**Avaliação de Interfaces de Sistemas Computacionais com Destaque para Avaliação Heurística**

**Fabio Dela Bruna¹, Márcio Ozório Teixeira¹**

¹Departamento de Ciência da Computação

UNISUL – Universidade do Sul de Santa Catarina

Caixa Postal: 370 CEP: 88704-900 Tubarão – SC - Brasil

{fabio.bruna, marcio.teixeira}@unisul.br

***Abstract.*** *The development of computer systems has grown over the years due to the need to improve daily processes that run in several areas. People feel the need to use either a cellphone, a laptop or a desktop. However, with the need to use a computer, there is the need to maintain a means of communication between the user and the system, this responsibility, the computer systems used in the computer. However, the quality of interface of these systems is essential for the user, making the process much less complicated to use or not. Therefore, there is a need to evaluate these systems, checking and adjusting the existing problems in this interaction. Therefore, this article will be presented an evaluation method of interface, called the heuristic evaluation.*

***Resumo.*** *O desenvolvimento de sistemas computacionais vem crescendo ao longo dos anos, devido à necessidade de aperfeiçoar os processos diários que são executados nas mais diversas áreas. As pessoas sentem a necessidade de utilizar seja um celular, um notebook, ou um desktop. Porém, junto à necessidade de utilizar um computador, há a necessidade de manter um meio de comunicação entre o usuário e o sistema, responsabilidade essa, dos sistemas computacionais utilizados no computador. Entretanto, a qualidade de interface destes sistemas é essencial para o usuário, tornando o processo de utilização muito menos complicado ou não. Por isso, há a necessidade de avaliarmos esses sistemas, verificando e ajustando os problemas existentes nessa interação. Sendo assim, neste artigo será apresentado um método de avaliação de interface, chamado de avaliação heurística.*

1. **Introdução**

Durante todo o processo de desenvolvimento de sistemas computacionais existe uma área de estudo conhecida como IHC (Interação Humano-Computador) que visa, através de estudos, desenvolver interfaces de alta qualidade, com o objetivo de promover o desenvolvimento de aplicações com um alto nível de usabilidade, visando apresentar uma interface agradável e eficaz, tornando esse processo de interação entre o usuário e a interface, algo que seja bom de realizar. Para isso, são definidos métodos, modelos e diretrizes. A partir daí, que entra a etapa de avaliação de interfaces durante o processo desenvolvimento de um sistema computacional, utilizando-se de métodos para avaliar quão bom é a interação de uma aplicação. Neste artigo, aborda-se um tipo de avaliação de interface, chamado de avaliação heurística, da qual abordaremos com maiores detalhes no decorrer deste documento.

1. **Interface**

Segundo Lima (2005) o termo interface é aplicado normalmente àquilo que interliga dois sistemas. Considerando-se que uma interface homem-máquina é a parte de um componente que permite a um usuário controlar e avaliar o funcionamento deste artefato por meio de dispositivos sensíveis às suas ações e capazes de estimular sua percepção. No processo de interação usuário-sistema a interface é o combinado de software e hardware necessário para viabilizar e facilitar os processos de comunicação entre o usuário e a aplicação. Ou seja, a interface é a “cara” do sistema com a qual o usuário está lidando quando o utiliza.

1. **Interação**

De acordo com Lima (2005) interação é um processo que engloba as ações do usuário sobre a interface de um sistema e suas interpretações sobre as respostas reveladas por esta interface. Ou seja, a interação é a resposta a uma ação executada pelo usuário, pois, a interface estará se comunicando com o usuário, seja, através de mensagens, áudios e etc, respondendo a requisição que ele efetuou para a interface.

1. **Usabilidade**

Lima (2005) também define que o termo usabilidade é utilizado para medir a qualidade das interfaces. Segundo Nielsen (1993), usabilidade está relacionada com facilidade de aprendizado, eficiência, facilidade de memorização, quantidade de erros e satisfação do usuário.

A usabilidade de um sistema é um conceito que se refere à qualidade da interação de sistemas com os usuários e depende de vários aspectos. Alguns destes fatores são citados, resumidamente, por Lima (2005):

**a.** Facilidade de aprendizado do sistema: tempo e esforço necessários para que os usuários atinjam um determinado nível de desempenho.

**b.** Facilidade de uso: avalia o esforço físico e cognitivo do usuário durante o processo de interação, medindo a velocidade e o número de erros cometidos durante a execução de uma determinada tarefa.

**c.** Satisfação do usuário: avalia se o usuário gosta e sente prazer em trabalhar com este sistema.

**d.** Flexibilidade: avalia a possibilidade de o usuário acrescentar e modificar as funções e o ambiente iniciais do sistema. Assim, este fator mede também a capacidade do usuário utilizar o sistema de maneira inteligente e criativa, realizando novas tarefas que não estavam previstas pelos desenvolvedores.

**e.** Produtividade: se o uso do sistema permite ao usuário ser mais produtivo do que seria se não o utilizasse.

Todavia, para os fatores citados acima existem alguns métodos padronizados com estudos específicos baseados na forma de como se deve avaliar a usabilidade de uma interface, deste modo, logo abaixo, entraremos em maiores detalhes em uma destas formas, a avaliação heurística.

1. **Avaliação de Interfaces**

De acordo com Prates (2010) antes de declarar um software pronto para uso, é importante saber se ele está em conformidade com os usuários que o utilizaram nas suas tarefas diárias e também, no ambiente em que será utilizado. Assim como testes de funcionalidade são necessários para se verificar a robustez da implementação, a avaliação de interface é necessária para se analisar a qualidade de uso de um software.

O quanto antes for encontrado os problemas de interface, menor o custo de se consertá-los. Além do mais, um projetista não deve supor que basta seguir métodos e princípios de projeto de interfaces para garantir uma alta qualidade de uso de seu software, pois, não deve-se crer que os usuários são como ele próprio, e que, portanto bastaria sua avaliação individual para atestar esta qualidade. Precisa-se ter em mente que alguém vai avaliar a qualidade de uso do seu sistema, nem que seja apenas o usuário final. Desta forma, Prates também destaca alguns dos principais objetivos de se realizar uma avaliação de sistemas interativos, como:

* Identificar as necessidades de usuários ou verificar o entendimento dos projetistas sobre estas necessidades;
* Identificar problemas de interação ou de interface;
* Investigar como uma interface afeta a forma de trabalhar dos usuários;
* Comparar alternativas de projeto de interface;
* Alcançar objetivos quantificáveis em métricas de usabilidade;
* Verificar conformidade com um padrão ou conjunto de heurísticas;

Atualmente, ainda é freqüente encontrarmos gerentes de projeto que pensam apenas nos custos envolvidos na realização de avaliações de seus sistemas. Isto se deve geralmente pelo desconhecimento dos benefícios associados a estas avaliações. Sendo assim, destacamos abaixo uma forma de avaliação de interface, que provê através de algumas heurísticas a melhor forma de desenvolver um software com um bom nível de usabilidade.

* 1. **Avaliação heurística**

De acordo com Winckler e Pimenta (2010) a avaliação heurística é um método tradicional de avaliação de usabilidade. O método foi utilizado pela primeira vez em uma interface Web em 1994, num estudo para o Web site da Sun Microsystems. Este método foi desenvolvido por Nielsen e Molich e consiste da inspeção sistemática da interface do usuário com relação à sua usabilidade. Basicamente um avaliador interage com a interface e julga a sua adequação comparando-a com princípios de usabilidade reconhecidos, as heurísticas.

Este método não envolve usuários, e deve ser efetuado por avaliadores especialistas. Geralmente, recomenda-se que três a cinco especialistas efetuem uma avaliação heurística. Além do mais, este método é bastante rápido, e de menor custo que a maior parte dos métodos de avaliação amplamente difundidos. Como em todo método de avaliação, a avaliação heurística envolve uma fase de preparação, definida por Nielsen(2007) como:

* proposta de design (papel ou protótipo);
* hipóteses sobre os usuários (opcional);
* cenário de tarefas (opcional);

Prates (2010) define que a avaliação deve ser realizada de acordo com o seguinte procedimento:

**1.** sessões curtas (1 a 2 horas) de avaliação individual, onde cada especialista...

• Julga a conformidade da interface com um determinado conjunto de princípios (“heurísticas”) de usabilidade;

• Anota os problemas encontrados e sua localização;

• Julga a gravidade destes problemas;

• Gera um relatório individual com o resultado de sua avaliação e comentários adicionais;

Sendo que, é importante que estas sessões sejam individuais, para que um avaliador não seja influenciado pela opinião de outros. Durante cada sessão de avaliação, o avaliador percorre a interface diversas vezes, inspecionando os diversos elementos de interface e comparando-os com a lista de heurísticas de usabilidade.

**2.** consolidação da avaliação dos especialistas

• Novo julgamento sobre o conjunto global dos problemas encontrados;

• Relatório unificado de problemas de usabilidade;

Nesta etapa, cada avaliador tem acesso aos relatórios individuais de todos os avaliadores, e pode expressar seu julgamento sobre os problemas apontados pelos outros avaliadores. Ao final desta etapa, deve-se gerar um relatório unificado e consolidado sobre todos os problemas encontrados.

**3.** seleção dos problemas que devem ser corrigidos

Esta etapa deve ser realizada junto ao cliente ou ao gerente de projeto. Trata-se de uma análise de custo/benefício das correções aos problemas encontrados na etapa anterior. Esta análise deve levar em conta não apenas a gravidade dos problemas, mas também os prazos e o orçamento do projeto, bem como a capacitação da equipe de desenvolvimento.

Lembra-se que cada elemento de interface (ou conjunto de elementos) deve ser analisado para verificar sua conformidade com cada uma das seguintes heurísticas propostas por Nielsen (2007):

1. **Diálogos Simples e Naturais:** as interfaces de usuários devem ser o mais simples possível. Interfaces devem combinar as tarefas do usuário de tal forma que o mapeamento entre os conceitos computacionais e os do usuário seja simples. Deve-se apresentar exatamente a informação que o usuário precisa - nem mais nem menos - na hora e lugar exatos onde é necessária.
2. **Falar a Linguagem do Usuário:** a terminologia da interface deve ser baseada na linguagem do usuário, e não orientada ao sistema. Para tanto, deve-se verificar quais termos são utilizados com maior freqüência pêlos usuários. As informações também devem ser organizadas conforme o modelo mental que o usuário possui do domínio.
3. **Minimizar a Sobrecarga de Memória do Usuário:** o software deve exibir elementos de diálogo para o usuário e permitir que o mesmo faça suas escolhas, sem a necessidade de lembrar-se deste ou daquele comando específico. Para facilitar a utilização da interface, deve ser apresentado ao usuário um pequeno número de regras que se aplicam por toda a interface. Se o número de regras é grande o usuário terá de aprender/lembrar todas as regras, o que pode não ser tão simples. Por outro lado, se o software não tiver regra alguma, então o usuário deverá lembrar-se de cada elemento de diálogo.
4. **Consistência:** consistência é um dos princípios básicos de usabilidade. Se os usuários souberem que um mesmo comando ou uma mesma ação terá sempre o mesmo efeito, eles ficarão mais confiantes no uso do software/ e serão encorajados a fazerem novas descobertas. A mesma operação deverá ser apresentada na mesma localização em todas as telas e deverá ser formatada da mesma maneira para facilitar o reconhecimento.
5. **Feedback:** O sistema deverá informar continuamente ao usuário sobre o que ele está fazendo. O tempo de resposta influi no tipo de feedback que deve ser dado ao usuário. Um décimo de segundo (0,1s) é o limite para o usuário pensar que o sistema está reagindo instantaneamente, o que significa que nenhum feedback especial é necessário; um segundo (1.0s) é o limite para que o fluxo de pensamento do usuário não seja interrompido, mesmo que o usuário perceba uma certa demora; e dez segundos (10s) é o limite para manter a atenção do usuário focalizada no diálogo. Às vezes feedbacks especiais são necessários para contextualizar uma navegação mais demorada do usuário.
6. **Saídas Claramente Marcadas:** De modo a fazer com que o usuário sinta que pode controlar o software, deverá ser fácil sair das situações mais variadas possíveis. Por exemplo, todas as caixas de diálogo devem possuir um botão de cancelar para trazer o usuário para a situação anterior. Muitas vezes, as saídas podem ser fornecidas por meio de uma facilidade de desfazer (*undo*) a última operação para retornar ao estado anterior. Os usuários rapidamente aprendem a confiar neste mecanismo, logo, deve estar disponível como um comando genérico por todo o software. Neste caso, o usuário poderá confiar no aprendizado por exploração, pois saberá desfazer eventuais erros.
7. **Atalhos:** embora deva ser possível operar a interface conhecendo-se apenas algumas regras gerais, deveria também ser possível para o usuário experiente executar mais rapidamente operações freqüentemente utilizadas, através de atalhos.
8. **Boas mensagens de erro:** as mensagens de erro devem seguir algumas regras: linguagem clara e sem códigos. Devem ser precisas. Devem ajudar o usuário a resolver o problema. Não devem intimidar ou culpar o usuário.
9. **Prevenir Erros:** melhor do que possuir boas mensagens é evitar situações de erro. Conhecendo-se as situações que mais provocam erro, sempre é possível modificar a interface a tornar muito difícil que este erro ocorra.
10. **Ajuda e Documentação:** o melhor é se ter um software que seja tão fácil de usar que não necessite de ajuda ou documentação para complementar a interface do usuário. Além disto, sabidamente usuários raramente lêem a documentação.

Para cada problema encontrado, ou seja, para cada heurística violada, deve-se definir ainda a localização do problema, ou seja, onde ele ocorre na interface, e sua gravidade.

Com relação à **localização**, o problema pode estar:

* em um único local na interface;
* em dois ou mais locais na interface, casualmente;
* na estrutura geral da interface, de forma sistemática; ou
* pode ser algo que não existe, ou seja, precisa ser incluído na interface;

Já a **gravidade** do problema é calculada, por cada especialista, como uma

combinação dos fatores:

* Freqüência com que o problema ocorre;
* Persistência do problema;

A gravidade do problema é definida por um valor da seguinte escala:

* 0 – Não concordo que isto seja um problema (este valor pode resultar da avaliação de um especialista sobre um problema apontado por outro especialista);
* 1 – Problema cosmético: não precisa ser consertado a menos que haja tempo extra no projeto;
* 2 – Problema pequeno: o conserto deste problema é desejável, mas deve receber baixa prioridade;
* 3 – Problema grande: importante de ser consertado, deve receber alta prioridade;
* 4 – Catastrófico: é imperativo consertar este problema antes do lançamento do produto;

Como produto da avaliação heurística, os especialistas redigem um relatório consolidado. Este relatório pode conter, por exemplo, os seguintes itens:

* problemas esperados (e possíveis consertos);
* o quão bem o sistema apóia as tarefas dos usuários;
* caminhos de interação primários (importantes e/ou freqüentes);
* caminhos de interação alternativos ou pouco utilizados;
* recomendações de projeto;

É possível realizar uma avaliação heurística nas etapas iniciais do ciclo de projeto e

desenvolvimento. Esta avaliação pode ser feita sobre interfaces que ainda não tenham

sido implementadas, representadas em papel.

**6. Conclusão**

Diante de todas as informações citadas neste artigo pode-se concluir que realmente há a necessidade de se utilizar a avaliação de interfaces durante o desenvolvimento de um software, entretanto, percebe-se que pelo fato de ser um processo um pouco caro, muitas empresas acabam não contratando, muito menos, mantendo pessoas com experiências na área de IHC dentro da organização, especificamente na área de avaliação de design de interfaces, para apresentar uma avaliação detalhada da interface desenvolvida.

Porém, vale a pena estar contratando estes especialistas para efetuarem a avaliação heurística, pois, identifica-se que ela é válida para casos em que o avaliador tenha experiência na área da qual o software é proposto, pois, como o usuário não participa do processo, algumas questões acabam passando despercebidas, e sendo identificadas somente quando o usuário for utilizá-lo. Outra questão, é quanto as heurísticas propostas por Nielsen, talvez, em alguns casos pode ocorrer que somente essas dez apresentadas aqui não sejam suficientes para avaliar de maneira adequada a interface, por isso, ressalta-se que se pode adequar ao contexto da avaliação e incrementar com outros fatores relevantes.

Quanto aos avaliadores, destaca-se que é altamente recomendado, que estes não participem do processo de desenvolvimento das interfaces, pois, supõe-se que ninguém gosta de encontrar erros nos próprios trabalhos e também, como ele estará habituado com as interfaces, possivelmente, muitos erros poderão passar despercebidos.

Contudo, a avaliação heurística apresenta resultados rápidos, talvez não tão precisos quanto outros métodos existentes, também acaba sendo um método barato, pois, será efetuado poucas vezes, durante a finalização do processo de desenvolvimento, e é altamente recomendável, sabe-se que não irá solucionar todos os problemas, pois, somente o usuário final irá lhe mostrar isto, mas já solucionará boa parte, fazendo com que não haja necessidade de que o usuário encontre erros considerados simples que poderiam ter sido evitados desde o inicio do processo.

1. **Referências**

Lima , Paulo Sergio Rodrigues. Tópicos de Interfaces Humano-Computador:conceitos, usabilidade e Avaliação de Interfaces. 2007. Disponível em: < http://www2.uepa.br/paulolima/disciplinas/TopicosIHC.pdf >. Acesso em: 20 maio 2010.

Nielsen, Jakob; Loranger, Hoa. Usabilidade na Web.Rio de Janeiro: Elsevier, 2007.

Prates, Raquel Oliveira; BARBOSA, Simone Diniz Junqueira. Avaliação de Interfaces de Usuário: Conceitos e Métodos. Disponível em: < http://www.scribd.com/doc/6893/Avaliacao-de-Interfaces-de-Usuario-Conceitos-e-Metodos >. Acesso em: 20 maio 2010.

Winckler, Marco; Pimenta, Marcelo Soares. Avaliação de Usabilidade de Sites Web. Disponível em: < http://ihcs.irit.fr/winckler/2002-winckler-pimenta-ERI-2002-cap3.pdf >. Acesso em: 21 maio 2010.