

FASE: Uma Ferramenta para Automatização do Processo de Avaliação de Software

Maria de Fátima C. de Souza
Departamento de Computação -
Universidade Federal do Ceará (UFC)
Campus do Pici, Instituto UFC
Virtual- bloco 901 - 1º andar, CEP:
60455-760 - Fortaleza - CE - Brasil
+55 85 3366.9509
fatimasouza@lia.ufc.br

Mauro C. Pequeno
Departamento de Computação -
Universidade Federal do Ceará (UFC)
Campus do Pici, Instituto UFC
Virtual- bloco 901 - 1º andar, CEP:
60455-760 - Fortaleza - CE - Brasil
+55 85 3366.9509
mauro@vdl.ufc.br

José Aires de Castro Filho
Programa de Pós-Graduação em
Educação Brasileira - Universidade
Federal do Ceará (UFC).
Campus do Pici, Instituto UFC
Virtual- bloco 901 - 1º andar, CEP:
60455-760 - Fortaleza - CE - Brasil
+55 85 3366.9509
j.castro@ufc.br

RESUMO

Inúmeras são as metodologias fornecidas pela literatura para se avaliar um software educativo (SE). No entanto, essas metodologias abordam aspectos isolados a serem avaliados no software, acarretando uma baixa qualidade em seu resultado final. Além disso, o processo de avaliação adotado pela maioria delas, exigem do avaliador determinados conhecimentos técnicos que não condizem com o perfil da grande maioria dos professores, que são os principais interessados nesse processo. Dessa forma, propomos nesse trabalho uma ferramenta que automatiza o processo de avaliação, baseada em uma metodologia que considera simultaneamente diversos aspectos importantes ao SE, tornando-o transparente para o usuário/avaliador.

Categorias e Descritores de Assunto

D.2.8 [Engenharia de Software]: Métricas - *Medidas de Complexidade*.

D.3.3 [Computadores e Educação]: Usos do Computador em Educação - *Instrução Assistida por Computador (CAI), Instrução Gerenciada por Computador (CMI)*.

Termos Gerais

Medição, Fatores Humanos e Verificação.

Palavras Chave

software educativo, avaliação, metodologia, ferramenta.

1. INTRODUÇÃO

Atualmente, existem diversas metodologias propostas na literatura para avaliação de software educativo (SE). Essas inúmeras metodologias visam colaborar com uma seleção mais adequada de um software desenvolvido com propósitos educacionais. Entenda-se por seleção mais adequada, aquela que seja capaz de auxiliar o avaliador a extrair dados considerados relevantes que levem em conta as necessidades do professor e do aluno. Diante do desafio de se conseguir selecionar um SE que seja o mais adequado a uma determinada aplicação, estamos propondo nesse trabalho uma ferramenta que irá automatizar o processo de avaliação, permitindo que o professor/avaliador consiga identificar em um

SE as características que sejam pertinentes ao aprendizado, respeitando sua perspectiva dentro do contexto escolar.

A ferramenta proposta nesse trabalho, denominada FASE (Ferramenta de Avaliação Automática de Software Educativo), implementa uma metodologia que se baseia no processo de técnicas utilizadas em engenharia de requisitos proposto por Sommerville [12], juntamente com a Teoria dos Campos Conceituais de Vergnaud [16]. Esta ferramenta possibilita realizar uma avaliação de SE, permitindo ao avaliador identificar todas as características que sejam pertinentes ao aprendizado, visto que a metodologia subjacente à ferramenta tem como ênfase a análise tanto de aspectos cognitivos como de aspectos de usabilidade, fornecendo uma visão ampla do processo de avaliação.

Através da utilização da FASE, o professor poderá realizar avaliações de diversos produtos de SE sem a necessidade de entender explicitamente os conceitos de engenharia de requisitos nem de aspectos cognitivos. Tais conceitos estão embutidos na metodologia de avaliação implementada pela ferramenta, visto que os termos técnicos utilizados pela ferramenta são apresentados de forma implícita em FASE, permitindo que seja utilizada a linguagem cotidiana do professor/avaliador na realização do processo de avaliação de seus produtos de software.

Na seção seguinte, apresentamos de forma concisa, algumas das metodologias mais utilizadas na avaliação de SE. Na seção 3, apresentamos a metodologia de avaliação que propomos para a avaliação comparativa de SE. Na seção 4, apresentamos a ferramenta que automatiza a metodologia de avaliação que propomos. Para finalizar apresentamos na seção 5 uma discussão sobre as vantagens do processo de automatização do processo de avaliação de SE além de algumas conclusões.

2. METODOLOGIAS DE AVALIAÇÃO DE SE

Nesta seção, iremos tratar de cinco metodologias apresentadas na literatura apropriadas para avaliação de SE. Essas metodologias buscam formas de orientar os professores ou responsáveis na instituição de ensino, na aquisição de software para o uso em contexto escolar. As metodologias que iremos abordar são: Reeves [8], Ergolist [9], Ticese [3], Maep [10] e Softmat [2].

A metodologia Reeves [8] trabalha com duas abordagens, sendo a primeira baseada em quatorze critérios e a outra baseada em dez critérios, ambos relacionados à interface com o usuário.

Já os métodos Ergolist [9] e Ticese [3], ambos da Universidade Federal de Santa Catarina, sendo que o primeiro caracteriza-se pela avaliação levando-se em conta a qualidade ergonômica com o uso de um checklist on-line. O segundo encontra-se em desenvolvimento e tem como base o fornecimento de um laudo técnico ao final da avaliação do software relativo a apenas aspectos ergonômicos/pedagógicos.

A metodologia Maep [10] é um modelo de avaliação que procura agregar tanto aspectos pedagógicos quanto ergonômico do software. Esses aspectos são analisados através de um checklist básico voltado para a avaliação do software educacional. Além desses dois aspectos, avalia-se também o material de apoio do software.

Por fim, o método Softmat [2], é um repositório virtual específico para software de matemática voltados para o ensino médio. Esse repositório comporta softwares acompanhados com suas respectivas avaliações. Essa metodologia funciona através de um questionário disposto em cinco blocos de questões, onde são considerados aspectos técnicos e específicos do setor educacional.

3. NOSSA METODOLOGIA DE AVALIAÇÃO

A nossa abordagem para o processo de avaliação de SE, apresentada em [12], fornece um mecanismo de avaliação que abrange tanto aspectos cognitivos/pedagógicos, como aspectos funcionais e de qualidade, possibilitando captura simultânea de diversos aspectos relevantes a uma avaliação de SE.

Em nossa metodologia, fizemos uso de conceitos próprios da engenharia de requisitos, como os requisitos funcionais, não funcionais e de domínio, de modo a contextualizar todos os aspectos da avaliação.

Nos requisitos funcionais, capturamos todas as declarações de funcionalidades que o SE deve fornecer. Já nos requisitos não funcionais, temos as restrições sobre os serviços ou as funcionalidades do sistema, particularmente primamos pelos aspectos de usabilidade. Finalmente, nos requisitos de domínio, capturamos características relevantes aos aspectos pedagógicos/cognitivos, aplicando a teoria dos campos conceituais de Vergnaud com o propósito de elicitá-los.

Adicionalmente, aplicamos a técnica de pontos de caso de uso [4] para gerar pontuações sob os SE avaliados, possibilitando a comparação entre os produtos e permitindo uma melhor seleção desses produtos por parte do professor/avaliador. Na Figura 1, apresentamos de forma simplificada a nossa metodologia de avaliação.

Seqüencialmente, as atividades propostas pela metodologia compreendem: selecionar o software candidato, fazer o levantamento de requisitos funcionais, levantamento de requisitos de domínio, definir e levantar os requisitos não funcionais, calcular as pontuações dos requisitos funcionais e de domínio, definir e calcular as métricas dos requisitos não funcionais, fazer o cálculo da avaliação final e por fim fazer a análise dos

resultados obtidos. Em [13], podem ser encontrados mais detalhes sobre essa metodologia, inclusive com exemplos de sua aplicação.

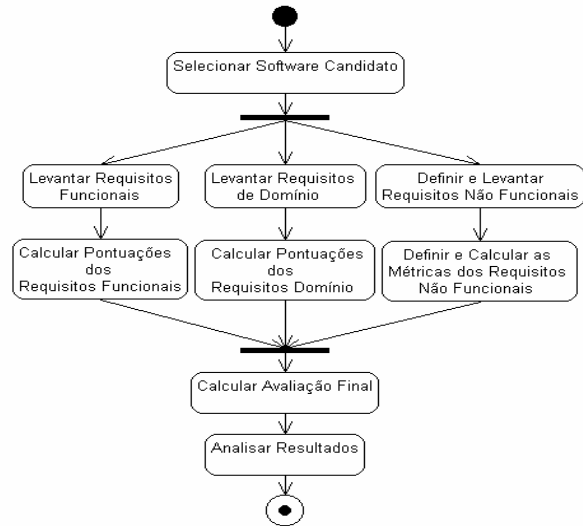


Figura 1 - Atividades da Metodologia de Avaliação Comparativa

4. FERRAMENTA DE AVALIAÇÃO AUTOMÁTICA DE SOFTWARE EDUCATIVO (FASE)

O objetivo principal da ferramenta que iremos apresentar é permitir que um professor seja capaz de realizar uma escolha mais minuciosa de quais SEs utilizar em sala de aula para uma determinada área de aplicação da matemática (Campo Conceitual). Nessa ferramenta, não se limita a levar em consideração para aferição de um software, apenas aspectos relacionados à qualidade da interface gráfica, que em muitos casos acabam escondendo outros aspectos importantes com relação a usabilidade do software, mas também permitirá se aferir outras qualidades de um produto através dos seus requisitos funcionais, considerando as funcionalidades oferecidas e de seus requisitos de domínio, que dizem respeito a fatores cognitivos/pedagógicos.

O uso da ferramenta FASE irá permitir ao avaliador/professor selecionar esses requisitos de maneira transparente, através de respostas a algumas questões sobre aspectos pertinentes aos SEs que estão sendo avaliados. Essas questões serão feitas de maneira isolada para cada requisito a ser avaliado.

4.1 PROJETO E ESPECIFICAÇÃO FUNCIONAL

Nessa primeira versão da FASE as avaliações dos softwares são realizadas de forma independente, ou seja, cada software é avaliado separadamente e seus resultados são apresentados de maneira isolada, ficando a cargo do avaliador realizar a escolha dos produtos considerando esses resultados.

É válido ressaltar que essa ferramenta deverá ser utilizada paralelamente ao software que está em avaliação. Em outras palavras, FASE permitirá ao avaliador registrar as informações sobre os softwares à medida que esses estão sendo utilizados. Ao final, a ferramenta emite um diagnóstico com o resultado da avaliação do software. Este diagnóstico fica armazenado para posterior consulta

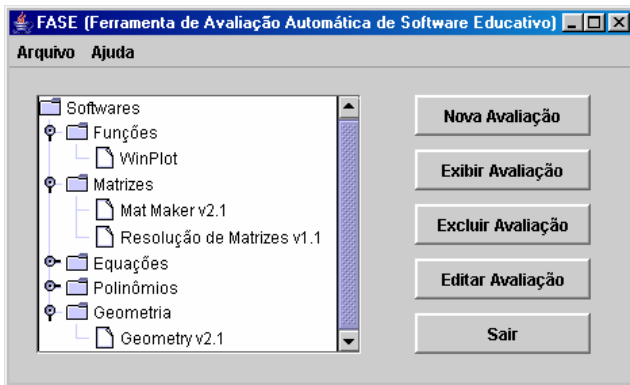


Figura 2 - Tela Principal da FASE

A Figura 2 apresenta a Tela Principal da ferramenta (FASE). Nessa tela, serão apresentados todos os softwares que já foram avaliados anteriormente, podendo o usuário selecionar um dos softwares na lista e utilizar os botões para exibir sua avaliação, excluí-lo da lista ou modificar a avaliação realizada. Além disso, o usuário poderá realizar uma nova avaliação, onde deverão ser fornecidos os dados gerais relativos ao software a ser avaliado.

Após ser definida a necessidade de uma nova avaliação, será apresentada a tela de dados gerais (Figura 3). Nessa tela, devem ser fornecidas informações como: nome do software, versão, data da avaliação, nome do avaliador e área de aplicação do software.

Vale ressaltar, que todas essas informações solicitadas na tela de dados gerais são normas pré-estabelecidas pela nossa metodologia, apresentada na seção 3.

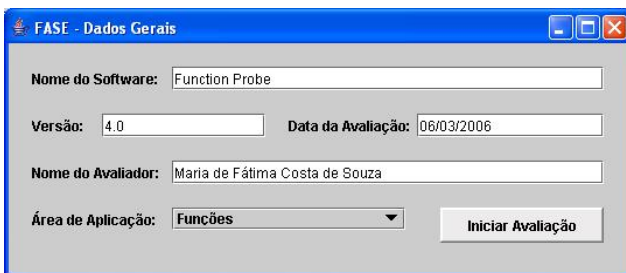


Figura 3 - Tela de Dados Gerais

4.1.1 Avaliação Funcional

Por se tratar de um termo específico da engenharia de software, requisito funcional não poderá ser tratado, ou melhor, mencionado dessa forma em nossa ferramenta. Como o próprio nome diz, requisitos funcionais são declarações de funcionalidades que o sistema deve fornecer, como o sistema deve agir a entradas específicas e como deve se comportar em determinadas situações, ou em alguns casos podem também declarar explicitamente o que o sistema não deve fazer [12].

Diante dessa definição, resolvemos “traduzir” todas essas informações técnicas de maneira que essa ferramenta seja viável a qualquer avaliador, isto é, o conceito de requisito funcional não será obstáculo para que a avaliação seja feita.

Para tornar a atividade de definição de requisitos funcionais, que são na verdade casos de uso do sistema em questão, fornecemos para cada área de atuação dos SE (funções, matrizes, equações etc), um conjunto mínimo de funcionalidades que podem ser utilizadas no processo de avaliação. Além disso, permitimos que o avaliador possa inserir novas funcionalidades, que na sua opinião, são importantes de serem levantadas. A Figura 4 apresenta a interface onde será feita a avaliação dos requisitos funcionais (Funcionalidades).

Nessa figura, apresentamos a tela de avaliação de funcionalidades da FASE. Observe que na figura 4 já apresentamos algumas funcionalidades, sendo que a primeira delas já é padrão para a área de funções e as demais foram adicionadas pelo avaliador, através do botão “Adicionar”.

Tivemos cuidado em tomar essa medida, visto que o nosso intuito é permitir que todo e qualquer avaliador seja capaz de manusear a ferramenta e não somente aqueles que tivessem o conhecimento em engenharia de requisitos.



Figura 4 - Tela de Avaliação Funcional

Adicionalmente, para cada funcionalidade listada na tabela da figura 4 devemos pontuar cada uma com valores entre 0 a 5, de acordo com o grau de importância que o avaliador atribuirá a funcionalidade (requisito). Sempre levando em consideração que 0 representa o requisito de menor importância e 5 o requisito de maior importância. Esses valores são utilizados em nossa metodologia de avaliação para classificar e pontuar os casos de usos, de forma a se aplicar à técnica de pontos de caso de uso para gerar o valor da avaliação do software.

Para melhor exemplificar a utilização dessa interface, suponha que queiramos avaliar um software na área de funções. Assim, as possíveis funcionalidades referente à essa área de atuação serão as seguintes:

- Possui Representação Formal?
- Possui Representação em forma de Tabela?
- Apresenta Representação Gráfica?

Para cada funcionalidade relacionada acima, será solicitada uma pontuação de 0 a 5 de acordo com o grau de importância do avaliador.

Como apresentado em [14] o cálculo referente aos requisitos funcionais é tratado dentro da nossa metodologia de avaliação, como sendo equivalente aos Pontos de Casos de Uso Não Ajustados, utilizados na técnica de pontos de casos de uso, ou seja, de posse de todas as informações referentes a esses requisitos, a FASE utilizará os valores associados a esses dados [15] para calcular de maneira pontual o valor desse requisito.

4.1.2 Avaliação de Qualidade

Pelo mesmo motivo explicado anteriormente em 4.1.1, requisitos não funcionais (ou requisitos de qualidade) não serão citados explicitamente em nossa ferramenta. Dentro da engenharia de requisitos esse termo significa restrições sobre os serviços ou as funções oferecidos pelo sistema [11]. Entre eles destacam-se restrições sobre o processo de desenvolvimento, padrões, entre outros. Pelo mesmo motivo citado anteriormente, “traduzimos” essas definições técnicas de maneira a torná-la compreensível a qualquer avaliador, ou seja, o conceito formal não será obstáculo para que a avaliação seja realizada.

Os fatores de qualidade dizem respeito aos requisitos não funcionais dos SE que estão sendo avaliados. Como esses requisitos devem ser avaliados através de métricas onde valores devem ser associados a cada requisito baseado na utilização do software, resolvemos, inicialmente, permitir a definição por parte do avaliador de quais serão os fatores que devem ser avaliados no software. Na figura 5, apresentamos a interface de definição de parâmetros de qualidade.

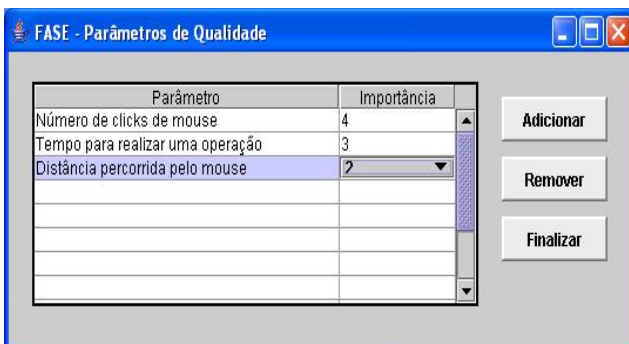


Figura 5 – Parâmetros de Qualidade

Da mesma forma que fizemos para a avaliação funcional, oferecemos para o avaliador um conjunto mínimo de parâmetros de qualidade que são gerais para qualquer tipo de software. Para exemplificar utilizamos o requisito usabilidade. Assim, esses parâmetros, que na verdade são métricas de usabilidade, serão avaliados automaticamente quando essa interface for finalizada, através do pressionamento do botão “Finalizar”.

Para realizar a aferição de cada um dos parâmetros listados, a FASE gerará automaticamente uma tela para cada um dos parâmetros, de forma que o avaliador possa capturar os valores de cada métrica.

Para exemplificar essa funcionalidade de FASE observe que o primeiro parâmetro listado na figura 5 é o número de clicks de mouse. Para essa funcionalidade a FASE gera uma tela,

apresentada na figura 6, onde podemos observar a presença do botão “Iniciar” e “Finalizar”.

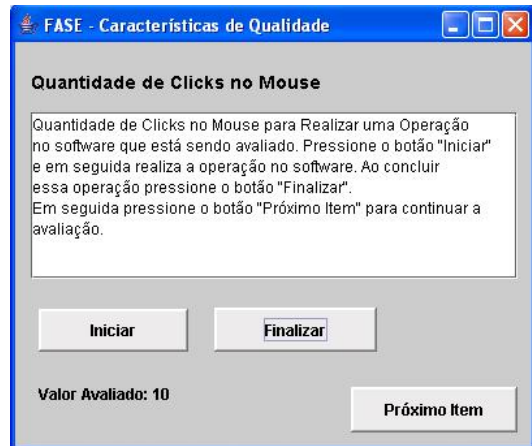


Figura 6 – Aferição de Requisito Não Funcional

Esses botões devem ser pressionados no início e no final, respectivamente, da realização de uma operação simples no software que está sendo avaliado. Ao pressionar o botão “Iniciar” a FASE captura qualquer click realizado no mouse até que seja pressionado o botão “Finalizar”.

Assim essa métrica é aferida e armazenada. Essa mesma técnica é utilizada para as demais métricas definidas (parâmetros) pelo avaliador. Ou seja, ao pressionar o “Próximo Item” será iniciada a avaliação da próxima métrica. Isso se repete até que sejam avaliados todos os parâmetros definidos previamente.

Como apresentado em [14] o cálculo referente aos requisitos não funcionais é tratado dentro da nossa metodologia de avaliação como sendo equivalente aos Fatores Ambientais, utilizados na técnica de Pontos de Casos de Uso, assim, a FASE utilizará os valores associados a esses requisitos para calcular de maneira pontual o valor dos mesmos.

4.1.3 Avaliação Pedagógica

Como explicado anteriormente, requisito de domínio é um termo próprio da engenharia de requisitos. Eles nada mais são que requisitos que definem funções específicas de determinados domínios de aplicação [11]. Esses requisitos tanto podem ser funcionais como não funcionais. Em nossa metodologia de avaliação tratamos os requisitos de domínio como requisitos relativos a fatores pedagógicos/cognitivos, onde fizemos uso da Teoria dos Campos Conceituais [16] para abordá-los.

Na ferramenta de avaliação, esses requisitos são apresentados ao avaliador como um conjunto de opções onde o avaliador irá selecionar as suas situações, as suas variantes e as suas representações, que são conceitos trabalhados na Teoria dos Campos Conceituais.

A figura 7 ilustra a situação descrita anteriormente.

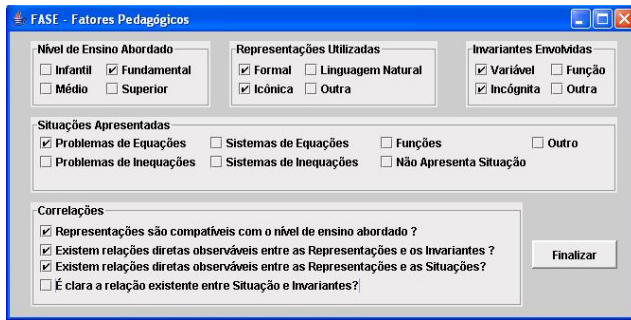


Figura 7– Fatores Pedagógicos

Em nossa metodologia fazemos uso de uma tabela [13] que denominamos de tabela de requisitos de domínio. Nesta tabela, o avaliador será capaz de fazer suas seleções, sem notar que suas escolhas, nada mais são que a seleção de requisitos de domínio. É válido ressaltar que utilizamos também a noção de correlação [13], que permite se avaliar as relações entre fatores relacionados à Teoria dos Campos Conceituais.

A avaliação desses requisitos dentro da FASE se dará da seguinte maneira:

- avaliador deverá informar o nível de ensino abordado, ou seja: infantil, fundamental, médio, superior ou não se aplica.
- Depois, deverá selecionar a situação apresentada, ou seja: problemas de equações, problemas de inequações, funções, sistema de equações, sistema de inequações ou outro.
- Em seguida, o avaliador deverá informar o(s) Tipo(s) de Invariantes envolvidas, ou seja: variáveis, incógnitas, equações, inequações, funções ou outro.
- Posteriormente, deverá selecionar qual o tipo de representação foi utilizado para as Invariantes envolvidas, ou seja: formal, linguagem natural ou icônica.
- Por fim, feitas às seleções acima, o avaliador deverá observar as correlações [14] existentes entre Situações, Invariantes e Representações. Essas correlações serão fornecidas ao avaliador através de uma lista de perguntas, onde ele (avaliador) irá responder com SIM ou NÃO.

Para exemplificar, podemos usar as seguintes perguntas para avaliar as correlações:

- As Representações são compatíveis com o nível de ensino abordado?
- Existem relações diretas observáveis entre as Representações e as Situações?
- Existem relações diretas observáveis entre as Representações e os Invariantes?
- É clara a relação existente entre Situação e Invariantes?

Como também apresentado em [14] o cálculo referente aos requisitos de domínio é tratado dentro da nossa metodologia de

avaliação, como sendo equivalente aos Fatores Técnicos, utilizados na técnica de pontos de casos de uso. Da mesma forma apresentada para os demais tipos de requisitos, a FASE utilizará os valores associados a esses dados [15] para também calcular de maneira pontual esse requisito.

4.1.4 Avaliação Final

Após realizada a avaliação de todos os requisitos necessários, o passo seguinte é verificar o resultado final da avaliação. A Figura 8 apresenta a tela de resultado final da avaliação. Esse resultado é o produto das avaliações dos requisitos de acordo com a metodologia proposta [13]. Uma explicação mais detalhada sobre esse cálculo e sobre a aplicação dessa técnica para a realização de avaliação de SE, pode ser encontrada em [13, 14].



Figura 8 – Resultado Final de Avaliação

4.2 IMPLEMENTAÇÃO

Todas as interfaces de levantamento de informações de avaliação da FASE já foram implementadas. A implementação da FASE está sendo realizada em Java 1.5.0 [5] através da IDE NetBeans 5.0 [7].

A definição dos arquivos XML que dão suporte a geração de interfaces para a avaliação e que armazenam os dados das avaliações está sendo realizada através do XML Spy 2005 [1].

Está em fase de desenvolvimento as operações de criação automática de interfaces para a extração de requisitos não funcionais. Essas operações devem permitir a definição de que eventos devam ser capturados e de como as interfaces devam ser geradas para se conseguir aferir os valores de requisitos não funcionais. Atualmente, apenas um conjunto fixo de requisitos não funcionais estão disponíveis em FASE.

5. CONCLUSÃO

As metodologias de avaliação de SE disponíveis atualmente realizam suas análises focando em distintos fatores, muito embora cada metodologia tenha ao final um único objetivo, identificar um software que seja capaz de atender as reais necessidades do usuário. Logicamente, isso acaba introduzindo fatores de distorção, que podem comprometer o resultado final da avaliação.

Em [13,14] argumentamos que é de suma importância o tratamento de aspectos funcionais, não funcionais e de domínio de maneira concomitante para possibilitar a realização de uma avaliação voltada para a seleção de software de mesma área de aplicação. Dessa forma, desenvolvemos uma metodologia para

avaliação de SE utilizando técnicas de engenharia de requisitos e a Teoria dos Campos Conceituais de Vergnaud de forma integrada, capturando simultaneamente diversos fatores relevantes ao processo de avaliação de SE. Entretanto, os professores por não serem especialistas na área de avaliação de software, têm dificuldade de aplicar os conceitos para uma boa avaliação.

Dessa forma, apresentamos nesse trabalho, uma ferramenta que automatiza a metodologia que propomos para a avaliação de SE, facilitando sobremaneira o trabalho do professor. Essa ferramenta, denominada FASE, permite que o próprio avaliador escolha quais requisitos são importantes para ele e para cada requisito escolhido deverá ser dada uma pontuação que indicará o grau de prioridade do requisito dentro dessa avaliação.

Dentre as contribuições dessa proposta podemos destacar o tratamento automatizado do processo de avaliação de SE. Nesse sentido, como a avaliação do SE deve ser realizada com a participação do professor [8], ou seja, esse deverá ser o fornecedor dos requisitos a serem atendidos, a automatização desse processo de avaliação permite que os professores possam participar de forma mais efetiva desse processo, no sentido de que aspectos técnicos relacionados à engenharia de software, por exemplo, possam ser tratados de forma transparente.

6. REFERÊNCIAS

- [1] Altova 2005. XML Spy 2005. Disponível em: http://www.altova.com/products_ide.html. Acessado em: 20/07/2005.
- [2] Batista, S.C.F. Softmat: Um Repositório de Softwares para Matemática do Ensino Médio – Um Instrumento em Prol de Posturas mais Conscientes na Seleção de Softwares Educacionais. Rio de Janeiro, 2004. Dissertação (Mestrado em Ciências de Engenharia), Universidade Estadual do Norte Fluminense.
- [3] Gamez, Luciano. Técnica de Inspeção de Conformidade Ergonômica de Software Educacional. Minho-Protugal, 1998. Dissertação (Mestrado em Engenharia Humana) – Programa de Pós-Graduação em Engenharia Humana, Universidade do Minho.
- [4] Karner, G. Metrics for Objectory. (Diploma Thesis, University of Linköping, Sweden).
- [5] Microsystems, S. Java 2 Platform, Standard Edition (J2SE). Disponível em: <http://java.sun.com/j2se/index.jsp>
- [6] NanoXML 2005. NanoXML 2.2.1. Disponível em: <http://nanoxml.sourceforge.net/orig/>. Acessado em: 20/07/2005.
- [7] NetBeans 2005. NetBeans 4.1. Disponível em: <http://www.netbeans.org/>. Acessado em: 20/07/2005.
- [8] Oliveira, Silvia S. de and Gomes, Alex S. and Borges Neto, Hermínio. Avaliação de Software educativo para o Ensino de Matemática – O Caso das Estruturas aditivas. In XV Encontro de Pesquisa Educacional do Nordeste. São Luís, Brasil, 2001.
- [9] Reeves, Thomas. Systematic Evaluation Procedures for Interactive Multimedia for Education and Training. Multimedia Computing: preparing for the 21 st century. Harrisburg, PA. Idea Group. 1994.
- [10] Silva, Cassandra R. and Vargas, Carlos L. S. Avaliação da Qualidade de Software Educacional. XIX Encontro Nacional de Engenharia de Produção e V Internacional Congress of Industrial Engineering. Anais CD-ROM. Rio de Janeiro, 1999.
- [11] Silva, Cassandra R. O. E. MAEP: Um Método Ergopedagógico Interativo de Avaliação para Produtos Educacionais Informatizados. Tese (Doutorado) – Universidade Federal de Santa Catarina – Programa de Pós-graduação em Engenharia de Produção, 2002.
- [12] Sommerville, I. Software Engineering (7th ed). Harlow: Addison-Wesley. 2004
- [13] Souza, Maria de Fátima C. de and Castro Filho J. A. and Pequeno, Mauro Cavalcante . Uma abordagem semi-automática para a avaliação comparativa de software educacional de matemática. In: XV - SBIE - Simpósio Brasileiro de Informática na Educação, 2004, Manaus. Anais do Simpósio Brasileiro de Informática na Educação, 2004. Lafort, L. LaTeX User's Guide and Document Reference Manual. Addison-Wesley, Reading, MA, 1986.
- [14] Souza, Maria de Fátima C. de, and Pequeno, Mauro C., and Castro Filho, J.ª and Souza, Cidley T. Uma Metodologia de Apóio à Seleção de Softwares Educativos para o Ensino de Matemática. Revista Latinoamericana de Tecnologia Educativa. Vol. 3 Número 2. Pág. 61 a 83. ISSN - 1695-288X. 2004
- [15] Souza, Maria de Fátima Costa de ; Pequeno, Mauro Cavalcante ; Castro Filho, J. A. . A Software Evaluation Approach Based on Vergnaud s Conceptual Fields Theory. In: The 5th IEEE International Conference on Advanced Learning Technologies, 2005, Kaohsiung, Taiwan. Proceedings of The 5th IEEE International Conference on Advanced Learning Technologies, 2005.
- [16] SUN Microsystems 2005. Java 2 Platform, Standard Edition (J2SE). Disponível em: <http://java.sun.com/j2se/index.jsp> . Acessado em: 20/07/2005.
- [17] Vergnaud, G. La Théorie des Champs Conceptuels. Recherches en Didactique des Mathématiques. 1990.