

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/266140853>

Poirot –Um Ambiente para Avaliação de Interfaces de Usuário

Conference Paper · January 2001

CITATIONS

0

READS

3,500

2 authors:



[Tatiana Tavares](#)

Universidade Federal da Paraíba

1 PUBLICATION 0 CITATIONS

SEE PROFILE



[Jair Leite](#)

Universidade Federal do Rio Grande do Norte

62 PUBLICATIONS 133 CITATIONS

SEE PROFILE

Poirot - Um Ambiente para Avaliação de Interfaces de Usuário

Tatiana. A. Tavares

tati@dimap.ufrn.br

Jair Leite Cavalcanti

jair@dimap.ufrn.br

Departamento de Informática e Matemática Aplicada - DIMAp

Universidade Federal do Rio Grande do Norte - UFRN

Campus Universitário – Lagoa Nova – 59072-970 – Natal - RN

Resumo

O presente trabalho apresenta o ambiente para avaliação de interfaces de usuário Poirot. Este ambiente é fundamentado, principalmente, nos métodos: inspeção formal de usabilidade [10] e testes de comunicabilidade [11]. A abordagem do Poirot provê mecanismos para diferentes critérios de avaliação. Pode-se utilizar as ferramentas que compõe o ambiente para avaliar tanto aspectos de usabilidade quanto à comunicabilidade de uma interface. A avaliação através do Poirot permite o envolvimento de pessoas de diferentes especialidades e ligadas a diferentes níveis de desenvolvimento do sistema, seguindo, assim, a perspectiva de design participativo. Já o material de avaliação irá depender do critério de avaliação, mas inclui, obrigatoriamente, protótipos da interface em avaliação e alguns modelos do sistema. Para tanto, o Poirot define um ambiente de avaliação virtual que utiliza recursos de trabalho cooperativo, permitindo assim, que a interação entre a equipe de avaliação e a construção dos resultados. Neste artigo, apresentamos o Poirot, a descrição do processo de desenvolvimento desse sistema e os primeiros resultados obtidos que ilustram a prática de uso do Poirot na avaliação de interfaces.

Palavras Chave: Avaliação de Interfaces de Usuário, Inspeção Formal de Usabilidade, Comunicabilidade, Testes com Usuários, Avaliação Remota, Trabalho Cooperativo, Design Participativo.

Abstract

The current work proposes the Poirot, an environment for user interface evaluation. The Poirot is based on two evaluation methods: formal usability inspection (Kah & Prail, 1994) and communicability tests (De Souza, 1999). Poirot is composed by several evaluation modules, each of them evaluating a specific criterion. In this work, our focus is on the communicability module in order to provide support to the identification and analysis of designer-user communicability breakdowns. We also present the Poirot development process which are based in the prototyping and evolutionary paradigms. So, we used UML (Unified Modelling Language) and Rose for the description, specification and documentation of Poirot. In order to validate our proposal, we conclude this work presenting the recent results to confirm and evaluate our goals in practice situations.

KeyWords: User Interface Evaluation, Formal Usability Inspection, Communicability, CSCW, Participative Design, Remote Evaluation, and User Tests.

1. Introdução

Nas primeiras décadas do desenvolvimento de software, programadores desenvolviam aplicações voltadas para uso pessoal ou de sua equipe de trabalho. Ao passar do tempo, essa visão de uma única e maciça equipe de desenvolvimento foi desmembrada em grupos menores com objetivos mais específicos dentro do processo de desenvolvimento de sistemas [01]. Logo, surgiram abordagens mais participativas de desenvolvimento de software que além de incluir especialistas de diferentes áreas, também passam a considerar o próprio usuário no processo de desenvolvimento. Outras preocupações também foram sendo incorporadas ao desenvolvimento de sistemas, assim como a preocupação com a interface de usuário.

Dix destaca que na visão do usuário, a interface da aplicação é o próprio sistema [02]. Então, a interface que antes era considerada um artefato menos prioritário, agora merece destaque no ciclo de desenvolvimento de sistemas. Este fato é evidenciado pela inclusão de atividades responsáveis pela criação, desenvolvimento, implantação e avaliação de interfaces no ciclo de desenvolvimento. Neste processo, a avaliação tem um papel fundamental. A subjetividade inerente ao seu humano não permite métodos de desenvolvimento de interfaces objetivos que possam garantir que uma determinada interface seja de excelente. É através da avaliação que se pode observar os defeitos e as virtudes da interface.

Todavia, na prática de desenvolvimento de sistemas, a implantação da etapa de avaliação ainda é uma realidade utópica. Isto é, uma realidade ligada aos laboratórios de pesquisa e às grandes empresas e raramente incorporada aos projetos de médio e pequeno porte. As dificuldades enfrentadas são decorrentes, sobretudo, do encargo teórico inerente a maioria dos métodos existentes e da carência de ferramentas de fácil acesso e uso.

Os esforços se concentram tanto na concepção de métodos e técnicas eficazes para avaliação quanto na construção de ferramentas que auxiliem esses métodos. A construção de ferramentas para avaliação vem atraindo a atenção de vários autores [03,04,05,06,07] os quais acreditam na importância das ferramentas para otimização e popularização da etapa de avaliação. As ferramentas automatizam, ainda que parcialmente, determinados processos o que agiliza a avaliação e torna esta etapa menos burocrática.

O objetivo deste trabalho é apresentar o Poirot, um ambiente computacional de apoio à avaliação de interfaces de usuário. Este ambiente disponibiliza ferramentas para auxiliar o avaliador nos diversos estágios da avaliação. Os resultados são construídos por uma equipe de avaliação que avalia o material disponível através das ferramentas do ambiente. A principal função do Poirot é gerenciar a equipe (usuários), o material (informações) e as ferramentas de avaliação (subsistemas). Este artigo apresenta o projeto e o desenvolvimento deste ambiente. Para tanto, é apresentada a fundamentação teórica necessária, seguido dos principais aspectos de projeto e desenvolvimento e, por fim, os resultados obtidos com os testes iniciais do ambiente proposto.

2. Fundamentação Teórica

A implantação de uma avaliação pressupõe a consideração de alguns fatores básicos, os quais delimitam o escopo e o tipo da avaliação, são eles: objetivo, critérios, estilo, participantes, métodos e técnicas e, por fim, as ferramentas envolvidas na avaliação [08]. A partir da definição destes fatores é delineado um determinado tipo de avaliação. Existem diferentes tipos de avaliação, cada tipo possui características próprias que avaliam objetivos específicos e implementam métodos de avaliação distintos.

Um dos fatores mais importantes é a definição dos critérios de avaliação que definem quais características da interface serão exploradas durante a avaliação. Atualmente, o critério mais comumente adotado é a usabilidade [09]. Este critério está presente na maioria dos métodos [09,10] e é fato comum a todas as ferramentas disponíveis [03,04,05,06,07]. Porém, a usabilidade ainda não constitui um conceito unânime na literatura. A variância deste conceito é dada através dos aspectos abordados pelos autores. Números de erros, aprendizado, facilidade de uso, satisfação subjetiva e performance são alguns desses aspectos.

Outro critério bem mais recente no contexto de avaliação é a comunicabilidade [11]. A comunicabilidade de um sistema é a propriedade através da qual o sistema transmite ao usuário de uma forma eficaz e eficiente as intenções e princípios de interação que guiaram o design da interface. A Figura 1 ilustra o escopo da avaliação por comunicabilidade e usabilidade, que podem ser vistas como métodos complementares uma vez que a usabilidade enfoca a comunicação usuário – sistema e comunicabilidade a comunicação designer – usuário através do sistema. Por outro lado, a avaliação de comunicabilidade envolve uma visão mais global do sistema voltada a avaliação da comunicação entre designer, sistema e usuários, considerando, sobretudo, a interação. Além disso, esta abordagem está fundamentada na proposta teórica da Engenharia Semiótica, o que propicia uma série de novas expectativas e justificativas para os fatores considerados.

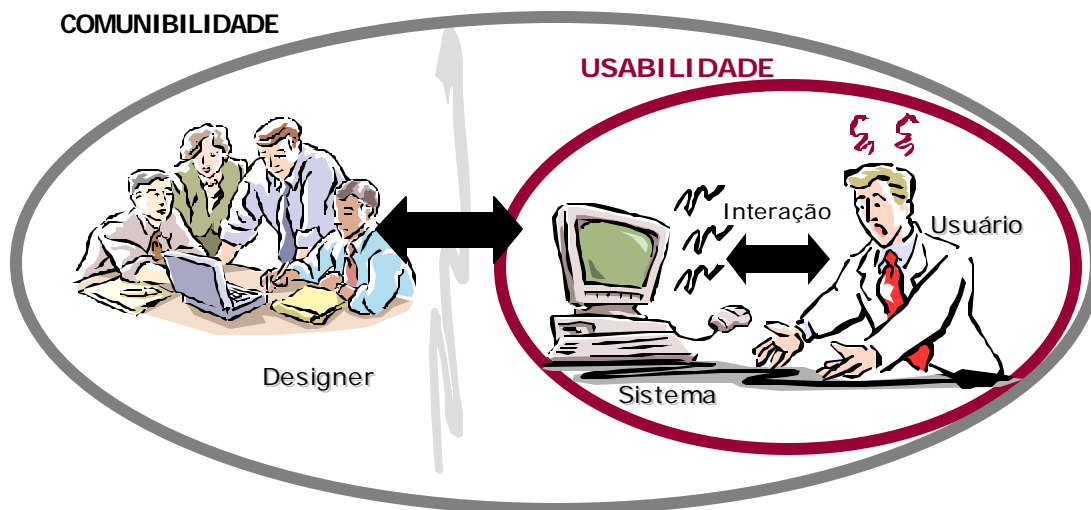


Figura 1: Comunicabilidade X Usabilidade.

Uma vez determinado o critério de avaliação, os passos subsequentes envolvem a definição da equipe, do material, do local, do método e das ferramentas de avaliação. Um dos diferenciais do Poirot é fornecer suporte a essa sequência de decisões relacionadas com a avaliação, já que permite a avaliação sob diferentes critérios: usabilidade e comunicabilidade. No contexto de usabilidade o Poirot vem acrescentar mais um recurso aos adeptos dessa abordagem. Tratando-se de testes de comunicabilidade, o Poirot é o pioneiro no oferecimento de ferramentas especialmente projetadas para auxiliar esse método.

3. Trabalhos Correlatos

Um dos passos rumo ao desenvolvimento do Poirot foi fazer um levantamento sobre as ferramentas de avaliação disponíveis. Nesta seção procuramos sintetizar os principais pontos desse levantamento destacando as vantagens e

desvantagens dessas ferramentas. A Tabela 1 apresenta as comparações entre as ferramentas estudadas, destacando, o critério, as técnicas e métodos, o ponto forte e o ponto fraco de cada ferramenta.

Parâmetros Ferramentas	Critério	Técnicas/Métodos	Ponto +	Ponto -
MetriStation [06]	Performance do usuário.	Métodos de monitoração e observação	Captura e análise de dados automática.	Necessita de configuração de hardware mínima.
Sherlock [05]	Consistência Visual e Textual de GUIs.	Métodos de inspeção/checagem de padrões.	Automatiza o processo de inspeção.	Necessita de uma entrada padrão
FaiWin [03]	Layout de Telas – Organização Visual	Métodos de monitoração.	Captura dados automaticamente.	Restrita a Plataforma de Desenvolvimento
QASU [07]	Satisfação Subjetiva do Usuário.	Métodos de Inspeção, Questionários e Avaliação Remota	Fácil de usar – Ferramenta em ambiente Web.	Depende excessivamente do usuário
SPYxSPY [12]	Usabilidade – performance do usuário.	Métodos de Inspeção – Monitoramento.	Registrar todas as ações dos usuários.	Depende do SO utilizado.

Tabela 1: Comparações entre as ferramentas de avaliação.

Dentre as características comuns entre as ferramentas merece destaque o critério da avaliação. Em alguns casos, como a MetriStation, a Sherlock e a Q.A.S.U. este critério apresenta-se bem específico, focalizando performance do usuário, consistência visual e textual de GUIs e satisfação subjetiva do usuário, respectivamente. No entanto, todas as ferramentas objetivam a avaliação de usabilidade ainda que de forma parcial. Outro ponto interessante destas ferramentas é que embora partilhem do mesmo critério de avaliação, a usabilidade, elas baseiam seu funcionamento em métodos e técnicas de avaliação diferentes o que as torna particular em determinados pontos. Por exemplo, a MetriStation baseasse nos métodos de monitoração e observação propondo uma estação de trabalho especialmente dedicada para tanto. Este fato torna a MetriStation uma ferramenta muito eficaz na captura e análise automática de dados

Atualmente, a utilização tanto de métodos quanto de ferramentas de avaliação no âmbito de interfaces de usuários ainda é um privilégio de projetos capazes de arcar com o investimento necessário. Deste modo, é preciso utilizar novas alternativas tecnológicas que tornem possível a popularização tanto dos métodos quanto das ferramentas de avaliação, fazendo com que a equipe de desenvolvimento se preocupe efetivamente com a elaboração e avaliação da interface do sistema durante o desenvolvimento do mesmo.

4. O Método de Avaliação do Poirot

A concepção do Poirot agregou várias idéias provenientes de teorias, técnicas, métodos e ferramentas de avaliação. Os métodos fomentadores do Poirot são a Inspeção Formal de Usabilidade – IFU [10] e os testes de comunicabilidade [11]. Porém, o Poirot também se baseia nas técnicas de avaliação remota, questionários, observação e testes.

Kahn e Prail [10] descrevem o método de Inspeção Formal de Usabilidade como uma evolução da Engenharia de Usabilidade proposta por Nielsen [09]. Este método foi desenvolvido com o objetivo de auxiliar os desenvolvedores na revisão de um produto. Para tanto, o método especifica um processo formal para detecção e descrição de erros de usabilidade, um conjunto de responsabilidades para cada participante da avaliação, e por fim, um framework que define estruturalmente todo o processo. A implantação deste método é feita de forma convencional sem o apoio de uma ferramenta computacional específica e exige reuniões presenciais dos participantes da avaliação.

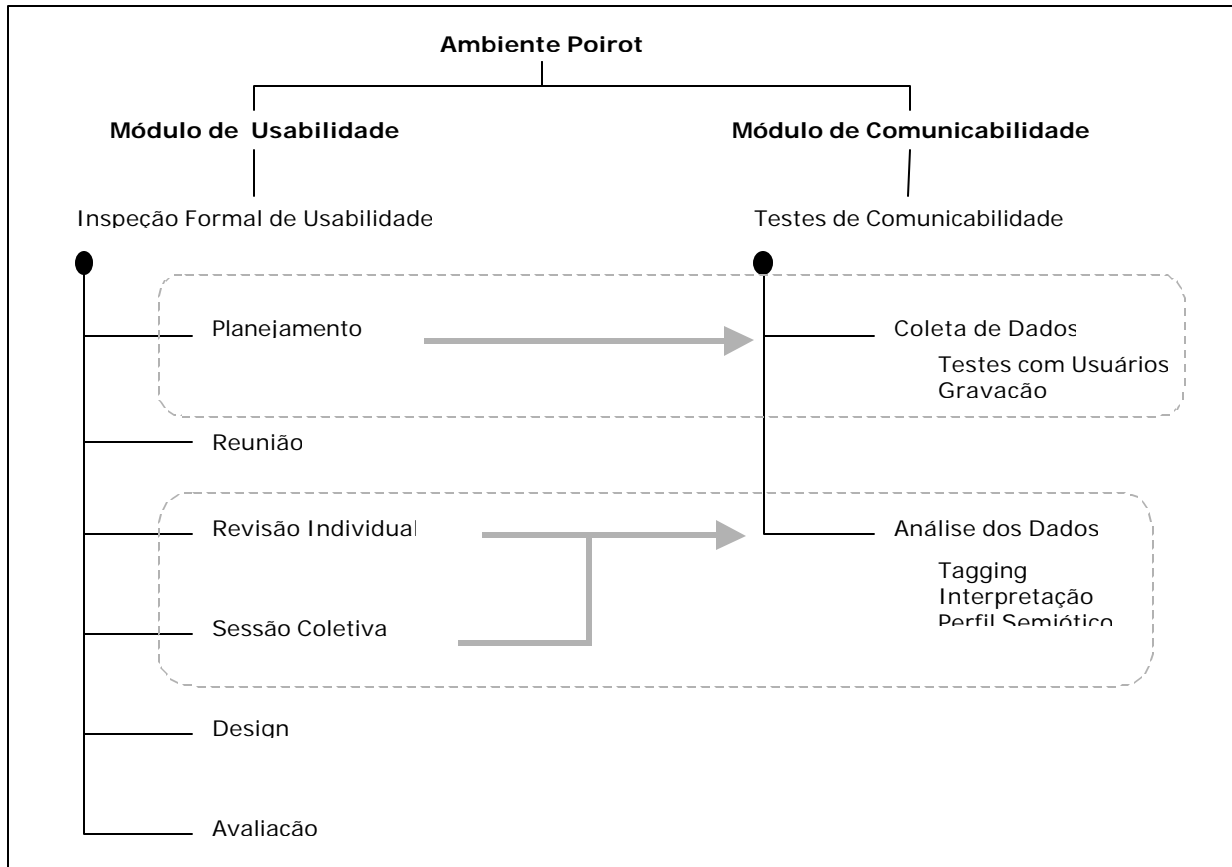


Figura 2: Método de Avaliação do Poirot.

O método proposto por De Souza, Prates e Barbosa [11] utiliza as idéias da Engenharia Semiótica numa aplicação prática para a avaliação do design de interfaces. Na abordagem da Engenharia Semiótica os sistemas são vistos como mensagens de designers para usuários constituindo uma espécie de conversação. Manter a continuidade nesta conversação significa manter a estabilidade da própria aplicação. O objetivo deste método é a detecção de rupturas na interação. Essas rupturas descrevem situações onde a continuidade na conversação foi rompida ou mal formulada.

O método de avaliação utilizado pelo Poirot é um híbrido destes dois métodos. A Figura 2 ilustra esse hibridismo: para o módulo de usabilidade o Poirot utiliza apenas o método de inspeção formal de usabilidade, para o módulo de comunicabilidade são mescladas as etapas de avaliação abordadas por ambos os métodos, adotando os testes de comunicabilidade na definição da estrutura interna. Deste modo, a sistemática externa é baseada na inspeção formal de usabilidade, isto é, implementa as etapas de gerenciamento da avaliação previstas pelo método (planejamento, reunião, revisão individual, sessão coletiva, design e avaliação). Dentro de cada etapa existem atividades que devem ser executadas. No módulo de comunicabilidade são encaixadas as atividades dos testes de comunicabilidade, o que define uma sistemática interna de trabalho.

5. O Ambiente Poirot

O Poirot é uma junção de ferramentas computacionais que executam funções estratégicas na avaliação de interfaces. Essas ferramentas estão divididas em quatro grupos: configuração, comunicação, inspeção e suporte. Como o ambiente Poirot provê dois módulos de avaliação: usabilidade e comunicabilidade, para cada módulo de avaliação existem ferramentas específicas e outras comuns.

A primeira etapa do processo de avaliação no Poirot é a submissão de um projeto Poirot responsabilidade de uma ferramenta de configuração. Durante a submissão de um projeto o solicitante define o critério de avaliação, descreve as características do produto e do projeto em avaliação, bem como, algumas informações pessoais. A Figura 3 apresenta a tela para submissão de um projeto ao ambiente Poirot. Uma vez que o projeto já é aceito pelo Poirot devem ser executadas as demais ferramentas de configuração que ajustam o escopo da avaliação solicitada as etapas de execução do Poirot. Estas ferramentas flexibilizam a avaliação, dando um tratamento mais específico aos projetos. As configurações definem as atividades de execução do processo de avaliação, o material e as ferramentas disponíveis, os usuários envolvidos, as tarefas que serão analisadas e o cronograma de trabalho.

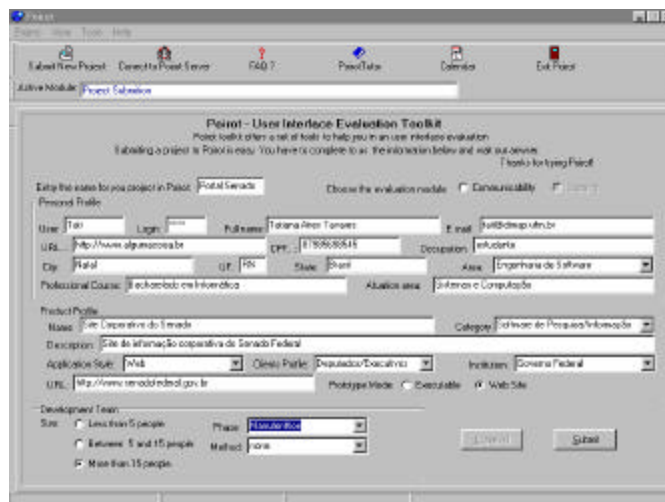
A imagem mostra a interface de avaliação de usabilidade do Poirot. No topo, há uma barra de menu com opções como 'File', 'Edit', 'Tools', 'Help'. Abaixo, uma barra de ferramentas com ícones para 'Edit New Project', 'Cancel to Poirot Server', 'F50 ?', 'Poirot Tools', 'Database', e 'Exit Poirot'. O título da janela é 'Poirot - User Interface Evaluation ToolKit'. O texto principal indica que o usuário está criando um projeto e deve preencher as informações. O formulário contém campos para: 'Enter the name for your project in Poirot' (com 'Project Evaluation' selecionado), 'Personal Profile' (com campos para Name, Login, Full Name, E-mail, URL, City, State, Country, Professional Course, e Address), 'Product Profile' (com campos para Name, Description, Application Site, Client Profile, Institution, e URL), e 'Development Team' (com campos para Size, Phase, e Method). Botões 'OK' e 'Cancel' estão no canto inferior direito.

Figura 3: Tela para Submissão de um Projeto.

As ferramentas de comunicação são destinadas ao envio e recebimento de mensagens de forma síncrona e assíncrona entre os usuários. Essa comunicação pode acontecer via correio eletrônico, lista de discussão, mecanismos de chat ou videoconferência. O conjunto dessas ferramentas implementa o que chamamos de sessão virtual de avaliação, onde os avaliadores discutem os problemas detectados e as soluções sugeridas. A **Figura 4** ilustra o que seria uma sessão virtual de avaliação através do Poirot.

As ferramentas de suporte auxiliam as atividades que antecedem a avaliação. O Editor de Modelo de Usuários e o Editor de Cenários de Tarefas já possuem versões funcionais, enquanto que o Repositório de Casos e o Editor de Modelo de Tarefas estão em versão de apresentação. Os softwares de Captura, ou ScreenCam utilizados são softwares comerciais.

As ferramentas de inspeção constituem o núcleo funcional da avaliação. Através destas ferramentas o Poirot trata de forma distinta os critérios de avaliação que aproveitam a mesma infra-estrutura do ambiente para avaliar diferentes enfoques. Para tanto, o módulo de usabilidade conta com a ferramenta GEU (Grade de Erros de Usabilidade) e o módulo de comunicabilidade das ferramentas USInspector e TagPanel.

O GEU disponibiliza uma área onde os inspetores devem anotar os erros de usabilidade detectados bem como as sugestões para solução. Depois que cada participante já preencheu sua grade de erros, o sistema unifica os resultados parciais obtidos constituindo uma listagem única de erros que será reavaliada buscando uma solução o mais completa e unificada possível para os problemas detectados.

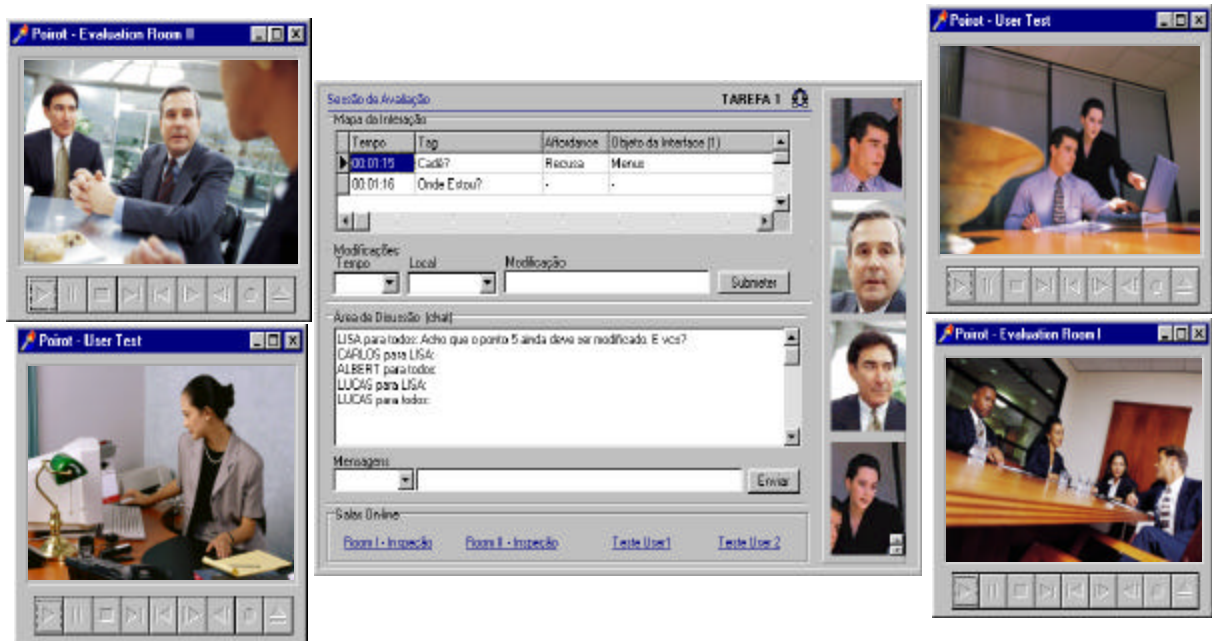


Figura 4: Sessão Virtual de Avaliação.

O USInspector conduz a realização dos testes com a aplicação avaliada. Sua função é apresentar ao usuário as tarefas que ele deve executar e acionar sua execução e gravação. Paralelamente o USInspector ainda disponibiliza um módulo para capturar as interjeições detectadas pelo usuário teste durante a interação.

O TagPanel exibe a gravação da interação dos usuários teste, isto é, exibe os arquivos gerados a partir de softwares de captura. Além disso, ele implementa mecanismos de anotação, disparados quando o vídeo é pausado permitindo que o avaliador faça anotações sobre a interjeição detectada, classifique-a de acordo com as situações de ruptura descrita por de Souza (1999) e sugira soluções de design que corrijam este problema. Além disso, a ferramenta, sincroniza as interjeições detectadas pelo USInspector, possibilitando que o avaliador concorde, discorde e comente as interjeições apontadas pelo usuário. Ao final, a ferramenta emite um relatório das informações obtidas denominado mapa da interação.

6. O Desenvolvimento do Ambiente Poirot

A metodologia adotada no desenvolvimento do Poirot utiliza os paradigmas de prototipação e evolutivo incremental. A característica marcante na prototipação é a construção rápida de protótipos os quais facilitam a validação e refinamento dos requisitos do sistema. Este modelo é especialmente útil em casos onde não há um modelo computacional a ser seguido, o que torna esta abordagem adequada ao escopo deste trabalho. O paradigma evolutivo incremental fundamenta a adoção da política de versões, isto é, da construção de protótipos que evoluem para o produto final através da inserção de novos módulos funcionais que complementam os objetivos da proposta.

A análise de requisitos centrou-se num levantamento bibliográfico em busca do embasamento teórico envolvendo a avaliação de interfaces, métodos, técnicas e ferramentas disponíveis. A partir do refinamento desses requisitos iniciais, o próximo passo foi o projeto rápido do sistema. Nesta etapa foi feita a modelagem conceitual do Poirot. Através do modelo construído foi possível identificar os principais módulos capazes de realizar funções, bem como, a maneira com a qual estes módulos se comunicam, além das informações necessárias. Para tanto, adotou-se como notação a UML (Unified Model Language) [13] e como ferramenta, o Rose [14].

A modelagem do sistema Poirot define a estrutura de alto nível do sistema em função de diferentes visões que representam a aplicação em diferentes níveis de abstração. Utilizamos as seguintes visões e modelos:

- Visão de Casos de Uso.
- Visão Lógica.
- Visão de Componente.
- Visão de Desenvolvimento ou de Implantação.

Na representação de cada visão foram utilizados modelos descritos por diagramas UML, cenários [16] e LEMD [15]. A descreve as visões, os modelos e a ferramentas utilizadas.

Visão	Modelo	Ferramentas Utilizadas
Casos de Uso	Modelo de Casos de Uso	Cenários Diagramas de Casos de Uso Diagramas de Atividades
Lógica	Modelo Conceitual da Aplicação	Diagrama de Seqüências
	Modelo Funcional da Aplicação	Diagrama Casos de Uso de Realização Diagrama de Atividades
	Modelo da Interação	Diagramas de Interação LEMD*
	Modelo de Dados	Diagramas de Classes
Componentes	Modelo de Componentes	Diagrama de Componentes
Desenvolvimento	Modelo de Implantação	Diagrama de Implantação

Tabela 2: Modelos e Ferramentas Utilizadas no Desenvolvimento do Poirot. (*) ferramentas externas a UML.

A documentação do Poirot é uma etapa fundamental para a proposta deste ambiente. Além da importância para o desenvolvimento do que se tem hoje, esse material é imprescindível para traçar os novos rumos do Poirot. Nossa proposta é bastante receptiva a extensões e para tanto, abre espaço em seu projeto para recebê-las. O projeto do Poirot suporta a inserção de novos módulos os quais poderão ser acrescidos na forma de outros pacotes sem desestruturar o projeto existente. Além de prover uma fonte de descrição e especificação das idéias propostas, a modelagem e documentação do Poirot podem auxiliar em outros projetos de sistemas, pois diversos modelos foram especificados o longo do projeto do ambiente.

Quanto à estratégia de implementação o Poirot está sendo desenvolvido em fases e utiliza como plataforma de desenvolvimento o Delphi e sua linguagem nativa, o Object Pascal . Atualmente o Poirot conta com protótipos funcionais para as ferramentas ligadas ao módulo de comunicabilidade e com protótipos de apresentação para o GEU e ferramentas de comunicação. Nas versões atuais as ferramentas USInspector e TagPanel já foram submetidas a testes em situações reais de uso, bem como, algumas ferramentas de configuração e de suporte (Editor de Modelo de Usuários e Editor de Cenários).

7. Resultados Obtidos

Os primeiros resultados obtidos foram a partir da execução de alguns testes com os protótipos do Poirot. Através destes testes foi possível avaliar e validar nossos objetivos com o desenvolvimento deste trabalho. Duas aplicações foram utilizadas para os testes: uma aplicação convencional *stand alone* [17] e *website* corporativo [18]. Além disso, foi envolvido um grupo de quatro voluntários que auxiliaram na execução dos testes.

Os primeiros resultados ressaltaram a aplicabilidade e eficácia das ferramentas de suporte na elaboração do material de avaliação, sobretudo, da ferramenta para modelagem de usuários (Figura 5). Além de auxiliarem na modelagem, elas agilizam e padronizam este processo tornando-o mais rápido e efetivo dentro da etapa de avaliação. A nossa contribuição ainda vai um pouco além desse contexto, já que as ferramentas propostas também podem ser utilizadas em outras etapas do processo de desenvolvimento de sistemas e podem ser utilizadas independentes do Poirot.

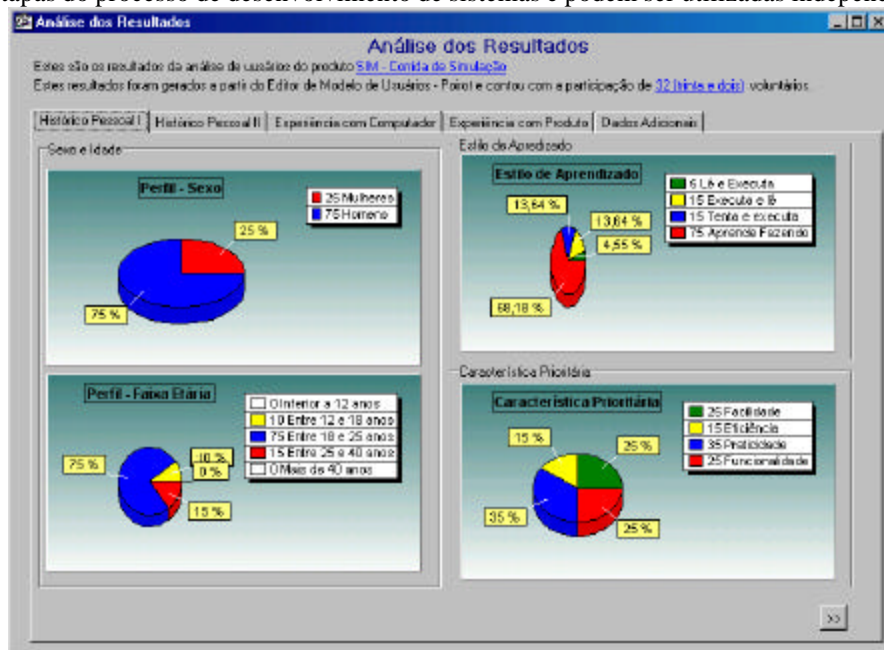


Figura 5: Tela da Ferramenta de Modelo de Usuários.

As ferramentas TagPanel e USInspector foram as nossas maiores contribuições neste trabalho. A ferramenta TagPanel mostrou-se capaz de auxiliar o avaliador na identificação e, sobretudo, na descrição das interjeições. Em geral, os resultados de uma avaliação enfocam os erros e não as correções. A possibilidade inovadora de descrever os objetos de interface relacionados com as interjeições foi um salto em direção a correção dos erros levantados pela avaliação e de como e onde corrigi-los.

A ferramenta USInspector na versão testada e com a configuração utilizada não se mostrou autônoma para realização dos testes. Os problemas detectados estão relacionados principalmente com o software de captura utilizado. Porém, a captura das interjeições junto ao usuário foi uma ótima estratégia para tornar os resultados obtidos mais realistas e precisos.

Por fim, a utilização do Poirot impõe a produção de resultados em diferentes níveis e direcionados a pessoas que atuam de forma diferenciada no processo de desenvolvimento. Esse fato torna os resultados do Poirot mais efetivos que outros, quando aplicados à correção dos problemas detectados e ao entendimento das situações que levaram a esses

erros, isto é, todo problema levantado pela avaliação do Poirot, possui uma descrição do problema e outra da solução. A descrição da solução está sempre atrelada a objetos da interface analisada, o que facilita o processo de correção.

O Poirot foi desenvolvido com o intuito de propor uma alternativa para apoiar o design de software. Dentro desta perspectiva, propor um ambiente de avaliação de interfaces sob diferentes aspectos é uma forma de contribuir na melhora da usabilidade e comunicabilidade dos sistemas. Neste sentido acreditamos que o Poirot atingiu nossas expectativas e abriu novos caminhos para estudo. Ainda há muito que ser feito, como, a implementação das ferramentas de comunicação, implementação da versão Web e incorporação de outros critérios de avaliação. Mesmo assim nossos objetivos foram alcançados e espera-se que o Poirot possa contribuir para popularização da avaliação de interfaces de usuários dentro do escopo de desenvolvimento de sistemas.

Referências

- [01] SHNEIDERMAN, Ben. Designing the User Interface: Strategies for Effective Human-Computer Interaction. USA: Addison-Wesley Publishing Company-Reading, 1987, 448p.
- [02] DIX, A., FINLAY, J., ABOWD, G., & BEALE, R. Human-Computer Interaction. Prentice Hall Europa: Londres, 1994.
- [03] DINIZ, Eliane da S. A. Ferramenta para Avaliação de Interfaces em Ambiente Windows, a partir da Monitoração de Dados – FAIWIN. Campina Grande - PB, 1996. Dissertação (Mestrado em Informática). Universidade Federal da Paraíba, Instituto de Informática.
- [04] LECEROF, Andreas e PATERNO, Fabio. Automatic Support for Usability Evaluation. In: IEEE Transactions on Software Engineering, Vol. 24, No. 10, Outubro. 863-888p. (1998)
- [05] MAHAJAN, Rohit e SHNEIDERMAN, Ben. Visual and Textual Consistency Checking Tools for Graphical User Interfaces In: IEEE Transactions on Software Engineering, nVol.23. No11 722-735p. (1997)
- [06] MAXION, Roy A. e SYME, Philip A. MetriStation: A Tool for User-Interface Fault Detection. In: Proceedings of the 27th International Symposium on Fault-Tolerant Computing - FTCS '97 (1997)
- [07] TAVARES, Tatiana Aires e DINIZ, Eliane da Silva Alcoforado. Uma Ferramenta para Avaliação Qualitativa de Interfaces em Ambiente Web. In: XXVI CONFERÊNCIA LATINO-AMERICANA DE INFORMÁTICA, 2000, Cidade do México. Anais do XXVI CONFERÊNCIA LATINO-AMERICANA DE INFORMÁTICA. 2000.
- [08] LEITE, Jair Cavalcanti. Notas de Aula – Disciplinas de Engenharia de Software e Projeto de Interfaces de Usuário. <<http://www.dimap.ufrn.br/~jair/pium>> (1999)
- [09] NIELSEN, Jakob & MACK, Robert L. Usability Inspection Methods. New York, NY: John Wiley & Sons, 1994.
- [10] KAHL, Michael & PRAIL, Amanda. Formal Usability Inspection. NIELSEN, Jakob & MACK, Robert L (ed.). Usability Inspection Methods. New York, NY: John Wiley & Sons, 1994, p. 141-169.
- [11] DE SOUZA, Clarisse Sieckenius; PRATES, Raquel Oliveira & BARBOSA, Simone.. A Method for Evaluating Software Communicability. In: C.J.P. de Lucena (ed.) Inf MCC11/1999. Computer Science Department, PUC-Rio.
- [12] MATIAS, Márcio; MASSARO Neto, Sílvio. & SANTOS, Néri dos. Uma Ferramenta de Apoio ao Registro da Interação Humano Computador In: 3RD WORKSHOP ON FACTORS IN COMPUTER SYSTEMS: MANY FACES IN INTERFACES, Gramado-RS, Brasil, 2000. Anais do 3RD WORKSHOP ON FACTORS IN COMPUTER SYSTEMS, p. 142-147.
- [13] BOOCH, Grady; RUMBAUGH, James & JACOBSON, Ivar. The Unified Modeling Language User Guide. SILVA, Fábio Freitas (org.) UML- Guia do Usuário. Rio de Janeiro: Editora Campus, 2000, 472 p.

- [14] BOOCH, Grady, RUMBAUGH, James & JACOBSON, Ivar. The Unified Modeling Language for Object-oriented Development. Documentation Set Version <<http://www.rational.com>> (1997)
- [15] LEITE, Jair Cavalcanti & DE SOUZA Clarisse Sieckenius. Uma Linguagem de Especificação para a Engenharia Semiótica de Interfaces de Usuário. In: II WORKSHOP SOBRE FATORES HUMANOS EM SISTEMAS COMPUTACIONAIS, Campinas-SP, Brasil, 1999. Anais Eletrônicos do II WORKSHOP SOBRE FATORES HUMANOS EM SISTEMAS COMPUTACIONAIS.
- [16] CARROLL, John M. Scenario Based-Design: Envisioning work and Technology in System Development. EUA, 1995.
- [17] CECHIN, Leonardo; GELATTI, Paola; NUNES, Isabel e TAVARES, Tatiana. Sistema para uma Corrida de Simulação. Pelotas-RS, 1998. Trabalho apresentado à Disciplina de Simulação Discreta. Universidade Federal de Pelotas, Instituto de Física e Matemática, Curso de Informática.
- [18] DIAS, Cláudia. Métodos de Avaliação de Usabilidade no Contexto de Portais Corporativos: um Estudo de Caso. Brasília-DF, 2001 (data prevista). Dissertação (Mestrado em Ciência da Informação). Universidade de Brasília.