

## Capítulo

# 6

## Avaliação de Interfaces de Usuário – Conceitos e Métodos

Raquel Oliveira Prates, Simone Diniz Junqueira Barbosa

### *Abstract*

*As personal computers become more and more popular, they are being used not only to perform work, but also as means to communicate, provide social insertion and household planning. The quality of the interface is mister, if interactive systems are to be successful. To design high quality interfaces, it is essential that they are evaluated during the design process, in order to indentify and adjust interaction problems. This chapter presents concepts of quality of use in interfaces and the main evaluation methods used to assess them.*

### *Resumo*

*Atualmente, com a popularização de computadores pessoais, cada vez mais computadores são incorporados ao cotidiano das pessoas, sendo utilizados não apenas como ferramenta de trabalho, mas também como meio de comunicação, de inserção social e de planejamento familiar. A qualidade da interface é fundamental para que sistemas interativos possam ser utilizados com sucesso. Para se obter interfaces de alta qualidade é essencial que estas sejam avaliadas durante o processo de design, permitindo assim a identificação e ajustes de problemas de interação. Neste capítulo serão apresentados conceitos de qualidade de uso de interfaces e os principais métodos de avaliação utilizados para sua apreciação*

### 6.1. Introdução

Em linhas gerais, a área de Interação Humano-Computador (IHC) investiga o “projeto (design), avaliação e implementação de sistemas computacionais interativos para uso humano, juntamente com os fenômenos associados a este uso” (Hewett et al., online). Os estudos relacionados ao **projeto** de IHC se referem a como construir interfaces com

alta qualidade. Para isto, são definidos métodos, modelos e diretrizes. Os estudos relacionados à **avaliação** de IHC, por sua vez, buscam avaliar a qualidade de um projeto de interface, tanto ao longo do processo de desenvolvimento como quando o software está pronto.

### 6.1.1 Interface x Interação

**Interação** é o processo de **comunicação** entre pessoas e sistemas interativos (). Neste processo, usuário e sistema trocam turnos em que um “fala” e o outro “ouve”, interpreta e realiza uma ação. Esta ação pode ser tão simples quanto dar uma resposta imediata à fala do outro, ou consistir de operações complexas que alteram o estado do mundo. A área de IHC estuda este processo, principalmente do ponto de vista do usuário: as **ações** que ele realiza usando a interface de um sistema, e suas **interpretações** das respostas transmitidas pelo sistema através da interface (Figura 1).

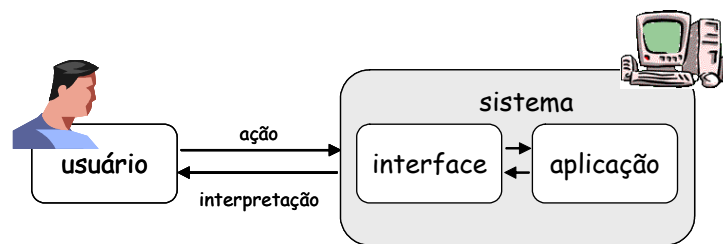


Figura 1: O processo de interação humano-computador.

**Interface** é o nome dado a toda a porção de um sistema com a qual um usuário mantém contato ao utilizá-lo, tanto ativa quanto passivamente. A interface engloba tanto software quanto hardware (dispositivos de entrada e saída, tais como: teclados, mouse, tablets, monitores, impressoras e etc.). Considerando a interação como um processo de comunicação, a interface pode ser vista como o **sistema de comunicação** utilizado neste processo. Uma definição de interface utilizada com frequência foi proposta por Moran:

*“a interface de usuário deve ser entendida como sendo a parte de um sistema computacional com a qual uma pessoa entra em contato — física, perceptiva ou conceitualmente” (Moran, 1981).*

A dimensão **física** inclui os elementos de interface que o usuário pode manipular, enquanto a dimensão **perceptiva** engloba aqueles que o usuário pode perceber. A dimensão **conceitual** resulta de processos de interpretação e raciocínio do usuário desencadeados pela sua interação com o sistema, com base em suas características físicas e cognitivas, seus objetivos e seu ambiente de trabalho.

Atualmente, as interfaces mais comuns envolvem elementos visuais e sonoros, com entrada de dados via teclado e mouse. Outros tipos de interfaces, como interface via voz e entrada de dados através de canetas estão se tornando freqüentes, devido à disseminação de dispositivos móveis.

### 6.1.2 Objetivos e Importância da Avaliação de Interação

Antes de declarar um software pronto para uso, é importante saber se ele apóia adequadamente os usuários, nas suas tarefas e 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. Quanto mais cedo forem encontrados os problemas de interação ou de interface, menor o custo de se consertá-los ().

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. Além disto, também não deve presumir que os usuários são como ele próprio, e que portanto bastaria sua avaliação individual para atestar esta qualidade (). Deve-se ter em mente que *alguém* vai avaliar a qualidade de uso do seu sistema, nem que seja apenas o usuário final...

Alguns dos principais objetivos de se realizar avaliação de sistemas interativos são (; );

- 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

Infelizmente, 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. Dependendo do momento em que for realizada a avaliação, estes benefícios podem ter efeito imediato, em consertos importantes no início do desenvolvimento; a médio prazo, no planejamento da estratégia de treinamento e marketing; ou até mesmo a longo prazo, apoiando o planejamento de versões futuras do software.

### 6.1.3 Conceitos de qualidades de uso

O conceito geral de **qualidade de uso** está estreitamente relacionado com a capacidade e a facilidade de os usuários atingirem suas metas com eficiência e satisfação. Quando os usuários têm vias alternativas para realizarem suas tarefas, com ou sem apoio computacional, o fato de escolherem espontaneamente utilizar um determinado sistema, e com certa freqüência, dependerá em grande parte da qualidade de uso daquele sistema.

O conceito de qualidade de uso mais amplamente utilizado é o de usabilidade, relacionado à facilidade e eficiência de aprendizado e de uso, bem como satisfação do usuário (). Mais recentemente, foi elaborado o conceito de comunicabilidade, que busca avaliar o processo implícito de comunicação designer-usuário, que se dá através da interface (Prates et al. 2000b,). Já o conceito de aplicabilidade está relacionado à

flexibilidade de um sistema, em particular com relação à sua utilidade em uma variedade de situações ().

## Usabilidade

O conceito de **usabilidade** permite avaliar a qualidade de um sistema com relação a fatores que os projetistas definem como sendo prioritários ao sistema. Alguns fatores típicos envolvidos no conceito de usabilidade são (; Preece et al., 2002):

- facilidade de aprendizado
- facilidade de uso
- eficiência de uso e produtividade
- satisfação do usuário
- flexibilidade
- utilidade
- segurança no uso

**Facilidade de aprendizado** se refere ao tempo e esforço necessários para que os usuários aprendam a utilizar uma determinada porção do sistema com determinado nível de competência e desempenho. Geralmente, um sistema pode ser analisado sob uma perspectiva de uso simples, considerando um nível intermediário ou avançado, por exemplo, cada qual requerendo tipos e graus de aprendizado distintos. Neste caso, o fator de facilidade de aprendizado pode ser analisado em diversos pontos, considerando cada passagem de um nível de capacitação ao próximo.

O fator **facilidade de uso** do sistema está relacionado não apenas com o esforço cognitivo para interagir com o sistema, mas também com o número de erros cometidos durante esta interação. É importante observar que um sistema fácil de aprender não é necessariamente fácil de utilizar ou vice-versa.

Sistemas fáceis de utilizar podem ser ineficientes de duas formas: com relação a **o que** permite o usuário fazer (eficiência de uso), e a **como** o usuário deve fazê-lo (produtividade). O fator **eficiência de uso** serve para analisar se o sistema faz bem aquilo a que se destina. Já o fator **produtividade** serve para avaliar se o usuário consegue fazer o que precisa de forma rápida e eficaz. Este fator é geralmente avaliado pelo tempo decorrido desde o início até a conclusão de uma tarefa e pelo número de passos que o usuário precisou realizar.

Como a aceitação de um sistema interativo é determinante do sucesso do sistema, o fator **satisfação do usuário** enfatiza a avaliação subjetiva do sistema feita por seus usuários, incluindo emoções que possam surgir durante a interação, sejam elas positivas, como prazer e diversão, ou negativas, como tédio ou frustração.

Pessoas diferentes podem seguir caminhos distintos para atingir um mesmo objetivo. Estas idiossincrasias vão desde operações primitivas como o uso de mouse ou teclas de atalho para acionar uma função do sistema, até mesmo estratégias de solução de problemas completamente distintas, como o uso “criativo” de um editor de textos

como software de apresentação de slides, por exemplo. O fator **flexibilidade** considera o quanto um sistema é capaz de acomodar estas idiosincrasias.

O fator **utilidade** de um sistema se refere ao quanto um sistema oferece o conjunto de funcionalidades necessárias para os usuários realizarem suas tarefas. Esta dimensão está intimamente relacionada ao conceito de aplicabilidade proposto por Fischer (1998), que será visto adiante.

A dimensão de **segurança no uso** se refere ao grau de proteção de um sistema contra condições desfavoráveis ou até mesmo perigosas para os usuários. Trata-se principalmente de como evitar e permitir que o usuário se recupere de condições de erro com consequências sérias para seu trabalho ou para sua saúde.

### **Comunicabilidade**

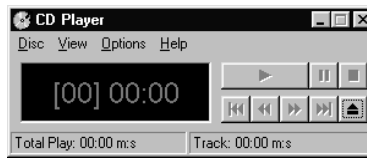
O conceito de **comunicabilidade** (de Souza et al. 1999, Prates et al., 2000b) se refere à capacidade de os usuários entenderem o design tal como concebido pelos projetistas. A hipótese subjacente ao conceito de comunicabilidade é que, se um usuário entende as decisões que o projetista tomou ao construir a interface, aumentam suas chances de fazer um bom uso daquele sistema. Em sistemas com alta comunicabilidade, os usuários são capazes de responder:

- para que o sistema serve
- qual é a vantagem de utilizá-lo
- como funciona
- quais são os princípios gerais de interação com o sistema

Durante o processo de design, o projetista elabora as respostas para estas perguntas, mas nem sempre se preocupa em transmiti-las adequadamente através da interface. Como resultado, o usuário pode ser incapaz de criar um modelo mental do sistema que seja compatível com o do projetista, o que frequentemente torna a interação um tedioso exercício de tentativa e erro.

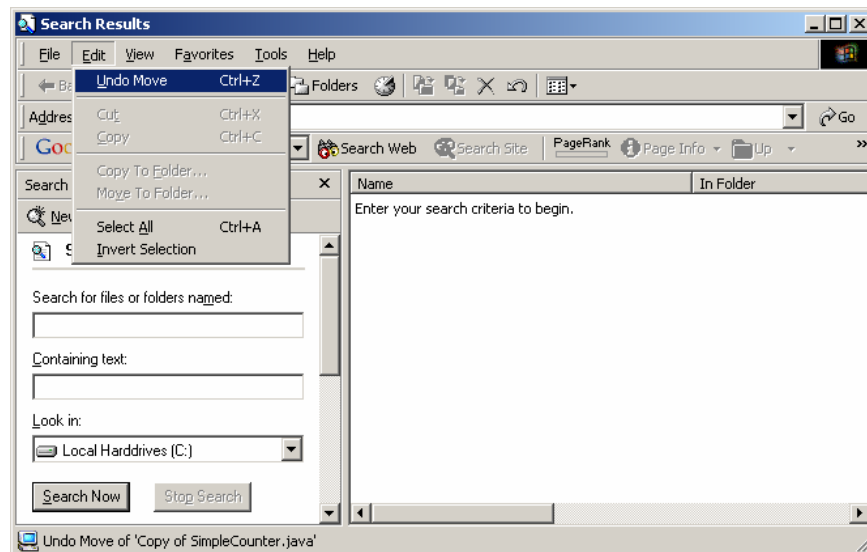
Uma interface com boa comunicabilidade permite que o usuário formule um modelo mental compatível com o do projetista. O uso de analogias com artefatos familiares ao usuário pode contribuir para isto, pois o usuário já possui um modelo mental sobre o comportamento desses artefatos. No entanto, é importante deixar claro qual é o escopo da analogia, ou seja, quais são as porções do modelo mental sobre o artefato conhecido que podem ser transportadas para a construção do modelo mental sobre a interface em questão.

Um exemplo de interface com boa comunicabilidade é a tela inicial do programa CD Player, do Windows 95™ (Figura 2). Ela tira proveito da familiaridade do usuário com os aparelhos comuns de CD, fornecendo elementos de interface análogos, tais como os botões de comando para acionamento das funções e o visor de trilhas e duração como elemento de *feedback*.



**Figura 2: Exemplo de boa comunicabilidade: interface de um software para tocar CD's.**

Um exemplo de baixa comunicabilidade pode ser encontrado na ferramenta de busca de arquivos no Windows™ 2000 (Figura 3). O usuário aciona a ferramenta de busca, mas a janela aparece reduzida e deslocada, de modo que as opções de busca não estão visíveis. O usuário move a janela para o centro da tela, mas ainda assim as opções não aparecem. Fazendo uso do conhecimento adquirido de que o menu dá acesso a todas as funções de um sistema, ele resolve procurar estas opções sob o menu *Edit*. Este menu não apresenta as opções de busca, como esperado, mas surpreendentemente, possui um item chamado *Undo Move*. Tentando entender o que significa este comando, o usuário imagina que sirva para restaurar a posição da janela ao local anterior ao deslocamento, e resolve experimentar, mas “nada acontece”. (Na verdade, o comando *Undo Move* desfez a última transferência de arquivo que o usuário fez antes de acionar a ferramenta de busca. Existe uma mensagem na barra de status indicando o que consiste o comando *Undo Move*, mas esta mensagem não atenua o fato de que transferência de arquivos não é uma tarefa que deva estar contida em uma ferramenta de busca de arquivos.)



**Figura 3: Exemplo de baixa comunicabilidade. Observe o item de menu *Undo Move* dentro da ferramenta de busca.**

## Aplicabilidade

A **aplicabilidade** de um sistema também determina sua qualidade de uso. Este conceito está relacionado com a utilidade deste sistema em uma variedade de situações e problemas (Fischer, 1998). Este conceito permite determinar:

- o quanto o sistema é útil para o contexto em que foi projetado
- em que outros contextos o sistema pode ser útil

Fischer acredita que as pessoas, por serem hábeis e especialistas no seu domínio, querem agir como designers, no sentido de participar ativamente dos processos de solução de problemas e de construção ou transformação dos seus próprios artefatos e espaços de trabalho. Ele coloca como desafios essenciais de IHC a melhoria das formas como as pessoas podem usar computadores para trabalharem, pensarem, se comunicarem, aprenderem, criticarem, explicarem, argumentarem, discutirem, observarem, decidirem, calcularem, simularem e projetarem.

Interfaces muito rígidas, nas quais os usuários têm apenas um caminho a seguir, com pouca possibilidade de cometer erros, são freqüentemente chamadas de “a prova de idiotas” (do inglês, *idiot-proof*). Na realidade, este tipo de interface trata todos os usuários como pessoas incapazes de tomar decisões apropriadas. Os usuários destes sistemas têm reações negativas de diversos tipos, conforme suas características e o contexto em que estão inseridos: eles fazem um mau uso do sistema, deixam de usá-lo, ou até mesmo se limitam a crer que o sistema tem sempre razão e que eles, usuários, não deveriam mesmo tomar decisões importantes.

Diversos pesquisadores têm chamado atenção para a necessidade de se desenvolver sistemas que **ampliem as capacidades dos usuários**, em vez de tentarem substituí-las, possibilitando que eles ajam de forma mais inteligente e eficiente (). O usuário pode ser considerado um **especialista** na sua tarefa: seu conhecimento, competência e forma de atuação devem ser respeitados.

Por que incluir, em um processo de desenvolvimento de sistemas interativos, procedimentos de avaliação de sistemas com relação à sua qualidade de uso? Do **ponto de vista do usuário**, a qualidade da interface e da interação determina a qualidade do sistema, e não seus algoritmos, arquitetura ou modelos de dados. Para ele, o **sistema é a interface**. O grau de qualidade de uso de um sistema pode causar aumento (ou queda) de produtividade dos usuários, e reduzir (ou aumentar) os custos com suporte técnico para atendimento aos usuários. Além disto, as iniciativas voltadas para a qualidade de uso de sistemas computacionais estão geralmente associadas a melhorias em processos de negócio, que ajudam a promover ainda mais um aumento de qualidade do produto final.

Interfaces com baixa qualidade de uso trazem diversos problemas, dentre os quais:

- requerem treinamento excessivo
- desmotivam a exploração
- confundem os usuários
- induzem os usuários ao erro
- geram insatisfação
- diminuem a produtividade
- não trazem o retorno de investimento previsto

Estes problemas podem ser detectados através de métodos de avaliação diversos, realizados ao longo do processo de desenvolvimento. Como será visto adiante, os métodos de avaliação mais utilizados se concentram em avaliar a usabilidade e a comunicabilidade de um sistema. Como não há um método de avaliação específico para o conceito de aplicabilidade, deve-se optar por um dos métodos qualitativos de avaliação, que provêm insumos preciosos para esta avaliação.

## **6.2. Características dos métodos de avaliação**

Os métodos de avaliação de interface diferem entre si em vários aspectos. É preciso entender as diferentes características de cada método, para se definir qual deles é o mais apropriado para se avaliar a interface de um software em um determinado contexto. As principais diferenças entre os métodos são a etapa do ciclo de design do software em que devem ou podem ser aplicados (durante o ciclo de desenvolvimento ou após ter o produto pronto), a técnica utilizada para coletar os dados (desde entrevistas até experimentos em laboratórios), os tipos de dados coletados (quantitativos ou qualitativos), e ainda o tipo de análise feito (o avaliador pode prever potenciais problemas ou interpretar os dados obtidos) (Preece et al., 1994). Nesta seção apresentaremos cada uma destas dimensões.

### **6.2.1 Etapa do ciclo de design**

A avaliação de uma interface pode ser feita durante diferentes etapas do ciclo de desenvolvimento do software. As avaliações formativas (Preece et al., 1994; Hartson, 1998) são aquelas que são feitas durante o processo de design, antes de o sistema estar terminado, e muitas vezes antes de uma linha de código sequer ser escrita. A avaliação pode ser feita utilizando-se desde cenários, *storyboards*, ou a modelagem conceitual da interação até protótipos do sistema. A vantagem de se fazer avaliação formativa é que problemas de interação são identificados e consertados antes de a aplicação ser terminada e liberada para uso. Quanto mais cedo no ciclo de design um problema é descoberto e reparado, menor o custo das alterações necessárias no software (Karat, 1993). Além disto, o software resultante a ser oferecido para o usuário é de melhor qualidade.

As avaliações de interface feitas em produtos já terminados são chamadas de avaliações somativas. Normalmente, enquanto as avaliações formativas têm por objetivo melhorar a qualidade do sistema, tornando-o mais usável para o usuário, as avaliações somativas buscam verificar a existência de determinados aspectos no sistema desenvolvido, como por exemplo a sua conformidade com um padrão estabelecido.

### **6.2.2 Técnicas de Coleta de Dados**

Existem várias técnicas disponíveis para se coletar dados sobre a interface de um software e se fazer a análise da sua qualidade de uso. A decisão sobre que técnica utilizar depende principalmente da disponibilidade dos recursos que se tem e objetivos da avaliação a ser feita. A seguir apresentamos as principais características de cada uma das técnicas.



### **Coleta da opinião de usuários**

A coleta da opinião de usuários tem por objetivo se obter uma apreciação dos usuários em relação ao sistema interativo. Normalmente, se deseja identificar o nível de satisfação dos usuários com o sistema, o que inclui aspectos como: se eles gostam do sistema, se a aparência estética do sistema é satisfatória, se o sistema faz aquilo que eles desejam, se tiveram algum problema ao usá-lo, e/ou se eles gostariam de (ou pretendem) usá-lo novamente. As principais técnicas utilizadas para se coletar a opinião de usuários são questionários e entrevistas. Os tipos de questionários e entrevistas a serem utilizados variam em diversos aspectos (Nicolaci-da-Costa, 2001; Preece et al., 2002). Eles podem ser feitos pessoalmente ou por telefone, email ou web, com um pequeno grupo de pessoas ou com centenas de pessoas, com cada usuário individualmente ou com grupos de usuários, e utilizando perguntas bem estruturadas ou livres.

### **Observação de usuários**

Nem sempre usuários percebem ou conseguem expressar a sua experiência de uso com o sistema. A observação do uso do sistema pelo usuário permite ao avaliador ter uma visão dos problemas sendo vivenciados pelos usuários e muitas vezes dos aspectos positivos experimentados durante o uso. A observação pode ser registrada usando-se anotações do observador, gravação de vídeo, áudio ou da interação, ou uma combinação destas.

O usuário pode ser observado no seu contexto de uso, ou seja utilizando o software no seu trabalho, ou até mesmo em sua casa. Nestas situações se consegue observar o usuário utilizando o software para realizar as tarefas que ele considerar relevantes e para as quais acredita que o software seja apropriado, e ainda da maneira que considera adequada ou desejável. Em tais situações, alguns dos desafios para os avaliadores são conseguir observar sem interferir no contexto ou inibir o usuário, e como fazer a análise dos dados coletados, principalmente quando se obtém várias horas de vídeo gravadas, ou quando diferentes tipos de registro feitos durante o uso devem ser integrados.

O avaliador pode desejar observar o uso feito pelo usuário em ambientes mais controlados, como laboratórios. Nestes ambientes o avaliador tem um controle maior sobre as variáveis que influenciam a avaliação, como o tempo de duração, a concentração do usuário e as tarefas a serem executadas. Assim, é possível coletar dados mais precisos sobre a utilização de diferentes usuários de forma a compará-los. Nestes casos, o avaliador normalmente determina as tarefas a serem executadas pelo usuário e pode coletar dados qualitativos ou quantitativos sobre o uso, como por exemplo, o tipo e frequência das dificuldades enfrentadas pelos usuários, caminhos preferenciais, o tempo gasto para executar a tarefa, ou o número de erros cometidos.

### **Registro de uso**

Observar o usuário requer que o usuário e o observador estejam presentes em uma mesma hora e local por um determinado período de tempo. No entanto, isto nem sempre é possível, ou se gostaria de ter informações sobre o uso em um período de tempo mais longo. Uma outra forma de coletar informações sobre como os usuários usam o sistema

é através de registros feitos durante o uso. Isto pode ser feito através de logs, que armazenam em um arquivo as ações executadas em um sistema, através da gravação da interação do usuário com o sistema, ou da gravação em vídeo da experiência do usuário. As diferentes formas de registro armazenam aspectos distintos do uso feito. Normalmente, registros de uso geram uma grande quantidade de informação e a análise destes dados pode ser bastante custosa para avaliadores.

### **Coleta da opinião de especialistas**

Em algumas situações os usuários não estão disponíveis para participar da avaliação, ou o seu envolvimento implica um custo elevado. Nestas situações, a avaliação pode ser feita sem a participação dos usuários. Neste caso, pode-se considerar a coleta da opinião de especialistas em IHC e/ou no domínio da aplicação. Os especialistas examinam a interface e identificam possíveis dificuldades que os usuários podem vir a ter ao utilizar o software. A seção 6.3 apresenta métodos de avaliação baseados na inspeção da interface por especialistas.

#### **6.2.3 Tipo de Dados Coletados**

Os dados coletados a partir de uma avaliação de interface podem ser quantitativos ou qualitativos. Dados quantitativos são aqueles que podem ser representados numericamente. Normalmente dados quantitativos são utilizados para se avaliar a eficiência e produtividade de um sistema, para se comparar alternativas de design ou ainda se determinar se o sistema atingiu algum objetivo de qualidade de uso pré-definido. A análise destes dados frequentemente é feita utilizando-se cálculos estatísticos simples, como desvio padrão ou médias. Alguns exemplos de dados quantitativos são o número de erros ocorridos ou o tempo gasto para completar uma tarefa.

**Dados qualitativos** são resultados não numéricos, tais como uma lista de problemas que os usuários tiveram ao utilizar a aplicação, ou suas sugestões sobre como melhorar o projeto de interação. Normalmente estes dados permitem identificar quais são as características de interação ou interface relacionadas com os problemas medidos e observados. Em alguns casos, dados qualitativos podem ser categorizados e então quantificados.

#### **6.2.4 Tipo de Análise**

Os avaliadores podem analisar os dados coletados de forma preditiva, interpretativa ou experimental. A **análise preditiva** é realizada quando os avaliadores, ao analisarem os dados coletados de especialistas, tentam prever que tipo de problemas os usuários enfrentarão. Esta análise pode ser feita através de uma inspeção da interface ou em função de técnicas de modelagem.

A **análise interpretativa** é realizada quando, ao analisarem os dados coletados a partir da interação do usuário com o sistema, os avaliadores procuram explicar os fenômenos que ocorreram durante esta interação. Normalmente se considera a análise como sendo interpretativa quando ela é feita sobre dados coletados em ambientes naturais sem interferência dos observadores nas atividades dos usuários.

Os dados coletados em ambientes controlados, como laboratórios, precisam ser analisados em função das variáveis sendo observadas. Embora esta análise também dependa da interpretação do avaliador, diferenciamo-na da anterior, pelo fato de as variáveis sendo manipuladas serem conhecidas. Assim, denominamos este tipo de análise de **experimental**.

### 6.2.5 Guia para o planejamento de uma avaliação

Para organizar as decisões que devem ser tomadas com respeito a uma avaliação de IHC, propõe-se utilizar o esquema DECIDE (Preece et al., 2002), para ajudar avaliadores inexperientes no planejamento e realização de uma avaliação. Os pontos relevantes a serem considerados são os seguintes:

- **Determinar os objetivos gerais que a avaliação deverá tratar.** Trata-se de responder as perguntas gerais: Quais são os objetivos gerais da avaliação? Quem quer realizá-la e por quê?
- **Explorar perguntas específicas a serem respondidas.** Trata-se de decompor as perguntas gerais em perguntas específicas ao sistema a ser avaliado, considerando-se os usuários-alvo e suas atividades. Estas perguntas são necessárias para permitir efetivamente que os objetivos da avaliação sejam atingidos.
- **Escolher (*Choose*) o paradigma e as técnicas de avaliação que responderão as perguntas.** Dentre os pontos a serem considerados, destacam-se o prazo, o custo, os equipamentos e o grau de conhecimento e experiência dos avaliadores exigidos por cada técnica.
- **Identificar questões práticas que devem ser tratadas.** Consideram-se aqui fatores como: perfil e número de usuários que participarão da avaliação; ambiente em que a avaliação será realizada; seleção das tarefas; planejamento e preparação do material de avaliação; alocação de pessoal, recursos e equipamentos para a realização da avaliação.
- **Decidir como lidar com questões éticas.** Quando uma avaliação envolve pessoas como participantes de testes, os avaliadores devem se certificar que os direitos destas pessoas estão sendo respeitados. Na seção 6.4 apontamos formas de lidar com as questões éticas envolvidas neste tipo de avaliação.
- **Avaliar (*Evaluate*), interpretar e apresentar os dados.** Como apontado nesta seção, os dados coletados durante uma avaliação podem variar bastante. Sendo assim, é importante considerar aspectos como a confiabilidade dos dados (se a técnica produz os mesmos resultados nas mesmas circunstâncias), sua validade (se mede o que deveria medir); potenciais distorções; escopo (o quanto as descobertas podem ser generalizadas); e validade ecológica (o quanto o ambiente em que a avaliação é feita influencia ou distorce os resultados).

### 6.3. Métodos de Avaliação Analíticos

Métodos de avaliação analíticos são aqueles nos quais avaliadores inspecionam ou examinam aspectos de uma interface de usuário relacionados a usabilidade (). Normalmente, os avaliadores são especialistas em usabilidade, mas também podem ser

consultores de desenvolvimento de software especializados em um determinado estilo de interface, ou ainda usuários finais conhecedores do domínio e da tarefa, entre outros.

A **avaliação analítica** ou por inspeção é utilizada geralmente para buscar problemas de usabilidade em um projeto de interface existente, e analisar estes problemas com vistas a fazer recomendações para consertá-los e assim melhorar a usabilidade do projeto. Mack e Nielsen (1994) identificam como principais objetivos deste tipo de avaliação:

- **identificação de problemas de usabilidade:** identificar, classificar e contar o número de problemas de usabilidade encontrados durante a inspeção;
- **seleção dos problemas que devem ser corrigidos:** após identificar os problemas, a equipe de projeto deve reprojeter a interface para corrigir o maior número possível de problemas. Os problemas a serem corrigidos são priorizados de acordo com a gravidade do problema e o custo associado à correção.

Além disto, ressaltam que este tipo de avaliação promove a **formação e capacitação da equipe** com relação a projetos de interface centrados no usuário.

Existem três tipos de conhecimento envolvidos em uma avaliação analítica: conhecimento sobre o domínio, conhecimento e experiência no projeto e avaliação de interfaces de usuário, e experiência em se realizar um tipo específico de avaliação ().

O **conhecimento sobre o domínio** é necessário para determinar o que os usuários querem, do que eles precisam, quais são as tarefas mais freqüentes e quais as mais importantes.

O **conhecimento e a experiência de projeto de interfaces de usuário** são necessários para que o avaliador seja capaz de analisar os aspectos mais importantes de um projeto de interfaces (navegação, terminologia, estruturas de controle, etc.) e para que ele conheça os princípios e diretrizes disponíveis na literatura. Além disto, como os princípios e diretrizes costumam ser bastante genéricos, esta experiência é fundamental para que o avaliador saiba distinguir quando aplicar e quando ignorar um princípio ou diretriz.

Já a **experiência em realizar um tipo de avaliação específico** dá ao avaliador a capacidade de representar um cliente, bem como o conhecimento sobre o que procurar e o que relatar como resultado da avaliação.

Com base nestes tipos de conhecimento, pode-se avaliar os perfis de possíveis avaliadores conforme a seguinte escala, em ordem de preferência:

- **ideal:** especialista “duplo”. Possui experiência tanto nos processos e princípios de usabilidade quanto nos processos e aspectos relevantes do domínio.
- **desejável:** especialista em IHC. Conhece o processo de avaliação, bem como os princípios e diretrizes relevantes. Pode aprender o suficiente do domínio para selecionar as tarefas que deverão ser realizadas.
- **menos desejável:** especialista no domínio. Tem conhecimento do domínio e estuda princípios de interface e o processo de avaliação para realizar a avaliação.

- **menos desejável ainda:** membro da equipe de desenvolvimento. Tem dificuldade em deixar de lado seu papel de desenvolvedor e assumir um ponto de vista semelhante ao de um usuário.

Caso o domínio do problema seja desconhecido dos avaliadores, pode ser necessário incluir no planejamento da avaliação um especialista no domínio, para esclarecer eventuais dúvidas dos avaliadores. Esta ajuda pode tomar a forma de criação de cenários de uso típicos, que listem vários passos que um usuário com determinado perfil deve seguir para realizar uma amostra realista de tarefas. Tais cenários devem ser construídos com base em uma análise das tarefas dos futuros usuários do sistema.

Existem diversos tipos de avaliação analítica: avaliação heurística, percurso cognitivo, percurso pluralista, conformidade com diretrizes e padrões, e inspeções de consistência, de características ou formais (). Nesta seção, descrevemos os métodos de avaliação heurística e percurso cognitivo. Como ilustração dos métodos de avaliação descritos nesta seção e na próxima será apresentado a avaliação feita na primeira versão do Projeto Oré (Oré, 2001). Nem todos os métodos apresentados foram utilizados no Projeto Oré, mas para aqueles que não foram, apresentaremos um exemplo de como poderia ter sido a aplicação do método. O Quadro 1 contém uma breve descrição do Projeto.

#### Quadro 1- Descrição do Projeto Oré

O Projeto Oré (Oré, 2001) tem por objetivo desenvolver um ambiente de *groupware* para apoiar o trabalho da comunidade de uma organização não-governamental, a Associação Saúde-Criança Renascer (ASCR). A primeira etapa no desenvolvimento deste ambiente foi o projeto de um quadro de avisos, onde funcionários e voluntários da ASCR poderiam colocar e ler avisos sobre as atividades e notícias relevantes. Durante a etapa de design identificou-se que grande parte dos membros da comunidade da ASCR tinham uma grande resistência ao uso de computadores (Barbosa et al., 2001). Assim, um dos objetivos do design do Quadro de Avisos era que este fosse simples o suficiente para que as pessoas o aprendessem rapidamente e facilmente, ajudando-as, em alguns casos, a superar seus sentimentos negativos relativos ao computador.

O conjunto de funcionalidades do Quadro de Avisos era pequeno e continha funções para criar, alterar, remover ou ler um aviso. Além disso, ele possuía dois espaços, um público e outro privativo da comunidade. O espaço público estaria aberto ao público em geral, e no qual só se podia ler os avisos classificados como públicos. No espaço privativo o membro tinha acesso a todos os avisos públicos e mais aqueles privativos relacionados com o trabalho em que estava envolvido na ASCR. Além disso, para facilitar o uso da aplicação decidiu-se oferecer mecanismos de explicação e ajuda ostensivos aos usuários. Quando acionados, estes mecanismos apresentavam uma ajuda em função do contexto em que o usuário se encontrava, de forma simples e direta. Caso desejasse, o usuário poderia se aprofundar mais nas explicações. Como avaliação do projeto foi feita uma inspeção informal e um teste em laboratório utilizando o protótipo proposto do quadro de avisos. A partir dos problemas identificados nesta avaliação foi projetada uma nova interface para o Quadro de Avisos.

### 6.3.1 Avaliação Heurística

O método de avaliação heurística é um método analítico que visa identificar problemas de usabilidade conforme um conjunto de heurísticas ou diretrizes (*guidelines*) (Nielsen, 1994). Ele se baseia em melhores práticas definidas por profissionais experientes e especialistas em IHC, ao longo de diversos anos de trabalho nesta área.

Este método não envolve usuários, e deve ser realizado por avaliadores especialistas. Em geral, recomenda-se que 3 a 5 especialistas realizem uma avaliação heurística. 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, na qual se definem:

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

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

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.

Nielsen propôs um conjunto básico<sup>1</sup> de heurísticas (). Cada elemento de interface (ou conjunto de elementos) deve ser analisado para verificar sua conformidade com cada uma das seguintes heurísticas:

- **visibilidade do estado do sistema:** mantenha os usuários informados sobre o que está acontecendo, através de *feedback* adequado e no tempo certo.
- **correspondência entre o sistema e o mundo real:** utilize conceitos, vocabulário e processos familiares aos usuários.
- **controle e liberdade do usuário:** forneça alternativas e “saídas de emergência”; possibilidades de *undo* e *redo*
- **consistência e padronização:** palavras, situações e ações semelhantes devem significar conceitos ou operações semelhantes; caso haja convenções para o ambiente ou plataforma escolhidos, estas devem ser obedecidas
- **prevenção de erro:** tente evitar que o erro aconteça, informando o usuário sobre as conseqüências de suas ações ou, se possível, impedindo ações que levariam a uma situação de erro
- **ajuda aos usuários para reconhecerem, diagnosticarem e se recuperarem de erros:** mensagens de erro em linguagem simples, sem códigos, indicando precisamente o problema e sugerindo de forma construtiva um caminho remediador
- **reconhecimento** em vez de memorização: torne objetos, ações e opções visíveis e compreensíveis
- **flexibilidade e eficiência de uso:** ofereça aceleradores e caminhos alternativos para uma mesma tarefa; permita que os usuários customizem ações freqüentes
- **design estético e minimalista:** evite porções de informação irrelevantes. Cada unidade extra de informação em um diálogo compete com as unidades de informação relevantes e reduz sua visibilidade relativa.
- **ajuda e documentação** devem ser fáceis de buscar, focadas no domínio e na tarefa do usuário, e devem listar passos concretos a serem efetuados para atingir seus objetivos

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 está lá”, ou seja, precisa ser incluído na interface

---

<sup>1</sup> Nielsen afirma que este conjunto pode ser estendido de acordo com características específicas a um domínio ou ambiente computacional.

Já a **gravidade** (severidade) do problema é calculada, por cada especialista, como uma combinação dos fatores:

- frequência com que o problema ocorre: É um problema comum ou raro?
- impacto do problema: Será fácil ou difícil para os usuários superarem o problema?
- persistência do problema: É um problema que ocorre apenas uma vez e que os usuários conseguem superar facilmente, ou os usuários serão incomodados pelo problema repetidas vezes?

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 atende as tarefas dos usuários
- caminhos de interação primários (importantes e/ou frequentes)
- caminhos de interação alternativos ou pouco utilizados
- consistência
- elementos de estilo
- 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.

#### **Quadro 2- Avaliação Heurística no Projeto Oré.**

Para avaliação do Quadro de Avisos foi feita a avaliação do método de comunicabilidade (seção 6.4.4). Além disso, o método de avaliação heurística foi aplicado informalmente (seção 6.5.3) por especialistas em IHC que conheciam o projeto, mas não participaram do desenvolvimento do seu projeto de interação. A Figura 4 apresenta a tela principal do Quadro de Avisos e a seguir o Quadro 3 apresenta alguns dos problemas identificados durante a avaliação heurística.



Prates, R. O., Barbosa, S. D. J. (2003) *Avaliação de Interfaces de Usuário - Conceitos e Métodos*. Jornada de Atualização em Informática, SBC. (Versão revista)



Figura 4 – Tela principal do Quadro de Avisos do primeiro protótipo do Projeto Oré

### Quadro 3 - Problemas identificados na Avaliação Heurística

**Problema:** O usuário não conseguirá entender que o texto “privativo da comunidade” lhe dá acesso a um espaço com mais funcionalidades do que aquele em que ele se encontra.

**Heurística violada:** correspondência entre o sistema e o mundo real

**Explicação:** Embora na sede da ASCR tenha alguns espaços que normalmente só são acessíveis por membros da comunidade, o usuário não utiliza a palavra “privativo” no seu cotidiano e não saberá a que ela se refere.

**Gravidade:** 4 – catastrófico. O usuário não conseguirá acessar as funcionalidades que estão disponíveis apenas para membros, como por exemplo ler avisos específicos ao trabalho em que está envolvido, ou criar um novo aviso.

**Problema:** O texto “Quadro geral” não transmite a idéia do que está sendo visualizado

**Heurística violada:** reconhecimento

**Explicação:** O que está sendo mostrado na seção denominada Quadro Geral são os avisos do Quadro de Avisos que foram colocados em destaque

**Gravidade:** 3 – grave. Como os usuários na sua maioria têm pouca experiência com informática, pode não ficar claro para eles que os avisos no Quadro geral são aqueles selecionados para estarem em destaque e podem aparecer também em outras seções. Isto pode comprometer o entendimento do usuário sobre como utilizar o Quadro de Avisos.

**Problema:** A representação das diversas seções do Quadro de Avisos não está clara

**Heurística violada:** correspondência entre o sistema e o mundo real

**Explicação:** O Quadro de Avisos foi dividido em vários setores, possibilitando diferentes visualizações deste. No entanto, no mundo real um quadro de avisos não tem visualizações distintas, e além disso na ASCR não existem quadro de avisos específicos aos setores com acesso restrito às pessoas envolvidas naquele setor.

**Gravidade:** 4 – catastrófico. O usuário pode não conseguir entender que conceitos do que ele conhece de quadro de avisos podem ser transportados para o sistema, e o que é diferente. Ele poderá ter grande dificuldade em entender e utilizar o sistema.

### 6.3.2 Percurso Cognitivo

O percurso cognitivo é um método analítico que avalia uma proposta de projeto de IHC no contexto de tarefas específicas do usuário (). Ele visa avaliar principalmente a facilidade de aprendizado do sistema, em particular pela exploração dos usuários. A motivação para este tipo de avaliação advém do fato de que muitas pessoas preferem aprender sobre a funcionalidade de um sistema computacional enquanto trabalham em suas tarefas típicas, adquirindo conhecimento sobre novas características ou funções apenas quando seu trabalho as requerer.

Neste método, o custo de aprendizado deve ser determinado pelo benefício imediato aos seus usuários. Este método investiga principalmente:

- a correspondência entre a conceitualização de uma tarefa por parte dos usuários e dos designers
- escolha adequada (ou inadequada) de termos, ou seja, o vocabulário utilizado
- *feedback* adequado (ou inadequado) para as conseqüências de uma ação

A hipótese subjacente a este método é que, num bom projeto de interface, as intenções dos usuários causam a seleção da ação adequada. Caso isto não aconteça, são levantadas hipóteses sobre as possíveis causas dos problemas, para que sejam estudadas e propostas soluções alternativas.

O percurso cognitivo também não envolve usuários. Ele pode ser realizado individualmente, pelo próprio projetista que está propondo a solução, ou em grupo. Este grupo pode incluir: o projetista, a equipe de projeto, pessoal de marketing e de documentação, além de especialistas em avaliação de interfaces.

Antes de realizar a avaliação, deve-se passar por uma fase de preparação, na qual se definem:

- hipóteses sobre os usuários e sobre o conhecimento que eles têm sobre a tarefa e sobre a interface proposta
- cenários de tarefas, construídos a partir de uma seleção de tarefas importantes e de tarefas freqüentes
- seqüência “correta” de ações para completar cada tarefa, tal como definida pelo projetista

- proposta de design em papel ou protótipo, ilustrando cada passo e indicando o estado da interface antes/depois de cada um

O procedimento de execução da avaliação compreende os seguintes passos:

1. o projetista apresenta uma proposta de design
2. os avaliadores constroem histórias plausíveis sobre a interação de um usuário típico com a interface, com base nos cenários de tarefas selecionados
3. os avaliadores simulam a execução da tarefa, efetuando uma série de perguntas sobre cada passo
4. os avaliadores anotam pontos-chave, como:
  - o que o usuário precisa saber antes de realizar a tarefa
  - o que o usuário deve aprender ao realizar a tarefa

A cada passo da tarefa, os avaliadores devem se fazer uma série de perguntas, buscando descobrir problemas em potencial, ou seja, que poderiam ocorrer durante a interação de usuários reais com o produto final implementado conforme aquela proposta de design. A lista a seguir apresenta as perguntas básicas, assim como perguntas adicionais mais específicas associadas a cada uma delas:

- O usuário tentará atingir a meta correta?
  - Dada a decomposição de uma tarefa em subtarefas, o usuário saberá por onde começar? Saberá qual é o próximo passo?
  - O que o usuário vai tentar fazer a cada momento?
- O usuário perceberá que a ação correta está disponível?
  - Onde está o elemento de interface correspondente ao próximo passo?
  - Que ações a interface torna disponíveis?
- O usuário associará o elemento de interface correto à meta a ser atingida?
  - O elemento de interface revela seu propósito e comportamento?
  - O usuário consegue identificar os elementos de interface?
- Se a ação correta é tomada, o usuário perceberá que progrediu em direção à solução da tarefa?
  - Como a interface apresenta o resultado de cada ação?
  - O resultado apresentado tem correspondência com o objetivo do usuário?

Considerando-se cada um destes critérios, há algumas soluções típicas para as falhas mais comuns:

- “O usuário tentará atingir a meta correta?” Se a interface falhar neste ponto, há pelo menos três soluções possíveis:
  - eliminar a ação, combinando-a com outra ou deixando o sistema assumi-la;

- fornecer uma instrução ou indicação de que a ação precisa ser realizada;
- alguma parte da tarefa pode ser modificada para que o usuário entenda a necessidade desta ação.
- “O usuário perceberá que a ação correta está disponível?” Se o usuário possui os objetivos certos mas não sabe que a ação está disponível na interface, a solução pode ser:
  - atribuir um operador mais óbvio para a ação. Isto geralmente requer um item de menu ou instrução, em vez de combinação de teclas;
  - atribuir à ação um controle menos visível mas como parte de uma estrutura em que possa ser facilmente encontrado, como um submenu.
- “O usuário associará o elemento de interface correto à meta a ser atingida?”

Para corrigir este tipo de falha, o designer precisa conhecer os usuários e ter uma idéia sobre como eles descreverão suas tarefas. Com esta informação, o designer poderá fornecer rótulos e descrições de ações que incluam palavras que os usuários utilizariam para descrever estas tarefas. Também pode ser necessário modificar os rótulos de outros controles, para que os usuários possam decidir sobre qual é o correto.

- “Se a ação correta é tomada, o usuário perceberá que progrediu em direção à solução da tarefa?”

Algum *feedback* é melhor do que nenhum. O *feedback* que indica **o que** ocorreu é melhor do que aquele que indica que **algo** ocorreu. Além disto, o *feedback* será mais útil quando for composto de termos ou símbolos relacionados à descrição do usuário para aquela tarefa. Observe que, em situações simples, a interface pode prescindir de *feedback* e chamar logo a próxima ação lógica

#### Quadro 4 - Percurso Cognitivo no Projeto Oré.

O método de Percurso Cognitivo não foi utilizado na avaliação do Quadro de Avisos do Projeto Oré. No entanto, abaixo ilustramos como uma tarefa típica, como criar aviso, poderia ter sido avaliada através deste método. A seguir mostramos o exemplo do material que seria gerado na etapa de preparação para a avaliação:

**Usuários típicos:** Membros da comunidade ASCR com pouca familiaridade no uso de computadores, e muitas vezes com uma resistência ao seu uso

**Tarefa:** Incluir um aviso no quadro de avisos

**Cenário:** O usuário deseja criar um aviso no quadro de avisos para os voluntários da recreação.

**Seqüência correta de ações:**

1. Entrar no espaço privativo do Quadro de Avisos
2. Entrar login e senha
3. Selecionar a opção Criar Avisos
4. Preencher campos (pelo menos os obrigatórios)
5. Confirmar criação do aviso

A Figura 5 mostra os campos a serem preenchidos na tela de criar aviso e o Quadro 5 mostra

algumas das perguntas e respostas plausíveis que poderiam ter sido geradas pelo especialista.

The screenshot shows a web browser window titled "Oré/Renascença - Quadro de Avisos - Microsoft Internet Explorer". The address bar shows a URL from "ilua.exe/qa/aviso\_editar.html". The form contains the following fields and elements:

- \* autor: [dropdown menu]
- \* título: [text input]
- \* seção: [dropdown menu] ☐ incluir no quadro geral
- \* chamada: [text area]
- aviso: [text area] (leia mais)
- prazo de validade: [text input] (dd/mm/aaaa)
- \* campos obrigatórios
- Buttons: incluir aviso, voltar

Figura 5 - Campos a serem preenchidos na tela de criar aviso do Projeto Oré

**Quadro 5 - Exemplos de possíveis perguntas e respostas para uma avaliação do Quadro de Avisos utilizando o Percurso Cognitivo**

**1. Entrar no espaço privativo do Quadro de Avisos**

–*Usuários saberão que devem entrar no espaço privativo?*

Não, os usuários não entenderão que estão em um espaço público em que têm disponíveis algumas funcionalidades e podem passar para um privativo onde terão acesso a mais opções. Logo, eles vão procurar uma opção para criar aviso no espaço público.

–*Usuários saberão que devem selecionar a opção privativo da comunidade?*

Não, os usuários não estão familiarizados com o termo privativo da comunidade. No ambiente físico o que é privativo a um grupo, fica localizado na sala onde aquele grupo exerce suas funções, mas o termo “privativo” não é utilizado.

**2. Entrar login e senha**

– *Usuário vai entender o que entrar quando aparecerem os campos login e senha?*

Sim, apesar de *login* ser um termo computacional, ele vem sendo bem difundido e pessoas em outros contextos utilizam o termo. Embora os usuários utilizem pouco o computador, vários deles tem acesso à *Internet*, onde se utiliza os conceitos de *login* e senha.

–Ao se logar o usuário vai perceber que está no espaço privativo?

Não. O feedback de entrada no espaço privativo é muito sutil, apenas mudanças nos setores de avisos disponíveis e novas opções de ações. O aspecto visual é o mesmo do espaço público, dificultando a percepção da mudança de espaço.

### **3. Selecionar a opção Criar Avisos**

– O usuário vai entender que a opção desejada é Criar aviso?

Sim. O conjunto de opções é pequeno, a opção Criar aviso está bem visível e o texto da opção está claro.

### **4. Preencher campos (pelo menos os obrigatórios)**

–O usuário saberá preencher os campos necessários para criar um aviso?

Não. O usuário pode não entender o campo chamada no qual deverá entrar o equivalente a um resumo ou indicador do assunto ao qual se refere o aviso, e como é um campo opcional vários usuários podem deixá-lo em branco. A opção incluir no quadro geral, não deixa claro que isto significa colocar o aviso também como destaque no espaço público e o usuário pode não entender quem terá acesso ao aviso sendo criado. O campo prazo de validade também não deixa claro que se refere a data no qual o aviso será retirado do quadro de avisos.

### **5. Confirmar criação do aviso**

– O usuário saberá que deve confirmar a ação de criação para que o aviso seja criado?

Sim. Mesmo um usuário com pouca experiência em uso de computadores está acostumado a confirmar suas ações em ambientes como caixas eletrônicos.

–O usuário reconhecerá o botão incluir aviso para que o aviso seja criado?

Sim. O texto deixa claro que o botão incluirá o aviso no Quadro de Avisos.

## **6.4. Métodos de Avaliação que Envolvem Usuários**

Nesta seção serão apresentados alguns dos principais métodos de avaliação que envolvem usuários para a coleta de dados, que são posteriormente analisados pelo especialista para identificar os problemas da interface. Em particular serão enfatizados os métodos de avaliação de interfaces feitos em ambientes controlados.

Em testes com usuários em laboratório o avaliador tem um maior controle sobre o ambiente e sobre as atividades do usuário. Assim, o avaliador consegue identificar problemas da interface que dificultam a interação, sem se preocupar com fatores externos, tais como o usuário sendo interrompido durante o uso do sistema. A principal desvantagem de testes em laboratórios é justamente fazer a avaliação fora do contexto em que a aplicação será utilizada de fato. Desta forma não se consegue identificar através de testes em laboratório fatores do ambiente que podem impactar uso do sistema.

Existem vários métodos que permitem se fazer avaliação de um sistema com usuário no laboratório. Enquanto eles têm várias características em comum, como parte da preparação e execução dos testes, eles variam no tipo de dado a ser coletado ou na análise a ser feita deste. Os testes de usabilidade, buscam avaliar a qualidade de usabilidade presente em um software, avaliando principalmente o desempenho do usuário com o software. Os testes de comunicabilidade apreciam a qualidade de

comunicabilidade, e buscam identificar os pontos do sistema que não foram bem comunicados pelo designer ao usuário. Os testes que fazem uso de protocolos verbais buscam apreciar a usabilidade de um software, mas se diferenciam do teste de usabilidade por tentarem explicitar o processo mental sendo vivenciado pelo usuário. A seguir, serão apresentadas as etapas de preparação de testes, e cada um destes métodos de teste em laboratório. Na última seção discutimos brevemente avaliação empírica feita no ambiente do usuário. Para cada etapa do processo de avaliação descrito será apresentado a avaliação feita na primeira versão do Projeto Oré (Quadro 1).

#### **6.4.1 Preparação de Testes em Laboratório**

Testes em laboratório requerem um planejamento minucioso para que o avaliador tenha de fato controle sobre as condições de teste. Isto envolve se certificar de que as condições de teste são as mesmas para todos os participantes, e de que o teste sendo executado permite a avaliação dos critérios desejados. Assim durante a etapa de preparação de teste o avaliador deve determinar o objetivo da avaliação e, em função deste, os critérios relevantes e pontos críticos, selecionar as tarefas, selecionar usuários participantes, gerar o material para o teste e executar o teste-piloto.

#### **Determinação do objetivo da avaliação**

O objetivo da avaliação pode ser tão geral quanto verificar se o software é usável ou bem compreendido. No entanto, para se preparar o teste o avaliador deve definir os critérios relevantes ou prioritários para que se verifique o objetivo desejado e pontos críticos da aplicação a serem testados. Os critérios prioritários normalmente são definidos durante a etapa de design, como sendo aspectos da qualidade de uso que se pretende que o sistema possua. Por exemplo, pode-se priorizar o desempenho e se desejar que o usuário execute a tarefa em um determinado período de tempo, ou que os usuários estejam satisfeitos com o sistema. Os pontos críticos do design normalmente estão relacionados com decisões de design, com tarefas que sejam estratégicas para o uso da aplicação, ou ainda com tarefas de uso muito freqüente. Decisões de design a serem testadas podem incluir estruturas de acesso, hierarquia e caminhos de navegação definidos na interface. Tarefas a serem consideradas estratégicas normalmente são aquelas essenciais para o sucesso da empresa ou do próprio usuário. Finalmente, as tarefas de uso freqüente são aquelas que se prevê que serão utilizadas comumente pelo usuário durante sua interação com o software.

#### **Quadro 6 - Objetivo da avaliação do Projeto Oré**

Como um dos objetivos do design era que o quadro de avisos fosse simples para pessoas leigas que usam o computador eventualmente fossem capazes de usá-lo bem, o principal objetivo da avaliação era observar se os usuários conseguiam compreender o que podiam fazer e como usar o Quadro de Avisos. Em função do objetivo da avaliação, definiu-se que a qualidade de uso a ser avaliada seria a comunicabilidade, e logo o planejamento para um teste de comunicabilidade foi feito.

Assim, os pontos críticos a serem avaliados foram identificados como:

- O usuário consegue entender e executar as tarefas básicas de achar e ler um aviso?
- O usuário consegue utilizar o sistema de ajuda oferecido? Este é útil para ele?

O usuário consegue entender a diferença entre os espaços público e privativo? Consegue passar do público para o privativo?

### **Seleção de tarefas**

Uma vez definidos os objetivos da avaliação, o avaliador deve determinar as tarefas a serem executadas durante o teste, que poderão fornecer indicadores sobre estes objetivos. As tarefas devem ser típicas, no sentido de serem tarefas tão realistas quanto se possa prever sobre o uso a ser feito do sistema. No caso de testes de usabilidade, o avaliador deve definir também as medidas a serem observadas para cada aspecto que se deseja apreciar. Por exemplo, para se avaliar o critério de produtividade, possivelmente será desejável medir o tempo gasto no desempenho de cada tarefa, e o número de erros cometidos por tarefa.

Na definição das tarefas, o avaliador deve tomar cuidado com o tempo previsto para cada tarefa e para o teste. Normalmente o tempo de execução de uma tarefa deveria ser de no máximo 20 minutos, e deveria se manter o tempo de teste inferior a uma hora para não ser muito cansativo para o participante (). Além disso, deve-se pensar no tempo necessário para a análise do material coletado durante o teste.

#### **Quadro 7 - Seleção das tarefas para o Projeto Oré**

Foram selecionadas 5 tarefas que permitiam aos observadores apreciar a compreensão que os usuários tinham sobre os pontos críticos identificados. As tarefas selecionadas foram:

- Tarefa 1: Encontrar um aviso no espaço público
- Tarefa 2: Identificar que um aviso deveria estar no espaço privativo, entrar no espaço privativo e encontrá-lo
- Tarefa 3: Criar um aviso
- Tarefa 4: Alterar um aviso
- Tarefa 5: Utilizar o mecanismo de busca para encontrar um aviso, identificar quem teria acesso a ele.

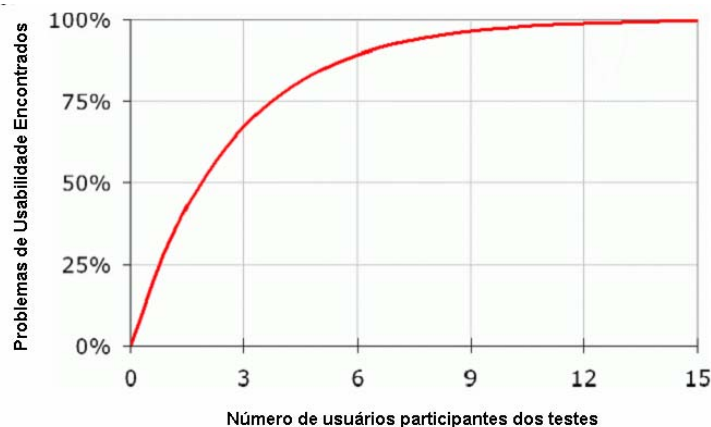
### **Seleção de usuários participantes**

O avaliador deve definir o perfil dos usuários a participarem do teste. Normalmente, o objetivo é ter usuários que representem usuários típicos do sistema. O usuário típico depende do tipo de sistema sendo desenvolvido e seu público alvo. Por exemplo, usuários típicos de um software educacional podem ser crianças, de um software de apoio a 3<sup>a</sup>. idade podem ser idosos, ou de um sistema de programação podem ser usuários experientes. Normalmente o público alvo do sistema é definido no início do ciclo de design. Para a avaliação deve-se identificar participantes que correspondam ao perfil desejado, o que normalmente pode ser feito através de um questionário breve. Em alguns casos a aplicação é destinada a grupos de usuários de diferentes perfis. Nestes casos, se recomenda que testes sejam feitos com participantes de cada perfil identificado.

Para a seleção dos participantes é recomendado que se tenha um número equivalente de homens e mulheres no conjunto de participantes do teste, a não ser que o sexo do participante seja uma das características do perfil desejado. Além disso, uma característica relevante de se considerar na seleção é a experiência dos participantes com o sistema (se este já estiver sendo utilizado), ou com outros similares.



Além de definir quais usuários poderiam participar do teste, o avaliador deve definir quantos usuários farão o teste. Vale ressaltar que o objetivo destes testes não é chegar a resultados estatisticamente válidos, mas se ter indicações de como melhorar a qualidade de uso da interface. Tipicamente em testes com usuários se envolve de 5 a 12 usuários (Dumas e Redish, 1999). Nielsen por sua vez recomenda que 5 usuários participem da avaliação (Nielsen, 2000). Segundo ele, estudos mostram que este número apresenta a melhor relação custo-benefício. Isto porque o teste com um usuário é capaz de identificar aproximadamente 30% dos problemas da aplicação. Cada novo usuário, encontra 30% de problemas, destes uma parte representa novos problemas, enquanto a outra representa problemas encontrados pelos usuários anteriores. Desta forma, a cada novo teste se reduz o número de novos problemas, e se aumenta o número de problemas já encontrados. Segundo Nielsen com 5 usuários se encontra aproximadamente 85% dos problemas da aplicação e o benefício dos novos erros encontrados vale o custo do teste executado. A Figura 6 abaixo ilustra o custo e benefício dos testes em função dos números de problemas encontrados e do número de testes executados. Nos casos em que a aplicação se destina a usuários de perfis distintos, Nielsen recomenda que para cada perfil identificado se faça a avaliação com 3 (três) usuários. Isto porque muitas vezes usuários de perfis distintos identificam os mesmos problemas.



**Figura 6 - Custo x benefício de execução de testes.**  
(Fonte: Nielsen, J.: *Test with 5 Users*, Alertbox, 2000)

Nielsen ressalta ainda que, embora um teste com 15 usuários permita potencialmente que se encontre todos os problemas de uma aplicação (ver Figura 6), vale mais a pena fazer três testes com 5 usuários do que um com 15. Isto porque com um teste com 15 usuários, todos os problemas poderão ser encontrados, mas a solução a ser desenvolvida após o teste não será avaliada e pode conter novos problemas. Em contrapartida três testes com 5 usuários, permitirá que se encontre potencialmente 85% dos problemas da aplicação no primeiro teste. A solução desenvolvida será também avaliada, e 85% dos problemas desta nova solução, sejam eles inseridos pela nova solução, sejam eles já existentes na versão anterior, mas não detectados pelo teste anterior, serão descobertos. Assim, incrementalmente se caminha para uma melhor solução final.

Outro fator a ser considerado na seleção de usuários e preparação do teste é a motivação do usuário. Normalmente, quando o participante é um potencial usuário e não tem resistência em relação à nova tecnologia, ele tem interesse em participar do teste e contribuir para a melhoria da aplicação. Quando o teste envolve representantes da sociedade, com frequência se oferece alguma compensação financeira para a participação no teste.

#### **Quadro 8 - Seleção dos participantes para o Projeto Oré**

Para participação dos testes foram selecionados 6 usuários que correspondiam aos perfis do público alvo definidos durante a etapa de design. Como o grupo de potenciais participantes nesta etapa não era muito extenso, a seleção foi feita pela equipe de design a partir de uma reunião com os parceiros do Projeto Oré na ASCR em que se discutiu os perfis de participante desejado para o teste. O grupo de usuários contava com funcionários e voluntários da ASCR, 5 mulheres e apenas 1 homem (note-se que na comunidade ASCR tem um número muito maior de mulheres envolvidas que homens), que já utilizavam o computador ou na ASCR ou particularmente e com a faixa etária variando de 30 e poucos a 60 e poucos anos.

#### **Geração do material para o teste**

Após definidos objetivos, tarefas e perfis desejados dos participantes do teste, deve-se então gerar o material a ser utilizado durante o teste. Este material inclui o questionário para seleção de participantes do teste, scripts de apresentação e explicação do processo de teste aos usuários, formulários de consentimento do usuário, texto de descrição da tarefa e roteiros de entrevista ou questionários.

O questionário para seleção de participantes identifica as características relevantes do perfil de usuário típico, e permite que candidatos sejam comparados ao perfil desejado e selecionados. Os scripts de apresentação e explicação do processo e teste aos usuários têm por objetivo garantir que todos os usuários serão tratados da mesma forma e receberão as mesmas informações. Para se executar testes com o usuário deve-se ter o consentimento formal deste (como será discutido na seção 6.4.2). Para isso, deve-se ter um formulário de consentimento a ser preenchido e assinado antes do teste.

As tarefas a serem executadas são também apresentadas aos usuários por escrito, na forma de um cenário, ou seja, de uma descrição de uma situação plausível e contextualizada de uso. Este cenário tem por objetivo permitir ao usuário se identificar com a situação e tomar a motivação apresentada como própria. Além disso, a apresentação da tarefa de forma uniforme a todos os participantes garante que todos terão as mesmas informações ao executar a tarefa.

Com frequência os testes são seguidos de entrevistas ou questionários que buscam entender melhor as ações do usuário ou avaliar sua percepção e satisfação com a aplicação. Nestes casos, os roteiros de entrevista ou questionários a serem apresentados aos participantes também devem ser gerados previamente.

#### Quadro 9 - Geração de material para o Projeto Oré

Para o teste de comunicabilidade do Quadro de Avisos foram gerados os seguintes materiais:

##### **Script de explicação sobre o projeto:**

Não foi gerado um script para explicação do teste e apresentação do laboratório. Esta decisão foi tomada, por ter a equipe de avaliação sido avisada de que os membros da comunidade, voluntários para participar do teste, estavam muito nervosos. Desta forma, com o objetivo de deixá-los mais à vontade, optou-se por recebê-los de forma mais informal explicando para cada um o objetivo e procedimento do teste, reforçando o fato de que era o Quadro de Avisos que estava sendo testado, e mostrando-lhes o ambiente.

##### **Script de apresentação do Quadro de Avisos e a sua utilidade dentro do contexto da ASCR:**

O **Quadro de Avisos** do Renascer tem por objetivo permitir a melhor divulgação das informações e necessidades do Renascer, tanto para os voluntários quanto para o público em geral. Além de acesso a informações gerais e públicas, os voluntários têm acesso a informações privativas, relacionadas com o trabalho que executam no Renascer. Para ter acesso a estas informações os voluntários devem: (1) estar cadastrados como usuários do **Quadro de Avisos**; (2) entrar na página deste sistema na *Internet*; e (3) optar por acesso ao espaço privativo da comunidade, para o quê devem informar seu *login* e senha na página de acesso. Para as atividades de hoje seu *login* é teste e sua senha é senha. Se você tiver alguma dúvida sobre como usar o **Quadro de Avisos**, você pode acessar tanto as informações de ajuda disponíveis localmente através do ponto de interrogação azul, quanto através da ajuda geral do sistema.

##### **Documento de consentimento:**

O documento garantia ao usuário o seu anonimato, explicava como seriam utilizados os dados coletados durante o teste, reforçava que ele poderia interromper o teste a qualquer momento caso desejasse e perguntava-lhe se tinha alguma condição a acrescentar para a realização dos testes e utilização dos dados. Após preenchido este documento era assinado pelo participante do teste e pelo avaliador.

##### **Texto de descrição da tarefa:**

Cada tarefa foi descrita em um cenário para permitir ao usuário se contextualizar e entender o que deveria fazer. Abaixo mostramos o exemplo de uma das tarefas utilizadas no teste do Quadro de Avisos:

##### **Atividade 2:**

Você é voluntário do Renascer e trabalha tanto no atendimento, quanto na recreação. A salinha de recreação do hospital no momento está fechada, mas na reunião mensal dos voluntários no Parque Lage você soube que ela deve ser reaberta logo. Você soube também que antes de todos poderem voltar a trabalhar haverá uma reunião com os voluntários da recreação para a apresentação das novas regras de uso da salinha. Porém, outro dia lhe disseram que a reunião já foi marcada e que seu horário/data/local estão disponíveis no Quadro de Avisos. Você então vai entrar no sistema para descobrir quando e onde será esta reunião.

##### **Formulário de acompanhamento de teste:**

Foi feito um formulário para os observadores do teste. O cabeçalho do teste possuía a identificação do participante através que deveria ser preenchido com o número atribuído a ele e a data do teste, a seguir continha um espaço para anotar as suas observações relativas ao teste e outro para anotar perguntas a serem feitas durante a entrevista sobre as ações observadas sobre as quais gostariam de uma explicação do participante.

##### **Roteiro para entrevista:**

Foi feito um roteiro simples que incluía perguntas como “Você achou alguma atividade difícil?” e “O que gostou mais/menos no Quadro de Avisos?”, além das perguntas que surgiam a partir da observação. Para cada pergunta o entrevistador poderia entrar na questão mais profundamente ou

não, conforme julgasse apropriado.

### Execução do teste piloto

Uma vez estando pronto o material para o teste, é fundamental que se faça testes-piloto para se avaliar a qualidade do material gerado. O que se procura observar durante o teste-piloto é se os participantes conseguiram entender corretamente todo o material apresentado, se o tempo de execução do teste está dentro do previsto e é viável, se através das tarefas propostas se consegue obter as medidas especificadas e avaliar o critério desejado. Além disso, pode-se aproveitar o teste-piloto para se praticar a habilidade do avaliador para deixar os participantes à vontade para o teste e os entrevistar. Desta forma se garante que os dados coletados durante o teste permitirão de fato avaliar os aspectos desejados da aplicação, e não causarão perda de dados ou no pior caso, invalidação do teste como um todo.

Idealmente, se deve executar testes-piloto até que não se identifique mas nenhuma necessidade de alteração do material. Caso isto não seja possível, deve-se garantir no mínimo a execução de um teste-piloto. Para participar do teste-piloto o ideal seria poder contar com potenciais usuários com o perfil desejado. No entanto, se o custo de envolver usuários for muito alto, ou o acesso a eles for limitado, pode-se executar os testes-piloto com colegas ou pedir-lhes para comentar o material.

#### Quadro 10 - Teste piloto para Projeto Oré

O material do teste preparado pela equipe de avaliação foi revisto e foi comentado por toda equipe de design. Uma vez feitas as alterações necessárias, foi feito um teste piloto com um voluntário não associado à ASCR. Embora este voluntário não fosse membro da comunidade Renascer, ele tinha várias características em comum com o perfil dos membros da comunidade desejado para o teste. Em função do teste piloto, mais uma vez o material foi ajustado para o teste com os usuários.

### 6.4.2 Execução dos Testes em Laboratório

A execução dos testes exige a escolha de um ambiente adequado para os testes, a atenção a questões éticas envolvidas no processo de teste com participação de outros seres humanos, e ainda deixar os usuários o mais à vontade possível para que possam agir tão naturalmente quanto consigam neste ambiente controlado e artificial.

#### Ambientes de teste

Os testes normalmente são executados em laboratórios de testes com usuários como por exemplo o mostrado na Figura 7. Os laboratórios costumam possuir 2 salas, uma para a execução do teste (Figura 7a) e outra para observação do teste (Figura 7b). As salas são separadas por um vidro espelhado, de forma que o participante não enxergue quem se encontra do outro lado do vidro, mas os observadores possam ver o participante e suas ações. A sala de teste costuma ser equipada com um computador e espaço para o participante e um avaliador. A sala de observação costuma ter um monitor que replica o que está sendo visto no monitor do usuário, um outro computador para anotações. Além disso, câmeras de vídeo e gravadores podem estar presentes em uma sala ou na outra, ou em ambas dependendo do teste a ser executado.



(a) Sala de teste: participante e um avaliador na sala. (b) Sala de observação: avaliador observando por trás do vidro espelhado.

**Figura 7 – Laboratório de comunicabilidade do Grupo de Pesquisa em Engenharia Semiótica (SERG), na PUC-Rio.**

Em alguns casos quando não se tem um laboratório disponível pode-se considerar o uso de equipamento móvel (como câmeras de vídeo, ou sistemas de registro de interação) e se utilizar uma sala disponível como laboratório. Esta opção também pode ser considerada para levar o laboratório até o usuário, quando este não pode comparecer ao laboratório disponível.

#### **Quadro 11 - Ambiente de teste para Projeto Oré**

Os testes foram realizados no laboratório. Um avaliador explicava para o participante o teste, verificava se tinha alguma dúvida em relação ao material que lhe era apresentado e ficava dentro da sala de teste junto com o participante para dar acesso a alguém da equipe durante o teste, caso alguma dúvida surgisse. O avaliador ficava no fundo da sala para interferir o mínimo possível (como mostrado na Figura 7a) e fazendo anotações. Na sala de observação (Figura 7b) ficaram 2 outros avaliadores observando o teste e fazendo anotações. Além disso, a interação de cada participante foi gravada para análise posterior.

### **Questões Éticas**

Na execução de testes que envolvem outros seres humanos o avaliador deve estar atento às questões éticas envolvidas e a como lidar com elas. Alguns governos, como o dos EUA<sup>2</sup>, regulamentam o processo de teste e têm exigências formais e burocráticas sobre

---

<sup>2</sup> No Brasil esta regulamentação existe na forma de uma resolução do Conselho Nacional da Saúde e se aplica a todos os profissionais da área de saúde (inclusive psicólogos) que realizam pesquisas com seres humanos. É de se esperar que em breve haja regulamentações semelhantes para profissionais de outras áreas que fazem pesquisas e trabalhos que afetem seres humanos em situações como a de testes de produtos, até porque estes profissionais muitas vezes têm bem menos treinamento para lidar com estas situações que profissionais da área de saúde. Disponível online em: <http://conselho.saude.gov.br/biblioteca/livro.htm>.

sua execução. Para garantir a proteção aos usuários e o atendimento a exigências éticas as seguintes diretrizes foram propostas ():

- Explicar aos participantes os objetivos do estudo sendo feito e exatamente como deverá ser a participação deles. Deve-se deixar claro o processo de teste, o tempo aproximado do teste, o tipo de dado sendo coletado e ainda como os dados serão analisados.
- Deixar claro as expectativas de anonimato dos usuários, deixando claro que dados particulares identificados durante o teste não serão divulgados.
- Certificar-se de que os usuários entendem que a qualquer momento podem interromper o teste, caso desejem.
- Sempre que trechos de depoimentos dos usuários forem utilizados eles devem ser anônimos e deve-se retirar descrições ou trechos que permitam a identificação do usuário. Nestes casos, deve-se requisitar autorização prévia do usuário e de preferência mostrar-lhe o relato a ser divulgado.

O usuário deve consentir por escrito na execução do teste e o documento de consentimento deve especificar as condições acordadas do teste e deve ser assinado tanto pelo participante do teste, quanto pelo avaliador. Além disso, o formulário deve permitir ao usuário acrescentar novas condições ao acordo, caso o deseje.

### **Deixando o usuário à vontade**

Muitas vezes, as pessoas em uma situação de teste se sentem avaliadas e ficam nervosas. Principalmente quando os testes são feitos em um ambiente controlado e artificial. Antes de iniciar o teste é importante deixar o participante o mais à vontade possível. Para isso, deve-se assegurar ao usuário que a aplicação está sendo testada e não ele, e lembrar-lhe que ele pode interromper o teste a qualquer momento. Além disso, tarefas de exploração do software e tarefas fáceis no início do teste ajudam o usuário a se sentir mais confiante. Da mesma forma, uma tarefa simples no final pode ajudar o usuário a se sentir bem em relação ao seu desempenho.

#### **Quadro 12 - Deixando o usuário à vontade no Projeto Oré**

O avaliador recebeu cada um dos participantes e inicialmente conversou amenidades e lhes ofereceu café com biscoito com o objetivo de deixá-los mais à vontade. A explicação dos objetivos e processos do teste foram feitos neste ambiente informal. O avaliador enfatizou sempre o fato de o Quadro de Avisos (e não o usuário) estar sendo testado para que a equipe pudesse melhorá-lo antes de entregá-lo à ASCR para ser usado.

### **Aplicação do teste**

Antes de iniciar o teste o avaliador deve ler para os usuários os scripts de apresentação e explicação. Apesar de o uso de script reforçar o sentimento de ambiente artificial, deve-se esforçar para que o usuário fique à vontade. Deve-se então pedir a ele que leia, preencha e assine o formulário de consentimento de execução do teste, que deve também ser assinado pelo avaliador. Algumas vezes se utiliza questionários para se conhecer especificidades sobre o usuário. Isto pode ser feito antes ou após o teste.

Dá-se então início à execução do teste propriamente dito. Primeiramente, se apresenta o material escrito relativo às tarefas ao participante, pede-o para ler o material

e verifica-se o entendimento do mesmo. O usuário então inicia a tarefa, com frequência pode-se desejar ou é necessário (no caso do teste de comunicabilidade) registrar o uso através do registro de logs, gravação da interação ou do vídeo. Enquanto o usuário executa a tarefa o(s) observador(es) faz(em) anotações sem interromper ou interferir com a concentração do usuário. O usuário muitas vezes se volta ao observador para fazer perguntas sobre as dificuldades sendo vivenciadas. As perguntas relacionadas com o uso da aplicação ou execução da tarefa não devem ser respondidas, enquanto que perguntas que não estão relacionadas com o que se quer observar podem ser respondidas.

Ao fim do teste, costuma-se entrevistar o participante ou lhe pedir para preencher o questionário. Quando se tem uma entrevista pode-se ser tirar dúvidas sobre as ações observadas do usuário, ou sobre as motivações ou expectativas que o levaram a realizá-las. Além disso, através da entrevista ou questionário pode-se colher a opinião pessoal do usuário sobre sua experiência, sua satisfação com o sistema e sugestões.

### **Análise dos dados coletados**

Note-se que durante o teste diferentes dados são coletados de diferentes formas: registro de uso, anotações de observação, preenchimento de questionários e condução de entrevistas. Na etapa de análise o avaliador deve analisar os dados coletados durante o teste para todos os usuários e a partir dele gerar o relatório do teste. O grande volume de dados normalmente implica um longo tempo necessário para sua análise. Em alguns casos, alguns dos dados coletados não são sempre analisados, mas apenas para verificação de comportamentos específicos ou para tirar dúvida sobre algum outro dado analisado. Por exemplo, quando se tem observadores fazendo anotações durante o teste, a gravação de vídeo pode ser utilizada apenas em pontos onde uma anotação feita não ficou clara, ou se gostaria de rever a reação do usuário durante um trecho da interação. Normalmente, quanto mais cedo se consegue fazer a análise dos dados após sua coleta melhor, uma vez que o que se foi observado ou relatado está fresco na memória do avaliador.

O relatório costuma descrever os testes feitos, os problemas encontrados, e dependendo do nível de informação que o avaliador tenha sobre as intenções e decisões de design, pode conter também hipóteses sobre as causas dos problemas observados. O relatório não necessariamente inclui propostas de redesign para se solucionar os problemas identificados.

### **6.4.3 Testes de Usabilidade**

O teste de usabilidade é executado em laboratório e tem por objetivo permitir que se apreciem os fatores que caracterizam a usabilidade de um software, ou seja, facilidade de aprendizado, facilidade de uso, eficiência de uso e produtividade, satisfação do usuário, flexibilidade, utilidade e segurança no uso (Nielsen, 1993; Preece et al., 2002). Quais destes fatores devem ser priorizados deve ser definido no projeto de design. Através do teste procura-se quantificar o desempenho do usuário. Para isso durante a preparação de testes em laboratório descrita anteriormente, para cada medida a ser observada, deve-se definir quais são os limites mínimos aceitáveis, os máximos possíveis (em outras palavras, o melhor e pior caso) e também o valor almejado para a medida no projeto. A quantificação do desempenho normalmente envolve a medição do

tempo e de ações de usuários. Apenas a satisfação do usuário se distingue e normalmente é medida através da coleta de opinião do usuário e cujos limites mínimos, máximos e almejados costumam ser definidos em função da porcentagem de usuários que se dizem ou não satisfeitos com o software e o seu nível de satisfação. Alguns exemplos de medidas comumente utilizadas no teste de usabilidade são tempo gasto para se executar uma tarefa, número de erros executados, porcentagem de usuários a conseguirem se recuperar de um erro, ou porcentagem de usuários a se dizerem satisfeitos com a aplicação, ou a preferirem a aplicação a um outro sistema sendo utilizado.

Na análise dos dados coletados durante o teste de usabilidade o avaliador primeiramente classifica os problemas pela sua gravidade (Nielsen, 1998):

- Problema catastrófico: impede que o usuário termine sua tarefa
- Problema sério: atrapalha a execução da sua tarefa
- Problema cosmético: atrasa a execução e/ou irrita usuários

Além disso, para cada medida observada, ele verifica a distância para limites mínimos, máximos e almejados, determinando se o critério está em um patamar desejável ou não. Através dos dados o avaliador verifica o cumprimento de metas que tenham sido previamente definidas durante a etapa de design.

#### Quadro 13 - Projeto de teste de usabilidade para Projeto Oré

Devido à grande importância da introdução de tecnologia neste projeto, e o objetivo de se avaliar principalmente como a mensagem estava sendo entendida, optou-se por utilizar o Teste de Comunicabilidade, e não o de Usabilidade, na avaliação do Quadro de Avisos. Aqui apresentamos quais seriam alguns critérios a serem considerados, e as possíveis medidas a serem alcançadas caso se tivesse decidido por um teste de usabilidade. Os principais critérios seriam:

- Facilidade de uso: O usuário consegue utilizar facilmente o Quadro de Avisos? Sem cometer muitos erros? O sistema de ajuda é eficiente no auxílio as dúvidas dos usuários?
- Produtividade: O usuário consegue criar e encontrar um aviso rapidamente? Ele é útil para a comunidade ASCR?
- Satisfação: Os usuários ficaram satisfeitos com o Quadro de Avisos?

Com base nestes critérios as tarefas propostas poderiam ser praticamente as mesmas. A Tarefa 5, possivelmente se limitaria a verificar o uso do mecanismo de busca e não do entendimento de quem poderia também ter acesso aos avisos encontrados. Além disso, a descrição dos cenários de apresentação da tarefa seriam mais diretos em relação ao objetivo da tarefa. Assim a descrição da Tarefa 2 para o usuário poderia ser por exemplo:

##### Atividade 2

Você é voluntário do Renascer e trabalha tanto no atendimento, quanto na recreação. Foi colocado no Quadro de Avisos um aviso destinado apenas aos voluntários da recreação sobre a reunião marcada para reabertura da salinha. Utilizando o Quadro de Avisos descubra o horário/data/local desta reunião.

As medidas a serem observadas seriam:

fator		método de medição	pior caso	nível almejado	melhor caso
Facilidade de uso	de	Número de erros cometidos	Mais de 10 erros	No máximo 3 erros	Nenhum erro
Facilidade de uso	de	Porcentagem de vezes que o usuário vai ao sistema de	Para cada tarefa vai pelo menos 1 vez.	Apenas a 1a. vez que realiza uma	Nunca



ajuda			tarefa complexa		
Eficiência para criar aviso	Tempo gasto para criar um aviso	5 min	40 segundos	20 segundos (tempo para digitar campos)	
Eficiência para encontrar aviso	Tempo gasto para encontrar um aviso	Não encontrar o aviso	30 segundos	10 segundos (tempo para digitar alguns campos no mecanismo de busca)	
Utilidade	Frequência de uso	Uma vez a cada três dias ou menos frequente	Uma vez ao dia	Mais de uma vez ao dia	
Eficiência do sistema de ajuda	Porcentagem das vezes que usuário encontrou o que procura no sistema de ajuda	Nunca	Acima de 90% das vezes	100% das vezes	
Eficiência do sistema de ajuda	Consegue resolver o problema com base no conteúdo de ajuda	Nunca	Acima de 90% das vezes	100% das vezes	
Avaliação inicial	Questionário (subjetivo)	Negativa	Positivo	Muito positivo	

#### 6.4.4 Testes de Comunicabilidade

Testes de comunicabilidade, como os de usabilidade, devem ser executados em laboratório. No entanto o seu objetivo é avaliar a interface com relação à qualidade da comunicação do designer para os usuários. Para isto, este método simula a comunicação do usuário para o designer sobre a interface. Isto é feito através de um pequeno conjunto de expressões que o usuário potencialmente pode usar para se exprimir em uma situação onde acontece uma ruptura na sua comunicação com o sistema (de Souza et al., 1999; Prates et al., 2000b).

No caso de testes de comunicabilidade, a gravação da interação do usuário com o sistema durante o teste deve ser feita, pois a análise será feita principalmente a partir deste registro. Além das anotações do observador durante o teste, as gravações em vídeo também podem ser feitas para enriquecer os dados, e permitir a verificação da reação do usuário relativa a algum trecho da interação observado.

A análise dos dados é dividida em 3 passos:

1. Uma etiquetagem, que consiste em assistir às gravações da interação e atribuir a expressão apropriada nos momentos de ruptura da interação;
2. Uma interpretação, que consiste em tabular e consolidar a informação obtida, ou seja, as expressões obtidas, associando-as a classificações de problemas de interação ou diretrizes de design; e
3. Um perfil semiótico, que consiste em interpretar a tabela resultante do passo anterior, dentro do quadro teórico da Engenharia Semiótica (de Souza, 1993; de Souza et al. 2001), em uma tentativa de se reconstruir a meta-mensagem sendo transmitida pelo designer ao usuário através da interface.

A seguir, apresentamos em detalhe cada uma destas etapas.

### **Etiquetagem**

Durante a etapa de etiquetagem, o avaliador<sup>3</sup> assiste as gravações da interação feitas durante a coleta de dados. Ao observar uma ruptura da interação o avaliador associa à sequência de ações problemática uma das expressões de comunicabilidade. O efeito desta etiquetagem é semelhante ao de um protocolo verbal (seção 6.4.6) “reconstruído” a partir das evidências de interação. A reconstrução é feita pelo avaliador, por meio de um conjunto de expressões, definido com o objetivo de ser o conjunto mínimo capaz de caracterizar suficientemente as rupturas de interação que acontecem durante o uso de uma aplicação. As expressões foram selecionadas com o intuito de serem naturais e espontâneas, e serem manifestações plausíveis por parte de usuários nestas situações. A seguir descrevemos o conjunto de expressões, seus significados e algumas ações de interface que caracterizam cada uma delas.

### **Cadê?**

Ocorre quando o usuário sabe a operação que deseja executar mas não a encontra de imediato na interface. Um sintoma freqüente é abrir e fechar menus e submenus e passar com o cursor de mouse sobre botões, inspecionando diversos elementos de interface sem ativá-los.

### **E agora?**

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 inspecionar os menus de forma aleatória ou seqüencial.

### **O que é isto?**

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, esperando que uma dica seja apresentada.

### **Epa!**

O usuário realizou uma ação indesejada e, percebendo imediatamente que isto ocorreu, desfaz a ação. Os sintomas incluem o acionamento imediato do *Undo* ou o cancelamento de um quadro de diálogo aberto indevidamente.

### **Onde estou?**

O usuário efetua operações que são apropriadas para outros contextos, mas não para o contexto atual (por exemplo, tenta digitar um dado em um campo desabilitado; digita um comando em um campo de dado ou um dado no campo reservado para comandos). Um sintoma típico é desfazer a ação incorreta e mudar em seguida para o contexto desejado.

### **Assim não dá.**

O usuário efetuou uma sequência (longa) de operações antes de perceber que estava seguindo um caminho improdutivo. Os sintomas incluem o acionamento de *Undo*

---

<sup>3</sup> Os autores do método acreditam que esta etapa também pudesse ser executada pelos próprios usuários (Prates et al., 2000b), mas esta hipótese ainda está sob investigação.

repetidas vezes ou o cancelamento de um ou mais quadros de diálogos abertos indevidamente.

### **Por que não funciona?**

A operação efetuada não produz o resultado esperado, mas o usuário não entende ou não se conforma com o fato. O sintoma típico consiste em o usuário repetir a ação.

### **Ué, o que houve?**

O usuário não percebe ou não entende a resposta dada pelo sistema para a sua ação (ou o sistema não dá resposta alguma). Os sintomas típicos incluem repetir a ação ou buscar uma forma alternativa de alcançar o resultado esperado.

### **Para mim está bom...**

Ocorre quando 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.

### **Desisto.**

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.

### **Vai de outro jeito.**

O usuário não consegue realizar a tarefa da forma como o projetista gostaria que ele o fizesse, e resolve seguir outro caminho, geralmente mais longo ou complicado. Cabe ao avaliador determinar, se possível junto ao designer, qual é a forma preferencial de execução da tarefa.

### **Não, obrigado.**

O usuário já conhece a solução preferencial do designer, mas opta explicitamente por uma outra forma de interação. O sintoma consiste em uma ocorrência da ação preferencial seguida de uma ou mais formas alternativas para se alcançar o mesmo resultado.

### **Socorro!**

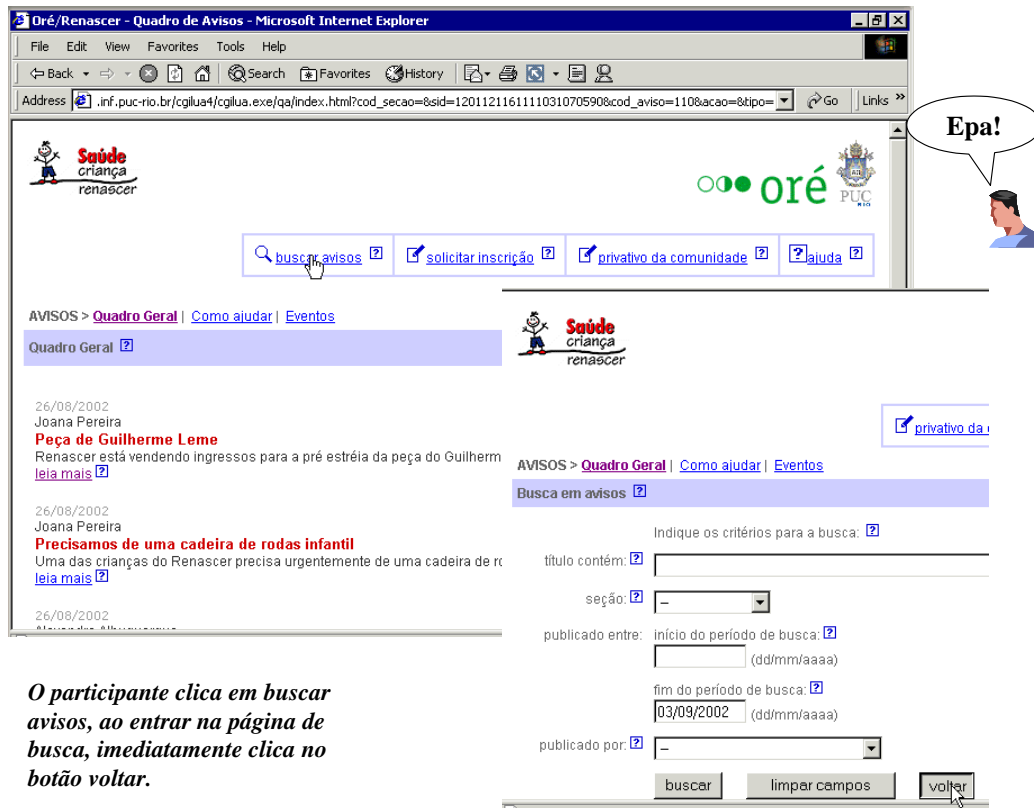
O usuário não consegue realizar sua tarefa através da exploração da interface. O sintoma é recorrer à documentação ou pedir explicação a outra pessoa.

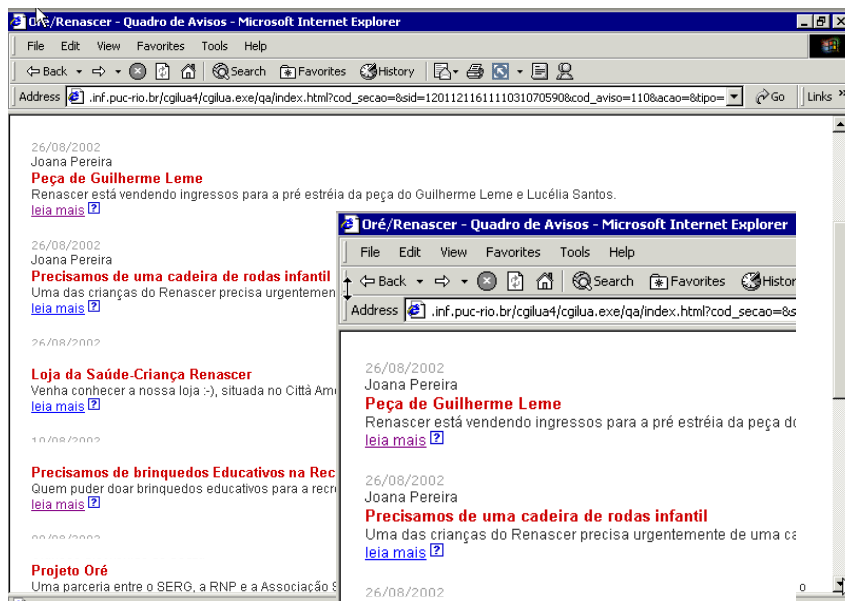
\* \* \*

Para exemplificar suponha que o participante do teste em um determinado momento pára o seu cursor sobre um elemento da interface com o objetivo de ver a dica relativa àquele elemento. Ao observar esta ação no filme da interação o avaliador associaria a ela a expressão “O que é isto?”. Ou ao observar que o usuário percorre opções do menu à procura de uma determinada função, o avaliador então associaria a esta porção da interação a expressão “Cadê?”. Assim, pode-se dizer que o processo de etiquetagem equivale ao avaliador “colocar palavras na boca do usuário” (o que chamamos acima de

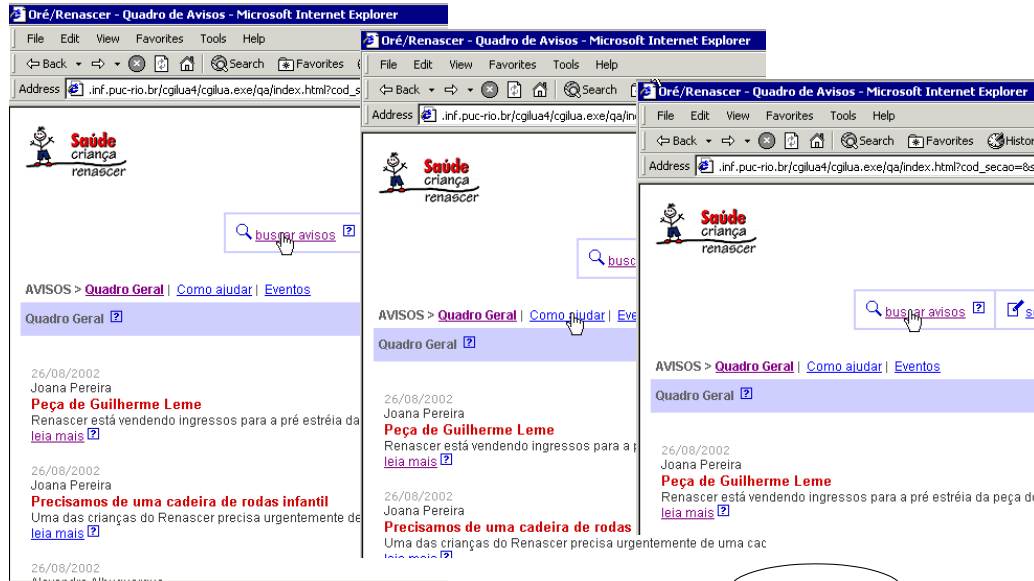
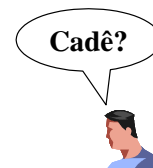
‘protocolo verbal reconstruído’), uma vez que ele estaria associando à sequência de ações o que o usuário poderia ter dito. Vale ressaltar que as expressões “Para mim está bom” e “Vai de outro jeito...” não poderiam ser ditas pelo usuário durante a interação, mas apenas pelo avaliador, ou pelo usuário ao rever o seu próprio filme.

A Figura 8 abaixo mostra a sequência da ações de um usuário durante o teste do Projeto Oré e a análise feita pelo avaliador usando as expressões de comunicabilidade.





*O participante clica na barra de rolagem para ver os avisos e vai passando o cursor sobre eles, procurando o desejado.*



*Participante fica em sem saber qual deve ser seu próximo passo. Fica em dúvida entre duas opções, mas acaba não selecionando nenhuma.*



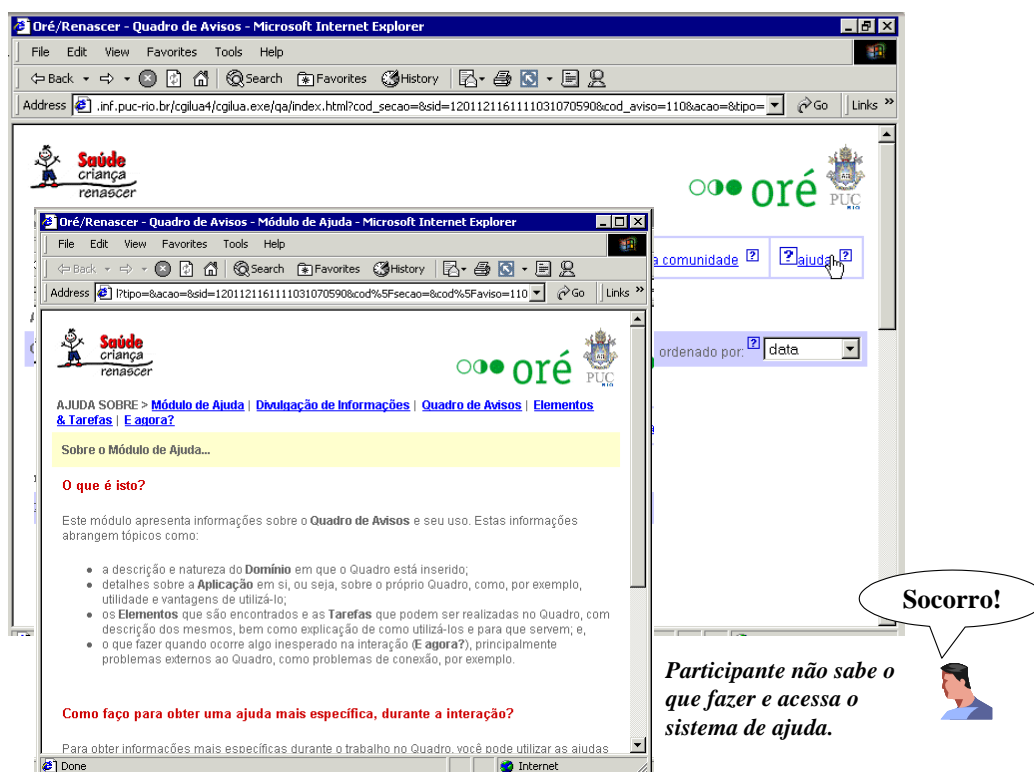


Figura 8 - Seqüência de ações de um usuário no teste do Projeto Oré

## Interpretação

Nesta etapa o avaliador deve associar as expressões identificadas a classificações de problemas de interação ou diretrizes de design. Utilizamos aqui uma classificação genérica que define os problemas de interação como sendo de navegação, atribuição de significado, percepção, falha de execução da tarefa, e incompreensão ou recusa de *affordance*<sup>4</sup>. Problemas de falha na execução da tarefa são os mais graves, uma vez que o usuário não consegue atingir o objetivo que o levou a usar a aplicação. Os de navegação se referem àqueles nos quais os usuários se “perdem” durante a interação com o sistema. Os de atribuição de significado, conforme o nome diz, acontecem quando o usuário não é capaz de atribuir um significado (relevante) a signos<sup>5</sup> encontrados na interface. Os de percepção são quando os usuários não conseguem perceber alguma resposta do sistema ou seu estado corrente. Para a associação de expressões a problemas o avaliador poderia utilizar outras classificações de problemas, como por exemplo as 8 regras de ouro de Shneiderman (1998) ou ainda as heurísticas de Nielsen (1994).

A classificação proposta apresenta duas outras classes de problemas, incompreensão e recusa de *affordance*, que normalmente não fazem parte das

<sup>4</sup> Termo que se refere às propriedades percebidas e reais de um artefato, em particular as propriedades fundamentais que determinam como este artefato pode ser utilizado (Norman, 1988).

<sup>5</sup> Signo é algo que representa alguma coisa para alguém (Peirce, 1931). Assim um signo da interface é tudo aquilo que tem um significado para alguém, no caso designer, usuário ou avaliador.

classificações mais populares (e.g as citadas acima). No caso do problema de incompreensão de *affordance*, o usuário não consegue entender uma solução oferecida pelo designer, e acaba por executar a tarefa desejada de uma forma mais complicada, que não caracteriza a solução principal do designer<sup>6</sup>. Finalmente, no caso de recusa de *affordance*, o usuário entende a solução principal oferecida, mas escolhe não utilizá-la e em seu lugar utilizar outra forma de interação que julga ser melhor. Esta outra forma pode ter sido prevista, pretendida, ou não, pelo designer.

A Tabela 1 mostra como as expressões podem ser associadas a estas classes de problemas. Observe que para as expressões “Epa!”, “Onde estou?” e “Assim não dá” a associação a uma classe de problemas não é unívoca e deve ser definida de acordo com o contexto em que foi observada a ruptura.

A partir da tabulação dos problemas encontrados o avaliador pode definir os pontos críticos da interação e gerar o relatório da avaliação.

**Tabela 1 - Associação entre expressões e classes de problemas**

Expressão de comunicabilidade	Problemas de Interação					
	Execução	Navegação	Atribuição de significado	Percepção	Incompreensão de <i>affordance</i>	Recusa de <i>affordance</i>
Cadê		X				
E agora?		X	X	X		
O que é isto?			X			
Epa!		X	X			
Onde estou?		X	X	X		
Assim não dá.		X	X	X		
Por que não funciona?			X	X		
Ué, o que houve?			X	X		
Para mim está bom...			X	X		
Não dá.	X					
Vai de outro jeito					X	
Não, obrigado.						X
Socorro!	X	X	X			

<sup>6</sup> Em casos em que o designer não participa do projeto de avaliação para esclarecer a sua solução principal, esta associação é feita em função do entendimento do avaliador sobre qual seria a solução principal oferecida.

**Quadro 14 – Teste de comunicabilidade – interpretação – Projeto Oré**

Os problemas descritos acima vivenciados pelo participante foram identificados como sendo de Atribuição de Significado, uma vez que a expectativa da equipe de design era que o participante entendesse que ele não encontrava o aviso, por ele estar disponível apenas no espaço privativo e não no público, e fosse então para este espaço.

**Perfil Semiótico**

A definição do perfil semiótico da aplicação deve ser feito por um especialista em Engenharia Semiótica (de Souza, 1993; de Souza et al. 2001). Neste passo o especialista interpreta a etiquetagem e tabulação feitas nos passos anteriores dentro do quadro teórico da Engenharia Semiótica, em uma tentativa de se reconstruir a meta-mensagem sendo transmitida pelo designer ao usuário através da interface. Desta forma, este passo acrescenta à avaliação problemas identificados na linguagem de interface da aplicação, podendo fazer considerações sobre possíveis premissas de design e conhecimentos táticos utilizados.

**Quadro 15 - Teste de comunicabilidade - perfil semiótico - Projeto Oré**

O designer pretendia que a aplicação fosse simples de usar, e procurou oferecer um apoio amplo aos usuários pouco conhecedores de tecnologia. Os usuários compreenderam bem para que servia o Quadro de Avisos, e acharam que ele seria bastante útil para a ASCR. No entanto, vários dos participantes não conseguiram entender como utilizar o mecanismo de busca e tiveram dificuldades em entender a distinção entre os espaços público e privativo. Desta forma, constatou-se que a solução do designer não estava adequada para os usuários pretendidos e precisava ser recontada de uma forma mais simples e direta. O problema parecia ser causado pelas expectativas do designer de um conhecimento básico prévio dos participantes. Por exemplo, que os participantes conhecessem estratégias de solução comumente utilizadas em tecnologia amplamente divulgadas, como mecanismos de busca no qual se especifica a busca à medida que se fornece um maior número de parâmetros, ou o controle de acesso diferenciado a diferentes espaços de uma aplicação. Com base, neste diagnóstico foi proposta uma nova solução mais explicativa e com o conjunto de ações possíveis mais evidenciada. Além disso, optou-se por oferecer apenas o espaço privativo durante a etapa de introdução e consolidação da tecnologia.

**6.4.5 Testes de Usabilidade x Testes de Comunicabilidade**

A maior diferença entre os testes de usabilidade e comunicabilidade está no conceito de qualidade de uso que eles pretendem apreciar, conforme seus próprios nomes indicam. Assim, testes de usabilidade pretendem avaliar a solução do designer, enquanto os de comunicabilidade buscam avaliar a comunicação sendo feita sobre esta solução. Para isso, os testes de usabilidade normalmente coletam dados quantitativos e buscam informar designers durante o ciclo de desenvolvimento quais critérios não correspondem ao objetivo almejado para o software. Testes de comunicabilidade, por sua vez, coletam dados qualitativos e têm por objetivo informar designers sobre pontos da sua solução que não estão sendo transmitidos com sucesso aos usuários.

**6.4.6 Protocolos Verbais**

Os métodos que envolvem protocolos verbais normalmente são testes executados em laboratório, nos quais os participantes explicitam aquilo que estão pensando, à medida que executam as tarefas propostas. A principal vantagem destes métodos sobre outros métodos de teste em laboratório é permitir que o avaliador tenha acesso aos processos



mentais dos participantes descritos por eles mesmos no decorrer da execução da tarefa. Desta forma avaliadores podem compreender melhor os comportamentos, sentimentos e atitudes dos usuários em relação à atividade sendo observada.

Um dos principais métodos de protocolo verbal utilizado é o de pensar alto (*think aloud*) (). Neste método o participante executa os testes individualmente no laboratório e narra o que está pensando à medida que o faz. Se o participante fica em silêncio por um longo período de tempo, o avaliador pode lembrar o participante de continuar dizendo o que está pensando, com perguntas como: “O que você está pensando?” ou “O que você acabou de fazer?”. No entanto, esta interrupção pode atrapalhar a atividade do participante. A maior desvantagem deste método é que o participante faz duas coisas ao mesmo tempo: executa a tarefa, e narra suas ações e pensamentos.

Uma forma de amenizar as desvantagens do método de pensar alto é colocar dois participantes juntos para executar a tarefa. Desta forma, os participantes trocam idéias entre si sobre o que fazer, o que significa o que estão vivenciando e seus sentimentos e atitudes relativos à atividade e à aplicação. Ao fazerem isso eles explicitam seus pensamentos e os avaliadores passam a ter acesso ao que estão pensando. Este método normalmente é conhecido por método de co-descoberta.

#### 6.4.7 Avaliação no Ambiente do Usuário<sup>7</sup>

Uma outra forma de se fazer avaliações é ao invés de fazê-las em um ambiente controlado e artificial, fazê-las no ambiente natural dos usuários onde a aplicação será utilizada de fato. Enquanto testes no laboratório permitem a identificação de problemas de interface e interação, avaliações no contexto do usuário normalmente são utilizadas principalmente para se facilitar a introdução de tecnologia ou avaliar o uso sendo feito desta no contexto do usuário (Bly, 1997).

Em avaliações no ambiente do usuário, normalmente a coleta de dados é feita através da observação do uso sendo feito da aplicação e conversas com os usuários. Os dados podem ser armazenados como anotações, gravações de áudio ou vídeo, ou uma combinação de formas. Podem ser armazenados também artefatos ou quaisquer indicadores que auxiliem no entendimento de como as pessoas trabalham no seu próprio ambiente. Normalmente o dado coletado é qualitativo e a análise feita sobre ele interpretativa.

Existem duas abordagens principais sobre o papel do avaliador quando atuando no contexto do usuário (). Na primeira abordagem, o avaliador atua apenas como **observador** e deve se esforçar para registrar os acontecimentos e **não interferir** no contexto. Na outra abordagem, o avaliador atua como **participante** do contexto e tem por objetivo explorar ao máximo os detalhes envolvidos no contexto e nas ações que sucedem nele. Neste caso abordagens etnográficas são apropriadas e bastante utilizadas.

Independente da abordagem de atuação do avaliador utilizada, com frequência o avaliador utiliza-se de esquemas de apoio à sua observação. O objetivo destes esquemas

---

<sup>7</sup> Em inglês *Field Studies*

é guiar a observação, e organizar os dados sendo coletados. Um exemplo de esquema é o proposto por Robson (1993):

- Espaço: Como é a disposição física do ambiente e como ele está organizado?
- Atores: Quais são os nomes e detalhes relevantes das pessoas envolvidas?
- Atividades: O que os atores estão fazendo e por quê?
- Objetos: Que objetos físicos, como por exemplo móveis, estão presentes?
- Atos: O que cada um dos atores está fazendo?
- Eventos: O que está sendo observado é parte de algum evento?
- Objetivos: Que objetivos os atores estão tentando atingir?
- Sentimentos: Qual o humor do grupo e dos indivíduos?

Uma vez feita a avaliação no ambiente do usuário recomenda-se rever as anotações e registros tão logo quanto possível (de preferência antes de passadas 24hs) para que os dados estejam frescos na memória e para que se possa explorar detalhes e descartar ambigüidades com outros observadores ou com as pessoas sendo observadas. Além disso, na preparação para a avaliação, aspectos sociais e de relacionamento com as pessoas sendo observadas devem ser levados em consideração, tais como de que forma ganhar a confiança das pessoas envolvidas na observação e como lidar com questões delicadas.

## 6.5. Considerações Finais

Neste capítulo foram apresentados alguns dos principais métodos de avaliação da qualidade de uso de usabilidade e comunicabilidade de interfaces, tanto analítico, quanto que envolvem usuários. Estes métodos foram desenvolvidos para avaliações de interfaces de sistemas mono-usuário de propósito geral. Todos os métodos propõem que o domínio da aplicação e o seu contexto de uso sejam considerados durante a execução da avaliação, seja pelos especialistas que inspecionam a interface, seja pelas tarefas a serem propostas aos usuários. No entanto, nenhum dos métodos se propõe a apreciar aspectos específicos relacionados ao domínio da aplicação.

Atualmente, com o barateamento da tecnologia e a sua adoção cada vez mais ampla no cotidiano das pessoas, alguns domínios carecem de métodos que possam avaliar interfaces não só sob a ótica da interação, mas também do objetivo do domínio propriamente dito. Pesquisas têm sido desenvolvidas para propor extensões em métodos existentes ou para apresentar novos métodos para sistemas de domínio específico, como por exemplo o de sistemas educacionais e ambientes de *groupware*. Além de domínios, novas tecnologias muitas vezes trazem consigo novos aspectos de interação que precisam também ser considerados na sua avaliação, como é o caso da *Web* e telefones celulares.

Além de domínios específicos trazerem a necessidade de novos métodos para se avaliar a usabilidade ou comunicabilidade de um sistema interativo, eles apontam para outros aspectos de qualidade de uso que são igualmente relevantes e importantes para o sucesso do sistema (). Nesta seção serão apresentadas brevemente extensões propostas a alguns dos métodos discutidos neste capítulo para os domínios de *groupware* e educação, e para a tecnologia *Web*. Em seguida serão descritas algumas outras

qualidades de uso que têm sido apontadas como relevantes para diferentes domínios. Finalmente, termina-se com uma discussão sobre o custo da avaliação no processo de design.

### **6.5.1 Outros ambientes a serem avaliados**

#### ***Groupware***

*Groupware* é um termo que se refere a sistemas que têm por objetivo apoiar o trabalho cooperativo de pessoas trabalhando em grupo. Os sistemas variam nas atividades, formas de trabalho, e objetivos aos quais oferecem apoio. Em sistemas multi-usuário os participantes devem interagir não apenas com o software, mas também com os demais participantes, através deste software. Desta forma, os aspectos considerados ao se fazer avaliação de interfaces de sistemas mono-usuário continuam sendo relevantes, mas no entanto não são suficientes para apreciar todas as dimensões de interação existentes nestas aplicações.

Atualmente, não existem métodos estabelecidos e reconhecidos para avaliação de interfaces de *groupware*. Pesquisadores têm apresentado propostas que visam estender os métodos reconhecidos para interfaces mono-usuário, para que eles incluam aspectos relevantes ao trabalho em grupo e comunicação entre membros. Descrevemos aqui brevemente algumas propostas que estendem os métodos discutidos neste capítulo.

As propostas apresentadas por métodos analíticos têm como principal motivação o fato de conseguirem ter um baixo custo. Baker e seus colegas (2001) propõem uma extensão à avaliação heurística de Nielsen para *groupware*. Para isso, eles apresentam 8 novas heurísticas que têm por objetivo guiar o especialista na avaliação do apoio à comunicação, colaboração e coordenação oferecida pelo sistema, assim como na representação das atividades do grupo na interface para cada um dos membros. Existe também uma proposta para se estender o método de percurso cognitivo para grupos (). Neste caso propõe-se que os avaliadores partam de um modelo de tarefas de grupo, construam cenários e a partir dos cenários avaliem as tarefas envolvidas no cenário e para cada uma delas faça um conjunto de perguntas que os possibilite identificar problemas na interface e nas atividades de colaboração. O método pode ser feito por um avaliador que se coloca no papel dos diversos membros do grupo, ou por diferentes avaliadores, cada um assumindo um papel.

Muitas vezes os ambientes são avaliados por observação de uso do grupo ou em avaliações no ambiente do grupo. No entanto, observar um grupo pode ser complexo pelo número de indivíduos a serem observados e as inúmeras variações entre grupos (Grudin, 1988). A proposta de extensão do método de avaliação de comunicabilidade (Prates e de Souza, 2002) tem por objetivo identificar os problemas de interação que podem impactar a comunicação, colaboração ou coordenação do grupo. Para isso, são propostas novas expressões que descrevam o problema da interação do grupo, além de se incluir no método a consideração sobre quem estaria “dizendo” a expressão para quem (e.g. um membro do grupo diria para o grupo “Quem está aqui?”). Foram propostos também novas categorias de problema que podem ocorrer na interação de um grupo.

## Sistemas educacionais

Ambientes educacionais têm por objetivo apoiar o ensino e aprendizado de um conteúdo. Os sistemas podem se destinar a oferecer material instrucional àqueles que estão fisicamente remotos, ou cujos horários disponíveis não se encaixam em horários determinados por estabelecimentos de ensino, ou ainda complementar o ensino sendo feito em sala de aula. A interface destes sistemas devem permitir não apenas a interação do usuário com o sistema, mas o aprendizado de um conteúdo. Desta forma, métodos de avaliação para este domínio devem permitir não apenas a apreciação de qualidades de uso da interface, mas também se ela consegue atingir com qualidade seus objetivos educacionais.

Alguns pesquisadores nesta área argumentam que para sistemas educacionais, a simples extensão de métodos de avaliação de IHC para este domínio não seriam suficientes (Squires, 1999), e novos métodos que levam em conta os objetivos educacionais precisariam ser propostos (Jones et al., 1999). Com o objetivo de integrar aspectos de usabilidade e aprendizado Preece and Squires (1999) revisam as heurísticas propostas por Nielsen, à luz da teoria educacional sócio-cognitivista. Como resultado eles propõem um novo conjunto de heurísticas para que educadores avaliem software educacional.

## Web

Com a disseminação da *Internet*, e posteriormente de portais corporativos *Intranet*, surgiram novos aspectos de interface e de interação que precisam ser avaliados. Estes sistemas são tipicamente multi-usuário, e podem ser assíncronos ou síncronos. Extensões aos métodos de avaliação existentes vêm sendo propostas para lidar com este tipo de tecnologia. Em geral, estas extensões não modificam os procedimentos de avaliação, mas sim o tipo de informação coletada e analisada.

Com relação aos métodos de avaliação analíticos, já foram propostas extensões tanto à avaliação heurística como ao percurso cognitivo. No caso de avaliações heurísticas, o conjunto de heurísticas a serem consideradas é estendido, como em Lynch & Palmiter (2002). Eles propõem que sejam incluídas heurísticas específicas do ambiente *Web* sobre, por exemplo: contexto, organização e estrutura; temas e objetivos de páginas e *subsites*; foco e distração; marca (*branding*) e identidade; mecanismos e estruturas de controle; navegação e busca; e *links* quebrados.

No caso do percurso cognitivo, foi proposto um novo método, chamado percurso cognitivo para a *Web* (), cujo objetivo principal é avaliar as atividades de navegação e busca por informação em um *website*. Este método visa estimar, de forma objetiva, o grau de semelhança entre descrições de objetivos do usuário e os textos de cabeçalhos e links de páginas do *website* avaliado. Para isto, utiliza técnicas de aquisição e representação de conhecimento com base em análise semântica latente.

Também foram propostas extensões aos métodos de avaliação que envolvem usuários. No caso de avaliação quantitativa, são acrescentados critérios como: profundidade na árvore de navegação, número de vezes em que o usuário volta à página anterior, número de *links* acionados a partir de uma mesma página, entre outros. Já no caso de avaliação qualitativa, pode-se citar uma proposta inspirada no método de

avaliação de comunicabilidade, aplicada fora do contexto da Engenharia Semiótica, e especificamente ao método de projeto hipermídia OOHDm. Este método visa associar padrões de comportamento durante a interação a problemas de projeto ().

### **6.5.2 Outras qualidades de uso a serem avaliadas**

Neste capítulo tratamos apenas de métodos de avaliação das qualidades de uso de usabilidade e comunicabilidade. No entanto, existem vários outros aspectos de qualidade de uso que também deveriam ser considerados ao se avaliar a qualidade de um sistema interativo. Assim como usabilidade e comunicabilidade, algumas destas qualidades se aplicam a qualquer software, como aplicabilidade (apresentada na seção 6.1.3) e acessibilidade, enquanto outras são específicas para determinados domínios, como aprendizado, colaboração, entretenimento e sociabilidade.

Atualmente um outro conceito de qualidade de uso com que desenvolvedores de sistemas interativos devem se preocupar é acessibilidade. Este conceito está relacionado com se possibilitar acesso ao sistema a indivíduos portadores de alguma deficiência física (WAI). Enquanto em alguns sistemas a acessibilidade é uma qualidade desejável, em outros ela é fundamental, como é o caso de sistemas do governo que oferecem aos cidadãos serviços disponíveis na *Web*. Em muitos países existem leis que garantem o acesso a cidadãos com deficiência a sistemas interativos públicos, fazendo com que governos coloquem a acessibilidade como uma exigência no desenvolvimento de seus sistemas (e.g. ver lei 508 dos EUA (Section 508, 1998)).

Outras qualidades têm se mostrado essenciais em determinados contextos. Em contextos como ambientes de trabalho colaborativo, não é suficiente que a interface de um sistema tenha boa usabilidade e comunicabilidade, deve-se avaliar também a qualidade da colaboração que membros alcançam através do sistema. Em sistemas multi-usuário que não tem por objetivo apoiar o trabalho de um grupo, mas sim as atividades de uma comunidade, uma qualidade de uso fundamental é a sociabilidade, ou seja, como o sistema apóia os objetivos de uma comunidade, e as influências e interações sociais que ocorrem nesta (Preece, 2000).

Como vimos, para sistemas educacionais é fundamental que se meça a capacidade do sistema de atingir seus objetivos educacionais e fomentar o aprendizado (). Da mesma forma, em sistemas de entretenimento (Westerink et al., 1994), como jogos, o sistema só terá sucesso se ele for capaz de proporcionar diversão aos usuários.

### **6.5.3 Custo de etapas de avaliação para o projeto**

Como dito no início deste capítulo, infelizmente há muitos gerentes de projeto que consideram alto o custo de se fazer avaliações da qualidade de uso dos sistemas interativos. No entanto, estudos demonstram que o retorno de investimento deste tipo de avaliação é alto (). Entre outros fatores, o estudo da MauroNewMedia revela que:

- Muito mais caro do que o dinheiro gasto para atrair um cliente, é o custo de convencê-lo a voltar por causa de baixa usabilidade ou serviço ruim ao cliente (em uma razão de 1:100).
- Gastos com a melhoria do design gráfico de um sistema trazem baixíssimo retorno de investimento. Entretanto, a melhoria do comportamento interativo,

principalmente no que diz respeito à busca e ao fornecimento de informação, traz um retorno de investimento da ordem de 1:50 ou até mesmo 1:100.

- A complexidade e conseqüentemente o custo do desenvolvimento de um sistema é reduzido quando problemas críticos de qualidade de uso são descobertos e solucionados no início do processo de desenvolvimento (em uma razão de 1:10).
- A qualidade de uso de um sistema determina o volume de chamadas à central de suporte, cujo custo operacional é muito alto.

Para mais informações sobre retorno de investimento de avaliação e projeto de sistemas interativos com alta qualidade de uso, sugere-se ao leitor visitar a coletânea de recursos indicados na página <http://www.rashmisinha.com/useroi.html> (última visita: maio de 2003).

Para diminuir os custos de avaliação durante o processo de design, muitas vezes utiliza-se avaliações informais e rápidas<sup>8</sup> (). Nestas avaliações, os designers têm um retorno de usuários ou consultores sobre suas idéias para o design. Este tipo de avaliação pode ser feito em qualquer etapa de design. Por exemplo, no início do design pode-se ter uma reunião com os usuários para se discutir as propostas de soluções, ou antes de se implementar o sistema, para se ter um retorno sobre decisões a serem implementadas, como a estrutura do menu em um sistema *Web*.

Muitas vezes utiliza-se para estas avaliações os métodos apresentados neste capítulo, mas se simplifica o método para se conseguir o retorno de forma mais rápida e barata. Por exemplo, ao invés de se executar a avaliação heurística com 5 especialistas, utiliza-se apenas 2, ou então se faz testes com uma quantidade menor de usuários, de forma mais informal e simplifica-se a etapa de análise. Este tipo de avaliação pode contribuir para o processo de design, mas de forma alguma substitui uma avaliação mais completa, que produz resultados mais abrangentes e confiáveis.

## Referências

- Adler, P. & Winograd, T. (1992) *Usability: Turning Technologies into Tools*. Oxford University Press. New York, NY, 1992.
- Asdi, K. and Daniels, H., (2000) “ ‘Learnability’ Testing in Learner-Centered Design”. In *CHI 2000 Extended Abstracts*.
- Baker, K., Greenberg, S. and Gutwin, C. (2001) “Heuristic Evaluation of Groupware Based on the Mechanics of Collaboration”. In M.R. Little and L. Nigay (Eds) *Engineering for Human-Computer Interaction (8th IFIP International Conference, EHCI 2001, Toronto, Canada, May)*, *Lecture Notes in Computer Science* Vol 2254, p123-139, Springer-Verlag.
- Barbosa, C. M. O., de Souza, C. S., Nicolaci-da-Costa, A. M., and Prates, R. O. P. (2002), “Using the Underlying Discourse Unveiling Method to Understand Organizations of Social Volunteers”, *Anais do V Simpósio sobre Fatores Humanos em Sistemas Computacionais (IHC 2002)*, Fortaleza, 15-26.

---

<sup>8</sup> Em inglês normalmente chamadas de avaliações “*Quick and Dirty*”.

- Prates, R. O., Barbosa, S. D. J. (2003) *Avaliação de Interfaces de Usuário - Conceitos e Métodos*. Jornada de Atualização em Informática, SBC. (Versão revista)
- Blackmon, M.H.; Polson, P.G.; Kitajima, M.; Lewis, C. (2002) "Cognitive Walkthrough for the Web". *ACM Proceedings of CHI 2002*. pp.463–469. Bly, S. (1997) "Field Work: Is it product work?" *Interactions*, Jan/Fev, 25-30.
- de Souza, C.S. (1993) "The Semiotic Engineering of User Interface Languages". *International Journal of Man-Machine Studies*, Vol.39, 1993, pp.753-773.
- de Souza, C.S.; Prates, R.O.; and Barbosa, S. D. J. (1999) "A Method for Evaluating Software Communicability". *Anais do II Workshop sobre Fatores Humanos em Sistemas Computacionais (IHC'1999)*. Campinas, Artigo 28.
- de Souza, C. S., Barbosa, S. D. J., Prates, R. O. (2001) "A Semiotic Engineering Approach to User Interface Design". *Journal of Knowledge-Based Systems*, Vol.14, Issue 8, 2001, pp 461-465.
- Dumas, J. S e Redish, J. C. (1999) *A Practical Guide to Usability Testing* (Revised Edition). Exeter, UK: Intellect, 1999.
- Ericsson, K. A. and Simon, H. A. (1985) *Protocol Analysis: Verbal Reports as Data*. Cambridge, MA: MIT Press.
- Fischer, G. (1998) "Beyond 'Couch Potatoes': From Consumers to Designers" In *Proceedings of the 5<sup>th</sup> Asia Pacific Computer-Human Interaction Conference*. IEEE Computer Society, 2-9, 1998.
- Güell, N.; Schwabe, D.; Barbosa, S.D.J. (2001) "Método de Avaliação de Usabilidade na Web Baseado em Modelo e Padrões de Comportamento". *Proceedings of the 7th Brazilian Symposium on Multimedia and Hypermedia Systems, SBMIDIA 2001*. Florianópolis, SC. Outubro de 2001. pp.15–36. Grudin, J. (1988) "Why CSCW Applications Fail: Problems in the Design and Evaluation of Organizational Interfaces". *Proceedings of ACM CSCW'88*, 85-93.
- Hartson, H.R. (1998) "Human-Computer Interaction: Interdisciplinary roots and trends". In *The Journal of System and Software*, **43**, 103-118. 1998.
- Hewett, T.; Baecker, R.; Card, S.; Carey, T.; Gasen, J.; Mantei, M.; Perlman, G.; Strong, G.; Verplank, W. (1992) "ACM SIGCHI Curricula for Human-Computer Interaction". *ACM SIGCHI Report*, ACM, NY. Disponível online em <http://sigchi.org/cdg/> [último acesso: maio de 2003].
- Jones, A., Scanlon, E. , Tosunoglu, C. , Morris, E., Ross, S., Butcher, P. e Greenberg, J. (1999) "Contexts for evaluating educational software". *Interacting with Computers*. Vol. 11 (1999) 499-516.
- Karat, J. (1993) The cost-benefit and business case analysis of usability engineering. *InterChi '93*, Amsterdam, Tutorial Notes 23.
- Lynch, G. and Palmiter, S. (2002) "Design and Rapid Evaluation of Usable Web Sites", *CHI2002 tutorial notes*.
- MauroNewMedia (2002) "Professional Usability Testing and return on investment as it applies to user interface design for web-based products and services". Disponível online em [http://www.taskz.com/ucd\\_testing\\_roi\\_summary.php](http://www.taskz.com/ucd_testing_roi_summary.php) [último acesso: maio de 2003]

- Prates, R. O., Barbosa, S. D. J. (2003) *Avaliação de Interfaces de Usuário - Conceitos e Métodos*. Jornada de Atualização em Informática, SBC. (Versão revista)
- Moran, T. (1981) "The Command Language Grammars: a representation for the user interface of interactive computer systems. Em *International Journal of Man-Machine Studies* **15**:3-50, Academic Press.
- Mack, R. & Nielsen, J. (1994) *Usability Inspection Methods*. New York, NY: John Wiley & Sons.
- Nicolaci-da-Costa, A. M. (2001), "Gerando conhecimento sobre os homens, mulheres e crianças que usam computadores: algumas contribuições da psicologia clínica", Anais do IV Workshop sobre Fatores Humanos em Sistemas Computacionais (IHC'2001). Florianópolis, 120-131.
- Nielsen, J. (1993) *Usability Engineering*. Academic Press.
- Nielsen, J. (1994) "Heuristic Evaluation", in Mack, R. & Nielsen, J. (eds.) *Usability Inspection Methods*. New York, NY: John Wiley & Sons, 1994, 25-62. Nielsen, J. (1998) "Cost of User Testing a Website", Alertbox. Disponível online em <http://www.useit.com/alertbox/980503.html> [último acesso: maio/2003].
- Nielsen, J. (2000) "Test with 5 Users", Alertbox. Disponível online em <http://www.useit.com/alertbox/20000319.html> [último acesso: maio/2003].
- Norman, D. (1988) *Psychology of Everyday Things*. BasicBooks. HarperCollins Publishers, 1988.
- Oré (2001) Site do Projeto Oré. Disponível em: <http://www.serg.inf.puc-rio.br/ore>.
- Peirce, C.S. (1931-1958). *Collected Papers*. Edição brasileira: *Semiótica*. São Paulo, Ed. Perspectiva (coleção estudo, n.46), 1977.
- Pinelle, D. and Gutwin, C. (2002) "Groupware Walkthrough: Adding Context to Groupware Usability Evaluation". *Proceedings of the CHI 2002*.
- Prates, R. O.; de Souza, C. S; Carey, T.; Muller, M. J. (2001) "Evaluating beyond usability evaluation". Presented as SIG at CHI 2001, Seattle, USA. Available at (<http://www.serg.inf.puc-rio.br>)
- Prates, R.O.; Barbosa, S.D.J.; de Souza, C.S. (2000a) "A Case Study for Evaluating Interface Design Through Communicability." *Proceedings of the International Conference on Designing Interactive Systems, DIS2000*. New York, NY: ACM Press, 308-317, 2000.
- Prates, R.O.; de Souza, C.S.; Barbosa, S.D.J.; (2000b) "A Method for Evaluating the Communicability of User Interfaces." *Interactions* **7**, 1. New York, NY: ACM Press, 31-38, 2000.
- Prates, R.O.; de Souza, C.S. (2002) "Extensão do Teste de Comunicabilidade para Aplicações Multi-usuário". *Cadernos do IME*, Volume 13. 46-56, 2002.
- Preece, J. (2000) *Online Communities*. NY, NY: John Wiley & Sons. 2000. Preece, 2000
- Preece, J.; Rogers, Y.; Sharp, E. (2002) *Interaction Design: Beyond Human-computer Interaction*. New York, NY: John Wiley & Sons. 2002.



- Prates, R. O., Barbosa, S. D. J. (2003) *Avaliação de Interfaces de Usuário - Conceitos e Métodos*. Jornada de Atualização em Informática, SBC. (Versão revista)
- Preece, J.; Rogers, Y.; Sharp, E.; Benyon, D.; Holland, S.; Carey, T. (1994) *Human-Computer Interaction*. England: Addison-Wesley, 1994. Robson, C. (1993) *Real World Research*. Oxford, UK: Blackwell.
- Section 508 (1998). Disponível online em <http://www.section508.gov/> [último acesso: maio/2003].
- Shneiderman, B. (1998) *Designing the User Interface, 3<sup>rd</sup> Edition*. Reading, MA: Addison Wesley, 1998. Squires, D. (1999) "Usability and Educational Software Design: Special Issue of Interacting with Computers." *Interacting with Computers*. Vol. 11 (1999) 463–466.
- Squires, D. e Preece, J. (1999). Predicting quality in educational software: "Evaluating for learning, usability and the synergy between them". *Interacting with Computers*. Vol. 11 (1999) 467–483.
- Web Accessibility Initiative (WAI). Disponível online em <http://www.w3.org/WAI/> [último acesso: maio/2003].
- Wharton, C., Rieman, J., Lewis, C. and Polson, P. (1994) "The Cognitive Walkthrough Method: A Practitioner's Guide." In Nielsen, J., and Mack, R.L. (Eds.), *Usability Inspection Methods*, John Wiley & Sons, New York, NY.
- Westerink, J. H. D. M., Rankin, P. J., Majoor, G. M. M., Moore, P. S. (1994) "A New Technique for Early User Evaluation of Entertainment Product Interfaces", *Proceedings of the Human Factors and Ergonomics Society 38th Annual Meeting* 1994 v.2 p.992