

# Integrando Padrões de Usabilidade na Especificação de Requisitos para Apoiar o Re-Projeto Participativo de Interfaces

Felipe Távora Pereira<sup>1</sup>, Elizabeth Furtado<sup>2</sup>

<sup>1</sup>Universidade de Fortaleza, Rua Isac Amaral, 112/501, CEP 60130120,  
Fortaleza, Ceará, Brasil  
felipe@sefaz.ce.gov.br

<sup>2</sup>Universidade de Fortaleza, Mestrado em Informática Aplicada, Av. Washington Soares, 1321,  
Fortaleza, Ceará, Brasil  
elizabet@unifor.br

## Abstract

*Este artigo propõe uma metodologia para dar maior suporte ao re-projeto de interfaces dos sistemas de software, contendo atividades integradas à disciplina de Requisitos do Processo Unificado (RUP). Estas atividades são baseadas nos princípios de projeto participativo, através dos quais os usuários podem participar ativamente do processo de tomada de decisão sobre alterações de interface, tendo maior consciência dos impactos que essas alterações irão causar no sistema a partir da relação entre os requisitos especificados. A utilização de padrões de usabilidade facilitará a comunicação entre usuários e projetistas, por apresentar exemplos ilustrativos referentes a soluções de usabilidade para certos problemas, ainda, uma maior segurança ao usuário sobre o que ele espera das modificações.*

## 1. Introdução

A busca pela satisfação dos usuários de um sistema, além da busca pela melhoria de sua manutenção e reutilização (pelos engenheiros de software) são critérios que motivam fortemente a mudança de um sistema [5]. No entanto, não são somente estes os critérios a serem considerados, sendo, também, importante verificar a relação custo e benefício das mudanças (impacto causado por essas mudanças), para se ter certeza da decisão de se evoluir um sistema.

As empresas de software, principalmente as pouco experientes, enfrentam dificuldades de delimitar com seu cliente quando uma solicitação de mudança faz parte ainda do contrato de desenvolvimento daquele software, ou se já está se caracterizando como um contrato de manutenção. Isso leva a uma reflexão sobre o conceito de manutenção.

Uma modificação se torna necessária quando um produto existente (mesmo que uma versão deste), de alguma forma, satisfaz de forma deficiente uma necessidade do cliente (usuário). Isso pode acontecer

mesmo quando as funcionalidades de um sistema estão perfeitas. Pode ser o caso em que o usuário não compreenda o sistema, em função de este sistema apresentar problemas em sua utilização. Isso quer dizer que uma necessidade leva ao projeto de um produto, enquanto um produto insatisfatório conduz ao seu re-projeto.

Dois dos motivos mais comuns para se realizar uma modificação em um software são: a solicitação de uma nova funcionalidade requerida pelo usuário e o informe de um defeito no funcionamento esperado do software [9]. Nestes dois casos, pode ser verificada a existência de solicitação da melhoria da usabilidade (facilidade de uso) de um software. No primeiro caso, essa solicitação pode acontecer quando a nova funcionalidade requerida é, por exemplo, a de visualização de todos os componentes de um elemento, antes de excluir um componente. No segundo caso, o defeito do funcionamento do software pode ser decorrente de uma metáfora mal projetada para representar uma função, impedindo que o usuário execute a opção correta.

A interface de um sistema é parte integrante deste e viabiliza sua interação com o usuário. Ela é, atualmente, uma das principais causas da necessidade de manutenção de sistemas de software, pois não adianta o sistema ser perfeitamente desenvolvido em suas funcionalidades (qualidade em Engenharia de Software), se não apresenta uma interface que seja de fácil entendimento para o usuário, de modo que a qualidade de suas funcionalidades possa ser, de fato, aproveitada.

Isso acontece, geralmente, devido a não serem levadas em consideração necessidades e experiências dos usuários finais desses sistemas, padrões universais de acessibilidade, etc. Outra possível causa é o pouco envolvimento dos usuários, às vezes nenhum, durante o desenvolvimento/ manutenção de software, como forma de auxiliar em sua validação. A prova disso está nos resultados obtidos por uma pesquisa feita em [11], na qual foram questionados quais seriam os fatores-chave de sucesso em um projeto de software, sendo o envolvimento do usuário declarado como o principal

deles (com 16%), seguido pelo suporte gerencial e a definição clara dos requisitos (aproximadamente 14%).

A qualidade no uso do sistema por parte de seus usuários é comprometida quando problemas de usabilidade (problemas no uso do sistema) são detectados em sua utilização. Alguns desses problemas detectados podem ser tratados pela mudança diretamente no sistema, sem que para isto seja realizada alteração nos requisitos do sistema. Já outros problemas de usabilidade levam a alterações nos requisitos (não-funcionais) do sistema, e nesse tipo de problema é que este trabalho pretende focar, ou seja, em uma metodologia que gereencie o re-projeto da interface de um sistema que apresentou problemas em seu uso, e que acarrete em mudanças em seus requisitos relacionados à interface.

Na próxima seção, será descrito o que motivou este trabalho para tratar problemas de usabilidade identificados durante o uso de um sistema na fase de requisitos e fazendo reuso de especificações de interface.

### 1.1. Motivação

A motivação deste trabalho surgiu quando uma equipe de especialistas em Interação Humano-Computador (IHC) foi procurada pelo líder do projeto de desenvolvimento de um sistema, para propor melhorias na usabilidade deste sistema. Tal sistema computacional visa apoiar a formação do profissional de segurança pública no exercício de suas atividades corriqueiras e tem muitos recursos gráficos (textos, simulações, animações, etc) para auxiliarem o usuário a refletir sobre situações criminais. Foi percebido que muito esforço foi despendido para disponibilizar todos os recursos, não se importando com a facilidade de uso de tais recursos. Além disto, os usuários não tiveram a oportunidade de participar do desenvolvimento do sistema. Apesar de o sistema ter sido desenvolvido seguindo um dos processos mais utilizados atualmente, como o RUP [6], tal processo não apóia a consideração de fatores relativos à IHC (como: fatores humanos, necessidades de usabilidade e acessibilidade, restrições e preferências do usuário, suporte à avaliação de alternativas de projeto de interface, etc.).

Diante disto, a equipe passou a acompanhar as sessões de utilização do sistema por parte dos usuários finais, para verificar os problemas de usabilidade e identificar as necessidades de usabilidade dos usuários associadas aos problemas. Foi percebido que não havia necessidade de mudar os requisitos funcionais, mas que era preciso definir as soluções para estas necessidades (como as características gráficas - guias de estilo, de alternativas de protótipos) e implementá-las no requisito funcional associado (como por exemplo, documentando no fluxo de eventos do caso de uso, modificando as classes envolvidas neste caso de uso, etc).

Duas conclusões foram importantes para definir as características da metodologia proposta neste trabalho: primeiro, que os elementos-chave envolvidos (problema, necessidade, características, caso de uso) faziam parte da pirâmide de requisitos, que é um artefato usado para o levantamento e especificação de requisitos do RUP, assim re-projeto é uma atividade que tem que ser tratada na disciplina de requisitos do RUP; segundo, que re-uso de soluções já utilizadas em problemas recorrentes, conhecidos como padrões, servem para que facilmente se encontre e avalie possíveis soluções para os problemas identificados.

A metodologia foi necessária para guiar as equipes envolvidas a re-projetar as telas com problemas de usabilidade identificados, buscando-se ter um sistema mais amigável aos usuários. A proposta é que essa metodologia se integre a um processo de engenharia de software e que dê suporte à participação dos usuários nas decisões sobre possíveis soluções de interfaces.

## 2. Estado da Arte

Considerando que padrões de usabilidade podem prover um caminho para produzir uma interface mais consistente, dando maior suporte ao re-uso e manutenibilidade, foram estudados inicialmente alguns trabalhos que classificam tais padrões e, em seguida, foram estudados outros trabalhos que os utilizam em processos de desenvolvimento de software para dar suporte a projetos de interface.

### 2.1. Trabalhos que suportam a classificação de Padrões

Neste primeiro estudo, foi identificado que diversas são as formas de se classificar padrões, sendo essas classificações apenas visões diferentes de uma linguagem. Essas visões são construídas com a utilização de padrões ou seus atributos. Foi realizado um estudo em 06 (seis) trabalhos que utilizam padrões de IHC, e com isso foi realizado um estudo comparativo entre eles seguindo 04 (quatro) critérios. Os trabalhos, identificados com as primeiras letras do alfabeto, são os seguintes.

O trabalho A [2] teve o objetivo de construir soluções para problemas de usabilidade em cenários de contextos diversos, aplicando padrões modelados em Linguagem de Modelagem Unificada – UML e seguindo uma metodologia que segue o passo a passo, desde a identificação do problema e suas causas, até a modelagem dos padrões.

Já o trabalho B [3] identificou padrões de interface voltados para aplicações de manutenção de informações na web (que realizam inclusões, exclusões, alterações e visualizações de dados solicitados) e aplicá-los no re-projeto de telas de manutenção de um sistema de educação do Departamento Estadual de Trânsito - DETRAN-CE.

O trabalho C [4] utilizou padrões em um nível de arquitetura de software, para buscar a usabilidade ainda quando o software está sendo construído. Segundo este trabalho, existe um relacionamento direto entre decisões tomadas em nível arquitetural e o cumprimento de seus requisitos com qualidade.

O trabalho D [1] propôs um método para construção de interfaces baseadas em padrões de interação, chamado CIU, que pode fazer parte de um processo de engenharia de software OO. Os padrões utilizados generalizam uma solução relativa a projetos de interação para, assim, registrar o conhecimento sobre este tema e reutilizá-lo, facilitando a comunicação entre as pessoas envolvidas no desenvolvimento de um software.

No trabalho E [12], os padrões de projeto de IHC foram classificados para sites da Web em quatro níveis, desde o mais abstrato até o mais concreto, classificados em uma rede de padrões interligados.

No trabalho F, de Jennifer Tidwell [10], os padrões propostos focalizaram a aplicabilidade e evidência, sendo organizados em uma linguagem, chamada de Common Ground [10]. A formatação de seus padrões não é uniforme no decorrer de seu trabalho, pois não considera relevante por se relacionar a padrões.

Neste estudo comparativo, procurou-se verificar como os padrões são disponibilizados para ajudar no envolvimento ativo do usuário no processo de geração de interfaces.

## **2.2. Trabalhos que usam padrões nos processos de desenvolvimento de software**

Nesta seção serão mostrados alguns processos de desenvolvimento de software, que utilizam padrões de usabilidade, identificando suas vantagens e desvantagens. Foram pesquisados 03 (três) trabalhos, o primeiro de Vianna [11], o segundo de Acosta e Zambrano [1] e o terceiro de Phillips e Kemp [8].

O trabalho [11] propôs uma metodologia híbrida baseada na aplicação de métodos de projeto participativo na gestão de requisitos do Processo Unificado, o RUP. É uma abordagem que aplica princípios de projeto participativo no gerenciamento de mudanças; entretanto, não fornece suporte sobre o histórico dessas modificações, e nem apresenta as alternativas de padrões de IHC com representações visuais, o que poderia tornar a participação do usuário mais lúdica no processo decisório.

O segundo trabalho é o já citado anteriormente, que foi descrito por [1], propõe um método baseado no RUP para construção de interfaces considerando padrões de interação, chamado CIU (Construção de Interface de Usuário). O CIU tem como principal meta diminuir o tempo gasto em desenvolvimento de interfaces e beneficiar a integração da ES com a IHC. Entretanto, em seus exemplos de uso, propõe outros modelos que devem ser construídos além dos inúmeros já propostos pelo RUP. Sendo o RUP, por si só, um

processo complexo, composto de muitos modelos, pode ser que o método CIU inserido no RUP não diminua o tempo gasto em desenvolvimento. Essa preocupação norteou a metodologia proposta neste trabalho, para serem aproveitados os modelos já existentes no RUP, sem propor nenhum novo modelo para realizar a mudança da interface. Acredita-se que as empresas não têm pretensão de incorporar novos artefatos aos seus processos de desenvolvimento de software dedicados inteiramente a Usabilidade. Elas querem criar e aplicar boas práticas e guias de usabilidade do ponto de vista prático, para diminuir os problemas de usabilidade nos produtos construídos.

O trabalho [8] propõe um método para projeto conceitual de interfaces utilizando o RUP, chamado CONDUIT. Este método, assim como o proposto no trabalho de [1], tenta suprir o RUP no que diz respeito a um maior foco nos projetos de interfaces. O projeto de interfaces no RUP envolve dois estágios, que são: modelagem da interface (projeto conceitual) e a prototipagem da interface, porém o foco deste trabalho é na modelagem. O método CONDUIT não tratou a participação do usuário no processo, ou a utilização de formas mais concretas de auxílio na construção da interface, fazendo com que se tenha a necessidade sempre presente, da participação de especialistas em interface neste processo. Como auxílio aos especialistas não foi citada a utilização de guidelines ou padrões.

## **3. Metodologia de Re-Projeto**

Nos subitens a seguir serão descritas as principais características da metodologia proposta, detalhando suas fases.

### **3.1. Características da Metodologia**

A metodologia tem como objetivos gerais a serem atingidos: auxiliar os projetistas de interface e/ ou desenvolvedores no re-projeto das interfaces de um sistema e apoiar a participação dos usuários finais e/ ou clientes na tomada de decisões sobre alterações das interfaces deste sistema.

As originalidades desta metodologia, para facilitar a tomada de decisões sobre as manutenções na interface são as seguintes: inserção de padrões no processo de re-projeto, como forma de direcionar usuários e projetistas na análise da solução a ser implementada; e, por meio desses padrões, realizar um processo participativo de tomada de decisão. Padrões e suas representações ilustram exemplos reais de uso de um componente de interface e utilizam uma linguagem mais próxima da realidade dos usuários, bem como de projetistas menos experientes.

Para entender melhor o relacionamento entre casos de uso, problemas e padrões, considere a utilização de pirâmide de requisitos do RUP integrada com padrões; ela será usada para auxiliar projetistas e/ ou desenvolvedores na busca de soluções de re-projeto

para problemas de usabilidade, através do re-uso de padrões.

O ponto inicial da pirâmide de requisitos do RUP é a identificação de um problema (ou vários problemas). Para certo problema, são identificadas as necessidades dos usuários capazes de resolver este problema. Em seguida, são associadas as características que um sistema deve ter para dar suporte às necessidades identificadas. Por fim, são geradas soluções para essas características, geralmente em formato de modelos de casos de uso que devem conter as funcionalidades do sistema.

Percebeu-se, através de estudos, que os padrões representam soluções contendo um conjunto de características que visa a resolver um dado problema. Por exemplo, o padrão “comparação” é uma solução para um problema do usuário de dificuldade de identificar entre dois produtos qual o melhor. Esse padrão tem como característica: comparação de múltiplos critérios entre produtos. Estas características são as propriedades e/ ou restrições tecnológicas ao sistema, seus requisitos de usabilidade. Dessa forma, é possível auxiliar projetistas e usuários na identificação de uma solução, pela associação do problema identificado às possíveis características (a partir das necessidades), estando elas associadas aos padrões.

Lembrando também que na abordagem descrita em [4], é visto como um Problema de Usabilidade (PU) é analisado para que atributos de usabilidade sejam relacionados com esses PU's. Trata-se de uma rede de relacionamento na qual os atributos são relacionados a propriedades de usabilidade, capazes de prover suporte ao problema em questão; enfim, é realizada a ligação entre essas propriedades e os padrões de usabilidade.

Existe uma associação genérica no mapeamento de propriedades de usabilidade aos padrões. Não se sabe porque um *log* permite o tratamento de erros. Que características o sistema deve ter para este padrão ser escolhido? As características do sistema (propriedades e restrições do sistema) ajudam a definir as tarefas interativas que o usuário deverá usar, que estilos de interação (manipulação direta, comando, *menu*, pergunta/ resposta) e que objetos visuais (protótipos) o sistema terá. Todas estas informações podem ser encontradas na definição dos padrões.

Neste trabalho foram feitas:

- A associação de atributos e propriedades de usabilidade de Juristo às necessidades da pirâmide do RUP;
- A associação da camada de características da pirâmide do RUP, a propriedades e restrições do sistema para atender as necessidades identificadas (inexistência na rede de Juristo por ser muito genérica);
- A associação da camada de soluções da pirâmide do RUP aos padrões de usabilidade de Juristo.

## 3.2. Fases da Metodologia

A metodologia está dividida em três fases, que são: Caracterização e Análise do Problema, responsável basicamente pela identificação do problema e dos artefatos em que ele se encontra; Efetivação de Mudanças Arquiteturais, que visa identificar e testar a solução para o problema encontrado; e Documentação da Alteração, que objetiva a documentação das alterações realizadas, atualizando os artefatos necessários. Estas três fases estão subdivididas em atividades, que serão descritas nas seções subseqüentes. Alguns passos desta metodologia levarão à construção da pirâmide de caso de uso. Nas figuras a seguir, os passos que constituem a criação desta pirâmide estão marcados com um triângulo no canto inferior direito da atividade.

### 3.2.1. Fase 1: Caracterização e análise do problema.

Nesta primeira fase (ver figura 1), estão alocados os passos relacionados à identificação e análise do problema, e seguem o seguinte fluxo: localizar casos de uso, com o objetivo de localizar a funcionalidade onde o problema foi identificado; depois disso, será possível realizar o próximo passo, que é o de analisar o impacto e alterações nessa funcionalidade, que tem o objetivo de analisar os requisitos afetados e os que estão associados a esses; depois, se pode fazer a análise de soluções já fornecidas, caso existam através do histórico das ações de re-projeto efetuadas no sistema e neste próprio caso de uso; e, por fim, a identificação das necessidades a partir dos problemas colocados e a validação dessas necessidades com os usuários, podendo, quando for efetuada a alteração e o resultado não for o esperado, retornar à identificação dessas necessidades.



Fig. 1. Metodologia Proposta: Detalhamento da Fase 1

**3.2.2. Fase 2: Efetivação de Mudanças Arquiteturais.** A principal meta desta fase (ver figura 2) é a de escolher, dentre as alternativas de solução a serem propostas, a solução a ser implementada como

forma de resolver o problema identificado na fase anterior. Para que se tenha um conjunto de alternativas de solução, deverão ser propostas, diante das necessidades dos usuários definidas anteriormente, características que a funcionalidade (caso de uso) deve conter.

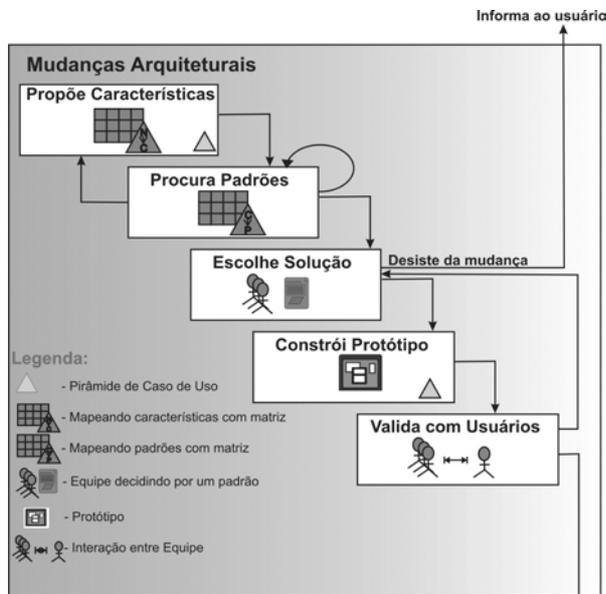


Fig. 2. Metodologia Proposta: Detalhamento da Fase 2

**3.2.3. Fase 3: Documentação da Alteração.** Esta fase é composta pelas seguintes atividades: alteração do fluxo de eventos do caso de uso, incremento do histórico de re-projeto e encaminhamento do protótipo para implementação. Essas atividades estão ilustradas na figura 3.

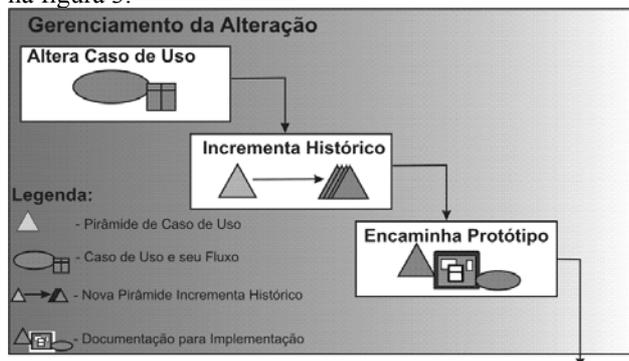


Fig. 3. Metodologia Proposta: Detalhamento da Fase 3.

## 4. Estudo de Caso

A metodologia de re-projeto de interfaces proposta foi aplicada no re-projeto (da interface) de um sistema. O objetivo geral, ao final deste experimento, foi o de vivenciar a possibilidade de entender melhor a integração desta metodologia com um processo de desenvolvimento, como também verificar se as metas a serem alcançadas com sua utilização foram atingidas.

Os objetivos desta aplicação foram os seguintes: (i) Como encontrar uma solução a partir de vários

critérios, tais como: atributos e propriedades de usabilidade definidos por Juristo [4]; restrições tecnológicas e preferências dos usuários; implicação da mudança sobre o que já foi feito; padrões (já conhecidos em sistemas semelhantes). (ii) Como fazer para tratar a solução: que artefatos modificar; que informações guardar (lições aprendidas); e como e quando envolver o usuário.

Pelos estudos realizados durante este trabalho, foi possível perceber que o RUP é carente de suporte ao tratamento de problemas de usabilidade. O experimento realizado mostrou uma empresa que utiliza o RUP tratando das mudanças, assim como algumas facilidades trazidas pelo uso da metodologia proposta por esta empresa. O experimento foi realizado na manutenção de um sistema da Secretaria da Fazenda do Estado do Ceará – SEFAZ, chamado de Sistema ECF, que utilizou o processo de desenvolvimento proposto pela instituição, que tem como base o RUP.

O Sistema Emissor de Cupons Fiscais, ou SECF, foi desenvolvido pela Secretaria da Fazenda do Estado do Ceará, para que pudesse haver um controle de todos os equipamentos emissores de cupons fiscais em uso no Estado, tendo-se um histórico desses equipamentos, armazenando seus dados desde o pedido de uso do equipamento, passando pelas intervenções técnicas realizadas nele, até sua baixa (cessação de uso) para um contribuinte. Neste trabalho, foi utilizada a metodologia proposta quando, no processo de desenvolvimento de um novo módulo para este sistema, foram solicitadas alterações nas interfaces até então definidas, as quais surgiram quando o sistema estava sendo utilizado por usuários.

Inicialmente, no fim da terceira iteração do projeto, ou seja, após a finalização da última iteração da fase de elaboração deste projeto, um grupo de cinco (05) usuários foi convidado a analisar os protótipos confeccionados, porém como não chegaram a navegar por esses protótipos, este grupo aprovou as telas sem solicitações de alterações significativas. Algumas funcionalidades foram alteradas, relacionadas com requisitos funcionais definidos e não implementados como esperado pelos usuários, outras dúvidas foram retiradas com a documentação que já havia sido aprovada por eles.

Posteriormente, após a conclusão da segunda iteração da fase de construção, ou seja, ao final da quinta iteração, seis (06) dos doze (12) casos de uso idealizados para o módulo de controle de etiquetas haviam sido finalizados. Com isso, as funcionalidades implementadas por esses casos de uso foram apresentadas para o mesmo grupo de usuários que havia homologado os protótipos. Este grupo de usuários representa os gestores do sistema e fiscais, responsáveis pela utilização da grande maioria das funcionalidades até então desenvolvidas.

De início foram apresentadas as funcionalidades, uma a uma, seguindo uma seqüência de necessidades das ações dos usuários. Em seguida, alguns usuários

começaram a utilizar as funcionalidades para que, assim, pudessem se ambientar e, no caso em que foram identificados cinco (05) problemas no uso (usabilidade) por parte dos usuários e/ ou desenvolvedores, o gerente de projetos foi abordado para efetuar a documentação desses problemas.

Uma vez identificado um problema no uso das funcionalidades, e sendo este problema inserido como uma solicitação de mudança, foi então gerado o primeiro artefato da metodologia de re-projeto. Um problema identificado pelos usuários, quando do uso da aplicação, foi na funcionalidade ‘Inclui Etiqueta em um Pedido de Uso’.

Através da metodologia, foi assim localizado o caso de uso, com seus fluxos de eventos, que são os artefatos iniciais para que, seguindo a metodologia, se construísse a pirâmide (definida como pirâmide do caso de uso) exibida na figura 4. Através desta pirâmide é possível se ter um mapeamento desde o problema identificado até a solução escolhida para aquele problema, passando pela escolha dos padrões.

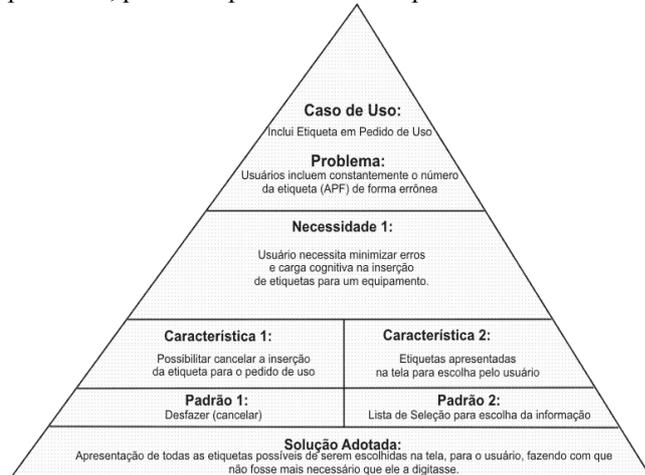


Fig 4. Pirâmide gerada a partir de um problema.

## 5. Uso dos Casos de uso e Escolha dos Padrões

Um padrão, segundo Juristo et al (2003), é classificado quanto às seguintes propriedades de usabilidade: minimizar carga cognitiva, orientação, tratamento de erros, prover feedback, etc. Na escolha, apesar do mapeamento entre características e padrões (propriedades de usabilidade) ser feito, simplesmente, levando em consideração essas propriedades, deve ser lembrado que essa escolha acontece juntamente com o usuário, levando-se em consideração as restrições tecnológicas, suas preferências e outros requisitos não funcionais (custo, desempenho, etc.). Isso serve como um filtro na escolha dos padrões a serem utilizados e, como são poucos os padrões cadastrados inicialmente neste trabalho, a busca não se tornou tão empírica e cansativa, como pode parecer pela estrutura.

Voltando ao experimento, para a característica 1 (ver figura 4) , foram encontrados vários padrões que

poderiam substituir o campo de edição pela exibição das etiquetas possíveis de serem escolhidas. A intenção inicial do usuário era que fosse feita a substituição por um elemento botão de rádio (Radio Button), porém foi visto no padrão que o representa que este não era apropriado para mais de sete itens. Com isso, a escolha foi para a utilização de uma lista de seleção na qual apareceriam de 5 em 5 itens, para a escolha de um só. Para a característica 2 foi encontrado o padrão ‘desfazer’ que, na lista de padrões, disponibilizava apenas um tipo de um padrão que implementava esta ação. Para a implementação desse padrão identificado, o caso de uso teria que passar por uma modificação funcional, pois não existia nenhuma função de excluir etiqueta da solicitação. Como essa alteração se tornaria mais complexa, pois os saldos teriam que ser atualizados sempre e isso ocasionaria em impacto em outros casos de uso, foi decidido, por projetistas e usuários, apostar apenas na troca do campo de edição por uma lista, ou seja, por implementar apenas o primeiro padrão.

O processo de não validação da implementação do segundo padrão encontrado ocorreu durante a validação dos usuários no protótipo. Isso foi interessante porque não houve custo de implementação. De posse do padrão implementado em forma de protótipo e validado pelos usuários, pôde-se passar para a fase seguinte, de documentação da alteração.

Ocorreu o mapeamento de necessidades, características e padrões utilizando-se os atributos, propriedades de usabilidade, restrições tecnológicas, preferências dos usuários, entre outros. Foi possível verificar as associações feitas entre os atributos confiabilidade e eficiência de uso às propriedades tratamento de erro e minimização da carga cognitiva, chegando aos padrões em forma de solução. No caso das preferências consideradas, o grupo de usuários utilizado pode ser tido como não representativo de seu estereótipo, entretanto foi o que se conseguiu trabalhar em um primeiro momento.

## 6. Discussão e Trabalhos Futuros

O conceito de padrões de usabilidade foi inserido na pirâmide de requisitos do RUP, mapeados por meio de matrizes de rastreabilidade, para que, assim, se tivesse uma forma fácil de encontrar soluções para problemas de usabilidade por via da análise de requisitos, mantendo a rastreabilidade desses requisitos. Além disso, a utilização de padrões e representações gráficas pode favorecer a participação dos usuários no processo de tomada de decisão sobre alterações no sistema, pois, assim, o processo de desenvolvimento passou a ser mais bem entendido pelos usuários. Dentro da validação das mudanças por parte dos usuários, foi acrescentada a utilização de protótipos na implementação dos padrões.

Espera-se, com isso, ter contribuído para o desenvolvimento de sistemas com mais qualidade em

seu uso (usabilidade), sem onerar o custo do projeto quanto à criação de muitos artefatos novos para ele, principalmente no que diz respeito a alterações de sistemas em manutenção, nos quais se tem toda a documentação já desenvolvida. A solução proposta neste trabalho pode ser aplicada para buscar solucionar qualquer problema de usabilidade onde houver mudança nos requisitos.

Como trabalho futuro será necessário provar que a pirâmide de requisitos integrada com padrões de usabilidade, e que os protótipos associados aos padrões podem ajudar a reduzir os custos no tempo de manutenção e aumentar a satisfação do usuário final. Acredita-se que padrões dão mais facilidade de entendimento e incentivam a participação do usuário na escolha de alternativas de solução para uma interface, devido apresentarem detalhes e exemplos concretos, mas é preciso verificar mais profundamente esta crença.

## 7. Conclusão

Este trabalho identificou as atividades e disciplinas do RUP nas quais a metodologia foi inserida, propondo um fluxo de mudanças por onde a solicitação de uma alteração de interface deve passar para que seja implementada, tornando-se possível documentar todos os tipos de solicitações de mudança de interface. Nele a busca por soluções para problemas encontrados foi bastante facilitada, através da rede de relacionamento e das características adicionadas pelos padrões. Ainda sobre os padrões, eles realmente auxiliaram a participação de usuários, principalmente no que diz respeito a decisões sobre escolha da solução (padrão) a ser utilizada. A validação, ainda em nível de requisitos por meio da implementação dos padrões em protótipos, já oferecia uma real noção sobre o que o usuário desejava, uma vez que tinham indicado os exemplos dentro dos padrões escolhidos, o que foi muito importante para a satisfação de todos. Imagina-se ainda que o tempo de implementação foi diminuído quando projetistas sabiam exatamente o que iriam alterar e qual o resultado esperado.

## 8. Referências

- [1] Acosta, E., Zambrano, N. Patterns and objects for user interface construction. *Journal of Object Technology (JOT2004)*, vol. 3, Março-Abril 2004, pp. 75-90. Disponível em: <[http://www.jot.fm/issues/issue\\_2004\\_03/article1](http://www.jot.fm/issues/issue_2004_03/article1)>. Acesso em 17.05.2005.
- [2] Gomes, F. A. Soluções de usabilidade usando padrões de projeto modelados em UML. Monografia de Graduação. Fortaleza: Universidade de Fortaleza, 2004.
- [3] Meireles, F.M.S. Uma análise comparativa de interfaces de usuários na web para aplicações de manutenção. Monografia de Graduação. Fortaleza: Universidade de Fortaleza, 2004.
- [4] Juristo, N., Lopez, M., Moreno A., Sanchez, M.I. Improving software usability through architectural patterns. *International Conference on Software Engineering*. Maio de 2003. Portland: Oregon USA.
- [5] Junior, S.L.; Silva, J.C. Processos de planejamento de reengenharia de software apoiados por princípios de usabilidade. *Congresso Latino-Americano de Interação Humano-Computador, CLIH 2003*, Rio de Janeiro, 2003.
- [6] Kroll, P. Kruchten, P. *Rational Unified Process made easy: a practitioner's guide to the RUP*. Addison-Wesley Publishers Ltd. 2003.
- [7] Kruchten, P., Ahlqvist S. e Bylund S. User interface design in the Rational Unified Process, in object modeling and user interface design. Van Harmelen, Addison-Wesley, Reading, Massachusetts. 2001.
- [8] Phillips, C.H.E., Kemp, E. In support of user design in the Rational Unified Process. *Anais do AUIC 2002*, Melbourne, Australia:28-30 Janeiro 2002, p21-27.
- [9] Reis, C. R. Uma visão geral do Bugzilla, uma ferramenta de acompanhamento de alterações. *XVI Simpósio Brasileiro de Engenharia de Software, SBES 2002*, Gramado – RS: 2002.
- [10] Tidwell, J. *Interaction design patterns*. Conferência de Linguagens Padrões de Programas, PLoP'98. 1998, Monticello, EUA.
- [11] Vianna, E. C da C. M. Estudo e proposta de práticas participativas na gestão de requisitos. Dissertação de Mestrado Profissional. Campinas: Universidade Estadual de Campinas, 2004.
- [12] Welie, M.V., Veer, G.C. Pattern languages in interaction design: structure and organization. In: *Proceedings of Interact '03*, 1-5 Setembro, Zürich, Suíça, p527-534, IOS Press, Amsterdam, Holanda.