

1

Introdução

O processo de desenvolvimento de sistemas interativos envolve basicamente as fases de elicitação de requisitos, análise, projeto, implementação, teste e implantação. Este processo conta com a participação de profissionais de várias áreas, cada um com o seu foco e objetivo. Dentre estas áreas, podemos citar a de interação humano-computador (IHC) e engenharia de software. Examinando a literatura destas áreas (Preece et al., 1994; Pressman, 2005), pode-se perceber que cada uma delas já possui um amadurecimento (em níveis diferentes) e teorias, métodos, processos, ferramentas e modelos que apóiam os seus profissionais nas suas atividades.

A área de IHC se concentra, de modo geral, em entender as características, necessidades e objetivos dos usuários da aplicação, o seu ambiente de trabalho e as tarefas que os usuários precisam ou desejam realizar através do sistema. Com isto definido, IHC parte para o projeto e prototipação da interface e interação humano-computador do sistema interativo, tendo como atividade constante a avaliação dos artefatos produzidos (Preece et al., 1994). É importante ressaltar que para IHC a interação é o processo de comunicação entre pessoas e sistemas interativos. Já a interface é toda a porção de um sistema com o qual o usuário mantém contato, englobando tanto software quanto hardware.

A área de engenharia de software tem como principal objetivo a especificação, implementação e teste das funcionalidades e arquitetura do sistema interativo. Alguns fatores de qualidade que são levados em consideração durante as atividades desta área são: modularidade (divisão do software em componentes para facilitar o seu gerenciamento); re-uso de código, artefatos, componentes, arquitetura e design, para diminuir custo de desenvolvimento e aumentar a qualidade; e performance (rapidez de processamento, tempo de resposta, consumo de recursos computacionais etc) (Pressman, 2005).

Mapeando algumas das atividades dos profissionais destas áreas nas fases do processo de desenvolvimento citadas acima (elicitação de requisitos, análise, projeto, implementação, teste e implantação), pode-se perceber que mais de uma área atua em uma mesma fase do processo de desenvolvimento (Tabela 1). É importante destacar que existem outras áreas, fases e atividades envolvidas no processo de desenvolvimento de um sistema interativo. Neste trabalho, estamos apenas fornecendo uma visão geral e simplificada dessas fases, pois não é o objetivo desta tese tratar o processo em si, como será visto mais adiante.

Fase/ Área	Elicitação	Análise	Projeto	Implementação	Teste	Implantação
IHC	Levantamento das necessidades e características dos usuários e seu ambiente de trabalho	Análise e modelagem dos usuários, suas tarefas e ambiente de trabalho	Projeto da interação humano-computador e da interface com o usuário e avaliação formativa		Teste de usabilidade	Teste de usabilidade
Eng. de Software			Projeto das funcionalidades internas do software e sua arquitetura	Construção do sistema interativo	Teste funcional	Instalação do sistema interativo

Tabela 1: Visão geral das atividades de cada área durante o processo de desenvolvimento.

Além de áreas distintas atuarem em uma mesma fase do processo de desenvolvimento, o trabalho realizado por cada uma delas afeta, direta ou indiretamente, o trabalho das outras, pois existe um objetivo comum: a criação de um sistema interativo que atenda bem as necessidades dos usuários da aplicação. Para que esse objetivo seja alcançado, a comunicação entre os profissionais dessas áreas durante o processo de desenvolvimento é fundamental para se criar um entendimento consistente do problema e do que deve ser construído, tentando evitar que cada profissional faça seu trabalho baseando-se em premissas diferentes uns dos outros e, além disto, diminuindo o retrabalho.

Esta questão da importância de uma comunicação eficiente entre os profissionais, e destes com os usuários durante a construção de um sistema já foi levantada em (Salles,

2001; Baranauskas et al., 2002). Estes autores argumentam que é importante se ter a visão geral da comunicação que ocorre entre os diferentes grupos em uma organização produtora de software para se poder investigar a qualidade do processo e do produto gerado. Segundo eles, a análise da comunicação pode indicar a “saúde” da organização. Com este objetivo, os autores propuseram um meta-modelo para se mapear as comunicações que ocorrem entre os membros da equipe de design e desenvolvimento durante um processo de desenvolvimento (modelo fractal) e um método de inspeção desta comunicação, o percurso comunicativo. Este método percorre o modelo fractal de uma equipe e detecta problemas na comunicação durante o processo de design e desenvolvimento.

Nesta tese, vamos focar a comunicação entre os profissionais de IHC e os engenheiros de software. Mais especificamente, entre os profissionais de IHC responsáveis pela especificação do conteúdo da interação que pode ocorrer entre o usuário e o sistema interativo e os engenheiros de software responsáveis pela especificação das funcionalidades do software. Desta forma, não está no escopo deste trabalho a comunicação, por exemplo, entre implementadores de software e projetistas de interface gráfica.

Como foi visto no início deste capítulo, IHC e engenharia de software “atacam” perspectivas diferentes na construção do sistema interativo para atender às necessidades do usuário final, isto é, estas áreas se concentram em aspectos diferentes do desenvolvimento de software. No trabalho de Walenstein (Walenstein, 2003) é defendida esta separação de perspectivas entre essas duas áreas, usando o argumento de que, como o foco de cada uma é bem distinto, a separação permite: o treinamento focado de cada tipo de profissional, o uso de técnicas e métodos específicos para a necessidade de cada área, e a exploração, por parte dos pesquisadores, de novas direções não exploradas em outras áreas. Baseando-se nisto, ele ressalta a necessidade de se criar uma ponte entre as duas áreas através do uso de representações compartilhadas por ambas. Para ele, estas representações devem ser fundamentadas na mesma teoria de design¹ para que se mantenha a coerência entre elas.

¹ O termo “teoria” no trabalho deste autor foi utilizado de forma imprecisa, e talvez até equivocada, pois ele chama artefatos de IHC, por exemplo, de teoria.

Nesta mesma linha de pensamento, isto é, manter as áreas independentes e se criar uma representação para apoiar a comunicação entre elas, existe o trabalho de Pyla e co-autores (2003). Segundo eles, cada área possui seu próprio ciclo de desenvolvimento, isto é, o trabalho de projeto de IHC envolve as fases de análise de requisitos, design da interação, prototipação, especificação da interface, implementação e avaliação e a área de engenharia de software contém as fases de análise de requisitos, design preliminar, design detalhado, design de especificações, implementação, integração, teste e avaliação (ciclos iterativos). Devido a esta separação, mas ao fato de o objetivo final ser o mesmo, isto é, a criação de um sistema que atenda às necessidades do usuário, é preciso, segundo os autores, a colaboração e a coordenação entre estes processos de desenvolvimento. Eles propõem que os profissionais de IHC (os autores falam em engenheiros de usabilidade) e os engenheiros de software, ao utilizarem a representação² que fará a ponte entre as duas áreas, lancem mão de um filtro. Isto é, o profissional de IHC, ao consultar a representação através do uso do filtro, só visualizará as questões de engenharia de software que impactarem o seu trabalho, e vice-versa. Através desta representação, cada tipo de profissional poderá ver as atividades e o status do seu próprio ciclo de desenvolvimento e do outro tipo de profissional, o cronograma, as técnicas já empregadas e a serem empregadas, os artefatos gerados e a serem gerados, e o mapeamento entre os domínios de trabalho. Todas estas informações passarão pelo filtro utilizado pelos profissionais envolvidos.

Os autores citam alguns tipos de comunicação que serão promovidas pelo uso da representação, como por exemplo, os profissionais de IHC passarem para os engenheiros de software a especificação das tarefas do sistema, para que estes possam fazer a decomposição funcional das atividades. Entretanto, o trabalho deles não entra em detalhes sobre o filtro a ser utilizado por cada profissional, isto é, o que cada um poderá/deverá visualizar de informações sobre o trabalho do outro. Além disto, como foi descrito, os autores citam tipos de informação bastante diferentes a serem compartilhadas, como por exemplo, informações sobre gerência de projeto (ex.: cronograma), ciclo de desenvolvimento e artefatos já produzidos. Nesta tese vamos focar e trabalhar a

² Esta representação é chamada de modelo de processo.

comunicação sobre a solução do projeto da interação, ou seja, apenas o compartilhamento de artefatos que contenham essa informação.

Estas iniciativas descritas acima mostram que tanto engenheiros de software como profissionais de IHC já atentaram para a necessidade de uma maior aproximação entre as áreas³. Esta pesquisa tem o objetivo de contribuir para esta aproximação, e como ressaltado por Walenstein (a necessidade de se basear em teorias de design), este trabalho se fundamenta em uma teoria de IHC – a teoria da engenharia semiótica.

1.1 Motivação Teórica

A engenharia semiótica descreve e explica a interação usuário-sistema como um processo de comunicação entre pessoas (projetistas de software, usuários, fabricantes) através de software (de Souza, 2005). Este fenômeno comunicativo acontece, pois os artefatos de IHC são construídos racionalmente, de acordo com as tomadas de decisão dos designers e são comunicados por eles aos usuários através de signos⁴ que devem ser interpretados, aprendidos, usados e adaptados para vários contextos. Devido a isto, esta teoria tem como objetivo apoiar a produção deste discurso, pelos designers, que apresenta os sistemas computacionais para os usuários.

Durante este discurso, o designer⁵ de IHC deve apoiar os usuários no alcance de seus objetivos através do sistema, comunicando-lhes: as metas que eles podem alcançar através do sistema, os vários caminhos que eles podem percorrer para alcançarem as metas, os signos de interface que podem ser utilizados em um caminho de interação para se atingir uma meta, e os signos que contam aos usuários as respostas do sistema durante a interação (de Souza, 2005, p.111). O designer, através do seu discurso, deve explicitar para os

³ Existe um *Working Group* da IFIP (*International Federation for Information Processing*) (IFIP, 2007) que promove uma série de *workshops* durante as principais conferências da área de IHC e engenharia de software, com o objetivo de aproximar estas duas áreas. Esse grupo de trabalho é o “IFIP WG 2.7/13.4 on user interface engineering: Bridging the SE & HCI Communities”(IFIP WG 2.7/13.4, 2007).

⁴ Signo é qualquer coisa que signifique algo para alguém (Peirce, 1931-1958).

⁵ Neste trabalho, o termo designer está sendo usado para representar a equipe de design, podendo ser uma ou mais pessoas.

usuários a lógica de design que está por trás do que está sendo apresentado, para que estes possam fazer uma interpretação do discurso compatível com o que foi definido pelo designer. A capacidade do designer em fazer esta comunicação se reflete na comunicabilidade do sistema (Prates et al., 2000).

Para apresentar este discurso para os usuários, o designer deixa na interface o seu representante – o preposto do designer, e é com ele que o usuário irá interagir durante toda a utilização do sistema interativo. Então, ao se projetar a interação, o designer deve focar esta comunicação do seu preposto com o usuário. Por isto a teoria ressalta que a comunicação é entre pessoas via software.

Como já foi dito no início deste capítulo, durante o processo de desenvolvimento de software, existem outros profissionais envolvidos na construção do sistema interativo, por exemplo, engenheiros de software. Estes também estão envolvidos na construção deste discurso interativo, pois eles são responsáveis por especificar as funcionalidades internas do software e fazer toda a implementação do sistema interativo (inclusive do discurso interativo).

Como foi dito na introdução desta tese, este trabalho irá focar a comunicação entre os profissionais de IHC responsáveis pelo projeto da camada de interação, ou seja, de toda a comunicação que pode ocorrer entre o usuário e a aplicação, e os engenheiros de software, responsáveis pela especificação da camada de aplicação do sistema interativo (Figura 1). A camada de interface e questões de implementação estão fora do escopo deste trabalho.

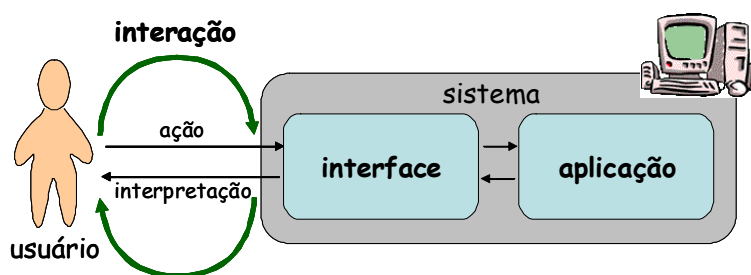


Figura 1: Sistemas interativos: camada de interação, interface e aplicação.

No momento da construção da camada de aplicação, o engenheiro de software não deve tomar decisões incompatíveis ou que afetem negativamente o discurso definido pelo

designer de IHC, pois isto irá impactar na interação do usuário com o sistema interativo. Um exemplo simples disto é o engenheiro de software, no momento de definir uma funcionalidade interna ligada a uma meta do usuário, alterar um caminho de interação para se atingir esta meta, definido pelo designer de IHC. Isto pode comprometer, por exemplo, a consistência das possíveis conversas que o usuário pode travar com a aplicação. E, por outro lado, o profissional de IHC não pode propor soluções que, por exemplo, não podem ser implementadas utilizando a tecnologia e/ou arquitetura definida pelo engenheiro de software.

Jerome e Kazman (2005) apresentam uma pesquisa realizada com profissionais de IHC e engenheiros de software (96 pessoas no total). Dentre outros dados, a pesquisa obteve como resultado o seguinte: 68% dos engenheiros de software (que responderam este item da pesquisa) disseram que decisões de projeto do software que afetam o projeto da interface são tomadas por engenheiros de software sem consultar profissionais de IHC. E 91% dos profissionais de IHC (que responderam este item da pesquisa) acreditam que engenheiros de software tomam decisões cruciais sem consultar os profissionais de IHC.

Então, para evitar que a camada de aplicação seja incompatível com o que foi (ou está sendo) pensando pelo designer de IHC e vice-versa, é importante que o engenheiro de software tome conhecimento do projeto da interação usuário-preposto do designer tão cedo quanto possível, para que as decisões de projeto possam ser negociadas entre estes dois tipos de profissionais. Desta forma, o discurso do designer de IHC estará sendo compartilhado com o engenheiro de software. É importante lembrar que o objetivo final é a qualidade de uso do produto gerado.

1.2 Proposta deste Trabalho

Para que o trabalho do designer de IHC seja transmitido para os engenheiros de software e possa haver uma negociação sobre as tomadas de decisão, propõe-se nesta

pesquisa o uso de uma ferramenta⁶ que represente o projeto da comunicação do usuário com o sistema interativo, ou seja, o conteúdo da camada de interação. Esta ferramenta envolve um conjunto estruturado de signos, produzido (fruto de tomadas de decisões) e comunicado por um emissor (designer de IHC) a um receptor (engenheiro de software). Este conjunto de signos deve ser transmitido de forma eficiente para que a sua interpretação pelos engenheiros seja compatível com a intenção do designer de IHC. Esta comunicação é privilegiada em relação à do usuário com o preposto do designer, pois neste caso o emissor se encontra presente durante a comunicação.

O conteúdo desta ferramenta está descrito na próxima seção e o capítulo 3 apresenta como este conteúdo deve ser estruturado e apresentado aos engenheiros de software.

1.2.1 Definição do Conteúdo da Ferramenta

Como foi visto, a engenharia semiótica ressalta que a interação usuário-sistema é um processo de comunicação entre o designer da aplicação e o usuário. Nesta conversa, o designer envia ao usuário uma mensagem onde ele apresenta uma possível solução para as necessidades e as preferências dos usuários, que foram identificadas por ele durante a análise e o projeto de IHC. Esta mensagem pode ser parafraseada da seguinte forma:

"Esta é a minha interpretação sobre quem você é, o que entendi que você quer ou precisa fazer, de que modos preferenciais, e por quê. Este é portanto o sistema que eu projetei para você, e esta é a forma como você pode ou deve usá-lo para conseguir atingir uma série de objetivos que se enquadram nesta minha visão." (de Souza, 2005, p. 84)

O objetivo da ferramenta proposta nesta tese é comunicar aos engenheiros de software o conteúdo do discurso interativo projetado pelo designer de IHC. Mais especificamente, objetiva-se comunicar:

- quem são os usuários da aplicação e o seu contexto [*"Esta é a minha interpretação sobre quem você é..."*];

⁶ Neste trabalho, "ferramenta" não se refere a uma ferramenta computacional, e sim a um instrumento de

- as metas que os usuários desejam alcançar através do sistema [*“...o que entendi que você quer ou precisa fazer...”*];
- o projeto da comunicação usuário-preposto do designer necessária para o alcance das metas dos usuários, e os motivos de a solução final ter sido proposta desta forma [*“...de que modos preferenciais e por quê. Este é portanto o sistema que eu projetei para você, e esta é a forma como você pode ou deve usá-lo para conseguir atingir uma série de objetivos que se enquadram nesta minha visão.”*].

Como, em sistemas não extensíveis, a mensagem é imutável e contempla o sistema como um todo, o projeto da comunicação usuário-preposto inclui a visão completa de todas as possíveis conversas que podem ser travadas ao utilizar o sistema interativo. Ou seja, ele inclui as conversas “desejáveis”, isto é, os caminhos preferenciais projetados pelo designer para o alcance de determinada meta, os caminhos alternativos e o tratamento de possíveis problemas que podem ocorrer durante a conversa usuário-preposto do designer. Desta forma, é importante que haja um entendimento abrangente de toda esta comunicação que pode ocorrer entre o usuário e o preposto do designer. Então, para facilitar este entendimento, devemos apresentar para os engenheiros de software o projeto de IHC de uma forma menos fragmentada e mais global, ou seja, apresentar um mapa que explicita todas estas possíveis conversas que podem ser travadas durante o uso da aplicação.

A comunicação dos itens listados acima para os engenheiros de software não contém o mesmo conteúdo da mensagem que é enviada via interface para os usuários da aplicação. Como será visto mais adiante, ela é direcionada para o que os engenheiros de software precisam saber sobre o conteúdo do projeto de IHC, e o que está por trás dele. O objetivo dessa comunicação é o entendimento pelos engenheiros de software do que deve ser comunicado, e por quê, para os usuários através da interface da aplicação.

É importante ressaltar que esta ferramenta pretende comunicar o **conteúdo** da mensagem e não a **expressão** desta mensagem, ou seja, como ela será apresentada em termos de elementos de interface concreta (telas, *widgets*, gráficos, etc) para o usuário final. O objetivo é que primeiramente o engenheiro de software entenda o que deve ser

transmitido para o usuário via interface, sem se preocupar inicialmente com a forma de apresentação deste conteúdo. A idéia é o compartilhamento de todos os possíveis caminhos de interação que o usuário poderá percorrer durante a utilização da aplicação, todas as ações que ele poderá executar e em qual sequência, todas as possíveis respostas do sistema a essas ações e, além disto, os motivos subjacentes a estas escolhas de design.

Desta forma, em um segundo momento, o engenheiro de software também precisará, durante o seu trabalho de especificação e implementação das funcionalidades do software, de uma representação que contemple a expressão deste conteúdo (por exemplo, um conjunto de telas ou um protótipo de interface).

1.3 Organização da Tese

O próximo capítulo deste trabalho (capítulo 2) apresenta os trabalhos relacionados a esta pesquisa, dando destaque à motivação para se criar uma nova forma de se apoiar a comunicação entre os designers de IHC e os engenheiros de software. O capítulo 3 apresenta a ferramenta ComunIHC-ES, criada com o objetivo de transmitir o projeto da interação para os engenheiros de software, e apoiar a comunicação destes com os designers de IHC. Esta ferramenta é fundamentada na teoria da engenharia semiótica descrita neste capítulo. O capítulo 4 descreve um estudo de caso com engenheiros de software, desenvolvido para analisar a utilidade da ferramenta ComunIHC-ES. No capítulo 5 é apresentada uma discussão sobre os resultados desta pesquisa. E por fim, o capítulo 6 apresenta as considerações finais e possibilidades de continuidade desta pesquisa.

Nos apêndices estão registradas informações importantes sobre a pesquisa realizada. No Apêndice A, encontra-se a modelagem do domínio “Círculo de livros” utilizando a ferramenta ComunIHC-ES. Esta modelagem serviu de exemplo para a explicação da ferramenta ComunIHC-ES e para o estudo de caso realizado com engenheiros de software. O Apêndice B apresenta o termo de consentimento para participação no estudo de caso realizado, que foi entregue pelo avaliador do estudo e assinado pelos engenheiros de software. No Apêndice C está disponível o material utilizado no minicurso ministrado para

os engenheiros de software durante o estudo de caso. E, por último, o Apêndice D apresenta uma visão resumida dos principais resultados do estudo de caso.