

# **AVALIAÇÃO DE INTERFACES DE FERRAMENTAS CASE PARA ELABORAÇÃO DE DIAGRAMAS DA UML**

Fábio Junior Alves  
Ernani Vinícius Damasceno  
Leandro Pinheiro Bevenuto

## **RESUMO**

Um dos benefícios oferecidos pelas ferramentas CASE é orientar e disciplinar o processo de modelagem durante da fase de desenvolvimento do projeto de um software. Porém, muitas vezes a ferramenta não é totalmente explorada do ponto de vista funcional, fundamentalmente pela relativa complexidade do assunto e mesmo por problemas de interfaces. Desta forma, o artigo visa analisar, através de heurísticas de usabilidade, a interface de algumas ferramentas CASE que são utilizadas para elaboração dos diagramas da UML, buscando verificar a contribuição desse aspecto na área de Computação.

**PALAVRAS-CHAVE:** Ferramentas CASE, Diagrama, UML

## **1 - INTRODUÇÃO**

De acordo com Booch et al. (2005) a UML (*Unified Modeling Language* – Linguagem de Modelagem Unificada), é uma linguagem gráfica padrão para especificação, visualização, construção e documentação de artefatos de um sistema. Atualmente a UML vem se tornando a cada dia um padrão para análise e projeto de software.

Segundo Melo (2004) a UML fornece vários diagramas para modelagem de sistemas, os quais apresentam um resultado passível de compreensão pelo usuário e

programador, facilitando as etapas de análise, projeto e implementação do sistema. Para utilizar seus diagramas é necessário o uso de uma ferramenta CASE (*Computer Aided Software Engineering* – Engenharia de Software Auxiliada por Computador) que são ferramentas que auxiliam os desenvolvedores nas diversas fases do projeto de um sistema ou no gerenciamento do projeto (Monteiro, 2004).

De acordo com Pressman (2002), a interface pode ser considerada um dos elementos mais importantes na interação humano computador, pois envolve aspectos que influenciam o usuário na realização de uma tarefa. Sendo assim, uma interface bem definida entre o usuário e uma ferramenta é fundamental para que as funções desta ferramenta possam ser exploradas com facilidade.

É fato que a necessidade do uso das ferramentas CASE para dar suporte às atividades de desenvolvimento de software é inquestionável. Porém, a produtividade muitas vezes não é explorada suficientemente pela dificuldade de interação entre o usuário e a ferramenta CASE.

O objetivo deste artigo é apresentar uma avaliação da interface de algumas ferramentas CASE que auxiliam a construção de diagramas da UML, utilizando-se de determinadas heurísticas de usabilidade, abordadas em (Nielsen, 1993).

## **2 - FUNDAMENTAÇÃO TEÓRICA**

Esta seção tem por objetivo definir e apresentar a base conceitual fundamentando os pontos essenciais para o trabalho realizado. Sua adequada elaboração tem o propósito de facilitar o entendimento do assunto aqui tratado.

### **2.1 FERRAMENTAS CASE**

As ferramentas CASE são utilizadas para apoiar a realização de atividades específicas, tais como planejamento e análise e especificação de requisitos (Pressman, 2002).

Apesar dos benefícios do uso de ferramentas CASE individuais, atualmente, o número e a variedade de ferramentas têm crescido a tal ponto que levou os engenheiros de software a pensarem não apenas em automatizar os seus processos, mas sim em trabalhar com diversas ferramentas que interajam entre si e forneçam suporte a todo ciclo de vida do desenvolvimento.

As seções que seguem apresentam três ferramentas CASE que são utilizadas na elaboração de diagramas da UML.

### **2.1.1 ArgoUML**

Segundo Monteiro (2004) a ArgoUML é uma ferramenta CASE baseada na notação UML, que foi desenvolvido pela comunidade de desenvolvedores de código livre Tigris. Ele roda na maior parte das plataformas por ser implementado em Java.

De acordo com Vieira (2004) a ferramenta ArgoUML é uma das principais ferramentas open source de modelagem UML e inclui suporte para todos os seguintes diagramas da UML: Diagrama de classe, Diagrama de estados, Diagrama de Atividades, Diagrama de casos de uso, Diagramas de colaboração, Diagrama de Utilização/Componentes e Diagrama de sequência.

Segundo a revista Java Magazine (2006) o grande diferencial do ArgoUML em relação a outras ferramentas CASE são os recursos cognitivos embutidos no produto. Em vez de ser apenas um diagramador, documentador e gerador de código, o ArgoUML procura orientar e auxiliar o desenvolvedor na construção dos modelos. Esta ajuda provém de várias regras que são aplicadas continuamente, verificando inconsistências, erros comuns e sugerindo próximos passos.

### 2.1.2 JUDE

O JUDE é uma ferramenta IDE de desenvolvimento e modelagem de dados (UML), criada sobre a linguagem Java e de uso fácil e intuitivo. Com o JUDE é possível realizar uma modelagem de dados complexa, apresentando os dados para o usuário de clara e objetiva. Após feita toda a modelagem é possível exportar tudo para um arquivo Java, HTML ou exportar cada diagrama em forma de imagem.

Segundo o site da Jude (2011) o JUDE/Community é uma ferramenta de modelagem UML gratuita com funcionalidades básicas. Ele é rico em funcionalidade, oferecendo características como edição e impressão de diagramas UML 2.0, importa/exporta código fonte Java, saída de gráficos e disposição automática.

Segundo o site da Jude (2011) os diagramas padrão da UML 1.4 e UML 2.0 que o JUDE/Community trabalha são: Diagrama de Classes, Diagrama de Casos de Uso, Diagrama de Estados, Diagrama de Atividades, Diagrama de Seqüência, Diagrama de Comunicação, Diagrama de Componentes e Diagrama de Estruturas Compostas.

Segundo o site da Jude (2011) o JUDE/Professional, além de aprimoramentos adicionais, é uma ferramenta para projetar sistemas com todas as funcionalidades do JUDE/Community. É aprimorado para o desenvolvimento em negócios, grandes moldes e criação de documentos. Ele provê diagramas UML 2.0 e diagramas adicionais, funcionalidade de impressão melhorada, habilidade de fazer merge com

outros projetos JUDE, especificações de caso de uso, *input-output* de modelos de/para arquivos XML, funções de copiar e colar copy em formato de vetor (EMF), e exportação de informações do projeto em formato CSV.

Segundo o site da Jude (2011) os diagramas padrão da UML 1.4 e UML 2.0 que o JUDE/Professional trabalha são: Diagrama de Classes, Diagrama de Casos de Uso, Diagrama de Estados, Diagrama de Atividades, Diagrama de Seqüência, Diagrama de Comunicação. Diagrama de Componentes, Diagrama de Implantação e Diagrama de Estruturas Compostas.

### **2.1.3 MICROSOFT VISIO**

O Microsoft Visio é um aplicativo gráfico e de desenho, que simplifica o processo de criação de diagramas e gestão de processo. É possível utilizar o Visio para transformar textos e tabelas complicadas e de difícil entendimento em diagramas que comunicam rapidamente as informações. São dezenas de ferramentas e templates prontos para serem utilizados.

O Visio fornece formas e modelos modernos para um conjunto diverso de necessidades de criação de diagramas, incluindo gerenciamento de TI, modelagem de processos, construção e arquitetura, design de interface do usuário, gestão de recursos humanos, gerenciamento de projetos e muito mais, podendo ser utilizado conjuntamente com o *SharePoint* para compartilhamento de seus diagramas com pessoas da mesma empresa.

Segundo o site Microsoft Office (2011) o Microsoft Visio fornecer formas de inteligência predefinidas que representam os elementos na notação UML e oferecem suporte para a criação dos seguintes tipos de diagramas UML: Diagramas de casos, Diagramas de estruturas estáticas, Diagramas de pacote, Diagramas de atividade,

Diagramas de gráfico de estado, Diagramas de seqüência, Diagrama de colaboração, Diagramas de componentes e Diagramas de implantação.

## **2.2 - A IMPORTÂNCIA DA INTERAÇÃO HUMANO-COMPUTADOR E SUAS TÉCNICAS DE AVALIAÇÃO**

Segundo a norma a norma ISO 9241, usabilidade é a “capacidade que um sistema interativo oferece a seu usuário, em um determinado contexto de operação, para a realização de tarefas de maneira eficaz, eficiente e agradável”. Já segundo a norma ISO/IEC 9126, usabilidade é a “facilidade com que um usuário pode aprender a operar, preparar entradas para e interpretar as saídas de um sistema ou componente”.

Simplificando, podemos dizer que a usabilidade está associada a uma característica de qualidade de software que se refere à sua adequação à utilização pelos usuários. A usabilidade trata da qualidade da interação usuário-computador proporcionada pela interface de um sistema de computação. É importante salientar que a usabilidade está sempre associada a um contexto de utilização do produto; a adequação ao uso significa adequação ao tipo de tarefas ou atividades que se pretende realizar com o produto de software, ao tipo de usuários que tipicamente utiliza o produto e ao ambiente de utilização do produto.

Segundo Zambalde e ALVES (2004) o cuidadoso estudo da comunidade de usuários e do conjunto de tarefas, deve considerar cinco fatores imprescindíveis que são os principais objetivos do design de interface com o usuário:

a) produtividade do usuário: a interface deve permitir bom desempenho do usuário na realização de suas tarefas. Não se está falando de desempenho do software,

que é um atributo de qualidade utilizado na engenharia de software, mas do desempenho do usuário em sua interação com um sistema de software;

b) facilidade de aprendizado: deve ser fácil para o usuário aprender a utilizar o software;

c) retenção do aprendizado com uso intermitente: a interface deve permitir que o usuário (esporádico) consiga utilizar o software adequadamente mesmo quando fica sem usá-lo por um período relativamente longo de tempo;

d) prevenção de erros do usuário: o sistema deve prevenir erros do usuário quando o utiliza em suas atividades. Cabe observar aqui também que não se está falando de erros no programa, mas sim de erros do usuário ao utilizar o sistema;

e) Satisfação subjetiva: o usuário deve gostar de utilizar o sistema. Observem que a satisfação é um aspecto subjetivo, pessoal, mas ainda assim importante e que deve ser buscado no desenvolvimento de um produto de software.

A avaliação de interface é um passo muito importante no processo de design de software interativo. Nesse sentido, segundo (ROCHA e BARANAUSKAS, 2000), os testes de usabilidade têm o objetivo de medir quantitativamente o valor alcançado pela interação com o sistema em cada um dos fatores de usabilidade de interesse. Tais testes são classificados em: i) métodos de inspeção; e ii) testes empíricos com usuários. Um dos métodos de inspeção é a avaliação heurística, ou seja, a avaliação de alguns princípios reconhecidos de usabilidade. (NIELSEN, 1993) propõe um conjunto de heurísticas que, dispostas em uma tabela, permite a adequada inspeção de usabilidade de uma interface.

### **3 - METODOLOGIA**

Uma das preocupações básica dos pesquisadores relacionada com as questões metodológicas de suas pesquisas é a reflexão sobre as características específicas dos procedimentos adequados, para a realização da pesquisa proposta. Sendo assim, nesse trabalho, foi elaborada pelos autores uma tabela, adaptada de (NIELSEN 1993) (Tabela 1) para análise de algumas ferramentas CASE utilizadas na elaboração de diagramas da UML. Para cada heurística, devem ser informados os problemas relacionados e a nota do avaliador entre 0 (RUIM) e 10 (ÓTIMO). A interação do usuário com o sistema foi avaliada de acordo com tais heurísticas, permite assim, obter um *score* em porcentagem (10 heurísticas, cada uma com pontuação máxima 10) e, por conseguinte, a avaliação da interface, a qual será considerada satisfatória acima de 80%, razoável entre 79% a 50% e insatisfatória abaixo de 49%. A avaliação foi realizada por pesquisadores da área de Sistemas de Informação no mês de Agosto de 2011.

**Tabela 1 - Formulário de Inspeção de usabilidade**

<b>HEURÍSTICAS</b>	<b>PROBLEMAS DETECTADOS</b>	<b>MÉDIA DO AVALIADOR</b>
H1 - Visibilidade do Estado do Sistema.		
H2 - Correspondência Entre o Sistema e o Mundo Real.		
H3 - Controle e liberdade do usuário.		
H4 - Consistência e Padrões		
H5 - Prevenção de Erros		
H6 - Reconhecimento no lugar de memorização		
H7 - Flexibilidade e Eficiência de uso		
H8 - Desenho estético e minimalista		
H9 - Auxílio no		

reconhecimento, diagnóstico e recuperação de erros		
H10 - Ajuda e documentação		

#### 4- AVALIAÇÃO DAS FERRAMENTAS CASE

Nesta seção, apresenta-se a avaliação da interface de algumas ferramentas CASE utilizadas na elaboração de Diagramas da UML, realizada pelos autores do trabalho, com base nas heurísticas da Tabela 1. A saber:

##### **Ferramenta ArgoUML:**

- Ao escolher o tipo de diagrama, aparece o estado do sistema no alto da tela dando ao usuário uma facilidade maior ao criar os diagramas. Isso é importante pois, cada diagrama usa ferramentas diferentes.
- Há uma boa interação com o usuário já que os botões são bem intuitivos e também há o uso de dicas (tooltips) para auxiliar as funções dos botões. O menu é bem completo e fácil de ser usado.
- O controle e liberdade do usuário é relativamente bom pois pode criar e apagar, porém, deveria ter a função desfazer (undo) e a opção colar e copiar que ficou a desejar.
- A ferramenta é bem consistente possuindo uma organização entre as áreas da tela. Ao fechar o programa ou abrir outra aplicação pronta, o sistema pergunta se deseja salvar o trabalho atual dando maior segurança e evitando perdas.

- Se há alguma ação que não pode ser executada, a ferramenta não executa tal ação. Por exemplo, se tentar inserir um <<extend>> ou <<include>> entre dois atores a ferramenta não aceita tal comando e não executa.
- O menu está em português de fácil utilização. Os botões estão bem ilustrados e organizados de maneira que o usuário, mesmo sendo a primeira vez ao usar a ferramenta, não deverá ter grandes problemas ao criar os diagramas.
- Os diagramas podem ser salvos na pasta a critério do usuário, podendo ser editado a qualquer momento sem nenhuma complicação ou restrição.
- O sistema possui uma boa interface com desenhos bem simples de entender. E como possui o “tooltips”, dicas dos botões, se torna mais fácil o uso da ferramenta.
- Não há mensagens de erro ao criar os diagramas. Por exemplo, posso colocar um <<extend>> e um <<include>> na mesma relação do caso de uso que ele aceita.
- A ajuda é bem completa, porém está no idioma inglês, assim como o site oficial sistema que também está em inglês.

### **Ferramenta Jude:**

- O JUDE não possui um apelo gráfico tão bom quanto o ArgoUML, ou tantas funcionalidades como Microsoft Visio;
- O sistema não interfere nas ações do usuário;
- Existe somente versões em inglês do sistema, porém a sua interface é amigável;
- Existe em seus menus a opção “Help”, um menu de ajuda que encontra-se em inglês;

- A ferramenta Jude Community suporta apenas diagramas da UML 1.4 e UML 2.0;
- Existe pré-requisito para a ferramenta ser executada, ter o JRE instalado em seu computador;
- A interface do sistema além de ser amigável é intuitiva, facilitando o seu uso;

### **Ferramenta Microsoft Visio:**

- Os diagramas da UML são fáceis de serem localizados na ferramenta;
- Em momento algum o sistema informa das ações que o usuário está fazendo se estão certas ou erradas, sendo assim, o usuário tem total liberdade de trabalhar;
- Existe uma versão da ferramenta em português facilitando assim o entendimento de todas as funções;
- A ferramenta apresenta um menu de ajuda e um tutorial para facilitar o entendimento de todas as funções;
- Comando desfazer (undo) limitado - número de movimentos limitado;
- Quando um componente de um diagrama é deslocado às setas de relacionamentos entre eles possuem um acompanhamento automático evitando erros;
- Os estereótipos, que denotam o relacionamento representado pela seta, não podem ser arrastados separadamente - estão “presos” à seta;
- A ferramenta não possui todos os diagramas da UML;
- Em algumas situações a barra de ferramentas não possui todos os elementos necessários para construir do diagrama;

- A interface da ferramenta é muito simples de ser utilizada, sendo assim o usuário não fica perdido na localização de alguma funcionalidade.

## 5. COMPARAÇÃO ENTRE AS FERRAMENTAS

A Tabela 2 apresenta uma comparação entre as ferramentas estudadas em relação aos aspectos mais relevantes segundo heurísticas de usabilidade abordadas neste trabalho.

**Tabela 2. Resultado da avaliação quantitativa**

HEURÍSTICAS	MÉDIA DOS AVALIADORES		
	ARGOUML	JUDE	VISIO
<b>H1</b>	8	7	8
<b>H2</b>	9	8	9
<b>H3</b>	7	7	6
<b>H4</b>	9	8	8
<b>H5</b>	7	6	7
<b>H6</b>	8	8	8
<b>H7</b>	9	9	10
<b>H8</b>	10	8	8
<b>H9</b>	9	7	9
<b>H10</b>	8	10	10
<b>TOTAL</b>	<b>84</b>	<b>78</b>	<b>83</b>

## 6. CONCLUSÃO

Este trabalho teve por objetivo fazer uma identificação e comparação de ferramentas CASE com suporte à elaboração de diagramas da UML, visando avaliar a usabilidade das ferramentas.

Os resultados obtidos permitem identificar as ferramentas Microsoft Visio, com 83 pontos e ArgoUML com 84 pontos, como alcançadora dos resultados satisfatórios, uma vez que atende aos critérios propostos por (NIELSEN, 1993) e às pontuações propostas pelos organizadores.

Entretanto, a ferramenta Jude, com 78 pontos, foi classificada como razoável, conforme problemas apresentados na seção de avaliação.

Na linha de raciocínio, a preocupação com a interface deve ser uma constante tanto durante o desenvolvimento de produtos de software – para os desenvolvedores – quanto durante a escolha de uma ferramenta CASE que possa auxiliar as atividades de desenvolvimento.

O presente trabalho valeu-se de um estudo exploratório sobre usabilidade em ferramentas CASE voltadas para a elaboração de diagramas da UML.

## **7. REFERÊNCIAS**

BOOCH, Grady; RUMBAUCH, James; JACOBSON, Ivar. UML: guia do usuário. Rio de Janeiro: Elsevier, 2005. 2 Edição.

MELO, Ana Cristina. Desenvolvendo aplicações com UML 2.0: do conceitual à Implementação. 2. ed. Rio de Janeiro: Brasport, 2004.

Nielsen, J. (1993) "Usability Engineering". Academic Press, Cambridge, MA.

MONTEIRO, Emilio Soares. Projeto de sistemas e bancos de dados. Rio de Janeiro: Brasport, 2004.

PRESSMAN, Roger S. **Engenharia de software**. 3. ed. São Paulo: Makron Books do Brasil, 2002.

ISO 9241-11 Ergonomic requirements for office work with visual display terminals (VDTs) -- Part 11: Guidance on usability 1998.

ISO/IEC 9126 Information technology – software product quality- part 1: quality model 1999, (FDIS).

Zambalde, A. L. e Alves, R. M. (2004) "Interface Homem-Máquina e Ergonomia", Textos Acadêmicos para o Curso de Pós- Graduação 'Lato Sensu' (Especialização) à Distância – Administração de Sistemas de Informação. UFLA/FAEPE.

Rocha, H. V. e Baranauskas, M. C. C. (2000) "Design e Avaliação de Interfaces Humano-Computador", IME-USP, São Paulo, SP.

Microsoft Office (2011). Disponível em < <http://office.microsoft.com/pt-br/visio-help/CH001026669.aspx> > Acessado em:18 de Agosto de 2011.

VIEIRA, Marcio Junior. Modelagem UML com Software Livre. Disponível em <<http://www.ambientelivre.com.br/~marcio/UML5FISL.pdf>>

CAMPOS, Augusto. A IDE para modelagem de dados. Disponível em < <http://br-linux.org/linux/node/3335> > Acessado em: 18 de Agosto de 2011.

Jude Desing e Communication. Disponível em < [http://jude.change-vision.com/jude-web/product/jude\\_pl.html](http://jude.change-vision.com/jude-web/product/jude_pl.html) > acessado em: 18 de Agosto de 2011.

Revista Java Magazine. Volume 43. Número 05. Ano 2006.