

3

WNH – Web Navigation Helper

3.1

O gravador de macros

O WNH teve dois momentos diferentes de desenvolvimento. Sua primeira versão será detalhada na próxima seção. Entretanto, antes de descrevermos o assistente, em suas duas versões, e o editor, é necessário comentarmos sobre a tecnologia na qual o WNH tem base. Ele foi criado a partir de uma ferramenta da IBM, chamada CoScripter (LESHED, HABER, *et al.*, 2008) (COSCRIPTER), uma extensão para o navegador Firefox que permite a criação de *scripts* de navegação na Web, funcionando como um gravador de macros. Um script é uma sequência de passos de interação em um navegador. As ideias principais são os recursos de automatização e compartilhamento de tarefas, pois, depois que um script de determinada tarefa é criado, ele pode ser executado várias vezes. Os scripts podem ser privados ou públicos, caso em que podem ser compartilhados por outros usuários. A Figura 3.1 mostra a tela inicial do gravador.

O CoScripter apresenta-se como uma barra lateral no navegador. Na parte superior da extensão, há a barra de ferramentas, com seus principais comandos. Nesta tela inicial, a barra apresenta apenas o botão “New”, para criar um novo script e o botão “Home”, presente em todas as outras telas e que serve para retornar o CoScripter ao seu estado inicial. Na região central, há uma breve descrição da ferramenta, e três listas com alguns scripts. As listas mostram os scripts acessados recentemente, os scripts favoritos e os scripts criados pelo usuário “logado”. Caso o script desejado não esteja em uma destas listas, o usuário pode acessar a página do CoScripter e procurá-lo. Na parte de baixo da barra lateral, há uma região chamada “Personal DB”. Trata-se do banco de dados pessoal do usuário, onde é possível armazenar variáveis a serem utilizadas nos scripts. Este é um dos principais recursos do gravador de macros, porque permite que os scripts sejam flexíveis o suficiente para serem utilizados tanto em situações distintas do momento da criação (por exemplo, fornecendo-se uma data, que varia

em cada uso), quanto por usuários diferentes (por exemplo, fornecendo-se dados pessoais, que varia entre usuários).

Observe ainda que, na Figura 3.1, há um pequeno ícone em forma de penas no canto inferior direito. Este ícone permite o acesso a algumas opções do CoScripter. A Figura 3.2 mostra, com mais detalhes, o menu suspenso que surge ao clicar no ícone. Entre os recursos disponíveis estão: acessar o site da extensão, exibir a barra lateral do CoScripter, e, na parte inferior do menu suspenso, o CoScripter exibe uma lista de scripts criados (quando existirem) para a página atualmente carregada no navegador. Na ilustração, a lista de scripts ainda não foi obtida (“Loading...”, na imagem).

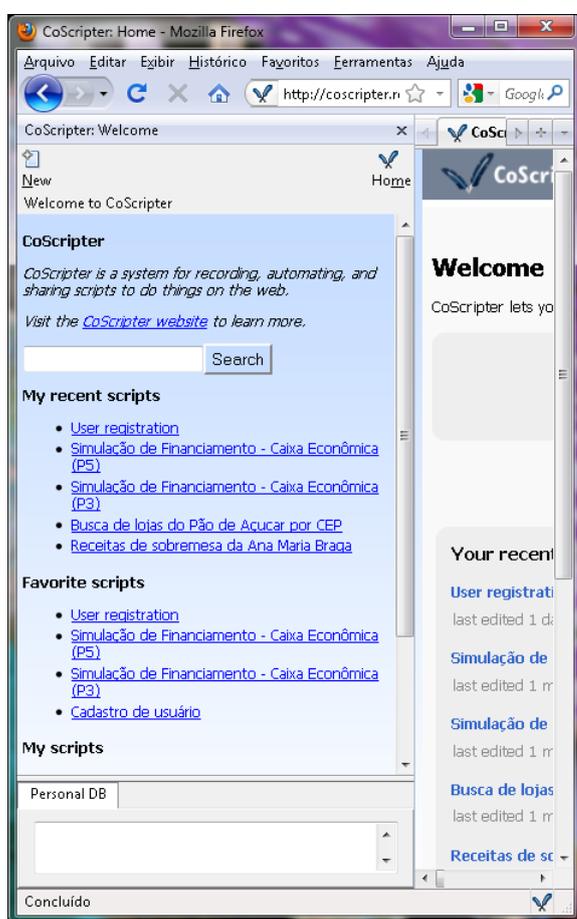


Figura 3.1 – Tela inicial do CoScripter

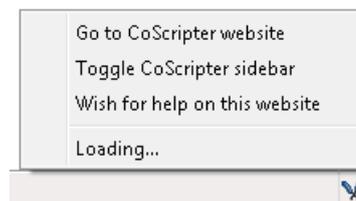


Figura 3.2 – Acesso à extensão e opções iniciais do CoScripter

Para criar um novo script, é necessário acessar a tela do editor de scripts do CoScripter, através do comando “New”, na tela inicial. O editor, em modo de gravação, é representado na Figura 3.3. Observe que a barra de ferramentas possui novos comandos, como “Run”, “Stop”, “Save”... O modo de gravação é representado pelo texto “Recording...”, em uma área reservada para mensagens do

CoScripter; pelo botão “Record” piscando e pelo cursor piscando na área do script.

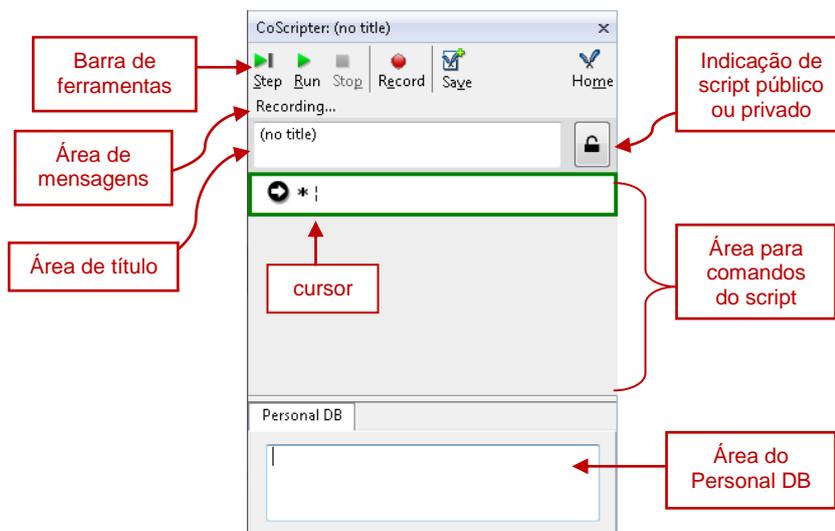


Figura 3.3 – Tela de edição de um script

A partir deste momento, toda a interação que ocorre no navegador é registrada, tanto ações nas páginas acessadas, quanto ações do próprio navegador, como os menus (Arquivo, Editar, Exibir...) e os botões para voltar, avançar e ir à página inicial. O ícone de cadeado abaixo do botão “Home” indica se o script vai ser público ou privado.

Para explicar melhor o funcionamento do CoScripter, considere a página fictícia de cadastro de usuários representada na Figura 3.4³. Trata-se de um formulário para preenchimento de dados pessoais. Após o usuário clicar em “Enviar”, é exibida uma página de sucesso.

Usando o CoScripter, vamos criar um script para efetuar o cadastro. Durante a gravação do script, toda a interação vai sendo registrada na área do script, conforme apresentado na Figura 3.5. Observe que a linguagem do CoScripter é bem simples, indicando três elementos: a ação a ser feita (“go to”, “enter”, “click”...), o componente da página que sofre a ação (“‘Nome’ textbox”, “‘Dia’ listbox”, “‘Enviar’ button”...) e o conteúdo a ser passado como entrada (“José”, “da Silva”, “12”...) ⁴. O Quadro 3.1 indica todos os passos⁵ criados para este script.

³ Esta página de cadastro de usuário será utilizada como exemplo por todo o capítulo, tanto na explicação sobre o CoScripter quanto nas explicações sobre o WNH e seu editor de diálogos.

⁴ Quando a ação não implica em entrada do usuário o comando só possui dois elementos: a ação e o elemento na página: “go to http://[alguma URL]”, “click the ‘Enviar’ button”.

⁵ Dentro deste texto, “passo”, “comando” e “instrução” serão usados como sinônimos e designam um item ou uma linha do script.

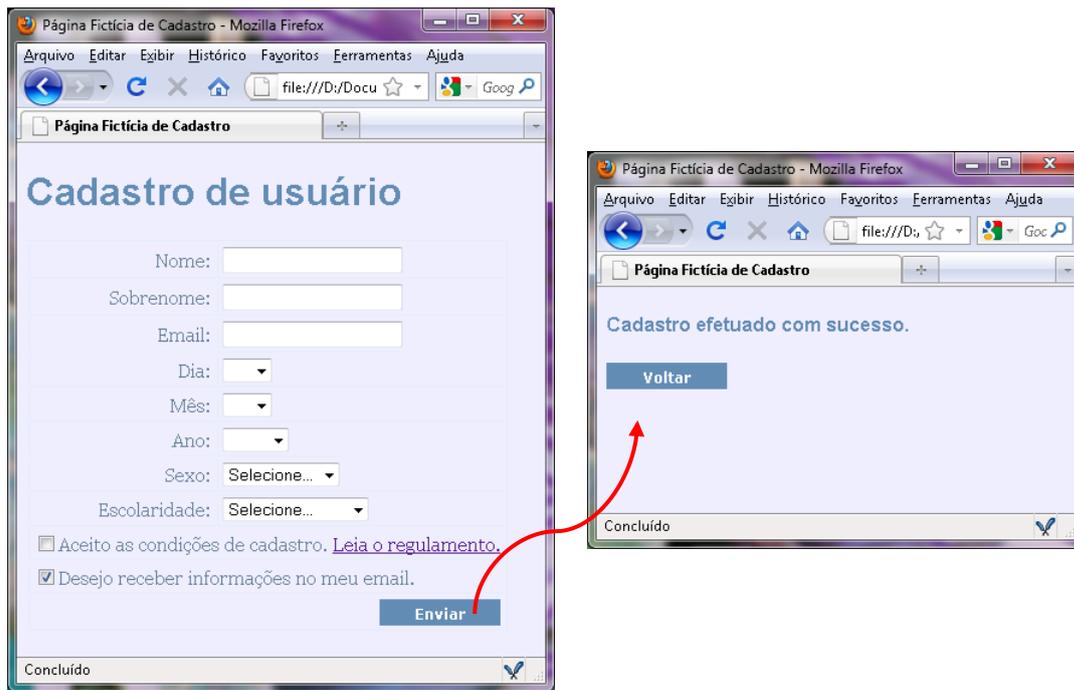


Figura 3.4 – Página fictícia de cadastro de usuário

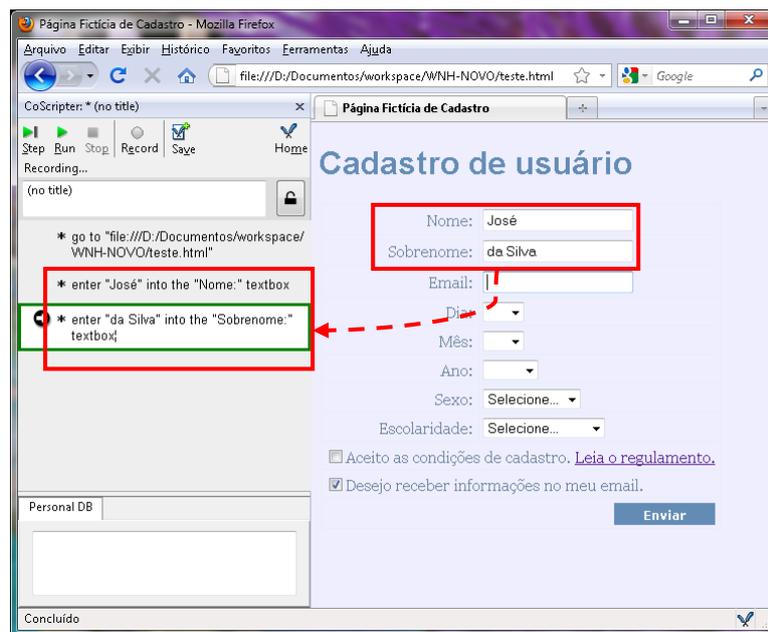


Figura 3.5 – Início da gravação do Script

1. go to "file:///D:/Documentos/workspace/WNH-NOVO/teste.html"
2. enter "José" into the "Nome:" textbox
3. enter "da Silva" into the "Sobrenome:" textbox
4. enter "josedasilva@email.com" into the "Email:" textbox
5. select "01" from the "Dia:" listbox
6. select "01" from the "Mês:" listbox
7. select "1970" from the "Ano:" listbox
8. select "Masculino" from the "Sexo:" listbox
9. select "Médio" from the "Escolaridade:" listbox
10. turn on the "Aceito as condições de cadastro." checkbox
11. turn off the "Desejo receber informações no meu email." checkbox
12. click the "Enviar" button

Quadro 3.1 – Script para criação do cadastro

Observe que todo o conteúdo de entrada do usuário, como nome, email, data de aniversário, está explícito no texto do script (Quadro 3.1). Isto significa que sempre que ele for executado, os valores indicados nos passos serão atribuídos aos elementos de entrada da página, ou seja, este script só fará sentido para o “José da Silva”, já que são os seus dados que estão presentes. Para que o script possa ser usado com outros valores, durante a sua criação, devem ser determinadas todas as variáveis desejadas, através do Personal DB. Por exemplo, o usuário pode criar uma entrada no Personal DB com o valor “nome = José”. Fazendo isso, ao criar o script, o comando associado ao preenchimento do nome (comando 2 no Quadro 3.1) deverá ser alterado para “enter your “nome” into the “Nome:” textbox”. Note que foi adicionada a palavra *your* ao comando, indicando ao CoScripter que o texto que se segue refere-se a uma variável no Personal DB, no caso, a variável “nome”. Desta forma, cada novo usuário que for utilizar este script, poderá informar o seu próprio valor para a variável “nome” no Personal DB.

A Tabela 3.1 mostra como ficam as instruções deste script depois de adicionadas todas as variáveis. Não são mostrados na tabela os comandos 1, 10, 11 e 12, já que não precisam de entrada do usuário e permanecem iguais nos dois casos. É importante destacar que é o criador do script quem decide quais comandos serão criados com variáveis. Não é obrigatório que todos aqueles que solicitem entrada do usuário sejam “relativizados” por meio das variáveis. Esta decisão é feita baseada em quais dados o criador quer deixar “em aberto”, para que o executor do script informe por meio dos valores no Personal DB. Por exemplo, o criador do script pode construí-lo para que apenas pessoas do sexo masculino o utilizem, ou apenas pessoas que possuam nível “Médio” de escolaridade. Em ambos os casos, não haveria variáveis no Personal DB para estas informações, que estariam fixadas no texto do script, conforme os comandos 8 e 9 do Quadro 3.1.

O WNH foi construído essencialmente a partir deste mecanismo de variáveis. Portanto, é importante que esse conceito fique claro para o leitor, a fim de que seja mais bem compreendido o funcionamento do assistente.

Tabela 3.1 – Comparação entre os passos com e sem variáveis

Passo	Passo sem variável	Passo com variável	Variável
2	enter "José" into the "Nome:" textbox	enter your "nome" into the "Nome:" textbox	nome
3	enter "da Silva" into the "Sobrenome:" textbox	enter your "sobrenome" into the "Sobrenome:" textbox	sobrenome
4	enter "josedasilva@email.com" into the "Email:" textbox	enter your "email" into the "Email:" textbox	email
5	select "01" from the "Dia:" listbox	select your "dia" from the "Dia:" listbox	dia
6	select "01" from the "Mês:" listbox	select your "mes" from the "Mês:" listbox	mes
7	select "1970" from the "Ano:" listbox	select your "ano" from the "Ano:" listbox	ano
8	select "Masculino" from the "Sexo:" listbox	select your "sexo" from the "Sexo:" listbox	sexo
9	select "Médio" from the "Escolaridade:" listbox	select your "escolaridade" from the "Escolaridade:" listbox	escolaridade

3.2

WNH – primeira versão

Os trabalhos com o WNH iniciaram-se ainda em 2008, em uma pesquisa de mestrado anterior a esta. A investigação empreendida no período, os experimentos realizados e uma ferramenta inicial desenvolvida foram abordados na dissertação relacionada (INTRATOR, 2009) e em publicações associadas (INTRATOR e DE SOUZA, 2008) (INTRATOR e DE SOUZA, 2009).

Nesta fase inicial, o WNH era considerado uma ferramenta colaborativa, pois a força-tarefa por trás do seu funcionamento estaria por conta de uma comunidade de voluntários encarregados de criar os scripts e os diálogos de mediação das tarefas a serem executadas por usuários com necessidades especiais. O esforço de desenvolvimento e os estudos exploratórios levaram em consideração dois perfis de usuários com necessidades especiais: os deficientes visuais e os analfabetos funcionais. A premissa básica na dissertação era: “WNH is a promising tool for helping blind and functionally illiterate users in navigating the web.” (INTRATOR, 2009, p. 14). Foram então elaboradas três versões do WNH, a saber:

- WNH-see, a web navigation and resource-finding tool for blind end-users;
- WNH-read, a web navigation and resource-finding tool for end-users with reading and understanding difficulties;
- WNH-support, a web tool for supporting a community of volunteer helpers in creating and maintaining the processes that will be run and executed by the end-users; (INTRATOR, 2009, p. 14)

As interfaces iniciais do WNH-see e do WNH-read são exatamente as mesmas. As diferenças surgem apenas durante a execução da tarefa. A Figura 3.6 mostra como o usuário inicia o uso do assistente. A lista na barra lateral mostra todas as tarefas disponíveis pelo WNH para a página carregada no navegador, no caso, a página do DETRAN-RJ.

Ao escolher uma tarefa, o script é executado e, quando uma entrada do usuário é necessária, um diálogo é exibido, semelhante aos que aparecem na Figura 3.7. A principal diferença, contudo, entre o WNH-read e o WNH-see é que neste último, para cada passo do script, é exibido um diálogo informando quanto da tarefa já foi realizado e quanto ainda falta (Figura 3.8).



Figura 3.6 – Primeira versão do WNH (INTRATOR, 2009, p. 35)

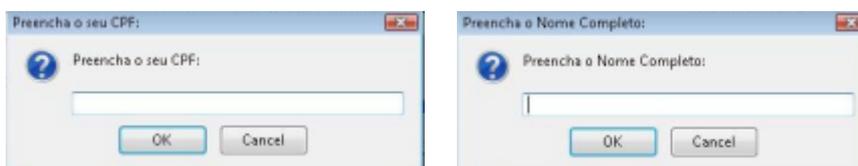


Figura 3.7 – Diálogos de mediação do WNH original (adaptado de (INTRATOR, 2009, p. 49,53))

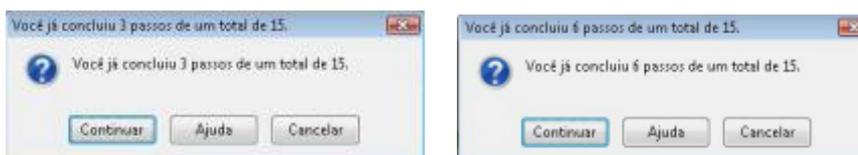


Figura 3.8 – Navegação no WNH-see (adaptado de (INTRATOR, 2009, p. 48))

Sobre o WNH-support, o principal objetivo era auxiliar os voluntários na construção dos scripts (através de uma interface integrada ao CoScripter) e na sua

manutenção (através de códigos-robôs que testam se os scripts ainda funcionam). É importante ressaltar, entretanto, que esta versão não contemplava a elaboração em si dos diálogos. Observe que os textos dos diálogos são bastante concisos. Eles são elaborados de forma quase automática pela ferramenta. A única autonomia dada aos voluntários é a informação que acompanha a instrução “Preencha”. Ou seja, ainda não existia um editor de diálogos disponível, um dos objetivos da fase atual de pesquisa com o WNH. Outras informações relevantes sobre esta versão serão apresentadas na Seção 4.1, que trata sobre os experimentos realizados durante esta primeira fase de pesquisa com o WNH.

As próximas seções descrevem o WNH e seus componentes, já em sua versão desenvolvida para a fase atual da pesquisa.

3.3 Diálogos de mediação do WNH

O sistema WNH, em sua fase atual, é constituído de duas partes: um editor de diálogos, através do qual os usuários criadores elaboram e configuram os diálogos, e o próprio assistente. O sistema é, em si, um “assistente de navegação na Web”, constituído por diálogos de mediação que acompanham um usuário com necessidades especiais durante sua navegação em um site na Web, auxiliando-o na execução de uma determinada tarefa, oferecida pela página em questão. As principais características do WNH que permitem esse auxílio são:

- O usuário interage com os diálogos do WNH, ao invés de diretamente na interface original do site.
- A navegação é feita em etapas, pois as informações são solicitadas individualmente ou em pequenos blocos, conforme for construído o diálogo.
- Há diálogos para todos os comandos do script que exigem uma entrada do usuário.
- Nos diálogos, é livre a adição de textos explicativos para esclarecer o preenchimento das informações ou esclarecer outros usos dos diálogos e do site.
- Podem ser criados diálogos de boas-vindas e de encerramento para complementar o auxílio.

- Há uma seção de dúvidas em cada diálogo, onde o usuário final pode ter acesso a perguntas e respostas sobre o diálogo em questão. Estas dúvidas devem ser antecipadas pelos criadores dos diálogos e adicionadas à seção durante sua criação.

Para entender melhor estas características e funcionalidades, demonstraremos a seguir os diálogos criados para a página de cadastro de usuário (Figura 3.4, p. 30). Para iniciar os diálogos do WNH, o usuário precisa acessar a lista de tarefas disponíveis. Este acesso é feito pela opção “Acessar tarefas mediadas”, exibida a partir do ícone do WNH no canto inferior do navegador (Figura 3.9)⁶. Ao escolher esta opção, uma barra lateral se abre, listando as tarefas mediadas disponíveis. A Figura 3.10 apresenta a tela inicial com a lista de tarefas. O usuário escolhe um item da lista, selecionando a linha correspondente e clicando no botão “Executar tarefa” ou ainda clicando duas vezes na linha selecionada.

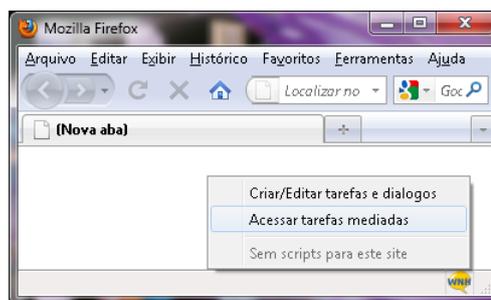


Figura 3.9 – Opções de acesso do WNH

Supondo que seja escolhida a tarefa de cadastro de usuário, após iniciar a execução, os diálogos de mediação são exibidos, um a um. A barra lateral é reduzida a dois botões, um para reiniciar a tarefa em execução e outro para retornar à lista de tarefas (Figura 3.11). Ao mesmo tempo em que a barra lateral é alterada, é exibido o primeiro diálogo da tarefa, conforme a Figura 3.12. Neste caso, o primeiro diálogo é de boas-vindas. Ele possui uma descrição breve da tarefa e dá instruções gerais sobre como interagir com a janela do diálogo.

⁶ A primeira opção no menu da Figura 3.9 leva à interface do editor de diálogos do WNH, detalhado mais à frente.

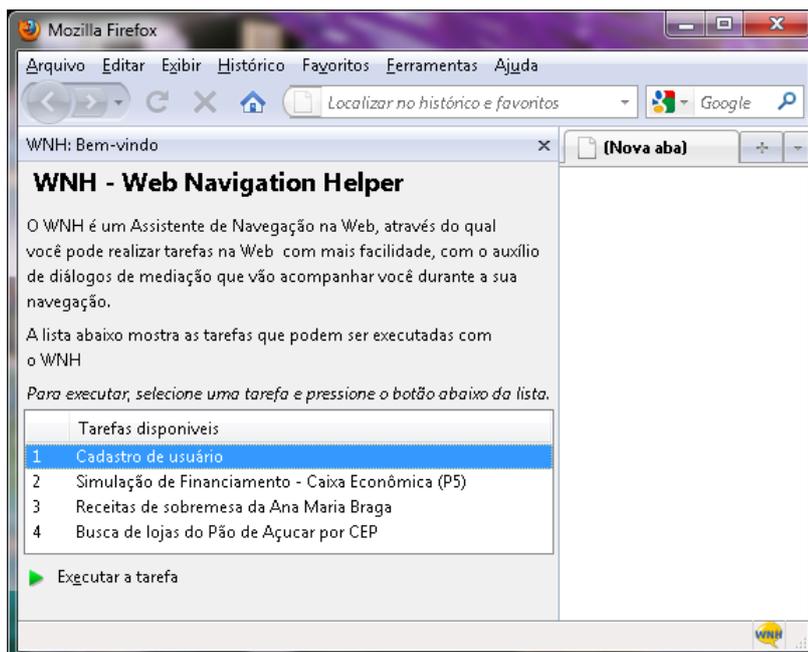


Figura 3.10 – Lista de tarefas mediadas pelo WNH

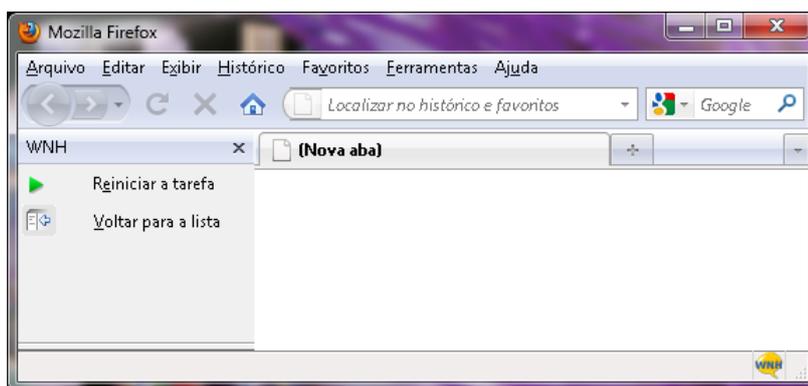


Figura 3.11 – Ações possíveis no WNH

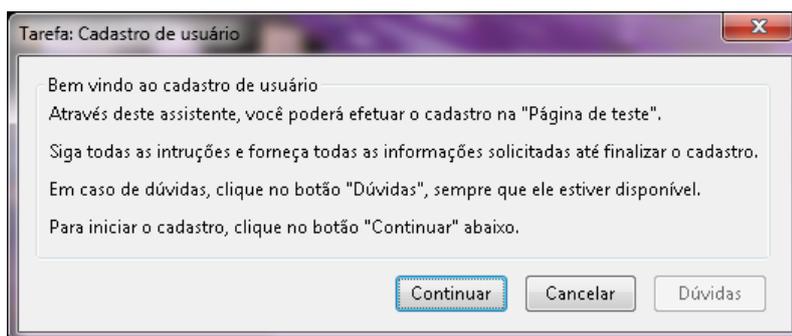


Figura 3.12 – Cadastro de usuário: diálogo de boas vindas

Conforme observado na imagem, o diálogo possui os seguintes elementos: 1) título da janela com o nome da tarefa escolhida na lista; 2) um título para o diálogo, definido durante a criação, sintetizando a sua função; 3) um texto descritivo e/ou explicativo, também adicionado pelo criador do diálogo; 4) um botão “Continuar”, para prosseguir a tarefa; 5) um botão “Cancelar”, que cessa a

execução da tarefa; 6) um botão “Dúvidas”, através do qual o usuário tem acesso à seção de dúvidas criadas para o diálogo corrente.

Apenas após clicar em “Continuar”, o script é ativado e entra em execução. Todos os passos automáticos são realizados até que se alcance um comando que possua variáveis. Os campos associados aos diálogos são destacados, pelo CoScripter, na interface do site, indicando quais informações na página estão sendo preenchidas naquele momento (observe a marcação vermelha na Figura 3.13). A janela do diálogo é, por padrão, posicionada no canto superior esquerdo da tela, assim, o usuário pode acompanhar o que está acontecendo na página. Caso deseje, ele pode ainda deslocar (arrastar) a janela para obter melhor visualização.

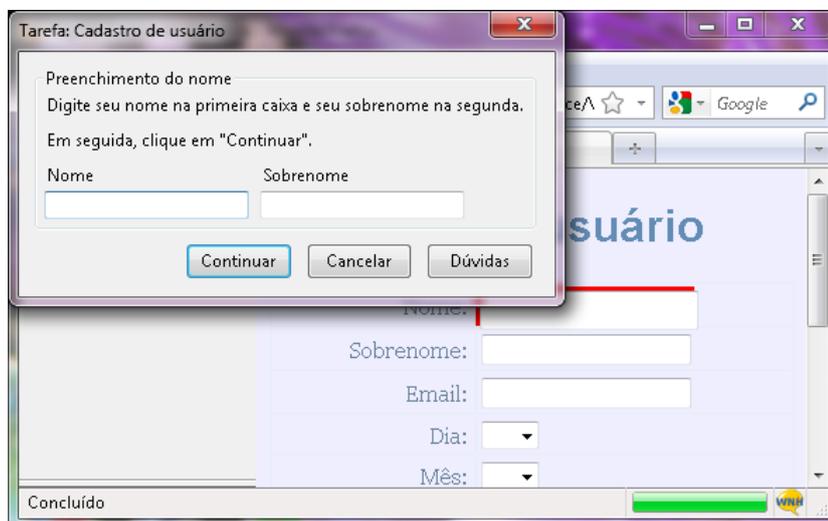


Figura 3.13 – Cadastro de usuário: nome e sobrenome

O diálogo apresentado na Figura 3.13 solicita duas entradas ao usuário: o nome e o sobrenome. A forma como o WNH, tecnicamente, trata as informações solicitadas se manteve em relação à primeira versão do assistente e pode ser entendida com a explicação dada por Intrator (2009), ao descrever este detalhe da ferramenta:

[...] opposed to the way CoScripter requests personal information in order to run the scripts (it does that by presupposing the user has already filled in a variable in his main Personal Database, so it can be retrieved when running the script), both WNH-see and WNH-read do not store variables values in any database. All information is requested on-the-fly, through a pop-up window (and discarded after the script completion). [...] Asking for the needed information during the task execution presented itself as a more natural and familiar way than doing that through the Personal Database. (INTRATOR, 2009, p. 35,36)

Este detalhe técnico é uma das semelhanças entre a versão atual do WNH e a versão pioneira. Entretanto, o próprio diálogo da Figura 3.13 traz uma das inovações da nova versão. Agora, o criador de diálogos pode agrupar entradas de usuário em um mesmo diálogo, no caso, nome e sobrenome. Através deste recurso, ele estabelece quantas e quais informações serão preenchidas em um mesmo diálogo. Ele pode juntar informações semelhantes (nome e sobrenome, campos para data de nascimento...) ou solicitá-las individualmente nos diálogos, como acontecia na versão antiga (Figura 3.7, p. 33). As telas da Figura 3.14 a Figura 3.18 mostram todos os outros diálogos criados para o cadastro de usuário, na sequência em que são apresentados. Os detalhes relevantes serão comentados a seguir.

Um dos recursos disponibilizados para os criadores é a opção de alterar o elemento de interação original do site. Observe que na página de cadastro (Figura 3.4, p. 30), a data de nascimento é preenchida escolhendo-se os três números que compõem a data, em três listas diferentes. No diálogo (Figura 3.15), por uma decisão do criador, as três listas passaram a ser três caixas de texto, nas quais o usuário final deve informar os valores correspondentes a dia, mês e ano. Algo semelhante ocorre com o diálogo do sexo (Figura 3.16). Antes, a informação era passada também pela escolha de uma opção na lista e, agora, mudou para o componente de interação *radiobutton*. Uma justificativa do autor (estes diálogos foram criados apenas para testes e demonstrações) seria que a interação por *radiobutton* é mais direta, já que exige apenas uma ação do usuário (um clique na opção), ao invés de mais de uma no formato original (clique no indicador da lista, procurar a opção e clicar nela). Outra razão é que como são apenas duas opções possíveis para o sexo, uma lista talvez não seja o elemento mais adequado.

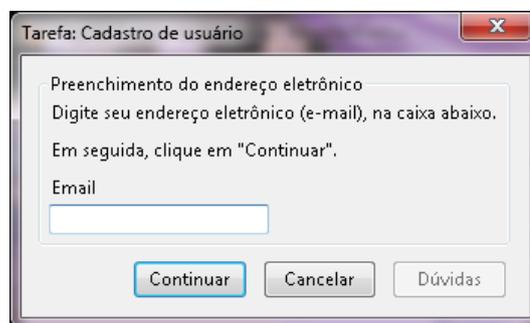


Figura 3.14 – Cadastro de usuário: e-mail

Outro recurso disponível aos usuários é a seção de dúvidas dos diálogos. Observe que todos eles apresentam um botão “Dúvidas” (na parte inferior das

Figura 3.12 a Figura 3.18, p. 36 a p. 39), através do qual o usuário tem acesso às perguntas e respostas criadas pelo autor dos diálogos.

Figura 3.15 – Cadastro de usuário: data de nascimento

Figura 3.16 – Cadastro de usuário: sexo

Figura 3.17 – Cadastro de usuário: escolaridade

Figura 3.18 – Cadastro de usuário: diálogo de encerramento

Ao experimentar alguma dificuldade durante a interação com o diálogo (ou apenas por curiosidade), o usuário clica o botão e observa as dúvidas. Cabe ao autor antecipá-las no momento da criação do diálogo, ou seja, é a sensibilidade de criar dúvidas adequadas ao diálogo que leva a uma maior qualidade deste recurso de ajuda. Note que, entre os diálogos, dois (Figura 3.13, p. 37 e Figura 3.17, acima) estão com o botão “Dúvidas” habilitado, indicando que foram criadas perguntas e respostas para eles. As listas de dúvidas estão, respectivamente, indicadas na Figura 3.19 e na Figura 3.20. Em ambos os casos, as perguntas e respostas criadas procuram esclarecer dúvidas técnicas acerca do uso dos elementos de interação do diálogo

(a caixa de texto e a lista). São dadas instruções de como utilizar o mouse para esses elementos e como identificá-los na área do diálogo.

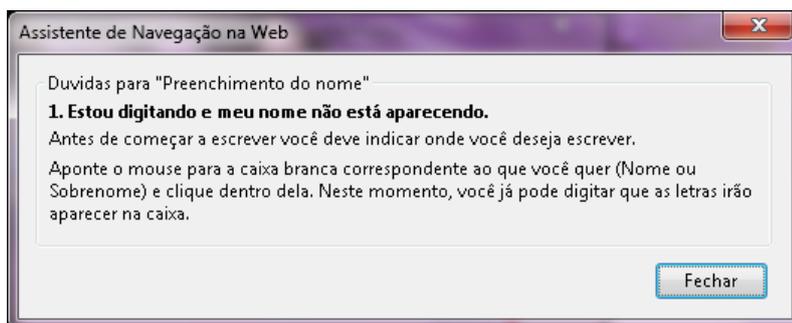


Figura 3.19 – Tela de dúvidas – diálogo Nome e Sobrenome

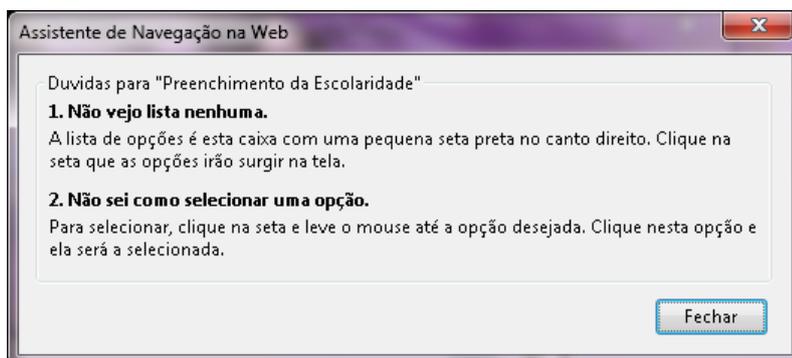


Figura 3.20 – Tela de dúvidas – diálogo Escolaridade

Além dos recursos acima descritos, podemos destacar ainda outras funcionalidades do WNH. Durante a navegação, caso o usuário deixe uma caixa de texto sem preencher, ou não escolha uma das opções de uma lista, uma mensagem de alerta é exibida, indicando o problema. Outra característica existente é que, durante a criação do diálogo, o autor pode indicar a formatação do campo de entrada, como data, CPF, apenas números, moeda, entre outros. Esta também é uma forma de evitar erros durante a navegação e principalmente durante a passagem dos valores informados no diálogo ao CoScripter, para sua devida atribuição aos campos na interface original.

Entretanto, por enquanto, estas são as duas únicas formas de prevenção e tratamento de erros disponíveis na ferramenta. Caso ocorra algum outro problema durante a navegação, como um valor passado erroneamente, o script para, e o usuário pode tentar novamente, por meio do botão “Reiniciar a tarefa” (Figura 3.11, p. 36). Isto acontece principalmente porque as validações que ocorrem na página original, tanto por JavaScript, quanto pelo próprio servidor, não são detectadas pelos diálogos do WNH.

3.4 Editor de diálogos do WNH

A primeira fase da pesquisa com o WNH considerou apenas os diálogos de mediação em uma versão simplificada, não tratando do seu processo de criação. Boa parte do esforço de desenvolvimento da fase atual da pesquisa com o WNH envolveu o design de uma ferramenta que permitisse a criação dos diálogos de mediação. Trata-se do editor de diálogos do WNH, apresentado nesta seção.

O comando que dá acesso ao editor foi incorporado à interface do CoScripter. Isto ocorreu porque o primeiro passo para se criar os diálogos é a criação de um script da tarefa para a qual eles farão a mediação. Assim, ao acessar a primeira opção no menu suspenso da Figura 3.9 (p. 35), a tela que surge é exatamente a mesma que a original do CoScripter (Figura 3.1, p. 28), com o diferencial de estar traduzida para o português (Figura 3.21). A partir desta tela, o usuário pode criar um novo script (botão “Novo”, na barra de ferramentas) ou acessar um dos scripts listados.

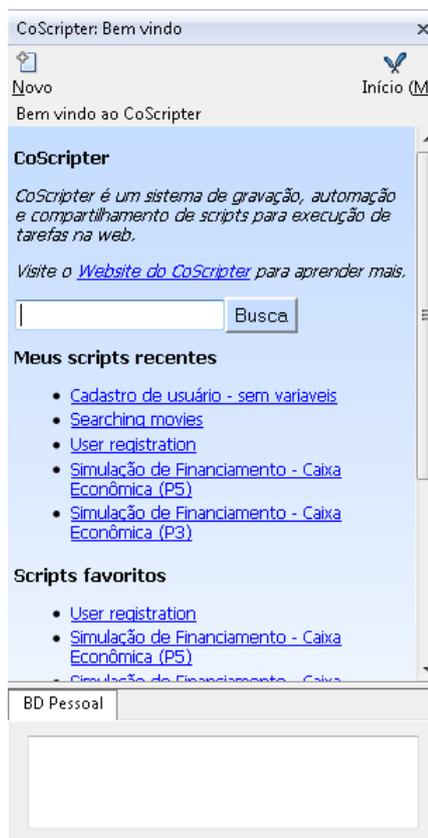


Figura 3.21 – Nova interface do CoScripter integrado ao WNH

Se o usuário decide criar um novo script, o editor de scripts é exibido de forma semelhante à já apresentada na Figura 3.3 (p. 29). As diferenças são que a interface está em português e, além da barra de ferramentas original do CoScripter, há uma nova barra, agora com comandos relacionados ao WNH. A Figura 3.22 mostra a tela de edição de scripts: à esquerda, com um novo script sendo criado e à direita, já com o script de cadastro de usuário gravado.

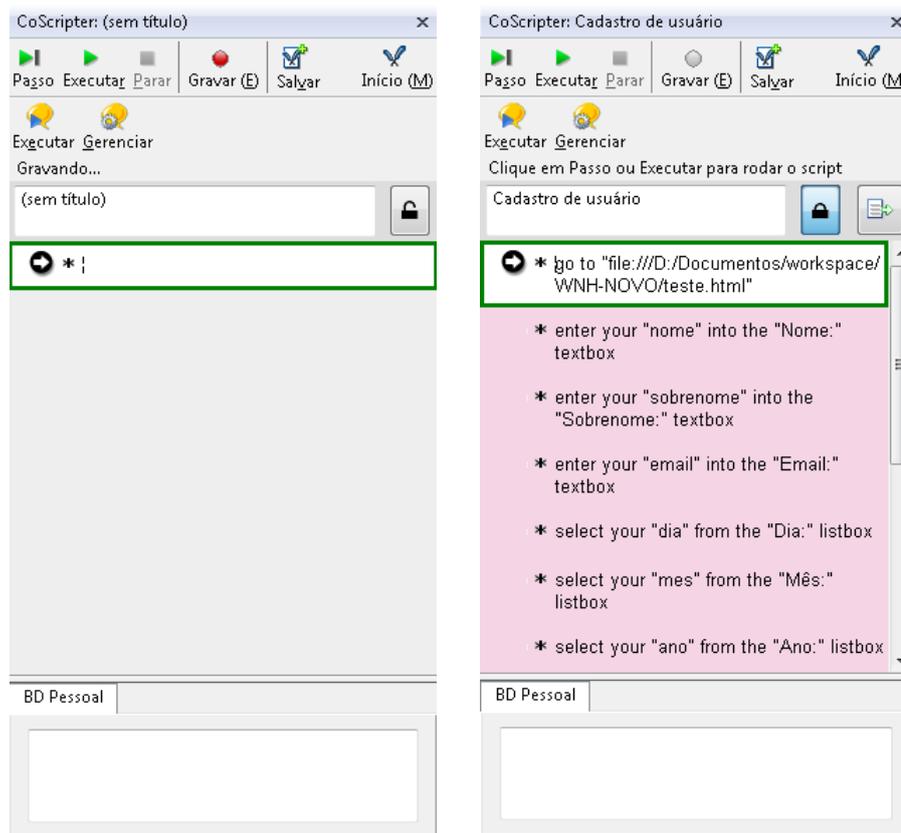


Figura 3.22 – Editor: (a) gravação de novo script e (b) script criado

Há apenas dois botões relacionados ao WNH: “Executar” e “Gerenciar”. O primeiro permite que o criador de diálogos execute-os, acompanhando todo o processo envolvido em relação ao script e à página: sequência de apresentação dos diálogos, instruções correspondentes no script e campos preenchidos na interface original. O segundo botão dá acesso ao real editor de diálogos. A partir dele, os diálogos para o script que está aberto no editor de scripts poderão ser criados. Se o usuário acionar a opção “Executar” do WNH sem que haja diálogos criados para o script aberto, será emitida uma mensagem de alerta. Caso os diálogos já tenham sido criados e o usuário altere o conteúdo do script, ao tentar acessar o editor de diálogos novamente, será exibido um alerta sobre as alterações feitas e uma pergunta se o usuário deseja reiniciar o processo de criação dos diálogos,

perdendo o conteúdo criado anteriormente. Além dos botões do WNH, todas as funções originais do CoScripter foram mantidas: criar e editar o script, criar variáveis, utilizar o banco de dados pessoal, executar o script, executar passo a passo, parar a execução etc.

Considerando um fluxo normal de criação, imagine que o usuário acabou de criar o script de cadastro de usuário (Figura 3.22 (b)), já com as variáveis corretamente colocadas nas instruções, seguindo a linguagem do CoScripter, com a palavra-chave *your*. O usuário-autor está pronto para iniciar a criação dos diálogos e então aciona o botão “Gerenciar” do WNH. A tela inicial do editor é exibida, conforme a Figura 3.23. As imagens seguintes mostram os detalhes da interface em uma melhor resolução. Na Figura 3.24, vê-se a parte superior do editor. Nota-se o nome do script (Cadastro de usuário), exibido na barra de títulos da janela do editor. Ainda nessa parte superior, há o botão “Visualizar”, no canto direito. A qualquer momento durante a criação dos diálogos, o autor pode ter uma pré-visualização de como ficarão os diálogos para o usuário final. O botão mostra os diálogos na sequência de exibição e com os textos e configurações atribuídos durante a criação. A diferença da visualização, em relação à execução real, é que o script não é executado e não há validações de campos não preenchidos, por exemplo. As demais características, como formatos, tamanho e seção de dúvidas são apresentadas como na execução real.

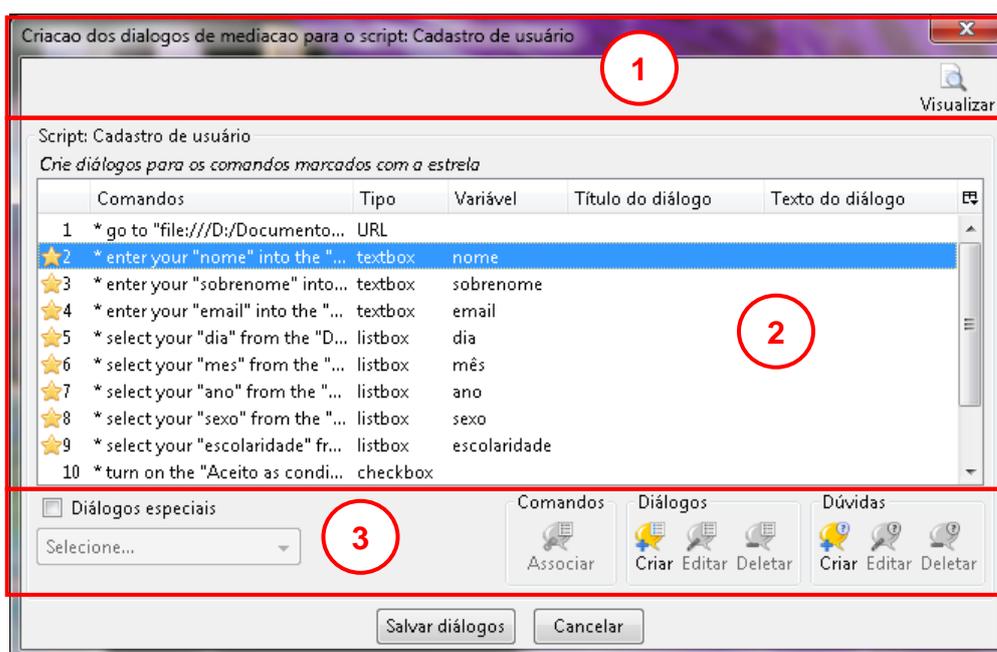


Figura 3.23 – Tela inicial do editor de diálogos

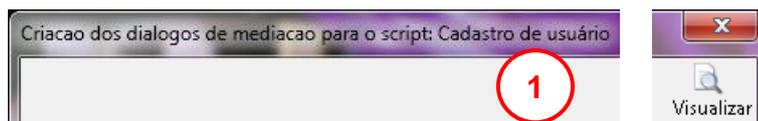


Figura 3.24 – Nome do script na barra de títulos e botão “Visualizar”

Script: Cadastro de usuário
Crie diálogos para os comandos marcados com a estrela

	Comandos	Tipo	Variável
1	* go to "file:///D:/Documentos/workspace/WNH-NOVO/teste.html"	URL	
★2	* enter your "nome" into the "Nome:" textbox	textbox	nome
★3	* enter your "sobrenome" into the "Sobrenome:" textbox	textbox	sobrenome
★4	* enter your "email" into the "Email:" textbox	textbox	email
★5	* select your "dia" from the "Dia:" listbox	listbox	dia
★6	* select your "mes" from the "Mês:" listbox	listbox	mês
★7	* select your "ano" from the "Ano:" listbox	listbox	ano
★8	* select your "sexo" from the "Sexo:" listbox	listbox	sexo
★9	* select your "escolaridade" from the "Escolaridade:" listbox	listbox	escolaridade
10	* turn on the "Aceito as condições de cadastro." checkbox	checkbox	
11	* click the "Leia o regulamento." link	link	
12	* turn off the "Desejo receber informações no meu email." checkbox	checkbox	
13	* click the "Enviar" button	button	

Figura 3.25 – Lista de comandos do script

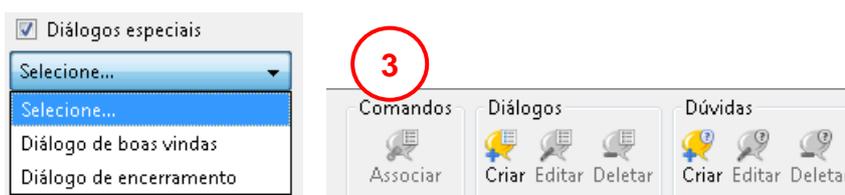


Figura 3.26 – Seleção de diálogos especiais e botões do editor

A lista de todos os comandos do script é mostrada com mais detalhes na Figura 3.25. O nome do script também é exibido no topo da lista. Abaixo do nome, há uma instrução para o usuário, indicando que os comandos marcados com o ícone em formato de estrela devem possuir diálogos correspondentes. Observe que estes comandos são exatamente os mesmos que possuem variáveis (coluna “Variável” está com valores preenchidos). Note que o nome das variáveis são os mesmos informados durante a criação do script no CoScripter (Tabela 3.1, p. 32). Os comandos que possuem variáveis correspondem às informações que serão solicitadas ao usuário durante a navegação pelos diálogos de mediação. Todos os comandos indicados com a estrela devem ter um diálogo associado. Caso o usuário tente finalizar a criação dos diálogos sem atentar para esta regra, será exibido um alerta. Analogamente, não é necessário criar diálogos para os passos que não possuem variáveis, pois eles serão executados de forma automática pelo CoScripter e não necessitam da intervenção do usuário final. De fato, ele nem precisa ter consciência de que estes passos são executados, por isso, durante a

execução do script com os diálogos, estes comandos intermediários ficam “ocultos”, à medida que ocorre comunicação direta com o usuário final, apenas quando são necessárias informações de entrada. Destacamos, entretanto, que o usuário não fica totalmente alheio ao que acontece, pois ele pode acompanhar todas as ações que ocorrem na interface original, enquanto o script e seus passos automáticos são executados.

Ainda na tabela de comandos do script (Figura 3.25), vê-se a indicação do tipo de elemento de interação (coluna “Tipo”) e, depois de o diálogo ter sido criado, seu título e texto aparecem nas colunas “Título do diálogo” e “Texto do diálogo” (colunas visíveis apenas na Figura 3.23, p. 43).

A Figura 3.26 mostra os botões de ação do editor além da lista de diálogos especiais. Os diálogos especiais são diálogos opcionais, que não precisam de entrada do usuário, ou seja, não possuem variáveis e não apresentam elemento de interação. Para a versão atual do editor, foram disponibilizados dois diálogos deste tipo: o diálogo de boas-vindas, através do qual o autor apresenta o objetivo da tarefa e dá instruções iniciais; e o diálogo de encerramento, no qual ele pode indicar a finalização da mediação e dar instruções sobre o estado final da interface original. Para criar um diálogo especial, é necessário selecioná-lo na lista e clicar no botão “Criar” do grupo “Diálogos”. Para criar as dúvidas dos diálogos, ele também escolhe na lista o tipo de diálogo e pressiona o botão “Criar” do grupo “Dúvidas”. As ações de editar e deletar para o diálogo e as dúvidas ocorrem de forma semelhante.

Para os diálogos “não especiais”, relacionados aos comandos indicados com a estrela na lista de comandos, basta selecionar a linha desejada e clicar no botão “Criar”⁷. Caso o usuário deseje que as informações de mais de uma variável sejam solicitadas no mesmo diálogo, por exemplo, nome e sobrenome (como mostrado na Figura 3.13, p. 37), ele tem duas formas de fazer: (a) cria o diálogo para o comando relacionado à primeira variável (por exemplo, “nome”), depois seleciona os dois comandos (linhas 2 e 3 na Figura 3.25) e clica em “Associar” (Figura 3.26); ou (b) seleciona as duas linhas e clica em “Criar”.

Após apresentar a tela inicial do editor, já podemos mostrar a criação dos textos e configurações dos diálogos. Vamos começar pela criação das boas vindas.

⁷ Outra forma de fazer isso é selecionar a(s) linha(s) desejada(s) e dar dois cliques no botão esquerdo do mouse. A tela de criação/edição será exibida.

O usuário marca a opção correspondente na lista de diálogos especiais e clica no botão “Criar” da área “Diálogos”. A área de criação/edição do editor para o diálogo de boas-vindas (e também de encerramento) é bastante simples (Figura 3.27). Ele é constituído apenas das áreas de inserção do título e do texto do diálogo, da área de visualização rápida e dos botões para salvar e cancelar. Após salvar a edição, estará criado o diálogo exibido na Figura 3.12 (p. 36).

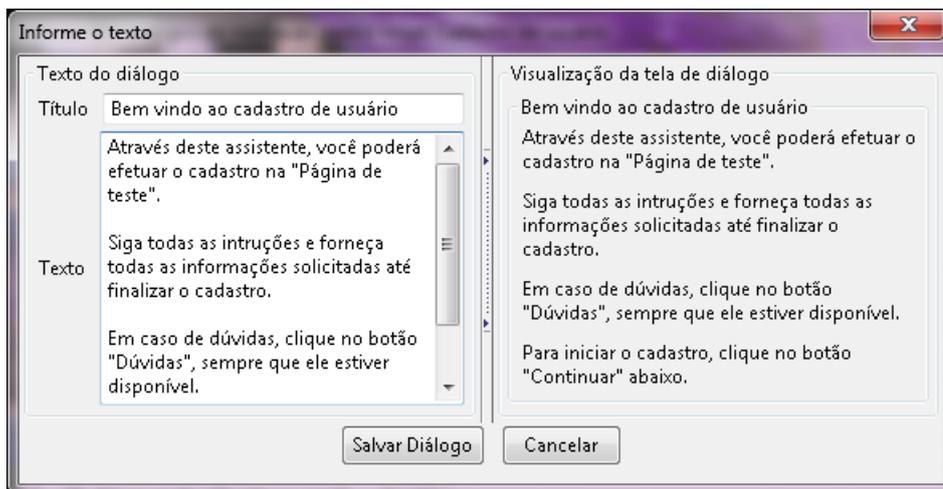


Figura 3.27 – Criação do diálogo de boas-vindas

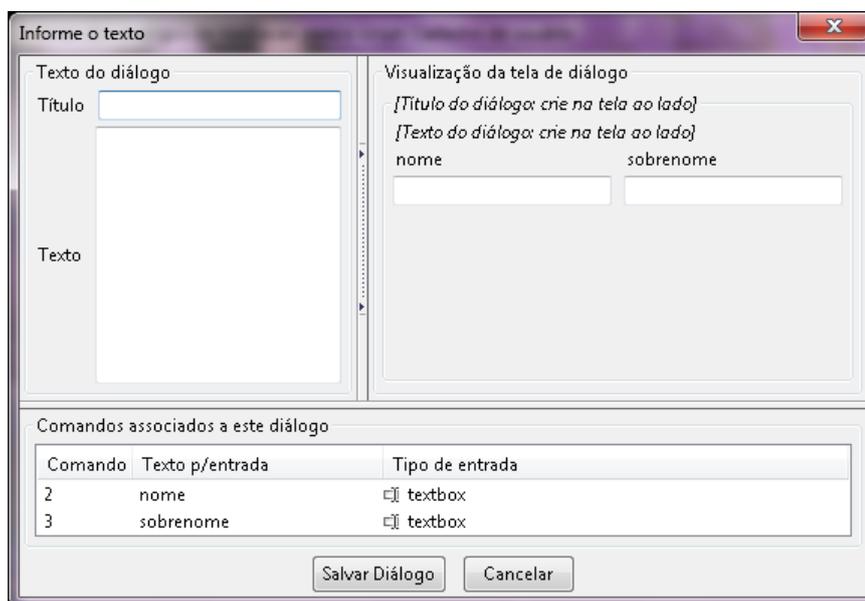


Figura 3.28 – Criação do diálogo para Nome e Sobrenome

O próximo diálogo a ser criado é o que solicita nome e sobrenome. Supondo que o usuário tenha selecionado as linhas correspondentes aos dois comandos e tenha clicado em “Criar”, a tela que aparece é como a da Figura 3.28. A estrutura superior da tela é semelhante à apresentada na Figura 3.27, com as caixas para inserção de título e texto e a área de visualização, que será atualizada quando

houver texto escrito. Note que na parte inferior da tela, há a lista de comandos associados a este diálogo. A tabela mostra o número correspondente ao comando no script, o nome da variável e o tipo original de entrada, no caso, *textbox* para ambos os comandos.

Quando o usuário clica em uma das linhas na lista de comandos, a área se expande e é exibida a seção de configuração dos elementos de entrada, conforme mostrado na Figura 3.29. As opções de alteração disponibilizadas pelo editor são as listadas a seguir:

1. Texto para a entrada: o editor, por padrão, usa como rótulo para o campo de entrada no diálogo, o nome da variável que vem do script. O usuário, caso deseje, pode alterar este texto para aquele que achar mais adequado.
2. Tipo da entrada: quando o tipo de entrada original é uma *listbox*, ela pode ser convertida em *radiobutton* ou *textbox*. Como a entrada para o nome é uma caixa de texto, não se pode alterar o tipo, por isso as opções estão desabilitadas na imagem.
3. Quantidade de caracteres: esta configuração aplica-se apenas quando o tipo de entrada é uma caixa de texto (caso da variável nome). O usuário informa o máximo de caracteres permitido na caixa. O valor escolhido também é utilizado pelo editor para ajustar o tamanho físico da caixa de texto na área do diálogo.
4. Formatação da entrada: quando a entrada é uma caixa de texto, o usuário pode escolher uma das opções disponíveis de formatação.

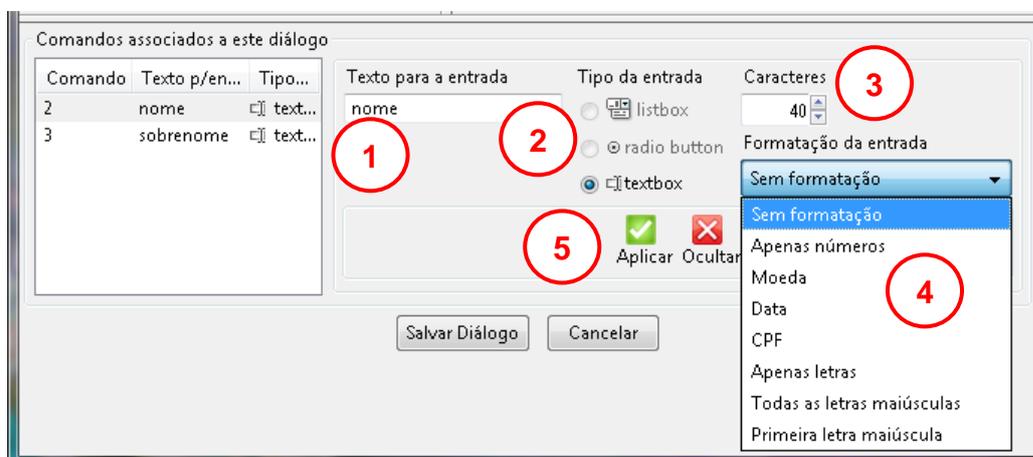


Figura 3.29 – Configurações da entrada

A quantidade de caracteres e o tipo de formatação são recursos que diminuem a probabilidade de erro durante o preenchimento dos campos nos diálogos de mediação. Ao clicar em “Aplicar” (Figura 3.29 (5)), as configurações são devidamente empregadas ao elemento em edição e tudo pode ser conferido na área de visualização. Observe na Figura 3.30, que o título e o texto já foram preenchidos e atualizados na área de visualização. Além disso, os nomes das variáveis passaram a ter a primeira letra capitalizada (Nome e Sobrenome).

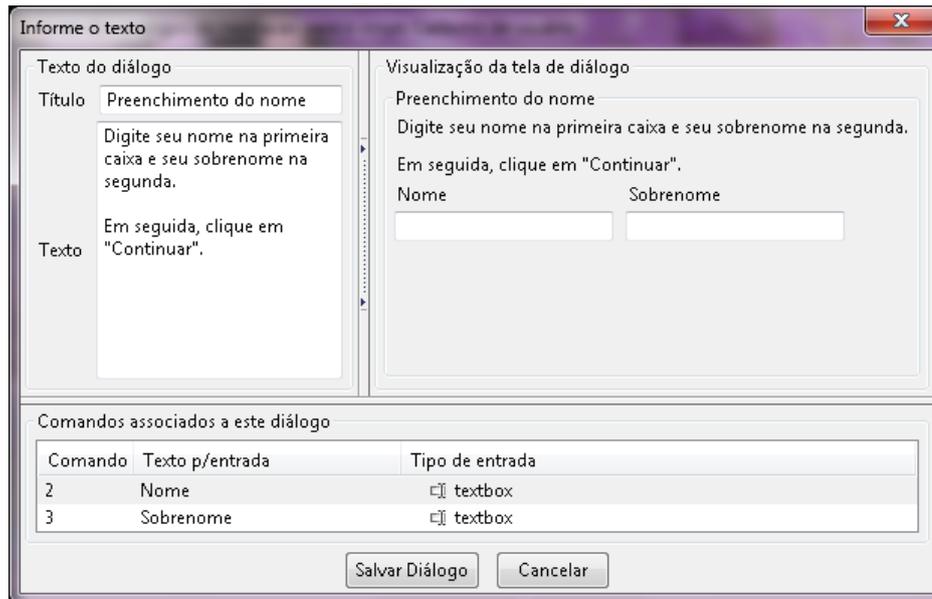


Figura 3.30 – Visualização do diálogo nome e sobrenome

Para complementar o entendimento sobre a alteração do tipo do elemento de entrada (Figura 3.29 (2)), mostraremos a seguir imagens com o processo feito para alterar a informação da data de nascimento para *textbox* e do sexo para *radiobutton*. A Figura 3.31 e a Figura 3.32 mostram a área de visualização com os elementos originais e as alterações feitas.

A cada diálogo que é construído, as linhas correspondentes na lista de comandos do script (Figura 3.25, p. 44) relacionadas ao diálogo são atualizadas com algumas informações. Observe, na Figura 3.33, como fica a lista depois de todos os diálogos para o cadastro de usuário estarem criados. A coluna “Tipo” mostra que os elementos de interação para a data de nascimento e para o sexo foram alterados. A coluna “Variável” apresenta as variáveis com seu texto atualizado. Neste caso, a única alteração em todas elas é a letra inicial maiúscula. As colunas para o título e para o texto do diálogo também estão preenchidas.

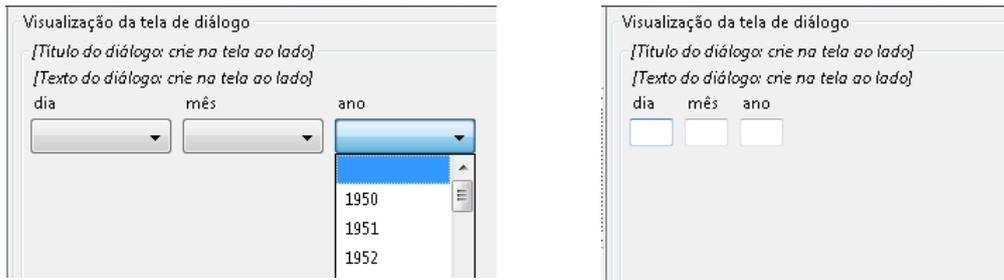


Figura 3.31 – Mudança de tipo na data: de lista para caixa de texto

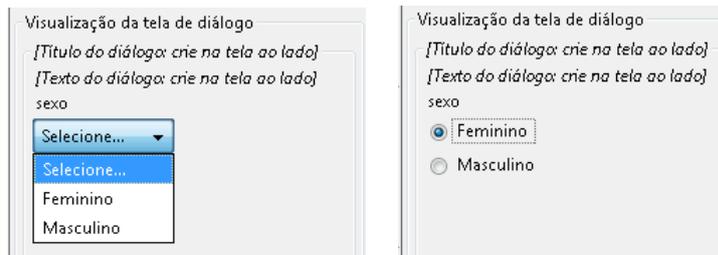


Figura 3.32 – Mudança de tipo no sexo: de lista para radiobutton

	Comandos	Tipo	Variável	Título do diálogo	Texto do diálogo
1	* go to "file:///D:/Documentos/workspa...	URL			
★2	* enter your "nome" into the "Nome:" te...	textbox	Nome	● Preenchimento do no...	Digite seu nome na p...
★3	* enter your "sobrenome" into the "Sobr...	textbox	Sobrenome	● [Associado: 2]	[Associado: 2]
★4	* enter your "email" into the "Email:" tex...	textbox	Email	Preenchimento do en...	Digite seu endereço e...
★5	* select your "dia" from the "Dia:" listbox	textbox	Dia	● Preenchimento da dat...	Abaixo, há três caixas ...
★6	* select your "mes" from the "Mês:" listb...	textbox	Mês	● [Associado: 5]	[Associado: 5]
★7	* select your "ano" from the "Ano:" listb...	textbox	Ano	● [Associado: 5]	[Associado: 5]
★8	* select your "sexo" from the "Sexo:" list...	radiobutton	Sexo	Preenchimento do Sexo	Informe o seu sexo, ...
★9	* select your "escolaridade" from the "Es...	listbox	Escolaridade	Preenchimento da Esc...	Informe seu nível de ...
10	* turn on the "Aceito as condições de ca...	checkbox			

Figura 3.33 – Tabela de comandos atualizada com os diálogos

Note que além do conteúdo textual, algumas linhas apresentam um ícone colorido no lado esquerdo da coluna “Título do diálogo”. Estas cores indicam os comandos do script que estão associados nos diálogos: os comandos 2 e 3, relacionados às informações de nome e sobrenome estão associados e representados pela cor verde; os comandos 5, 6 e 7, relacionados à data de nascimento, têm sua associação representada pela cor vermelha. Além do ícone colorido, a associação é indicada, nas colunas de título e texto, também pela expressão “[Associado: x]”, sendo “x” o número do primeiro comando da associação⁸.

Em qualquer momento durante a criação, o usuário pode visualizar, em conjunto, os diálogos até então criados. Basta estar na tela inicial do editor e clicar

⁸ Depois de o editor ter sido implementado e testado, percebemos que a escolha das cores para representar a associação de comandos devia ter sido feita com mais cuidado. Em geral, em interfaces, a cor vermelha remete às ideias de “alerta”, “cuidado”, “erro”; e a cor verde, às ideias de “correto”, “permitido”, “sucesso”. Para as próximas versões do editor, estas cores serão retiradas da paleta, mantendo-se apenas cores mais neutras.

no botão “Visualizar” (Figura 3.24, p. 44). Os diálogos são exibidos na ordem que aparecerão ao usuário final: caso haja um diálogo de boas-vindas, ele é o primeiro (Figura 3.34), caso não, será exibido o diálogo correspondente ao primeiro comando com variável da lista (marcado com a estrela). Ao clicar em “Continuar”, os diálogos subsequentes são exibidos.

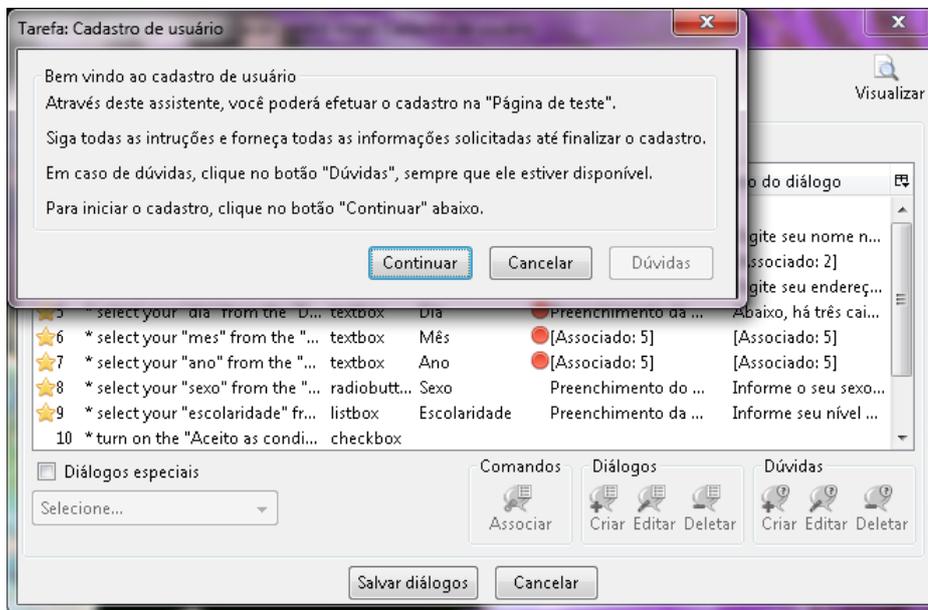


Figura 3.34 – Visualização dos diálogos criados

Conforme já citado, outra nova funcionalidade da versão atual do WNH é a possibilidade de se acrescentar perguntas e respostas aos diálogos de mediação, numa tentativa de auxiliar ainda mais os usuários com necessidades especiais. Para criar dúvidas de um determinado diálogo, o usuário deve selecionar a linha na lista de comandos ou escolher um dos diálogos especiais na lista correspondente e clicar em “Criar” na região de dúvidas (Figura 3.26, p. 44). A tela de criação/edição das dúvidas está representada na Figura 3.35. Há um campo para a pergunta e outro para a resposta. Quando uma pergunta é incluída, ela é inserida na lista de dúvidas para o diálogo em questão.

A Figura 3.36 mostra a lista com duas dúvidas criadas. Quando uma delas é selecionada, seu conteúdo é exibido nos campos correspondentes, caso o usuário queira editá-la. Ele pode ainda excluir a dúvida da lista e visualizar a janela com as dúvidas criadas, de forma semelhante àquela que aparece ao usuário final durante a execução dos diálogos de mediação (Figura 3.19 e Figura 3.20, p. 40). Para isso, basta clicar no botão “Visualizar”, posicionado acima da lista (Figura 3.36).

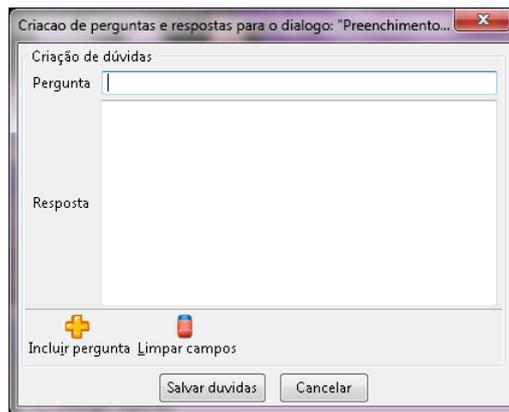


Figura 3.35 – Criação de dúvidas

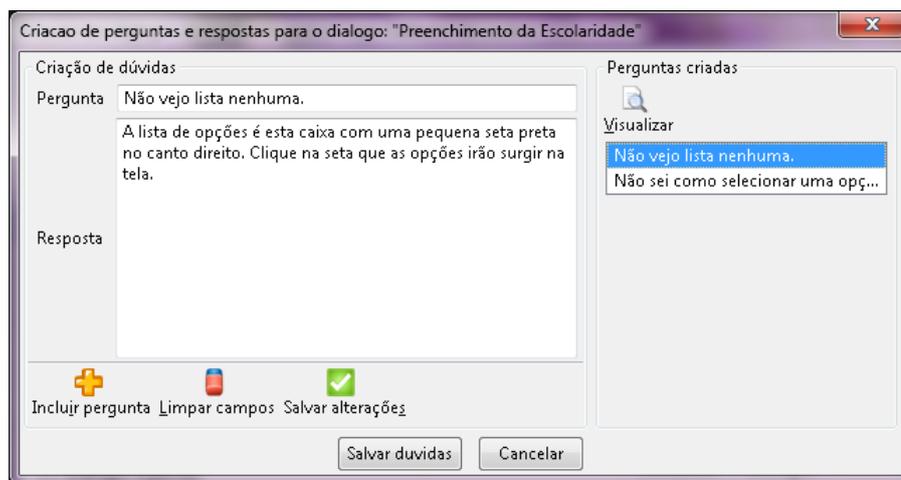


Figura 3.36 – Lista de dúvidas

Considerando que os diálogos e as dúvidas foram criados, o usuário salva seu trabalho, o editor é encerrado e ele pode então utilizar o botão “Executar” da área do WNH no editor do CoScripter (Figura 3.22, p. 42). Este botão inicia a execução dos diálogos, mas mantendo a interface do CoScripter ativa. Dessa forma, o usuário, além de ver as alterações acontecendo na página, pode acompanhar todo o processo de comunicação entre o CoScripter e o WNH. A Figura 3.37 apresenta o momento exato de preenchimento do segundo diálogo criado. As marcações em vermelho são feitas pelo próprio CoScripter, relacionando o comando do script que está sendo executado ao campo em atuação na página. Observe que as marcações estão de acordo com a informação (Nome) que está sendo solicitada no diálogo. Após se preencher os dois campos e se clicar em “Continuar”, os valores informados são passados para os campos correspondentes na página e o próximo diálogo é exibido, até que a mediação seja finalizada ou cancelada. É importante frisar que esta visualização da execução em conjunto com o script está habilitada apenas no modo de criação do diálogo. O

acesso aos diálogos, pelo usuário final, através da lista de tarefas mediadas (Figura 3.10, p. 36), acontece de uma forma mais simplificada, sem visualização do script e com poucas opções (Figura 3.11, p. 36), além das apresentadas nos diálogos.

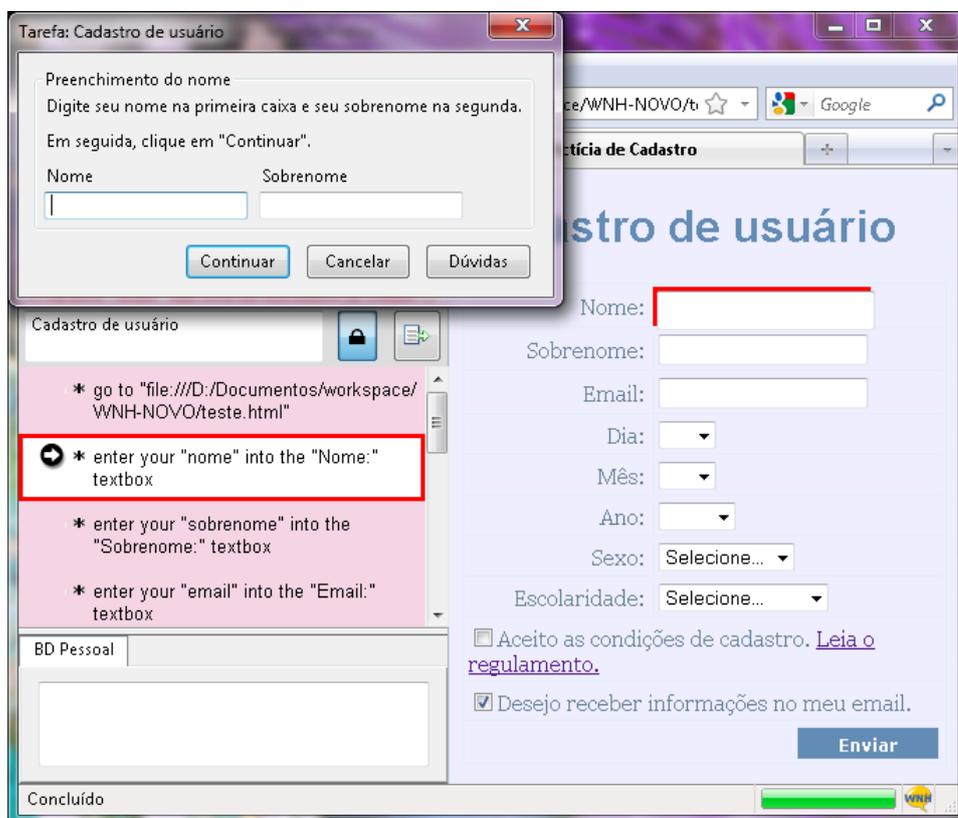


Figura 3.37 – Diálogo em execução no editor de scripts

3.5 Limitações do WNH

Antes de encerrarmos esta seção sobre o WNH, é essencial comentarmos algumas limitações técnicas da ferramenta. O primeiro fator é a “herança” vinda do CoScripter. Elementos que não sejam codificados em HTML puro, como Flash ou Applets Java não são reconhecidos pelo CoScripter. Isso acontece porque o gravador é “sensibilizado” apenas com os eventos tradicionais do formato HTML, disparados no navegador. A consequência disso é que páginas que possuam estas tecnologias não podem ser mediadas pelo WNH, já que não podem ter scripts criados para elas.

Uma limitação igualmente importante é que, para os diálogos do WNH, não é possível criar scripts que contenham, como valores das variáveis, termos que identifiquem elementos, na interface. Por exemplo, imagine que, na página de

cadastro de usuário, a informação do sexo fosse originalmente solicitada por meio de *radiobuttons*. Desta forma, o comando do script associado seria “turn on the “Feminino” radiobutton”, ou seja, o “Feminino” já é a identificação do elemento. Isto é diferente do comando da lista: “select “Feminino” from the “Sexo:” listbox”, que é identificada pela expressão “Sexo:” (e não por “Feminino”). Na forma como a arquitetura do CoScripter foi feita, antes de o diálogo ser apresentado, o elemento a sofrer uma ação é previamente localizado na página. Neste último caso, a lista “Sexo” é localizada e destacada na interface e, somente após isso, o diálogo solicitando a informação é exibido. Quando a “resposta” da informação é a mesma expressão que identifica o elemento (caso do sexo como *radiobutton*), o CoScripter não consegue identificá-lo na interface pois ele seria solicitado pelo diálogo que ainda nem foi exibido! Este problema ocorre com *radiobuttons*, *checkboxes* e links. Portanto, scripts que possuem comandos que contenham esses elementos como variáveis, atualmente, não podem ser mediados pelo WNH. Outra consequência desta limitação é que a conversão de tipos de elementos de interface ainda não ocorre, por exemplo, de um *radiobutton* para uma listbox ou *textbox*. Entretanto, o oposto não é verdadeiro, pois, como vimos, foi possível converter *listbox* tanto em *textbox* quanto em *radiobutton*.

Outra limitação da versão atual do WNH são as situações em que ocorrem atualizações nos elementos da página de forma dinâmica, não tratadas pelo script, durante sua criação no CoScripter. Por exemplo, suponha uma lista com as unidades federativas que, ao ter uma opção selecionada, atualiza outra lista correspondente aos municípios do estado escolhido. Por uma dependência técnica da arquitetura do CoScripter, o conteúdo das listas é salvo pelo WNH, ainda (e apenas) durante a gravação do script. Isto significa que, durante esta tarefa, serão salvas apenas as cidades correspondentes ao estado selecionado durante a gravação. Quando o usuário for criar o diálogo e desejar manter o elemento de entrada como lista, ele só terá acesso à lista de tais cidades. Caso o elemento seja alterado para caixa de texto, entretanto, não ocorre este problema, já que não é necessário recuperar dados da página previamente. É importante deixar claro ainda que, para atender aos objetivos dos experimentos finais (com o uso do editor real), o código foi adaptado (em um modo *hardcoded*) para capturar os dados esperados no cenário, não inviabilizando assim os testes com o uso de listas nos diálogos.

3.6 O potencial de acessibilidade

O WNH comporta-se, conforme sugerido nas imagens apresentadas, como um intérprete não só da página a que está associado, mas de toda a navegação por ela. O usuário final, *a priori*, interage *apenas* com os diálogos previamente criados, tem “mais chances” de ser poupado de eventuais problemas na página, sejam eles de usabilidade, navegabilidade, comunicabilidade, acessibilidade etc. A partir do momento em que se leva o usuário direto ao ponto que ele deseja, encurtando seu caminho durante a navegação – atalho esse obtido graças ao CoScripter –, há um caminho para a acessibilidade. O mesmo ocorre quando se evita, por exemplo, que um usuário de leitor de telas precise percorrer vários links e botões até chegar ao seu destino. E, principalmente, quando se abre a possibilidade de uma comunicação alternativa – os diálogos de mediação – à original na página, a fim de superar as barreiras encontradas, falamos de uma acessibilidade possível para diversos perfis de usuários com necessidades especiais. Esta é a ideia básica do WNH, vinda desde a sua concepção original: “Through this one-way channel, support-users are able to orient and instruct end-users during their script execution. Every problem and difficulty faced by end-users during the process could be guided by support-users.” (INTRATOR, 2009, p. 36). Para exemplificar o modo como deficientes visuais, ou demais usuários de leitores de telas, seriam beneficiados pela objetividade presente na automatização dos scripts e nos diálogos, observe o script listado no Quadro 3.2.

- | |
|---|
| <ol style="list-style-type: none"> 1. go to "http://www.boadica.com.br" 2. click the second "Login" link 3. click the "aqui" link 4. click the first "Cadastrar" link 5. enter your "NOME" into the "Nome:" textbox 6. enter your "CIDADE" into the "Cidade:" textbox 7. select your "UF" from the "Estado:" listbox 8. enter your "EMAIL" into the "E-mail:" textbox 9. You click the "Enviar" button |
|---|

Quadro 3.2 – Script de cadastro no site BoaDica

O script contém os passos necessários para efetuar um cadastro no site BoaDica. Observe que após entrar na página inicial do site (passo 1 no script), até que se tenha acesso ao formulário de cadastro (a partir do passo 5), são necessários três links: “Login”, “aqui” e “Cadastrar” (passos 2, 3 e 4,

respectivamente). Imagine quantos links um usuário de leitor de telas, teria que percorrer para encontrar estes três links e chegar à tela do formulário (Figura 3.38). Além disso, na página original, há uma grande quantidade de texto e vários campos de preenchimento não obrigatório (cinco campos de nove), conforme apresentado na Figura 3.38 (apenas os campos em vermelho são obrigatórios). Utilizando os diálogos do WNH para este script, são solicitadas somente as informações obrigatórias, e as instruções da página (ou parte delas) podem ser incluídas na seção de Dúvidas, caso o autor do diálogo julgue necessário.

Figura 3.38 – Página de cadastro do site Boadica⁹

Essa objetividade em se usar os scripts e os diálogos beneficia, além de deficientes visuais, pessoas com dificuldades motoras, pessoas com dificuldades de leitura, como analfabetos funcionais e até surdos, idosos etc., já que há menos ações a executar em relação à interface original.

Além disso, pelos diálogos pode-se “alterar” a linguagem e até mesmo o conteúdo da página original de acordo com a necessidade. Por exemplo, idosos podem ter acesso a diálogos com uma linguagem mais apropriada; aprendizes de informática podem ter acesso a diálogos preocupados em explicar a forma de usar os elementos de interação (botões, links, listas, etc.); pessoas com dificuldades de leitura e interpretação (e até surdos) podem ter acesso a diálogos com texto

⁹ <http://www.boadica.com.br/cadastros.asp>. Acesso em novembro de 2011.

simplificado. Enfim, os diálogos, como “intérpretes” e como interfaces alternativas à página, possuem o potencial de se adequar às peculiaridades de cada deficiência ou necessidade especial.

Retomando a forma de acessibilidade tradicional, baseada em diretrizes, brevemente abordada na Introdução, sabemos que o foco é a criação de HTML adequado ou a adequação de um HTML já criado. Percebe-se o caráter original e inovador do WNH ao afastar-se desta abordagem tradicional. O objetivo de acessibilidade é o mesmo, mas sem precisar mudar o que já existe. O que se propõe é a criação das interfaces alternativas – os diálogos de mediação – sem alterar o conteúdo original da página.

Todavia, não defendemos aqui que os criadores de sites deixem de se preocupar em desenvolver páginas acessíveis. Na verdade o WNH desponta como um possível elemento de transição: poderá ser usado enquanto houver sites inacessíveis. E, pensando de forma mais abrangente, sabemos que um site que atenda a todas as diretrizes de acessibilidade não está livre de apresentar dificuldades a perfis de usuários tão específicos como analfabetos funcionais ou idosos. Finalmente, o importante é que a abordagem do WNH não segue a clássica, mas também não lhe é uma ameaça ou concorrente. Ambas são aliadas e estão engajadas na mesma causa da acessibilidade.