

4

Descrição do processo ágil proposto

Neste capítulo, definimos o processo ágil proposto nesta dissertação, o eXCeeD. Para tanto, descrevemos na próxima seção os valores e práticas do eXCeeD e, em seguida, as atividades envolvidas neste processo.

4.1.

Valores e práticas do eXCeeD

Análogo ao XP e fundamentado na teoria da EngSem, o eXtreme Communication-Centered Design proposto neste trabalho visa agilizar o projeto da interação humano-computador, ou seja, a construção da metamensagem de sistemas computacionais que envolvam equipes pequenas de designers da interação, contendo de duas a três pessoas. Para defini-lo, vamos nos apoiar nas idéias propostas para os métodos ágeis de desenvolvimento, mais especificamente no XP, adaptadas ao processo de projeto da interação. Em seguida, analisamos e adaptamos os valores e práticas do XP para a o projeto da interação ancorado na EngSem, para que a construção da metacomunicação seja feita de maneira adequada.

Como queremos enfatizar a importância da relação entre os membros da equipe de designers, a colaboração com o usuário e a característica de mudanças inerente ao projeto de interface do usuário, sugerimos manter os seguintes valores dos métodos ágeis na forma como propostos inicialmente:

- **indivíduos e interações** mais do que processos e ferramentas
- **colaboração com o cliente** mais do que negociação de contratos
- **responder a mudanças** mais do que seguir um plano

Como o processo ágil proposto é focado no projeto da interface do usuário de um software e não na sua implementação, deveremos adaptar o valor **software funcional** mais do que documentação abrangente, modificando-o para **esboços e simulação da interface** mais do que documentação abrangente, enfatizando esta diferença. Assim, mais do que uma documentação abrangente, no eXCeeD, a medida de produção de resultados do projeto da interface serão os esboços produzidos e a obtenção de *feedback* por parte dos usuários será realizada através da simulação da interface proposta, com base no diagrama de interação da MoLIC. A maneira como esta simulação se dará será descrita mais adiante neste capítulo, durante a definição das atividades do processo. Da mesma forma que para os métodos ágeis, a pouca documentação deverá ser compensada pela comunicação presencial entre os membros da equipe de projeto e os usuários.

Depois de adaptar os valores dos métodos ágeis ao eXCeeD, o processo é fundamentado nos valores de comunicação, simplicidade, *feedback* e coragem do processo de desenvolvimento de software XP. No eXCeeD, esses valores são adaptados ao contexto de processo para o projeto da interação proposto, como descrito a seguir:

- **Comunicação**

Assim como a comunicação face-a-face é uma atividade constante no XP, ela o deverá ser também para o eXCeeD. Como este processo está sendo proposto para o projeto de interface de sistemas computacionais realizado por equipes de designers pequenas, a pouca especificação proposta pelas descrições de metas do usuário e pelos diagramas de interação MoLIC deverá ser compensada pela comunicação face-a-face entre os membros da equipe de projeto e os usuários presentes durante todo o processo. Assim, designers e usuários deverão trocar idéias durante todo o projeto da interface. Contudo, vale lembrar que os designers são os responsáveis últimos pelas decisões que serão incorporadas ou descartadas do projeto;

- **Simplicidade**

Para agilizar o processo de projeto da interação, semelhante ao que acontece no XP, é recomendável projetar a solução que for mais simples para o momento, deixando-se de lado soluções mais complexas e que ataquem problemas futuros que ainda não se sabe ao certo se surgirão ou não. Depois, caso uma solução mais complexa se mostre realmente necessária, será possível modificar o projeto do sistema de maneira que as novas soluções se adequem às necessidades que surgirem;

- **Feedback**

Assim como os métodos ágeis, o eXCeeD enfatiza a forte colaboração entre a equipe de designers e os usuários do sistema durante todo o projeto da interação. À medida que a equipe de designers vai aprendendo mais sobre o sistema que estão projetando, seja pela elaboração das descrições das metas dos usuários ou pela modelagem da interação e da interface, novos questionamentos surgem e, conseqüentemente, novas soluções podem ser necessárias. Como os usuários estarão disponíveis durante todo o processo, eles poderão colaborar com a equipe de designers, fornecendo suas necessidades, visões e opiniões a respeito dos rumos que o projeto deverá seguir. Além disso, durante a atividade de avaliação dos protótipos junto aos usuários, espera-se que o *feedback* obtido deles forneça insumos para a melhoria do fluxo de interação e interface sendo projetados; e

- **Coragem**

Para trabalhar com o processo ágil proposto também será necessário ter coragem. Primeiramente, coragem para modificar parte do projeto da interação em caso de detecção de problemas, de uma nova solicitação do usuário ou para torná-lo mais simples, caso seja necessário. Também será essencial coragem para explorar várias alternativas de interface, mesmo que algumas delas não sejam incorporadas ao final do projeto ou para refazer parte do projeto da interação novamente.

Da mesma maneira como o XP, e inspirado nele, o eXCeed propõe um conjunto de práticas para apoiar esses valores. A seguir, apresentaremos cada uma delas:

- **Planejamento mínimo**

Assim como no XP, a execução do projeto da interação seguindo o eXCeed deve ter um planejamento inicial mínimo, mas que seja facilmente adaptável a mudanças. Este planejamento deve conter a ordem em que as descrições de metas do usuário deverão ser projetadas e quando um determinado conjunto delas deverá ser simulado junto aos usuários. Por exemplo, se o sistema possui um total de quinze descrições de metas do usuário⁸, os designers e usuários podem decidir que três destas metas fortemente relacionadas entre si serão projetadas em quatro dias, outras duas em mais três dias e assim sucessivamente. Ao final do prazo estipulado, o projeto para cada grupo de metas deverá ser mostrado e simulado junto aos usuários que participam do projeto da interface do sistema. É interessante ressaltar que todas estas decisões devem ser estabelecidas pela equipe de designers e os usuários em comum acordo. Além disso, qualquer modificação que se mostre necessária no planejamento firmado também deverá ser decidida sob aprovação dos usuários;

- **Reestruturação**

Com o decorrer do projeto, conforme vão aprendendo e adquirindo um maior conhecimento sobre o sistema, os usuários podem solicitar novas funcionalidades que poderão ser incorporadas ao projeto, caso os designers julguem necessário. Conseqüentemente, as soluções correntes podem ser modificadas, para que elas possam acomodar as

⁸ As descrições de metas do usuário serão definidas em uma fase do eXCeed denominada elaboração de metas do usuário, conforme veremos na Seção 4.2.1.

novas exigências dos usuários. Um processo de aprendizado semelhante ocorre com a equipe de designers durante o projeto do sistema: à medida que adquirem conhecimento sobre o sistema, eles vão refinando as suas soluções, de maneira a aprimorar as soluções obtidas anteriormente. Esta característica, inerente ao projeto da interação de um artefato computacional, está em linha com a EngSem, conforme afirma de Souza (2005): *“quanto mais os usuários pensam sobre o problema, mais eles compreendem este problema. Assim, novos insights surgem constantemente da sua reflexão”* (p. 33).

Neste ponto, vale destacar novamente que o projeto da interface do sistema é de responsabilidade da equipe de designers, pois é ela quem detém o conhecimento necessário para fornecer uma solução adequada às necessidades identificadas dos usuários. Logo após uma modificação, será necessário simular novamente a porção dos esboços de interface com o intuito de verificar se aquela determinada parte alterada está de fato adequada ao restante do projeto de interface proposto;

- **Design de interface simples**

Para cada meta do usuário, os diagramas de interação MoLIC e os esboços de interface devem apresentar uma solução que resolva o problema de forma imediata, sem pensar muito na solução de problemas que possam vir a ocorrer no futuro. Com o decorrer do projeto, se soluções mais simples que a atualmente proposta forem pensadas, elas poderão ser incorporadas ao projeto;

- **Soluções alternativas**

Durante o projeto do sistema, a equipe de designers será incentivada a propor alternativas de interface para o sistema. Assim, cada membro da equipe poderá propor uma ou mais alternativas para uma mesma descrição de meta ou para uma determinada parte do sistema. Esta prática funciona como um *brainstorming* de idéias, criando um

conjunto de soluções alternativas que, com o decorrer do projeto, podem ser (i) escolhidas e refinadas, dando continuidade a uma das soluções propostas, (ii) reservadas, aguardando um momento oportuno no qual elas possam ser utilizadas, ou ainda (iii) descartadas;

- **Metáforas**

Metáforas são amplamente utilizadas no nosso cotidiano, muitas vezes sem sequer notarmos (Erickson, 1990). Elas são extremamente importantes na criação de novos conceitos a partir de conceitos antigos, expressando algum tipo de relação que o primeiro tem com o último (de Souza, 2005). Em IHC, o uso de metáforas é bastante comum desde que as interfaces gráficas do usuário foram criadas (ibid.). A metáfora do *desktop* é citada como o exemplo mais famoso de metáfora de IHC (Erickson, 1990; de Souza, 2005).

Erickson (1990) aponta que uma das maneiras de se identificar metáforas para um novo sistema é verificar quais metáforas já estão implícitas na descrição do problema que deve ser apoiado com o sistema sendo projetado. Além disso, ele defende também que é possível criar novas metáforas para a interface do sistema a partir dos problemas dos usuários, observando acontecimentos e objetos do mundo real que possam incorporar soluções para estes problemas. Com base nesta idéia, a prática das metáforas utilizada no eXCeed propõe que, durante a conversa com o usuário, os designers percebam eventos, informações ou objetos do mundo real que possam servir como base para a construção de metáforas de interface nas soluções de design propostas;

- **Design coletivo**

A programação em pares facilita a disseminação do conhecimento entre os membros da equipe de desenvolvimento no XP. Com este mesmo intuito, será adotado o design coletivo. Durante a concepção do projeto de interface proposto, ou seja, no momento em que se

reflete sobre as possíveis soluções para a interface proposta, o trabalho em equipe facilitará a troca de idéias entre os designers participantes do projeto. Esta estratégia permitirá que todos os membros da equipe de projeto participem ativamente do projeto de interface e tenham conhecimento sobre as soluções propostas. Desta maneira, espera-se que esta prática contribua para a coerência e consistência do projeto elaborado, uma vez que o conhecimento compartilhado das soluções propostas facilita a adoção de soluções semelhantes para problemas semelhantes. Ademais, todos os designers terão autonomia para modificar qualquer parte do projeto feito por outros membros da equipe de projeto, com a finalidade de consertar ou refinar o projeto de interface proposto;

- **Simulação dos esboços**

A simulação dos esboços deverá ocorrer de duas maneiras no eXCeeD: por meta e por sistema. A simulação ocorrerá por meta ou por um conjunto de metas estreitamente relacionadas, pois a modelagem da interação e da interface é realizada exatamente dessa forma. Assim, depois que os diagramas de interação MoLIC e esboços de interface forem propostos como solução para uma determinada meta, é necessário simular o fluxo de interação descrito nos artefatos criados. O intuito desta atividade é verificar se, de fato, o modelo de interação e os esboços estão coerentes entre si. Depois, a simulação deve ocorrer não mais sobre uma única meta, mas sobre todo o sistema, para verificar a consistência da porção de esboços de interface construídos com o restante da aplicação proposta. Além disso, de acordo com os prazos estipulados durante o planejamento mínimo, partes da ferramenta deverão ser simuladas junto aos usuários para verificar se não há problemas de interação decorrentes da interface proposta;

- **Integração contínua**

Além das descrições de metas do usuário, o resultado do projeto de interface definido para o eXCeeD será composto por diagramas de

interação MoLIC abreviados e pelos esboços de interface. Para cada meta, devem ser considerados modelos de interação e de interface alternativos, cada qual representando uma forma diferente de se atingir a meta. Para garantir a consistência da interface, quando a modelagem da interação e da interface para uma nova descrição de meta for proposta, ela deverá ser encaixada num modelo geral. Depois da integração, é importante que se faça uma simulação dos esboços de interface para verificar se a parte adicionada está coerente com o restante do modelo proposto. Os diagramas de interação MoLIC e os esboços de interface estarão relacionados uns com os outros, ou seja, trechos do diagrama de interação vão corresponder a trechos dos esboços de interface. Desta maneira, a organização dos esboços de interface é guiada pela semântica geral da interação fornecida no conjunto de diagramas de interação;

- **Usuário presente**

Durante a elaboração das descrições de metas dos usuários e a avaliação dos esboços junto a eles, é imprescindível a presença de um ou mais usuários acompanhando estas atividades. Eles devem indicar quais são as suas reais necessidades e fornecer *feedback* sobre as soluções de interface propostas pela equipe de designers. Para a modelagem da interação e da interface, a presença do usuário também é importante. Porém, como os usuários possuem os seus próprios afazeres, muitas vezes não poderão ficar à disposição da equipe de designers em tempo integral. No entanto, eles deverão estar sempre disponíveis para retirar dúvidas ou lacunas que, porventura, surjam durante a execução desta atividade. Desta maneira, teremos uma colaboração constante entre a equipe de designers e os usuários para os quais o sistema está sendo construído. Uma vez que o usuário presente durante o projeto da interface não deverá ser o único usuário do sistema, ele representará diversos outros usuários deste sistema, falando em seu nome. Além disso, caso tenhamos diferentes perfis de usuário para o sistema, devemos envolver ao menos um representante de cada perfil para

que ele possa informar as particularidades necessárias para o projeto das metas definidas para o seu perfil. O envolvimento de apenas um usuário por perfil pode restringir a visão de conjunto necessária para o projeto do sistema;

- **Anotações regulares**

Avaliar constantemente o desenvolvimento de um projeto é fundamental para que o seu andamento seja satisfatório. Assim, depois da execução de cada uma das atividades, ou ao final de cada dia de trabalho, é importante fazer uma discussão rápida recapitulando as decisões que foram tomadas em cada uma das atividades e os problemas que ocorreram durante o projeto da interface. Após a discussão, um breve relato por escrito dos pontos mais importantes pode ser usado como um histórico ou uma avaliação regular do projeto, reforçando as decisões que foram tomadas no seu decorrer.

As práticas **testes** e **releases curtos** do XP foram substituídos por uma única prática, **simulação dos esboços**, uma vez que, no contexto do eXCeeD, testar os modelos construídos significa simular a interação através dos esboços de interfaces produzidos. Esta simulação será guiada pelo conjunto de diagramas de interação MoLIC, dado que eles fornecem a semântica da interação. Uma outra razão para tal modificação é que um *release* no eXCeeD também constitui-se na avaliação junto aos usuários dos esboços produzidos, sendo, portanto, uma simulação dos esboços. As práticas **programação em pares** e **código coletivo** do XP, por sua vez, não fazem sentido no contexto do processo proposto. Estas práticas foram substituídas pela prática **design coletivo**, que agrega as idéias propostas por elas em uma única. Já a prática **reuniões diárias** em pé, apresentada por Teles (2004), foi substituída pela prática **anotações regulares**, cujo intuito é realizar uma avaliação rápida do projeto ao final de cada atividade ou ao final de um dia de trabalho. Apesar de permanecer com a mesma idéia geral do **jogo do planejamento**, esta prática teve sua denominação modificada para **planejamento mínimo**, por se tratar apenas do agrupamento e priorização das metas a cada passo do design. Como o termo **refactoring** é utilizado exclusivamente para

implementação, e o eXCeeD trata de projeto, a prática com este nome foi substituída pela prática **reestruturação**. A prática **metáfora** foi substituída pela prática **metáforas**, a qual já é bastante comum entre designers de IHC. Com a definição desta prática, procuramos enfatizar a pluralidade de metáforas consideradas durante o projeto do sistema e a viabilidade de incorporação de uma ou mais metáforas à interface produzida. Como o foco desta dissertação é o projeto de interface e não o projeto do sistema como um todo, substituímos a prática **design simples** por **design de interface simples**, de maneira que ela representasse as nossas necessidades. Apesar de Beck afirmar, na prática **cliente presente**, que usuários devem colaborar durante o desenvolvimento do sistema, modificamos o título desta prática para **usuário presente**, objetivando enfatizar a colaboração deles no processo. Por outro lado, mantivemos o mesmo título para a prática **integração contínua**, em razão de esta prática ter sido definida nos dois processos com a mesma intenção: unir as diferentes partes do sistema em produção. No entanto, há uma diferença na forma como estas práticas são realizadas nas duas abordagens. No XP, os programadores constroem porções de código de maneira independente e depois integram estas porções verificando a integridade entre elas através da realização de testes no sistema. Já no eXCeeD, todos os designers constroem diferentes soluções e concordam com uma solução antes que ela seja integrada às soluções correspondente às metas anteriores, representada no diagrama MoLIC sendo construído incrementalmente. A verificação de integridade neste caso é realizada por inspeção do diagrama de interação construído até então.

A única prática acrescentada ao eXCeeD foi a prática intitulada **soluções alternativas**. Ela foi acrescentada ao conjunto de práticas do eXCeeD por se tratar de uma atividade comprovadamente importante no processo de design de interfaces (Tohidi et al., 2006; Buxton, 2007). Por fim, a prática **semana de 40 horas** está mais voltada para aspectos gerenciais do XP, enquanto a prática **código padronizado** diz respeito ao código produzido durante a implementação do sistema. Como tais aspectos estão fora do escopo desta dissertação, estas práticas não foram consideradas no eXCeeD. A seguir, na Tabela 2, apresentamos um resumo das modificações realizadas nas práticas do XP para a definição das práticas do eXCeeD.

Tabela 2 – Modificações realizadas nas práticas do XP para as práticas do eXCeeD.

Práticas no XP	Práticas no eXCeeD
Testes	Simulação dos esboços
Releases curtos	
Programação em pares	Design coletivo
Código coletivo	
Reuniões em pé	Anotações regulares
Jogo do planejamento	Planejamento mínimo
Refactoring	Reestruturação
Metáfora	Metáforas
Design simples	Design de interface simples
Integração contínua	Integração contínua
Cliente presente	Usuário presente
--	Soluções alternativas
Semana de 40h	--
Código padronizado	--

4.2. Atividades envolvidas no eXCeeD

Para agilizar o projeto da interação humano-computador, o eXCeeD é definido como um processo iterativo, onde constantemente o trabalho desenvolvido é reavaliado, de acordo com as necessidades e interesses dos usuários para os quais o sistema está sendo desenvolvido. Assim, durante a execução das atividades definidas para o eXCeeD, será possível perceber lacunas ou problemas que deverão ser revistos. Para tanto, os designers da interação envolvidos no projeto poderão voltar às atividades onde os problemas foram

detectados e corrigi-los. Em linha com o modelo de ciclo de vida simples apresentado por Sharp e co-autoras (2007), cada iteração é constituída por três atividades principais:

- elaboração das descrições de metas do usuário;
- modelagem da interação e da interface; e
- avaliação junto aos usuários,

conforme ilustrado na Figura 5.

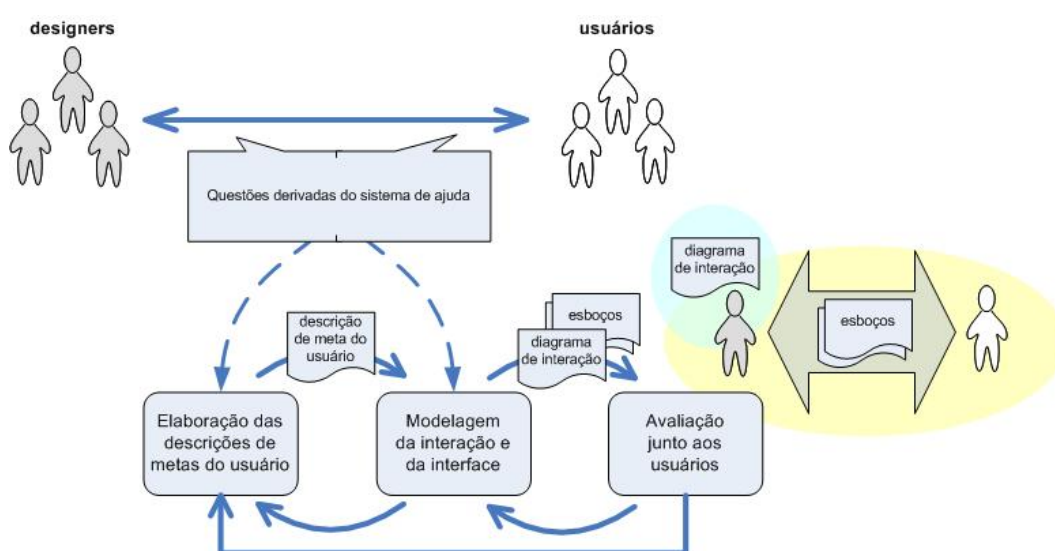


Figura 5 – eXtreme Communication-Centered Design.

O eXCeeD inicia com a atividade de elaboração das descrições de metas do usuário, onde são identificadas as necessidades e os objetivos dos usuários que devem ser apoiados com o sistema sendo projetado. Nesse momento, os designers se concentram na primeira parte da metamensagem: *“Eis a minha interpretação de quem você (usuário) é, o que aprendi que você tem de fazer, preferencialmente de que forma, e por quê.”* (de Souza, p. 84)

Em seguida, para cada descrição de meta elaborada, é feita a modelagem da interação através da construção de diagramas de interação MoLIC e a modelagem da interface através de esboços em papel. Nesse momento, então, os designers passam a trabalhar principalmente na segunda parte da metamensagem: *“Eis, portanto, o sistema que conseqüentemente concebi para você, o qual você pode*

ou deve usar assim, a fim de realizar uma série de objetivos associados com esta (minha) visão.” (ibid., p.84)

Os diagramas de interação e os esboços de interface produzidos estão intimamente relacionados, uma vez que o fluxo de interação descritos nestes diagramas fornece a organização para estes esboços. Esses artefatos são utilizados em seguida durante a atividade de avaliação junto aos usuários. Nesta avaliação, o fluxo de interação descrito nos diagramas de interação é utilizado como guia pelos designers para a simulação dos esboços que serão mostrados aos usuários. Com o *feedback* obtido desta avaliação é possível revisar iterativamente o projeto em construção, retornando a qualquer uma das atividades anteriores para corrigir as novas necessidades ou os problemas identificados. Para auxiliar as atividades de elaboração das descrições de metas do usuário e de modelagem da interação, o eXCeeD faz uso das expressões de comunicabilidade do sistema de ajuda (listadas na Seção 2.2.1) como base para a construção das questões que deverão guiar o processo de reflexão envolvido nestas atividades. Chamaremos tais questões de questões derivadas do sistema de ajuda e veremos mais adiante como foi feita essa derivação.

Como dissemos anteriormente, com este processo propomos unir o apoio à reflexão da teoria da EngSem com a agilidade dos métodos ágeis e da prototipação rápida através de esboços de interface. O apoio à reflexão será proporcionado pela elaboração coletiva da metacomunicação, apoiada pelas descrições de metas do usuário e pelo diagrama de interação MoLIC. O diagrama de interação será utilizado da maneira abreviada, como descrito na primeira etapa do projeto da interação da segunda edição desta linguagem proposta por Silva (2005). Além dos valores e práticas definidos na seção anterior, a agilidade no processo será fornecida pela simultaneidade das atividades de modelagem de interação e de modelagem da interface através de esboços.

Para auxiliar as atividades de elaboração das descrições de metas do usuário e modelagem da interação e da interface, utilizaremos as expressões propostas por Silveira (2002) para a construção dos sistemas de ajuda segundo a EngSem. De acordo com o trabalho apresentado por Barbosa e co-autoras (2004), trazer estas expressões auxilia o processo de reflexão do designer em tempo de projeto.

Depois de produzidos, os diagramas de interação MoLIC e esboços de interface serão utilizados para avaliação junto aos usuários. Resumidamente, esta avaliação constitui-se de uma simulação dos esboços guiada pelos diagramas de interação que representam todo o fluxo de interação proposto. Detalharemos nas seções seguintes as atividades do eXCeeD, os artefatos produzidos e consumidos, bem como as questões derivadas utilizadas em cada uma destas atividades.

4.2.1.

Elaboração das descrições de metas do usuário

É durante a elaboração das descrições de metas que os designers identificam as necessidades e as metas dos usuários. As metas de um usuário podem ser classificadas em duas categorias, as metas finais e as metas instrumentais (Silva e Barbosa, 2007). As metas finais são aquelas que constituem o objetivo final do usuário e podem ser representadas pela seguinte fala do usuário *“Eu quero utilizar o sistema para <atingir metaFinal>”*. Já as metas instrumentais auxiliam o atingimento da meta final. As metas instrumentais podem ainda ser classificadas em metas planejadas e oportunistas. As instrumentais planejadas podem ser representadas pela seguinte fala do usuário: *“Quero <atingir metaInstrumental> para <atingir metaFinal> [de forma mais eficiente|fácil|...]”*. As metas instrumentais oportunistas, por sua vez, surgem naturalmente durante a interação e podem ser representadas pela fala do usuário: *“Aqui de onde estou no sistema, vou <atingir metaInstrumental> para <atingir metaFinal> [de forma mais eficiente|fácil|...]”*. Por exemplo, suponha que um usuário tem como meta final *Salvar uma nova versão de um arquivo utilizando o Microsoft® Office Visio*. Para tanto, será necessário irmos ao menu **Arquivo > Salvar como...**, onde poderemos formular a meta instrumental **Criar nova pasta** para salvar a nova versão do documento em um novo diretório ou a meta instrumental **Renomear arquivo**, caso precisemos utilizar o nome de um arquivo existente no diretório selecionado.

Com base nestas definições, cada descrição de meta do usuário, como o próprio nome sugere, contém uma descrição de algo que o usuário quer fazer com o sistema sendo projetado. Apesar de terem um papel muito semelhantes às

estórias no XP, optamos por utilizar um outro termo, pois para a comunidade de IHC, uma estória remete ao conceito de cenários, que não são simples descrições de uma funcionalidade do sistema, mas narrativas contextualizadas de pessoas e suas atividades (Erickson, 1995; Carroll, 2000). Levando-se em consideração esta definição, se utilizássemos o termo estória no eXCeeD, elas deveriam obrigatoriamente incorporar não somente os objetivos que os usuários querem alcançar com o sistema sendo projetado, mas também uma narrativa contendo os personagens envolvidos no fluxo de eventos descrito, inseridos em um contexto de realização de determinada atividade. No eXCeeD, as informações de contexto serão discutidas durante todo o processo, e incorporadas nas soluções propostas. Essas informações podem ser registradas por escrito junto às descrições de algumas metas, quando julgado necessário, no entanto isto não é obrigatório

Durante esta atividade, serão identificadas as necessidades e exigências dos usuários através das conversas entre a equipe de designers e os usuários. Durante esta conversa, e à medida que as metas dos usuários vão sendo identificadas, as metas são registradas conforme os designers achem mais apropriado, contendo informações que julguem necessárias para a modelagem da interação e da interface que será feita em seguida. Neste momento, os designers estão preocupados em conhecer o usuário, as atividades que ele faz no mundo real, como ele realiza tais atividades e por quê.

Com o intuito de enriquecer a elaboração destas descrições, lançaremos mão do conjunto de expressões propostas para o sistema de ajuda, apresentado em (Silveira, 2002). Barbosa e co-autoras (2004) apresentaram, no CCD, um conjunto estendido de expressões de comunicabilidade, do qual faziam parte as expressões do sistema de ajuda. No CCD, conforme comentado anteriormente, estas expressões foram utilizadas em tempo de design, desde a fase de elicitação de requisitos até a definição da interface do usuário, com o objetivo de garantir o entendimento compartilhado do problema pela equipe de designers e evitar que os problemas de comunicação usuário-preposto representados pelas expressões ocorram.

Ao conjunto de expressões propostas originalmente para o sistema de ajuda, adicionamos as questões *O que isto afeta?* e *Do que isto depende?* contidas

apenas no conjunto de expressões de comunicabilidade apresentado no artigo de 2004. Estas questões foram adicionadas porque as consideramos de extrema importância ao projeto de interação de um sistema computacional, visto que elas permitem obter informações referentes ao fluxo de ações das tarefas realizadas pelos usuários. O conjunto de expressões de comunicabilidade resultante, apresentado na Tabela 3, foi o conjunto de expressões de comunicabilidade utilizado neste trabalho.

Tabela 3 – Expressões de comunicabilidade utilizadas no eXCeed.

<i>Expressões de comunicabilidade</i>	
A quem isto afeta?	O que aconteceu?
O que isto afeta?	O que é isto?
De quem isto depende?	Onde está?
Do que isto depende?	Onde eu estava?
Quem pode fazer isto?	Para que serve isto?
Como faço isto?	Por que devo fazer isto?
E agora?	Por que não funciona?
Epa!	Socorro!
Existe outra maneira de fazer isto?	

Diferente do que foi proposto nos trabalhos anteriores, ao invés de utilizarmos as expressões literalmente, derivamos questões que foram mapeadas às expressões inicialmente definidas para facilitar a elicitação de informações necessárias para o design. As expressões do sistema de ajuda serviram como base para a construção de novas questões que deverão guiar a atividade de reflexão envolvida durante o projeto da metacomunicação designer-usuário, bem como a discussão entre os designers e os usuários. Ao contrário das expressões do sistema de ajuda que representam uma fala do usuário ao designer, as questões derivadas representam falas do designer direcionadas aos usuários do sistema.

Durante a elaboração das descrições de metas, os designers estão preocupados com *quem vai fazer o quê e por quê*, enquanto que, na modelagem da interação e da interface, eles estão preocupados em elaborar *como* o que foi definido poderá ser feito pelo usuário. Assim, resolvemos utilizar em cada atividade apenas as expressões que refletem os interesses principais dos designers naquela atividade⁹. A seguir, na Tabela 4, endereçamos as expressões de ajuda utilizadas na atividade de elaboração das descrições de metas do usuário, as questões derivadas destas expressões e o tipo de informação que pode ser obtido a partir do conhecimento elicitado com cada questão.

⁹ Como se trata de um processo iterativo, os designers podem a qualquer momento fazer perguntas relacionadas a qualquer artefato ou atividade do eXCeed. No entanto, optamos por restringir as expressões utilizadas em cada atividade para auxiliar o designer em manter o escopo de cada artefato (Hoover et al., 1991).

Tabela 4 – Expressões, questões derivadas e tipo de informação que pode ser obtido a partir do conhecimento elicitado com a questão derivada.

Expressão	Questão derivada	Tipo de informação que pode ser obtido a partir do conhecimento elicitado com a questão derivada
Para que serve isto?	[sistema] O que você {precisa gostaria de} fazer com o sistema?	Metas do usuário
Para que serve isto?	[meta] Por que você {quer deve gostaria de} fazer isto?	utilidade do atingimento de uma meta
Por que devo fazer isto?	[meta] Para que serve isto?	Motivo pelo qual o usuário {quer solicitou} uma meta
	[informação, conceito, artefato] Para que serve isto? (Como você {pode deve} utilizá-lo?)	utilidade de informações, artefatos ou conceitos {criados manipulados utilizados destruídos} no atingimento de uma meta
O que é isto?	[informação, conceito, artefato] O que é isto?	significado de informações, artefatos ou conceitos que são {criados manipulados utilizados destruídos} durante o atingimento de uma meta
Quem pode fazer isto?	[meta] Quem pode fazer isto? (Além de você, mais alguém pode fazer isto? E se você não estiver disponível, quem poderá fazer?)	papéis de usuários envolvidos no atingimento de uma meta

Expressão	Questão derivada	Tipo de informação que pode ser obtido a partir do conhecimento elicitado com a questão derivada
Quem pode fazer isto?	[informação, conceito, artefato] Quem pode {criar manipular utilizar destruir} esta informação?	papéis de usuários envolvidos na {criação manipulação utilização destruição} de informações, conceitos ou artefatos no atingimento de uma meta
O que isto afeta?	[meta] Qual será o resultado disto?	possíveis modificações ou saídas resultantes do atingimento de uma meta
A quem isto afeta?	[meta] Que informações você {terá gostaria de ter} ao final disto? (Que outras informações você {terá gostaria de ter} ao final disto?)	Informações resultantes do atingimento de uma meta
	Quem poderá utilizar as informações resultantes disto?	papéis de usuários atingidos pelo resultado do atingimento de uma meta
	Quem {pode deve} ser avisado sobre a realização desta atividade?	papéis de usuários que devem ser comunicados sobre o atingimento de uma meta
Do que isto depende?	[meta] O que precisa ter sido realizado antes disto?	Metas que restringem o atingimento de outras metas
De quem isto depende?	[meta] De quem depende a realização disto?	papéis de usuário responsáveis por prover informações, artefatos ou conceitos

Expressão	Questão derivada	Tipo de informação que pode ser obtido a partir do conhecimento elicitado com a questão derivada
Do que isto depende?	[meta] Que informações são necessárias para a realização disto?	informações, artefatos ou conceitos que restringem o atingimento de uma meta
De quem isto depende?	[informação, conceito, artefato] De onde vem esta informação? (Quem as gera?)	papéis de usuários, informação ou fórmula dos quais depende a realização de uma meta

Além do texto que informa o significado da meta identificada, uma descrição de meta do usuário pode conter campos adicionais agregando informações apropriadas ao projeto em questão. Propõe-se que uma meta seja descrita com os seguintes campos:

- **identificador:** um número único representando uma chave para identificação da meta;
- **ordem:** um número indicando a ordem na qual a meta deverá ser projetada; e
- **meta:** breve texto que indica uma meta do usuário, que corresponderá a uma funcionalidade do sistema.

As descrições de metas do usuário apresentadas no exemplo mais adiante contêm os campos aqui sugeridos.

Como o intuito do eXCeeD é projetar interfaces de maneira ágil, devemos evitar a criação de documentação desnecessária durante a execução do processo. Por esta razão, quando uma determinada meta do usuário possuir características de pleno conhecimento da equipe de designers, não é necessário despender tempo elaborando um texto que descreva tal meta. Vamos supor, por exemplo, que a equipe de designers utilizando o processo proposto tenha bastante experiência no projeto da interação de sistemas do tipo CRUD (*Create, Retrieve, Update, Delete*) e o novo projeto com o qual eles estejam começando a trabalhar necessite de uma meta do usuário que seja *Cadastrar usuário*. Como os designers estão familiarizados com este tipo de meta, o seu detalhamento poderá conter apenas as informações necessárias para efetuar o cadastro do usuário, tais como *nome*, *telefone*, *e-mail*, *login* e *senha*, por exemplo.

No entanto, se durante esta etapa qualquer outra informação extra que o designer julgue importante para o projeto da interface do sistema tenha sido percebida, ela também deverá ser anotada na descrição de meta do usuário. Por exemplo, se um dos usuários para os quais um determinado sistema está sendo projetado possui alguma necessidade especial, como cegueira ou daltonismo, ou deseja que dado campo de entrada de texto tenha um formato específico, este tipo

de informação deverá ser anotada para ser levada em consideração durante o restante do projeto do sistema. Estas informações adicionais podem chegar até o ponto de se incorporar pequenos cenários de uso às descrições de metas. Por exemplo, considere a meta *Cadastrar livro* para o sistema de uma livraria qualquer. Um dos usuários que a solicitou pode querer cadastrar algumas vezes apenas um livro, em situações em que relatou “*comumente ocorre de um livro raro ser encomendado por um cliente e quando este livro chega é necessário que ele seja registrado no sistema, para que seja armazenado no histórico de vendas (...)*”, ou, outras vezes, com livros quando “*chega uma nova remessa de pedido de livros feito pela livraria (...)*”. Estes trechos do relato de um usuário podem ser adicionados à descrição de meta do usuário, pois cada uma destas situações pode requerer uma solução diferente de projeto. Por este motivo, o registro de informações deste gênero, onde o contexto de uso é relatado, é importante para que os designers possam incluir, na modelagem do sistema, meios que apóiem de maneira adequada as necessidades do usuário.

Como é proposto para projetos realizados em curto prazo por equipes pequenas de designers de IHC, o eXCeeD acredita que a pouca documentação escrita é compensada principalmente pelos valores de **comunicação** e **feedback**, aliados à boa memória das pessoas que estão envolvidas no projeto do sistema. No entanto, todo o processo de projeto do sistema deverá ser registrado em áudio como garantia de que, se alguma informação for perdida, ela possa ser recuperada. No entanto, os áudios gravados somente serão utilizados se necessário, em caso de dúvida, esquecimento ou discordância sobre o que foi discutido. Por esta razão, as descrições de metas do usuário e os artefatos da modelagem resultantes do projeto de interface de um sistema seguindo o eXCeeD deverão capturar apenas a essência do que foi discutido, necessária para prosseguir com o projeto.

A etapa de elaboração de descrições de metas do usuário é de fundamental importância para o eXCeeD, pois as descrições produzidas servirão de insumo para a atividade de modelagem da interação e da interface descrita mais adiante. Depois que tiverem redigido um determinado conjunto de descrições de metas do usuário para o sistema sendo projetado, os usuários e a equipe de designers devem concordar acerca do critério de priorização destas descrições, estabelecendo a ordem na qual elas deverão ser modeladas.

Para ilustrar a execução do eXCeeD proposto neste trabalho, vamos considerar o seguinte exemplo. Suponhamos que Gabriela seja uma professora de pós-graduação e, por isso, possui uma grande quantidade de livros em sua biblioteca particular. Estes livros são constantemente emprestados aos seus vários alunos ou colegas de trabalho e, por esta razão, ela precisa manter um controle de saída destes livros. Atualmente, ela faz este controle escrevendo os empréstimos em um cartão de papel, onde os dados são armazenados em três colunas:

- (i) o título do livro emprestado;
- (ii) o nome da pessoa a quem ela emprestou; e
- (iii) a data de empréstimo do livro.

Algumas vezes, Gabriela está em casa, precisa de um livro, mas não tem como saber se ele está emprestado ou se está em sua sala na universidade, pois o cartão com a lista dos empréstimos permanece na universidade. Um outro problema ocorre quando um livro emprestado é repassado a uma outra pessoa e Gabriela não é avisada. Desta maneira, Gabriela não está mais satisfeita com esta forma artesanal de gerenciar o empréstimo dos seus livros. Assim, a professora procurou uma empresa que adota o eXCeeD para o projeto de IHC para que fizessem o projeto de um sistema para o cadastro e o controle de empréstimo de seus livros particulares para solucionar os seus constantes contratempos.

Enquanto Gabriela relatava as suas necessidades, a equipe de designers anotava as informações importantes nas descrições de metas do usuário e utilizava as questões derivadas apresentadas anteriormente para conduzir a conversa com a futura usuária do sistema. Por exemplo, a questão *O que você precisa fazer com o sistema?* teve como resposta que a professora gostaria de poder ter uma lista de todos os livros que possui, de poder registrar os empréstimos que faz e de poder consultar os livros que estão na universidade ou em casa, entre outras coisas. Em seguida, a questão *Para que você quer fazer isto?* foi utilizada para saber que uma das metas já definidas, *Emprestar livro*, era necessária para que ela pudesse substituir os registros de empréstimos no cartão de papel pelo cadastro de

empréstimos no sistema. Uma outra questão utilizada foi a questão *Quem pode fazer isto?* se referindo a quem poderia emprestar um livro, cuja resposta foi que apenas Gabriela poderia realizar esta meta.

Depois de ouvir a solicitação de Gabriela e redigir as informações necessárias nas descrições de metas do usuário, a equipe de designers sugeriu um sistema contendo as seguintes metas: *Cadastrar livro*, *Emprestar livro*, *Visualizar empréstimos* e *Buscar livros*. Gabriela relatou que uma das dificuldades que ela tinha atualmente era que os seus livros emprestados eram repassados a uma outra pessoa e ela não era avisada. Assim, os designers disseram que seria interessante que estas pessoas, que constantemente pegavam os livros da professora emprestados, tivessem acesso apenas a uma determinada parte do sistema onde elas poderiam visualizar informações como com quem está determinado livro e há quanto tempo, e poderiam notificar Gabriela da alteração do responsável de um determinado livro. Assim, a equipe de designers sugeriu duas outras metas: *Cadastrar usuário* e *Alterar responsável por livro*. O conjunto de metas resultante é apresentado na Figura 6.¹⁰

¹⁰ Apenas as metas 1, 2 e 3 estão completas, por serem utilizadas no restante do exemplo.

Identificador: 1 Ordem: 1 Meta: Cadastrar livro Que informações são necessárias para a realização disto? Usuário deve informar os seguintes campos obrigatórios: ✓ título do livro ✓ nome(s) do(s) autor(es)	Identificador: 3 Ordem: 3 Meta: Cadastrar usuário Que informações são necessárias para a realização disto? ✓ nome ✓ telefone ✓ e-mail ✓ login ✓ senha Quem pode fazer isto? Apenas Gabriela.
Identificador: 2 Ordem: 2 Meta: Emprestar livro Que informações são necessárias para a realização disto? Usuário deverá informar o livro que foi emprestado e os dados da pessoa que o pegou: ✓ nome ✓ telefone ✓ e-mail Quem pode fazer isto? Apenas Gabriela. O que precisa ter sido realizado antes disto? Um livro precisa estar cadastrado no sistema.	Identificador: 4 Ordem: 4 Meta: Alterar responsável por livro ...
Identificador: 5 Ordem: 5 Meta: Visualizar empréstimos ...	
Identificador: 6 Ordem: 6 Meta: Buscar livros ...	

Figura 6 - Metas para o sistema de cadastro de livros solicitado por Gabriela.

Gabriela concordou que, inicialmente, este sistema resolveria os seus problemas. Como mostrado na Figura 6, a equipe de designers organizou todas as descrições de metas do usuário, agrupando-as em duas colunas: na da esquerda estão as metas relacionadas a livros, e na da direita as metas relacionadas a usuários. Como as partes mais importantes do sistema para a Gabriela eram as de cadastro e de empréstimo de livros, os designers, juntamente com a usuária, decidiram que o projeto do sistema seria realizado em duas etapas: na primeira, estariam as metas referentes a *Cadastrar livro* e *Emprestar livro*, e na segunda etapa, estariam o restante das metas: *Visualizar empréstimos*, *Buscar livro*, *Cadastrar usuário* e *Alterar responsável por livro*. Como não estavam dedicados apenas a este projeto, os designers estipularam, em comum acordo com a Gabriela, que entregariam a primeira parte do projeto do sistema em uma semana.

Além disso, eles também decidiram acerca do critério de priorização destas descrições, indicando a ordem na qual elas deveriam ser projetadas. Ao final, a decisão tomada foi que primeiro seria modelada a descrição *Cadastrar livro* seguida da descrição *Emprestar livro* nesta primeira etapa.

4.2.2. Modelagem da interação e da interface

Após a definição e elaboração das descrições iniciais das metas dos usuários, a modelagem da interação e da interface é realizada através da construção intercalada e incremental de diagramas de interação MoLIC e esboços de interfaces. Assim como proposto por Schön (1983; Schön e Bennett, 1996), espera-se que essa construção intercalada promova a *reflexão-em-ação*: ao elaborar os diagramas e esboços, os designers “conversam com os materiais”, que por sua vez podem disparar novas reflexões sobre a interação representada no diagrama. Além disso, a modelagem é incremental: a partir de cada descrição de meta do usuário ou conjunto de descrições de metas estreitamente relacionado é construído um pequeno conjunto de diagramas de interação MoLIC e de esboços de interfaces referentes àquela descrição. Esses artefatos serão posteriormente acrescentados ao modelo geral, que contém todo o sistema projetado até então.

Durante a atividade de modelagem, a equipe de designers será encorajada a elaborar diversas alternativas de interação e de interface para uma mesma descrição de meta do usuário, seguindo a prática de **soluções alternativas**. Esse tipo de incentivo é comprovadamente importante durante o projeto de sistemas interativos. Uma pesquisa apresentada por Tohidi e co-autores (2006) confirmou que o *feedback* obtido dos usuários, quando mais de uma alternativa é explorada por eles, oferece resultados superiores aos obtidos em relação à demonstração de apenas uma alternativa. No eXCeeD, uma alternativa poderá ser composta por um ou mais diagramas de interação MoLIC alternativos e os esboços de interfaces correspondentes. Ao final de cada iteração, as alternativas propostas deverão ser avaliadas junto aos usuários e, a partir do seu *feedback*, serão descartadas, revisadas, refinadas ou simplesmente aprovadas.

De maneira semelhante à atividade de elaboração das descrições de metas do usuário, questões derivadas das expressões do sistema de ajuda são utilizadas na atividade de modelagem. Caso os usuários não possam estar sempre presentes neste momento para responder as perguntas dos designers, estes últimos assumirão algumas respostas às perguntas, tomando as decisões em nome dos usuários com base no que já aprenderam sobre eles. Mais tarde, ainda durante a modelagem ou até mesmo no momento de avaliação, essas respostas deverão ser ratificadas ou retificadas.

Como dito anteriormente, as questões foram selecionadas de acordo com os interesses inerentes a cada uma das atividades, relacionados ao escopo e nível de detalhe das representações utilizadas (Hoover et al., 1991). No caso da modelagem da interação e da interface, os designers elaboram *como o usuário {pode | deve} fazer* o que foi definido nas descrições de metas do usuário¹¹. Assim, as questões utilizadas neste momento serão aquelas que estão relacionadas a este tipo de preocupação. Além disso, uma nova expressão *E se ...?* foi acrescentada ao conjunto, para que, neste momento, os designers pudessem refletir sobre as situações alternativas ou problemáticas. Apesar de as questões utilizadas para a modelagem da interação e para a modelagem da interface serem as mesmas, como os modelos são diferentes, o tipo de conhecimento adquirido irá diferir. A seguir apresentamos, na Tabela 5, as questões utilizadas durante a modelagem e o tipo de informação que pode ser obtido a partir do conhecimento elicitado com cada questão derivada.

¹¹ Assumimos aqui que “o que” o usuário quer fazer e “por quê” já foram definidos no momento de elaboração das descrições das metas. No momento de modelagem da interação e da interface, os designers consideram essas informações. Caso essas informações precisem ser revistas, assumimos que se trata de uma iteração, retornando à atividade anterior.

Tabela 5 – Expressões, questões derivadas e tipo de informação que pode ser obtido a partir do conhecimento elicitado com a questão derivada.

Expressão	Questão derivada	Tipo de informação que pode ser obtido a partir do conhecimento elicitado com a questão derivada	
Como faço isto?	[meta] Como você { faz isto atualmente gostaria de fazer isto }? (Quais os passos necessários para fazer isto atualmente?)	Im: ¹²	Seqüência de passos necessários (contendo cenas, diálogos, falas sobre signos, etc.) que determinam o valor de interação que o usuário { poderá deverá gostará de } percorrer para alcançar determinada meta
		le: ¹³	- seqüência de operações (falas ou diálogos) sobre os signos de interface que determinam quais signos e em que ordem eles { podem devem } estar dispostos na tela; - como determinado signo será representado no esboço de interface
Por que não funciona? E se ...?	[meta, passo de meta] Quais os problemas que podem surgir disto? / Quais as dificuldades que você possui ao realizar isto? (Como resolver este(s) problema(s)? / Como resolver esta(s) dificuldade(s)?)	Im:	caminho de interação para a recuperação de uma fala de <i>breakdown</i>
		le:	quais os erros e possíveis mensagens que o sistema poderá informar ao usuário, vinculados a cada signo ou composição de signos

¹² Im é a abreviação para “Informação para construção da MoLIC”.

¹³ le é a abreviação para “Informação para construção dos esboços de interface”.

Expressão	Questão derivada	Tipo de informação que pode ser obtido a partir do conhecimento elicitado com a questão derivada	
Do que isto depende? Onde eu estava?	[passo de meta] Por onde você começa a fazer isto?	Im:	ponto inicial (e.g. uma cena qualquer do modelo ou um acesso ubíquo) da execução da seqüência de passos de uma meta
		le:	signo de interface que vai alterar a aparência ou comportamento de outro signo de interface
	[passo de meta] O que é necessário para você fazer isto? (O que você precisa para fazer isto? O que acontece antes de você fazer isto?)	Im:	• pré-condições para a realização de determinado caminho de interação; • seqüência de passos que determinam o caminho de interação que o usuário {poderá deverá gostará de} percorrer para alcançar determinada meta; • caminhos de interação realizados antes do passo atual
		le:	signos de interface que precisam ter sido utilizados (e.g. campos que precisam ter sido preenchidos ou campo de seleção que precisa ter uma opção selecionada) antes de concluir a execução de determinado conjunto de operações

Expressão	Questão derivada	Tipo de informação que pode ser obtido a partir do conhecimento elicitado com a questão derivada	
O que isto afeta? O que aconteceu? E agora?	[meta, passo de meta] O que acontece depois que você faz isto? / Qual vai ser o resultado disto?	Im:	• pós-condições resultantes da realização de determinado passo no contexto atual; • caminhos de interação resultantes da realização de determinado passo no contexto atual
		le:	resultado da utilização (leitura, manipulação, etc.) de determinado signo de interface
	[meta, passo de meta] O que {pode ser feito deve ser feito gostaria de fazer} a partir daqui? / (O que {pode deve} ser realizado depois disto? / Existe mais alguma coisa que possa ser realizada depois disto?)	Im:	• possíveis caminhos de interação realizados depois do passo atual de interação; • caminhos de interação para uma meta relacionada (podem ser preferenciais ou não) (e.g., cena que leva a uma outra meta) • indicar possíveis relações entre metas (e.g., duas metas finais que agora percebeu-se que uma delas pode ser instrumental para outra)
		le:	signos de interface que precisam ter sido utilizados depois de se realizar determinada operação sobre um ou mais signos de interface

Expressão	Questão derivada	Tipo de informação que pode ser obtido a partir do conhecimento elicitado com a questão derivada	
Epa!	[meta, passo de meta] Como você poderá desfazer isto?	Im:	• caminho de interação necessário para cancelar um passo que está sendo realizado; • caminho de interação necessário para desfazer um passo que foi realizado
		le:	signos de interface necessários para cancelar uma operação; signos de interface que permitirão desfazer uma ou mais ações do usuário
Existe uma outra maneira de fazer isto? E se ...?	[meta, passo de meta] Você {faz gostaria de fazer} isto de outra maneira? (E se você tentasse fazer isto de outra maneira, como seria?)	Im:	• caminho de interação alternativo para a realização de uma seqüência de passos para alcance de determinada meta; • caminho de interação adicional para possíveis relações entre metas que não foram percebidas antes (e.g., duas metas finais que agora percebeu-se que uma delas pode ser instrumental para outra)
		le:	• signos de interface alternativos para a realização de uma seqüência de operações; • elementos de interface para possíveis relações entre metas que não foram identificadas (e.g., duas metas finais que agora percebeu-se que uma delas pode ser instrumental para outra)

Expressão	Questão derivada	Tipo de informação que pode ser obtido a partir do conhecimento elicitado com a questão derivada	
Onde está?	[informação, artefato, conceito] Onde é que esta informação será {criada utilizada manipulada destruída}?	Im:	em que ponto do caminho de interação determinado artefato ou conceito será {criado utilizado manipulado destruído}
		le:	em que lugar dos esboços de interfaces o signo relacionado a determinado artefato ou conceito será {criado utilizado manipulado destruído} e como ele será representado

Como o eXCeeD é fundamentado na EngSem, a maior preocupação que temos ao propô-lo como processo reside na construção de um discurso consistente e coeso do preposto do designer a partir de diferentes descrições de metas do usuário. Ao descrever brevemente a MoLIC na Seção 2.2.2, dissemos que o projeto de IHC proposto em sua segunda edição possui duas etapas: a da estruturação da conversa e a de detalhamento da conversa (Silva, 2005). A modelagem da interação no eXCeeD consistirá apenas na etapa de estruturação da conversa, suficiente para fornecer a visão geral da interação do sistema necessária à construção do discurso do preposto a partir de diferentes descrições de metas do usuário. As conversas serão detalhadas já nos esboços de interface, visando eliminar redundâncias desnecessárias que limitariam a agilidade do processo.

Os esboços de interfaces são uma representação amplamente utilizada para expressar a interface de software de maneira informal. Através do uso de esboços de interface, os designers podem delinear as diferentes idéias para as futuras interfaces de um software de forma rápida (Rettig, 1994; Buxton, 2007). Além disso, é possível testar estas diferentes idéias com os usuários ainda nas fases iniciais do desenvolvimento do sistema, a um custo muito baixo. Os esboços de interface também permitem que os designers mantenham-se focados apenas nas estruturas básicas da interface, em quais estruturas os elementos de interface serão utilizados ou qual o seu posicionamento na tela proposta, sem se preocupar com detalhes irrelevantes para aquele momento, como cores ou tamanho da fonte utilizada, por exemplo (Landay e Myers, 2001). Assim, com os esboços de interface é possível expressar as conversas usuário-preposto modeladas com os diagramas de interação, considerando algumas das questões envolvidas na etapa de detalhamento da conversa proposta no projeto da interação humano-computador da segunda edição da MoLIC. Os esboços de interface permitem definir e estruturar os signos de interface, os diálogos que farão parte da conversa usuário-preposto modelados na MoLIC, bem como a seqüência em que estes diálogos vão surgir durante a comunicação usuário-preposto. Desta maneira, podemos dizer que os diagramas de interação apresentam o fluxo de interação para o sistema sendo projetado, enquanto que os esboços de interface dizem como esta interação está expressa em termos dos signos estáticos da interface.

Temos que cada descrição de meta do usuário vai servir de insumo para um conjunto de trechos de diagramas de interação MoLIC e esboços de interface referentes àquela descrição. Sabemos que o modelo de interação MoLIC construído seguindo-se a etapa de estruturação da conversa nos fornece a visão geral da interação. Sabemos também que os esboços de interface fornecem uma expressão para a interação descrita nos diagramas em termos dos signos de interface. Agora, só nos resta informar como é que cada conjunto de trechos de diagramas de interação MoLIC e esboços de interface para determinada descrição estarão relacionados uns aos outros para que o discurso do preposto seja coeso e consistente. Esta informação será dada a seguir com a explicação da construção incremental proposta para atividade da modelagem da interação e da interface. Mas como se dará a construção incremental de tais artefatos?

A construção incremental dos modelos deverá ocorrer da seguinte maneira: durante todo o projeto do sistema, serão mantidos: (i) um diagrama de interação MoLIC integrado, fornecendo a visão geral da interface; e (ii) os respectivos esboços projetados até o presente momento. Para cada descrição de meta, constroem-se trechos alternativos de diagrama de interação MoLIC e os respectivos esboços de interface. Esses trechos de interação devem ser encaixados no diagrama de interação MoLIC que fornece a visão geral do sistema. Após ser avaliada a proposta para essa meta, prossegue-se com as próximas da mesma maneira, conforme as prioridades de projeto acordadas durante o planejamento.

O fluxo de atividades oferecido pelo eXCeeD permite que o aprendizado da equipe de designers projetando um sistema seja contínuo, ou seja, à medida que cada uma das atividades é realizada, a equipe de designers aprende um pouco mais sobre a aplicação que está sendo construída e a prática da **reestruturação** permite que alterações decorrentes deste aprendizado constante sejam facilmente incorporadas. Durante a modelagem, o aprendizado é mais intenso, já que a criação de modelos promove a “conversa com materiais” (Schön, 1983; Schön e Bennett, 1996).

Além de a construção dos modelos ser realizada de maneira incremental, as práticas do eXCeeD como **integração contínua**, **design coletivo** e **reestruturação** contribuem para que a consistência da interface seja mantida. Com a prática da

integração contínua garantimos que, se um novo conjunto de trechos de diagramas de interação MoLIC e esboços de interfaces for construído, ele será prontamente integrado ao modelo geral da interface. A prática do **design coletivo** permite que todos os designers envolvidos tenham a visão geral do projeto e, conseqüentemente, levem este conhecimento em consideração ao fazer a modelagem para uma próxima descrição de meta. Por exemplo, ao modelar a descrição de meta do usuário *Emprestar livro*, os designers devem levar em consideração o conhecimento adquirido com as modelagens anteriores, no caso com a modelagem da descrição de meta do usuário *Cadastrar livro*. Percebendo que estas metas são semelhantes, eles podem propor soluções semelhantes que vão contribuir para a consistência do discurso interativo. Com a prática da **reestruturação** é possível garantir que todos os designers têm autonomia para modificar o projeto da interface conforme seja necessário. Algumas modificações tanto nos diagramas de interação MoLIC quanto nos esboços de interfaces serão inevitáveis durante todo o projeto de interface.

Além destas, outras práticas são evidenciadas durante a atividade de modelagem da interação e da interface. São elas: **metáforas** e **design de interface simples**. É neste momento que o conhecimento adquirido durante a elaboração das descrições de metas do usuário, ou seja, as informações obtidas dos usuários podem ser aplicadas para a criação das **metáforas** de interface no projeto do sistema. É durante esta atividade também que a equipe de designers pode aplicar a prática do **design de interface simples**, pois está procurando propor as soluções simples para os problemas encontrados no momento.

Voltando ao exemplo do sistema solicitado por Gabriela, temos que a primeira descrição a ser modelada será a descrição *Cadastrar livro*, representando o primeiro passo nessa atividade. Utilizando as questões derivadas propostas para a modelagem da interação e da interface, Gabriela começa respondendo a pergunta *Como você gostaria de fazer isto?* e os designers iniciam a modelagem em função da resposta fornecida por ela. Assim, a construção dos diagramas de interação e dos esboços de interfaces propostos informa que, para cadastrar um

livro, será necessário informar os dados deste livro listados na descrição de meta reproduzida na Figura 7.

Identificador: 1
 Ordem: 1
 Meta: Cadastrar livro

Que informações são necessárias para a realização disto?

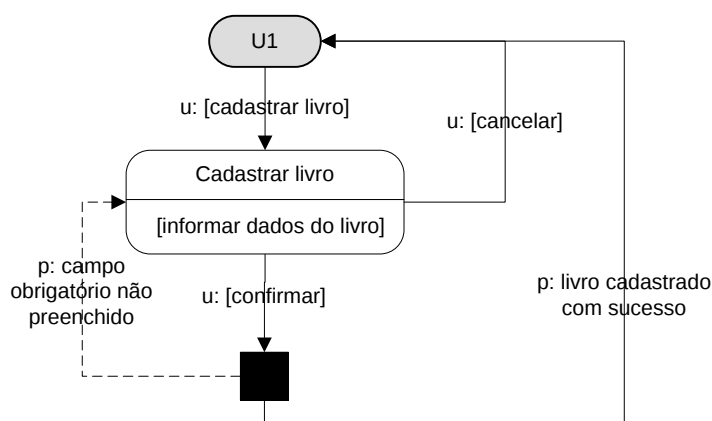
Usuário deve informar os seguintes campos obrigatórios:

- ✓ título do livro
- ✓ nome(s) do(s) autor(es)

Figura 7 – Descrição de meta do usuário para a meta *Cadastrar livro*.

Em seguida, a equipe de designers pergunta *O que pode ser feito daqui?* e Gabriela responde que deveria poder concluir o cadastro do livro. Com base no conhecimento que o designer tem a respeito das necessidades de Gabriela, ele responde em seu nome as seguintes questões: *Quais os problemas que podem surgir disto?* informando que se os campos obrigatórios não forem preenchidos, o sistema (preposto) do designer deverá informar o ocorrido e oferecer uma nova chance ao usuário para o cadastro do livro e *Como você poderá desfazer isto?* informando que o sistema permitirá o cancelamento do cadastro do livro. O resultado para a modelagem da interação e da interface desta descrição é apresentado no diagrama de interação MoLIC e esboços de interfaces da Figura 8 (a) e Figura 8 (b), respectivamente. Inicialmente, estes modelos fornecem a visão geral do projeto da interface do sistema até o momento. Note que para evitar redundância e agilizar ainda mais o processo, os campos obrigatórios listados na meta não foram definidos detalhadamente na MoLIC. Desta maneira, caso ocorram modificações nesta lista de campos na descrição de meta, somente esta descrição de meta e o esboço de interface serão modificados.

(a)



(b)

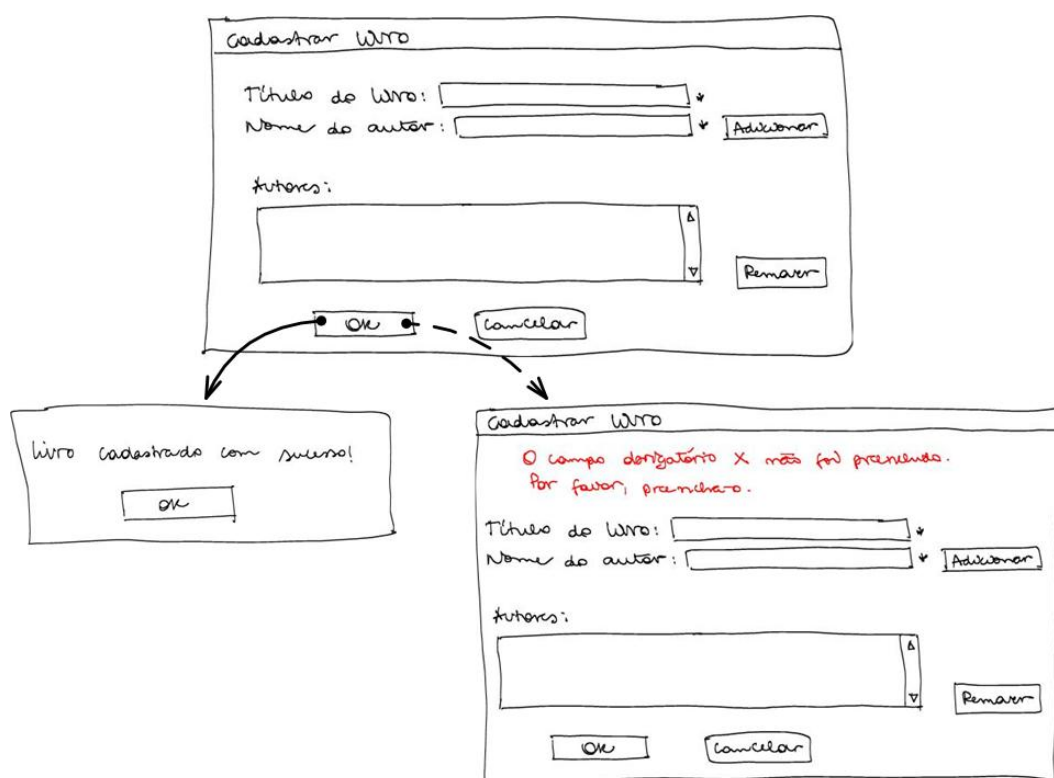


Figura 8 – (a) Diagrama de interação e (b) esboço de interface para a meta *Cadastrar livro*.

Depois de terminar a modelagem para a descrição de meta do usuário *Cadastrar livro*, os designers prosseguem com a modelagem da descrição de meta do usuário *Emprestar livro*. No entanto, ao começar esta modelagem, eles percebem que poderia acontecer de muitos livros possuírem o mesmo autor. Os designers resolvem, então, conversar com Gabriela sobre esta dúvida e acabam percebendo que esta era de fato uma necessidade real de Gabriela e que, por isso, deveria ser incorporada ao conjunto de metas. Assim, a descrição de meta do

usuário *Cadastrar autor* apresentada na Figura 9 é acrescentada ao conjunto de descrições de metas do usuário.

Identificador: 7
 Ordem: 1a
 Meta: Cadastrar autor

Que informações são necessárias para a realização disto?

Usuário deve informar os seguintes campos obrigatórios:

- ✓ nome do autor

Figura 9 – Descrição de meta do usuário *Cadastrar autor*.

Além disso, a equipe de designers acrescentou que seria interessante que essa meta do usuário fosse modelada antes da meta *Emprestar livro*, uma vez que o cadastro de autores seria utilizado ao cadastrar um livro. A professora também concordou com esta modificação. Assim, os designers incorporaram a modificação às descrições de meta, de maneira que à descrição de meta do usuário *Cadastrar autor* atribuiu-se **Ordem: 1a**, indicando que ela deveria ser modelada antes da descrição *Emprestar livro* que possui **Ordem: 2**, conforme apresentado anteriormente (Figura 9). Assim sendo, a ordem de projeto agora será *Cadastrar autor* seguida por *Emprestar livro*, uma vez que a descrição *Cadastrar livro* já foi modelada. Assim, o segundo passo nesta atividade será a modelagem da descrição de meta *Cadastrar livro*.

Como estamos propondo a construção incremental dos modelos, para a modelagem da interação e da interface da descrição de meta *Cadastrar autor*, vamos tomar como base os diagramas de interação e os esboços já construídos para a descrição *Cadastrar livro*. Sendo assim, a modelagem para a nova descrição de meta será acrescentada às representações feitas anteriormente. Desta maneira, antes de adicionar a modelagem para uma nova descrição, a equipe de designers pode verificar como estão os modelos criados anteriormente e adotar soluções semelhantes para problemas semelhantes. Como o atingimento das metas *Cadastrar livro* e *Cadastrar autor* é similar, as questões utilizadas durante a modelagem da descrição de meta foram as mesmas, apresentando respostas semelhantes, com exceção dos dados. Ao incorporar a nova modelagem para a descrição de meta do usuário *Cadastrar autor* ao diagrama de interação existente,

temos a solução apresentada na Figura 10 (a). Logo após, na Figura 10 (b) são apresentados os esboços de interface correspondentes à descrição de meta do usuário *Cadastrar autor*.

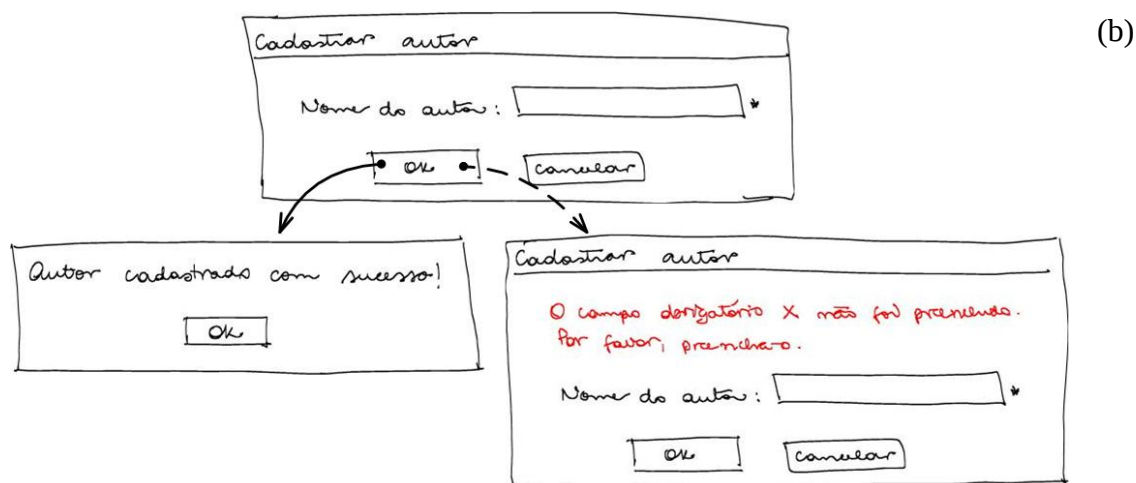
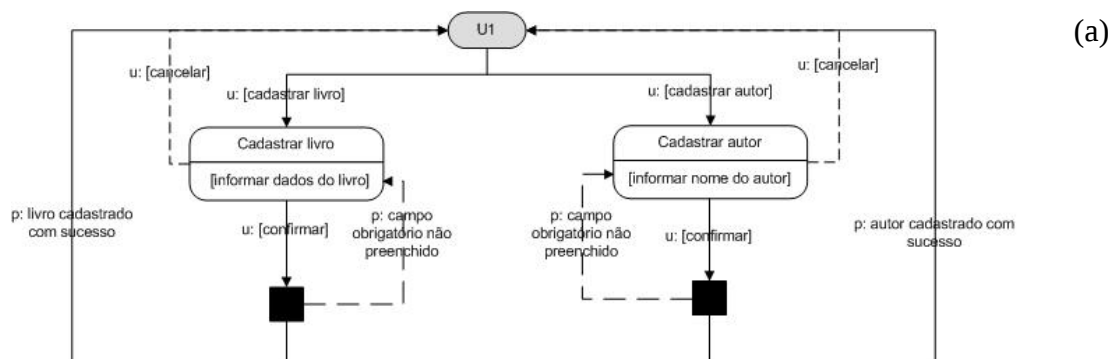


Figura 10 – (a) Diagrama de interação MoLIC e (b) esboços de interface da meta *Cadastrar autor*.

Como o sistema possuirá um cadastro de autores, então os esboços de interfaces para a descrição de meta do usuário *Cadastrar livro* foram alterados pelos designers da interação. Conforme apresentado na Figura 11, o campo de texto para o **Nome do autor** foi substituído por um elemento de seleção. Vale lembrar que o diagrama de interação para a mesma descrição não sofreu nenhuma alteração.

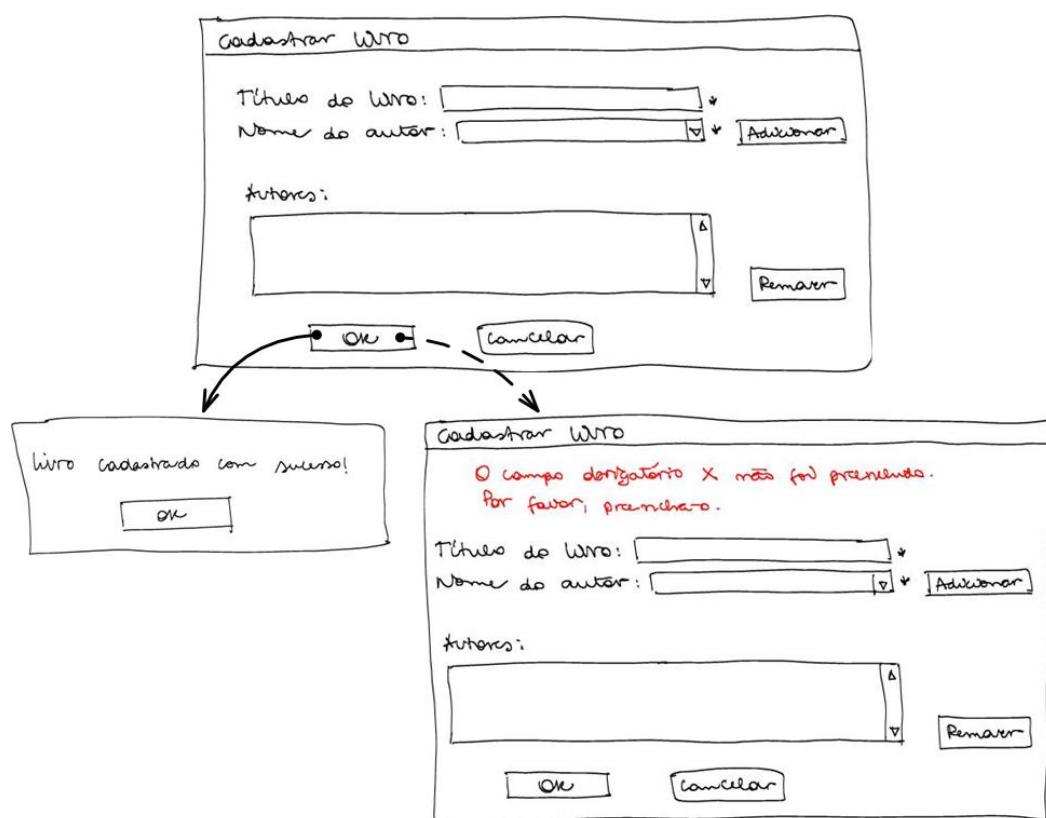


Figura 11 – Esboços de interfaces para a descrição de meta *Cadastrar livro* com a possibilidade de selecionar um autor do cadastro de autores.

Depois de elaborar a descrição de meta do usuário *Cadastrar autor*, Gabriela já não se encontrava mais presente nas discussões com os designers do seu sistema. No entanto, eles continuavam utilizando as questões para a modelagem e, neste momento, surgiu a seguinte questão *Você gostaria de fazer isto (a meta Cadastrar autor) de outra maneira?*. Com o conhecimento que tinham sobre a usuária principal do sistema, os designers consideraram que o cadastro de autor poderia ser realizado de duas maneiras: durante o momento em que o usuário está cadastrando o livro ou em um outro momento independente, quando o usuário somente deseja cadastrar um novo autor na base de dados dos

autores. Assim, de acordo com a definição para metas dada anteriormente, o usuário poderá utilizar o sistema para atingir a meta *Cadastrar autor* como meta instrumental ou final, respectivamente.

Levando-se em consideração as duas maneiras pelas quais pode acontecer um cadastro de autor, foram necessárias as seguintes alterações adicionais:

- (i) acrescentar a fala de transição **u:[cadastrar autor]** saindo da cena **Cadastrar livro** e chegando na cena **Cadastrar autor**, para disparar o cadastro de autor enquanto se está cadastrando um livro;
- (ii) para o atingimento da meta *Cadastrar autor* como meta final, acrescentamos a pré-condição **pré: cena anterior ≠ cadastrar livro** à fala de transição que liga o processo de sistema após a confirmação do cadastro de autor ao acesso ubíquo U1, para confirmar o cadastro de autor feito independentemente, ou seja, como meta final; e
- (iii) para o atingimento da meta *Cadastrar autor* como meta instrumental, acrescentamos a fala de transição **p: autor cadastrado com sucesso** com a pré-condição **pré: cena anterior = cadastrar livro**, para indicar que o usuário está tentando atingir a meta *Cadastrar autor* durante a execução da meta *Cadastrar livro*.

Esta modificação foi, então, refletida nos modelos de interação e interface propostos conforme apresentado na Figura 12.

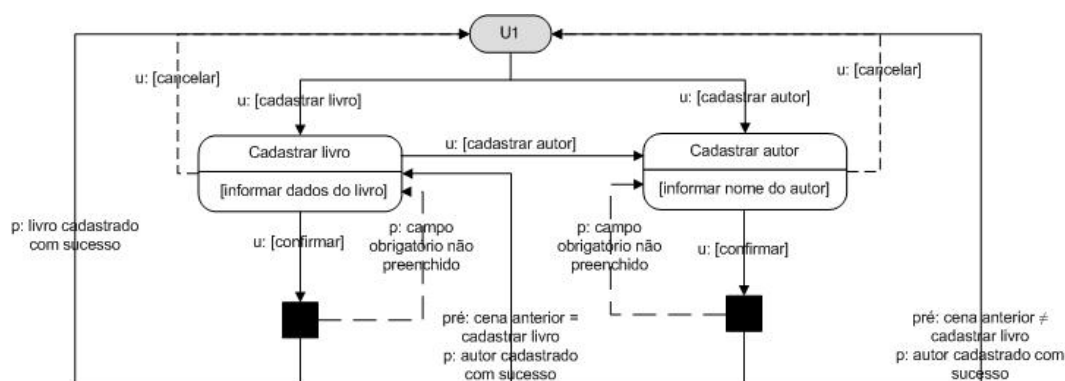


Figura 12 – Diagrama de interação geral para o sistema de cadastro de livros.

Para se adequar às duas possibilidades para o cadastro de autor, foi necessário adicionar, aos esboços de interface, um botão **Cadastrar novo autor...** que possibilitará a chamada ao cadastro de autores. Essa modificação é apresentada na Figura 13.

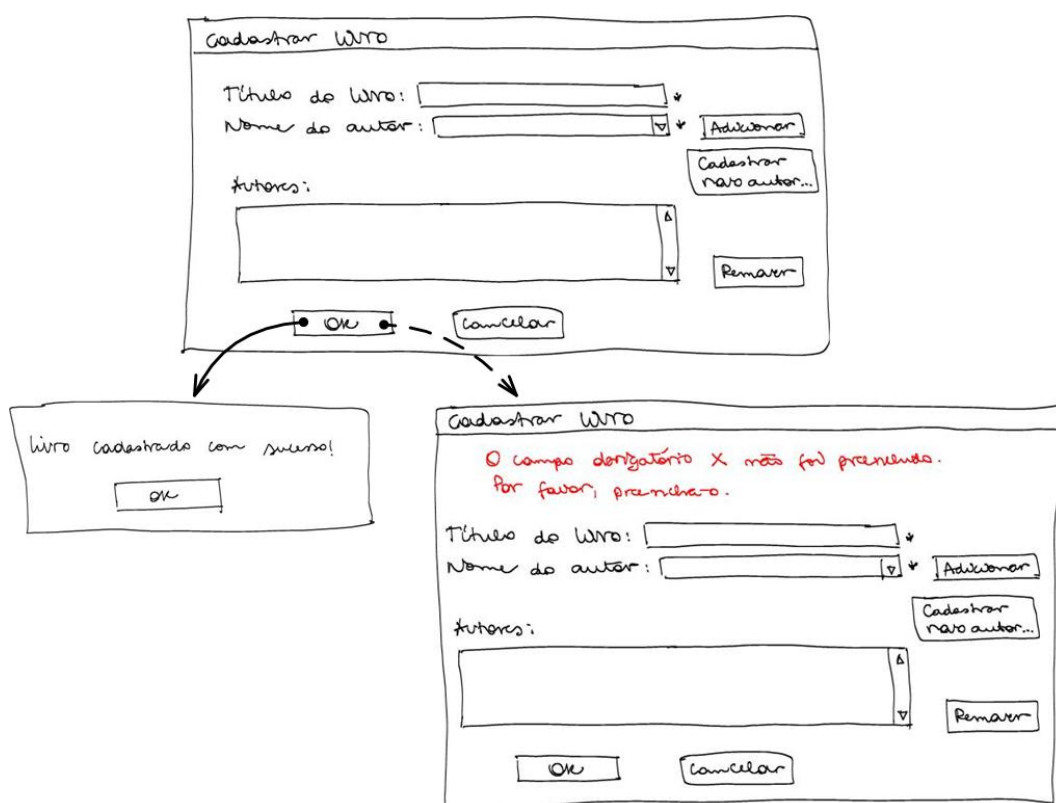


Figura 13 – Cadastro de livro com a possibilidade de cadastrar um novo autor durante o cadastro do livro.

Ao final da criação destes esboços, os designers perguntaram *E se você tentasse fazer isto (o passo de meta Adicionar autor) de outra maneira, como seria?*, obtendo como resposta a nova alternativa de esboço de interface apresentada na Figura 14. A diferença desta nova alternativa é que mais de um autor pode ser selecionado por vez, através dos dois campos de seleção disponíveis, onde estão listados separadamente os autores cadastrados no sistema e os autores do livro que já tiverem sido selecionados pelo usuário para aquele livro.

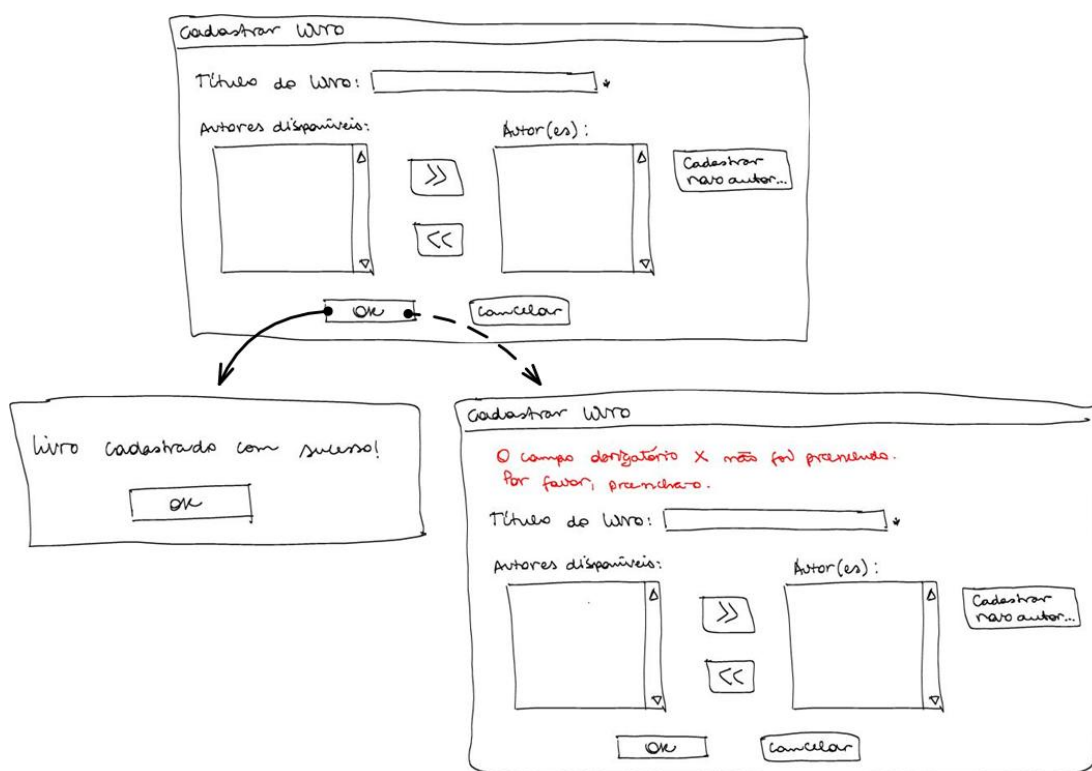


Figura 14 - Outra alternativa de interface para a meta *Cadastrar autor*.

Depois de terminada a construção de uma nova alternativa de interface para a descrição de meta *Cadastrar autor*, os designers prosseguem para o próximo conjunto de metas, incrementando o diagrama de interação MoLIC e elaborando os esboços de interface correspondentes. Mantendo a metacomunicação definida até o momento representada num único diagrama de interação, é possível avaliar, a cada ciclo de design, a consistência da metacomunicação representada no diagrama e entre as diferentes representações (metas, diagramas e esboços de interface).

4.2.3. Avaliação junto aos usuários

Depois da modelagem da interação e da interface, os designers deverão avaliar os esboços de interfaces produzidos junto aos usuários, evidenciando desta maneira a prática da **simulação dos esboços**. Para tanto, nos inspiramos na técnica da prototipação em papel proposta por Snyder (2003). Esta atividade é realizada com o intuito de avaliar o fluxo de interação e esboços de interfaces propostos para o projeto, bem como obter *feedback* dos usuários a respeito da metacomunicação proposta.

Como dito anteriormente, para realizar uma avaliação utilizando a prototipação em papel é desejável a participação de quatro personagens: **usuário**, pessoa para simular as ações do **computador**, **facilitador** para conduzir a avaliação e **observador** fazendo anotações a respeito do teste. Para facilitar a simulação e avaliação dos esboços proposta no eXCeeD, deveremos utilizar apenas três dos quatro personagens propostos, reunindo na mesma pessoa que desempenhar o papel de **computador**, as funções do **facilitador** da avaliação. Assim, como o eXCeeD é proposto para uma equipe de designers de duas a três pessoas, além do **computador-facilitador** teremos até dois **observadores** registrando os problemas e observações dos usuários durante as avaliações.

Uma vez que os usuários representativos do sistema que está sendo projetado estão participando do processo desde o seu início, para planejar a avaliação junto aos usuários através da simulação dos esboços no eXCeeD, a equipe de designers deverá se preocupar em:

- (i) definir qual o papel que será desempenhado por cada um dos membros da equipe de designers;
- (ii) definir quais as tarefas que serão realizadas pelos usuários durante a simulação; e
- (iii) verificar se todo o material necessário para a simulação foi construído (além dos esboços de interface, outros elementos,

tais como menus ou algumas opções de um campo de seleção, por exemplo, podem ser necessários).

Ao final da modelagem da interação e da interface, os designers têm um diagrama de interação único representando toda a semântica da interação do sistema projetado. Todos os caminhos de interação para o sistema projetado estão definidos neste diagrama. Assim, enquanto na prototipação em papel original temos um conjunto de esboços de interface, para os quais não há documentação do fluxo de interação, na avaliação aqui proposta os caminhos de interação estarão registrados por escrito ao modelar esses caminhos em diagramas de interação MoLIC. Conseqüentemente, teremos que o personagem **computador-facilitador** será guiado pela interação já representada, consultando o diagrama sempre que um novo passo de interação vier a ser executado. Esta idéia ficará mais clara com o exemplo mais adiante.

O designer que desempenha o papel de **computador-facilitador** é o responsável por informar ao usuário qual a tarefa que ele tem que fazer, mostrando também o esboço de interface através do qual ele deverá realizar a tarefa proposta. Durante a simulação, o usuário e o **computador-facilitador** podem conversar e discutir sobre os esboços construídos, de maneira que esta conversa forneça indicativos dos problemas e dúvidas do usuário que devem ser capturados pela simulação. Ademais, esta simulação é uma oportunidade que os designers têm de estar presente fisicamente no momento de interação do usuário com os protótipos do sistema, explicitando as decisões que foram tomadas por eles durante a construção da metacomunicação.

Voltando ao exemplo do sistema de gerenciamento de empréstimo de livros solicitado pela professora universitária Gabriela, podemos realizar a avaliação dos esboços apresentada a seguir. Para ilustração do exemplo, a meta escolhida para ser simulada será a meta *Cadastrar livro*. Com o intuito de verificar o fluxo de interação proposto, a equipe de designers irá simular para Gabriela os esboços de interface propostos para a referida descrição de meta. Durante a simulação, a equipe de designers solicita que Gabriela cadastre um livro e apresenta o esboço

da tela *Cadastrar livro* (Figura 10 (b)). Depois, Gabriela confirma o entendimento da solicitação da equipe de designers, interagindo com o esboço de interface: primeiro, preenche o campo de **Título do livro**, depois seleciona o **Nome do autor** do livro e clica no botão **Adicionar** para que ele seja acrescentado à lista de autores do livro e, por último, clica no botão **Confirmar**. Esta seqüência de ações é ilustrada na Figura 15. Quando a simulação é feita com esboços de interface produzidos em papel, o preenchimento de campos de entrada do usuário pode ser realizado com a colagem de notas Post-it® contendo os dados e um clique do mouse corresponde a pressionar com o dedo a parte do esboço referente ao botão, por exemplo.

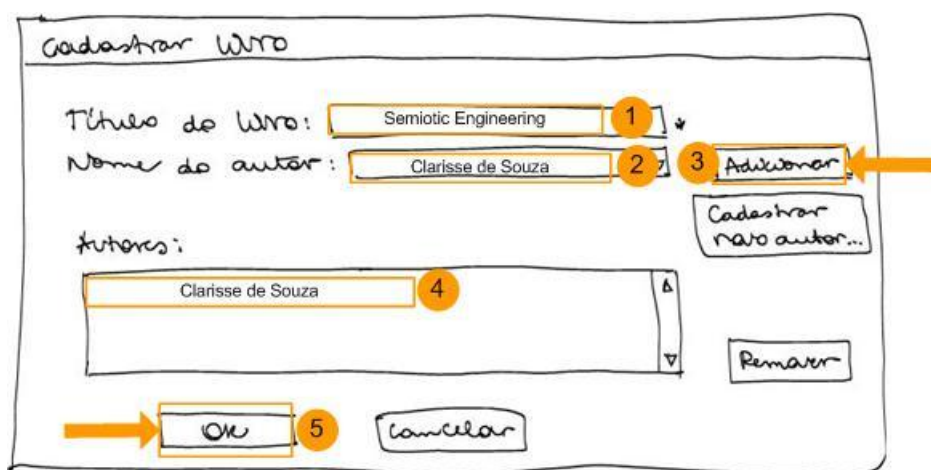


Figura 15 – Ilustração do preenchimento do esboço de interface *Cadastrar livro*.

Uma vez que os dados foram todos preenchidos e o usuário clicou no botão **Confirmar**, um membro da equipe (responsável pela simulação dos esboços) deverá ler o diagrama de interação MoLIC para que possa determinar a próxima tela a ser exibida. A leitura desse exemplo é ilustrada com a setinha na fala de transição destacada no diagrama de interação MoLIC, apresentado na Figura 16.

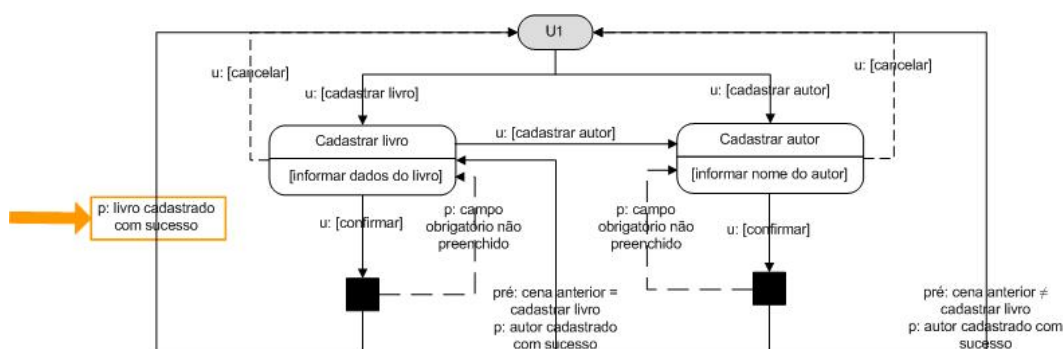


Figura 16 – Simulação da observação do diagrama MoLIC pelo designer responsável pela simulação.

Como, nesta simulação, todos os campos obrigatórios foram preenchidos corretamente, o cadastro do livro será realizado com sucesso. Depois de conferir o fluxo da interação, o designer mostra então a tela de sucesso, conforme ilustrado na Figura 17.

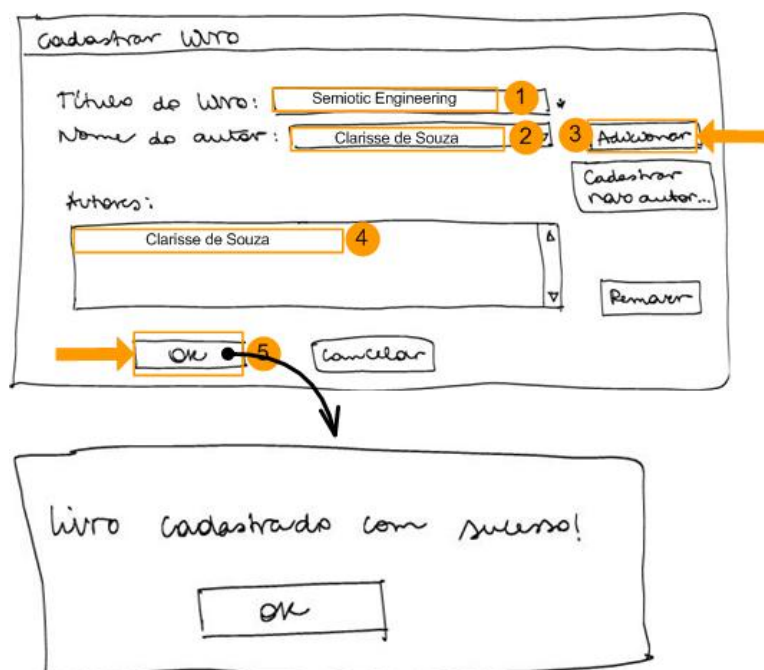


Figura 17 – Simulação do designer mostrando a tela de sucesso.

Se novas solicitações do usuário surgirem com a simulação dos esboços, elas deverão ser analisadas e incorporadas ao conjunto de descrições de metas do usuário ou à modelagem da interação e da interface. Um mesmo diagrama MoLIC deverá ser elaborado, estendido ou adaptado ao longo de todo o projeto da

interação. Desta maneira, o modelo de interação será construído incrementalmente à medida que novas descrições de metas forem sendo consideradas. Como os designers são os responsáveis pela construção da metacomunicação, é importante salientar que todas as modificações no projeto serão realizadas somente caso eles as julguem necessárias. Neste caso, as práticas como **reestruturação** e **design coletivo** contribuirão para permitir que, mesmo com as modificações introduzidas, os designers mantenham o projeto feito para todo o sistema integrado. Neste caso, a utilização da MoLIC apoiará o trabalho dos designers na direção de um projeto consistente e coerente.