

**10.** *Signos de interface.* Examine uma aplicação que você não esteja acostumado a utilizar e procure identificar e classificar os diversos signos de interface que ela apresenta.

**11.** *Artefatos de metacomunicação.* Examine uma aplicação disponível em diferentes dispositivos de base computacional e estilos de interação (e.g., calendário num ambiente gráfico de janelas, na Web e em telefone celular). Com base nos elementos do espaço de design e nas perguntas propostas pela engenharia semiótica, caracterize a metacomunicação, implícita ou explícita, nesses diferentes artefatos. A partir dessa caracterização, analise a facilidade ou dificuldade de um usuário acostumado com uma dessas aplicações passar a utilizar uma outra, ou ainda ficar alternando entre elas.

# 4

## Processos de Design de IHC

### Objetivos do Capítulo

- Discutir as atividades envolvidas no design em geral e no design de um artefato computacional interativo em particular.
- Descrever os fenômenos de IHC sob diferentes perspectivas.
- Apresentar processos de design de IHC propostos na literatura: modelo de ciclo de vida simplificado, ciclo de vida em estrela, engenharia de usabilidade, design baseado em cenários, dirigido por objetivos e centrado na comunicação.
- Discutir formas de integrar atividades de IHC e engenharia de software, incluindo métodos ágeis de desenvolvimento de software.



Desde sua concepção e durante todo o seu desenvolvimento, um sistema interativo deve ter o propósito de apoiar os usuários a alcançarem seus objetivos. O projeto de um sistema interativo é um processo iterativo de análise, síntese e avaliação, no qual artefatos são coletados e produzidos visando não apenas à construção do sistema, mas também à promoção de uma boa experiência de uso desse sistema. Neste capítulo, primeiramente discutimos sobre a atividade de design e em seguida descrevemos alguns processos de design de IHC propostos na literatura.

#### 4.1 O Que é Design?

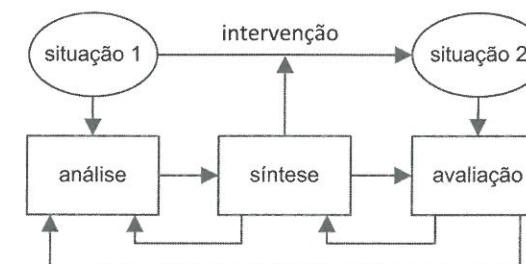
Em nosso cotidiano, com frequência lidamos com diversos artefatos. Artefatos são produtos artificiais, fruto da inteligência e do trabalho humano, construídos com um determinado propósito em mente. São artefatos, por exemplo: um copo, um pente, um sofá, um carro, uma música e uma receita de bolo. Um artefato não surge espontaneamente na natureza. Alguém decide sua função, forma, estrutura e qualidade, e o constrói com seu trabalho (Löwgren e Stolterman, 2004). A inserção de um artefato numa situação do cotidiano representa uma intervenção sobre ela, em alguma medida, e a própria situação influencia a forma como o artefato é utilizado. Por exemplo, ter ou não uma geladeira em casa modifica significativamente a forma como os alimentos são conservados. Por sua vez, a conservação de alimentos influencia a escolha e a quantidade de produtos comprados, bem como a frequência de compra.

A introdução de um artefato pode trazer consequências positivas e negativas para a situação atual. Por exemplo, adquirir uma bicicleta pode significar: mudar o meio de transporte utilizado para várias atividades, como ir ao trabalho, à padaria ou fazer um passeio; a economia de dinheiro gasto em outros meios de transporte; a prática de esportes para uma vida mais saudável; e ajuda na preservação do meio ambiente. Nesse sentido, adquirir uma bicicleta é uma intervenção positiva. No entanto, o dono da bicicleta agora precisa reservar um local em sua casa para armazená-la, e comprar um capacete e outros equipamentos para utilizar a bicicleta com segurança. Essas podem ser consideradas consequências negativas da intervenção realizada.

Quando analisamos uma situação, considerando as pessoas, artefatos, processos, demais elementos e relações entre eles, podemos encontrar problemas, características desagradáveis ou algo que podemos melhorar.

É comum tomarmos atitudes para resolver os problemas, diminuir as características desagradáveis e melhorar o que mais for possível. Essas atitudes envolvem atividades de design, conscientes ou inconscientes, realizadas com um cuidado maior ou menor. Em termos bem gerais (Lawson, 2006; Löwgren e Stolterman, 2004), podemos caracterizar a atividade de **design** como sendo (Figura 4.1):

- a **análise** da situação atual: estudar e interpretar a situação atual;
- a **síntese** de uma intervenção: planejar e executar uma intervenção na situação atual;
- a **avaliação** da nova situação: verificar o efeito da intervenção, comparando a situação analisada anteriormente com a nova situação, atingida após a intervenção.



**Figura 4.1** Atividades de design envolvidas na intervenção para transformar a situação atual (situação 1) em uma situação desejada (situação 2).

Na **análise da situação atual**, buscamos conhecer os elementos envolvidos e as relações entre eles. Dentre os diversos elementos, geralmente são analisados: pessoas, artefatos, e processos. Como resultado, obtemos uma interpretação da realidade estudada, através de um enquadramento e um recorte particular dela. A complexidade e a dinâmica da realidade fazem com que alguns (aspectos dos) elementos e suas relações sejam enquadrados e outros não. O foco da análise da situação atual depende de vários fatores, como, por exemplo, os assuntos sendo tratados (o domínio), os objetivos das pessoas envolvidas (usuários e demais *stakeholders*), tempo, orçamento e mão-de-obra disponíveis, e até mesmo a filosofia de trabalho. Diferentes focos de análise contribuem para diferentes interpretações da situação atual. Por exemplo, se o objetivo de uma organização for aumentar a produtividade dos seus funcionários, a análise da situação atual pode concentrar esforços na investigação de atividades inefficientes. De outra forma, se o objetivo da organização for compreender os motivos pelos quais seus funcionários não utilizam como esperado os sistemas computacionais oferecidos, a análise da situação atual pode concentrar esforços em compreender a opinião dos funcionários sobre esses sistemas.

A análise da situação atual é frequentemente denominada de análise do *problema*. Entretanto, a análise nem sempre aborda uma situação problemática. Mesmo que a situação atual seja satisfatória, pode surgir uma oportunidade, por conta de uma nova tecnologia, por exemplo, de atingirmos uma situação ainda mais desejável. Nesse caso, a análise da situação atual é importante para identificar as condições em

que uma nova tecnologia pode ser empregada para melhorar o que já é satisfatório. O “problema” a ser resolvido é *encontrar uma boa forma de melhorar* uma ou mais características da situação atual. Em outras palavras, resolver um problema de design significa responder a pergunta: “Como melhorar a situação atual?”

Quando se trata de sistemas computacionais, é comum investigarmos todos os elementos envolvidos durante o uso: os usuários com suas características, necessidades e preferências; as atividades e os objetivos em questão, considerando os artefatos e sistemas computacionais utilizados; e o contexto físico, social e cultural de uso ao longo do tempo (Hackos e Redish, 1998; Sharp *et al.*, 2007). Além desses elementos diretamente envolvidos no uso atual desses artefatos, também é preciso conhecer e articular os interesses das pessoas indiretamente envolvidas, como aquela que paga pelo sistema, geralmente conhecida como cliente, e aquela que constrói o sistema, normalmente chamada de desenvolvedor.

A análise da situação atual aponta as necessidades e oportunidades de melhoria para as quais será projetada uma intervenção. Essas razões para a intervenção costumam ser representadas por metas de design. Em IHC, metas de design referem-se não apenas aos objetivos dos usuários que precisamos ou desejamos apoiar, mas também aos critérios de qualidade de uso descritos na Seção 2.2. Por exemplo, durante a análise de uma situação é possível perceber que os usuários gastam muito tempo processando informações que um sistema computacional poderia fazer mais rapidamente. Nesse caso, uma meta de design poderia ser: aumentar a eficiência das atividades (ou tarefas) do usuário, que é um dos fatores de usabilidade. Já a análise de uma outra situação identifica que vários usuários encontram dificuldades para usar sistemas semelhantes porque não compreendem como funcionam. Nesse caso, uma meta de design poderia ser: comunicar adequadamente através da interface a visão do designer sobre as operações que o usuário pode realizar com o sistema, ou seja, aumentar a comunicabilidade de um artefato computacional interativo. Os Capítulos 5 e 6 apresentam técnicas e representações utilizadas na atividade de análise.

A diferença entre a situação atual e uma situação desejada é a motivação principal para projetarmos e sintetizarmos uma **intervenção**. Frequentemente, uma intervenção é denominada de *solução*, pois responde a pergunta que define um problema a ser resolvido: “Como melhorar esta situação?”. Em alguns casos, uma intervenção adequada é inserir na situação atual um novo sistema interativo ou uma nova versão de um sistema existente, com o intuito de transformá-la em uma situação desejada. Em outros casos, pode ser necessária apenas uma mudança em processos, sem alteração dos sistemas utilizados.



Quando a intervenção envolve o desenvolvimento de sistemas interativos, ela deve articular os interesses dos *stakeholders* com: (1) o conhecimento adquirido na análise da situação atual; (2) o conhecimento sobre intervenções bem e mal avaliadas em casos semelhantes; e (3) o conhecimento sobre as possibilidades e limitações das tecnologias disponíveis. Em particular, é fundamental tomar o devido cuidado com a experiência que pretendemos oferecer aos usuários durante o uso do sistema projetado. Dessa forma, o projeto de um sistema interativo deve definir uma solução de IHC com alta qualidade de uso para impactar a situação atual e a vida dos usuários conforme pretendido. Como visto na Seção 2.1, uma solução de IHC comprehende tanto a *interface* com usuário como os *processos de interação* concebidos para apoiarem-no a alcançar seus objetivos. Os Capítulos 7 e 8 discutem a atividade de design (síntese) de uma solução de IHC, bem como as representações, diretrizes e padrões utilizados.

Uma vez definida uma intervenção, é preciso avaliar se ela modifica a situação atual da forma desejada. A **avaliação de uma intervenção** pode ocorrer em vários pontos do processo de desenvolvimento:

- durante a concepção e o desenvolvimento da intervenção, para tentar prever seus possíveis impactos na situação atual (por exemplo, inspecionando as telas produzidas durante o projeto da interface com usuário);
- logo antes da introdução da intervenção, para identificar consequências negativas ou problemas que possam ser evitados (por exemplo, fazendo testes com usuários e produzindo material de treinamento a partir dos seus resultados); ou
- depois da intervenção ter sido aplicada, para verificar os impactos ocorridos (por exemplo, avaliando como os usuários se apropriaram do sistema interativo desenvolvido e quais mudanças ocorreram na sua prática de trabalho).

Quando a intervenção envolve um sistema interativo, existem vários aspectos a serem avaliados, alguns relacionados com a construção do sistema, como a facilidade de manutenção e robustez (Pressman, 2002), outros com o seu uso, como a usabilidade e acessibilidade. Em IHC, os esforços de avaliação se concentram na experiência vivenciada pelos usuários durante a interação com a interface de um sistema interativo, ou seja, durante o uso do sistema. Uma avaliação de IHC deve verificar se a interação e a interface atendem aos critérios de qualidade de uso definidos como prioritários pela análise da situação atual. A avaliação de IHC pode ser feita ao longo do processo de design ou depois do produto pronto. Sempre que possível, devemos avaliar a qualidade de uso desde o início do processo de design dos sistemas interativos, pois o custo de correção de eventuais problemas será menor. Os Capítulos 9 e 10 apresentam conceitos e métodos de avaliação de IHC em detalhes.

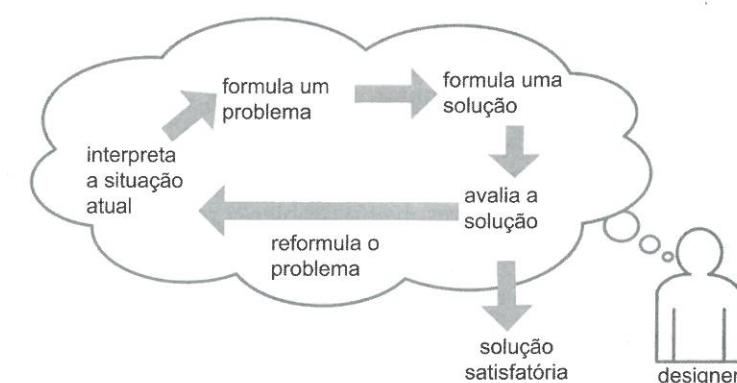


## 4.2 Perspectivas de Design

Existem diferentes interpretações para a atividade de design. Simon (1981) interpreta o design sob uma perspectiva de **racionalismo técnico** (*technical rationality*). Nessa perspectiva, o designer pressupõe que, para um determinado problema, há soluções conhecidas ou métodos bem definidos e precisos para gerá-las. Isso significa que as soluções esperadas certamente serão produzidas se esses métodos forem seguidos. Essas soluções e métodos empregam leis, princípios, normas e valores, geralmente estabelecidos pela natureza, com base em disciplinas de ciências naturais e exatas, como a Física, a Química e a Matemática. Desse modo, a atividade de design se resume a enquadrar uma situação num tipo geral de problema cuja forma de solução seja conhecida. Nessa perspectiva, não existe espaço para o designer questionar ou mudar as verdades estabelecidas pelas relações de causa e consequência (“se eu fizer isto vai acontecer aquilo”).

Em oposição à perspectiva de racionalismo técnico, Schön (1983) propôs uma perspectiva de **reflexão em ação** (*reflection-in-action*) para a atividade de design, como visto na Seção 3.8.3. Nessa perspectiva, uma situação do cotidiano pode estar associada a um problema, que é considerado único. Cada caso é diferente do outro. Consequentemente, o processo de design e a solução encontrada também são únicos. Desse modo, o designer não “está procurando descobrir dicas [da situação atual que apontam] para uma solução padrão. Em vez disso, procura descobrir as características particulares de sua situação tida como problemática, e a partir dessa descoberta gradual, projeta uma intervenção” (Schön, 1983 p.129). Portanto, o processo de concepção de uma solução —o design— é semelhante ao processo de pesquisa científica, no qual a construção de uma hipótese (a partir de uma interpretação da situação atual), a experimentação (proposta de intervenções) e a avaliação são fundamentais.

Na perspectiva de reflexão em ação (Figura 4.2), o designer inicia seu trabalho identificando e interpretando os elementos envolvidos na situação atual, os interesses dos envolvidos direta e indiretamente com o sistema (*stakeholders*) e as possibilidades e limitações das tecnologias disponíveis. Isso lhe permite formular um problema único a ser resolvido. Sendo assim, ele pode resolver tal problema único explorando livremente sua criatividade e utilizar ou não soluções e métodos de resolução de problemas conhecidos, conforme julgar apropriado.



**Figura 4.2** Processo de reflexão em ação.

Durante a concepção de uma solução, é muito comum o designer explorar diferentes ideias para poder compará-las e aproveitar o melhor de cada uma delas (Buxton, 2007). As alternativas de solução geradas durante a exploração de ideias costumam ser representadas por algum desenho, maquete ou modelo. A representação de uma alternativa de solução tem um papel de manifestar (parte de) as ideias do designer para outras pessoas envolvidas no projeto. Dessa forma, o cliente, o usuário e o desenvolvedor podem compreender e opinar sobre o projeto do sistema, antes mesmo de ele ser construído e inserido no cotidiano das pessoas.

Segundo Schön (1983), a manifestação das ideias do designer em alguma representação possui outro benefício importante. Enquanto o designer expressa suas ideias, ele acaba “conversando” com a representação, um fenômeno por ele denominado **conversa com os materiais** (*conversation with materials*). Ora o designer “fala” com a representação, expressando suas ideias para a solução sendo concebida, como, por exemplo, “E se eu definir isso deste jeito?”, “Posso utilizar essa mesma ideia em outro lugar” ou “O que acontece se eu modificar isso aqui?”, ora a representação “fala” com o designer no momento em que ele percebe o que concebeu e o avalia, se questionando, por exemplo, sobre “O que é isso que eu representei?” e descobrindo que “Eu não entendi isso direito”, “Isso é diferente do que eu pensei que seria, mas é interessante!”, “Isso está desajeitado, isso não”, “Aquilo não parece bom para mim” ou “Isso não funciona” (Schön e Bennett, 1996).

Depois de expressar suas ideias em alguma representação, o designer tem uma condição melhor de avaliá-las para verificar se a solução proposta satisfaz seus objetivos. Caso não encontre uma solução satisfatória, o designer critica não apenas as soluções que se apresentam, mas também sua própria formulação do problema, e experimenta reformulá-lo. Essa reformulação pode modificar os elementos envolvidos na situação analisada e seus significados de uma maneira que não havia sido prevista

pelo designer. Dessa forma, o designer precisa continuar descobrindo e refletindo sobre quais são as consequências dessa reformulação para a próxima tentativa de resolver o problema. Ele pode se perguntar, por exemplo: “As consequências dessa reformulação são desejáveis?”, “Que novos (potenciais) problemas foram criados?”, “O que melhorou com essa nova reformulação?”, e assim por diante. Nesse processo reflexivo, “o esforço do designer para resolver o problema reformulado produz novas descobertas que estimulam novas reflexões em ação” (Schön, 1983, p. 132), até que uma ação do designer dê origem a uma solução satisfatória.

Resumindo, a reflexão em ação ocorre enquanto o designer conversa com (age sobre) a representação, refletindo, avaliando e aprendendo sobre o que está fazendo enquanto o faz, de forma que essas reflexões influenciem ações futuras em direção à concepção da solução. Segundo o próprio Schön:

Refletir em ação é “interagir com o modelo, obter resultados surpreendentes, tentar interpretá-los, e então inventar novas estratégias de ação com base nas novas interpretações” (Schön e Bennett, 1996, p.181).

Algumas pessoas, na tentativa de “resolver um problema” em IHC, buscam aplicar diretamente padrões de interface no projeto de sistemas interativos, numa perspectiva de racionalismo técnico. Entretanto, esses padrões não fornecem soluções prontas para um problema genérico, mas sim um repertório de soluções conhecidas para problemas específicos e contextualizados. Como descrito na Seção 8.3, padrões de design de interface devem ser cuidadosamente analisados e adaptados a cada projeto (Tidwell, 2005; Borchers, 2001). Em outras palavras, devem ser utilizados numa perspectiva reflexiva alinhada à epistemologia da prática de Schön (1983).

Conceber uma solução adequada ao problema não é uma tarefa simples, e geralmente requer uma equipe multidisciplinar de design. Ela exige do designer as seguintes habilidades e conhecimentos (Löwgren e Stolterman, 2004, p.45):

- criatividade e capacidade de análise para criar e modelar ideias;
- capacidade de crítica e julgamento para decidir;
- capacidade de comunicação e negociação para trabalhar com clientes, usuários e desenvolvedores;
- conhecimento sobre as tecnologias disponíveis para projetar qualidades estruturais e funcionais;
- conhecimento sobre valores e ideais dos envolvidos para projetar qualidades éticas;
- capacidade de apreciar e compor coisas agradáveis aos sentidos para projetar qualidades estéticas.



### 4.3 Processos de Design de IHC

Como visto na seção anterior, o design é um processo que envolve as seguintes atividades básicas: a análise da situação atual (identificação do problema), a síntese de uma intervenção e a avaliação dessa intervenção projetada ou já aplicada à situação atual. Cada processo de design detalha essas atividades básicas de uma forma particular, definindo: como executar cada atividade; a sequência em que elas devem ser executadas; quais atividades podem se repetir, e por quais motivos; e os artefatos consumidos e produzidos em cada uma delas.

Uma característica básica dos processos de design de IHC é a execução das atividades de forma **iterativa**, permitindo refinamentos sucessivos da análise da situação atual e da proposta de intervenção. Dessa forma, o designer tem boas oportunidades de aprender mais e melhor tanto sobre o problema a ser resolvido quanto sobre a solução sendo concebida.

Geralmente os processos de design de IHC começam analisando a situação atual. Quando o designer considera ter adquirido conhecimento suficiente sobre essa situação e identificado as necessidades e oportunidades de melhoria, ele prossegue seu trabalho sintetizando (concebendo, modelando e construindo) uma intervenção. Enquanto ele projeta uma intervenção, pode sentir necessidade de *conhecer mais* ou *rever uma interpretação anterior* sobre a sua análise da situação atual. Assim, ele volta à atividade anterior para ampliar, refinar ou reformular sua interpretação, numa nova iteração do processo de design. Com a revisão da análise, o designer amplia, refina ou reformula a sua proposta de intervenção.

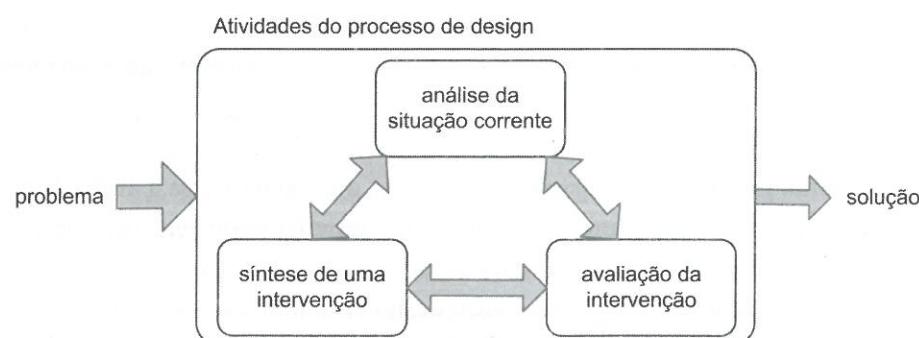
Tendo uma proposta de intervenção em mãos, o designer passa a avaliá-la para julgar se ela é satisfatória. Durante essa avaliação, o designer pode perceber que ainda precisa rever sua análise ou sua proposta de intervenção. Esse processo iterativo se repete quantas vezes forem necessárias, até o designer obter uma intervenção satisfatória. O início do processo de design tende a ter uma iteração maior entre a análise da situação atual e a síntese da intervenção, enquanto o final do processo tende a ter uma iteração mais acentuada entre a síntese e a avaliação da intervenção.

Mesmo executando essas três atividades básicas do processo de design de forma iterativa, é possível empregar quantidade de tempo e esforço diferente em cada uma delas. Por exemplo, podemos ter um design dirigido pelo problema ou dirigido pela solução. O *design dirigido pelo problema* despende mais tempo analisando a situação atual, as necessidades e as oportunidades de melhoria (o problema), e menos tempo explorando possíveis intervenções (as soluções). O *design dirigido pela solução* faz exatamente o contrário, emprega pouco tempo analisando a situação atual, e mais tempo explorando possíveis intervenções. Kruger e Cross (2006) realizaram um ex-



perimento com designers experientes desempenhando a mesma tarefa para comparar as estratégias de design seguidas por eles (dirigidas pelo problema ou pela solução) e a qualidade das soluções obtidas. Eles concluíram que a qualidade geral das soluções não estava diretamente relacionada com a estratégia seguida. Observaram ainda que os designers que atuaram dirigidos pela solução produziram, em média, resultados mais criativos, mas com menos preocupação com os aspectos estéticos, ergonômicos, técnicos e comerciais. A criatividade parece ter sido favorecida pela exploração de mais ideias para possíveis soluções.

Alguns processos de design de IHC prescrevem qual deve ser a primeira atividade a ser realizada, bem como a sequência de transições entre elas. Lawson (2006) questiona essa sequência estrita de atividades de design. Para ele, é possível iniciar o processo de design em qualquer atividade e realizar qualquer transição durante o processo quantas vezes forem necessárias (Figura 4.3). Dessa forma, cabe ao designer decidir qual será a primeira atividade a ser executada e as transições entre atividades que ele vai realizar. O que realmente importa, segundo Lawson, é partirmos de um problema (as necessidades e oportunidades de melhoria na situação atual), realizarmos o processo de design (atividades de análise, síntese e avaliação) e no final chegarmos a uma solução (uma proposta de intervenção).



**Figura 4.3** Sequência genérica de atividades durante o processo de design (adaptado de Lawson, 2006).

Os processos de design de IHC buscam atender e servir em primeiro lugar aos usuários e aos demais envolvidos (*stakeholders*), e não às tecnologias. Boa parte desses processos é **centrada no usuário**, isto é, seguem estes princípios (Gould e Lewis, 1985, p. 300):

- **foco no usuário:** o designer deve projetar a interação e a interface de um sistema interativo para atender às necessidades dos usuários e ajudá-los a alcançarem seus objetivos. Sendo assim, o designer deve estudar quem serão os usuários do sistema, seus objetivos, suas características físicas (capacida-

de de movimentos, visão, audição etc.), cognitivas e comportamentais, sua formação educacional (capacidade de leitura e escrita, conhecimentos adquiridos etc.), e o que eles costumam fazer para alcançar seus objetivos;

- **métricas observáveis:** o processo de design deve permitir a realização de experimentos (estudos empíricos) em que representantes dos usuários usem simulações ou protótipos do sistema para realizarem suas atividades e alcançarem seus objetivos. Durante o experimento, a performance e as reações dos usuários devem ser observadas, registradas e analisadas;
- **design iterativo:** quando problemas forem encontrados durante os experimentos com usuários, eles deverão ser corrigidos. Isso significa que as atividades do processo de design devem ser iterativas, ou seja, o ciclo de projeto, avaliação com medições empíricas e reprojeto deve se repetir quantas vezes forem necessárias.

Os processos de design de IHC destacam a importância de envolver os usuários durante suas atividades para dar-lhes oportunidade de participar, direta ou indiretamente, nas decisões tomadas. Quanto mais cedo os usuários forem envolvidos no processo de design, mais cedo será possível aprender sobre suas necessidades e assim influenciar positivamente a síntese da solução, bem como identificar e corrigir eventuais problemas. Isso nos permite desenvolver um sistema interativo mais interessante para os usuários e com maior qualidade de uso, porque temos acesso às interpretações e opiniões dos usuários sobre o resultado do design.

Como discutido no Capítulo 1, uma equipe multidisciplinar contribui para o design de IHC. Cada profissional observa e interpreta a situação atual de um ponto de vista particular, que contribui para enriquecer a identificação das necessidades e oportunidades de melhoria (a identificação e análise do problema). Uma equipe de pessoas formadas em diferentes áreas de conhecimento favorece a síntese (projeto) de uma intervenção, pois facilita o surgimento e a exploração de ideias diversas. Além disso, uma equipe multidisciplinar também permite uma avaliação mais rica e diferenciada das intervenções (soluções) propostas. Sempre que houver recursos disponíveis, devemos investir em uma equipe multidisciplinar de design de IHC.

Preece, Sharp e Rogers (Preece *et al.*, 2002; Sharp *et al.*, 2007) organizaram as atividades de design de IHC em um modelo de processo de design simples, ilustrado na Figura 4.4. Esse processo de design de IHC destaca a importância do design centrado no usuário, de avaliações da proposta de solução usando versões interativas e da iteração entre as atividades.

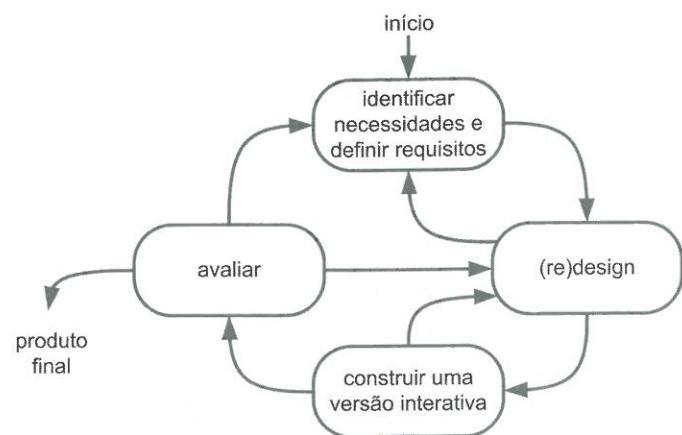


Figura 4.4 Modelo simples de processo de design de IHC (adaptado de Sharp *et al.*, 2007).

Considerando a sequência genérica de atividades de design identificada por Lawson (2006), observamos que Preece e coautoras segmentam a atividade de síntese em: design (ou redesign) conceitual e na construção de uma versão interativa.

Durante o (re)design da interação e da interface, o designer explora diferentes ideias em alternativas de design para elaborar uma solução adequada às necessidades e aos requisitos definidos na atividade de análise. O resultado dessa atividade de design pode ser registrado em descrições textuais da interação (cenários), esboços de interface (desenhos de tela) ou em qualquer modelo ou representação da interface e da interação usuário-sistema.

Para melhor avaliar o design resultante, o designer constrói **versões interativas** das propostas de solução que simulem o funcionamento da interface e deixem clara a interação projetada. Isso facilita a participação dos usuários durante a avaliação de IHC.

Assim como a maior parte dos processos de design, esse modelo de processo também é bastante iterativo. Cada atividade pode revelar a necessidade de retornar a uma atividade anterior para ampliar, refinar ou retificar algum entendimento ou artefato produzido. A iteração entre as atividades ocorre quantas vezes forem necessárias, limitada apenas pelo orçamento, tempo e recursos disponíveis. Idealmente o processo de design é concluído com uma avaliação de que a solução de IHC atende às necessidades e aos requisitos identificados. O produto final é uma especificação da interação e da interface com usuário que servirá de insumo nas fases seguintes de desenvolvimento do sistema iterativo.

Existem várias outras propostas de processos de design de IHC. Cada uma privilegia uma forma de pensar, uma sequência de atividades ou o emprego de certos



artefatos. Dentre as propostas existentes este livro apresenta brevemente o ciclo de vida em estrela (Hix e Hartson, 1993), dois ciclos de vida para Engenharia de Usabilidade (Nielsen, 1993; Mayhew, 1999), o design contextual (Beyer e Holtzblatt, 1998), o design baseado em cenários (Rosson e Carroll, 2002), o design dirigido por objetivos (Cooper *et al.*, 2007) e o design centrado na comunicação (Barbosa *et al.*, 2004).

#### 4.3.1 Ciclo de Vida em Estrela

O ciclo de vida em estrela foi desenvolvido por Hix e Hartson no início da década de 1990 (Hix e Hartson, 1993) e foi um dos primeiros ciclos de vida voltados para IHC amplamente difundidos. Esse processo é composto por seis atividades, conforme ilustrado na Figura 4.5.

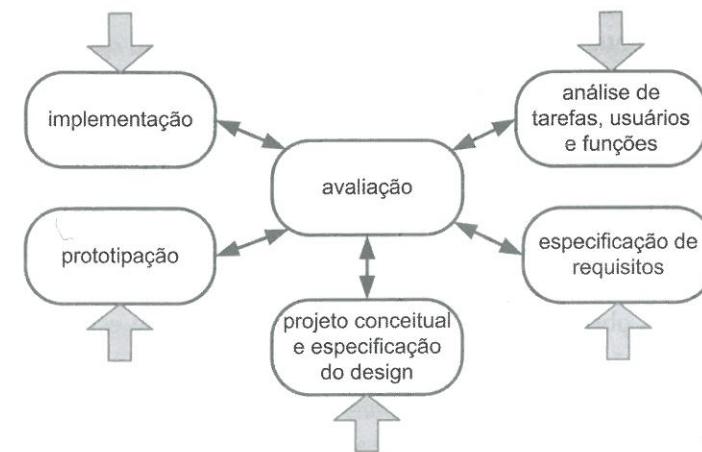


Figura 4.5 Ciclo de vida em estrela.

A **análise de tarefas, de usuário e funções** é a atividade responsável pelo aprendizado da situação atual e pelo levantamento das necessidades e oportunidades de melhoria. A atividade de **especificação de requisitos** de IHC consolida uma interpretação da análise, definindo os problemas que devem ser resolvidos com o projeto de uma solução de IHC.

A atividade geral de síntese é segmentada em três atividades: **projeto conceitual e especificação do design**, na qual a solução de IHC é concebida; **prototipação**, na qual versões interativas das propostas de solução são elaboradas para serem avaliadas; e **implementação**, na qual o sistema interativo final é desenvolvido. Essa última atividade está fora do escopo deste livro, pois já é abordada extensamente na literatura de Engenharia de Software.

A atividade de **avaliação** aparece no modelo como central, e é de fato desdobrada na avaliação dos resultados de cada uma das demais atividades. A avaliação deve



verificar se os dados coletados na atividade de análise e os requisitos especificados estão de acordo com a realidade e atendem às necessidades dos usuários. Deve também detectar problemas de usabilidade (e dos demais critérios de qualidade de uso discutidos na Seção 2.2) nas representações de design, nos protótipos e no sistema final. Hix e Hartson ressaltam que é importante avaliar o que foi produzido desde o início, enquanto ainda há tempo para corrigir os erros com um custo reduzido.

No ciclo de vida em estrela, cabe ao designer decidir qual atividade deve ser realizada primeiro, dependendo do que estiver disponível quando iniciar o processo. Por exemplo, se a intenção for projetar uma nova versão do sistema, o designer pode começar o projeto da nova versão pela avaliação da versão atual. Em outro caso, pode ser necessário implementar o mesmo sistema em outra plataforma semelhante, como outro sistema operacional, por exemplo. Sendo assim, o designer pode optar por começar pela implementação do sistema, aproveitando o projeto que havia sido feito para a plataforma anterior. Para projetar um novo sistema, é comum começar pela análise de tarefas, usuários e funcional e seguir no sentido horário pelas atividades na Figura 4.5.

Assim como na sequência genérica de atividades de design identificada por Lawson (2006), o ciclo de vida em estrela também é iterativo e não prescreve a sequência das atividades. A única exigência aqui é que, após concluir cada atividade, o designer avalie os resultados obtidos para verificar se ele encontrou ou está no caminho de encontrar uma solução satisfatória. Todas as atividades do ciclo de vida em estrela estão interligadas pela atividade de avaliação, ou seja, sempre é preciso passar por uma avaliação ao concluir uma atividade e antes de iniciar outra.

### 4.3.2 Engenharia de Usabilidade de Nielsen

Jakob Nielsen (1993) definiu **engenharia de usabilidade** como um conjunto de atividades que devem ocorrer durante todo o ciclo de vida do produto, ressaltando que muitas delas ocorrem nos estágios iniciais do projeto, antes que a interface com usuário em si seja projetada. Nielsen propõe o seguinte conjunto de atividades em seu ciclo de vida:

1. Conheça seu usuário
2. Realize uma análise competitiva
3. Defina as metas de usabilidade
4. Faça designs paralelos
5. Adote o design participativo
6. Faça o design coordenado da interface como um todo

7. Aplique diretrizes e análise heurística
8. Faça protótipos
9. Realize testes empíricos
10. Pratique design iterativo

Nesse ciclo de vida, o primeiro passo consiste em **estudar os usuários e os usos** pretendidos do produto. Segundo Nielsen, as características de usuários individuais e a variabilidade nas tarefas são os fatores de maior impacto na usabilidade. Ele utiliza o termo “usuário” de forma ampla, incluindo todos aqueles cujo trabalho é afetado de algum modo pelo produto, isto é, usuários diretos e demais *stakeholders*.

Essa atividade envolve conhecer as características individuais dos usuários e do seu ambiente físico e social de trabalho (veja Seção 5.2), suas atividades (veja Seção 6.4) e as formas como lidam com circunstâncias excepcionais e emergenciais. Nielsen sugere procurar usuários especialmente eficientes e que desenvolveram suas próprias estratégias para *contornar* as limitações dos sistemas existentes. Durante esse estudo, é fundamental ir além das atividades dos usuários tal como são realizadas atualmente, buscando identificar a *razão funcional* subjacente a cada atividade: o que realmente precisa ser feito, ou seja, os objetivos finais dos usuários e demais *stakeholders*.

Nielsen alerta para o fato de que os usuários não serão os mesmos após a introdução do sistema. O sistema modifica os usuários, e à medida que isso ocorre eles usarão o sistema de novas formas, num fenômeno denominado por Carroll e Rosson (1991) “*coevolução de tarefas e artefatos*”. Embora seja impossível prever todas as mudanças, um design mais flexível, adaptável ou extensível tem mais chances de apoiar esses novos usos (de Souza, 2005a; de Souza e Barbosa, 2005).

A **análise competitiva** consiste em examinar produtos com funcionalidades semelhantes ou complementares. Como esses produtos já estão prontos, podem ser testados com mais facilidade e realismo do que protótipos. Eles podem ser inspecionados e testados visando avaliar tanto as funcionalidades que apoiam como as questões de IHC tidas como relevantes para o projeto. Como resultado, o designer pode obter um conjunto de informações sobre o que funciona e o que não funciona naquele domínio, o que pode ser aperfeiçoado, e por quê. Nielsen ressalta que a análise competitiva envolve não apenas sistemas computacionais interativos, mas também qualquer outra forma de atividade de usuário com objetivos semelhantes. Por exemplo, antes de projetar uma agenda eletrônica, devemos investigar não apenas as agendas eletrônicas disponíveis, mas também a forma como as pessoas utilizam agendas em papel.

A **definição das metas de usabilidade** envolve definir os fatores de qualidade de uso que devem ser priorizados no projeto, como serão avaliados ao longo do proce-

so de design, e quais faixas de valores são inaceitáveis, aceitáveis e ideais para cada indicador de interesse. Com frequência, essa priorização se baseia nos indicadores atuais de desempenho dos usuários ao utilizarem o sistema. O Exemplo 4.1 ilustra uma definição de metas de usabilidade para um sistema de busca de um livro num quiosque de livraria.

#### **Exemplo 4.1 – Metas de usabilidade para um sistema de busca de livros em uma livraria**

Considere um sistema de quiosque de livraria pouco utilizado, em que 50% dos usuários desistem de fazer uma busca por um livro antes de concluir-la. Podemos estabelecer como metas que mais pessoas utilizem o sistema e que somente 30% dos usuários abandonem a tarefa de busca. As metas de usabilidade para esse projeto podem ser: aumentar a *facilidade de aprendizado* e a *eficiência do sistema*. Os indicadores correspondentes poderiam ser: número de usuários que acessam o sistema em diferentes dias da semana; proporção de usuários que completam/abandonam a tarefa de busca; tempo que cada usuário leva para concluir a tarefa com sucesso; tempo que cada usuário despende antes de abandonar a tarefa; número de erros cometidos.

Para avaliar melhor essas metas, podemos conduzir um estudo para avaliar qual é o problema principal. Vamos supor que o estudo revele que os principais problemas enfrentados pelos usuários são a dificuldade de uso (e.g., o usuário não sabe o que fazer num determinado momento por falta de instruções e controles claros na interface de usuário) e a ineficiência no uso do sistema (e.g., o usuário desiste quando descobre que há passos intermediários aparentemente desnecessários no processo de busca). Com base nesses resultados e nos dados quantitativos coletados, podemos então estabelecer as faixas de valores aceitáveis e ideais para cada indicador, conforme ilustrado pela Figura 4.6.

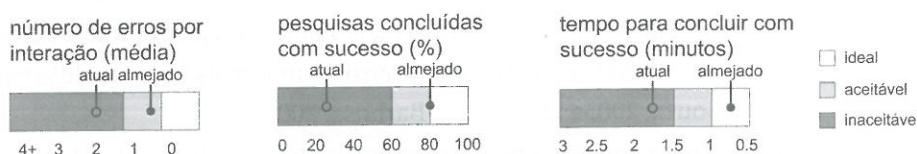


Figura 4.6 Faixas de valores para indicadores de usabilidade.

Durante a definição das metas de usabilidade, podemos estabelecer também metas de **retorno de investimento**, através de uma análise de custo e benefício envolvendo o sistema ou a prática atual, as despesas com o projeto do novo sistema e a economia que o seu uso deve proporcionar. Bias e Mayhew (2005) discutem diversas formas de avaliar o retorno de investimento. Podemos, por exemplo, calcular o tempo que um funcionário leva para realizar o seu trabalho antes e depois da introdução do novo sistema, e computar o ganho monetário com base no salário desse funcionário e o tempo economizado.

Nielsen advoga o **design paralelo**, que consiste em elaborar diferentes alternativas de design, de preferência por três ou quatro designers trabalhando de forma in-



dependente, para então selecionar as que vão ser detalhadas nas atividades seguintes do processo. Nessa etapa, cada designer deve empregar pouco tempo (desde algumas horas até dois dias) para elaborar seus designs iniciais e, portanto, trata-se de uma forma bastante barata de explorar o espaço de solução. Para motivar soluções bem diferentes, podemos solicitar a cada designer que explore um aspecto diferente do problema, tais como: usuários novatos vs. experientes; computador desktop vs. dispositivo móvel; interface gráfica vs. verbal vs. por caneta vs. por toque. Ao final dessa etapa, as soluções alternativas são analisadas e um design consolidado é elaborado, geralmente combinando elementos de mais de uma alternativa.

O **design participativo**, também advogado por Nielsen, consiste em a equipe de design ter acesso permanente a um conjunto de usuários tidos como representativos da população-alvo de usuários. Isso é importante porque, mesmo após as atividades iniciais de investigação, invariavelmente surgem questões ao longo do processo de design que requerem novas consultas aos usuários. Nielsen alerta, no entanto, que os usuários não são designers, então não podemos esperar que eles produzam designs ou entendam especificações produzidas utilizando notações que eles desconhecem, nem mesmo que saibam definir com clareza o que querem ou precisam. Em vez disso, ele chama atenção para a necessidade de produzirmos representações dos designs propostos que eles entendam facilmente, como protótipos, maquetes ou esboços de tela, para que eles possam reagir às propostas, fornecer *feedback* informativo, levantar novas questões e participar ativamente das discussões acerca das soluções propostas. Além disso, devemos envolver diferentes usuários ao longo do projeto, para que as decisões de design não se baseiem em idiossincrasias de uma ou duas pessoas, e para que tenhamos sempre um olhar novo sobre os designs propostos.

Para evitar inconsistências na interface com usuário projetada, é importante haver um responsável pelo **design coordenado da interface**, ou seja, da interface como um todo. Isso inclui não apenas os elementos de interface propriamente ditos, mas também toda a documentação, o sistema de ajuda e tutoriais produzidos sobre o sistema. Caso o produto deva ser introduzido como parte de uma família de produtos, devemos considerar a consistência entre eles de forma holística, mas sempre tomando o cuidado para que a consistência não adquira uma importância desmedida a despeito das metas de usabilidade prioritárias para o projeto.

Nielsen sugere que a equipe de design siga **diretrizes**, princípios bem conhecidos para o design da interface com usuário. À medida que a interface for projetada, deve ser feita uma avaliação heurística para avaliar se as diretrizes não estão sendo violadas (veja Seção 10.1.1). As diretrizes podem ser *gerais*, aplicáveis a todas as interfaces com usuário; *específicas a uma categoria* ou plataforma computacional



(e.g., interfaces gráficas baseadas em janelas para computadores desktop; interfaces gestuais para grandes monitores; interfaces verbais para atendimento automático pelo telefone; interfaces de toque para dispositivos móveis); ou ainda específicas a um *produto* individual (e.g., planilha eletrônica, editor de texto, editor gráfico). Por exemplo, “forneça feedback” é uma diretriz geral, à qual podemos relacionar diretrizes específicas a certas categorias (e.g., para uma interface gráfica: “certifique-se de que os principais objetos de interesse estejam visíveis na tela e que revelem seus principais atributos”; para um sistema Web: “certifique-se de que o usuário sabe onde ele está, de onde veio e para onde pode ir”) e produtos (e.g., “o ícone de uma lixeira deve estar visível e indicar se ela contém ou não algum item”). A Seção 8.2 apresenta algumas diretrizes comumente encontradas na literatura de IHC.

Antes de iniciar os esforços de implementação da interface com usuário, Nielsen recomenda fazer **protótipos** dos sistemas finais, que podem ser desenvolvidos rapidamente e a um custo baixo, para que sejam avaliados junto a usuários e modificados à medida que a equipe de design adquire um melhor entendimento dos problemas, visando oferecer uma solução mais adequada. Para que essa atividade possua custo baixo, apenas parte do sistema é prototipada. Podemos desenvolver um **protótipo horizontal**, que visa apresentar o sistema em abrangência mas com pouca profundidade (i.e., a aparência da interface e navegação entre telas, mas sem a funcionalidade subjacente, como algoritmos e acessos a bases de dados), ou um **protótipo vertical**, no qual pouca funcionalidade é explorada em profundidade para que seja testada em circunstâncias realistas. Nielsen enumera diversas estratégias para produzir protótipos mais rapidamente:

- não se importar muito com a eficiência da implementação (e.g., utilização de espaço em disco e tempo de processamento), desde que não seja essencial para a avaliação junto ao usuário;
- aceitar código de qualidade mais baixa ou pouco confiável;
- utilizar algoritmos simplificados e que não conseguem lidar com todos os casos específicos;
- fazer uma simulação com uma pessoa atuando como o computador “por trás da cortina”, numa técnica denominada *Wizard of Oz* (mágico de Oz), na qual o usuário, sem saber, interage com uma pessoa que determina as respostas do protótipo, e não com um processamento computacional;
- utilizar uma plataforma computacional diferente da almejada para facilitar o desenvolvimento do protótipo (e.g., utilizar uma ferramenta de autoria em vez da linguagem de programação que será utilizada para construir o produto final);

- utilizar protótipos de baixa fidelidade, mas que representem a essência da interação;
- utilizar dados falsos e conteúdos fictícios, desde que não atrapalhem a avaliação junto aos usuários;
- utilizar maquetes em papel em vez de um sistema funcional;
- utilizar um protótipo verbal, no qual o avaliador descreve oralmente uma interface possível e explora uma série de perguntas do tipo “E se...?”;
- utilizar cenários (veja Seção 4.3.5).

A partir dos protótipos, os designers devem fazer **testes empíricos**, que consistem principalmente na observação dos usuários ao utilizarem os protótipos para realizar certas tarefas. A Seção 10.2.1 descreve a avaliação através de testes de usabilidade.

Com base nos problemas de usabilidade e nas oportunidades reveladas pelos testes empíricos, os designers produzem uma nova versão da interface, e repassam pelas atividades do processo, num **design iterativo**. A cada iteração de design e avaliação, alguns problemas são corrigidos (e infelizmente outros podem ser introduzidos), e o processo deve ser repetir até que as metas de usabilidade tenham sido alcançadas. Nesse processo iterativo, é importante tornar as decisões de design explícitas e registrá-las para referência futura, ou seja, capturar o **design rationale** (Moran e Carroll, 1996). É importante registrar as decisões tomadas para que, no futuro, não sejam tomadas decisões que sacrificuem metas de usabilidade importantes ou que introduzam inconsistências por falta de informação sobre o histórico do projeto.

Finalmente, após a introdução de um produto, devemos **coletar dados de uso**, não apenas para avaliar o retorno de investimento, mas também para planejar a próxima versão do produto.

#### 4.3.3 Engenharia de Usabilidade de Mayhew

Deborah Mayhew (1999) propôs um ciclo de vida para a engenharia de usabilidade. Com uma visão holística, esse processo de design reúne e organiza diferentes atividades propostas na área de IHC para orientar o trabalho do designer em direção a uma boa solução interativa. A Figura 4.7 apresenta as três fases desse processo iterativo: análise de requisitos, design/avaliação/desenvolvimento e instalação.

Na fase de **análise de requisitos** são definidas as metas de usabilidade com base no perfil dos usuários, análise de tarefas, possibilidades e limitações da plataforma em que o sistema será executado e princípios gerais de design de IHC. Nesse processo, as metas de usabilidade costumam ser representadas em “guias de estilos” para auxiliar sua verificação durante as demais atividades do processo (veja Seção 8.4).

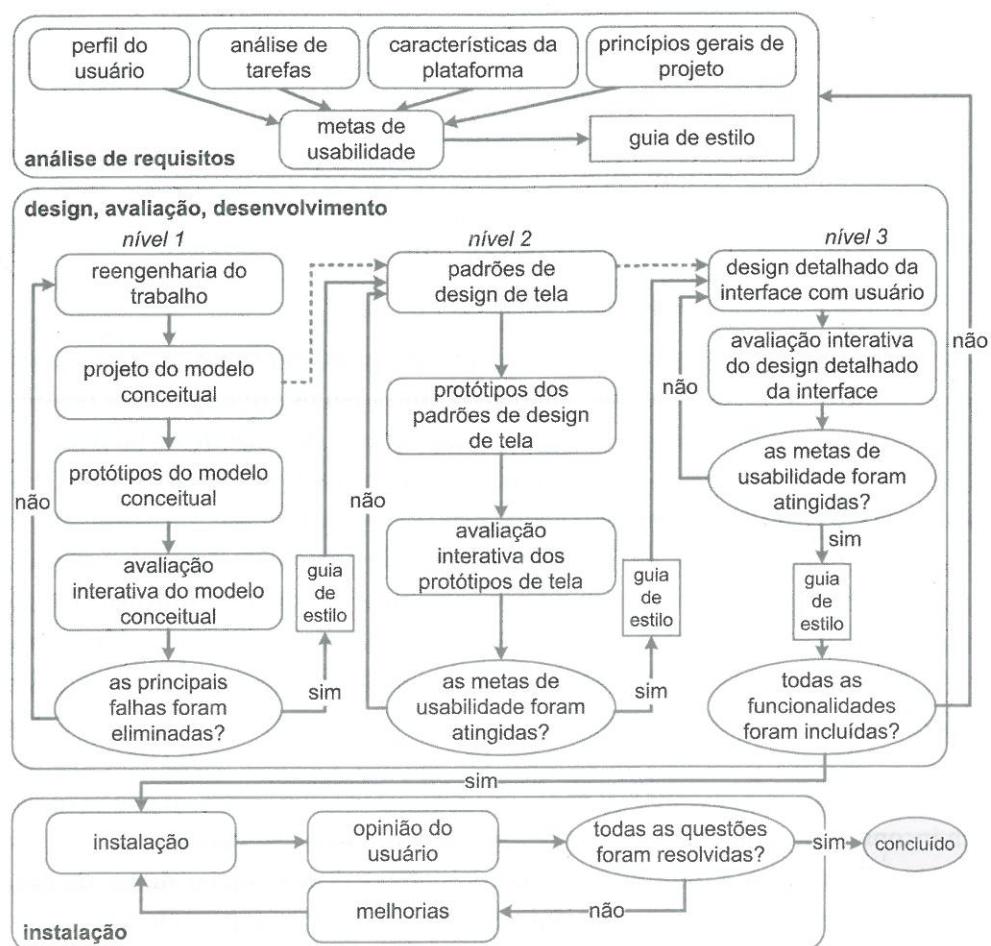


Figura 4.7 Ciclo de vida para a engenharia de usabilidade (adaptado de Mayhew, 1999).

A fase de **design, avaliação e desenvolvimento** tem por objetivo conceber uma solução de IHC que atenda às metas de usabilidade estabelecidas na fase anterior. Esse processo propõe projetar a solução de IHC em três níveis de detalhes. No primeiro nível, o designer precisa realizar a reengenharia do trabalho, repensando a execução das tarefas para alcançar os objetivos dos usuários, elaborar alternativas de solução do modelo conceitual, elaborar protótipos de baixa fidelidade e avaliar tais protótipos. No segundo nível, o designer deve estabelecer padrões de design de IHC para a solução sendo concebida, construir protótipos de média fidelidade de acordo com esses padrões e avaliá-los. No terceiro nível, o designer realiza o projeto detalhado da interface, com alta fidelidade, para ser implementado. Durante o desenvolvimento do sistema, a interface deve ser avaliada com a participação dos usuários.

Na fase de **instalação**, o designer deve coletar opiniões dos usuários depois de algum tempo de uso. Essas opiniões serão úteis para melhorar o sistema em versões futuras ou até mesmo para apontar a necessidade de desenvolver novos sistemas interativos ainda não previstos.

#### 4.3.4 Design Contextual

O design contextual (*contextual design*) é um processo de design de IHC que orienta o designer a compreender profundamente as necessidades dos usuários<sup>1</sup> através de uma investigação minuciosa do contexto de uso (Beyer e Holtzblatt, 1998; Holtzblatt *et al.*, 2001). Essa apreciação cuidadosa do que ocorre em contexto é fundamental para o designer elaborar uma solução de IHC adequada. As atividades do design contextual são: investigação contextual, modelagem do trabalho, consolidação, reprojeto do trabalho, projeto do ambiente do usuário, prototipação e teste com usuários.

Na **investigação contextual** (*contextual inquiry*), o designer busca conhecer quem são os usuários, suas necessidades, seus objetivos e a forma como ele trabalha no seu dia a dia. Essa investigação ocorre diretamente no ambiente de trabalho do usuário para que o designer possa ter acesso a informações do contexto. O objetivo da investigação contextual é revelar detalhes e motivações implícitas no trabalho dos usuários a fim de informar o designer sobre suas necessidades reais. Essas informações contextuais são importantes para apoiar as decisões de design. O método de investigação contextual é descrito em detalhes na Seção 5.5.7.

O conhecimento adquirido na investigação contextual permite ao designer **modelar o trabalho** de cada usuário investigado separadamente. Existem cinco tipos de modelos de trabalho utilizados no design contextual: modelo de fluxo, modelo de sequência, modelo de artefato, modelo de cultura e modelo físico. Cada um representa um aspecto do trabalho sob a perspectiva de um usuário investigado, podendo conter conceitos complexos e muitos detalhes. Eles são utilizados para registrar e compartilhar com a equipe de projeto os conhecimentos adquiridos na investigação contextual.

A **consolidação** dos modelos de trabalho possibilita organizar e atribuir significado ao trabalho desempenhado por cada papel, perfil ou classe de usuário investigado. Um diagrama de afinidade deve ser elaborado para estruturar coletivamente

1 Beyer e Holtzblatt empregam o termo *customers* (clientes) para referenciar aqueles que usam o sistema. Neste livro, utilizamos o termo *usuário* para nos referirmos às pessoas que realizam as atividades que já são ou serão apoiadas pelos sistemas computacionais interativos investigados ou sendo concebidos. Denominamos *stakeholders* as demais partes interessadas no sistema.



a forma como os usuários trabalham, sem perder as particularidades de cada caso. O diagrama de afinidade sintetiza grande quantidade de dados qualitativos em um grande mapa. Além disso, os modelos de trabalho de todos os usuários investigados devem ser combinados em cinco modelos de trabalho consolidados, um para cada tipo de modelo. O resultado dessa consolidação é um conjunto de dados corporativos que vai guiar o projeto de IHC e podem ser reutilizados em projetos futuros. Os diagramas de afinidade são descritos na Seção 5.5.4, no contexto de sessões de *brainstorming*.

A consolidação dos modelos de trabalho fornece insumos para o designer **re-projetar** a forma como os usuários trabalham. O designer utiliza *storyboards* para explorar ideias sobre como melhorar a prática de trabalho com o suporte oferecido pela tecnologia.

Uma vez concebida uma nova maneira de trabalhar, o designer segue **projetando uma solução de interação e de interface** que apoie essa nova forma de trabalhar. Por fim, o designer deve construir **protótipos** do sistema e **avaliá-los** junto aos usuários. Isso permite revisar e refinar o projeto iterativamente até chegar a uma solução satisfatória.

#### 4.3.5 Design Baseado em Cenários

O **design baseado em cenários** é um processo que utiliza diferentes tipos de cenários como representação básica e fundamental durante todas as atividades envolvidas na concepção de uma solução de IHC (Rosson e Carroll, 2002; Carroll, 1995). Um *cenário* é “simplesmente uma história sobre pessoas executando uma atividade” (Rosson e Carroll, 2002, p. 2). Como os cenários são geralmente escritos em linguagem natural, o seu uso motiva todos os interessados no sistema a participarem e contribuírem com as decisões de design, direta ou indiretamente.

Ao escrever, ler e revisar cenários, a equipe de design (incluindo representantes dos usuários) tem a oportunidade de discutir e analisar como as atividades dos usuários são afetadas pela tecnologia existente e como elas poderiam ser afetadas pelo sistema sendo desenvolvido. As histórias dos cenários estimulam a imaginação da equipe de design e encorajam a análise de caminhos alternativos. Por exemplo, perguntas do tipo “E se...” permitem à equipe de design imaginar outros caminhos para a história descrita no cenário, possivelmente contendo alguns elementos diferentes. Desse modo, cenários são uma ferramenta útil e barata para gerar e avaliar diversas ideias durante as atividades de design.

A Figura 4.8 ilustra as atividades propostas pelo design baseado em cenários. Basicamente, as atividades são: análise do problema, projeto de uma solução de IHC,

prototipação e avaliação da solução proposta. O diferencial desse processo está na forma como essas atividades são executadas. Esse processo inicia com a elaboração de cenários de problema, e continua com sucessivas análises e transformações de cenários de acordo com a atividade sendo executada. Apesar de as atividades serem apresentadas sequencialmente, o processo é iterativo, ou seja, sempre que necessário, a equipe de design pode revisar o que foi feito anteriormente.

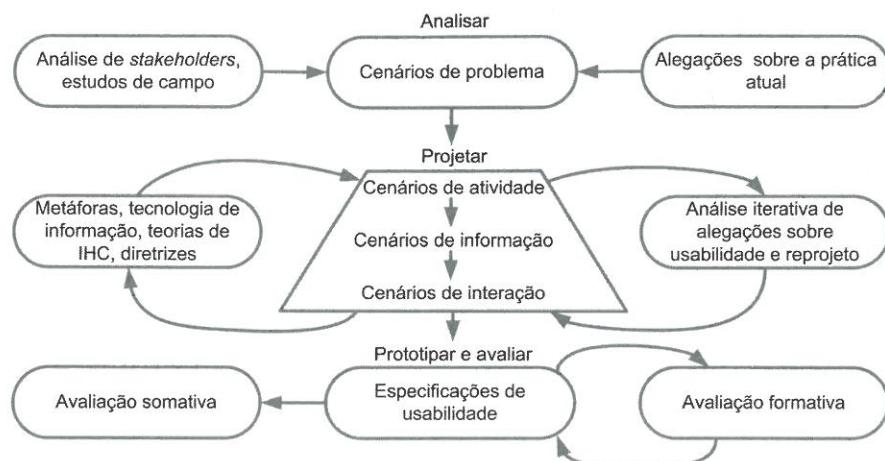


Figura 4.8 Atividades do design baseado em cenários.

Na análise do problema, a equipe de design estuda a situação atual junto aos interessados no sistema (*stakeholders*: clientes, usuários etc.). Com o conhecimento adquirido sobre a situação atual, a equipe de design deve formular **cenários de problemas** que cobrem características dos usuários, suas atividades típicas e críticas, os artefatos que eles utilizam e o contexto de uso (Rosson e Carroll, 2002). Uma vez que a compreensão da equipe de design sobre a situação atual tenha sido consolidada em cenários de problema, o próximo passo é elaborar uma solução de IHC adequada que resolva os problemas descritos.

Na atividade de projeto, a equipe de design deve explorar ideias para a solução de IHC elaborando três tipos de cenários: cenários de atividade, de informação e de interação. Um **cenário de atividade** é uma narrativa sobre as tarefas típicas e críticas que os usuários vão executar com ajuda do sistema. Eles concentram-se em relatar as funcionalidades do sistema, sem, no entanto, especificar ainda como os usuários vão utilizá-lo ou como deve ser a aparência do sistema. Um **cenário de informação** é uma elaboração de um cenário de atividade que descreve as informações fornecidas pelo sistema ao usuário durante a interação. Um **cenário de interação** especifica em



detalhes as ações do usuário e as respectivas respostas (*feedback*) do sistema necessárias para executar as tarefas apoiadas pelo sistema.

As ideias para a solução de IHC devem ser avaliadas continuamente durante o processo de design. Isso normalmente é realizado através de um protótipo que implementa ou demonstra partes da solução de IHC descritas em cenários. Os cenários também são responsáveis por guiar a avaliação, seja durante a concepção da solução de IHC ou depois que ela estiver pronta. Os cenários descrevem hipóteses sobre o uso da solução de IHC que permitem prever: os perfis dos usuários, seus objetivos, algumas tarefas que tentarão realizar para atingirem seus objetivos, algumas sequências de ações que tentarão realizar para concluir as tarefas escolhidas, as respectivas respostas do sistema e algumas possíveis reações dos usuários. Cenários são discutidos nas Seções 6.3 e 7.2, no contexto de análise e design de IHC, respectivamente.

#### 4.3.6 Design Dirigido por Objetivos

O processo de design dirigido por objetivos orienta o designer a projetar uma solução de IHC criativa que apoie os usuários em atingirem seus objetivos (Cooper *et al.*, 2007). O diferencial desse processo é incentivar o designer a explorar as tecnologias disponíveis da melhor forma possível para oferecer aos usuários maneiras mais criativas, inovadoras e eficientes de alcançarem seus objetivos. Por exemplo, se a tecnologia permite sacar dinheiro de contas no banco em um caixa eletrônico que pode funcionar 24 horas durante toda a semana, por que limitar esse objetivo ao atendimento de um funcionário que trabalha seis horas em cinco dias na semana? Se a tecnologia permite enviar fotos em poucos minutos para familiares em locais bem distantes, como em outros países, por que esperar dias para as fotos chegarem ao destinatário através do correio tradicional? É claro que as tecnologias possuem várias limitações e não possuem as mesmas capacidades humanas. Contudo, o importante é explorar as possibilidades da tecnologia em favor dos usuários.

Segundo Cooper e seus colegas, alguns designers podem tentar projetar um sistema que permita ao usuário realizar as mesmas tarefas ou ações que ele realizava anteriormente. Dessa forma, a nova solução será mediana, pois sempre oferecerá o mesmo tipo de apoio ao usuário, limitado à forma como os usuários atingem seus objetivos atualmente. As novas soluções continuarão limitadas ao que as tecnologias anteriores já permitiam fazer, sem explorar novas possibilidades oferecidas pelas tecnologias atuais. Essa perspectiva de design abre pouco espaço para soluções criativas, capazes de explorar tecnologias antigas e novas de maneira inovadora para apoiar o usuário. Se por um lado é preciso estar ciente do que as pessoas sabem e estão acostumadas a fazer, por outro a nova solução não pode estar limitada a isso.

Como ser criativo e inovar sem estar limitado às tarefas executadas anteriormente pelos usuários? Para responder essa pergunta, primeiro é preciso diferenciar objetivos de tarefas ou ações do usuário. Cooper, Reimann e Cronin (2007, p. 15) definem um **objetivo** como sendo “uma expectativa de uma condição final, em que ações e tarefas são passos intermediários (em diferentes níveis de organização) que ajudam alguém a atingir um objetivo ou conjunto de objetivos”. Segundo essa definição, um objetivo seria o destino final, alcançado quando um usuário percorre certos caminhos definidos por tarefas e ações. Por exemplo, um objetivo do usuário pode ser comunicar algo a um colega. Ele pode fazer isso realizando diferentes tarefas, como, por exemplo, escrevendo uma carta convencional, enviando um e-mail, fazendo uma ligação telefônica, ou enviando uma mensagem de texto via telefone celular. Uma tarefa pode ser composta de outras tarefas mais simples. Por exemplo, a tarefa de escrever um e-mail pode ser composta pelas tarefas de digitar um texto e formatá-lo.

Os objetivos representam as motivações que levam o usuário a realizar suas tarefas. Conhecer esses objetivos permite compreender o significado das tarefas realizadas atualmente. Com isso, é possível repensar as tarefas com liberdade para imaginar novas possibilidades de atingir os objetivos do usuário, aproveitando ao máximo as tecnologias antigas e novas de forma criativa, inovadora e eficiente. Quando o design de IHC é dirigido pelos objetivos do usuário, é possível explorar a tecnologia para eliminar tarefas irrelevantes e aperfeiçoar as demais.

Por exemplo, pense no objetivo de comprar produtos em uma loja. Uma sequência de tarefas possível seria o caixa pegar cada produto, digitar seu valor na caixa registradora (ou obter a informação através da leitura de um código de barras), o cliente pagar e levar a mercadoria. Outra sequência possível de tarefas seria o próprio sistema identificar no carrinho do cliente o tipo e a quantidade dos produtos escolhidos (por exemplo, através de RFID — identificação por rádio frequência) e o cliente pagar e levar os produtos. Hoje em dia já é comum o próprio cliente informar a um sistema de comércio eletrônico quais produtos deseja comprar, como vai pagar, qual meio de transporte deve ser utilizado e onde os produtos devem ser entregues pela empresa.

Essas são apenas três formas diferentes de atingir o objetivo de comprar produtos, cada qual com uma sequência de tarefas diferente e com vantagens e desvantagens. Ao enunciar o objetivo do usuário, é possível explorar e comparar diferentes formas de atingi-lo. Assim, podemos empregar as tecnologias de maneira mais apropriada para apoiar e satisfazer os objetivos dos usuários.

O design dirigido por objetivos é um processo sistemático proposto para investigar e atender às necessidades e aos objetivos dos usuários, bem como atender aos

requisitos técnicos, do negócio e da organização. Esse processo é dividido em seis fases (Figura 4.9): pesquisar, modelar, definir requisitos, projetar, refiniar e manter.



**Figura 4.9** Processo de design dirigido por objetivos (adaptado de Cooper *et al.*, 2007).

Na fase de **pesquisa**, o designer está interessado em conhecer o usuário, o domínio do sistema e o contexto de uso. Ele investiga comportamentos dos usuários que sugerem seus objetivos e motivações ao realizar suas atividades enquanto manipulam certos artefatos. O comportamento dos usuários pode estar associado a um papel ou função exercida, ou ainda corresponder a suas preferências pessoais.

A fase de **modelagem** tem por objetivo organizar e registrar o conhecimento adquirido na fase de pesquisa através da elaboração de modelos do usuário, domínio e contexto de uso. Os modelos são úteis para representar conceitos e relações entre eles, facilitando à equipe de design registrar, compreender, visualizar e discutir sobre o conhecimento adquirido na fase anterior. Sem a organização proporcionada por modelos, é possível que dados coletados na fase de pesquisa permaneçam inexplorados durante o processo de design por falta de atenção ou de um trabalho sistemático.

Na fase de **definição de requisitos**, o designer interpreta as informações coletadas e estruturadas nos modelos para definir os requisitos do usuário, do negócio e técnicos. Algumas vezes esses requisitos são conflitantes, e é preciso fazer concessões. Por exemplo, em um sistema de comércio eletrônico, alguns usuários podem estar interessados em comprar os produtos mais baratos, enquanto a empresa também deseja vender produtos mais caros. Como equilibrar isso? Uma possível solução seria expor tanto os produtos mais caros quanto os mais baratos em diferentes “vitrines” do sistema, e também permitir ao usuário consultar os produtos mais baratos de alguma maneira.

Na fase de **projeto conceitual** (*framework definition*), o designer concebe uma solução de interação e um esboço de interface pouco detalhado. Sua preocupação principal está na concepção da estrutura e do comportamento da interface. Na fase de **refinamento**, o foco é detalhar a solução de interface, definindo todas as características dos elementos de interface, tais como tamanho, cores e ícones. Nessa fase o designer verifica a coerência das tarefas percorrendo a interface. A solução detalhada pode ser avaliada junto aos usuários, e, caso seja necessário, ela é revisada. O resultado da fase de refinamento é uma documentação detalhada da solução de interação e de interface projetada.



Durante a implementação da solução projetada, podem surgir limitações técnicas que impeçam ou atrapalhem sua construção. Por isso, na última fase do processo, o designer tem a responsabilidade de **manter** a coerência da solução proposta enquanto acomoda as limitações técnicas imprevistas. A presença do designer nessa fase ainda é de extrema importância. Se o designer não estiver mais disponível durante a construção da solução, a equipe de desenvolvimento pode ter de tomar decisões de IHC sem o conhecimento necessário sobre usuários, objetivos e contexto de uso.

Cada fase do design dirigido por objetivos é iterativa. Podemos observar que não existe atividade exclusiva para avaliação nesse processo. A avaliação deve ser realizada durante cada atividade, principalmente nas fases de projeto e de refinamento da solução.

### 4.3.7 Design Centrado na Comunicação

O design centrado na comunicação (Barbosa *et al.*, 2004) tem como base teórica a engenharia semiótica (de Souza, 2005a), apresentada na Seção 3.8. Essa teoria comprehende a interação humano-computador como um processo de comunicação entre o usuário e o designer do sistema, através da sua interface. Em outras palavras, a interface revela, durante o uso do sistema, a metacomunicação do designer (ou seja, as intenções de design e os princípios interativos). Quando o usuário tem acesso a essa metacomunicação, ele possui melhores condições de aprender e usar o sistema de forma produtiva, eficiente e criativa. A motivação principal do design centrado na comunicação é elaborar uma solução de IHC que transmita a metacomunicação do designer de forma eficiente e eficaz, ou seja, produzir um sistema interativo com alta comunicabilidade (conceito apresentado na Seção 2.2.3). Para isso, esse processo orienta o designer a se posicionar como um dos interlocutores das conversas que ocorrem durante a interação.

Para aumentar as chances de projetarmos uma interface que transmita adequadamente a metacomunicação do designer, a equipe de projeto deve definir o conteúdo dessa mensagem e compartilhá-la efetivamente entre seus membros. Em outras palavras, se a equipe de design não possui uma visão de design compartilhada consistente, ela dificilmente terá sucesso em comunicá-la aos usuários através da interface. O design centrado na comunicação parte do pressuposto que, se os designers conseguirem registrar a metacomunicação em um conjunto de artefatos e comunicá-la aos membros da equipe, eles terão melhores condições de comunicá-la aos usuários através da interface (Barbosa *et al.*, 2004).

Desse modo, o design centrado na comunicação busca representar a metacomunicação e promover uma compreensão compartilhada dela por todos os membros da



equipe de projeto. Essa compreensão deve ser registrada desde o início do processo de design, ainda na fase de análise do problema, e chegar até as atividades de implementação e testes do sistema. Isso ajuda a evitar e corrigir interpretações incompletas ou equivocadas, pois o compartilhamento da metacomunicação representa uma oportunidade para todos os membros da equipe contribuírem e revisarem o que foi produzido até o momento. Cada membro contribui com a sua perspectiva particular sobre o assunto em questão.

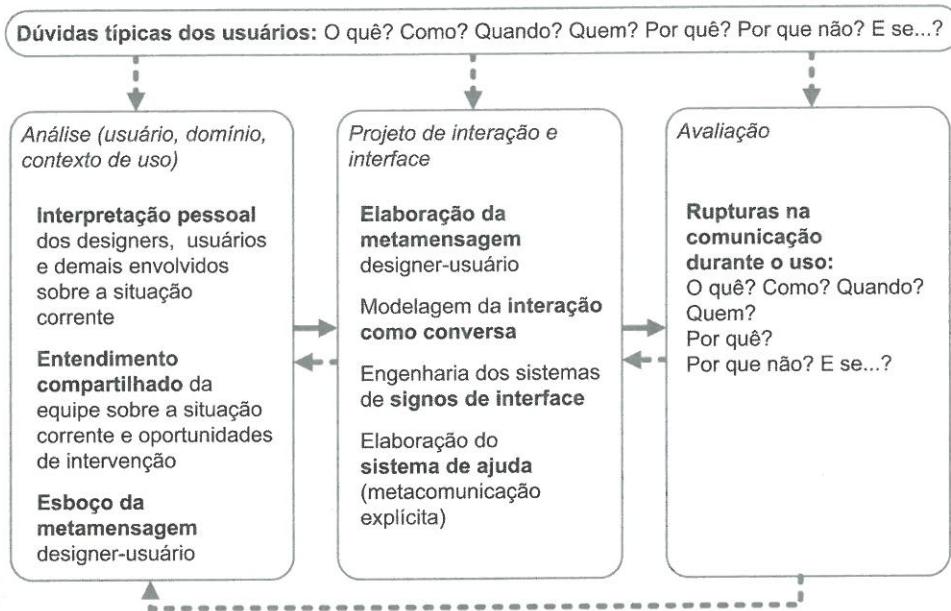
Mas onde os designers buscam insumos para construir a metacomunicação? Além do levantamento e da análise tradicionais dos objetivos, necessidades e preferências dos usuários, o design centrado na comunicação faz uso dos resultados de pesquisa sobre a construção da ajuda on-line de Silveira e colegas (Silveira *et al.*, 2004; Silveira, 2002).

A ajuda on-line é uma forma privilegiada de transmitir a metacomunicação ao usuário, pois fornece meios de transmitir direta e explicitamente as intenções de design e os princípios de interação. Silveira explora dúvidas comuns dos usuários para construir um sistema de ajuda on-line (veja Seção 7.5). Ela propõe coletar informações durante todo o processo de desenvolvimento para responder as perguntas que os usuários costumam fazer quando encontram problemas durante o uso. Por exemplo, a ajuda on-line deve buscar responder perguntas como “O que é isto?”, “Para que serve isto?” e “Como faço isto?” quando se referem a conceitos representados na interface. Essas perguntas são utilizadas para auxiliar na elaboração do conteúdo da metacomunicação e na sua comunicação aos usuários através da engenharia dos sistemas de signos da interface e da ajuda on-line.

O design centrado na comunicação leva adiante a proposta de Silveira para apoiar todas as atividades envolvidas no projeto de uma solução de IHC com alta comunicabilidade, e não apenas a construção da ajuda on-line. Desse modo, o design centrado na comunicação propõe a elaboração da metacomunicação orientada por um conjunto de perguntas derivadas das dúvidas comuns dos usuários. Essas perguntas serão respondidas durante a atividade de análise da situação atual e o projeto da solução de IHC. Por exemplo, durante a atividade de análise, o designer deve procurar responder perguntas do tipo: “O que o usuário deseja ou desejará fazer com o sistema?”, “Quem pode fazer isso?” e “O que deve ser feito antes disso?”. Já durante o projeto da solução, o designer deve buscar responder perguntas do tipo: “Como o usuário costuma atingir esse objetivo atualmente?”, “Como ele gostaria de fazer isso no futuro?”, “Que problemas podem ocorrer enquanto o usuário busca atingir esse objetivo?”, “Como solucionar tais problemas?” e “Como desfazer os resultados de uma ação indesejada?”. Em particular, a preocupação com as possíveis dúvidas dos usuários motiva a equipe

de design a tentar prever possíveis problemas durante o uso, a projetar formas de prevenção e recuperação desses problemas, bem como a comunicar sob demanda suas intenções de design e princípios interativos através da interface.

O design centrado na comunicação propõe três atividades: a análise do usuário, domínio e contexto de uso, o projeto de interação e interface, e a avaliação do que foi projetado (Figura 4.10). O diferencial desse processo consiste em nortear os esforços de design desde o início do processo pelas dúvidas que os usuários costumam ter durante a interação. Dessa forma, a solução de IHC é projetada para evitar o surgimento das dúvidas dos usuários e comunicar adequadamente informações necessárias para sanar dúvidas que eventualmente surjam.



**Figura 4.10** Design centrado na comunicação.

Embora um objetivo importante seja evitar que os usuários tenham dúvidas durante a interação, o design centrado na comunicação reconhece que é inevitável que diferentes usuários tenham dúvidas em alguns momentos, assim como é inevitável que mal-entendidos ocorram durante a conversação humana. Portanto, ele orienta o designer a projetar uma solução de IHC que comunique adequadamente não apenas as situações esperadas (que “dão certo”), mas também que ajude o usuário a evitar e se recuperar das rupturas de comunicação durante a interação (veja Seção 7.3.3).

O design centrado na comunicação diferencia a interface e a interação a ponto de propor o projeto de cada uma separadamente, mantendo a consistência e coerência entre elas. Recomenda que o projeto da interação seja realizado utilizando a MoLIC,



Linguagem para a Modelagem da Interação como Conversa (Barbosa e Paula, 2003; Silva e Barbosa, 2007). Para o projeto de interfaces gráficas, esse processo recomenda o uso de esboços de tela, mas não faz recomendações específicas para outros tipos de interface (via voz, gestos, realidade virtual etc.). O Capítulo 7 descreve as atividades de projeto de interação e interface desse processo de design, bem como os modelos e representações utilizados.

Os objetivos do usuário serão atingidos através de conversas entre usuário e o preposto do designer durante a sua interação com o sistema. Segundo essa interpretação, a interface representa uma linguagem e um meio de comunicação entre usuário e o preposto do designer, linguagem essa que define a forma e o conteúdo do que os interlocutores podem falar, em que ordem as falas podem ocorrer e quem pode falar em cada momento da conversa. Sendo assim, o designer deve projetar a conversa (a interação) e a forma de representá-la na interface. Em linha com Hoover, Rinderle e Fischer (1991), o design centrado na comunicação envolve modelos com diferentes focos e níveis de abstração e detalhes. Ele recomenda primeiro projetarmos a conversa usuário–designer em um modelo de interação, concentrando-nos nas trocas de turno de fala e nos tópicos e foco de cada turno de fala, e abstraindo-nos dos detalhes da interface. Em seguida, devemos projetar estrutura, *layout* e formatação da interface propriamente dita (isto é, projetar a expressão da conversa usuário–designer na interface).

Como sempre, a proposta de solução de IHC (interação e interface) deve ser avaliada para verificarmos se ela atende às necessidades dos usuários. No design centrado na comunicação, avaliamos se a metacomunicação foi enviada e recebida adequadamente. Se algum problema for encontrado, devemos rever o projeto de interação, o de interface, ou ambos. Durante as atividades de projeto e de avaliação, também pode ser necessário ampliar, refinar ou revisar a análise realizada anteriormente. Por isso, o design centrado na comunicação é bastante iterativo, com ciclos envolvendo as três atividades básicas de design: análise, síntese e avaliação, até que uma solução seja avaliada como satisfatória. O Capítulo 10 apresenta os métodos de inspeção semiótica e de avaliação de comunicabilidade, utilizados no design centrado na comunicação.

Os objetivos, as necessidades, as preferências e os pontos de vista dos usuários ganham importância em todas as atividades deste processo de design, pois desde o início a atenção da equipe está centrada na comunicação com os usuários. A definição compartilhada da metacomunicação que guia o projeto de IHC facilita construir um sistema interativo consistente, coerente e com alta comunicabilidade.

Vale observar que, embora os usuários e demais *stakeholders* participem intensamente das atividades de análise e avaliação ao longo de todo o projeto, cabe aos

designers elaborarem a metacomunicação e a solução de IHC com base no que observaram e ouviram durante essas atividades. Embora a participação dos usuários seja imprescindível para a qualidade e o sucesso da solução, eles não são *diretamente* codesigners da solução projetada. Em outras palavras, no design centrado na comunicação, a solução de IHC é projetada envolvendo os usuários e *para* eles, mas não *por* eles.

#### 4.4 Integração das Atividades de IHC com Engenharia de Software

As áreas de IHC e de Engenharia de Software (ES) possuem diferentes perspectivas sobre o que é importante em um sistema interativo, sobre o que significa utilizá-lo e sobre como desenvolvê-lo. Cada área vem evoluindo historicamente por um caminho próprio e independente. Embora a preocupação com a qualidade de uso apareça desde os primeiros trabalhos sobre qualidade de software no final de década de 1970 (McCall *et al.*, 1977; Cavano e McCall, 1978; Boehm, 1981), a ES tem direcionado seus esforços para fatores de qualidade mais relacionados com engenharia — construção, instalação e manutenção — deixando para segundo plano a forma como os sistemas interativos serão utilizados.

Na **perspectiva de design centrada no sistema**, comum na área de Engenharia de Software, um sistema interativo é um artefato circunscrito e encapsulado por uma interface que recebe dados de entrada, processa esses dados com algum programa (codificado em software ou hardware) e retorna dados de saída (Figura 4.11). O que mais importa nessa perspectiva é aquilo que ocorre *dentro* do sistema. Tudo o que ocorre na fronteira ou fora dele, inclusive a própria interface, acaba recebendo pouca ou nenhuma atenção. O objetivo principal seria construir um sistema fidedigno (*dependable*) que seja capaz de processar adequadamente os dados de entrada e saída transmitidos através de uma interface bem definida. Os fatores de qualidade mais valorizados por essa perspectiva estão relacionados com a **construção** de um sistema interativo, como, por exemplo, disponibilidade, integridade, robustez, manutenibilidade e recuperabilidade (Avizienis *et al.*, 2004). Para atender a esses fatores de qualidade é preciso concentrar em conceitos fundamentais para construção de sistemas, tais como: arquitetura do sistema, estrutura de dados, mecanismos de persistência de dados, formas de comunicação entre sistemas, algoritmos, sistema operacional, linguagem de programação, ambiente de desenvolvimento, e assim por diante.

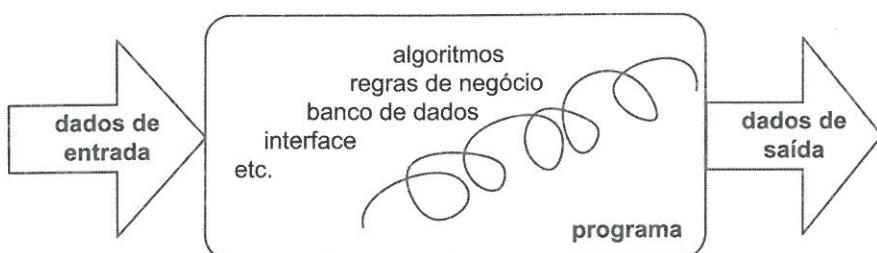


Figura 4.11 Perspectiva de design centrado no sistema.

A definição de uma interface permite ao engenheiro de software especificar a forma como um sistema irá interagir com o mundo externo. Tudo o que é possível solicitar ao sistema e receber dele será definido pela interface. Dessa maneira, o engenheiro de software geralmente abstrai o mundo externo ao construir o sistema, pois ele espera que o mundo se comunique “corretamente” com o sistema, conforme estabelecido pela interface. Abstrair o mundo externo pode funcionar bem quando estamos lidando com a comunicação entre sistemas, porque eles podem ser construídos para se comunicarem obedecendo a interfaces bem definidas. Contudo, a abstração do mundo externo costuma trazer problemas quando igualamos a interface com pessoas à interface com outros sistemas computacionais. A comunicação entre sistemas computacionais possui características bem diferentes da comunicação usuário-sistema. Quando o usuário entra em cena, as características humanas devem ser consideradas e endereçadas adequadamente durante a interação. Além disso, o contexto e o ambiente em que o usuário e o sistema estão inseridos também influenciam o uso do sistema e devem ser considerados.

Compreender e endereçar os problemas que ocorrem durante a interação usuário-sistema exigem conhecimentos além daqueles de base Matemática e Lógica que fundamentam a Engenharia de Software e a Computação como um todo. Exigem conhecimentos relacionados com características particulares das pessoas e das culturas em que estão inseridas. Um sistema interativo é construído para sempre executar um conjunto bem definido de instruções, que processam dados de entrada e saída. Diferente das máquinas, uma pessoa sente, pensa, interpreta, reinterpreta, é criativa, compartilha de uma cultura, vive em sociedade, muda de opinião, gostos, costumes, enfim, uma pessoa está sempre evoluindo. Não é possível prever ou controlar com exatidão a interpretação e o comportamento de alguém durante o uso de um sistema. De fato, tratar adequadamente o uso de sistemas interativos exige conhecimentos e esforços multidisciplinares muito além da Computação, incluindo áreas como Linguística, Psicologia, Sociologia, Ergonomia, Artes, dentre outras (Sharp *et al.*, 2007; Seção 1.4).



Como os conhecimentos matemáticos e lógicos que fundamentam a ES não dão conta das questões relacionadas ao uso que uma pessoa faz de sistemas interativos, a área de IHC vem se estabelecendo como área multidisciplinar que propõe uma perspectiva diferente para o desenvolvimento de sistemas interativos. Na perspectiva do design centrado no uso, comum a profissionais de IHC, conforme discutido no Capítulo 1, um sistema interativo é um artefato com o qual o usuário interage durante a realização de suas atividades em determinado contexto. O foco dessa perspectiva deixou de ser o que ocorre dentro do sistema e passou para aquilo que ocorre *fora* do sistema e *através* da sua interface. O mais importante nessa perspectiva é a forma como o usuário se apropria daquilo que o sistema pode oferecer em apoio aos seus objetivos em um determinado contexto. Assim, o objetivo do design centrado no uso é conceber (mais no sentido de projetar e avaliar do que implementar) um sistema interativo que sirva de apoio ao usuário na realização de suas atividades e no alcance dos seus objetivos. Para isso, é preciso trabalhar com conceitos relacionados ao uso do sistema, como, por exemplo, o contexto de uso, os objetivos do usuário, as formas como ele costuma alcançá-los, as características do usuário (formação, habilidades, prática na atividade sendo apoiada pelo sistema, cultura, gostos, preferências etc.) e possíveis formas de interação através da sua interface com usuário.

Desde sua origem, IHC tem sido proposta em contraste com o desenvolvimento centrado no sistema tipicamente praticado pela ES (Norman e Draper, 1986; Shneiderman, 1998; Sharp *et al.*, 2007). Para IHC, o uso que as pessoas vão fazer do sistema é o que deve guiar seu desenvolvimento. O usuário não deveria ser obrigado a adequar ao sistema sua forma de pensar, de realizar suas atividades, de trabalhar, de interagir com outras pessoas ou com instituições, e assim por diante. Na verdade, o sistema é que deveria ser construído de forma adequada ao usuário e suas necessidades e desejos.

As diferentes perspectivas de IHC e ES sobre o desenvolvimento de sistemas interativos deram origem a métodos, técnicas e processos próprios de cada área. Recentemente, alguns pesquisadores têm investigado a integração de métodos e técnicas de IHC em processos de desenvolvimento de software propostos em ES (Seffah *et al.*, 2005). As principais abordagens de integração de processos de IHC e ES são:

- definição de características de um processo de desenvolvimento que se preocupa com a qualidade de uso;
- definição de processos de IHC paralelos que devem ser incorporados aos processos propostos pela ES;
- indicação de pontos em processos propostos pela ES em que atividades e métodos de IHC podem ser inseridos.



Uma maneira de integrar as áreas de IHC e de ES é definir as características que um processo de desenvolvimento deve ter para tratar adequadamente a qualidade de uso. Gulliksen e seus colegas (2005) identificaram 12 princípios-chave que um processo de desenvolvimento deve ter para cuidar adequadamente da qualidade de uso. São eles:

- *foco no usuário*: os objetivos e as necessidades do usuário devem guiar o processo de desenvolvimento desde o início, para evitar que o desenvolvimento seja guiado pela tecnologia;
- *participação ativa do usuário*: representantes dos usuários devem participar ativamente durante todo o processo de desenvolvimento;
- *desenvolvimento iterativo e incremental*: o desenvolvimento do sistema deve ser iterativo e incremental para permitir a avaliação e revisão das propostas de solução, bem como liberar logo para o usuário partes do sistema que já tenham sido desenvolvidas;
- *representações de design simples*: o resultado do design deve ser representado de forma que possa ser facilmente compreendido pelos usuários e demais envolvidos no processo de desenvolvimento;
- *prototipação*: protótipos em diferentes níveis de detalhes devem ser utilizados para visualizar e avaliar propostas de solução junto aos usuários;
- *avaliar o uso em contexto*: avaliar as propostas de solução considerando os critérios de qualidade de uso definidos como prioridade para o sistema em questão, sempre atento às reações dos usuários no contexto de uso;
- *atividade de design explícita e consciente*: o processo de desenvolvimento deve conter atividades dedicadas ao design da solução de interação e de interface com usuário;
- *atitude profissional*: o processo de desenvolvimento deve ser executado por uma equipe multidisciplinar;
- *defensor da qualidade de uso*: um profissional de IHC deve participar continuamente do processo de desenvolvimento com a responsabilidade de tomar as decisões necessárias para favorecer a qualidade de uso;
- *design holístico*: todos os aspectos que influenciam o uso devem ser considerados em conjunto durante o processo de desenvolvimento;
- *customização do processo*: o processo de desenvolvimento deve ser adaptado a cada organização;
- *atitude centrada no usuário*: todos os envolvidos no processo de desenvolvimento devem estar cientes de e concordar com a importância da qualidade de uso e da participação ativa do usuário durante o processo.

Esses princípios-chave são comuns em processos de design de IHC, porém não costumam ser considerados em processos de desenvolvimento propostos na ES. Certamente um profissional de IHC está mais bem preparado para colocar em prática esses princípios dentro de processos de desenvolvimento, pois são necessários muitos conhecimentos de IHC que engenheiros de software geralmente não possuem na profundidade necessária. Além disso, é importante manter a perspectiva do usuário e do sistema durante o desenvolvimento para que as qualidades de uso e de construção recebam a atenção e o cuidado desejados.

Outra forma de integrar IHC e ES é através da execução de processos de IHC paralelos a processos de ES (Seffah *et al.*, 2005). O ciclo de vida em estrela, o design dirigido por objetivos e o design centrado na comunicação, por exemplo, poderiam ser executados em paralelo a processos propostos pela ES. Nesse caso, é necessário manter a consistência entre os resultados das atividades de cada processo. Se considerássemos a interface com usuário como algo completamente separado e isolado do restante do sistema, essa integração seria fácil de ser resolvida. Entretanto, as decisões de design de interação e de interface também afetam as funcionalidades do sistema e o que está por trás da interface com usuário. Por exemplo, John e seus colegas (2004) mostram como questões de IHC estão relacionadas com a arquitetura do software e não somente a uma camada ou “casca” independente.

O terceiro tipo de abordagem para a integração de IHC e ES aponta as atividades nos processos da ES em que métodos e práticas de IHC podem ser aplicados. Ferre e seus colegas (Ferre, 2003; Ferre, *et al.*, 2004; 2005) realizaram uma revisão bibliográfica sobre os métodos e as práticas de IHC e definiram onde podem ser utilizados em um processo genérico de desenvolvimento de software. A Figura 4.12 apresenta o mapeamento das atividades de IHC em atividades de um processo genérico de desenvolvimento de software da ES que eles propuseram. Identificar onde os conhecimentos de IHC podem ser empregados num processo de desenvolvimento representa um passo importante para a ampla utilização desses conhecimentos na prática.

Na elicitação de requisitos em um processo de ES, devem ser coletadas informações sobre os usuários, seus objetivos, como costumam atingi-los (suas tarefas) e sobre o contexto de uso. A intenção dessa atividade é coletar dados sobre os elementos envolvidos durante o uso, que vão além da interface com usuário.

A interpretação desses dados é realizada durante a análise de requisitos em um processo de ES. Durante essa atividade, o engenheiro de requisitos costuma representar informações e tomar decisões que influenciam ou dizem respeito à interação e à interface com usuário. Por exemplo, quando ele define em um cenário ou em um caso de uso a sequência de ações que deve ser executada no sistema para o usuário

atingir um objetivo, ele toma decisões relacionadas com a interação e com a interface com usuário. De forma mais explícita, ele pode utilizar protótipos (que incluem a interface) para representar suas ideias sobre o que o sistema deve fazer e como ele vai apoiar o usuário durante a análise de requisitos. Essas decisões sobre a interação e a interface com usuário deveriam ser resultado de um projeto cuidadoso sob a responsabilidade de um profissional com conhecimentos de IHC, e não um subproduto das representações utilizadas durante a análise de requisitos. Os requisitos de IHC e as metas de design devem ser definidos com os requisitos do sistema durante a atividade de especificação de requisitos em um processo de ES. Desse modo, idealmente o design de IHC deveria ser iniciado durante a especificação de requisitos, mas sob a responsabilidade de um profissional de IHC. E mesmo após a definição da interação e da interface, e durante o projeto e a implementação do sistema, pode ser necessário também rever o projeto de IHC quando surgir alguma dificuldade para construir a solução projetada ou quando algo não tiver sido completamente especificado.

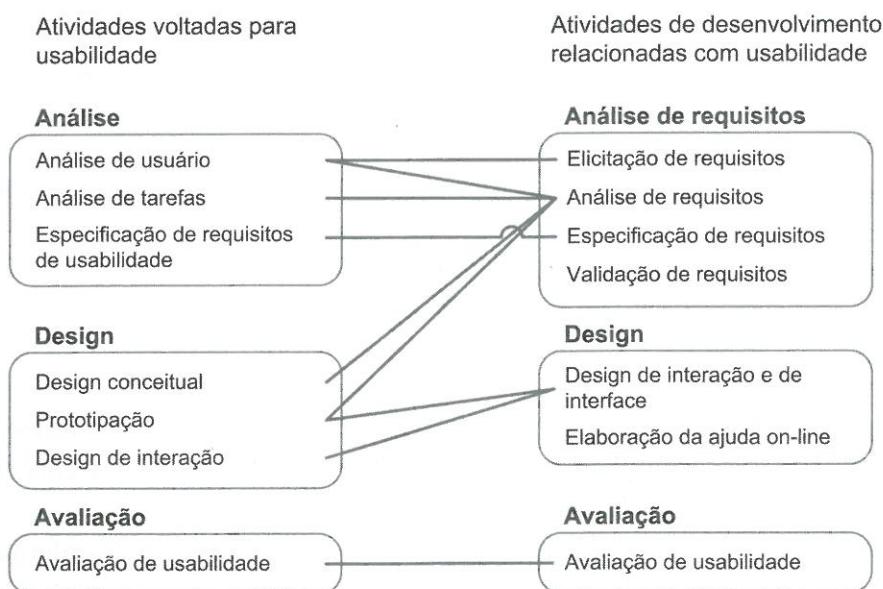


Figura 4.12 Mapeamento entre atividades de IHC e de ES adaptado de Ferre, 2003).

A avaliação do sistema deve ir além da validação de requisitos do sistema e testes do sistema, e englobar também uma avaliação de IHC da solução proposta, considerando os critérios de qualidade de uso definidos ao longo do processo de design e desenvolvimento. Sempre que possível, representantes dos usuários devem participar da avaliação de IHC do sistema para que a equipe de desenvolvimento possa perce-



ber os problemas que ocorrem durante o uso e coletar a opinião dos usuários sobre o sistema.

Como em qualquer equipe multidisciplinar, quando engenheiros de software e profissionais de IHC trabalham em conjunto, podem surgir problemas de comunicação, colaboração e coordenação. Parte do trabalho desses profissionais é solucionar o mesmo problema: definir a interação e a interface com usuário do sistema interativo sendo desenvolvido. Entretanto, eles possuem formações distintas, diferentes vocabulários, objetivos, perspectivas e focos na resolução desse mesmo problema. Portanto, é importante que os engenheiros de software e os profissionais de IHC reconheçam e valorizem o conhecimento, as preocupações e o trabalho uns dos outros. As atividades desses profissionais inevitavelmente influenciam umas as outras e, desse modo, devem estar bem coordenadas para produzirem um resultado consistente e coerente. A decisão de como será a interação e a interface deve ser tomada em conjunto pelos diferentes profissionais que compõem a equipe de design, sempre ponderando os diferentes pontos de vista e as questões levantadas por todos os membros da equipe de design.

O desenvolvimento de um sistema interativo exige muitos conhecimentos para cuidar adequadamente da sua construção e do seu uso, e envolve várias atividades realizadas em perspectivas diferentes do mesmo problema. Sendo assim, dificilmente um profissional será capaz de cuidar sozinho desses dois interesses, seja porque ele não possui os recursos necessários para adquirir conhecimento das duas áreas na profundidade e amplitude exigidas, seja porque ele não consegue articular ao mesmo tempo dois interesses tão distintos.

#### 4.5 Métodos Ágeis e IHC

Os métodos ágeis de desenvolvimento de software, como o eXtreme Programming (Beck, 2000) e Scrum (Schwaber e Beedle, 2002), podem ser interessantes para IHC porque buscam colaborar com o cliente (*customer*) através de pequenos ciclos de desenvolvimento de forma iterativa e incremental, para obter retorno (*feedback*) do cliente e corrigir o rumo do processo de desenvolvimento (Armitage, 2004; Blomkvist, 2005). Contudo, ainda carecem de cuidado adequado em relação à qualidade de uso. Segundo Armitage (2004, p. 18), “a comunidade de métodos ágeis raramente menciona os usuários ou a interface com usuário como um todo, o que significa que ou eles negligenciam a experiência de uso, ou estão focando projetos com menor necessidade de uma experiência de uso sofisticada”.

Quando se trata de métodos ágeis, nem sempre existe uma distinção entre clientes e usuários do sistema sendo desenvolvido. Em IHC, é fundamental fazer essa dis-



tinção. Os clientes (que não os usuários) possuem apenas uma visão limitada e frequentemente equivocada das atividades dos usuários propriamente ditos.

Quando clientes ou usuários são envolvidos nos processos de desenvolvimento ágeis, existe um grande risco de eles serem os responsáveis por especificar os requisitos do sistema e por projetar a interação e a interface (Jokela e Abrahamsson, 2004). Apesar de conhecerem bem o domínio, eles podem não ter os conhecimentos necessários sobre a tecnologia e sobre design para realizarem essas atividades. Além disso, podem ter uma visão restrita do domínio, limitada pela sua atuação em apenas parte do processo que precisa ser apoiado pelo sistema.

O desenvolvimento incremental proposto pelos métodos ágeis exige modificações na interface a cada incremento construído. Isso dificulta a manutenção da consistência e coerência da interação e da interface, afetando direta e significativamente a qualidade de uso. Quando se trata de qualidades relacionadas com a construção do sistema, os testes automáticos do sistema são uma ferramenta adequada e eficiente para manter a qualidade do código quando um novo incremento for desenvolvido. Entretanto, esses tipos de testes não são adequados para avaliar a qualidade de uso (Blomkvist, 2005). Mesmo em um processo de desenvolvimento ágil, ainda se faz necessário avaliar a interação e a interface de acordo com os critérios de qualidade de uso, com a participação de usuários reais.

Blomkvist (2005) faz algumas sugestões concretas para integrar métodos e práticas de IHC em processos de desenvolvimento ágil:

- o objetivo principal é entregar ao cliente software que *funcione* e que seja *usável*. Atividades de IHC são importantes, mas não podem consumir muito tempo. Devemos equilibrar o tempo necessário para entregar um sistema que funcione com a qualidade de uso oferecida;
- é comum existir a necessidade de *priorizar as funcionalidades* que o sistema deve possuir para permanecer dentro do prazo disponível. Os usuários indicam de quais funcionalidades precisam, mas o designer de IHC deve auxiliá-los nessa decisão, para melhor atender aos seus objetivos e promover uma alta qualidade de uso;
- *envolver ativamente os usuários* e não somente os clientes em todas as fases do desenvolvimento. O designer deve buscar informações sobre o contexto de uso, e não apenas consultar os usuários e clientes no ambiente de desenvolvimento;
- o *designer de IHC* deve ser responsável pelas decisões relacionadas com a qualidade de uso;

- é necessário *realizar avaliações de IHC* durante diferentes estágios do ciclo de desenvolvimento. Essas avaliações não podem ser executadas com a mesma frequência que testes automáticos de funcionalidades. Entretanto, sempre que possível, devemos executar avaliações de IHC mais simples e rápidas, idealmente com a participação dos usuários;
- devemos realizar uma *análise da situação atual* mais abrangente e rica em contexto de uso do que as histórias de uso (*user stories*) e os casos de uso (*use cases*) amplamente utilizados em métodos ágeis.

Vimos neste capítulo que o design de IHC é um processo iterativo que revisa e refina interpretações sobre uma situação para realizar uma intervenção na forma de um sistema computacional interativo. De modo geral, as atividades de um processo de design de IHC envolvem a análise da situação atual, a identificação do que deve ou pode ser melhorado, a elaboração e a avaliação de uma intervenção nessa situação. Os processos de design de IHC organizam essas atividades de diferentes maneiras, sugerem ou prescrevem os artefatos consumidos e produzidos em cada uma delas, e orientam como cada atividade deve ser executada.

O restante deste livro apresenta algumas técnicas e representações utilizadas ao longo do processo de design. O Capítulo 5 descreve técnicas de análise, o Capítulo 6 descreve representações que organizam o espaço de problema, o Capítulo 7 descreve a atividade de design de IHC, o Capítulo 8 apresenta diretrizes e padrões de IHC, e os Capítulos 9 e 10 descrevem a atividade e os métodos de avaliação de IHC.

## Atividades

1. *O que é design.* Escolha uma situação cotidiana em que é preciso realizar uma atividade de design explorando a criatividade. Por exemplo, comprar uma roupa ou calçado, preparar uma refeição ou planejar as férias. Analise a situação escolhida, identificando o que geralmente é feito na:
  - análise da situação atual;
  - definição das necessidades e oportunidades de intervenção (i.e., do que é possível melhorar na situação analisada);
  - proposta de uma intervenção;
  - avaliação da intervenção.
 Faça em seguida a mesma análise, visando construir um sistema que permite aos usuários atingir os mesmos objetivos.
2. *Processo de design de IHC.* Investigue um processo de design de IHC de um sistema interativo. Escolha um sistema a cuja documentação ou equipe de design você tenha acesso direto, ou ainda um sistema de software livre que disponibilize

na Internet o material gerado durante seu processo de design de IHC. Por exemplo, investigue o processo de design do OpenOffice, do Mozilla Thunderbird, do KDE ou do GNOME. Identifique, descreva e ilustre:

- as atividades executadas;
  - os artefatos consumidos e produzidos em cada atividade;
  - algumas decisões de design tomadas em cada atividade.
3. *Processos de design de IHC.* Discuta as principais diferenças entre os processos de design de IHC descritos neste capítulo. Identifique características dos projetos que ajudam a tomar decisões sobre quais processos podem ou devem ser adotados.
4. *Conhecimento de IHC envolvido nos processos de design.* Identifique quais conhecimentos de IHC precisam ser adquiridos por equipes de desenvolvimento para desempenhar atividades de IHC voltadas à qualidade de uso do sistema sendo projetado.
5. *Integração das atividades de IHC com engenharia de software.* Investigue alguma experiência de integrar atividades de IHC com atividades de ES, seja em um processo ágil ou não. Você pode conversar com colegas que tenham participado de iniciativas desse tipo ou ler relatos de iniciativas do gênero. Analise e discuta os pontos positivos e negativos da experiência de integração investigada.
6. *Inserção de atividades de IHC em processos de desenvolvimento de software.* Considerando um processo de desenvolvimento de software de que você tenha participado, identifique os pontos desse processo em que uma ou mais atividades de IHC voltadas à qualidade de uso poderiam ser realizadas.

# 5

## Identificação de Necessidades dos Usuários e Requisitos de IHC

### Objetivos do Capítulo

- Caracterizar o espaço de análise no processo de design de IHC.
- Descrever o planejamento da coleta de dados de análise em IHC.
- Discutir os aspectos éticos de pesquisas envolvendo pessoas.
- Apresentar técnicas de investigação e análise comumente utilizadas: entrevistas, questionários, grupos de foco, *brainstorming*, classificação de cartões, estudos de campo e investigação contextual.