

## O que é *Design* de Interação?

- 1.1 Introdução
- 1.2 Um bom e um mau *design*
  - 1.2.1 O que projetar
- 1.3 O que é *design* de interação?
  - 1.3.1 Breve história do *design* de interação
  - 1.3.2 Trabalhando juntos como uma equipe multidisciplinar
  - 1.3.3 *Design* de interação e negócios
- 1.4 O que está envolvido no processo de *design* de interação?
- 1.5 As metas do *design* de interação
  - 1.5.1 Metas de usabilidade
  - 1.5.2 Metas decorrentes da experiência do usuário
- 1.6 Um pouco mais sobre usabilidade: princípios de usabilidade e *design*
  - 1.6.1 Heurística e princípios de usabilidade

### 1.1 Introdução

Quantos produtos interativos existem em nossa vida cotidiana? Pense um minuto sobre o que você utiliza em um dia normal: telefone celular, computador, agenda eletrônica, controle remoto, máquina de refrigerantes, cafeteira, caixa eletrônico, guichê eletrônico de venda de passagens, sistema informatizado de biblioteca, *web*, fotocopadora, relógio, impressora, rádio, calculadora, *videogame*... a lista é interminável. Agora pense por um minuto sobre usabilidade. Quantos deles são realmente fáceis e agradáveis de utilizar? Todos, alguns, ou apenas um ou dois? É provável que agora a lista se mostre consideravelmente menor. Por quê?

Pense em alguma ocasião em que um desses dispositivos tenha lhe causado algum incômodo – quanto tempo você gastou tentando fazê-lo funcionar? Dois dispositivos interativos e bem conhecidos que causam muito aborrecimento às pessoas são a fotocopadora que não copia da maneira que se espera e o videocassete que não grava o programa que queremos, ou nem mesmo grava. Por que você acha que essas coisas acontecem? E além disso, podemos fazer alguma coisa a esse respeito?

Muitos produtos que requerem a interação dos usuários para a realização de suas tarefas (p. ex.: comprar um ingresso pela Internet, fotocopiar um artigo, gravar um programa de TV) não foram necessariamente projetados tendo o usuário em mente; foram tipicamente projetados como sistemas para realizar determinadas funções. Pode ser que funcionem de maneira eficaz, olhando-se da perspectiva da engenharia, mas geralmente os usuários do mundo real é que são sacrificados. O objetivo do *design* de interação consiste em redirecionar essa preocupação, trazendo a usabilidade para dentro do processo de *design*. Essencialmente, isso significa desenvolver produtos interativos<sup>1</sup> que sejam fáceis, agradáveis de utilizar e eficazes – sempre na perspectiva do usuário.

Neste capítulo, iniciamos avaliando o que é o *design* de interação. Olhamos para a diferença entre um bom e um mau *design*, ressaltando como os produtos podem ser radicalmente diferentes com respeito à sua usabilidade. Descrevemos o que e quem está envolvido no *design* de interação. Na última parte do capítulo, delineamos aspectos centrais referentes à usabilidade e como são utilizados para a avaliação de produtos interativos. Uma tarefa é apresentada ao final do capítulo. Nela você terá a oportunidade de colocar em prática o que leu, avaliando um produto interativo a partir de critérios de usabilidade.

Os principais objetivos deste capítulo são os seguintes:

- Explicar as diferenças entre os bons e os maus *designs* de interação.
- Descrever o que é o *design* de interação e como ele se relaciona à interação homem-computador e a outros campos.
- Explicar o que é usabilidade.
- Descrever o que está envolvido no processo de *design* de interação.
- Delinear as diferentes formas de orientação utilizadas no *design* de interação.
- Permitir que você avalie um produto interativo e explique o que é bom e o que é ruim, no que diz respeito a objetivos e princípios de *design* de interação.

## 1.2 Um bom e um mau *design*

Uma preocupação central do *design* de interação é desenvolver produtos interativos que sejam utilizáveis, o que genericamente significa produtos fáceis de aprender, eficazes no uso, que proporcionem ao usuário uma experiência agradável. Um bom ponto de partida para pensar sobre como projetar produtos interativos utilizáveis consiste em comparar bons e maus exemplos. Mediante a identificação de pontos fracos e fortes específicos de sistemas interativos diferentes, podemos começar a entender o que quer dizer algo ser usável ou não. Aqui, iniciamos com o exemplo de um sistema mal projetado – o correio de voz (*voice mail*) —, utilizado em muitas organizações (empresas, hotéis e universidades). Realizamos a comparação desse aparelho com uma secretária eletrônica, que seria o exemplo de um bom *design*.

Imagine o seguinte cenário. Você está hospedado em um hotel por uma semana, em uma viagem de negócios. Você então descobre que deixou o telefone celular em casa e que tem de contar com os serviços que o estabelecimento oferece. O hotel dispõe de um sistema de correio de voz para cada quarto. A fim de verificar se há algu-

<sup>1</sup> Utilizamos o termo *produtos interativos* genericamente para nos referirmos a todas as classes de sistemas, tecnologias, ambientes, ferramentas, aplicações e aparelhos interativos.

ma mensagem, você pega o fone e ouve o sinal. Se ele faz “bip bip bip”, há mensagem. Para descobrir como acessá-la, você tem de ler uma série de instruções que estão próximas ao aparelho.

Você então lê e segue o primeiro passo:

“1. Disque 491”.

O sistema responde: “Você discou para o centro de mensagens do Hotel Sunny. Por favor, digite o número do quarto para o qual você deseja deixar uma mensagem”.

Você espera para saber como ouvir uma mensagem gravada. Não há, porém, outras instruções. Você então olha novamente para a folha de instruções e lê:

“2. Pressione \*, o número do quarto e #”. Você o faz, e o sistema responde:

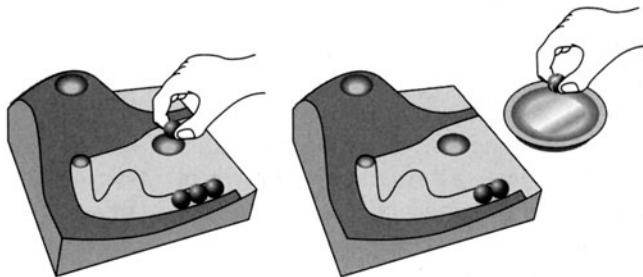
“Você discou para a caixa de mensagens do quarto 106. Para deixar uma mensagem, digite sua senha.”

Você não sabe qual é a senha. Pensou que fosse o número do quarto. Não é. A esta altura, você desiste e liga para a recepção. O funcionário explica o procedimento correto para gravar e ouvir mensagens; consiste em digitar, na hora certa, o número do quarto e o número da extensão do telefone (que é a sua senha, e não o número do quarto). Além disso, são seis os passos para acessar uma mensagem e cinco para deixar uma mensagem. Você resolve sair e comprar um celular novo.

Qual é o problema desse sistema de correio de voz?

- É irritante.
- É confuso.
- É ineficiente; exige muitos passos para tarefas básicas.
- É difícil de utilizar.
- Não permite saber rapidamente se há alguma mensagem ou quantas mensagens há. É necessário pegar o fone e realizar toda uma série de passos para ouvi-las.
- Os procedimentos não são óbvios: as instruções são fornecidas parcialmente pelo sistema e parcialmente pela folha ao lado do telefone.

Vamos agora pensar em uma secretária eletrônica. A Figura 1.1 mostra dois pequenos esquemas representando uma secretária eletrônica. As mensagens que chegam são representadas pelas esferas – as que estão enfileiradas indicam o número de mensagens. Ao colocar uma dessas esferas no orifício presente na máquina, aciona-se a mensagem gravada. Ao colocar a mesma esfera em outro orifício, disca-se para quem deixou a mensagem.



**Figura 1.1** Dois pequenos esquemas representando uma secretária eletrônica.

Como a secretária do esquema se diferencia do sistema de correio de voz?

- Utiliza objetos físicos familiares, que indicam visualmente e de forma rápida quantas mensagens foram deixadas.
- É esteticamente fácil e agradável de utilizar.
- Requer ações de apenas um passo para a realização de tarefas importantes.
- Apresenta um *design* simples mas elegante.
- Oferece menos funcionalidade e permite a qualquer um ouvir as mensagens.

Essa secretária eletrônica de esferas foi projetada por Durrel Bishop quando era estudante no Royal College of Art de Londres (descrito por Crampton-Smith em 1995). Um dos objetivos era projetar um sistema de mensagens que tivesse sua funcionalidade básica representada pelo comportamento de objetos comuns. Para tanto, ele aproveitou o conhecimento que as pessoas têm sobre o funcionamento do mundo físico em seu dia-a-dia. Em especial, fez uso de uma ação muito comum da vida cotidiana que consiste em pegar um objeto físico e colocá-lo em outro lugar. Eis aí um exemplo de produto interativo projetado tendo o usuário em mente. O foco está em proporcionar uma experiência agradável que, não obstante, torne também a atividade de receber mensagens eficiente. Contudo, é importante destacar que, embora a secretária eletrônica de esferas apresente um *design* elegante e usável, ela não seria prática em um ambiente de hotel. Uma das razões principais refere-se ao fato de não apresentar resistência suficiente para ser utilizada em locais públicos. As esferas, por exemplo, poderiam ser perdidas ou levadas como *souvenir*. A necessidade de identificar o usuário antes de permitir que as mensagens sejam ouvidas é fundamental em um hotel. Ao se considerar a usabilidade de um *design*, portanto, é fundamental levar em consideração *onde* ele será utilizado e *por quem*. A secretária eletrônica aqui apresentada seria mais adequada a uma casa – contanto que não houvesse crianças que pudessem sentir-se tentadas a brincar com as bolinhas!

### 1.2.1 O que projetar

Projetar produtos interativos usáveis requer que se leve em conta quem irá utilizá-los e onde serão utilizados. Outra preocupação importante consiste em entender o tipo de *atividades* que as pessoas estão realizando quando estão *interagindo* com os produtos. Aquilo que for mais apropriado para os diferentes tipos de interfaces e para o planejamento de dispositivos de entrada e saída de informação vai depender de que tipos de atividades receberão suporte. Por exemplo, se a atividade consiste em deixar as pessoas comunicarem-se à distância, um sistema que permita fácil recebimento de mensagens (faladas ou escritas) que possam ser prontamente acessadas pelo destinatário é mais apropriado. Além disso, uma interface que permitisse aos usuários interagir com as mensagens (p. ex.: editar, fazer anotações, armazenar) seria bastante útil.

A gama de atividades passível de receber suporte é bastante diversa. Pense por um minuto no que você pode fazer atualmente utilizando sistemas baseados em computador: enviar mensagens, buscar informações, escrever artigos, controlar uma usina elétrica, programar, desenhar, planejar, calcular, jogar – para citar apenas algumas atividades. Agora pense no número de interfaces e dispositivos interativos que estão disponíveis. São igualmente diversos: aplicações multimídia, ambientes de realidade virtual, sistemas baseados na fala, assistentes digitais pessoais (PDAs) e gran-

des *displays* – para citar apenas alguns exemplos. Existem também muitas maneiras de projetar o modo como os usuários podem interagir com um sistema (p. ex.: por meio de menus, comandos, formulários, ícones, etc.). Além disso, estão surgindo formas de interação cada vez mais inovadoras que combinam dispositivos físicos com computação embarcada, como tinta eletrônica, brinquedos interativos, geladeiras inteligentes e roupas que permitem conectar-se à rede (veja Figura 1.2, na Página de Ilustrações 1). Isso tudo representa uma multiplicidade de escolhas e decisões com as quais o *designer* vai se deparar quando estiver desenvolvendo produtos interativos.

Uma questão fundamental para o *design* de interação é a seguinte: como você otimiza as interações do usuário com o sistema, ambiente ou produto, de forma que combinem com as atividades que estão sendo estendidas ou recebendo suporte? Poder-se-ia utilizar a intuição e esperar pelo melhor. Ou, então, criteriosamente decidir que escolhas fazer, baseando-as em uma maior compreensão acerca dos usuários. Tal processo envolve o seguinte, a saber:

- Considerar no que as pessoas são boas ou não
- Considerar o que pode auxiliar as pessoas na sua atual maneira de fazer as coisas
- Pensar no que pode proporcionar experiências de qualidade ao usuário
- Ouvir o que as pessoas querem e envolvê-las no *design*
- Utilizar técnicas baseadas no usuário “testadas e aprovadas” durante o processo de *design*

O objetivo deste livro é abordar esses aspectos, com o intuito de ensinar como realizar o *design* de interação; mais especificamente, como identificar as necessidades do usuário e, a partir desse entendimento, projetar sistemas usáveis, úteis e agradáveis.

#### ATIVIDADE 1.1 Qual é a diferença entre realizar uma chamada telefônica

- de um telefone público
- de um celular?

Como esses dispositivos foram desenvolvidos com relação a (a) tipo de usuários, (b) tipo de atividade a receber suporte e (c) contexto de uso?

#### Comentário

- (a) Os telefones públicos são projetados para ser utilizados pelo público em geral. Muitos dispõem de código em braille gravado nas teclas e de controle de volume para permitir que deficientes visuais e auditivos os utilizem.

Os telefones celulares são projetados para todos os grupos de usuários, embora possam ser de difícil uso para deficientes visuais ou pessoas com destreza manual limitada.

- (b) A maioria das cabines telefônicas são projetadas com um modo de interação simples: inserir cartão ou dinheiro e digitar o número do telefone. Se estiver ocupado ou não completar a ligação, o dinheiro ou cartão é devolvido quando o fone é colocado no gancho. Há também a opção de permitir ao usuário fazer chamadas seguiu pressionando um botão, em vez de retirar e recolocar o dinheiro a cada vez. Essa função permite realizar várias chamadas de uma maneira mais eficiente.

Os telefones celulares possuem um modo de interação mais complexo. Mais funcionalidades são oferecidas, exigindo que os usuários passem algum tempo aprendendo como utilizá-las. Por exemplo, poderão salvar números de telefone em uma agenda de endereços e vinculá-los a teclas de atalho; dessa forma, os nú-

meros selecionados poderão ser acessados pressionando-se simplesmente uma ou duas teclas.

- (c) Telefones públicos são destinados a uso em lugares abertos, na rua ou em estações de ônibus. São projetados para proporcionar ao usuário uma certa privacidade e proteção contra ruídos durante as ligações, por meio do uso de orelhões ou das cabines.

Os telefones celulares foram projetados para ser utilizados a qualquer hora e em qualquer lugar. No entanto, o fato de essa flexibilidade afetar as outras pessoas que estão no mesmo local público (trens e ônibus) não foi muito levado em consideração.

---

## 1.3 O que é *design* de interação?

Por *design* de interação, entendemos o seguinte:

*Design de produtos interativos que fornecem suporte às atividades cotidianas das pessoas, seja no lar ou no trabalho.*

Especificamente, significa criar experiências que melhorem e estendam a maneira como as pessoas trabalham, se comunicam e interagem. Winograd (1997) descreve o *design* de interação como “o projeto de espaços para comunicação e interação humana”. Nesse sentido, consiste em encontrar maneiras de fornecer suporte às pessoas. Tal projeto contrasta com o da engenharia de *software*, que enfoca principalmente a produção de soluções de *software* para certas aplicações. Podemos fazer uma outra analogia simples com outra profissão, para tentar explicar melhor essa distinção.

Ao descrever o *design* de interação, Terry Winograd questiona em que medida arquitetos e engenheiros civis se diferenciam ao deparar-se com o problema de construir uma casa. Os arquitetos estão preocupados com as pessoas, suas interações e o interior da casa. Por exemplo, existe a proporção certa entre áreas privadas e sociais? As áreas para cozinhar e fazer as refeições estão próximas? As pessoas utilizarão os espaços projetados da maneira como foram pensados? Em contrapartida, os engenheiros estão interessados em questões relacionadas com a realização do projeto, o que inclui aspectos práticos, como custo, durabilidade, aspectos estruturais, aspectos ambientais, regulamentações contra incêndio e métodos de construção. Assim como há uma diferença entre projetar e construir uma casa, há também uma diferença entre o *design* de interação e a engenharia de *software*. Resumindo, aquele está para esta como a arquitetura está para a engenharia civil.

### 1.3.1 Breve história do *design* de interação

Sempre se soube que para o sucesso do *design* de interação muitas disciplinas precisariam estar envolvidas. A importância de se entender como os usuários agem e reagem a situações e como se comunicam e interagem acarretou o envolvimento de pessoas de disciplinas diferentes, como psicólogos e sociólogos, em questões referentes ao *design* de interação. Da mesma forma, a crescente importância de se entender como projetar diferentes tipos de mídias interativas de maneira eficaz e prazerosa envolveu também uma diversidade de outros profissionais, incluindo *designers* gráficos, artistas, animadores, fotógrafos, especialistas de cinema e *designers* de produtos. A seguir, apresentamos uma breve história acerca do *design* de interação.

A princípio, os engenheiros projetavam sistemas de *hardware* para eles próprios utilizarem. A interface do computador era relativamente direta, reunindo vários painéis com chaves e mostradores que controlavam um conjunto de registros internos.

Com o advento dos monitores (hoje conhecidos como VDUs – *visual display units*) e de estações de trabalho pessoais, no final dos anos 70 e início dos 80, o *design* da interface passou a existir (Grudin, 1990). O novo conceito de interface com o usuário apresentava muitos desafios:

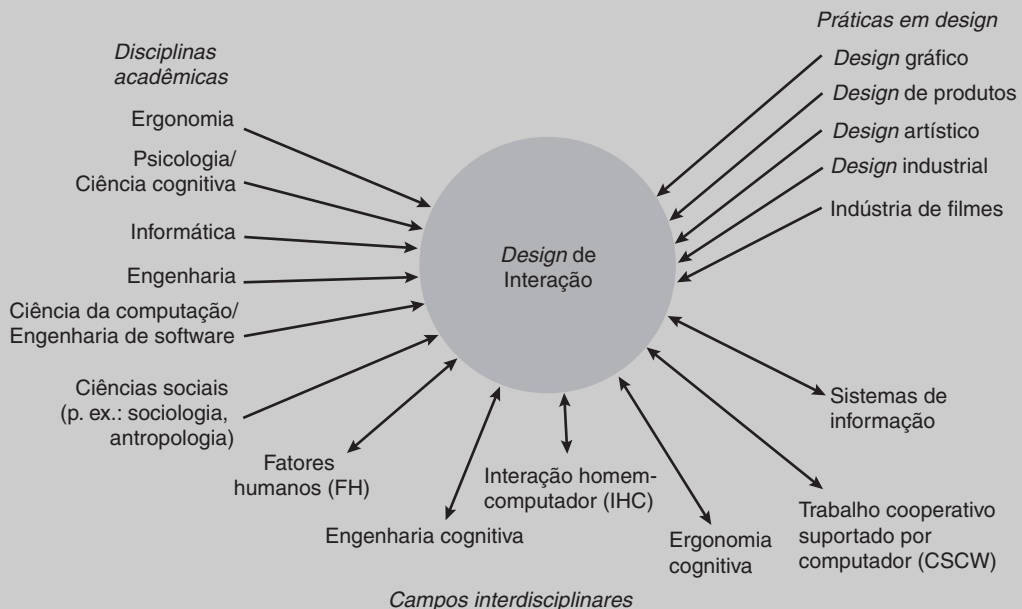
*É um horror. Você tem que se confrontar com a documentação. Tem que aprender uma nova linguagem. Você já havia utilizado a palavra “interface” antes de começar a usar um computador?*

Arthur Einstein – executivo de propaganda

#### QUADRO 1.1 A relação entre *design* de interação, interação homem-computador e outras abordagens

Entendemos o *design* de interação como fundamental para todas as disciplinas, campos e abordagens que se preocupam com pesquisar e projetar sistemas baseados em computador para pessoas (veja Figura 1.3). O campo interdisciplinar mais conhecido é a interação homem-computador (IHC), que se preocupa com “o *design*, a avaliação e a implementação de sistemas computacionais interativos para uso humano e com o estudo de fenômenos importantes que os rodeiam” (ACM SIGCHI, 1992, p. 6). Até o início dos anos 90, o foco da IHC era projetar interfaces para um único usuário. Em resposta a uma crescente preocupação com a necessidade de se fornecer suporte a múltiplos indi-

víduos que estejam trabalhando juntos e utilizando sistemas de computador, surgiu então o campo interdisciplinar de trabalho cooperativo suportado por computador (CSCW – em inglês, *computer-supported cooperative work*) (Grief, 1988). Os sistemas de informação constituem uma outra área preocupada com a aplicação de tecnologia de computação em domínios como negócios, saúde e educação. Outros campos relacionados ao *design* de interação incluem fatores humanos, ergonomia cognitiva e engenharia cognitiva – todos preocupados com projetar sistemas que vão ao encontro dos objetivos dos usuários, ainda que cada um com o seu foco e a sua metodologia.



**Figura 1.3** Relação entre disciplinas acadêmicas, práticas de *design* e campos interdisciplinares que se preocupam com o *design* de interação.

Um dos maiores desafios naquele tempo era desenvolver computadores que pudessem ser acessíveis e utilizáveis por outras pessoas, além dos engenheiros, para a realização de tarefas que envolvessem a cognição humana (p. ex.: fazer resumos, escrever documentos, gerenciar contas bancárias, esboçar planos). Para tornar isso possível, cientistas da computação e psicólogos envolveram-se com o *design* de interfaces de usuário. Os cientistas e engenheiros de *software* desenvolveram linguagens de programação de alto nível (p. ex.: BASIC, Prolog), arquiteturas de sistemas, métodos de desenvolvimento de *software* e linguagens baseadas em comando (*command-based languages*) para auxiliar tais tarefas, enquanto os psicólogos forneciam informações a respeito das capacidades humanas (p. ex.: memória, decisão).

Os recursos oferecidos pela tecnologia da computação interativa daquele tempo (isto é, o uso combinado de painéis visuais e teclados interativos) fizeram surgir novos desafios. A pesquisa e o desenvolvimento de interfaces gráficas (abreviadas GUI, do inglês *Graphical User Interface*, pronunciado “guu-ii”) para sistemas de automação de escritórios cresceram enormemente. Havia muitas pesquisas sobre o *design* de produtos (p. ex.: menus, janelas, paletas, ícones), no que diz respeito à melhor forma de estruturá-los e apresentá-los em uma GUI.

Em meados dos anos 80, a nova onda de tecnologias da computação – incluindo reconhecimento de voz, multimídia, visualização da informação e realidade virtual – apresentou ainda mais oportunidades de *design* de aplicações para fornecer ainda mais suporte às pessoas. Educação e treinamento foram duas áreas que receberam muita atenção. Ambientes de aprendizagem interativos, *softwares* educacionais e simuladores para treinamento foram alguns dos principais resultados. No entanto, construir esse novo tipo de sistemas interativos exigia um tipo diferente de conhecimento especializado, vindo de psicólogos e programadores. Especialistas em tecnologia educacional, psicólogos do desenvolvimento e especialistas em treinamento envolveram-se no empreendimento.

Como surgiram novas ondas de desenvolvimento tecnológico nos anos 90 – redes, computação móvel e sensores infravermelhos —, a criação de uma diversidade de aplicativos para *todas* as pessoas tornou-se uma possibilidade real. Todos os aspectos da vida de um indivíduo – em casa, em movimento, no lazer e no trabalho, sozinho, com a família ou os amigos – começaram a ser vistos como áreas que podiam ser melhoradas e estendidas projetando-se e integrando várias combinações de tecnologias computacionais. Novas formas de aprender, comunicar, trabalhar, descobrir e viver começaram a ser pensadas.

Em meados dos anos 90, muitas empresas perceberam que era necessário expandir novamente suas equipes multidisciplinares de *design*, para que as mesmas incluíssem profissionais treinados em mídia e *design*, como *design* gráfico, industrial, produção de filmes e desenvolvimento de narrativas. Sociólogos, antropólogos e dramaturgos foram incorporados ao quadro das equipes, todos com uma atitude diferente da dos psicólogos quanto à interação humana. Esse conjunto de pessoas promoveria a combinação certa de habilidades com a compreensão das diferentes áreas de aplicação necessárias para projetar a nova geração de sistemas interativos. Por exemplo, para projetar um aplicativo de recados para uma família é necessário saber como seus membros interagem; criar um *kit* interativo de histórias para crianças exige que se saiba como elas escrevem e entendem uma narrativa; desenvolver um guia interativo para visitantes de uma galeria de arte requer que se avalie o que as pessoas fazem e como se movimentam em espaços públicos.

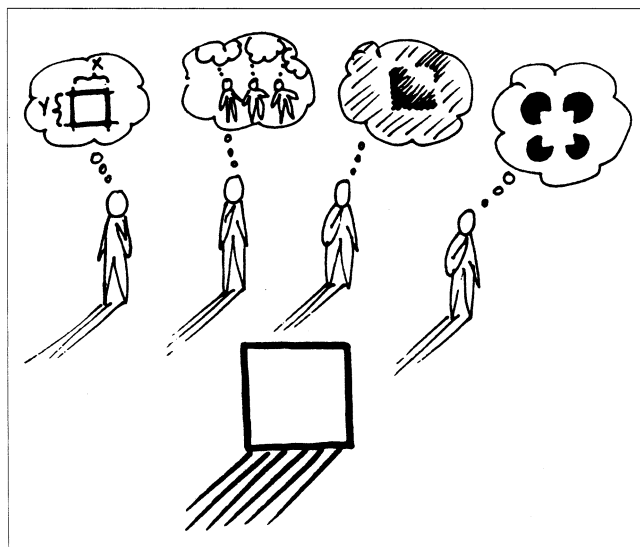


Hoje, nos anos 2000, as possibilidades proporcionadas pelas capacidades emergentes de *hardware* (p. ex.: etiquetas de radiofrequência, grandes telas interativas e *information appliances*) nos levam à conclusão de que os engenheiros, que conhecem *hardware*, *software* e eletrônica, devem estar preparados para configurar, montar e programar eletrodomésticos e outros dispositivos de maneira que se comuniquem entre si (o que geralmente é chamado de *middleware*).

### 1.3.2 Trabalhando juntos como uma equipe multidisciplinar

Reunir tantas pessoas com formações e treinamento diferentes significa muito mais idéias sendo geradas, novos métodos sendo desenvolvidos e *designs* mais criativos e originais sendo produzidos. No entanto, o fator negativo diz respeito aos custos envolvidos. Quanto mais pessoas com formações diferentes em uma equipe de *design*, mais difícil pode ser fazê-las comunicarem-se e fazer avançar os projetos desenvolvidos. Por quê? Pessoas com formações diferentes apresentam perspectivas e maneiras diferentes de ver e falar sobre o mundo (veja Figura 1.4). Algo que uma pessoa considere importante outras podem nem enxergar (Kim, 1990). Da mesma forma, a maneira como um cientista da computação entende o termo *representação* é geralmente muito diferente da do *designer* gráfico ou da de um psicólogo.

O que isso significa na prática é que confusão, desentendimento e falhas de comunicação podem freqüentemente surgir em uma equipe. Seus vários membros podem apresentar maneiras diferentes de falar sobre *design*, assim como utilizar os mesmos termos para se referir a coisas diferentes. Outros problemas podem também surgir quando pessoas em um grupo são colocadas juntas, sem nunca antes terem trabalhado como um time. Por exemplo, as equipes multidisciplinares do projeto Philips Vision of the Future, responsáveis pelo desenvolvimento de idéias e produtos para o futuro, passaram por uma série de dificuldades, quais sejam: os membros da equipe



**Figura 1.4** Quatro membros diferentes de uma mesma equipe olhando para o mesmo quadrado, mas cada um enxergando-o de maneira diferente.

nem sempre tinham uma idéia clara de quem necessitava de qual informação, quando e de que forma (Lambourne *et al.*, 1997).

**ATIVIDADE 1.2** Na prática, a formação de uma equipe de *design* depende do tipo de produto interativo que está sendo construído. Quem você acha que deveria estar envolvido no desenvolvimento de:

- (a) um quiosque público de informações sobre as exposições disponíveis em um museu de ciências?
- (b) um *website* educacional interativo que acompanha uma série de TV?

**Comentário** Cada equipe precisará de um número diferente de pessoas com diferentes habilidades. Por exemplo, o primeiro produto precisaria de:

- (a) *designers* gráficos e de interação, curadores de museus, conselheiros educacionais, engenheiros de *software*, *designers* de *software*, engenheiros de usabilidade, ergonomistas.

O segundo projeto precisaria de:

- (b) produtores de TV, *designers* gráficos e de interação, professores, especialistas em vídeo, engenheiros de *software*, *designers* de *software*, engenheiros de usabilidade.

Além disso, como os dois sistemas estão sendo desenvolvidos para o uso do público em geral, usuários representativos, como crianças e pais, devem estar envolvidos.

Na prática, as equipes de *design* geralmente acabam sendo bastante grandes, especialmente se estiverem trabalhando em um projeto importante, com prazo fixo. Por exemplo, é comum encontrar equipes de 15 ou mais pessoas trabalhando no projeto de um *website* por um longo período de tempo, como seis meses. Isso significa que provavelmente existirão várias pessoas de cada área trabalhando na equipe do projeto.

---

### 1.3.3 Design de interação e negócios

O *design* de interação é agora um grande negócio. Consultores para *websites*, empresas iniciantes e indústrias de computação móvel perceberam seu papel central em produtos interativos de sucesso. Para ser notado no campo de produtos para a *web*, que é altamente competitivo, é preciso destacar-se. Poder dizer que seu produto é eficaz e fácil de utilizar representa um fator fundamental para que isso aconteça. Os departamentos de *marketing* estão percebendo como a marca, o número de acessos, o índice de retorno do consumidor e a sua satisfação são afetados pela usabilidade de um *website*. Além disso, a presença ou ausência de um bom *design* de interação podem determinar o sucesso ou fracasso de uma companhia – tal como ocorreu com aquela famosa indústria de roupas *ponto.com*, que não avaliou adequadamente a importância de um bom *design* de interação e acabou pagando caro por essa distração, falindo em poucos meses após ir a público.<sup>2</sup> Ela contava, então, com uma interface gráfica em 3D, com painel colorido e animado no estilo “pinheirinho de Natal”. Um dos problemas quanto a isso dizia respeito ao tempo excessivo levado para realizar o *download*. Ademais, em geral era necessário mais de 20 minutos para se conseguir realizar um pedido por meio de um formulário *on-line* demasiado extenso e demora-

---

<sup>2</sup> Isso aconteceu antes da crise das *ponto.com*, em 2001.

### QUADRO 1.2 Denominando a profissão: de *designers* de interface a arquitetos da informação

Há 10 anos, quando uma empresa pretendia desenvolver uma interface para um produto interativo, ela procurava por *designers* (ou projetistas) de interface. Tais profissionais envolviam-se principalmente com o *design* e a avaliação de produtos para aplicações em *desktops*. Agora que a faixa potencial de produtos interativos diversificou-se bastante, juntamente com a percepção de que é importante contar com uma interface correta, um grande número de outros profissionais começou a surgir:

- *designers* de interação (pessoas envolvidas no *design* de todos os aspectos interativos de um produto, não somente no *design* gráfico de uma interface)
- engenheiros de usabilidade (pessoas que avaliam produtos utilizando métodos e princípios de usabilidade)
- *web designers* (pessoas que desenvolvem e criam o *design* visual de *websites* – os leiautes, por exemplo)
- arquitetos da informação (pessoas que têm idéias de como planejar e estruturar produtos interativos, especialmente *websites*)
- *designers* de novas experiências aos usuários (pessoas que realizam todas as tarefas supracitadas, mas que também podem realizar estudos de campo a fim de fomentar o *design* de produtos)

do de preencher – tão-somente para depois se descobrir que o pedido não havia sido feito. Os clientes frustravam-se com o *site* e nunca mais retornavam.

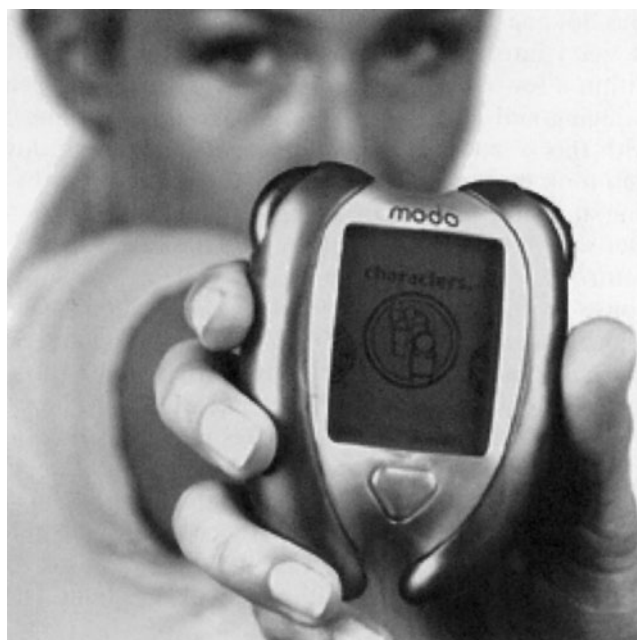
Em resposta à crescente demanda por *design* de interação, um número cada vez maior de consultores está se estabelecendo como especialista na área. Uma empresa a seguir essa tendência é a Swim, estabelecida por Gitta Salomon visando a auxiliar clientes no *design* de produtos interativos (veja a entrevista que ela nos concedeu ao final deste capítulo). Gitta aponta que com frequência as empresas percebem a importância do *design* de interação, mas não sabem como realizá-lo. Acabam entrando, pois, em contato com outras empresas, como a Swim, em busca de auxílio e tendo seus produtos parcialmente desenvolvidos. O auxílio pode vir na forma de uma revisão detalhada da usabilidade e do *design* do produto feita por um especialista. (Para saber mais sobre avaliação especializada, veja Capítulo 13). De maneira mais extensiva, pode-se ajudar os clientes a criar seus produtos.

Uma outra empresa de *design* já estabelecida e praticante do *design* de interação é a IDEO, que hoje dispõe de muitas filiais em todo o mundo. Contando com mais de 20 anos de experiência na área, projeta produtos, serviços e ambientes para outras empresas, proporcionando novas experiências aos usuários (Sprenberg *et al.*, 1995). A IDEO desenvolveu milhares de produtos para numerosos clientes, sempre seguindo sua marca particular de *design* centrado no usuário (veja Figura 1.5).

## 1.4 O que está envolvido no processo de *design* de interação?

Essencialmente, o processo de *design* de interação envolve quatro atividades básicas:

1. Identificar necessidades e estabelecer requisitos.
2. Desenvolver *designs* alternativos que preencham esses requisitos.
3. Construir versões interativas dos *designs*, de maneira que possam ser comunicados e analisados.
4. Avaliar o que está sendo construído durante o processo.



**Figura 1.5** Um produto inovador desenvolvido pela IDEO: Scout Modo, um dispositivo sem fio que exibe informações atualizadas sobre o que está acontecendo na cidade.

Espera-se que tais atividades complementem-se umas às outras e que sejam sempre repetidas. Por exemplo, medir a usabilidade do que foi construído, no que se refere à facilidade de uso, dá um retorno a respeito de que mudanças devem ser feitas ou de quais requisitos não foram preenchidos.

Avaliar o que foi construído está no centro do *design* de interação. É preciso assegurar que o produto é usável. A avaliação é geralmente realizada com uma abordagem centrada no usuário, isto é, – como o nome sugere – procurando envolver usuários em todo o processo de *design*. Existem muitas maneiras diferentes de atingir esse objetivo: por exemplo, observando os usuários, conversando com eles, entrevistando-os, testando-os utilizando tarefas de desempenho, modelando sua *performance*, pedindo que preencham questionários e até mesmo pedindo que se tornem *co-designers*. O que se descobre a partir das diferentes maneiras de se envolver com o conhecimento do usuário e de obtê-lo é então interpretado com relação às atividades do *design* em andamento (fornecemos mais detalhes sobre todos esses aspectos de avaliação nos Capítulos 10–14).

Tão importante quanto envolver os usuários na avaliação de um produto interativo é entender como as pessoas realizam normalmente as tarefas. Essa forma de pesquisa deveria ser realizada antes da construção de um produto interativo. Os Capítulos 3, 4 e 5 tratam sobre o assunto, explicando detalhadamente como os indivíduos agem e interagem uns com os outros, com a informação e com as várias tecnologias, e descrevendo seus pontos fortes e fracos. Tal conhecimento pode auxiliar os *designers* a determinar que soluções escolher dentre as muitas alternativas de *design* disponíveis e como desenvolvê-las e testá-las posteriormente. O Capítulo 7 descreve como as necessidades dos usuários podem ser traduzidas em requisitos, ao

passo que o Capítulo 9 explica como envolver efetivamente os usuários no processo de *design*.

Uma das principais razões para se ter um melhor entendimento acerca dos usuários se deve ao fato de que usuários diferentes têm necessidades diferentes e produtos interativos precisam ser projetados de acordo com tais necessidades. Por exemplo, as crianças apresentam expectativas diferentes das dos adultos quanto à maneira como querem aprender ou jogar. Nesse sentido, podem considerar desafios interativos e personagens de desenhos animados altamente motivadores, ao passo que a maioria dos adultos os tem como algo aborrecido. Em contrapartida, os adultos geralmente apreciam discussões sobre os tópicos, ao passo que as crianças, por sua vez, as consideram maçantes. Assim como os objetos de uso diário – roupas, comida e jogos – foram projetados de maneira diferente para crianças, adolescentes e adultos, os produtos interativos devem ser projetados para ir ao encontro das necessidades dos tipos diferentes de usuários.

Além das quatro atividades básicas de *design*, existem três características-chave quanto ao processo *design* de interação:

1. Os usuários devem estar envolvidos no desenvolvimento do projeto.
2. A usabilidade específica e as metas decorrentes da experiência do usuário devem ser identificadas, claramente documentadas e acordadas no início do projeto.
3. A iteração em todas as quatro atividades é inevitável.

Já mencionamos a importância de envolver usuários no projeto e retornaremos a esse tópico ao longo de todo o livro. Também trataremos de *design* iterativo mais tarde, quando falarmos dos vários *designs* e métodos de avaliação. Na próxima seção, descrevemos a usabilidade e as metas decorrentes da experiência do usuário.

## 1.5 As metas do *design* de interação

Parte do processo de entender as necessidades do usuário, no que diz respeito a projetar um sistema interativo que as atenda, consiste em ser claro quanto ao objetivo principal. Tratar-se-ia, então, de projetar um sistema muito eficiente que permitisse aos usuários ser altamente produtivos em seu trabalho? Ou de projetar um sistema desafiador e motivador que fornecesse suporte a um aprendizado eficaz? Ou ainda de alguma outra coisa? Denominamos essas preocupações principais *metas de usabilidade* e *metas decorrentes da experiência do usuário*. As duas diferem no que se refere ao modo como são operacionalizadas, isto é, como podem ser atingidas e por que meios. As metas de usabilidade estão preocupadas com preencher critérios específicos de usabilidade (p. ex.: eficiência), e as metas decorrentes da experiência do usuário, com explicar a qualidade da experiência desta (p. ex.: ser esteticamente agradável).

### 1.5.1 Metas de usabilidade

Para recapitular, a usabilidade é geralmente considerada como o fator que assegura que os produtos são fáceis de usar, eficientes e agradáveis – da perspectiva do usuário. Implica otimizar as interações estabelecidas pelas pessoas com produtos interativos, de modo a permitir que realizem suas atividades no trabalho, na escola e em casa. Mais especificamente, a usabilidade é dividida nas seguintes metas:

- ser eficaz no uso (eficácia)

- ser eficiente no uso (eficiência)
- ser segura no uso (segurança)
- ser de boa utilidade (utilidade)
- ser fácil de aprender (*learnability*)
- ser fácil de lembrar como se usa (*memorability*)

Para cada meta apresentamos uma descrição detalhada, seguida de uma importante questão relacionada.

*Eficácia* é uma meta bastante geral e se refere a quanto um sistema é bom em fazer o que se espera dele.

*Pergunta:* o sistema é capaz de permitir que as pessoas aprendam bem, realizem seu trabalho de forma eficiente, acessem as informações de que necessitam, compre os produtos que desejam, etc.?

*Eficiência* se refere à maneira como o sistema auxilia os usuários na realização de suas tarefas. A secretária eletrônica descrita no início do capítulo foi considerada eficiente, pois permitia ao usuário realizar tarefas simples mediante um número mínimo de passos (p. ex.: ouvir as mensagens). Em contraste, o sistema de correio de voz (*voice mail*) foi considerado ineficiente porque exigia que o usuário executasse muitos passos e aprendesse um conjunto arbitrário de seqüências para a mesma tarefa comum. Nesse sentido, uma maneira eficiente de fornecer suporte a tarefas comuns é permitir que o usuário utilize um único botão ou tecla. Um exemplo de onde esse tipo de mecanismo de eficiência tem sido empregado efetivamente é em *e-tailing*. Nesse sistema, uma vez que os usuários tenham preenchido todos os detalhes pessoais em um *site* de comércio eletrônico para proceder a uma compra, eles terão a oportunidade de deixar que o *site* salve todas essas informações. Assim, quando pretenderem realizar alguma outra compra nesse mesmo *site*, não precisarão fornecer novamente tais dados. Um mecanismo inteligente patenteado pela Amazon.com é a opção *one-click*, que, com apenas um clique, permite ao usuário realizar uma outra compra.

*Pergunta:* Uma vez que os usuários tiverem aprendido como utilizar um sistema para realizar suas tarefas, conseguirão eles manter um alto nível de produtividade?

*Segurança* implica proteger o usuário de condições perigosas e situações indesejáveis. Com relação ao primeiro aspecto ergonômico, refere-se às condições externas do local de trabalho. Por exemplo, onde houver condições de risco – como com máquinas de raio X ou em indústrias químicas —, os operadores deverão estar aptos a interagir com e controlar sistemas baseados em computador remotamente. O segundo aspecto diz respeito a auxiliar qualquer tipo de usuário, em qualquer tipo de situação, a evitar os perigos de realizar ações indesejáveis acidentalmente. Também diz respeito ao possível medo dos usuários diante das conseqüências de seus erros e a como isso afeta o seu comportamento. Fazer os sistemas baseados em computador mais seguros nesse sentido envolve (i) prevenir o usuário de cometer erros graves reduzindo o risco de que ele pressione teclas/botões errados por engano (um exemplo implica *não* colocar o comando *sair* (*quit*) ou *remover* (*delete*) próximo ao comando *salvar* (*save*), em um menu) e (ii) fornecer aos usuários várias formas de recuperação ou retorno, no caso de cometerem erros. Os sistemas interativos seguros propiciam confiança e permitem que o usuário tenha a oportunidade de explorar a interface a fim de experimentar outras operações (veja Figura 1.6a). Outros mecanismos de segurança incluem os recursos de desfazer ações (*undo*) e caixas de diálogo para confirmações. Tais recursos oferecem ao usuário uma outra oportunidade de reconsiderar suas intenções (uma opção

**QUADRO 1.3** A regra dos dez minutos

Um critério para avaliar se um sistema é fácil de entender consiste em aplicar a “regra dos dez minutos” (Nelson, 1980). Segundo esse critério, os usuários inexperientes deverão conseguir utilizar um sistema em menos de dez minutos; do contrário, esse sistema apresenta falhas. Como apontado por Rubinstein e Hersh (1984), muitos sistemas computacionais não atingem esse requisito. Para tornar os sistemas mais fáceis de ser aprendidos, eles sugerem que os *designers* tirem proveito do conhecimento das pessoas: “Não se espera que um sistema computacional para arquitetos ensine arquitetura. Muito pelo contrário: a regra dos dez minutos exige que aquilo que um arquiteto já saiba seja útil no aprendizado do sistema” (Rubinstein e Hersh, 1984, p. 9).

**Quando a regra dos dez minutos não é apropriada?**

Trata-se de uma regra útil e não muito rigorosa para a avaliação de muitos tipos de sistemas.

Entretanto, não é apropriada para o uso com sistemas complexos, levando-se em conta que seria difícil, e até uma falta de consideração, pensar que o usuário pudesse aprender a utilizá-los em menos de dez minutos. Por exemplo, você se sentiria seguro, sabendo que o piloto do seu avião levou apenas dez minutos para aprender a utilizar todos aqueles dispositivos da cabine de comando? Espera-se que ele tenha passado algum tempo (além dos anos de treinamento para pilotos) aprendendo a utilizar todos os controles e painéis referentes àquele avião específico e que também saiba o que fazer caso algum deles não esteja funcionando corretamente. Da mesma forma, está fora da realidade pensar que dez minutos seriam suficientes para aprender a lidar com um sistema que apresenta diversas funcionalidades (p. ex.: um processador de texto) ou que exige um alto nível de habilidades (p. ex.: um *videogame*).

bem conhecida e utilizada em aplicações de *e-mail* é a caixa de diálogo, que aparece ao se selecionarem mensagens a serem apagadas, fazendo a seguinte pergunta: “Você tem certeza de que deseja apagar todas essas mensagens?” Veja Figura 1.6b).

*Pergunta:* O sistema previne os usuários de cometer erros graves e – se mesmo assim o fizerem – permite que esses erros sejam recuperados facilmente?

*Utilidade* refere-se à medida na qual o sistema propicia o tipo certo de funcionalidade, de maneira que os usuários possam realizar aquilo de que precisam ou que desejam. Um exemplo de sistema com alta utilidade é um pacote de *software* de contabilidade que fornece uma ferramenta computacional poderosa, utilizada pelos contadores para calcular as restituições de impostos. Um exemplo de sistema com baixa utilidade é uma ferramenta em um *software* de desenho que não permite aos usuários desenhar com as mãos livres, obrigando-os a fazer uso do *mouse* em suas criações, utilizando somente formas poligonais.

*Pergunta:* O sistema fornece um conjunto apropriado de funções que permita aos usuários realizar todas as suas tarefas da maneira que desejam?

*Capacidade de aprendizagem (learnability)* refere-se a quão fácil é aprender a usar o sistema. É fato sabido que as pessoas não gostam de passar muito tempo aprendendo como fazê-lo. Preferem utilizá-lo logo e tornar-se competentes para realizar tarefas sem muito esforço. Tal fato se verifica especialmente com relação aos produtos interativos de uso diário (p. ex.: TV interativa, correio eletrônico) mas também aos utilizados com menos frequência (videoconferência). Até um certo ponto, as pessoas estão preparadas para passar mais tempo aprendendo sistemas mais complexos, que propiciam um conjunto de funcionalidade maior (p. ex.: ferramentas de autoria para a *web*, processadores de texto). Nessas situações, o CD-ROM e os tutoriais *on-line* podem auxiliar fornecendo material interativo passo a passo, com exercícios práticos. Muitas pessoas, porém, consideram esses recursos entediante e muitas vezes até difíceis de ser relacionados com as tarefas que pretendem realizar. Uma questão-chave

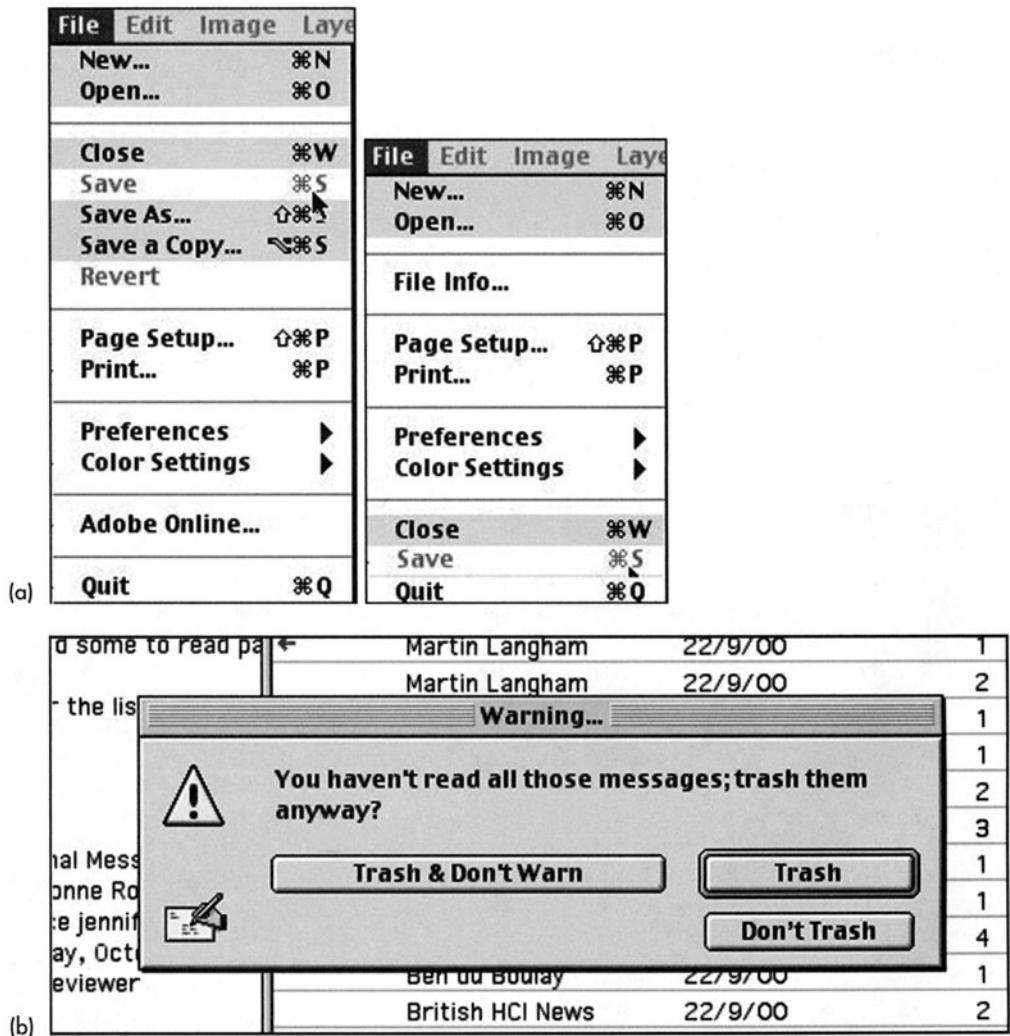


Figura 1.6 (a) Um menu seguro e um não-seguro. Qual é qual e por quê? (b) Mensagem de aviso do Eudora.

consiste em determinar quanto tempo os usuários estarão preparados para gastar conhecendo um sistema. Não há muito sentido em desenvolver uma série de funcionalidades se a maioria dos usuários não pode ou não está preparada para passar algum tempo aprendendo a utilizá-las.

*Pergunta:* Quão fácil é e que tempo se leva para (i) iniciar o uso das tarefas fundamentais de um sistema e (ii) aprender o conjunto de operações necessárias para realizar um conjunto mais amplo de tarefas?

*Capacidade de memorização (memorability)* refere-se à facilidade de lembrar como utilizar um sistema, depois de já se ter aprendido como fazê-lo – algo especialmente importante para sistemas interativos que não são utilizados com muita frequência. Se os usuários não utilizam um sistema ou uma operação por alguns meses ou mais, devem poder lembrar ou pelo menos ser rapidamente lembrados sobre como



fazê-lo, e não ficar reaprendendo como realizar as tarefas. Infelizmente, isso tende a acontecer sempre que as operações que devem ser aprendidas são obscuras, ilógicas ou pobremente encadeadas. Os usuários necessitam ser ajudados a lembrar como realizar as tarefas. Há muitas formas de projetar a interação para que ela forneça suporte a esse aspecto. Por exemplo, os usuários podem ser auxiliados a lembrar a sequência de operações em estágios diferentes de uma tarefa por meio de ícones representativos, nomes de comandos e opções de menu. Além disso, estruturar opções e ícones, de maneira que sejam colocados em categorias relevantes de opções (p. ex.: dispor todas as ferramentas de desenho no mesmo local na tela), pode ajudar o usuário a lembrar onde procurar por uma determinada ferramenta em um certo estágio de uma tarefa.

*Pergunta:* Que tipos de suporte de interface foram fornecidos com o objetivo de auxiliar os usuários a lembrar como realizar tarefas, especialmente para sistemas e operações que não são utilizadas com muita frequência?

### ATIVIDADE 1.3

Quanto tempo você acha que se *deveria* levar para aprender a utilizar os seguintes produtos interativos e quanto tempo *realmente* a maioria das pessoas leva para tal? Qual o grau de capacidade de memorização (*memorability*) deles?

- (a) utilizar um videocassete para passar um filme
- (b) utilizar um videocassete para programar a gravação de dois programas
- (c) utilizar uma ferramenta de autoria para criar um *website*.

### Comentário

- (a) Ligar um vídeo para assistir a um filme deveria ser tão simples como ligar o rádio; deveria levar menos de 30 segundos para funcionar e depois realizar sua atividade sem maiores questões. A maioria das pessoas sabe como fazê-lo; no entanto, alguns sistemas requerem que o usuário selecione o canal do vídeo, dentre 50 ou mais, utilizando um ou dois controles remotos. Outras configurações também precisam ser ajustadas antes de o vídeo começar a rodar. A maioria das pessoas consegue lembrar como ligá-lo, uma vez que provavelmente já devem ter utilizado algum.
- (b) Essa é uma operação mais complexa. Aprender como programar o videocassete e checar se está tudo da forma correta leva um pouco mais de tempo. Na realidade, muitos desses aparelhos são tão mal projetados, que 80% da população não consegue realizar essa tarefa, apesar de inúmeras tentativas. Poucas pessoas lembram como programar o vídeo para gravar um programa, muito por a interação necessária ter sido mal projetada, com pouco ou nenhum *feedback*, e por não ser lógica da perspectiva do usuário. Dessas pessoas, poucas irão ter paciência para olhar o manual novamente.
- (c) Uma ferramenta de autoria bem projetada deveria permitir ao usuário criar uma página básica em cerca de 20 minutos. Aprender a totalidade das operações e possibilidades provavelmente levará muito mais tempo – possivelmente alguns dias. Na realidade, existem algumas boas ferramentas de autoria que permitem ao usuário iniciar o uso logo, oferecendo *templates* que podem ser adaptados. A maioria dos usuários irá ampliar seu repertório, levando uma hora ou mais para aprender mais funções. No entanto, algumas poucas pessoas realmente aprendem a utilizar todo o conjunto de funções que uma ferramenta dessas oferece. Os usuários tendem a lembrar operações frequentemente utilizadas (p. ex.: recortar e colar, inserir imagens), especialmente se forem consistentes com relação à forma como essas ações são realizadas em outras aplicações. Contudo, algumas outras operações usadas com menos frequência talvez tenham que ser reaprendidas (p. ex.: formatar tabelas).

As metas de usabilidade discutidas até aqui se encaixam bem com o *design* de sistemas destinados a apoiar práticas de trabalho. Em particular, são altamente relevantes para empresas e organizações que estejam introduzindo ou atualizando aplicações para *desktops* e sistemas em rede – dos quais espera-se que aumentem a produtividade, melhorando e aperfeiçoando a maneira de realizar o trabalho. Além de guiar as empresas no que diz respeito a questões específicas, as metas de usabilidade transformam-se em *critérios de usabilidade*. Essas são metas específicas que permitem à usabilidade de um produto ser avaliada em termos de como podem aprimorar (ou não) o desempenho de um usuário. Exemplos de critérios de usabilidade comumente utilizados são os seguintes: tempo para completar uma tarefa (eficiência), tempo para aprender uma tarefa (*learnability*) e o número de erros cometidos quando se realiza uma tarefa num dado tempo (*memorability*).

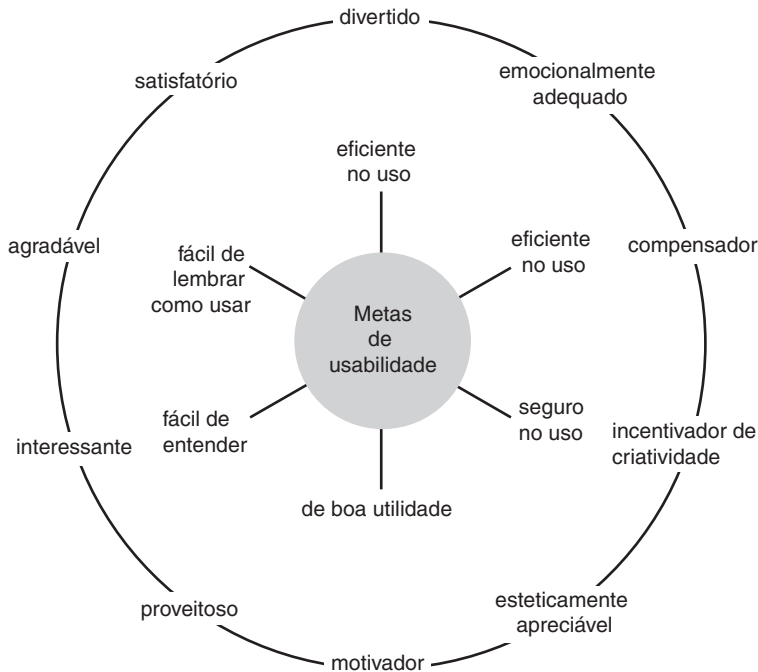
### 1.5.2 Metas decorrentes da experiência do usuário

Ao perceberem que as novas tecnologias estão oferecendo maiores oportunidades para fornecer suporte às pessoas em seu dia-a-dia, pesquisadores e profissionais foram levados a considerar outras metas. A emergência de tecnologias (p. ex.: realidade virtual, *web*, computação móvel) inseridas em uma diversidade de áreas de aplicação (p. ex.: entretenimento, educação, residências, áreas públicas) trouxe à tona um conjunto muito maior de interesses. Além de enfocar principalmente a melhoria da eficiência e da produtividade no trabalho, o *design* de interação está cada vez mais preocupado com a criação de sistemas que sejam:

- satisfatórios
- agradáveis
- divertidos
- interessantes
- úteis
- motivadores
- esteticamente apreciáveis
- incentivadores de criatividade
- compensadores
- emocionalmente adequados

O objetivo de desenvolver produtos interativos agradáveis, divertidos, esteticamente apreciáveis, etc. está principalmente na experiência que estes proporcionarão ao usuário, isto é, como o usuário se sentirá na interação com o sistema. Isso envolve explicar a natureza da experiência do usuário em termos subjetivos. Por exemplo, um novo pacote de *software* infantil que permita às crianças criar suas próprias músicas pode ser projetado com o objetivo principal de ser engraçado e interessante. Assim, as metas decorrentes da experiência do usuário diferem das metas de usabilidade, que são mais objetivas, no sentido de que estão preocupadas com maneira como os usuários lidam com um produto interativo. A relação entre os dois é mostrada na Figura 1.7.

Muito do trabalho sobre diversão, prazer, etc. foi desenvolvido na indústria de jogos e entretenimento, que apresenta um certo interesse em entender o papel do prazer mais detalhadamente. Os aspectos descritos como contribuintes para o prazer



**Figura 1.7** Metas de usabilidade e metas decorrentes da experiência do usuário. As metas de usabilidade são fundamentais para o *design* de interação e são operacionalizadas por meio de critérios diferentes. No círculo externo são mostradas as metas decorrentes da experiência do usuário, as quais são menos claramente definidas.

incluem o seguinte: atenção, ritmo, jogo, interatividade, controle consciente e inconsciente, envolvimento e estilo de narrativa. Foi até mesmo sugerido que nesses contextos poderia ser interessante a construção de sistemas que não fossem fáceis de usar, dando oportunidades para experiências diferentes das projetadas com base nas metas de usabilidade (Frohlich e Murphy, 1999). Interagir com uma representação virtual utilizando um dispositivo físico (p. ex.: bater em um prego virtual, representado na tela, com um martelo de borracha), comparando-se com uma maneira utilizar um meio mais eficiente de fazer a mesma coisa (p. ex.: selecionar uma opção utilizando chaves de comando), poderá exigir *mais esforço*, ainda que, por outro lado, resultar em uma experiência *mais agradável e divertida*.

Reconhecer e entender o equilíbrio entre as metas de usabilidade e as decorrentes da experiência do usuário é importante. Em particular, permite aos *designers* conscientizar-se das conseqüências de buscar combinações diferentes dessas metas, levando em consideração as necessidades dos usuários. Obviamente, nem todas as metas de usabilidade e as decorrentes da experiência do usuário se aplicam a todo produto interativo em desenvolvimento. Algumas combinações irão também ser incompatíveis. Por exemplo, pode não ser possível ou desejável projetar um sistema de controle de processo que seja ao mesmo tempo seguro e divertido. Como enfatizado durante todo este capítulo, o que é importante depende do contexto de uso, da tarefa a ser realizada e de quem são os usuários pretendidos.

**ATIVIDADE 1.4**

A seguir, apresentamos alguns produtos interativos. Quais você acha que sejam as metas de usabilidade e as decorrentes da experiência do usuário em cada um deles?

- (a) um dispositivo portátil que permite a crianças comunicar-se umas com as outras e brincar com jogos colaborativos
- (b) um sistema de vídeo e computador para conferências que permite aos estudantes aprender em casa
- (c) uma aplicação da Internet que permite ao público em geral acessar seu prontuário médico através de uma TV interativa
- (d) um sistema CAD para arquitetos e engenheiros
- (e) uma comunidade *on-line* que fornece suporte a pessoas que recentemente tiveram amigos e familiares falecidos

**Comentário**

- (a) Esse dispositivo colaborativo deve ser de uso simples, eficaz, eficiente, fácil de aprender a utilizar, engraçado e interessante.
  - (b) Esse dispositivo para aprendizagem deve ser fácil de entender e de utilizar, eficaz, motivador e compensador.
  - (c) Esse sistema pessoal deve ser seguro, fácil de usar e de lembrar como usar, eficaz e eficiente.
  - (d) Essa ferramenta precisa ser fácil de aprender, fácil de lembrar como usar, ter uma boa utilidade, ser segura, eficiente, eficaz, fornecer suporte à criatividade e ser esteticamente agradável.
  - (e) Esse sistema deve ser fácil de aprender, fácil de usar, motivador, emocionalmente adequado e compensador.
- 

## 1.6 Um pouco mais sobre usabilidade: princípios de usabilidade e *design*

Uma outra maneira de conceitualizar a usabilidade se dá em termos de princípios de *design*. Trata-se de abstrações generalizáveis, destinadas a orientar os *designers* a pensar sobre aspectos diferentes de seus *designs*. Um exemplo bem conhecido é o *feedback*: os sistemas devem ser projetados para fornecer *feedback* adequado aos usuários, assegurando que eles saibam os próximos passos durante a realização das tarefas. Os princípios de *design* são derivados de uma mistura de conhecimento baseado em teoria, experiência e senso comum. Tendem a ser escritos de maneira prescritiva, sugerindo aos *designers* o que utilizar e o que evitar na construção de uma interface – o “sim e não” (*do’s and don’ts*) do *design* de interação. Mais especificamente, são destinados a auxiliar os *designers* a explicar e melhorar o projeto (Thimbleby, 1990). Todavia, não servem para especificar como realizar o *design* de uma interface real (p. ex.: dizer ao *designer* como projetar um certo ícone ou como estruturar um portal na *web*), mas servem mais como um conjunto de itens que devem ser lembrados, assegurando que certas coisas foram acrescentadas à interface.

Vários princípios de *design* já foram desenvolvidos. Os mais conhecidos referem-se a como determinar o que os usuários devem ver e fazer quando realizam tarefas utilizando um produto interativo. Descrevemos aqui, brevemente, os mais comuns: visibilidade, *feedback*, restrições, mapeamento, consistência e *affordance*. Todos eles já

foram descritos extensivamente por Don Norman (1988), em seu *bestseller* *The design of everyday things*.

**Visibilidade** A importância da visibilidade é exemplificada pelos dois exemplos contrastantes que apresentamos no início do capítulo. O sistema de correio de voz não deixava visíveis a presença e o número de mensagens recebidas, ao passo que a secretária eletrônica deixou esses dois aspectos totalmente visíveis. Quanto mais visíveis forem as funções, mais os usuários saberão como proceder. Por outro lado, quando as funções estão “fora de alcance”, torna-se mais difícil encontrá-las e saber como utilizá-las. Norman (1988) descreve os controles de um carro a fim de enfatizar esse ponto. Os controles para diferentes operações são claramente visíveis (p. ex.: faróis, pisca-pisca, buzina, pisca-alerta), indicando o que poderá ser feito. A relação entre a maneira como os controles são dispostos no carro e o que eles realizam facilita na hora de encontrar com mais rapidez o controle apropriado para as tarefas.

**Feedback** O *feedback* está relacionado ao conceito de visibilidade. Podemos melhor ilustrar esse princípio fazendo uma analogia com o modo como seria a vida diária sem ele. Imagine-se tentando tocar uma guitarra, fatiando um pão ou escrevendo com uma caneta, sem que nenhuma dessas ações produzisse qualquer efeito durante alguns segundos. A música, o pão fatiado e as palavras no papel surgiriam com algum atraso, tornando quase impossível prosseguir com a realização da tarefa.

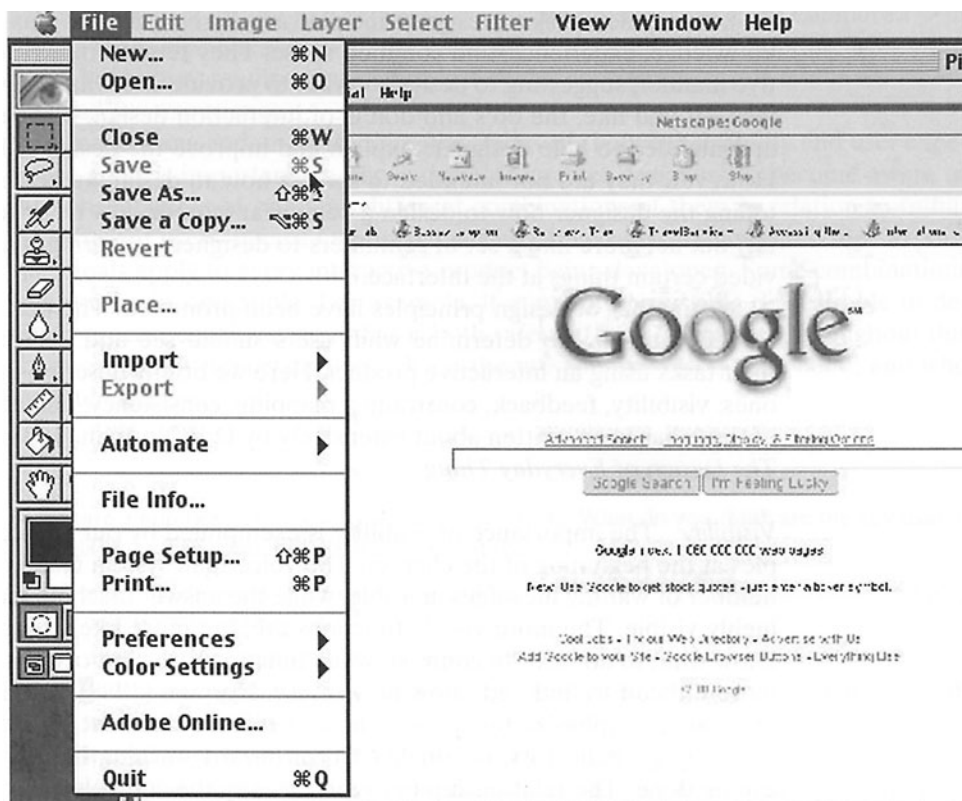
O *feedback* se refere ao retorno de informações a respeito de que ação foi feita e do que foi realizado, permitindo à pessoa continuar a atividade. Vários tipos de *feedback* estão disponíveis para o *design* de interação – áudio, tátil, verbal, visual e combinações dos mesmos. Decidir quais combinações são apropriadas para os diferentes tipos de atividades e interatividades constitui um ponto central. Utilizar o *feedback* da maneira certa pode proporcionar a visibilidade necessária para a interação do usuário.

**Restrições** O conceito de restrição refere-se à determinação das formas de delimitar o tipo de interação que pode ocorrer em um determinado momento. Existem várias maneiras de fazer isso. Uma prática comum no *design* de interfaces gráficas consiste em desativar certas opções do menu sombreando-as, restringindo as ações do usuário somente às permitidas naquele estágio da atividade (veja Figura 1.8).

Uma das vantagens dessa forma de restrição é impedir o usuário de selecionar opções incorretas e, portanto, reduzir a chance de erro. O uso de diferentes tipos de representações gráficas pode também restringir a interpretação de um problema ou o espaço para informação. Por exemplo, fluxogramas mostram quais objetos estão relacionados a outros, restringindo, assim, a maneira de perceber a informação.

Norman (1999) classifica as restrições em três categorias: física, lógica e cultural. As restrições físicas referem-se à forma como objetos físicos restringem o movimento das coisas. Por exemplo, o modo como um disco externo pode ser colocado em um *drive* é restringido fisicamente por sua forma e seu tamanho, de maneira que ele pode ser inserido no local somente de um modo. Da mesma forma, as teclas também só podem ser pressionadas de uma maneira.

As restrições lógicas são as que dependem do entendimento que a pessoa tem sobre a maneira como o mundo funciona (conforme as esferas da secretária eletrônica). Dependem do senso comum dos indivíduos a respeito das ações e de suas consequências. Pegar uma esfera física e colocá-la em outro local no telefone cria a expectativa de que alguma outra coisa irá acontecer. Tornar óbvios as ações e seus efeitos permite às pessoas deduzir logicamente que outras ações são necessárias. Desabilitar algumas opções do menu, quando essas não são apropriadas para a tarefa ime-



**Figura 1.8** Menu ilustrando a disponibilidade restrita de opções, como um exemplo de restrição lógica. As áreas sombreadas indicam opções desativadas.

diata, implica uma restrição lógica, pois permite que os usuários raciocinem por que (ou por que não) foram projetadas dessa forma e que opções estão disponíveis.

As restrições culturais acontecem no âmbito das convenções aprendidas – como o uso de vermelho para alertas, de certos tipos de sinais de áudio para perigo e de uma carinha sorridente para representar alegria. A maioria das restrições culturais são arbitrárias, no sentido de que sua relação com o que está sendo representado é abstrata e que poderia tê-lo sido de outra maneira (p. ex.: o uso de amarelo, em vez de vermelho). Assim sendo, elas devem ser aprendidas. Uma vez aprendidas e aceitas por um grupo cultural, tornam-se convenções universalmente aceitas. Duas convenções de interfaces universalmente aceitas são o uso de janelas para a apresentação de informações e o uso de ícones para representar operações e documentos.

**Mapeamento** Refere-se à relação entre os controles e os seus efeitos no mundo. Quase todos os artefatos necessitam de algum tipo de mapeamento entre controles e efeitos, seja uma lanterna, um carro, uma usina elétrica ou uma cabine (*cockpit*). Um exemplo de um bom mapeamento entre controle e efeito são as setas utilizadas para representar o movimento para cima ou para baixo do cursor em um teclado de computador. O mapeamento da posição relativa dos controles e seus efeitos é também importante. Considere os vários tipos de dispositivos para executar músicas (p. ex.: MP3, CD *player*, gravador). Como são mapeados os controles *rewind*, *play* e *fast forward*.

*ward*, de acordo com os efeitos desejados? Geralmente eles seguem uma convenção comum, apresentando uma sequência de botões em que o *play* fica ao meio, o *rewind* à esquerda e o *fast forward* à direita. Essa configuração está mapeada diretamente de acordo com a direção das ações (veja Figura 1.9a). Imagine como isso seria difícil, caso fosse utilizado o mapeamento da Figura 1.9b. Observe a Figura 1.10 e determine qual mapeamento é adequado e qual causaria problemas ao usuário.

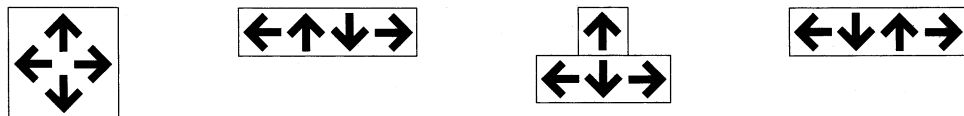
**Consistência** Refere-se a projetar interfaces de modo que tenham operações semelhantes e que utilizem elementos semelhantes para a realização de tarefas similares. Uma interface consistente é aquela que segue regras, tais como o uso da mesma operação para selecionar todos os objetos. Por exemplo, uma operação consistente utiliza a mesma ação para marcar qualquer objeto gráfico na interface, como clicar sempre o botão esquerdo do *mouse*. Interfaces inconsistentes, por outro lado, permitem exceções à regra. Um exemplo disso se dá quando certos objetos gráficos (p. ex.: mensagens eletrônicas apresentadas em uma tabela) podem ser marcados somente utilizando-se o botão direito do *mouse*, enquanto todas as outras operações são marcadas utilizando-se o botão esquerdo. Um dos problemas relacionado a esse tipo de inconsistência se deve ao fato de ela ser um tanto arbitrária, dificultando a tarefa de lembrar das ações, o que aumenta a propensão ao cometimento de erros.

Um dos benefícios de interfaces consistentes, portanto, é serem mais fáceis de aprender e usar. Os usuários aprendem apenas um único modo de operação aplicável a todos os objetos. Esse princípio funciona bem para interfaces simples com operações limitadas, como um mini *CD player* com um pequeno número de operações mapeadas em botões separados. Nesse exemplo, tudo o que o usuário tem de fazer é aprender o que cada botão representa e selecioná-los. No entanto, pode ser mais problemático aplicar o conceito de consistência a interfaces mais complexas, especialmente quando muitas operações diferentes precisam ser projetadas. Por exemplo, imagine como seria projetada uma interface para uma aplicação que oferecesse centenas de operações (p. ex.: uma aplicação para processamento de texto). Simplesmente não existe espaço suficiente para milhões de botões, cada um mapeando uma operação individual. Mesmo se houvesse, seria extremamente difícil e demorado para o usuário procurar qual dentre eles realiza a operação desejada.

Uma solução de *design* muito mais eficaz é a criação de categorias de comandos que podem ser mapeados em subconjuntos de operações. Para o processador de texto, as centenas de operações disponíveis são categorizadas em subconjuntos de diferentes menus. Todos os comandos relacionados a operações com arquivos (p. ex.: salvar, abrir, fechar) são colocados juntos no mesmo menu-arquivo. Da mesma forma, todos os comandos relacionados à formatação de texto são colocados em um menu de formatação. Selecionar uma operação implicará então uma questão de dirigir-se à categoria correta de opções (menu) e procurar a operação desejada, em vez de ter que procurá-la em uma longa lista. Contudo, a regra de consistência – de



**Figura 1.9** (a) Mapeamento natural entre as ações *rewind*, *play* e *fast forward*, em um dispositivo de gravação de fita cassete. (b) Um mapeamento arbitrário alternativo.



**Figura 1.10** Quatro possíveis combinações de mapeamentos com setas. Qual é o mais natural?

se ter um mapeamento visível entre comando e operação – é quebrada. As operações não são imediatamente visíveis na interface; pelo contrário, estão escondidas em categorias diferentes de menus. Além disso, ao ser acionado um menu principal, alguns itens do menu são imediatamente visíveis, enquanto outros permanecem escondidos até que os itens visíveis sejam arrolados. Assim sendo, os usuários necessitam aprender quais itens são visíveis em cada categoria de menu e quais estão escondidos em submenus.

A maneira como os itens estão divididos entre as categorias de itens de menus também pode parecer inconsistente para os usuários. Várias operações aparecem em menus a que não pertencem. Por exemplo, a operação de ordenação (muito útil para listas de referências ou nomes em ordem alfabética) do Microsoft Word 2001 está no menu Tabela (versão para Mac). Na versão Word 98, encontrava-se tanto no menu Ferramentas como no menu Tabela. Sempre pensamos nessa operação como pertencente ao menu Ferramentas (como Contar Palavras), e houve uma certa frustração ao descobirmos que, para a versão 2001, era *default* que estivesse somente no menu Tabela – o que é inconsistente de duas maneiras: (i) com relação à versão anterior e (ii) na categoria que foi incluída. É claro, podemos personalizar a nova versão de maneira que os menus sejam estruturados como pensamos que deveriam ser, mas isso leva tempo (especialmente quando se utilizam várias máquinas diferentes: no trabalho, em casa e em viagem).

Outro problema de consistência é determinar que aspecto de uma interface deva ser consistente com qual. Geralmente há muitas escolhas, algumas podendo ser inconsistentes com outros aspectos da interface ou com outras maneiras de realizar ações – por exemplo, o problema do desenvolvimento de um mecanismo que permite aos usuários trancar seus arquivos em um servidor compartilhado. Deveria o *designer* tentar projetá-lo de modo a ser consistente com a maneira como as pessoas trancam coisas no mundo externo (denominada consistência externa), ou com a maneira como trancam objetos no sistema (denominada consistência interna)? Existem, porém, muitas maneiras diferentes de se trancarem objetos no mundo físico (num cofre, com cadeado, à chave, com trava de segurança para crianças), assim como existem maneiras diferentes de se trancarem coisas eletronicamente (utilizando senhas, identificações, permissões, fechando as travas físicas de disquetes). O problema que os *designers* enfrentam é saber com qual deles ser consistente.

*Affordance* é um termo utilizado para se referir ao atributo de um objeto que permite às pessoas saber como utilizá-lo. Por exemplo, um botão de *mouse* convida-nos a pressioná-lo (o que acaba ativando o clique), pela forma como está fisicamente posicionado em sua concha de plástico. Simplificando, *affordance* significa “dar uma pista” (Norman, 1988). Quando a *affordance* de um objeto físico é perceptualmente ób-



via, é fácil saber como interagir com ele. Por exemplo, uma maçaneta permite que a puxemos, a alça de uma xícara permite que a seguramos e o botão do *mouse* permite que o pressionemos. Norman introduziu esse conceito nos anos 80, ao falar sobre o *design* de objetos de uso diário. Desde então, o conceito foi muito popularizado, sendo utilizado para descrever como objetos de interface deveriam ser projetados de maneira a tornar óbvio o que se pode fazer com eles. Por exemplo, elementos gráficos como botões, ícones, *links* e barras de rolagem deveriam fazer parecer óbvio o modo como deveriam ser usados: ícones devem ser projetados para permitir que sejam clicados; barras de rolagem, para que sejam movidas para cima e para baixo; botões, para que sejam pressionados.

Infelizmente o termo *affordance* tornou-se uma espécie de clichê, perdendo muito de sua força como um princípio do *design*. Norman (1999), que foi amplamente responsável por promover originalmente esse conceito em seu livro *The design of everyday things* (1998), agora se desespera com a forma como o termo vem sendo empregado na linguagem comum:

*“Coloquei uma affordance ali”, diria um participante, “gostaria de saber se há affordance aqui...”. Affordance isso, affordance aquilo. E nenhum dado, apenas opiniões. Nossa! O que foi que eu coloquei no mundo? Reação de Norman (1999), durante recente discussão sobre IHC na web.*

#### QUADRO 1.4 Uma tela pode ter *affordance*?

Um problema relativo a se aplicar o conceito de *affordance* a interfaces deve-se ao fato de objetos virtuais possuírem propriedades um tanto diferentes das de objetos físicos. Uma maçaneta permite ser puxada porque suas propriedades físicas restringem o que se pode fazer com ela, com relação à pessoa e ao ambiente. É possível abrir a porta (se estiver fechada) e fechá-la (se estiver aberta). É óbvio para uma pessoa o modo como ela deve interagir com uma maçaneta. No entanto, um objeto virtual como um ícone convida a ser clicado somente porque o usuário aprendeu inicialmente que o elemento gráfico na tela constitui uma representação que, quando clicada, faz alguma coisa acontecer (como ir para uma outra página). Poderia igualmente desencadear outras respostas do sistema, como o fechamento de uma janela. Assim, como o mapeamento entre uma representação virtual e seu comportamento é arbitrário, o usuário terá que aprender as convenções estabelecidas.

Um problema relativo a se empregar o conceito de *affordance* nesse contexto é a possibilidade de ele ser enganoso. Poderá haver equívocos por parte dos *designers*, se pensarem que objetos virtuais devem ser projetados com comportamentos e semelhanças de objetos físicos, só porque as pessoas sabem intuitivamente como interagir com eles. Isso pode levá-los a inferir que as interfaces que exi-

bem esse tipo de realismo seriam mais fáceis de entender e utilizar. No entanto, tais suposições são incorretas, pelas razões expostas acima. Para melhor ilustrar esse ponto, considere o *design* dos botões em uma tela. Vários deles foram projetados como se estivessem em 3D, aparentando ser salientes. Supõe-se que esse tipo de ilusão dê aos botões a *affordance* de serem pressionados, convidando o usuário a clicar neles, como faria com botões realmente físicos. Enquanto os usuários podem fazer essa associação prontamente, é igualmente verdade que conseguiriam fazer a mesma associação com uma simples representação em 2D de um botão na tela. O esforço empreendido para projetar botões em 3D, entretanto, é provavelmente maior do que para botões em 2D.

O perigo de se tentar projetar interfaces gráficas com permissibilidade semelhante à de objetos físicos é inadvertidamente poder acarretar um *design* muito pobre. O uso de sombreadimento e de outras ilusões de percepção para o efeito de 3D pode acabar por indesejavelmente abarrotar a interface, tornando geralmente mais difícil encontrar certos objetos. Formas abstratas simples, em 2D (p. ex.: quadrados ou círculos), utilizadas para representar objetos como botões, podem, por sua vez, ser mais facilmente percebidas e reconhecidas na interface (veja Figura 1.11, na Página de Ilustrações 1).

Desde então, Norman tem tentado esclarecer seu argumento sobre a funcionalidade do conceito afirmando que há dois tipos de *affordance*: a percebida e a real. Diz-se que os objetos físicos dispõem de *affordances* reais, como a de segurar, que são perceptualmente óbvias e não precisam ser aprendidas. Por outro lado, as interfaces com o usuário, que estão na tela, são virtuais e não possuem esses tipos de *affordances* reais. Utilizando essa distinção, ele argumenta que não faz sentido tentar projetar *affordances* reais para interfaces – exceto quando se projetam dispositivos físicos, como painéis de controle, onde possibilidades como as de puxar e pressionar ajudam a orientar o usuário a saber o que fazer. As interfaces na tela são melhor conceitualizadas como *affordances percebidas*, que fundamentalmente são convenções aprendidas. Concluindo, Norman salienta que outros conceitos de *design* – convenções, *feedback* e restrições lógicas e culturais – são muito mais úteis para o *designer* que desenvolve interfaces gráficas do usuário.

### 1.6.1 Heurística e princípios de usabilidade

Os princípios do *design*, quando usados na prática, normalmente são chamados de heurística. Esse termo enfatiza que algo deve ser feito com esses princípios, quando aplicados a um dado problema. Em particular, precisam ser interpretados no contexto do *design*, utilizando-se experiências já realizadas sobre, por exemplo, como projetar *feedback* e sobre o que significa algo ser consistente.

Outra forma de orientação são os princípios de usabilidade. Um exemplo é “fale a língua do usuário”. Esses princípios são bastante parecidos com os princípios do *design*, exceto por terem a tendência de ser mais prescritivos. Além disso, enquanto os princípios do *design* tendem a ser utilizados principalmente para informar um *design*, os princípios de usabilidade são utilizados sobretudo como base para a avaliação de protótipos e sistemas existentes. Em particular, fornecem a estrutura para avaliação heurística (veja Capítulo 13). Os princípios de usabilidade também são chamados de heurística quando utilizados como parte de uma avaliação. A seguir, apresentamos os dez princípios fundamentais de usabilidade, desenvolvidos por Nielsen (2001) e seus colegas. Veja como alguns deles coincidem com os princípios do *design*.

1. *Visibilidade do status do sistema* – o sistema mantém os usuários sempre informados sobre o que está acontecendo, fornecendo um *feedback* adequado, dentro de um tempo razoável
2. *Compatibilidade do sistema com o mundo real* – o sistema fala a linguagem do usuário utilizando palavras, frases e conceitos familiares a ele, em vez de termos orientados ao sistema
3. *Controle do usuário e liberdade* – fornece maneiras de permitir que os usuários saiam facilmente dos lugares inesperados em que se encontram, utilizando “saídas de emergência” claramente identificadas
4. *Consistência e padrões* – evita fazer com que os usuários tenham que pensar se palavras, situações ou ações diferentes significam a mesma coisa
5. *Ajuda os usuários a reconhecer, diagnosticar e recuperar-se de erros* – utiliza linguagem simples para descrever a natureza do problema e sugere uma maneira de resolvê-lo

6. *Prevenção de erros* – onde possível, impede a ocorrência de erros
7. *Reconhecimento em vez de memorização* – tornar objetos, ações e opções visíveis
8. *Flexibilidade e eficiência de uso* – fornece aceleradores invisíveis aos usuários inexperientes, os quais, no entanto, permitem aos mais experientes realizar tarefas com mais rapidez
9. *Estética e design minimalista* – evita o uso de informações irrelevantes ou raramente necessárias
10. *Ajuda e documentação* – fornece informações que podem ser facilmente encontradas e ajuda mediante uma série de passos concretos que podem ser facilmente seguidos

#### ATIVIDADE 1.5

Um dos principais conceitos em *design* que Nielsen dogmatizou, especialmente para o *design* de *websites*, foi a simplicidade. Ele propõe que os *designers* analisem todos os elementos de seu projeto e os removam um a um. Caso ele funcione bem sem um desses elementos, elimine-o. Você considera ser esse um bom princípio do *design*? Se você possui seu próprio *website*, tente aplicar esse conceito e ver o que acontece. Em que ponto a interação é quebrada?

#### Comentário

A simplicidade constitui certamente um princípio importante no *design*. Muitos *designers* tentam abarrotar muita coisa dentro do espaço de uma tela, tornando impraticável encontrar o que se deseja. Remover elementos do projeto a fim de conferir o que pode ser descartado sem que afete a função geral do *website* pode ser algo bastante útil. Ícones, botões, quadros, linhas, gráficos, sombreado e textos desnecessários podem ser retirados, tornando o *website* mais limpo, interessante e fácil de navegar. No entanto, uma certa quantidade de gráficos, sombreados, cores e formatação podem tornar o *site* esteticamente agradável e proporcionar uma interação mais prazerosa. *Sites* muito simples, apresentando apenas textos e alguns *hiperlinks*, podem não ser tão atrativos e fazer com que alguns usuários não mais retornem a ele. A questão é encontrar o equilíbrio certo entre o apelo estético e a quantidade e o tipo certos de informação por página.

Os princípios do *design* e de usabilidade também foram operacionalizados em prescrições ainda mais específicas, denominadas regras. Trata-se de orientações que devem ser seguidas. Um exemplo é “em uma aplicação, sempre coloque o botão de fechar ou sair na parte inferior da primeira lista de opções de um menu em uma aplicação”.

## Tarefa

*Esta tarefa requer que você coloque em prática o que leu neste capítulo. Especificamente, o objetivo é permitir que você defina as metas decorrentes da experiência do usuário e as metas de usabilidade e que utilize princípios de design e usabilidade para avaliar a usabilidade de um produto interativo.*

Encontre um dispositivo pequeno (*handheld*) (p. ex.: controle remoto, computador, telefone celular) e examine como ele foi projetado, prestando atenção especialmente em como se espera que o usuário interaja com ele.

**QUADRO 1.5** Usabilidade usável: que termos empregar?

Os vários termos propostos para descrever os diferentes aspectos da usabilidade podem ser confusos. Geralmente são intercambiáveis e apresentam combinações diferentes. Algumas pessoas falam sobre princípios do *design* da usabilidade; outras, sobre heurística de usabilidade; e outras, sobre conceitos de *design*. A questão é entender como utilizar os diferentes níveis de orientações. As recomendações (*guidelines*) constituem um termo mais genérico, utilizado para referência a *todas as formas*

de orientação. As metas dizem respeito aos objetivos de usabilidade de alto nível do sistema (p. ex.: ser eficiente no uso). Os princípios implicam a orientação geral pretendida para informar o *design* e a avaliação de um sistema. As regras constituem-se em diretrizes de baixo nível, que se referem a uma determinada prescrição a ser seguida. A heurística é um termo geral, utilizado para se referir a princípios de *design* e usabilidade, quando aplicados a um problema de *design* em particular.

Conceito	Nível de orientação	Às vezes também chamado de	Como utilizar
Metas de usabilidade	Geral		Estabelecer critérios de usabilidade para avaliar a aceitabilidade de um sistema (p. ex.: “Quanto tempo leva para a realização de uma tarefa?”).
Metas decorrentes da experiência do usuário	Geral	Fatores de satisfação	Identificar os aspectos importantes da experiência do usuário (p. ex.: “Como se pode tornar o produto interativo divertido e agradável?”).
Princípios de <i>design</i>	Geral	Heurística, quando utilizados na prática. Conceitos de <i>design</i>	Como lembretes do que fornecer e do que evitar durante o <i>design</i> da interface (p. ex.: “Que tipo de <i>feedback</i> você vai fornecer na interface?”).
Princípios de usabilidade	Específica	Heurística, quando utilizados na prática	Avaliar a aceitabilidade das interfaces, utilizadas durante a avaliação heurística (p. ex.: “O sistema fornece saídas claramente indicadas?”).
Regras	Específica		Determinar se uma interface adere a uma regra específica, quando está sendo projetada e avaliada (p. ex.: “Sempre oferecer um botão <i>backward</i> e <i>forward</i> em um navegador?”).

- (a) Das suas primeiras impressões, escreva a que primeiro vem à sua mente, a respeito do que é bom ou ruim na maneira como o dispositivo funciona. Liste (i) sua funcionalidade e (ii) a variedade de tarefas que um usuário típico poderia querer realizar com o dispositivo. A funcionalidade é maior, igual ou menor do que a desejada pelo usuário?
- (b) Baseando-se na leitura deste capítulo e em outro material que você tenha encontrado, compile um conjunto de metas de usabilidade e de metas decorrentes da experiência com o usuário as quais você considera que seriam mais úteis na avaliação do dispositivo. Decida quais são os mais importantes e explique por quê.
- (c) Transforme as principais metas de usabilidade e decorrentes da experiência com o usuário que você selecionou em duas ou três perguntas. Use-as

**DILEMA**      Compensações de usabilidade

Um dos problemas de se aplicar mais de um dos princípios do *design* no *design de interação* é a possibilidade de surgirem desequilíbrios entre eles. Por exemplo, quanto mais você restringe os elementos de uma interface, menos visível se torna a informação. O mesmo também pode acontecer quando se tenta aplicar apenas um dos princípios. Por exemplo, vimos que quanto mais uma interface for projetada para ter *affordance*, por meio de semelhanças com os objetos do mundo real, mais abarrotada e difícil de utilizar ela será. Consistência é um outro princípio do *design* cuja aplicação pode ser problemática. Como vimos anteriormente, tentar projetar uma interface que seja consistente com uma coisa pode torná-la inconsistente com outra. Além disso, algumas vezes as interfaces inconsistentes são, na verdade, mais fáceis de usar do que as consistentes. Pode-se, entretanto, levar mais tempo para aprender a utilizá-la.

Grudin (1989) ilustra o dilema da consistência fazendo uma analogia com o lugar onde são guardadas as facas em uma casa. Elas apresentam uma variedade de formas – faca para manteiga, para carne, de mesa, para peixe. Um local fácil de colocá-las e conseqüentemente de encontrá-las é na gaveta abaixo da pia. Isso torna fácil para to-

dos localizá-las e segue uma regra consistente. Mas quanto às facas que não cabem ou que são muito afiadas para ser guardadas na gaveta, como as para carne ou a faca do pão? Elas são colocadas em um porta-facas de madeira. E quanto às melhores facas, guardadas para ocasiões especiais? Constituem elas uma outra exceção, pois, por segurança, são guardadas em uma outra peça da casa. E quanto às facas do tipo espátula, usadas para raspar tinta ou passar estuque (ficam na garagem), e aos canivetes (ficam no bolso ou na mochila)? Rapidamente a regra da consistência começa a falhar.

Grudin aponta para o fato de que, ao se aumentar o número de lugares onde as facas são mantidas, surge a inconsistência, que, por sua vez, aumenta o tempo necessário para *aprender* onde elas estão guardadas. Mesmo assim, a colocação de facas em locais diferentes geralmente torna mais fácil encontrá-las, pois estarão à mão, no contexto em que forem utilizadas, e também próximas a outros objetos utilizados em uma tarefa específica (p. ex.: todas as ferramentas são guardadas juntas em uma caixa na garagem). O mesmo acontece quando se projetam interfaces: a inconsistência pode fazer com que seja mais difícil de entender uma interface, mas a longo prazo pode facilitar o seu uso.

para avaliar se o dispositivo funciona bem (p. ex.: *Metas de usabilidade*. Que mecanismos específicos foram utilizados para garantir segurança? É fácil aprender a utilizá-los? *Metas decorrentes da experiência do usuário*: É divertido utilizar os mecanismos? O usuário frustra-se facilmente? Se sim, por quê?).

- (d) Repita (b) e (c) para os conceitos de *design* e princípios de usabilidade (escolha novamente um conjunto relevante).
- (e) Finalmente, discuta possíveis melhorias para a interface, baseando-se em sua avaliação de usabilidade.

## Resumo

Neste capítulo, falamos sobre o que é o *design* de interação e sobre como ele evoluiu. Examinamos brevemente sua formação e os vários processos envolvidos. Apontamos como a noção de usabilidade é fundamental para o *design* de interação, descrevendo com detalhe em que ela consiste e como é operacionalizada para a avaliação de adequação, eficácia e qualidade de produtos interativos. Uma variedade de princípios de *design* de alto nível foi também introduzida, mostrando como tais princípios constituem diferentes formas de orientação para o *design* de interação.

## Pontos principais

- O *design* de interação preocupa-se com o projeto de produtos interativos que apóiem os indivíduos em sua vida diária e em seu trabalho.
- O *design* de interação é multidisciplinar, envolvendo muitas contribuições de uma ampla variedade de disciplinas e áreas.
- O *design* de interação é hoje um grande negócio: muitas empresas o querem, mas poucas sabem como fazê-lo.
- Otimizar a interação entre usuários e produtos interativos requer que se levem em consideração vários fatores interdependentes, incluindo contexto de uso, tipo de tarefa e tipo de usuário.
- Os produtos interativos precisam ser projetados para atender às metas de usabilidade, como facilidade de uso e entendimento.
- As metas decorrentes da experiência do usuário estão preocupadas com criar sistemas que melhorem esta experiência, fazendo com que sejam mais agradáveis, divertidos, úteis, motivadores e satisfatórios.
- Os princípios de *design* e de usabilidade, como *feedback* e simplicidade, constituem heurísticas úteis para analisar e avaliar aspectos de um produto interativo.

## Leituras adicionais

*A seguir, recomendamos algumas leituras importantes. Uma lista mais abrangente de livros, artigos, sites, vídeos e outros materiais úteis encontra-se em nosso website.*

WINOGRAD, T. (1997) From computing machinery to interaction design. In P. Denning e R. Metcalfe (eds.) *Beyond Calculation: the Next Fifty Years of Computing*. New York: Springer-Verlag, 149-162. Terry Winograd apresenta uma visão geral de como o *design* de interação surgiu como uma nova área, explicando como não se encaixa em nenhuma outra área existente de *design* ou computação. Descreve os novos desafios e demandas da profissão de *designer*.

NORMAN, D. (1988) *The Design of Everyday Things*. New York: Doubleday, (especialmente o Capítulo 1). O texto de Norman é bastante acessível e agradável de ler. Explora bastante o *design* e a usabilidade de objetos de uso diário, como portas, torneiras e frigideiras. Esses exemplos contribuem muito para o *design* de interfaces. O CD-ROM Voyager (infelizmente não mais publicado), apresentando a coletânea de seus trabalhos, oferece vídeos e animações adicionais que ilustram de uma maneira leve muitos dos problemas, idéias e questões de *design* levantadas neste texto.

NORMAN, D. (1999) *ACM Interactions Magazine*, May/June, 38-42. *Affordances*, convenções e de-

*sign*. Pequena e instigante crítica aos princípios de *design*.

GRUDIN, J. (1990) The computer reaches out: the historical continuity of interface design. In *CHI'90 Proc.* 261-268.

GRUDIN, J. (1989) The case against user interface consistency. *Communications of the ACM*, 32(10), 1164-1173. Jonathan Grudin é um escritor prolífico, e muitos de seus primeiros trabalhos constituem-se em explicações instigantes e bem documentadas de questões centrais em IHC. O primeiro artigo trata de como o *design* da interface expandiu-se visando a abranger muito mais aspectos em sua história relativamente curta. O segundo artigo, considerado um clássico de seu tempo, discute por que o conceito de consistência – universalmente aceito, até agora, como um bom *design* de interface – foi, na verdade, altamente problemático.

*Interactions*, January/February, 2000, ACM. Este número especial traz vários pontos de vista, críticas e opiniões, apresentadas por um grande número de pesquisadores, *designers* e profissionais técnicos, sobre as realizações e o futuro da IHC.

A IDEO oferece um arquivo *on-line* ilustrado de vários produtos interativos já projetados (consulte [www.ideo.com](http://www.ideo.com)).

## ENTREVISTA com Gitta Salomon



Gitta Salomon é consultora em *design* de interação. Fundou a Swim Interaction Design Studio ([swimstudio.com](http://swimstudio.com)) em 1996, como uma empresa de consultoria criada para auxiliar clientes com o *design* de produtos inter-

terativos. Nos últimos anos, sua clientela têm se constituído de empresas iniciantes no desenvolvimento de produtos, tanto baseados na *web* como outros, que perceberam ser o *design* de interação importante para garantir o sucesso dos produtos – ainda que não soubessem como desenvolvê-lo. Geralmente, essas empresas entram em contato com a Swim, tendo seus produtos já parcialmente desenvolvidos, e pedem ajuda para o *design* de interação. A Swim presta consultoria para vários clientes, incluindo a Apple Computer, a Nike, a IBM, a DoubleClick, a Webex e a RioPort.

**YR: Qual é a sua abordagem para o *design* de interação?**

**GS:** Criei a minha própria definição: o *design* de interação é o *design* de produtos que se revelam com o tempo. Os usuários não vêem necessariamente toda a funcionalidade nos produtos interativos quando olham para eles pela primeira vez. Por exemplo, a primeira tela que você vê em um telefone celular não mostra tudo o que é possível fazer com ele. Quando você o usa, outras funcionalidades são reveladas. A mesma coisa acontece com uma aplicação para a *web* ou o Windows – à medida que vamos utilizando-os, encontramos-nos em estágios diferentes e de repente percebemos poder fazer coisas diferentes. Essa idéia de revelar-se com o tempo é possível porque há um microprocessador por trás do produto, assim como geralmente um *display* dinâmico. Acredito que essa definição caracteriza os tipos de produto com o qual trabalhamos – que são de muitos tipos, não somente para a *web*.

**YR: O que você diria que mudou no *design* de interação desde que você criou a Swim?**

**GS:** Não acho que as atividades que nós realizamos tenham mudado fundamentalmente, mas o período de tempo para o desenvolvimento de um produto é muito menor. E apa-

rentemente mais pessoas acham que precisam da assistência do *design* de interação. Isso realmente mudou. Muitos indivíduos não sabem necessariamente o que ele significa, mas estão nos chamando e dizendo “Nós precisamos dele”. De repente há muito dinheiro e foco em todos esses produtos virtuais e computacionais, exigindo que se pense o seu *design* de maneira diferente.

**YR: Em que tipos de projetos você estava trabalhando quando iniciou a Swim?**

**GS:** Eles eram menos centrados na *web*. Havia mais *design* de aplicações de *software* e poucas coisas do tipo *software/hardware*. Do último ano para cá o foco mudou quase que exclusivamente para as aplicações baseadas na *web*. No entanto, essas aplicações eram bastante semelhantes a aplicações de *software* – apenas com diferentes restrições de implementação. No momento, os produtos de *hardware/software* estão começando a surgir novamente – parece que os *information appliances* vão decolar. A natureza dos problemas que resolvemos não mudou muito; são a plataforma e as restrições a ela associadas que mudam.

**YR: Quais você diria serem os maiores desafios que você e outros consultores que trabalham com *design* de interação enfrentam hoje?**

**GS:** Um dos maiores desafios é lembrar que metade do que fazemos é trabalho de *design* e que a outra metade diz respeito à comunicação desse trabalho. Os clientes quase nunca preenchem essa lacuna: nós é que temos que fazer isso. Sempre temos de resolver como entregar o trabalho de modo a ser impactante. Somos aqueles que precisam assegurar que o cliente vai entendê-lo e saber o que fazer com ele. Essa parte do trabalho é, na maioria das vezes, a mais difícil. Significa que precisamos entender o que está acontecendo internamente com o cliente e decidir de que forma o que estamos entregando será eficiente. Em alguns casos você começa a ver que não há lugar para se conectar com o cliente. Considero esse um problema muito difícil. A maioria das pessoas não tem um processo de desenvolvimento de produtos. Simplesmente vão fazendo. Cabe a nós resolver como entrar em algo que, pode-se dizer, assemelha-se a um trem em movimento.



**Figura 1** *Showroom* do varejo da Steelcase Worklife New York – um dos projetos em que Gitta Salomon esteve envolvida. O objetivo era desenvolver um *showroom* interativo para a Steelcase, de Nova Iorque. O ambiente de vendas foi projetado de modo a fornecer várias ferramentas de vendas, incluindo um dispositivo interativo que permitia aos vendedores acessar vídeos com estudos de caso, podendo ser projetados em uma grande tela.

**YR:** E o que vocês utilizam quando tentam comunicar-se com os clientes? Uma combinação de conversa, reuniões e relatórios?

**GS:** Fazemos várias coisas. Geralmente passamos a eles um documento escrito, como um relatório ou uma crítica a respeito do seu produto. Algumas vezes oferecemos protótipos interativos em Director ou HTML; coisas que simulam como seria uma experiência com o produto. Nos materiais escritos, geralmente nomeio as coisas sobre as quais precisaremos conversar. Assim, temos pelo menos uma terminologia em comum para discutir as coisas. Teremos uma medida de nosso sucesso se eles começarem a usar as palavras que lhes passamos; isso significará que estamos de fato causando alguma influência. Muitas vezes damos ao cliente um diagrama de seu sistema, porque até então ninguém o visualizou. Servimos como visualizadores, estabelecendo uma seleção aleatória de conceitos definidos vagamente e dando a eles alguma forma. Fazemos um artefato que lhes permita dizer “Sim, é bem assim” ou “Não, não é isso, é isso...”. Sem ter algo em que se basear os clientes não poderão nem mesmo dizer “Não, não é isso o que eu quero”, pois não saberão nem se estávamos falando sobre a mesma

coisa. Muitas vezes utilizamos diagramas esquemáticos para representar o comportamento do sistema. Uma vez que os clientes estiverem com esses diagramas em mãos, poderão dizer “Ah, não. Precisamos de mais tudo isso aqui. Esquecemos de dizer a vocês”. Parece que ninguém está escrevendo listas completas de funcionalidade, especificações dos requisitos ou fazendo uma documentação completa. Isso significa que as idéias sobre o produto ficam na cabeça de alguém até que as tornamos tangíveis através da visualização.

**YR:** Então esse processo de comunicação é tão importante quanto as idéias?

**GS:** Acho que sim, muitas vezes.

**YR:** Como inicia o trabalho com o cliente?

**GS:** Para aqueles que já têm algo construído, acho que a melhor maneira de começar é com o cliente realizando uma demonstração de seu produto. Geralmente passamos o dia coletando informações. Além da demonstração, eles nos falam sobre seu mercado-alvo, sobre competidores e uma infinidade de outras coisas. Levamos então um período maior de tempo utilizando o produto e observando outras pessoas utilizarem-no, a fim de obtermos um qua-



dro da interação. Como a visão que o próprio cliente tem de seu produto é muito restrita, temos que manter um pé atrás com relação ao que eles inicialmente nos mostraram.

**YR:** *Aí vocês fazem anotações, testam e tentam reuni-las, ou o quê?*

**GS:** Utilizamos todos os tipos de coisas. Anotações, vídeo, papel e canetas. Quando revisamos os materiais, eu geralmente os testo e agrupo de alguma maneira temática. É muito complicado trabalhar com um *software* cujos elementos tenham sido colocados em uma estrutura coerente, mas de qualquer jeito. É fácil escrever uma lista de observações, mas queremos reunir uma estrutura e um *framework* maiores, o que leva semanas para ser construído. Precisamos de tempo para refletir e deixar baixar o pó do que fizemos e do que talvez tivéssemos que ter feito. Precisamos salientar as questões e colocá-las em algum tipo de ordem maior. Se você sempre opera em um nível baixo de detalhe, como, por exemplo, preocupando-se com e criticando o tamanho de um botão, você acaba resolvendo apenas questões menores; não chega nunca aos grandes problemas do *design* de interação do produto – os que deveriam ser resolvidos primeiro.

**YR:** *Se você recebe um produto ou um protótipo para avaliação e descobre que são mesmo ruins, o que você faz?*

**GS:** Bem, nunca me sinto impulsionada a dizer que algo seja realmente um erro. E, de qualquer sorte, essa talvez não seja a melhor estratégia, porque é a sua palavra contra a deles. Pelo contrário, acho que se deve tentar saber por que algo está errado ou apresenta falhas. Algumas vezes penso que agimos como advogados. Temos que reunir os fatos para descobrir o que há de errado com o produto. Temos que apre-

sentar um argumento convincente. Muitas vezes, considero o tipo de argumentação que fazemos muito parecido com o dos advogados.

**YR:** *Por fim, como você vê o movimento do design de interação para os próximos cinco anos? Enfrentando problemas apresentados pelas novas tecnologias emergentes? Ou você acha que haverá mais desafios, especialmente com a integração de software/hardware?*

**GS:** Acredito que com o surgimento de novas tecnologias haverá diferentes restrições. Não importa o que estivermos projetando, temos que entender as restrições da implementação. E penso que sim, diferentes coisas irão acontecer quando estivermos projetando mais produtos de *hardware/software*. Existem diferentes tipos de restrições de custos e diferentes tipos de interações que você pode realizar quando há *hardware* para fins específicos envolvidos. Enquanto projetar a interação para aplicações requer conhecimento em *design* visual, projetar *information appliances* ou outros produtos de *hardware* requer experiência com o *design* de produtos. Definitivamente, teremos novos desafios.

Espero que nos próximos anos as pessoas parem de buscar regras para o *design* de interação. Há, ultimamente, um grande esforço em tentar transformá-lo em uma ciência. Talvez isso tenha acontecido porque muitos estão tentando fazê-lo e não sabem por onde começar, pois não dispõem de muita experiência. Espero que as pessoas comecem a entender que o *design* de interação é uma disciplina de *design* – que existem algumas orientações e formas de se fazer um bom trabalho – e que criatividade combinada com pensamento analítico são necessários para se chegar a bons produtos. Será então uma atividade ainda mais interessante e excitante do que já é.