assim, novas aplicações - já com as considerações feitas para outros tipos de sistemas educacionais seria necessária para uma avaliação mais ampla das propostas feitas aqui.

Por fim, a aplicação do MIS, MAC e MISI nos estudos de caso realizados permitiu analisar a comunicabilidade sob o ponto de vista do especialista, aluno e professor. No entanto, um contraste dos resultados da avaliação usando esses três métodos por sistema pode trazer contribuições relevantes ao se fazer uma apreciação da natureza dos problemas identificados por cada um deles, e gerando conhecimento que auxilie nas decisões sobre avaliações de ambientes educacionais.

Anexo A

Material de Inspeção do MIS no VCalc

Roteiro para avaliação do sistema VCalc Usando o Método de Inspeção Semiótica (MIS)

- 1) Objetivo:** analisar a metacomunicação do designer para o usuário do sistema de apoio à aprendizagem VCalc.
 - Reconstrução das intenções de design do VCalc, em particular das estratégias de comunicação de suas funções básicas.
 - Identificação dos pontos de vista dos especialistas sobre as principais dificuldades e facilidades envolvidas na interface do VCalc.

- 2) Participantes:** 3 especialistas em IHC e EngSem

Cenário

No Departamento de Matemática da UFMG vários professores têm utilizado softwares para dar aula de matemática. Em particular, o Grupo de Estudos e Pesquisa em Educação Matemática e Novas Tecnologias (GEPEMNT) têm desenvolvido aplicativos voltados para o ensino de Cálculo Diferencial e Integral e testado esses novos recursos nos processos de ensino-aprendizagem (aulas e oficinas) elaborados pelo grupo. Porém, até o momento, não foi feita nenhuma avaliação de Usabilidade nestes aplicativos e por isso o grupo solicitou os serviços de especialistas na área de IHC.

Em um primeiro momento foi pedido a avaliação do aplicativo VCalc. O VCalc tem sido utilizado no ensino de **taxa de variação** e de **derivada**, em aulas de caráter investigativo. Nestas aulas os alunos, diante do software, realizam atividades de investigação de conceitos da matemática, através de manipulações gráficas, zoom e outros recursos, tais como: plote a função $x \cdot \cos(x)$ e mostre, utilizando o *applet* VCalc, qual será, aproximadamente, a derivada desta função no ponto $x = -3.43$.

Passos para a realização do MIS:

1º Passo) Meta-mensagem do Designer através do Help e da Documentação

- a) Segundo o Help
- b) Segundo a Documentação (Site, PDFs, etc)

2º Passo) Meta-mensagem do Designer através dos signos estáticos da interface.

Signos estáticos são aqueles que expressam o estado do sistema. Normalmente podem ser captados apenas olhando-se para a interface ou mesmo uma imagem desta.

3º Passo) Meta-mensagem do Designer através dos signos dinâmicos da interface

Signos dinâmicos expressam o comportamento do sistema, e só podem ser percebidos quando o usuário interage com o sistema.

4º Passo) Contrastar meta-mensagens

5º Passo) Apreciando a qualidade

Anexo B

Material de Aplicação do MAC no VCalc

TERMO DE CONSENTIMENTO DE PARTICIPAÇÃO

Título: Avaliação da Comunicabilidade do VCalc (Visual Calculus)

Data: Novembro/2007

Instituição: DCC / UFMG

Pesquisadores Responsáveis:

Prof. Raquel O. Prates (rprates@dcc.ufmg.br)

Erica Oliveira (erica@dcc.ufmg.br)

Luiz Luz (luizluz@dcc.ufmg.br)

Introdução: Este Termo de Consentimento contém informações sobre a pesquisa indicada acima. Para assegurar que você esteja informado sobre a sua participação nesta pesquisa, pedimos que leia este Termo de Consentimento. Caso tenha alguma dúvida, não hesite em perguntar ao pesquisador responsável. Você também deverá assinar o termo do qual receberá uma cópia.

Objetivo da avaliação: O objetivo desta avaliação é identificar, na interface do VCalc – sistema de apoio ao ensino de Calculo Diferencial –, problemas de comunicabilidade que possam dificultar o seu uso pelo aluno.

Informação geral sobre a pesquisa: Você será solicitado a realizar algumas tarefas simples utilizando o sistema. A realização dessas tarefas será gravada para posterior análise pelos investigadores. Ao fim da execução das tarefas, será realizada uma entrevista sobre sua experiência com o sistema.

Utilização dos dados coletados: Os dados coletados durante a avaliação serão utilizados para estudo do Método de Avaliação da Comunicabilidade (MAC). Quaisquer dados utilizados para publicação serão apresentados de forma a garantir o anonimato dos participantes da avaliação.

Privacidade: Informações que possam identificar os participantes da pesquisa não serão divulgadas. O seu nome não aparecerá em nenhum relatório. Caso deseje, poderá solicitar uma cópia dos dados gerados por você.

Se você decidir não participar na pesquisa: Você é livre para decidir, a qualquer momento, se quer participar ou não nesta pesquisa. Sua decisão não afetará sua vida estudantil e nem qualquer relacionamento com os avaliadores, professores ou a Instituição por trás desta.

Compensação: A participação nesta pesquisa é voluntária, e não será oferecida nenhuma remuneração aos seus participantes.

Se tiver algum problema ou se tiver outras perguntas: Se você tiver algum problema que pensa que pode estar relacionado com sua participação nesta pesquisa, ou se tiver qualquer pergunta sobre a pesquisa, poderá entrar em contato com os pesquisadores a qualquer momento pelo e-mail luizluz@gmail.com ou pelo telefone 9334-9006

Novas condições: Caso deseje, você pode especificar novas condições que devem ser atendidas para que você participe desta avaliação.

Consentimento Livre e Esclarecido (Acordo Voluntário)

O documento mencionado acima descrevendo os benefícios, riscos e procedimentos da pesquisa Avaliação da Comunicabilidade do Sistema VCalc foi lido e explicado. Eu tive a oportunidade de fazer perguntas sobre a pesquisa, que foram respondidas satisfatoriamente. Eu estou de acordo em participar como voluntário.

_____ Assinatura do participante: _____

Data Nome do participante: _____

Assinatura do pesquisador: _____

Nome da pesquisadora: _____

Questionário Pré Teste

Data: ____/11/2007

Identificador: _____

Nome: _____

Sexo: () Feminino () Masculino

Idade: ____

Por favor, preencha os campos com a opção que melhor representa a sua resposta a cada pergunta.

1: Qual curso está fazendo?

() Ciência da Computação

() Sistemas de Informação

() Matemática Computacional

() Engenharia Elétrica

() Engenharia de Controle e Automação

() Outro. Qual? _____

2: Em que ano e semestre ingressou no curso: _____

3: Você já estudou taxas de variação?

() Sim

() Não

4: Você já estudou derivada?

() Sim

() Não

5: Como você classifica o seu nível de conhecimento em Cálculo I?

() Não tenho conhecimento em Cálculo I

() Já estudei mas não me lembro de nada.

() Já estudei e me lembro de quase tudo.

() Tenho total domínio do conteúdo.

Tarefa a ser executada

Você é aluno do curso de Cálculo I e está estudando alguns conceitos de derivada. Para facilitar o entendimento de alguns conceitos, o professor resolveu dar aula utilizando o computador. O professor indicou o endereço de um site e, ao abri-lo, foi executado o *applet* VCalc. Utilizando este *applet* o professor solicitou a realização da seguinte atividade:

Atividade

Plote a função $x \cdot \cos(x)$ e mostre, utilizando o *applet* VCalc, qual será, aproximadamente, a derivada desta função no ponto $x = -3.43$.

Em seguida, verifique qual é a taxa de variação desta função no ponto 0 (zero).

Roteiro para Entrevista Pós-Teste

1. Experiência com softwares educacionais e de visualização; Cálculo I ou outras disciplinas.
2. Pontos positivos do VCalc.
3. Pontos negativos do VCalc.
4. Taxa de Variação: utilizou ou não, se sim: achou útil; se não: por que não.
5. Algum outro comentário ou sugestão.

Formulário de Acompanhamento do Teste

Pesquisa: Avaliação da Comunicabilidade do VCalc (Visual Calculus)

Avaliador: _____ **Data do Teste:** ____/____/____

Identificação do Participante: _____

Tarefa 1:

Plote a função $x \cdot \cos(x)$ e mostre, utilizando o applet VCalc, qual será, aproximadamente, a derivada desta função no ponto $x = -3.43$.

Em seguida, verifique qual é a taxa de variação desta função no ponto 0.

Observações:

Perguntas para a Entrevista:

Terminou: () Sim () Não

Sobre o recurso de Taxa de Variação:

Utilizou: () Sim () Não

O Encontrou: () Sim () Não

Anexo C

MAC: Relatório da Etiquetagem do VCalc

VCalc: Aplicação do Método de Avaliação de Comunicabilidade

PARTICIPANTE 1

Etiqueta	Tempo	Interação	Conteúdo	Explicação
Cadê?	0:10	X		Usuário procura o local onde deverá informar a função. [Ao iniciar o contato com o software não seria necessário criar uma etiqueta do tipo “Deixa eu ver o que tem aqui.” No caso desta etiqueta, o usuário estava ‘observando’ o software.]
Cadê?	0:52	X		O usuário estava procurando alguma maneira de mudar o ponto vermelho.
Ué, o que houve?	1:06	X		Usuário utilizou o campo do zoom para especificar o valor X do ponto vermelho.
Epa!	1:43	X		O usuário clicou na “setinha” da direita e, ao disparar a animação, ele clicou na outra setinha para desfazer o que ele fez.
Cadê?	1:57	X		O usuário estava procurando alguma maneira de definir com precisão o valor X do Ponto Vermelho.
Ué, o que houve? Seguido de Vai de outro jeito!	2:18	X		O usuário tenta mover o ponto vermelho, clicando sobre ele e não obtém resposta. Então ele parte para a utilização de outra ferramenta.
Epa!	2:26	X		O usuário aciona esses botões (↔ e ⇄) tentando mover o ponto vermelho.
Epa!	2:36	X		O usuário tenta mudar o ponto vermelho de lugar usando a função Nova Área.
Epa!	3:25	X		O usuário selecionou a opção Taxa de variação para informar o valor do delta x.

Observações:

No questionário inicial, o usuário respondeu que possuía bons conhecimentos em Cálculo e que se lembrava de quase tudo. Essa afirmação pode ser confirmada por meio da utilização do software, pois ele não teve nenhum problema de conteúdo e os problemas que ocorreram durante a interação devem-se mais à interface do sistema. Ele concluiu a tarefa com êxito, conseguindo visualizar a taxa de variação e a derivada no ponto solicitado.

PARTICIPANTE 2

Etiqueta	Tempo	Interação	Conteúdo	Explicação
E agora?	1:49	X		<p>Usuário insere a função e o valor de x0 e x1. Porém, insere o valor de x1 errado. Resolve então deletar o valor, mas parece ficar sem saber qual valor deveria informar.</p> <p>[Ao iniciar o contato com o software não seria necessário criar uma etiqueta do tipo “Deixa ver o que tem aqui.” No caso desta etiqueta, o usuário estava ‘observando’ o software.].</p>
Socorro!	1:53		X	O aluno não sabia qual valor de X1 ele deveria colocar.
E agora?	2:00 a 3:50	X	X	<p>Usuário resolve deixar o valor de x1 em branco. A partir disso, começa a fazer uma experimentação pelo sistema, clicando em todos os botões e tentando identificar o que cada botão faz.</p> <p>Na entrevista pós-teste, quando questionado sobre essa atitude, o usuário disse que estava perdido, além de não saber o que deveria fazer.</p>
E agora?	2:03	X	X	O usuário não sabe o que fazer e clica na opção Y0, mas não altera o valor existente.
E agora? Epa!	2:08	X	X	Usuário clica na opção de zoom (0.5x) e não sabe o que fazer.
Ué, o que houve? O que é isto?	2:13	X	X	O usuário não sabe o que fazer e está explorando o software.
Ué, o que houve?	2:15	X	X	O usuário não sabe o que fazer e está procurando por algo, para isso clica em todos os botões da interface.
Ué, o que houve?	2:17	X	X	O usuário não sabe o que fazer e continua procurando por algo.
E agora?	2:20	X	X	O usuário continua perdido...
Epa!	2:25	X	X	Usuário, sem saber o que fazer, clica na opção A função, mas não faz nada nesta

				janela.
				Usuário está perdido, sem saber o que fazer. Além de ter problemas com a interface, demonstra não ter domínio do conteúdo.
Epa!	3:02	X		Usuário não entende como funcionam as setas para movimentação do ponto vermelho (<< e >>)
Por que não funciona?	3:30	X		O usuário tenta andar com o ponto uma unidade e não consegue, pois é disparada uma animação.
Socorro! Seguido de Epa!	3:42	X		Usuário não consegue colocar o ponto no local onde deveria.
Epa!	3:43	X		Usuário selecionou uma opção indesejada, quando estava tentando colocar o ponto na posição solicitada pela tarefa (opção INFO).
Por que não funciona? O que é isto?	3:46	X		Usuário clica na opção ESC e nada acontece. O usuário não sabe o que fazer e resolve explorar o software.
O que é isto?	3:47	X		O usuário não sabe o que fazer e esta explorando o software (clica nas opção 1 x 1).
				De 3:50 até 6:00 o software deu problema e o usuário ficou mais perdido ainda.
				Em 6:03 o software foi reiniciado. E o teste começou novamente...
Para mim, está bom!	6:22		X	Usuário novamente não informa o valor de X1.
	6:36		X	Análise Luiz Luz: O usuário não sabe o significado do termo "Delta X". Porque ele confundiu o termo duas vezes com o valor X do Ponto Vermelho.
Para mim, está bom!	6:45		x	Usuário colocou um valor inválido e o software travou novamente. Usuário não informou o valor de x, apenas o delta X (e

				errado).
			x	Ao reiniciar a tarefa o aluno cometeu os mesmos erros e apresentou as mesmas dúvidas.
Socorro! Seguido de Epa!	6:47		X	Usuário não sabe o que fazer e por isso entra em loop, abrindo o comando da função e clicando em confirmar repetidas vezes.
E agora? Seguido de Epa!	8:18		X	Usuário não entendeu o que era os objetos X0 e X1 da janela da Função. Ele fica uns 20 segundos vagando com o mouse pela tela de Nova Função.
Socorro!	8:39		X	Após sair da tela Nova Função, usuário fica vagando pela interface principal. Logo em seguida, retorna à janela de Nova Função.
Socorro!	8:42		X	Na janela Nova Função, ele fica vagando com o mouse sem saber o que fazer.
Ué, o que houve?	8:57		X	Usuário não entende o que ocorreu na tela.
Epa! Seguido de Ué, o que houve?	9:13		X	Usuário continua sem entender o que houve e continua clicando nas setas de movimentação do ponto (<< e >>).
Pq não funciona?	9:50	X		Usuário queria calcular a taxa de variação no ponto e clicou no Enter...
Cadê?	9:59	X		Usuário está procurando alguma maneira de calcular a taxa de variação no ponto.
Epa!	10:06	X		Usuário continua procurando uma forma de calcular a taxa de variação.
Socorro!	10:25	X	X	Usuário não sabe o que fazer para calcular a taxa de variação.
Epa!	11:13	X		Usuário não sabe o que fazer para calcular a taxa de variação e clica na opção Nova Área.
E agora?	11:18	X		O usuário não sabe o que faz e clicou no botão 1x1.
Pq não funciona?	11:20	X		Usuário clica no botão ESC 3x.
Para mim, está bom!	12:13		X	Usuário diz que concluiu a tarefa.

PARTICIPANTE 3

Etiqueta	Tempo	Interação	Conteúdo	Explicação
Porque não funciona?	0:31	X		O usuário estava com a tecla Caps lock ligada, por isso o software não entendeu a variável X e a função COS.
Ué, o que houve?	0:33	X		O software exibiu a função nula, $f(x) = 0$, e o usuário não entendeu o porquê, pois ele colocou a função $X * \text{COS}(X)$.
E agora?	0:40 a 1:02	X		Usuário não sabia o que fazer, pois a função estava correta. Por volta de 1:03 o usuário desligou a tecla Caps Lock e digitou novamente.
Por que não funciona?	1:53 a 3:12	X		Ao informar o valor do x, o usuário utilizou uma vírgula ao invés de ponto e por isso o software não respondeu ao clique do mouse.
Para mim, está bom.	4:03		X	O usuário ficou satisfeito com o resultado da Taxa de Variação, mesmo com um Delta X grande.

Observações

O usuário terminou a tarefa sem se preocupar com a redução do Delta X, para obter uma taxa de variação mais precisa. Na entrevista pós-teste, foi possível verificar que o usuário não percebeu os botões de + e - e também não percebeu a reta secante.

PARTICIPANTE 4

Etiqueta	Tempo	Interação	Conteúdo	Explicação
O que é isto?	Até 0:42	X		Usuário está tentando identificar os objetos presentes na interface.
Pq não funciona?	0:49	X	X	Usuário insere o valor do delta x no lugar errado e não entende pq não acontece nada.
Epa!	1:26		X	Usuário clica e abre uma janela q ele não esperava. Então ele cancela a operação.
Ué, o que houve?	2:18	X	X	Ao inserir a função, o sistema altera o valor do X0. O usuário não concorda e altera

				novamente para o valor do delta X.
Epa!	2:50	X		Usuário usa vírgula ao invés de . e o sistema dá mensagem de erro.
	2:55			A partir do instante anterior (2:55) o sistema travou por causa da vírgula.
	3:44			O sistema foi reiniciado.
Epa!	4:10	X		Sistema travou novamente porque o usuário inseriu uma vírgula no valor do X0.
Ué, o que houve?	4:42		X	O usuário colocou um intervalo de X que não continha o ponto vermelho e tanto a ferramenta quanto o ponto vermelho desapareceram.
E agora?	4:50		X	Usuário não sabe o que fazer, fica vagando com o mouse pela interface.
Desisto!	5:15		X	Usuário desiste de prosseguir com a tarefa 1. Não entendeu os elementos da interface, mas também demonstrou que tinha dificuldade de conteúdo, apesar de responder no questionário que se lembrava de quase tudo da disciplina de Cálculo 1.
	6:29			Usuário reiniciou o sistema para fazer a tarefa 2.
Epa!	6:45	X		Usuário seleciona a opção Função. Entra nesta janela, seleciona a função existente e depois cancela a operação.
Pq não funciona?	7:15	X		Usuário usa os botões de zoom, mas não percebe a mudança que ocorre na janela principal.
Desisto	8:40	X	X	Usuário desiste também da 2ª tarefa.

PARTICIPANTE 5

Etiqueta	Tempo	Interação	Conteúdo	Explicação
O que é isto?	0:09	X		Usuário estava explorando o software.
E agora?	0:35		X	O usuário clica no campo X0, não faz nenhuma alteração e depois de aproximadamente 10 segundos ele clica em Confirmar.

O que é isto?	0:54 a 1:02	X		O usuário estava tentando mudar o ponto vermelho de lugar e para isso, passava o botão do mouse sobre os elementos para ver o que faziam. Penso que ele procurava por hints.
Epa!	1:06	X		O usuário clica na setinha para fazer o botão andar discretamente e é disparada uma animação.
Por que não funciona?	1:10	X		O usuário clica sob a curva com a intenção de mover o ponto para aquele local clicado.
Epa!	1:30	X		Usuário tenta mover o ponto vermelho para o valor solicitado pela tarefa (3,43), mas tem dificuldade por causa da animação disparada na interface.
E agora?	1:35	X		Sem conseguir mover o ponto para o local desejado, ele fica vagando pela interface.
O que é isto?	1:45	X		O usuário estava tentando especificar mais precisamente a posição X do ponto vermelho (clicou na opção Nova Área).
E agora?	1:50	X		O usuário clica no botão Enter tentando calcular a taxa de variação.
O que é isto?	1:55	X		O usuário procura por informações ou meios de calcular a taxa de variação.
Cadê?	2:05	X		O usuário procura por meios de especificar o valor X do ponto vermelho.
Por que não funciona?	2:08	X		O usuário tenta mover o ponto vermelho clicando sobre ele e o arrastando. Porém, não acontece nada.
E agora?	2:10	X		O usuário clica no ponto vermelho tentando ver se apareceria alguma informação sobre a taxa de variação
Epa!	2:19	X		O usuário clica nestes botões para mover o ponto. Porém, o que acontece é a translação do eixo Y.
	2:28			Usuário desiste de colocar o ponto vermelho na posição -3.43 (solicitado na tarefa) e parte para a segunda atividade que é a definição da taxa de variação da função no ponto

				indicado. E, ao clicar no botão de taxa de variação, ele encontra o local para especificar o ponto em que seria calculada a taxa de variação.
O que é isto?	2:42	X		O usuário procura identificar a informação oferecida por este objeto (delta x).
Por que não funciona?	3:00	X		O usuário tenta aproximar o ponto preto do ponto vermelho clicando nele e o arrastando para perto do Ponto Vermelho (PV). Sua intenção era de calcular a taxa de variação com uma maior precisão.
Ué, o que houve?	3:05	X		Usuário clica na opção Taxa de variação e a janela desaparece. Em seguida, ele clica novamente e a janela reaparece na tela.
Ué, o que houve? Seguido de um Vai de outro jeito!	3:14	X		O usuário coloca um Delta X igual a 0 e a reta secante desaparece. Como ele não consegue identificar visualmente a taxa de variação ele procura outra forma de exibi-la.
	4:05 a 4:30			Usuário faz leitura da primeira tarefa, novamente.
	4:58 a 5:30			Usuário faz leitura da segunda tarefa.
Por que não funciona?	5:34	X		O usuário tenta mover o ponto vermelho clicando sobre ele e o arrastando. Porém, não acontece nada.
Por que não funciona?	6:20	X		O usuário queria cancelar a sucessão de zooms que ele deu e clicou no botão ESC.
Pq não funciona? Seguido de Vai de outro jeito!	7:10	X		Usuário selecionou a opção 0.5x para voltar ao zoom q estava antes e nada aconteceu. Em seguida, ele digitou o valor que estava antes.

Anexo D

Material de Aplicação do MAC no ProfesSort

Termo de Consentimento de Participação

Título: Avaliação da Comunicabilidade do ProfesSort Versão Beta1.4

Data: Maio/2007

Instituição: Grupo de Pesquisa em Interação Humano Computador / DCC / UFMG

Pesquisadores Responsáveis:

Prof. Raquel O. Prates (rprates@dcc.ufmg.br)

Gisleide Aidano (gisleide@dcc.ufmg.br)

Juliana Faleiro (julianaf@dcc.ufmg.br)

Introdução: Este Termo de Consentimento contém informações sobre a pesquisa indicada acima. Para assegurar que você esteja informado sobre a sua participação nesta pesquisa, pedimos que leia este Termo de Consentimento. Caso tenha alguma dúvida, não hesite em perguntar ao pesquisador responsável. Você também deverá assinar o termo do qual receberá uma cópia.

Objetivo da avaliação: O objetivo desta avaliação é identificar na interface do ProfesSort - sistema de apoio ao aprendizado do método Quicksort - problemas de comunicabilidade que possam dificultar o seu uso pelo aluno.

Informação geral sobre a Pesquisa: Você será solicitado a realizar algumas tarefas simples utilizando o sistema. A realização dessas tarefas será gravada para posterior análise pelos investigadores. Ao fim da execução das tarefas, será realizada uma entrevista sobre sua experiência com o sistema.

Utilização dos dados coletados: Os dados coletados durante a avaliação serão utilizados para melhorias no desenvolvimento do sistema ProfesSort e/ou para pesquisa de tópicos relacionados com o sistema ou método de avaliação utilizado. Quaisquer dados utilizados para publicação serão apresentados de forma a garantir o anonimato dos participantes da avaliação.

Privacidade: Informações que possam identificar os participantes da pesquisa, não serão divulgadas. O seu nome não aparecerá em nenhum relatório. Caso deseje, poderá solicitar uma cópia dos dados gerados por você.

Se Você Decidir Não Participar na Pesquisa: Você é livre para decidir a qualquer momento se quer participar ou não nesta pesquisa. Sua decisão não afetará sua vida estudantil e nem qualquer relacionamento com os avaliadores, professores ou a Instituição por trás desta.

Compensação: A participação nesta pesquisa é voluntária, e não será oferecida nenhuma remuneração aos seus participantes.

Se tiver algum problema ou se tiver outras perguntas: Se você tiver algum problema que pensa que pode estar relacionado com sua participação nesta pesquisa, ou se tiver qualquer pergunta sobre a pesquisa, poderá entrar em contato com os pesquisadores a qualquer momento pelo e-mail gisleide@dcc.ufmg.br ou pelo telefone 3499-5860

Novas condições: Caso deseje, você pode especificar novas condições que devem ser atendidas para que você participe desta avaliação.

--

Consentimento Livre e Esclarecido (Acordo Voluntário)

O documento mencionado acima descrevendo os benefícios, riscos e procedimentos da pesquisa Avaliação da Comunicabilidade do Sistema de ensino do método Quicksort foi lido e explicado. Eu tive a oportunidade de fazer perguntas sobre a pesquisa, que foram respondidas satisfatoriamente. Eu estou de acordo em participar como voluntário.

	Assinatura do participante: _____
Data	Nome do participante: _____

Assinatura do pesquisador: _____
Nome da pesquisadora: _____

Questionário Pré Teste

Data: ____/05/2007

Identificador: _____

Nome: _____

Sexo: () Feminino () Masculino

Idade: ____

Por favor, preencha os campos com a opção que melhor representa a sua resposta a cada pergunta.

1: Qual curso está fazendo?

() Ciência da Computação

() Sistemas de Informação

() Matemática Computacional

() Engenharia Elétrica

() Engenharia de Controle e Automação

() Outro. Qual? _____

2: Em que ano e semestre ingressou no curso: _____

3: Em que ano e semestre está prevista sua formatura: _____

4: Você conhece o método de ordenação denominado Quicksort?

() Sim

() Não

5: Esta é a primeira vez que você está estudando o Quicksort?

() Sim.

() Não, já tinha feito AEDS II anteriormente . Esta é a _____ vez que faço a disciplina

() Não, já tinha estudado em outro curso ou disciplina.

Qual(is)? _____ - _____

Em que instituição? _____

6: Em sala vocês viram quais métodos ? (marque todas as alternativas de métodos estudados)

() Quicksort recursivo

() Quicksort não recursivo

() Não sei

Vire a folha

7: Como você adquiriu o conhecimento sobre o método? (marque todas as alternativas que julgar adequadas à resposta)

- Não tenho conhecimento sobre o método
- Assisti aulas expositivas sobre o método
- Li o material didático indicado pelo professor sobre o algoritmo
- Li material didático conseguido na Internet ou indicado por colegas sobre o algoritmo.
- Usei sistemas de visualização do algoritmo
- Simulei manualmente a aplicação do algoritmo

8: Quanto tempo vc já dedicou no estudo deste método, após as aulas?

- 1 hora ou menos
- 2 a 3 horas
- 4 a 6 horas
- Mais de 6 horas

9: Como você classifica o seu nível de conhecimento sobre o método?

- Não tenho conhecimento sobre o método
- Conheço a teoria
- Conheço bem a teoria, mas não entendo o funcionamento na prática.
- Conheço bem a teoria, mas não consigo aplicar o método em alguns casos
- Conheço bem a teoria e consigo aplicar o método manualmente

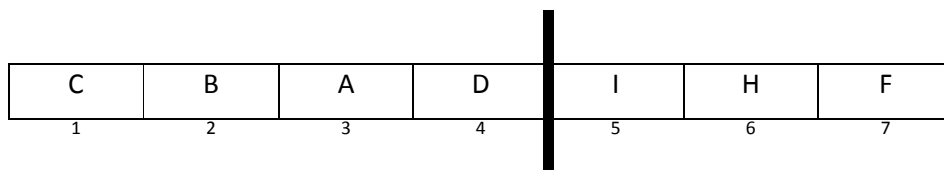
Texto de apresentação do sistema ProfesSort

O ProfesSort é um software educacional que tem como objetivo ajudar no aprendizado de métodos de ordenação. Nesta versão especificamente avaliaremos o apoio ao aprendizado do método Quicksort não recursivo. O sistema foi desenvolvido no DCC/UFMG para alunos da matéria de Algoritmos e Estruturas de Dados II (AEDsII) a fim de consolidar o método visto em sala, de forma eficaz e agradável aos alunos.

O ProfesSort permite que o aluno resolva um problema através do método Quicksort. Para isso o aluno deve dizer qual é o passo do método a ser executado e tomar as decisões relativas a ele. O sistema disponibiliza ao aluno diferentes formas de apoio para que ele execute a tarefa. As principais são:

- menu de perguntas que correspondem a potenciais dúvidas que o aluno possa ter e suas respectivas respostas
- garantia de execução de cada passo corretamente: o sistema não permite que o aluno prossiga sem que o passo sendo executado esteja correto, e em caso de erro incentiva o aluno a pensar e corrigi-lo.

No Quicksort não recursivo, o algoritmo implementa uma pilha na qual armazena os índices do intervalo que será resolvido posteriormente. Por exemplo:



Neste caso, o algoritmo resolveria o subvetor de índice de 1 a 4 e empilharia o subvetor de índice de 5 a 7 para resolver posteriormente. Uma vez que tivesse finalizado a ordenação do subvetor de 1 a 4 ele desempilharia o subvetor de 5 a 7 para então resolvê-lo.

Usando o ProfesSort:

Você é aluno do curso de AEDsII e está estudando o método de ordenação Quicksort, que se caracteriza por ser um método rápido e eficiente. Seu professor lhe apresenta o programa ProfesSort, a fim de auxiliá-lo no aprendizado do Quicksort. Como o professor falou que vocês poderiam usar o ProfesSort na resolução dos exercícios, você resolve usá-lo para garantir que você não vai cometer nenhum erro e também para tirar eventuais dúvidas que apareçam durante o exercício.

No primeiro exercício, o professor forneceu o vetor abaixo para ser ordenado, e solicitou que você mostrasse passo a passo como ordenar usando o Quicksort:

4	9	5	3	7	1	8
1	2	3	4	5	6	7

Assim, você selecionou o **tamanho** do vetor como sendo **7** e entrou **manualmente** os **valores dos elementos**. O exercício define ainda que o **elemento central** do vetor deve ser usado como **pivô** e que sempre deve-se **empilhar o lado esquerdo**.

Você então resolve o problema passo a passo e então grava a solução para entregar ao professor.

Exercício 2

Como você gostou do ProfesSort e achou que lhe poderia ser útil, então você resolveu fazer mais um exercício. Você resolve ordenar um vetor também de 7 elementos para não ser muito demorado, e define que os elementos serão escolhidos aleatoriamente pelo ProfesSort. Como o professor disse que o pivô poderia ser um elemento aleatório você resolve utilizar o primeiro elemento como pivô e empilhar sempre o maior lado.

Roteiro para Entrevista Pós-Teste

1. Experiência com softwares educacionais e de visualização; AEDS2 ou outras disciplinas.
2. Pontos positivos do ProfesSort
3. Pontos negativos do ProfesSort
4. Menu de perguntas: utilizou ou não, se sim: achou útil; se não: por que não.
5. Níveis de erro: percebeu os níveis – acha que é interessante ou não.
6. Percebeu vetor na memória, com cores marcando estado dos subvetores.
7. Outros algoritmos de ordenação acha que seria interessante ter um apoio. Tipo de apoio.
8. Gostaria de utilizar (ou ter utilizado) o ProfesSort durante o aprendizado do Quicksort
9. Recomendaria para colegas. Em caso afirmativo, quais colegas.
10. Algum outro comentário ou sugestão.

Formulário de Acompanhamento do Teste

Pesquisa: Avaliação da Comunicabilidade do ProfesSort Versão Beta1.4

Avaliador: _____ **Data do Teste:** ____/____/____

Identificação do Participante: _____

Tarefa 1:

Ordenar:

4	9	5	3	7	1	8
1	2	3	4	5	6	7

Pivô: elemento central; Empilhar: lado esquerdo.

Observações:

Perguntas para a Entrevista:

Terminou: () Sim () Não

Sobre o menu de perguntas:

Utilizou: () Sim () Não

Achou o que procurou: () Sempre () Algumas vezes () Nunca

Níveis de erro:

Errou mais de uma vez: () Sim () Não

Percebeu os diferentes níveis: () Sim () Não

Tarefa 2:

Ordenar vetor de 7 elementos aleatórios; Pivô: primeiro elemento; Empilhar: maior lado.

Observações:

Perguntas para a Entrevista:

Terminou: () Sim () Não

Sobre o menu de perguntas:

Utilizou: () Sim () Não

Achou o que procurou: () Sempre () Algumas vezes () Nunca

Níveis de erro:

Errou mais de uma vez: () Sim () Não

Percebeu os diferentes níveis: () Sim () Não

Anexo E

MAC: Relatório da Etiquetagem do ProfesSort

ProfesSort: Aplicação do Método de Avaliação de Comunicabilidade

PARTICIPANTE 2

Tarefa 1

Etiqueta	Tempo	Interação	Conteúdo	Explicação
Ué, o que houve?	1:43	X		
Epa!	2:36		X	P não selecionou o elemento j corretamente. Clicou em outro elemento e o sistema apresentou a 1ª msg de erro/ajuda. Após a ajuda, selecionou o elemento j correto.
Epa!	3:19		X	Selecionou o próximo passo errado (Finalizar trocas).
E agora?	3:38		X	Escolheu o elemento i, mas não sabia o que fazer depois. Ficou vagando com o mouse pela interface.
Epa!	3:57		X	Selecionou o passo errado
Epa!	4:22		X	Selecionou a posição j incorretamente.
Epa!	4:33		X	Novamente não selecionou o elemento j correto.
E agora?	5:12		X	P totalmente perdido, vagando pela interface durante um longo tempo.
Socorro!	5:45		X	Finalmente o usuário pede ajuda.
E agora?	6:54		X	De novo, não sabe o que fazer.
Epa!	7:12		X	Seleciona o elemento j errado.
Epa!	7:19		X	Seleciona o elemento j errado.
Epa!	7:36		X	Seleciona o próximo passo errado.
Epa!	7:44		X	Seleciona o próximo passo errado.
Socorro!	8:01		X	Recorre à ajuda.
E agora?	8:15		X	Com o 'socorro', conseguiu selecionar o elemento j corretamente, mas novamente ficou sem saber qual seria o próximo passo.
Epa!	8:20		X	Selecionou o passo errado.
Socorro!	8:24		X	Recorre à ajuda e faz a ação correta.
Epa!	9:00		X	Seleciona o passo errado.
Epa!	9:18		X	Seleciona elemento i incorreto.
Socorro!	9:27		X	Recorre à ajuda.
Epa!	9:58		X	Selecionou ação incorreta.
Epa!	10:19		X	Selecionou o elemento i incorreto.
Socorro!	10:25		X	Recorreu à ajuda.
Epa!	10:51		X	Selecionou a posição j errada.
Socorro!	11:03		X	Recorreu à ajuda.
Epa!	11:21		X	Selecionou o passo errado.
Epa!	12:34		X	A partir deste epa, foi percebido uma melhoria do participante. Estava executando as ações de forma correta sem recorrer à ajuda.
Epa	13:50		X	Selecionou o passo errado.
Epa!	14:21		X	Selecionou o passo errado.
Epa!	14:43		X	Selecionou o passo errado.

Tarefa 2

Etiqueta	Tempo	Interação	Conteúdo	Explicação
Epa!	3:16		X	Selecionou o i errado.
Epa!	4:14		X	Selecionou o passo errado.
Epa!	4:23		X	Selecionou o i errado.
Epa!	4:47	X		Usuário não selecionou a ação e clicou no elemento.

PARTICIPANTE 2

Tarefa 1

Etiqueta	Tempo	Interação	Conteúdo	Explicação
Epa!	1:34	X		Usuário clicou no elemento para selecionar o i, mas antes deveria selecionar a opção 'Efetuar trocas'.
E agora?	1:57		X	Usuário não sabia o q fazer, qual seria o próximo passo.
Socorro!	2:03		X	Recorre à ajuda para realizar a próxima ação.
Socorro!	2:58		X	Recorre à ajuda para selecionar o j.
Epa!	3:27		X	Fez apenas uma troca e já selecionou a opção finalizar trocas.
E agora?	3:31		X	Pede para mostrar o próximo passo.
Epa! Porque não funciona?	3:36 a 3:44		X	Seleciona o mesmo elemento duas vezes.
Socorro!	4:03		X	Recorre à ajuda para selecionar o j.
E agora?	4:35		X	P não sabe o que fazer. Fica vagando com o mouse pela tela.
Cadê?	4:58	X		Procura pela pergunta que pode auxiliá-lo.
Socorro!	6:06		X	Recorre à ajuda para selecionar o j.
Epa!	6:24		X	Seleciona a opção de troca incorretamente.
E agora?	6:31		X	Sem saber o que fazer, pede para mostrar o próximo passo.
E agora?	6:45		X	Pede novamente para mostrar o próximo passo.
Epa!'	7:59		X	Selecionou o j errado.
Socorro!	8:05		X	Recorre à ajuda.
Epa!	8:16		X	Quis finalizar antes de terminar a troca
E agora?	8:26		X	Selecionou mostrar próximo passo.
E agora?	8:57		X	Ficou na dúvida se deveria trocar elemento ou escolher j. Resolveu mostrar o próximo passo.
Epa!	9:18		X	Selecionou o pivô errado.
O que é isso?	9:30		X	Clicou na opção código, para saber o q era. Parece q só neste momento ele percebeu a outra aba.
Socorro	9:37		X	Recorreu à ajuda.
Epa!	10:07		X	Tentou finalizar antes de concluir as trocas.
E agora?	10:44		X	Pedi para mostrar o próximo passo.
Epa!	11:35	X		Selecionou o pivô antes de selecionar a opção de 'selecionar pivô'.
E agora?	11:58		X	Pede para mostrar o próximo passo
E agora?	12:10		X	Pede para mostrar o próximo passo

Tarefa 2

Etiqueta	Tempo	Interação	Conteúdo	Explicação
Epa!	1:08		X	Selecionou o i errado.
Socorro!	1:14		X	Recorreu à ajuda para saber qual o i selecionar.
Epa!	1:29		X	Selecionou novamente o i errado.
Epa!	1:40		X	Selecionou novamente o i errado. Mensagem de 3º nível.
Epa! Pq não funciona?	1:54		X	Mesmo com a resposta, o p ainda selecionou de novo o i errado.
Socorro!	2:55		X	Recorre à ajuda para selecionar j
Epa!	3:15		X	Tentou fazer a troca.
E agora?	3:21		X	Pedi para mostrar o próximo passo.
E agora?	3:30		X	Pedi para mostrar o próximo passo.
Epa!	3:48		X	Selecionou o i errado.
Socorro!	4:00		X	Recorreu à ajuda para selecionar j.
E agora?	4:19		X	Pedi para mostrar o próximo passo.
E agora?	4:35		X	Pedi para mostrar o próximo passo.
Epa!	4:48		X	Seleciona próximo passo errado.
Epa!	5:05	X		Selecionou o elemento antes de selecionar a opção 'selecionar pivô'.
Epa!	5:49		X	Seleciona opção errada. Deveria finalizar trocas.
Epa!	6:40	X		Selecionou o elemento antes de selecionar a opção 'selecionar pivô'.
Socorro!	6:48		X	Recorre à ajuda.
Epa!	7:44		X	Seleciona opção errada. Deveria finalizar trocas ao invés de trocar elementos
Epa!	8:01		X	Deveria selecionar o pivô, antes de selecionar i.
E agora?	8:26		X	Pedi para mostrar o próximo passo.

PARTICIPANTE 5

Tarefa 1

Etiqueta	Tempo	Interação	Conteúdo	Explicação
Epa!	1:21	X		Usuário deveria selecionar o pivô, antes de clicar no elemento.
Epa!	1:57		X	Selecionou o j errado.
E agora?	2:18 a 2:35		X	Não sabia o q fazer.
Epa!	2:38		X	Selecionou o j errado.
Epa!	3:17		X	Usuário deveria selecionar o j, antes de clicar no elemento.
E agora?	4:01		X	Pede para mostrar o próximo passo.
Epa!	4:34		X	Seleciona o passo errado, quer trocar mas i < j.
Epa!	5:06		X	Selecionou o j errado.
Epa!	5:27		X	Selecionou o j errado.
Epa!	5:43		X	Selecionou o j errado pela 3ª vez.
Epa!	5:58		X	Selecionou a troca errada.
E agora?	6:02		X	Pedi para mostrar o próximo passo.
Epa!	6:44		X	Seleciona i errado.

Epa!	7:01		X	Seleciona j errado.
Epa!	7:35		X	Seleciona passo errado.
Epa!	7:58		X	Seleciona passo errado ($i = j$) e ele pede para trocar ($i > j$)
E agora?	8:01		X	Pede para mostrar o próximo passo.
Epa!	8:11		X	Seleciona passo errado ($i = j$) e ele pede para finalizar troca ($i > j$)
E agora?	8:15		X	Pede para mostrar o próximo passo.
Epa!	8:26		X	Seleciona o i errado.
Epa!	8:42		X	Seleciona o i errado.
Epa!	5:55		X	Seleciona o i errado.
Epa!	9:06		X	Seleciona o i errado, apesar do sistema ter apresentado a resposta.
E agora?	10:01		X	Pede para mostrar o próximo passo.
Epa!	10:08		X	Seleciona i errado.
Epa!	10:23		X	Seleciona j errado.
Epa!	10:46		X	Seleciona passo errado.
Epa!	10:50		X	Seleciona passo errado.
E agora?	10:58		X	Pede para mostrar próximo passo.
Epa!	11:46		X	Seleciona passo errado.

Tarefa 2

Etiqueta	Tempo	Interação	Conteúdo	Explicação
Epa!	0:56		X	Seleciona i errado.
Socorro!	1:06		X	Recorre à ajuda.
Vai de outro jeito	1:31		X	Seleciona o mesmo elemento para ler a mensagem de nível 2 e 3
Epa!	2:41		X	Seleciona i errado.
Epa!	2:57		X	Seleciona i errado (outro elemento).
Epa!	3:22		X	Seleciona i errado (outro elemento).
Epa!	4:12		X	Seleciona j errado
Vai de outro jeito.	4:15		X	Seleciona J errado novamente em outro elemento. Lê mensagem de 2 nível.
Epa!	4:29		X	Troca elemento quando $i > j$
Epa!	4:57		X	Seleciona i errado.
Vai de outro jeito	5:51		X	Seleciona i errado. Como eram 3 opções, foi selecionando, sem pensar.

Anexo F

Material de Aplicação do MAC no Bipide

SCRIPT PARA O SISTEMA BIPIDE

- **Qual é o sistema?**
 - Sistema de apoio à aprendizagem de Organização de Computadores
- **Objetivo da pesquisa:**
 - Avaliar a comunicabilidade do software (explicitar que o que será avaliado é o sistema e não o usuário).
- **Como será realizado o teste?**
 - Explicar sobre a sala de teste, sobre os observadores, sobre o anonimato da pesquisa...
 - Explicar os passos do teste (leitura e assinatura do termo de consentimento, entrevista pré-teste para identificação do perfil do usuário, duração média do teste, realização das tarefas pelo usuário e entrevista pós-teste).
 - Tirar todas as dúvidas do usuário.
- **Leitura/assinatura do termo de consentimento**
- **Preenchimento do questionário pré-teste.**
- **Agradecimento** pela participação voluntária.

TERMO DE CONSENTIMENTO DE PARTICIPAÇÃO

Título: Avaliação da Comunicabilidade do sistema Bipide

Data: Dezembro/2009

Instituição: DCC / UFMG

Pesquisadores Responsáveis:

Raquel Prates (rprates@dcc.ufmg.br)

Erica Oliveira (erica@dcc.ufmg.br)

Introdução: Este Termo de Consentimento contém informações sobre a pesquisa indicada acima. Para assegurar que você esteja informado sobre a sua participação nesta pesquisa, pedimos que leia este Termo de Consentimento. Caso tenha alguma dúvida, não hesite em perguntar ao pesquisador responsável. Você também deverá assinar o termo do qual receberá uma cópia.

Objetivo da avaliação: O objetivo desta avaliação é identificar, na interface do Bipide – sistema de apoio à aprendizagem de Organização de Computadores –, problemas de comunicabilidade que possam dificultar o seu uso.

Informação geral sobre a pesquisa: Você será solicitado a realizar algumas tarefas simples utilizando o sistema. A realização dessas tarefas será gravada para posterior análise pelos investigadores. Ao fim da execução das tarefas, será realizada uma entrevista sobre sua experiência com o sistema.

Utilização dos dados coletados: Os dados coletados durante a avaliação serão utilizados para estudo do Método de Avaliação da Comunicabilidade (MAC). Quaisquer dados utilizados para publicação serão apresentados de forma a garantir o anonimato dos participantes da avaliação.

Privacidade: Informações que possam identificar os participantes da pesquisa não serão divulgadas. O seu nome não aparecerá em nenhum relatório. Caso deseje, poderá solicitar uma cópia dos dados gerados por você.

Se você decidir não participar na pesquisa: Você é livre para decidir, a qualquer momento, se quer participar ou não nesta pesquisa. Sua decisão não afetará sua vida estudantil e nem qualquer relacionamento com os avaliadores, professores ou a Instituição por trás desta.

Compensação: A participação nesta pesquisa é voluntária, e não será oferecida nenhuma remuneração aos seus participantes.

Se tiver algum problema ou se tiver outras perguntas: Se você tiver algum problema que pensa que pode estar relacionado com sua participação nesta pesquisa, ou se tiver qualquer pergunta sobre a pesquisa, poderá entrar em contato com os pesquisadores a qualquer momento pelo e-mail erica@dcc.ufmg.br.

Novas condições: Caso deseje, você pode especificar novas condições que devem ser atendidas para que você participe desta avaliação.

Consentimento Livre e Esclarecido (Acordo Voluntário)

O documento mencionado acima descrevendo os benefícios, riscos e procedimentos da pesquisa Avaliação da Comunicabilidade do Sistema Bipide foi lido e explicado. Eu tive a oportunidade de fazer perguntas sobre a pesquisa, que foram respondidas satisfatoriamente. Eu estou de acordo em participar como voluntário.

_____ Nome do participante: _____

Data

_____ Nome da pesquisadora: _____

Questionário Pré Teste

Data: ____/12/2009

Identificador: _____

Nome: _____

Sexo: () Feminino () Masculino

Idade: ____

Por favor, preencha os campos com a opção que melhor representa a sua resposta a cada pergunta.

1: Qual curso está fazendo?

- () Ciência da Computação
- () Sistemas de Informação
- () Matemática Computacional
- () Engenharia Elétrica
- () Engenharia de Controle e Automação
- () Outro. Qual? _____

2: Em que ano e semestre ingressou no curso: _____

3: Em que ano e semestre está prevista sua formatura: _____

4: Você utiliza/utilizou algum sistema de apoio à aprendizagem?

- () Sim. Qual(is)? _____
- () Não

5: Como você classifica o seu nível de conhecimento sobre conceitos de lógica de programação?

- () Não tenho conhecimento
- () Conheço a teoria
- () Conheço bem a teoria, mas não entendo o funcionamento na prática.
- () Conheço bem a teoria e consigo desenvolver programas usando a lógica de programação.

6: Como você classifica o seu nível de conhecimento sobre processamento de hardware?

- () Não tenho conhecimento
- () Conheço a teoria
- () Conheço bem a teoria, mas não entendo o funcionamento na prática.
- () Conheço bem a teoria e entendo bem o processamento do hardware.

7: Como você adquiriu o conhecimento sobre processamento de hardware? (marque todas as alternativas que julgar adequadas à resposta)

- () Não tenho conhecimento.
- () Assisti aulas expositivas.
- () Li o material didático indicado pelo professor.
- () Li material didático conseguido na Internet ou indicado por colegas.
- () Usei sistemas de visualização sobre processamento de hardware.

Tarefas para o sistema *Bipide*

Tarefa 1 – Edição de programa em linguagem portugol

Você é aluno da disciplina de Organização de Computadores. Você tem conhecimento em lógica de programação e está estudando sobre o funcionamento do hardware ao executar um programa. Seu professor lhe apresentou o programa Bipide, a fim de auxiliá-lo no aprendizado do processo que o hardware faz para executar um programa. Como o professor falou que você poderia usar o Bipide para escrever códigos na linguagem portugol e simular o funcionamento do hardware ao executar este código, você resolve usá-lo para tirar eventuais dúvidas que apareçam e assim consolidar seu aprendizado.

Na primeira atividade, o professor solicitou que você deveria criar um código na linguagem portugol para somar 2 números.

Uma vez criado o código, você decide salvá-lo e faz o teste para ver se seu código está correto.

Tarefa 2 – Visualização e simulação do processador Bip

Como você gostou do Bipide e achou que lhe poderia ser útil, então você resolveu fazer mais uma atividade. Você resolve simular um código dado pelo professor e verificar como o hardware processa este código.

```
programa prog2
declarações
    inteiro x
    inteiro y
início
    y <- 0
    para x <- 1 até 5
        passo 1
            y <- 1 + x
        fimpara
fim
```

Por medida de segurança, você também opta por salvar este código e aproveitá-lo para testar os recursos de simulação oferecidos pelo Bipide.

Roteiro para Entrevista Pós-Teste

1. Experiência com softwares de apoio à aprendizagem.
2. Você acha que o seu nível de conhecimento foi compatível com as telas e vocabulário empregado no sistema?
3. As mensagens de erro lhe auxiliaram de forma adequada?
4. O sistema lhe proporcionou *feedback*, ou seja, você conseguiu visualizar todas as suas ações?
5. A estrutura dos painéis do menu lhe parece disposta de forma lógica por agrupamento de tipos de opções?
6. O sistema atendeu às suas necessidades? Você o utilizaria?
7. Você indicaria este sistema para outras pessoas? Por quê?
8. Pontos positivos do Bipide.
9. Pontos negativos do Bipide.
10. Algum outro comentário ou sugestão.

Formulário de Acompanhamento do Teste

Pesquisa: Avaliação da Comunicabilidade do sistema Bipide

Avaliador: _____ **Data do Teste:** ____/____/____

Identificação do Participante: _____

Tarefa 1 – Configuração

Você é aluno da disciplina de Organização de Computadores. Você tem conhecimento em lógica de programação e está estudando sobre o funcionamento do hardware ao executar um programa. Seu professor lhe apresentou o programa Bipide, a fim de auxiliá-lo no aprendizado do processo que o hardware faz para executar um programa. Como o professor falou que você poderia usar o Bipide para escrever códigos na linguagem português e simular o funcionamento do hardware ao executar este código, você resolve usá-lo para tirar eventuais dúvidas que apareçam e assim consolidar seu aprendizado.

Na primeira atividade, o professor solicitou que você deveria criar um código na linguagem português para somar 2 números.

Uma vez criado o código, você decide salvá-lo e faz o teste para ver se seu código está correto.

Observações:

Perguntas para a Entrevista:

Concluiu a tarefa: () Sim () Não

Formulário de Acompanhamento do Teste

Pesquisa: Avaliação da Comunicabilidade do sistema Bipide

Avaliador: _____ **Data do Teste:** ____/____/____

Identificação do Participante: _____

Tarefa 2 – Visualização e simulação do processador Bip

Como você gostou do Bipide e achou que lhe poderia ser útil, então você resolveu fazer mais uma atividade. Você resolve simular um código dado pelo professor e verificar como o hardware processa este código.

```
programa prog2
declarações
    inteiro x
    inteiro y
início
    y <- 0
    para x <- 1 até 5
        passo 1
            y <- 1 + x
        fimpara
fim
```

Por medida de segurança, você também opta por salvar este código e aproveitá-lo para testar os recursos de simulação oferecidos pelo Bipide.

Observações:

Perguntas para a Entrevista:

Concluiu a tarefa: () Sim () Não

Principais comandos do Bipide

Portugol	Descrição
programa	Símbolo inicial da gramática
declaracoes	Definição de bloco de declaração de variáveis
defina	Definição de constantes
inicio	Identifica o início do algoritmo
fim	Identifica o fim do algoritmo
inteiro	Tipo de dado numérico inteiro ¹⁰
se ___ entao	Desvio condicional
senao	Negação do 'SE'
fimse	Fim de bloco de desvio condicional
enquanto ___ faca	Laço de repetição com condição no início
fimenquanto	Fim de bloco de laço condicional
repita ___ quando	Laço de repetição com condição no fim
para __ ate __ passo __	Laço condicional com repetição incremental
fimpara	Fim de bloco de laço condicional com repetição incremental
<-	Operador de atribuição
(,)	Parênteses
+, -	Operadores aritméticos
>, <, <=, >=, !=, =	Operadores relacionais

Anexo G

MAC: Relatório da Etiquetagem do Bipide

BIPIDE: Aplicação do Método de Avaliação de Comunicabilidade

PARTICIPANTE 2

Tarefa 1

Etiqueta	Tempo	Interação	Conteúdo	Explicação
				No início da tarefa, o aluno releu a tarefa e os comandos do Bipide que estavam à sua disposição. Após a leitura, ele perguntou se poderia fazer perguntas no início do teste.
O que é isto?	3:13	X		Usuário coloca o mouse sobre o botão Salvar arquivo como e aguarda até que o label apareça.
Porque não funciona?	3:53 a 4:10		X	Usuário acredita que o código está correto e tenta simulá-lo (o código está incorreto, faltando parâmetros).
Socorro!	4:12		X	Usuário ficou lendo o material com explicação sobre o Bipide por mais de um minuto.
Porque não funciona?	5:06		X	Após ler o texto sobre o Bipide, usuário não faz nenhuma alteração. Parece não entender o porque o sistema deu erro. Clica novamente no botão simular e o mesmo erro é apresentado (comando em formato inválido).
E agora?	5:43		X	Usuário recorta parte do código (início e fim) e, a partir da leitura do texto, acrescenta o nome para programa. Em seguida, fica vagando com o curso, sem saber o que fazer.
Porque não funciona?	5:51		X	Após as modificações do código, usuário clica novamente no botão simular e o erro persiste.
Assim não dá.	6:25		X	Usuário insere novamente as opções de início e fim do programa. O código continua dando a mesma msg de erro.
E agora?	7:00 à 7:30		X	Usuário fica vagando com o cursor pela tela, sem saber o que fazer.
Vai de outro jeito	8:10		X	Usuário não entende porque o código está errado e resolve declarar as variáveis de uma forma diferente da que ele estava declarando.
Porque não funciona?	8:20		X	Após as alterações, usuário clica em simular e o erro persiste.
Porque não funciona?	9:25		X	Usuário faz pequenas modificações e tenta simular o código criado, sem sucesso.
Socorro!	9:41		X	Usuário lê o material impresso com explicação sobre o Bipide.
Socorro!	10:04		X	Usuário recorre à ajuda do Bipide.
Socorro!	10:40		X	Usuário copia, do help, um trecho de código

				sobre como criar um algoritmo em portugol. Na tela principal, ele apaga tudo que havia feito e cola o código copiado do help. A partir deste código, ele faz as alterações para o código solicitado na tarefa 1.
Porque não funciona?	13:19		X	Usuário faz as modificações, salva o código e tenta simular. A simulação dá erro, mas ele não entende o porquê.
E agora?	13:57		X	Usuário parece não saber o que fazer. Inicia um processo de alterar ou excluir partes do código e a todo momento, compila, para ver se o código funciona.
Socorro	14:40 à 15:40		X	Usuário, após várias tentativas sem sucesso, recorre novamente à ajuda do sistema para entender o porque seu código não compila.
Porque não funciona?	16:12		X	Usuário faz as modificações, salva o código e tenta simular. A simulação dá erro, mas ele não entende o porquê.
Socorro!	16:18		X	Usuário recorre novamente à ajuda.
E agora?	17:18		X	Usuário, após várias tentativas erradas, resolve salvar o programa com outro nome. Compila novamente o programa, mas o erro persiste.
Desisto	18:15		X	Usuário desiste de concluir a tarefa.

Tarefa 2

Etiqueta	Tempo	Interação	Conteúdo	Explicação
E agora.	2:21		X	Usuário, ao concluir a digitação do código da tarefa, clica em compilar. O programa contém um erro, mas o usuário tem dificuldade em corrigir este erro.
Socorro	2:45		X	Usuário recorre à folha impressa sobre o Bipide e revê o código a fim de encontrar o erro. Após algumas alterações, o código é compilado.
O que é isto?	4:42	X		Usuário clica para pausar a simulação e vai com o cursor do mouse sobre a Unidade Lógica Aritmética (ULA).
O que é isto?	5:54	X		Usuário pausa a simulação e acompanha os valores das variáveis no processador Controle.
Epa!	6:16	X		Usuário altera a velocidade da simulação para rápida, mas percebe que vai rápido demais e ele não consegue acompanhar o que está acontecendo com as variáveis. Então desfaz a ação.

PARTICIPANTE 3

Tarefa 1

Etiqueta	Tempo	Interação	Conteúdo	Explicação
Socorro!	00:00 a 01:00		X	No início da tarefa, o aluno releu a tarefa e os comandos do Bipide que estavam à sua disposição.
E agora?	03:20 a 3:33		X	Usuário digita o código que ele acredita estar correto, salva e pede para simular. O sistema emite uma msg de erro. O usuário fica vagando com o cursor pela interface do sistema.
Porque não funciona?	4:05		X	Usuário faz uma modificação no código, clica em compilar. O código continua com erro. Ele retorna ao código sem entender o pq ele está errado.
O que é isto?	4:12 a 4:17	X		Usuário passa o cursor sobre alguns elementos da interface (copiar, colar, exportar e salvar arquivo).
Porque não funciona?	4:21		X	Sem fazer qualquer alteração, ele clica novamente no botão compilar.
Socorro	4:26 a 4:35		X	Usuário recorre à ajuda do Bipide, mas não se demora muito nela.
Epa!	4:43	X		Usuário clica na opção Instruções sem querer e retorna à tela Programação.
E agora?	5:00		X	Usuário não sabe o que fazer, fica vagando com o cursor pela interface.
Porque não funciona?	5:16		X	Sem fazer qualquer alteração, ele clica no botão simular.
Porque não funciona?	5:45		X	Exclui a palavra declarações e clica em simular. Não entende o porquê o programa continua com erro e digita a palavra declarações novamente.
E agora?	6:25		X	Usuário não sabe o que fazer e fica vagando com o cursor sobre os elementos da interface.
Porque não funciona?	6:55		X	Usuário faz duas pequenas modificações, clica em compilar e o erro persiste.
E agora?	7:00		X	Usuário vaga com o cursor pela interface sem saber o que fazer.
Socorro!	7:28 a 7:50		X	Usuário recorre à folha impressa com instruções sobre o programa.
Socorro!	7:53 a 8:16		X	Usuário recorre à ajuda do sistema.
E agora?	8:46 a 8:53		X	Usuário faz algumas modificações a partir da leitura do help, mas o programa continua com erro. Usuário fica sem saber o que fazer e vaga

				com o cursor pela tela.
Socorro!	9:00 a 9:10		X	Usuário recorre à ajuda do sistema e consegue compilar o código.

Tarefa 2: não houve ruptura.

PARTICIPANTE 4

Tarefa 1

Etiqueta	Tempo	Interação	Conteúdo	Explicação
Socorro!	0:04		X	No início da tarefa, o aluno releu a tarefa e os comandos do Bipide que estavam à sua disposição.
O que é isto?	0:12	X		Passou o cursor do mouse sobre a opção novo.
Socorro!	0:25 a 0:36		X	Recorreu à ajuda do sistema, pois disse não se lembrar do Portugol.
Socorro!	0:55 a 1:18		X	Recorreu à ajuda do sistema para saber como poderia iniciar a escrita do código.
E agora?	2:33 a 2:47		X	Usuário não concluiu a digitação do código e clicou na aba simulação. Na tela de simulação, ficou vagando com o cursor sem saber o que fazer.
O que é isto?	2:49	X		Ainda sem saber o que fazer, clicou na opção instruções.
Socorro!	3:03 a 3:41		X	Recorreu à ajuda do sistema para saber como proceder com a escrita do código. Em seguida, fez algumas inserções no código.
Socorro!	04:00		X	Recorre à ajuda para obter mais informações sobre a escrita do código.
E agora?	4:20		X	Crendo que o código estava certo, o usuário clica em simular e tentar simular o código.
E agora?	4:45	X		Usuário clica em instruções e fica vagando com o cursor pela tela, clicando em vários botões aleatórios.
Vai de outro jeito.	5:10		X	Usuário não consegue fazer o código compilar na aba Programação e resolve copiar e colar o código na aba Instruções.
Epa!	5:21		X	Usuário clica na opção LD (Instruções) e o sistema emite uma mensagem de erro.
Porque não funciona?	5:28		X	Usuário clica novamente no botão LD (instruções) e aparece novamente a mensagem de erro.
E agora?	5:40		X	Usuário fica perdido e não sabe o que fazer. Clica na opção simulação e vaga com o cursor pela tela.
Porque não	5:48		X	Usuário clica no botão simular e não entende pq

funciona?				sua ação não tem um retorno.
Socorro!	5:52 a 7:35		X	Usuário recorre à ajuda do sistema.
O que é isto?	8:25	X		Usuário passa o cursor do mouse sobre as ferramentas disponíveis na tela de programação.
Porque não funciona?	8:35		X	Usuário clica em compilar e o programa dá mensagem de erro.
Socorro!	8:58 a 9:07		X	Usuário recorre à ajuda do sistema.
Porque não funciona?	9:28			Após recorrer à ajuda, usuário faz algumas alterações e clica em compilar, mas o programa dá mensagem de erro.
E agora?	9:35		X	Usuário vaga com o cursor sem saber o que fazer.
Vai de outro jeito.	10:00		X	Usuário não consegue fazer o código compilar na parte de programação. Usando ctrlc + ctrlv, ele cola o código na parte de instruções.
Socorro!	10:25 a 11:34		X	Usuário recorre à ajuda do sistema.
Desisto.	11:45		X	Usuário desiste da tarefa por não conseguir compilar o código.

Tarefa 2

Etiqueta	Tempo	Interação	Conteúdo	Explicação
Epa!	1:52	X		Usuário clica errado na opção simulação e retorna à aba programação.
Vai de outro jeito.	2:13		X	Usuário copia e cola o código na aba Instruções com a intenção de compilar o código.
Porque não funciona?	2:30		X	Usuário clica na aba simulação e depois em simular, mas o código não compila, pq está com erro.
Porque não funciona?	3:34		X	Usuário insiste em clicar na opção simular, na aba simulação, mas o código tem erro. Esta ação foi feita várias vezes.

PARTICIPANTE 5

Tarefa 1

Etiqueta	Tempo	Interação	Conteúdo	Explicação
E agora?	0:50		X	Como o código estava escrito em C, o programa deu erro.
Socorro!	1:02 a 2:16		X	Usuário recorre à ajuda do sistema.
Socorro!	2:53		X	Usuário recorre à ajuda do sistema para saber como é a sintaxe para declaração de variável.

Porque não funciona?	3:18		X	Usuário clica em compilar 2x seguidas e não entende o porque o código está errado. Detalhe: O código misturava c e portugol.
Socorro!	3:34 3:47	a	X	Usuário recorre à ajuda para sanar o erro.
Socorro!	3:57 4:32	a	X	Usuário faz algumas modificações, o código continua com erro e ele recorre à ajuda novamente.
Porque não funciona?	4:40		X	Usuário faz uma modificação e clica em compilar 2x, mas o código continua errado.
Socorro	4:42 4:50	a	X	Usuário recorre à ajuda do sistema.
Porque não funciona?	5:08		X	Usuário faz alterações e clica em compilar, mas o erro persiste.
Socorro	5:17 6:11	a	X	Usuário recorre à ajuda do sistema.
O que é isto?	6:31	X		Usuário para o cursor sobre um ícone (copiar) na tela.
Porque não funciona?	6:40		X	Usuário clica 2x seguidas no botão compilar e o código continua com um erro de variável.
Socorro	6:43 7:50	a	X	Usuário recorre à ajuda do sistema.
Epa	7:53	X		Usuário clica na aba simulação sem querer e volta à aba programação.
Socorro	8:14		X	Usuário recorre à ajuda novamente.
O que é isto?	8:22	X		Usuário clica na aba configurações.
Porque não funciona?	9:32		X	Usuário faz modificações no código e tenta compilar, mas dá erro e ele não entende o porque.
Vai de outro jeito	9:49		X	Usuário insere novos comandos no código, a fim de corrigi-lo, mas as informações inseridas estão erradas.
E agora?	10:00		X	Usuário faz várias tentativas de mudança no código, mas os erros persistem. Ele fica perdido, parece não entender as mensagens de erro fornecidas pelo sistema.
Socorro	12:00		X	Após várias tentativas, ele recorre à ajuda novamente.

Tarefa 2

Etiqueta	Tempo	Interação	Conteúdo	Explicação
O que é isto?	2:00	X		Usuário para o cursor do mouse sobre o botão novo.

PARTICIPANTE 06

Tarefa 1

Etiqueta	Tempo	Interação	Conteúdo	Explicação
Socorro!	00:12 a 1:00		X	Usuário foi direto à ajuda do sistema por não conhecer o português. Em seguida, recorreu à folha impressa com informações sobre o Bipide. À medida que digitava o código, ele recorria à folha ou à ajuda do sistema para saber como declarar as variáveis e a sintaxe do código. Com este auxílio, o usuário criou o código corretamente.
				Por volta de 3:30 ele concluiu a digitação do código.

Tarefa 2: não houve rupturas

Anexo H

Roteiro do MISI - Sistema VCalc

Termo de Consentimento de Participação

Título: Avaliação do VCalc utilizando o Método de Inspeção Semiótica Intermediado – MISI

Data: Outubro/2007

Instituição: Grupo de Pesquisa em Interação Humano-Computador / DCC / UFMG

Pesquisadores Responsáveis:

Prof. Raquel O. Prates (rprates@dcc.ufmg.br)

Erica Oliveira(erica@dcc.ufmg.br)

Luiz Carlos Luz (luizluz@dcc.ufmg.br)

Introdução: Este Termo de Consentimento contém informações sobre a pesquisa indicada acima. Para assegurar que você esteja informado sobre a sua participação nesta pesquisa, pedimos que leia este Termo de Consentimento. Caso tenha alguma dúvida, não hesite em perguntar ao pesquisador responsável. Você também deverá assinar o termo do qual receberá uma cópia.

Objetivo da avaliação: O objetivo desta avaliação é identificar, na interface do VCalc - sistema de apoio aprendizado de funções algébricas - problemas de comunicabilidade que possam dificultar o seu uso pelo aluno.

Informação geral sobre a Pesquisa: Você será solicitado a realizar algumas tarefas simples utilizando o sistema. A realização dessas tarefas será gravada para posterior análise pelos investigadores. Ao fim da execução das tarefas, será realizada uma entrevista sobre sua experiência com o sistema.

Utilização dos dados coletados: Os dados coletados durante a avaliação serão utilizados para melhorias no desenvolvimento do sistema VCalc e/ou para pesquisa de tópicos relacionados com o sistema ou método de avaliação utilizado. Quaisquer dados utilizados para publicação serão apresentados de forma a garantir o anonimato dos participantes da avaliação.

Privacidade: Informações que possam identificar os participantes da pesquisa não serão divulgadas. O seu nome não aparecerá em nenhum relatório. Caso deseje, poderá solicitar uma cópia dos dados gerados por você.

Se você decidir não participar na pesquisa: Você é livre para decidir, a qualquer momento, se quer participar ou não desta pesquisa. Sua decisão não afetará seu relacionamento com os avaliadores, professores ou a Instituição por trás desta.

Compensação: A participação nesta pesquisa é voluntária, e não será oferecida nenhuma remuneração aos seus participantes.

Se tiver algum problema ou se tiver outras perguntas: Se você tiver algum problema que pensa que pode estar relacionado com sua participação nesta pesquisa, ou se tiver qualquer pergunta sobre a pesquisa, poderá entrar em contato com os pesquisadores a qualquer momento pelos endereços eletrônicos erica@dcc.ufmg.br, raqprates@dcc.ufmg.br ou luizluz@dcc.ufmg.br ou pelo telefone 3499-5895.

Novas condições: Caso deseje, você pode especificar novas condições que devem ser atendidas para que você participe desta avaliação.

Consentimento Livre e Esclarecido (Acordo Voluntário)

O documento mencionado acima descrevendo os benefícios, riscos e procedimentos da pesquisa Avaliação do VCalc utilizando o MISI foi lido e explicado. Eu tive a oportunidade de fazer perguntas sobre a pesquisa, que foram respondidas satisfatoriamente. Eu estou de acordo em participar como voluntário.

Data: _____

Nome do participante:

Nome da pesquisadora:

MIS Intermediado

Sistema: VCalc

Ponto 1 – Perfil do entrevistado (como prof. De Cálculo)

Incluir aqui alguns pontos sobre o perfil do professor (parte do MEDS, mas não do MIS):

- Formação (quando se doutorou)
- Tempo de ensino de Cálculo I
- Método de ensino (recursos que usa em sala e fora para aprendizado e consolidação dos alunos)
- Utiliza-se de algum sistema de apoio ao ensino/aprendizado.
 - Quais.
 - Em que situações (em sala ou fora; para parte específica da matéria; etc).
 - Como chegou a eles.
 - Como definiu se eram ou não apropriados para seus objetivos.

Ponto 2 – Inspeção do sistema usando MIS guiado

1º Passo)

1) Dê uma olhada na documentação e descreva:

- o que este software faz
- qual é o seu público (professor, aluno, ambos)
- Como utilizá-lo (professor, aluno, ambos)
- Qual seu objetivo educacional

2º Passo e 3º Passo)

##Contato com o Software##

1) Dê uma olhada nos objetos (Botões, imagens, etc.) da interface do software. Você seria capaz de dizer o que eles significam e qual é a função de cada um deles? Tente fazer isso em voz alta. Por exemplo, diga isto aqui é um botão e deve servir para realizar tal função ou isto aqui me parece ser um botão, porém eu não sou capaz de dizer o que ele faz.

2) Você acha que seus alunos terão interpretações diferentes das suas para algum desses objetos? Quais objetos e quais poderiam ser a sua interpretação?

##Mudando a Função##

3) Vamos mudar de função. Você saberia identificar onde poderemos fazer isso? Pois bem, faça. O que você vê nesta tela? Interprete as informações. Você acha que os alunos teriam alguma interpretação diferente?

4) Agora que você já mudou a função, me responda. Qual é então a função deste botão "A Função"?

##Fim do Mudando a Função##

##Trabalhando com Taxa de Variação##

5) Vamos trabalhar agora com taxa de variação. Você saberia identificar onde poderemos fazer isso? Pois bem, faça. Interprete os objetos que você está vendo na tela. O que você acha que vai acontecer quando confirmar? Você acha que os alunos teriam uma interpretação diferente?

6) O que você acha que são as informações no canto superior esquerdo da tela? Interprete as informações e os objetivos. Você acha que os alunos as interpretariam de maneira diferente?

7) Interprete as informações do gráfico ou dos gráficos (Normal e do Zoom). Você acha que os alunos teriam alguma interpretação diferente?

##Trabalhando com Zoom##

8) Vamos olhar para a Janela do Zoom. O que acontecerá ao clicar nos botões “0.5x” e “2x”? Matematicamente, o que está acontecendo? Os alunos seriam capazes de ter esta mesma interpretação? Se não, o que você acha que eles pensariam?

##Fim do Trabalhando com Zoom##

9) Clique novamente no botão “Taxa de Variação”. Agora que você já trabalhou com Taxa de Variação, responda: Qual é então a função deste botão “Taxa de Variação”?

##Fim do Trabalhando com Taxa de Variação##

##Trabalhando com Mudança de Intervalos##

10) Vamos mudar os intervalos. Como você faria para ver o comportamento da função em um outro intervalo?

11) Agora que você utilizou esses objetos, você teria uma interpretação deles diferente da que você tinha antes de utilizá-los?

##Fim do Trabalhando com Mudança de Intervalos##

12) Tem algum objeto a mais do software que você interpreta de maneira diferente de quando nós começamos a realização do método?

13) A partir desta exploração do software, descreva novamente o que este software faz e qual é o seu público.

##Fim do Contato com o Software##

1) De novo:

- a. o que este software faz
- b. qual é o seu público (professor, aluno, ambos)
- c. Como utilizá-lo (professor, aluno, ambos)
- d. Qual seu objetivo educacional

Ponto 3 – Potencial de uso do sistema

- Utilidade do sistema para o ensino/aprendizado de Cálculo I. Justificativa
- Sistema poderia ser usado em seu curso. Como ou. Por que não.
- Este uso, estaria adequado ao seus métodos de ensino atuais, ou precisariam ser adaptados.
- Recomendaria a outros.

Anexo I

Roteiro do MISI - Sistema ProfesSort

Termo de Consentimento de Participação

Título: Avaliação do ProfesSort utilizando o Método de Inspeção Semiótica Intermediado – MISI

Data: Maio/2009

Instituição: Grupo de Pesquisa em Interação Humano-Computador / DCC / UFMG

Pesquisadores Responsáveis:

Prof. Raquel O. Prates (rprates@dcc.ufmg.br)

Erica Oliveira(erica@dcc.ufmg.br)

Gisleide Aidano (gisleide@dcc.ufmg.br)

Introdução: Este Termo de Consentimento contém informações sobre a pesquisa indicada acima. Para assegurar que você esteja informado sobre a sua participação nesta pesquisa, pedimos que leia este Termo de Consentimento. Caso tenha alguma dúvida, não hesite em perguntar ao pesquisador responsável. Você também deverá assinar o termo do qual receberá uma cópia.

Objetivo da avaliação: O objetivo desta avaliação é identificar, na interface do ProfesSort - sistema de apoio ao aprendizado dos métodos de ordenação de algoritmos - problemas de comunicabilidade que possam dificultar o seu uso pelo aluno.

Informação geral sobre a Pesquisa: Você será solicitado a realizar algumas tarefas simples utilizando o sistema. A realização dessas tarefas será gravada para posterior análise pelos investigadores. Ao fim da execução das tarefas, será realizada uma entrevista sobre sua experiência com o sistema.

Utilização dos dados coletados: Os dados coletados durante a avaliação serão utilizados para melhorias no desenvolvimento do sistema ProfesSort e/ou para pesquisa de tópicos relacionados com o sistema ou método de avaliação utilizado. Quaisquer dados utilizados para publicação serão apresentados de forma a garantir o anonimato dos participantes da avaliação.

Privacidade: Informações que possam identificar os participantes da pesquisa não serão divulgadas. O seu nome não aparecerá em nenhum relatório. Caso deseje, poderá solicitar uma cópia dos dados gerados por você.

Se você decidir não participar na pesquisa: Você é livre para decidir, a qualquer momento, se quer participar ou não desta pesquisa. Sua decisão não afetará seu relacionamento com os avaliadores, professores ou a Instituição por trás desta.

Compensação: A participação nesta pesquisa é voluntária, e não será oferecida nenhuma remuneração aos seus participantes.

Se tiver algum problema ou se tiver outras perguntas: Se você tiver algum problema que pensa que pode estar relacionado com sua participação nesta pesquisa, ou se tiver qualquer pergunta sobre a pesquisa, poderá entrar em contato com os pesquisadores a qualquer momento pelos endereços eletrônicos erica@dcc.ufmg.br, ragprates@dcc.ufmg.br ou gisleide@dcc.ufmg.br ou pelo telefone 3499-5895.

Novas condições: Caso deseje, você pode especificar novas condições que devem ser atendidas para que você participe desta avaliação.

Consentimento Livre e Esclarecido (Acordo Voluntário)

O documento mencionado acima descrevendo os benefícios, riscos e procedimentos da pesquisa Avaliação do ProfesSort utilizando o MISI foi lido e explicado. Eu tive a oportunidade de fazer perguntas sobre a pesquisa, que foram respondidas satisfatoriamente. Eu estou de acordo em participar como voluntário.

Data: _____

Nome do participante:

Nome da pesquisadora:

Documentação sobre o Sistema ProfesSort

Tópico do Conteúdo: Métodos de ordenação.

Tipo: Ambiente de apoio à aprendizagem sobre métodos de ordenação.

Objetivo: Permitir ao aluno executar os passos do método, observando a consequência de cada passo no vetor ou memória, de forma a consolidar o conteúdo aprendido em sala. O sistema oferece diferentes formas de apoio (*scaffolds*) aos alunos durante a execução do método.

Histórico: O ProfesSort foi desenvolvido no DCC/UFMG para apoiar os alunos da disciplina de Algoritmos e Estruturas de Dados II (AEDS II) na consolidação do aprendizado dos diversos métodos de ordenação vistos em sala de aula. Atualmente é disciplina obrigatória dos cursos de Ciência da Computação, Sistemas de Informação, Matemática Computacional, Engenharia Elétrica e Engenharia de Controle e Automação, somando aproximadamente 300 alunos por semestre.

Descrição do sistema:

O ProfesSort é um sistema que tem por objetivo auxiliar os alunos no aprendizado dos algoritmos de ordenação. Através da execução passo a passo dos algoritmos, os alunos têm a oportunidade de compreender melhor e exercitar alguns dos algoritmos de ordenação estudados em sala. O sistema permite que os alunos observem e controlem a execução do algoritmo indicando, por exemplo, qual é o próximo passo que deve ser executado, quais elementos devem ser escolhidos para troca, etc.

O ProfesSort foi modelado segundo o Modelo de Representação de Apoio ao Aprendizado (MRA). Este modelo tem por objetivo permitir ao educador representar os apoios (ou *scaffolds*) que devem ser oferecidos ao usuário a cada passo da tarefa para auxiliá-lo na consolidação do conteúdo desejado. Durante a execução, são oferecidos quatro tipos de apoio (*scaffolds*) aos alunos:

1. **Perguntas sobre o método** – a cada passo do método o sistema oferece a possibilidade de tirar suas dúvidas através do uso de perguntas pré-determinadas sobre o conteúdo sendo aprendido. Essas perguntas são divididas nas categorias “O que é...?”, “Como...?”, “E agora,...?”, “Quando...?”, e “Por que...?”, e são ficam habilitadas conforme o passo do método.
2. **Mensagens de erro graduais** – o aluno é encorajado a refletir sobre o erro cometido, recebendo aos poucos dicas sobre a resposta correta, ao invés de receber a resposta diretamente.
3. **Visão completa das opções** – todos os passos possíveis no algoritmo são exibidos para que o aluno faça a escolha do próximo passo a ser executado. O vetor sendo ordenado também é visualizado com a utilização de diferentes cores para indicar, por exemplo, qual sub-vetor está em execução, qual já foi ordenado, etc.
4. **Botões de auxílio** – permite ao aluno retornar ao passo anterior, ver qual passo deve ser executado ou executar um ou mais passos automaticamente.

A Figura 1 abaixo mostra a tela do sistema, destacando como os diferentes apoios são oferecidos ao usuário.

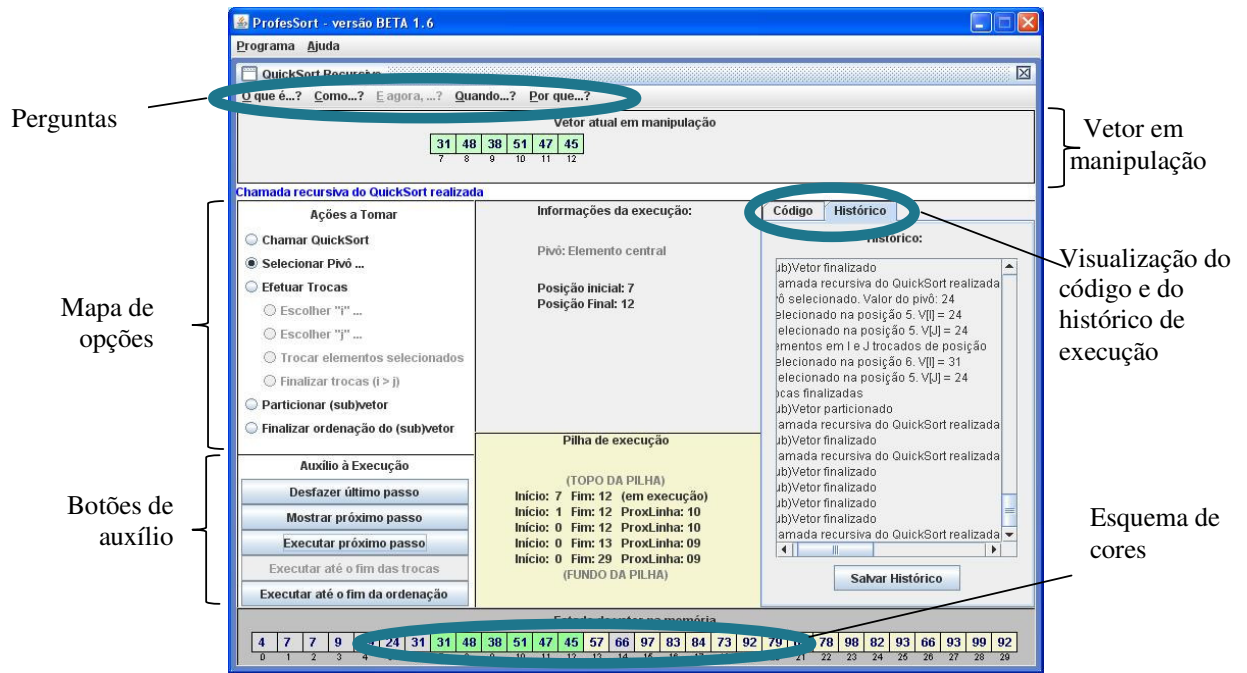


Figura 1: Tela do ProfesSort.

Atualmente o ProfesSort oferece a possibilidade de ordenar utilizando os métodos quicksort recursivo e não recursivo, heapsort, seleção e inserção [Ziviani, 2004; Cormem, 2001]. Vale ressaltar que diferentes métodos podem requerer visualizações distintas. Por exemplo, o quicksort mostra a pilha de execução, enquanto o heapsort apresenta uma visualização em árvore (inserção e seleção não requerem nenhuma visualização específica).

MIS Intermediado – ProfesSort

Apresentação da pesquisa

Falar do objetivo da pesquisa e método – uso com entrevista

Ponto 1 – Perfil do entrevistado

Pontos sobre o perfil do professor (como professor de AEDS II)

- Formação (quando se doutorou)
- Tempo de ensino de Algoritmos e Estrutura de Dados II.
 - Caso não ensine mais, há quanto tempo parou
 - Listar os métodos de ordenação ensinados em sala
 - (Se for entrevistar alguém que não é professor de AEDS II – qual disciplina relacionada utiliza, se lembra dos métodos de ordenação).
- Método de ensino
 - estilo da aula: é participativo OU é do tipo professor fala e aluno ouve
 - recursos que usa em sala e fora para aprendizado e consolidação dos alunos
- Utiliza-se de algum sistema de apoio ao ensino/aprendizado.
 - Quais?
 - Em que situações (em sala ou fora; para parte específica da matéria; etc.)?
 - Como chegou a eles?
 - Como definiu se eram ou não apropriados para seus objetivos?
- Visão sobre o uso de ferramentas de apoio à aprendizagem aos alunos
 - Pode auxiliar os alunos na compreensão da disciplina?

Ponto 2 – Inspeção do sistema usando MIS Intermediado

1º Passo)

- 1) Leia a documentação.
- 2) Com base no seu entendimento sobre a documentação, descreva:
 - O que este software faz.
 - Qual é o seu público alvo (professor, aluno, ambos).
 - Como cada um destes pode/deve utilizá-lo (professor, aluno, ambos).
 - Qual seu objetivo (educacional).

2º Passo e 3º Passo)

##Contato com o Software##

- 1) Dê uma olhada nos objetos (botões, imagens, etc.) da interface do software.
 - a. Descreva o que imagina que seja a função dos diversos elementos da interface. Tente fazer isso em voz alta. Por exemplo, diga isto aqui é um botão e deve servir para realizar tal função ou isto aqui me parece ser um botão, porém eu não sou capaz de dizer o que ele faz.
- 2) Expectativa que os alunos terão sobre estes elementos de interface (consistentes ou diferentes da do professor). Justificar. Identifica algum problema que poderiam ter.

##Utilizando o método de ordenação QuickSort Recursivo##

- 3) Escolher o método de ordenação QuickSort recursivo
 - a) Identificar onde fazer a seleção. Executar a ação.
 - b) Descreva o que imagina ser a função dos diversos elementos de interface.
 - c) Expectativa que os alunos terão sobre estes elementos de interface (consistentes ou diferentes da do professor). Justificar. Identifica algum problema que poderiam ter.

- 4) Na tela inicial de configuração dos parâmetros para ordenação:
 - a. Definir o tamanho do vetor como sendo 12 e selecionar o preenchimento aleatório. O pivô será o elemento central.

- 5) Na tela principal da ordenação com Quicksort:
 - a. Descreva o que imagina ser a função dos diversos elementos de interface.
 - i. Vetor em manipulação
 - ii. Estado do vetor na memória
 - b. Expectativa que os alunos terão sobre estes elementos de interface (consistentes ou diferentes da do professor). Justificar. Identifica algum problema que poderiam ter.

 - c. Observe os menus disponíveis no canto superior da tela (O que é? Como?)?
 - i. Interprete as informações e os objetivos. Você acha que os alunos as interpretariam de maneira diferente?

Iniciando o QuickSort recursivo

- 6) Vamos iniciar a ordenação utilizando o método QuickSort recursivo.
 - a. Descreva o que imagina ser a função da opção "Ações a tomar".
 - i. Você conseguiria dizer o que deve ser feito para iniciar a ordenação com o método QuickSort?
 - ii. Como você acha que pode interagir com esta opção? Pois então, faça. Sua percepção se confirmou? Você acha que seus alunos poderiam ter alguma interpretação diferente?

- 7) Um dos primeiros passos do método QuickSort recursivo é escolher o pivô.
 - a. Você sabe como realizar essa escolha? Pois bem, faça.
 - b. Agora que o elemento pivô já foi selecionado, como você acha que se podem selecionar os elementos i e j e efetuar as trocas?

Níveis de ajuda

- 8) Execute uma ação esperada erroneamente várias vezes.
 - a. Percebeu os níveis de ajuda.
 - b. Opinião sobre níveis de ajuda.
 - c. Você acha que estes níveis de ajuda podem auxiliar o aluno na aprendizagem dos métodos de ordenação?

- d. Você acha que seus alunos poderiam ter alguma interpretação diferente? Por quê?

Auxílio à execução

- 9) Vamos observar os botões de auxílio à execução.
- Descreva como você acha que estes botões podem ser utilizados e em que tipo de situação?
 - Vamos agora interagir com esses botões.
 - E então? Descreva novamente sua percepção. Ela se confirmou ou não?
 - Agora descreva sua opinião sobre estes botões em termos de apoio à aprendizagem dos métodos de ordenação.
 - Você acha que os alunos teriam uma interpretação diferente?

Informações da execução e estado da pilha

- 10) Agora vamos observar as informações que são apresentadas durante a execução do método QuickSort.
- Descreva, do ponto de vista da ordenação, o que você acha é informado nesta opção.
 - Você acha que os alunos poderiam ter alguma interpretação diferente?
- 11) Vamos observar as informações que apresentadas sobre o estado da pilha.
- Descreva, sob o ponto de vista da ordenação, o que você acha é informado nesta opção.
 - Você acha que esse recurso pode auxiliar os alunos na aprendizagem do QuickSort?
 - Você acha que os alunos poderiam ter alguma interpretação diferente?

Opções: código e histórico

- 12) Observe as abas presentes na parte direita da tela.
- Descreva o que você acha que são as abas código e histórico.
 - Descreva como você acha que poderia interagir com elas. Pois bem, interaja.
 - Ao interagir, sua percepção mudou?
 - Você acha que os alunos poderiam ter alguma interpretação diferente?
- 13) Agora que você utilizou todos os recursos do QuickSort, você teria uma interpretação dos elementos presentes na interface diferente da que você tinha antes de utilizá-los?
- 14) Tem algum objeto a mais do software que você interpreta de maneira diferente de quando nós começamos a realização do método?

##Fim do Contato com o QuickSort recursivo##

##Utilizando o método de ordenação HeapSort##

- 15) Agora vamos retornar à tela principal e escolher o método de ordenação HeapSort. Vamos definir o tamanho do vetor 12 e selecionar o preenchimento aleatório.
- 16) Agora que estamos novamente na tela principal do ProfesSort, dê uma olhada nos **novos** objetos (botões, imagens, etc.) presentes na interface do software (que não tinham aparecido antes) e nos que já estavam e mudou.
 - a. Descreva o que você acha que eles significam e qual é a função de cada um deles.
 - i. Tente fazer isso em voz alta apenas para os elementos que são diferentes daqueles que você observou na tela do método do QuickSort recursivo.
 - b. Você acha que seus alunos terão interpretações diferentes das suas para algum desses objetos? Quais objetos e quais poderiam ser a interpretação dos alunos?

Iniciando o HeapSort

- 17) Observe as opções referentes às ações a tomar.
 - a. Descreva o que você acha que são as informações presentes nesta opção. Por favor, interprete as informações e os objetivos.
 - b. Agora interaja com essas opções. Sua percepção mudou? Caso tenha mudado, por gentileza, verbalize a mudança que houve após sua interação.
 - c. Você acha que os alunos poderiam interpretá-las de maneira diferente?
- 18) Observe que apareceu um novo elemento na interface para a visualização do Heap.
 - a. Apenas observando este elemento, diga para que você acha que ele serve e como pode interagir com ele.
 - b. Agora, vamos interagir com este elemento. E então? Sua percepção mudou após a interação?
 - c. Você acha que seus alunos poderiam ter uma interpretação diferente?
- 19) Agora que você utilizou todos os recursos do HeapSort, você teria uma interpretação dos elementos presentes na interface diferente da que você tinha antes de utilizá-los?
- 20) Tem algum objeto a mais do software que você interpreta de maneira diferente de quando nós começamos a realização do método?

##Fim do Contato com o HeapSort##

Perguntas pós-inspeção:

- 1) De novo:
 - a) O que este software faz?
 - b) Qual é o seu público (professor, aluno, ambos)?
 - c) Como utilizá-lo (professor, aluno, ambos)?
 - d) Qual seu objetivo educacional?

Ponto 3 – Potencial de uso do sistema

- Utilidade do sistema para o ensino/aprendizado de Algoritmos e Estrutura de Dados 2. Justificativa.
- Sistema poderia ser usado em seu curso? Como? Ou por que não?
- Este uso estaria adequado aos seus métodos de ensino atuais. Justificativa.
- Qual sua opinião sobre as possibilidades de apoio à aprendizagem oferecidas pelo sistema?
- Se os alunos de AEDS utilizassem este software como ferramenta de apoio à aprendizagem, qual pode ser o ganho com relação à aprendizagem sobre os métodos de ordenação?
- Recomendaria a outros?

Referências Bibliográficas

- Andres, D. (2005). Avaliação de Usabilidade nos Softwares Educacionais. *Logos Revista de Divulgação Científica*, pp. 111--117.
- Ardito, C.; Costabile, M.; Marsico, M.; Lanzilotti, R.; Levialdi, S.; Roselli, T. & Rossano, V. (2006). An approach to usability evaluation of e-learning applications. *Universal Access in the Information society*, 4(3):270--283.
- Ardito, C.; De Marsico, M.; Lanzilotti, R.; Levialdi, S.; Roselli, T.; Rossano, V. & Tersigni, M. (2004). Usability of e-learning tools. In *Proceedings of the working conference on Advanced visual interfaces*, pp. 80--84. ACM.
- Bento, L.; Prates, R. & Chaimowicz, L. (2009). Using Semiotic Inspection Method to Evaluate a Human-Robot Interface. In *2009 Latin American Web Congress*, pp. 77--84. IEEE.
- Borges, J. A. S. & Silva, G. P. (2006). Neanderwin – um simulador didático para uma arquitetura do tipo acumulador. In *Workshop Sobre Educação em Arquitetura de Computadores*, Ouro Preto - MG.
- Brush, T. & Saye, J. (2001). The Use of Embedded Scaffolds with Hypermedia-Supported Student-Centered Learning. *Journal of Educational Multimedia and Hypermedia*, 10(4):333--357.
- Carroll, J. (2003). *HCI Models, Theories, and Frameworks: Toward a multidisciplinary science*. Morgan Kaufmann Pub.
- Carroll, J.; Mack, R.; Robertson, S. & Rosson, M. (1994). Binding objects to scenarios of use. *International Journal of Human Computer Studies*, 41(1):243--276.
- Castro, T. & Fuks, H. (2009). Inspeção semiótica do ColabWeb: proposta de adaptações para o contexto da aprendizagem de programação. *Revista Brasileira de Informática na Educação*, 17(1):71.

- Cormen, T. H.; Leiserson, C.; Rivest, R. & C., S. (2001). *Introduction to Algorithms - 2nd Edition*. MIT Press.
- de Castro, I. H. L. & Prates, R. O. (2009). Professort: A sorting algorithms learning support system. In *XVII CIESC - XXXV CLEI*, Pelotas - RS.
- de Oliveira, E. R.; Luz, L. C. S. & Prates, R. O. (2008). Aplicação semi-estruturada do método de inspeção semiótica: Estudo de caso para domínio educacional. In *IHC'08: Proceedings of the VIII Brazilian Symposium on Human Factors in Computing Systems*, pp. 50--59, Porto Alegre, RS, Brazil. Sociedade Brasileira de Computação.
- de Souza, C.; Leitão, C.; Prates, R. & da Silva, E. (2006). The Semiotic Inspection Method. In *IHC'06: Proceedings of VII Brazilian Symposium on Human Factors in Computing Systems*, pp. 148--157, New York, NY, USA. ACM.
- de Souza, C. S. (2005). *The Semiotic Engineering of Human-Computer Interaction*. MIT Press, Cambridge MA, U.K.
- de Souza, C. S. & Leitão, C. F. (2009). *Semiotic Engineering Methods for Scientific Research in HCI*. Morgan and Claypool, University of Sheffield, U.K.
- de Souza, C. S.; Leitão, C. F.; Prates, R. O.; Bim, S. A. & da Silva, E. J. (2010). Can inspection methods generate valid new knowledge in hci? the case of semiotic inspection. *International Journal of Human Computer Studies - IJHCS*, 68:22--40.
- De Villiers, R. (2004). Usability evaluation of an e-learning tutorial: criteria, questions and case study. In *Proceedings of the 2004 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries*, pp. 284--291. South African Institute for Computer Scientists and Information Technologists.
- Denzin, N. & Lincoln, Y. (2000). The discipline and practice of qualitative research. *Handbook of qualitative research*, 2:1--28.
- Freitas, R. & Dutra, M. (2009). Usabilidade e Interatividade em Sistemas Web para Cursos Online. *Revista Brasileira de Informática na Educação*, 17(2).
- Gomes, A. S.; Castro F., J. A.; Gitirana, V.; Spinillo, A.; Alves, M.; Melo, M. & Ximenes, J. (2002). Avaliação de software educativo para o ensino de matemática. In *VII Workshop Informática em Educação*, Florianópolis - SC.

- Gravina, M. A. & Santarosa, L. M. (1998). A aprendizagem da matemática em ambientes informatizados. In *IV Congresso RIBIE*, Brasília - DF.
- Greenberg, S. & Buxton, B. (2008). Usability evaluation considered harmful (some of the time), in 'CHI'08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems'. *ACM, New York, NY, USA*, pp. 111--120.
- Hornbæk, K. (2006). Current practice in measuring usability: Challenges to usability studies and research. *International journal of human-computer studies*, 64(2):79--102.
- Hostins, H. & Raabe, A. L. A. (2007). Auxiliando a aprendizagem de algoritmos com a ferramenta webportugol. In *XV WEI, Anais do XXVII CSBC*, Rio de Janeiro - RJ.
- Iahad, N.; Dafoulas, G.; Kalaitzakis, E. & Macaulay, L. (2004). Evaluation of online assessment: The role of feedback in learner-centered e-learning. In *Proceedings of HICSS*. Citeseer.
- Jones, A.; Scanlon, E.; Tosunoglu, C.; Morris, E.; Ross, S.; Butcher, P. & Greenberg, J. (1999). Contexts for evaluating educational software. *Interacting with Computers*, 11(5):499--516.
- Lanzilotti, R.; Ardito, C.; Costabile, M. & De Angeli, A. (2006). eLSE Methodology: a Systematic Approach to the e-Learning Systems Evaluation. *Journal of Educational Technology and Society*, 9(4):42.
- Leitão, C.F. e Dias-Romão, D. (2003). Pesquisas em IHC: um debate interdisciplinar sobre a ética. In *Nicolaci-da-Costa, A. M. E. and Leite, J. C. Atas do Workshop sobre Interdisciplinaridade em IHC, CLIHC*.
- Lewis, C.; Brand, C.; Cherry, G. & Rader, C. (1998). Adapting user interface design methods to the design of educational activities. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 619--626. ACM Press/Addison-Wesley Publishing Co. New York, NY, USA.
- Lyra, A.; Leitão, D.; Amorim, G. & Gomes, A. (2003). Ambiente virtual para análise de software educativo. In *Anais do Workshop de Informática na Escola*, volume 1, p. 236.
- Masemola, S. & De Villiers, M. R. (2006). Towards a framework for usability testing of interactive e-learning applications in cognitive domains, illustrated by a case study.

- In *Proceedings of the 2006 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries*, p. 197. South African Institute for Computer Scientists and Information Technologists.
- Matera, M.; Costabile, M.; Garzotto, F.; Paolini, P. & e Inf, D. (2002). SUE inspection: an effective method for systematic usability evaluation of hypermedia. *IEEE Transactions on Systems, Man and Cybernetics, Part A*, 32(1):93--103.
- Mattos, B. A. M. (2010). *Uma extensão do método de avaliação de comunicabilidade para sistemas colaborativos*. PhD thesis, Dissertação de Mestrado. Programa de Pós Graduação em Ciência da Computação. Universidade Federal de Minas Gerais.
- Mattos, B. A. M.; Santos, R. L. & Prates, R. O. (2009). Investigating the applicability of the semiotic inspection method to collaborative systems. In *VI Simpósio Brasileiro de Sistemas Colaborativos*, Fortaleza - CE. Sociedade Brasileira de Computação.
- Molich, R. & Nielsen, J. (1990). Improving a human-computer dialogue. *Communications of the ACM*, 33(3):348.
- Morandi, D. & et al. (2006). Um processador básico para o ensino de conceitos de arquitetura e organização de computadores. In *Hifen*, volume 30, pp. 73--80, Uruguaiana - Brasil.
- Mota, M. P.; Pereira, L. W. K. & Favero, E. L. (2008). Javatool: uma ferramenta para ensino de programação. In *XVI WEI, Anais do XXVIII CSBC*, Belém - PA.
- Moura, J. R.; dos Santos, R. L.; de Oliveira, V. C.; Silva, R. F. & Prates, R. O. (2008). Avaliação da comunicabilidade e possíveis impactos sociais do orkut. *Competição de Avaliação do IHC*.
- Nicolaci-da Costa, A.; Leitão, C. & Romão-Dias, D. (2004). Como conhecer usuários através do Método de Explicitação do Discurso Subjacente (MEDS). *VI Simpósio Brasileiro sobre Fatores Humanos em Sistemas Computacionais, Curitiba*, pp. 47--56.
- Nielsen, J. (1994). Usability inspection methods. In *Conference companion on Human factors in computing systems*, pp. 413--414. ACM.
- Nielsen, J. & Molich, R. (1990). Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems: Empowering people*, pp. 249--256. ACM.

- Pardo, S.; Vetere, F. & Howard, S. (2006). Teachers' involvement in usability testing with children. In *Proceedings of the 2006 conference on Interaction design and children*, p. 92. ACM.
- Peirce, C.; Hartshorne, C. & Weiss, P. (1932). *Collected papers of Charles Sanders Peirce*. Harvard University Press.
- Pinelle, D. & Gutwin, C. (2000). A review of groupware evaluations. *IEEE 9th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2000.(WET ICE 2000). Proceedings*, pp. 86--91.
- Prates, R. (2004). A Engenharia Semiótica para o domínio educacional. In *Workshop de Design e Avaliação de Interfaces para Ambientes Educacionais. In: IHC*.
- Prates, R. & Barbosa, S. (2003). Avaliação de interfaces de usuário - conceitos e métodos. In *Jornada de Atualização em Informática*. Sociedade Brasileira de Computação.
- Prates, R. & Barbosa, S. (2007). Introdução à teoria e prática da interação humano computador fundamentada na engenharia semiótica. *T. Kowaltowski; KK Breitzman.(Org.). Atualizações em Informática 2007*, pp. 263--326.
- Prates, R.; de Souza, C. & Barbosa, S. (2000). Methods and tools: a Method for Evaluating the Communicability of user interfaces. *interactions*, 7(1):31--38.
- Prates, R.; Leitão, C. & Figueiredo, R. (2004). Desafios de Avaliação de Interfaces de Ambientes Educacionais—Um Estudo de Caso. *Anais do IHC 2004*.
- Preece, J.; Sharp, H. & Rogers, Y. (2004). *Interaction design: beyond human-computer interaction*. Apogeo Editore.
- Quintana, C.; Carra, A.; Krajcik, J. & Soloway, E. (2001). Learner-centered design: Reflections and new directions. *Human-computer interaction in the new millennium*, pp. 605--624.
- Quintana, C. & Fishman, B. (2006). Supporting science learning and teaching with software-based scaffolding. In *AERA*, San Francisco, CA.
- Quintana, C.; Krajcik, J. & Soloway, E. (2002). A case study to distill structural scaffolding guidelines for scaffolded software environments. In *Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves*, p. 88. ACM.

- Rentróia-bonito, A.; Guerreiro, T.; Martins, A.; Fern, V. & Jorge, J. (2008). Evaluating Learning Support Systems Usability An Empirical Approach. *Communication & cognition. Monographies*, 41(1-2):143--158.
- Squires, D. & Preece, J. (1996). Usability and learning: evaluating the potential of educational software. *Computers & Education*, 27(1):15--22.
- Squires, D. & Preece, J. (1999). Predicting quality in educational software: Evaluating for learning, usability and the synergy between them. *Interacting with computers*, 11(5):467--483.
- Squires, D. J. (1994). A comparison of learner and designer models in the use of direct manipulation educational software in the context of learning about interacting variables in photosynthesis.
- Ssemugabi, S. & de Villiers, R. (2007). A comparative study of two usability evaluation methods using a web-based e-learning application. In *Proceedings of the 2007 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries*, p. 142. ACM.
- Tergan, S. (1998). Checklists for the evaluation of educational software: critical review and prospects. *Innovations in Education and Teaching International*, 35(1):9--20.
- Triacca, L.; Bolchini, D.; Botturi, L. & Inversini, A. (2004). MiLE: Systematic usability evaluation for e-learning web applications. *EDMEDIA 2004, Lugano, Switzerland*, pp. 4398--4405.
- Tselios, N.; Avouris, N. & Komis, V. (2008). The effective combination of hybrid usability methods in evaluating educational applications of ICT: Issues and challenges. *Education and Information Technologies*, 13(1):55--76.
- Vieira, P. V.; Raabe, A. L. A. & Zeferino, C. A. (2009). Bipide: Ambiente de desenvolvimento integrado para utilização dos processadores bip no ensino de programação. In *XX Simpósio Brasileiro de Informática na Educação*, Florianópolis - SC. Sociedade Brasileira de Computação.
- Wharton, C.; Rieman, J.; Lewis, C. & Polson, P. (1994). The cognitive walkthrough method: A practitioner's guide. *Usability inspection methods*, pp. 105--140.
- Zaharias, P. (2006). A usability evaluation method for e-learning: focus on motivation to learn. In *CHI'06 extended abstracts on Human factors in computing systems*, p. 1576. ACM.

Zaharias, P.; Vasslopoulou, K. & Poulymenakou, A. (2001). Designing on-line learning courses: Implications for usability. *Retrieved October, 9:2006.*

Ziviani, N. (2004). Projeto de Algoritmos com Implementações em Pascal e C. *Thomson Pioneira.*

