

DESENVOLVIMENTO DE UM FÓRUM DE DISCUSSÕES ON-LINE NO ÂMBITO DA INTERAÇÃO HUMANO-COMPUTADOR

THE DEVELOPMENT OF AN ONLINE FORUM FOR HUMAN-COMPUTER INTERACTION

¹Rodrigo Mattos Mengue, ²Juliano Ratusznei, ³Lucélia de Souza

¹rodrigo@rcenter.com.br, ²julianojules@gmail.com, ³lucelia@unicentro.br

Universidade Estadual do Centro-Oeste - UNICENTRO

Departamento de Ciência da Computação - DECOMP

Recebido para publicação em 20/05/2008

Aceito para publicação em 30/06/2008

RESUMO

Este trabalho apresenta o desenvolvimento de um fórum de discussões on-line relacionado ao ensino da Programação de Computadores, sendo modelado no contexto da Engenharia Semiótica da área de Interação Humano-Computador. Foram aplicadas avaliações de comunicabilidade em interfaces *web* em fóruns nesse contexto, por diferentes perfis de usuários. Considerando os resultados dos testes, o *design* foi modelado por meio da linguagem MoLIC (*Modeling Language for Interaction as Conversation*) e implementado em PHP/PostgreSQL. Espera-se que os usuários relacionados à área de programação possam interagir com o ambiente de forma que a interface seja um facilitador para a aprendizagem.

Palavras-Chaves: Interação humano-computador. Engenharia semiótica. MoLIC. Sistema de ajuda on-line. Fórum on-line.

ABSTRACT

This paper presents the development of an online discussion forum related to the teaching of computer programming. This project is being modeled in the context of the Engineering Semiotics in the area of Human-Computer Interaction. Users with different profiles evaluated the development of communicability in web interfaces in forums. Considering the results of the tests, the design was developed by means of the MoLIC language (*Modeling Language for Interaction as Conversation*) and implemented in PHP/PostgreSQL. It is expected that users from the programming area can interact with the environment so that the interface becomes a facilitator for their learning.

Keywords: Human-computer interaction. Semiotic engineering. MoLIC. Online help system. Online forum.

1 Introdução

Ingressantes em cursos de TI podem apresentar dificuldades no aprendizado da Programação de Computadores, necessitando de fazer uso de recursos tais como acesso a livros, artigos e técnicas de estudo apropriadas.

A *Internet* também se constitui como fonte de pesquisa, composta por diversas ferramentas, que possuem um potencial de acesso à informação, como é o caso dos fóruns on-line, sendo também utilizados pela oportunidade de troca de experiências.

Entretanto, o desenvolvimento desses sistemas de discussões on-line, na maioria das vezes, é feito por meio de Sistemas Gerenciadores de Conteúdo, *Content Management System* - CMS, que são recursos para o desenvolvimento de fóruns de forma padronizada e sem a necessidade de se fazer uso de uma linguagem de programação.

Nesse sentido, foi realizado um estudo em ferramentas para gerenciamento de fóruns já existentes na *Internet*, tais como JOOMLA (JOOMLA, 2007) e PHPBB (PHPBB, 2007), sendo observados alguns pontos que podem ser melhorados quanto às propriedades da área de Interação Humano-Computador - IHC.

Partindo dessa análise, e com base na teoria de *design* de interfaces denominada Engenharia Semiótica - ES da área de IHC, é proposto um ambiente voltado à Programação de Computadores, denominado Fórum Pascal (FÓRUM PASCAL, 2007).

A escolha dessa linguagem como base do ambiente deu-se pelo fato de a mesma ser ainda bastante utilizada no meio acadêmico como uma linguagem didática para o ensino de Programação.

O desenvolvimento é fundamentado com o objetivo de se propor um ambiente projetado para apoiar as necessidades do usuário durante a interação, incluindo a modelagem do sistema de ajuda on-line da ES.

Este artigo segue conforme: a seção 2 apresenta a fundamentação teórica para o desenvolvimento do Fórum Pascal; a seção 3 apresenta o estudo dos ambientes existentes; a seção 4

apresenta a modelagem no contexto da ES e sua implementação; a seção 5 apresenta as conclusões e os trabalhos futuros, seguida das referências.

2 Fundamentação teórica

Os tópicos subsequentes abordam o embasamento teórico relacionado à área de IHC.

2.1.1 Interação humano-computador

A IHC é uma área que trata do *design*, avaliação e implementação de sistemas computacionais interativos para o uso de pessoas, devendo analisar todos os fatores envolvidos na interação usuário-sistema (ROCHA; BARANAUSKAS, 2003).

Nessa área, existem duas teorias principais de *design* de interfaces que fundamentam o desenvolvimento de sistemas: de um lado, a Engenharia Cognitiva - EC (NORMAN, 1986), sendo esta a teoria mais utilizada em IHC, que apresenta como propriedade-chave a usabilidade das interfaces, ou seja, a facilidade de uso dos sistemas.

Por outro lado, ES constitui uma teoria recente de *design* de sistemas que considera como propriedade-chave a comunicabilidade, segundo a qual a interação só ocorre quando houver um ato comunicativo entre o *designer* (projetista) e o usuário (DE SOUZA, 2005).

Considerou-se como base neste trabalho a ES por esta abranger a EC, visto que não basta apenas os sistemas apresentarem usabilidade e funcionalidade, se estas não forem bem comunicáveis para que os usuários usufruam dos recursos disponíveis em um ambiente.

2.1.2 Engenharia semiótica

A palavra semiótica vem do grego que significa "*Ciência que estuda os signos e os sistemas de significação*" (PEIRCE, 1931-1958). A definição de signo para Peirce (1931-1958) é: "*qualquer coisa que signifique algo para alguém*", ou seja, qualquer objeto ou ação presente na interface pode

ser definido como um signo. Como exemplo de um signo, pode-se citar os campos em um formulário de cadastro.

A ES visualiza a interface como uma mensagem que é transmitida do *designer* aos usuários do sistema, para que estes possam interagir com o ambiente de forma clara e amigável, facilitando a realização de suas tarefas.

2.1.3 Linguagem MoLIC

A linguagem MoLIC (*Modeling Language for Interaction as Conversation*) possibilita a especificação e modelagem de sistemas computacionais no contexto da ES. Utiliza a metáfora da interação por meio de conversação, permitindo ao *designer* refletir sobre a comunicação usuário-sistema (BARBOSA; PAULA, 2003).

A MoLIC compreende três representações relacionadas: a ontologia de domínio e de aplicação, diagrama de metas dos usuários e um modelo da interação, descritos a seguir:

- *Signos de Domínio e de Aplicação*: a partir dos Cenários (narrativas textuais da interação), são extraídos os signos de domínio e de aplicação. Estes signos são tratados geralmente como dados, e dentro da ES passam a ser um vocabulário compartilhado entre o *designer* e o usuário. Servem de base para os *designers* identificarem o nível de informação necessária a ser disponibilizada na interface, como, por exemplo, é o caso dos signos de aplicação, que necessitam de mais informações na interface porque não são comuns no dia-a-dia dos usuários, como é o caso do identificador alfanumérico.
- *Metas do Usuário*: a primeira etapa ao utilizar a linguagem MoLIC é extrair, a partir dos Cenários, quais as principais funcionalidades do sistema relacionadas aos diferentes perfis de usuários, considerando: qual o nível dos usuários, quais os conhecimentos necessários para usar o sistema, qual a frequência com que os objetivos esperam ser alcançados, entre outros. As metas são representadas no

Diagrama Hierárquico de Metas (DHM), que é expandido para o Diagrama Hierárquico de Tarefas (DHT), contendo de forma textual a descrição dos signos e dos tipos de mensagens de erros que podem ocorrer durante a interação e/ou serem prevenidos.

- *Diagrama de Interação (DI)*: com o DHM elaborado, os *designers* têm elementos para construir o Diagrama de Interação e assim poderem dar forma à solução computacional. A interação é vista como um ato de conversação, em que o usuário “conversa” com o sistema, mesmo que mentalmente, e as “falas” do *designer* são representadas por meio das cenas de interação.

2.1.4 Sistema de ajuda on-line

O sistema de ajuda on-line é uma forma alternativa de fornecer apoio ao usuário no momento em que o mesmo apresentar dúvidas durante a interação. Sabe-se que, na maioria dos casos, os usuários não acessam o sistema de ajuda, visto que o mesmo é desenvolvido de forma genérica, levando o usuário a ler informações demasiadas para conseguir solucionar suas dúvidas e, mesmo assim, nem sempre isso ocorre.

Entretanto, o sistema de ajuda proposto em Silveira (2002) é contextualizado e deve ser especificado em paralelo à modelagem da interação.

Apresenta seis modelos para sua especificação: Domínio, Aplicação, Tarefas, Agentes/Usuários, Interface e Interação. Isso difere de um sistema de ajuda tradicional, servindo para apoiar os usuários no momento em que apresentarem alguma dúvida de interação. Os seis modelos encontram-se descritos a seguir:

- *Modelo de Domínio*: apresenta informações relativas ao domínio da aplicação, baseado na natureza do trabalho e no detalhamento dos seus respectivos signos.
- *Modelo da Aplicação*: contém informações relativas à aplicação, sua utilidade, vantagens, atividades e opções possíveis, assim como os signos relacionados.

- *Modelo de Tarefas*: apresenta informações relativas às tarefas que os usuários podem realizar, contendo sua descrição, a utilidade, o motivo pelo qual deve ser realizada, de qual tarefa-mãe provém, o operador que a conecta à próxima tarefa, as condições necessárias para sua execução e os signos de domínio e de aplicação relacionados.
- *Modelo de Agentes/Usuários*: apresenta informações relativas aos usuários da aplicação, relacionando os papéis (sendo que para cada papel descreve-se seu nome, as tarefas que o mesmo pode executar e os usuários relacionados) e os perfis de usuários existentes, identificando a forma como o usuário prefere ou pode interagir com a aplicação.
- *Modelo de Interface*: contém informações sobre os elementos de interface da aplicação, sendo que, para cada elemento, encontram-se seu tipo, os valores que ele pode assumir, seu valor *default*, sua localização na interface e os signos de domínio e aplicação relacionados.
- *Modelo de Interação*: contém informações sobre as possíveis formas de interação na aplicação, especificando como efetivamente ocorrerá a conversa ao se realizar uma determinada tarefa na aplicação.

2.1.5 Avaliação de comunicabilidade (MAC)

Atualmente é inviável disponibilizar um software ou produto para uso sem que o mesmo antes tenha passado por um processo de avaliação de interfaces, pois, com os resultados dos testes, é possível prevenir muitos problemas que poderiam aparecer posteriormente com seu uso, diminuindo inclusive o número de versões a serem lançadas.

Um dos métodos de avaliação de interfaces da ES e que têm se destacado em IHC é o Método de Avaliação de Comunicabilidade - MAC (PRATES; SOUZA, 2000). Trata-se de um método empírico, que envolve a presença de usuários para sua aplicação, apresentando resultados qualitativos e interpretativos sobre a interação.

A fim de se identificar como a mensagem *designer*-usuário está sendo recebida e entendida pelo usuário, o MAC consiste em três etapas: a etiquetagem (associação de expressões comunicativas às dúvidas dos usuários), a interpretação (análise das expressões segundo algum critério pré-definido) e a geração do perfil semiótico (reconstrução da meta-comunicação *designer*-usuário) (PRATES; SOUZA, 2000).

A seguir, é apresentado o estudo feito em sistemas gerenciadores de conteúdo e em fóruns desenvolvidos segundo este aspecto.

3 Estudo dos ambientes existentes

As seções subsequentes apresentam a análise dos gerenciadores estudados, incluindo testes de comunicabilidade, aplicados com usuários reais.

3.1. Análise dos CMSs

Antes do desenvolvimento do ambiente, os autores desse artigo fizeram análises nos sistemas para gerenciamento de conteúdo para o desenvolvimento de fóruns.

Foram analisadas questões de praticidade, facilidade de uso e usabilidade nos CMSs supracitados, sendo ambos de licença *open source* (código livre) e que utilizam banco de dados *MYSQL*.

Os fóruns se mostraram práticos pela facilidade de instalação, configuração e, principalmente, quanto ao gerenciamento. O processo de autoinstalação é apresentado em ambos os gerenciadores estudados, sendo de fácil utilização e configuração, sem a necessidade de fazer uso de uma linguagem de programação.

Os módulos de gerenciamento são eficazes e proporcionam ao administrador a aplicação de filtros para inibir a criação de tópicos com conteúdo indesejáveis, assim como fornecem um gerenciamento eficaz sobre os usuários, permitindo dividi-los em grupos de permissões que limitam o acesso de determinados usuários a um fórum.

Como resultado do estudo preliminar das propriedades de usabilidade e comunicabilidade

da IHC nos CMSs analisados, destacam-se:

- i. as interfaces são baseadas em *templates* (separação em estrutura e conteúdo) que permitem ao desenvolvedor modificar cores e imagens dos fóruns. Porém, os modelos suportados pelas plataformas não oferecem liberdade ao *designer* de modificar a estrutura de funcionamento do fórum, como a ordem e disposição dos módulos no *design*.
- ii. as ferramentas possibilitam a mudança no idioma dos textos de mensagem de erros, mas não proporcionam ao *designer* realizar o tratamento sobre as mensagens de erro apresentadas aos usuários. Ou seja, se algum erro ocorrer, diferente daqueles programados no fórum, não existe a possibilidade de o administrador tratar a mensagem de erro, deixando o usuário sem retorno sobre os motivos que o impediram de realizar determinada tarefa;
- iii. não existe um método para implantação de um sistema de ajuda on-line que auxilie o usuário na realização de suas tarefas. Apesar de ambos os ambientes apresentarem ajuda em alguns campos de formulários do módulo de gerenciamento, nenhum apresenta a possibilidade de habilitar o mesmo no fórum com um todo.

3.2 Análise de fóruns desenvolvidos por meio de CMS

Foi feita uma pesquisa por meio da utilização de questionários com vinte pessoas com os seguintes perfis: conhecimento ou não em Informática e também com professores da área, com o objetivo de se obter um levantamento das principais dificuldades de interação em fóruns on-line, desenvolvidos por meio de CMS. Como resultado, destacam-se:

- i. o cadastro é obrigatório na maioria dos ambientes, fato que diminui o interesse pelo acesso ao sistema;
- ii. a busca por tópicos e temas de interesse, na

- maioria das vezes, é uma tarefa complexa;
- iii. o processo de recebimento de *e-mails* de confirmação de cadastro é demorado;
- iv. a inexistência e os problemas de comunicação nas mensagens de erro;
- v. a falta de *feedback* que confirme a execução das tarefas e ações do usuário nos ambientes.

Um exemplo dos pontos destacados acima é o fato de um usuário levar muito tempo para localizar um tópico com o tema de interesse e, quando opta por interagir com o tópico, acaba sendo 'forçado' a efetuar o *login*, ou até mesmo a fazer um cadastro para continuar a operação. Porém, na quase totalidade dos casos, o usuário perde todo o tempo despendido na localização do tópico de interesse, precisando refazer a tarefa.

3.3 Aplicação do MAC em fóruns existentes

Para a aplicação do MAC, foram selecionadas vinte pessoas de diferentes perfis, variando o conhecimento em Informática e quanto ao uso de fóruns (professores de Informática, secretários, estagiários e alunos).

Os participantes do teste receberam instruções sobre sua aplicação e também sobre questões éticas, tais como: foi explicado que quem estava sendo avaliado era o sistema e não os usuários, que os mesmos poderiam desistir de realizar o teste a qualquer momento e que seus dados jamais serão revelados, apenas os resultados para confirmação do teste.

Os testes foram aplicados em três ambientes destinados à programação como um todo e desenvolvidos por meio de CMS, não especificamente sobre a linguagem Pascal, visto que os mesmos apenas apresentam categorias e/ou tópicos sobre essa linguagem.

A realização dos testes ocorreu em um cenário construído em um laboratório de pesquisa para que interferências externas não atrapalhassem seu desenvolvimento. Os autores desse artigo atua-

ram como observadores diretos. A interação dos usuários foi registrada com o software HyperCam (HYPERIONICS, 2007) (que é *shareware*, sendo usado em uma versão de testes), o qual gera um

rio não entendeu o diálogo do *designer* ou de seu representante na interface. Em ambos os casos, a interação é momentaneamente interrompida.

Portanto, cita-se que os principais problemas identificados nos fóruns avaliados relacionam-se com a forma de execução de algumas tarefas e com o conteúdo apresentado nas mensagens de erro ao usuário, por meio do levantamento das expressões de dúvidas ocorridas e seus tipos de problemas. A seção a seguir aborda um exemplo de problemas identificados com a aplicação do teste de comunicabilidade e seu *redesign*.

3.3.1 Exemplo de problemas identificados com a aplicação do MAC

Interação versus Comunicabilidade		Ocorrência das interjeições		
Interjeição	Problema	Fórum 1	Fórum 2	Fórum 3
Cadê?	Durante a navegação	3	7	5
E agora?		6	-	3
Êpa!	Na atribuição de significado e/ou de navegação	-	2	2
Onde estou?		-	-	-
Que é isso?	Na atribuição de significado	1	-	-
Por que não funciona?		4	1	3
Ué, o que houve?		10	-	5
Vai de outro jeito	Affordance baixa / inexistente / recusada	1	4	1
Não, obrigado.		1	-	3
Desisto!	FRACASSO NA TAREFA.	-	4	1

Tabela 1 - Comparativo entre as expressões comunicativas identificadas nos fóruns on-line (DE SOUZA ET AL, 1999).

filme da interação.

A Tabela 1 apresenta a análise dos resultados dos testes e a ocorrência das expressões comunicativas que identificam as principais rupturas na comunicação usuário-sistema.

A etapa de etiquetagem do MAC consistiu em os autores assistirem o registro da interação de cada usuário, identificando possíveis rupturas na comunicação, que foram associadas às expressões comunicativas, tais como “*Cadê?*”, “*E agora?*”, “*O que é isto?*”, “*Êpa!*”, entre outras.

Assim, as expressões que ocorreram com maior frequência foram: “*Cadê?*” e “*Ué, o que houve?*”, caracterizando-se como falhas de comunicabilidade relacionadas ao problema de navegação e de atribuição de significado, respectivamente.

Quanto à interpretação das expressões, ambas caracterizam-se como falhas temporárias, sendo que no caso de “*Cadê?*”, a semiose, ou seja, a interpretação do usuário é temporariamente interrompida porque o usuário não encontra a expressão apropriada para sua comunicação, e a expressão “*Ué, o que houve?*”, indica que o usuá-

Como exemplo de um problema identificado com a aplicação do MAC e posteriormente tratado no ambiente proposto, cita-se o caso de quando os usuários tentam postar uma mensagem e, após concluir a tarefa, recebem uma mensagem de erro, questionando-se “*Cadê?*”, como um problema durante a navegação, constituindo-se numa falha de interação temporária.

Em seguida, ao voltar à página anterior e tentar novamente postar a mensagem, recebem outro tipo de mensagem comunicando que houve falha na postagem. Porém, assim que os usuários retornam ao fórum, a mensagem encontra-se devidamente postada.

Nesse contexto, a expressão “*Ué, o que houve?*”, é associada à Figura 1, e também se constitui em uma falha temporária, caracterizando-se como um problema de atribuição de significado quanto ao texto da mensagem de erro, onde o usuário tenta realizar uma tarefa e o sistema acaba retornando uma mensagem inapropriada: “*Mensagem não postada*”, com a seguinte mensagem fora de contexto: “*Não é permitido spam neste fórum!*”.



Figura 1 - Mensagem de erro quanto à postagem de uma mensagem em um fórum avaliado.

Portanto, neste caso, o ambiente necessita de reformulação quanto à visualização da postagem e à descrição do conteúdo das mensagens de erro, apresentando o que ocorreu e o que o usuário deveria fazer para corrigir o problema.

A seção a seguir apresenta a geração do perfil semiótico com a aplicação do MAC, contemplando a modelagem e a implementação do ambiente proposto, tratando os problemas identificados no contexto da ES.

4 Modelagem do fórum pascal

A modelagem do ambiente proposto leva em consideração os resultados da aplicação do MAC e foi desenvolvida por meio da utilização dos diagramas especificados com a MoLIC, como forma de facilitar a interação, os quais são apresentados a seguir, assim como a implementação do ambiente.

4.1 Cenários

A seguir, um exemplo de um Cenário elaborado para ‘Postagem de uma Mensagem’ no Fórum, com destaque para signos que são extraídos do próprio cenário para especificação na Tabela de Signos.

“Lorena está aprendendo a programar em Pascal. Ela já se inscreveu para participar do Fórum

*Pascal. Resolve tirar algumas dúvidas, e pretende utilizar o fórum para trocar ideias e sanar suas dificuldades. Ela acessa a página principal e, em destaque, visualiza o formulário de login. Digita seu **usuário** e sua **senha** e acessa a página principal do sistema. Como sua maior dificuldade está em entender o funcionamento de estruturas de repetições aninhadas, ela busca uma categoria que contemple esse assunto. Seleciona a **categoria** “Estruturas de Repetição”. Ela está disposta a abrir um **tópico** para discutir o uso da estrutura aninhada, então acessa os tópicos já abertos para verificar se não existe algum que trate do assunto pretendido. Como não encontra, resolve abrir seu próprio tópico utilizando o botão “Novo Tópico”. A tela que se abre a seguir pede que ela insira alguns campos, sendo todos obrigatórios. Entre eles, contam-se: **Nome do Tópico**, **Descrição do Tópico** e **Mensagem**, onde ela cita suas dificuldades. Agora ela vai aguardar até que algum usuário possa ajudá-la.”*

4.2 Tabela de signos

A Tabela de Signos contém a descrição dos signos que serão utilizados no sistema, incluindo seu respectivo tipo: Domínio ou Aplicação. A Tabela 2 apresenta um exemplo da Tabela de Signos referente ao signo composto *Mensagem*, contendo dois signos classificados como de domínio do usuário, *Mensagem* e *Data postagem*, não necessitando,

portanto, de informações explicativas a seu respeito na interface.

Tabela 2 - Um exemplo da Tabela de Signos.

Signo composto: Mensagem	
Mensagem	Domínio
Data postagem	Domínio

4.3 Diagrama hierárquico de metas (DHM)

O DHM fornece uma visão geral da aplicação, envolvendo as metas e a ordem em que elas devem estar distribuídas no sistema. Permite também uma organização na distribuição de módulos pelos componentes de uma equipe de desenvolvimento.

Por trabalhar com a ordem lógica em que

as metas estão distribuídas, o DHM inicia através do acesso do usuário ao sistema. Neste caso, pela execução de uma tarefa obrigatória (T1 – Acessar o Fórum), como mostra a Figura 2.

Através do DHM é possível traçar os caminhos que deverão ser percorridos pelo usuário para realizar seu objetivo com o sistema. Por exemplo, para o usuário alterar sua imagem no Fórum Pascal, será preciso percorrer o seguinte caminho, conforme apresenta a Figura 1:

- i. T1 – Acessar o fórum;
- ii. T2 – Efetuar o *login*;
- iii. I – Alterar o cadastro;
- iv. K – Alterar a imagem.

Para cada meta do DHM da Figura 2, é feita a especificação hierárquica das tarefas para sua realização e a apresentação da ordem em que as ações devem ser executadas, seguidas das possíveis mensagens de erro, as quais podem ser de prevenção e/ou tratamento.

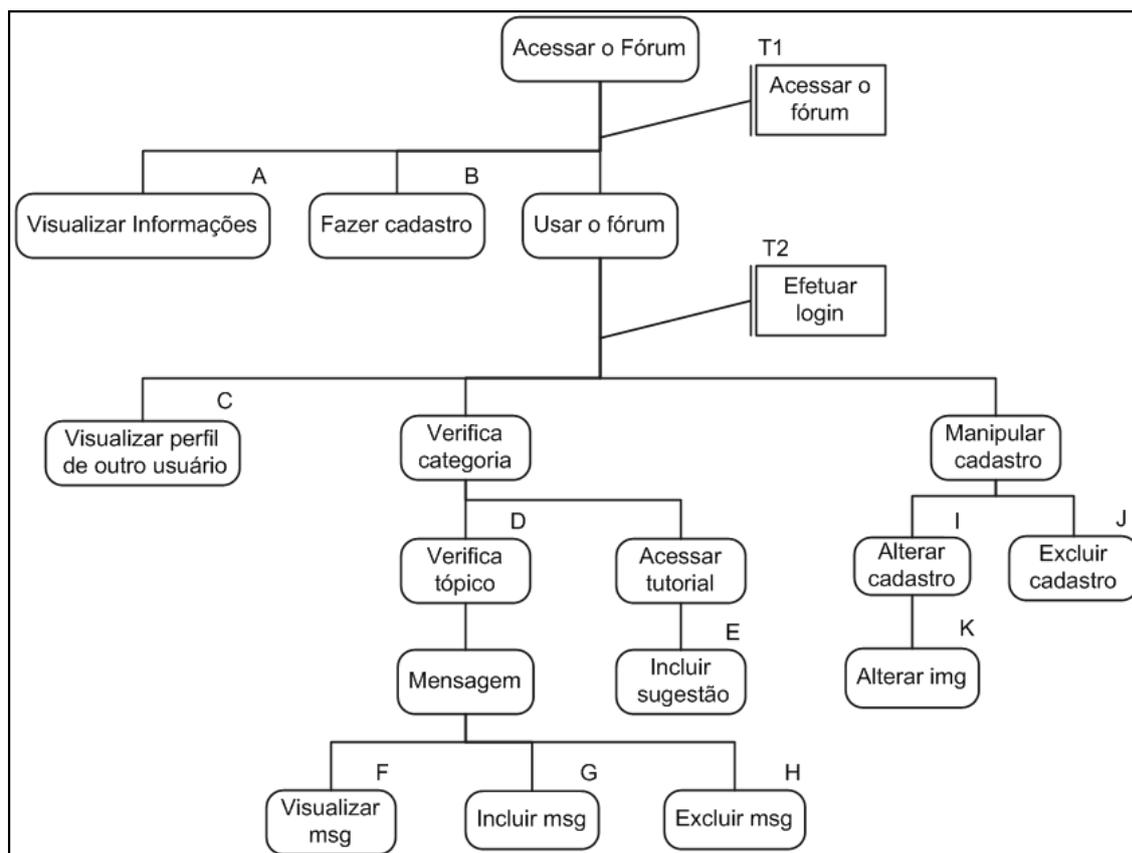


Figura 2 - Diagrama Hierárquico de Metas do Fórum Pascal.

4.4 Diagrama hierárquico de tarefas e especificação textual

A Figura 3 mostra a especificação hierárquica das tarefas da meta B – Fazer Cadastro, com suas respectivas tarefas *Sequenciais*: *Inserir dados*, *Fornecer login e Confirmar*.

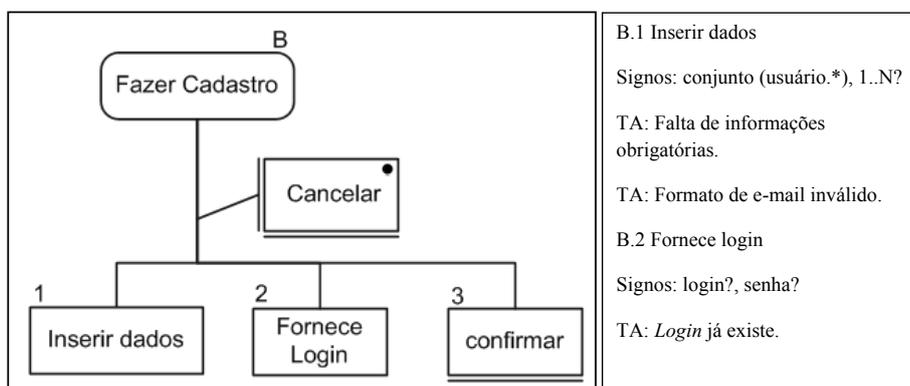


Figura 3 - Especificação gráfica e textual da meta B – Fazer Cadastro.

4.5 Diagrama de interação (DI)

O DI é baseado no DHM e explora a comunicação do usuário com o sistema na realização de uma meta. Expressa os problemas de interação que podem ocorrer na execução das metas e apresenta a ordem lógica de funcionamento do sistema, facilitando o desenvolvimento, principalmente em equipe, por orientar o andamento das atividades dentro do ambiente.

A Figura 4 mostra o DI e sua respectiva

interface no sistema para a meta “Alterar Imagem”, contendo as falas do Usuário na cena *Alterar Imagem*: ‘U:[Confirmar]’ e a mensagem de Tratamento Apoiado (TA), identificada pelo *designer* sobre o tamanho da imagem a ser enviada.

Portanto, toda a interação do usuário com o sistema foi modelada, especificando as ‘falas’ do usuário e do *designer* e, para cada tarefa, as mensagens de prevenção e/ou tratamento foram elaboradas, como forma de facilitar a interação usuário-sistema.

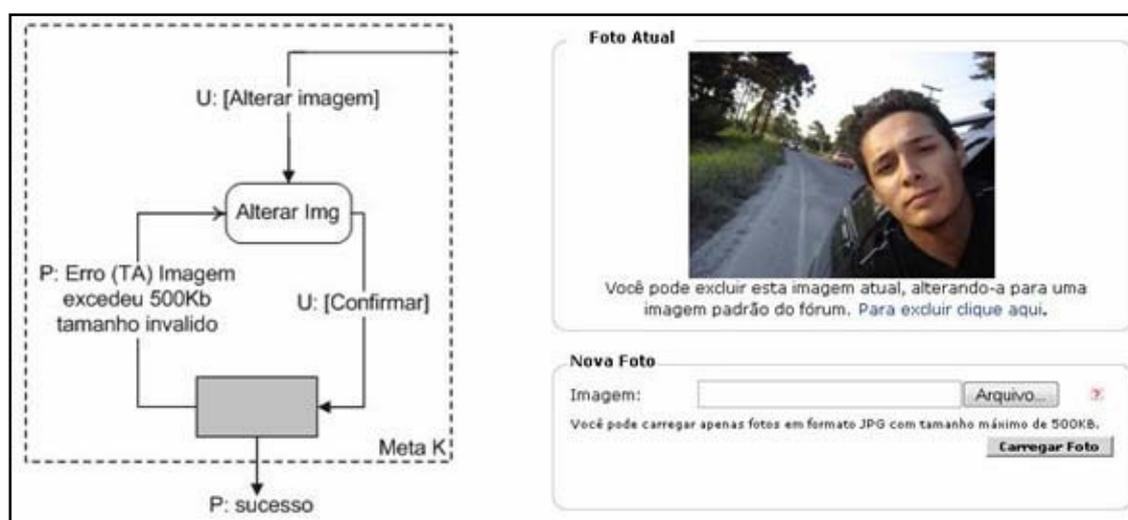


Figura 4 - DI da meta “Alterar Imagem” e sua respectiva interface no sistema.

4.6 Especificação do sistema de ajuda on-line

O sistema de ajuda on-line foi especificado utilizando os seis modelos propostos em (SILVEIRA, 2002). Dentre os modelos especifica-

dos, o Quadro 2 apresenta um exemplo do signo *Data_Nascimento* com a sua descrição no Modelo de Domínio, contendo a resposta para as questões: “*O que é isso?*”, por meio da Descrição, e “*Para que serve isto?*”, por meio da Utilidade do signo.

```

1. MODELO DE DOMÍNIO
Domínio (qual o domínio da aplicação?)
  Ambiente de aprendizado da linguagem Pascal.
Descrição (qual a natureza do trabalho?)
  Permite a participação de visitantes, usuários cadastrados e
  administradores para a visualização e/ou postagem de tópicos relacionados
  com o ensino da programação em linguagem Pascal.
SIGNO DE DOMÍNIO
{
  NOME (Data_Nascimento)
  DESCRIÇÃO (Armazena a data de nascimento do usuário)
  UTILIDADE (Serve para o cálculo da idade do usuário)
}

```

Quadro 2 - Exemplo da especificação do Modelo de Domínio para o sistema de ajuda on-line.

Toda a especificação do sistema de ajuda on-line do ambiente contribuiu para a implementação da ajuda do sistema, que foi desenvolvida utilizando a linguagem *Java Script*, apresentando de maneira dinâmica a ajuda referente a cada signo da interface, à medida que o usuário movimenta o cursor do mouse sobre o mesmo, como um *hint*, conforme a Figura 5.

Após a fase de modelagem do ambiente, algumas considerações foram feitas para o desenvolvimento do Fórum Pascal, apresentadas na seção a seguir.

4.7 Considerações quanto à implementação

No processo de implementação em PHP/POSTGRESQL (CONVERSE, 2003), algumas metodologias foram aplicadas ao desenvolvimento visando estabelecer maior segurança e mobilidade na utilização do Fórum Pascal, entre elas:

- i. uso de *cookies*, que são informações armazenadas em forma de variáveis que permitem ao ambiente manter a comunicação com

Dados pessoais

Nome: ?

Sobrenome: ?

Data de Nascimento: 5 ▾ fevereiro ▾ 1989 ▾ ?

Sexo: ?

Cidade: ?

Estado: ?

Data de Nascimento
 Selecione o dia, mês e ano de seu nascimento.
 Exemplo: 01/janeiro/1991

Figura 5 - Interface do cadastro com o sistema de ajuda on-line para cada signo.

os navegadores, armazenando-as de forma acessível durante todo o processo de navegação. No Fórum Pascal, os *cookies* ajudam a manter a segurança sobre parte do ambiente que só pode ser acessado por usuários cadastrados e logados;

- ii. uso de identificadores alfanuméricos, sendo que estes começaram a ser implantados na *Internet* no ano de 2001, tendo como objetivo assegurar que o uso de funções dos *websites* sejam usadas por usuários reais, e não por softwares mal intencionados. Isso inibe o cadastro de usuários fictícios e a criação de tópicos de propaganda. O uso dos identificadores é feito pela confirmação de caracteres aleatórios no ato de conclusão de tarefas, como mostra a Figura 6;
- iii. modularização de componentes, que se constitui como uma alternativa encontrada para facilitar as restrições de segurança, assim como favorecer a manutenção do *designer* por meio da utilização de *templates*. Com esse processo, todos os *scripts* (instruções formais escritas para um interpretador específico) de segurança, assim como os estilos utilizados por todo o ambiente, encontram-se na página central e, dependendo do parâmetro passado à mesma, é possível saber qual módulo está sendo requisitado para fazer a importação para a página, a fim de ser exibido.



Figura 6 - Confirmação do identificador alfanumérico para a troca de senha no Fórum Pascal.

5 Conclusões

O desenvolvimento do Fórum Pascal, no contexto da ES da área de IHC, teve por objetivo principal disponibilizar um ambiente para facilitar a interação de usuários para o aprendizado de Programação de Computadores.

Observou-se, por meio da análise feita em fóruns desenvolvidos por meio de CMSs, que os fóruns gerados apresentam uma padronização pelo uso de *templates*, que impedem a modificação da estrutura de funcionamento do ambiente, como a ordem e disposição dos módulos no *design*. Observou-se também a impossibilidade de o *designer* tratar as mensagens de erro geradas pelo sistema e a ausência de um sistema de ajuda on-line que apoie os usuários no momento em que apresentar alguma dúvida.

O método de avaliação de interfaces MAC foi aplicado em ambientes desenvolvidos com CMSs, sendo possível verificar que a interpretação do usuário é momentaneamente interrompida em algumas situações, como é o caso da associação das expressões “Cadê?” e “Ué o que houve?”, respectivamente relacionadas a problemas de navegação e de atribuição de significado.

Isso pôde ser observado pela gravação da interação nos testes aplicados, por exemplo, quando os usuários efetuavam uma postagem, questionavam-se (mesmo que mentalmente) “Cadê?” e, na sequência, “Ué o que houve?”, pois a visualização da postagem não fica evidente, assim como a mensagem de erro não é condizente à interação ocorrida.

Portanto, neste trabalho, considera-se o uso da linguagem MoLIC apropriada para o desenvolvimento em questão, pois permitiu aos *designers* realizarem a modelagem da interação usuário-sistema, por meio da especificação das cenas de conversação, bem como o tratamento das mensagens de erro e a elaboração do sistema de ajuda on-line em paralelo ao *design*.

Como trabalhos futuros, pretende-se aplicar o MAC com diferentes perfis de usuários no Fórum Pascal como forma de verificar possíveis falhas de interação para serem tratadas em posterior *redesign*.

Também uma propriedade de interface bastante importante a ser considerada é quanto à acessibilidade, não tratada por nenhum fórum analisado.

Agradecimentos

Os autores agradecem à Bolsa Institucional BIC/UNICENTRO para a realização deste trabalho.

Referências

BARBOSA, S. D. J.; PAULA, M. G. **Designing and evaluating interaction as conversation: a modeling language based on semiotic engineering**. In: Design, verification, and specification of interactive systems - DSV'IS 2003, Funchal, Ilha da Madeira. Lecture Notes on Computer Science - Proceedings of DSV-IS 2003.

CONVERSE, T; PARK, J. **PHP: a biblia**. Editora Campos, 2003, Vol.1.

DE SOUZA, C.S. **The semiotic engineering of human-computer interaction**. Cambridge, The MIT Press, 2005.

DE SOUZA, C.S.; PRATES, R.O.; BARBOSA, S.D.J. **A method for evaluating software communicability**. IHC'99 Proceedings. Campinas, SP, Brazil. Oct., 1999. (CD-ROM).

FÓRUM PASCAL. Disponível em: <<http://www.forumpascal.com.br/index.php>>. Acesso em: abr. 2007.

HYPERIONICS TECHNOLOGY **Hypercam**. Versão 1.17.05. Disponível em: <<http://www.hyperionics.com/>>. Acesso em fev. 2007.

JOOMLA. Disponível em: <<http://www.joomla.org>>. Acesso em: nov. 2006.

NORMAN, D. Cognitive engineering. In D. Norman; S. Draper. (Eds.) **User centered system design**. Hillsdale, NJ. Lawrence Erlbaum. pp.31-61.1986.

PEIRCE; C.S. **Collected papers of Charles Sanders Peirce**, v. 1-8, C. Hartshome and P. Weiss. Cambridge, MA: Harvard University Press. 1931-1958.

PHPBB. Disponível em: <<http://www.phpbbbrasil.com.br/phpBB/portal.php>>. Acesso em: nov. 2006.

PRATES, R. O.; SOUZA, C. S.; B., S. D. J. **A method for evaluating the communicability of user interfaces**. Interactions, New York, v. 7, n. 1, p. 31-38, 2000.

ROCHA, H. V.; BARANAUSKAS, M. C. C. **Design e avaliação de interfaces humano-computador**. São Paulo: NIED/UNICAMP, 2003. 224p.

SILVEIRA, M. **Metacomunicação designer-usuário na interação humano-computador**. PhD Thesis. Departamento de Informática. PUC-Rio. Rio de Janeiro, 2002.