

Geração de Interfaces Adaptativas para Dispositivos Móveis

Giani Carla Ito², Nilson Sant'anna², Maurício Ferreira²,
Douglas T. S. Finkler¹, André A. dos Santos¹, Marilson M. dos Santos¹

¹Laboratório de Computação Aplicada
Instituto Nacional de Pesquisas Espaciais (INPE) – São José dos Campos, SP
¹Departamento de Sistemas de Informação
Universidade Paranaense (UNIPAR) – Cascavel, PR – Brasil
{giito,nilson}@lac.inpe.br, mauricio@ccs.inpe.br, dtsfinkler@hotmail.com,
aasantos@pop.com.br, marilson_ms@hotmail.com

Resumo: Com o crescimento da necessidade de mobilidade, surgem novas maneiras de utilização do computador a qualquer hora e em qualquer lugar. A diminuição de tamanho, peso e energia consumida criam uma combinação de várias tecnologias no mesmo dispositivo, tornando os meios de ligação entre computadores cada vez mais flexíveis, onde recursos não necessitam de uma localização fixa e nem precisam estar fisicamente conectados para se comunicar. A proposta deste artigo é apresentar uma metodologia, bem como uma ferramenta, para a Geração de Interfaces Adaptativas (GIA) que têm como objetivo, a criação de interfaces de usuário que se ajustem às características de diversos tipos de dispositivos móveis.

Palavras-chave: *Web*, Interfaces Adaptativas, CC/PP, DELI, Dispositivos Móveis.

1. Introdução

Com o crescimento da variedade de dispositivos móveis e seu uso em ambientes totalmente dinâmicos por vários tipos de usuários, adaptar a interface a este novo contexto tornou-se uma necessidade. O desenvolvimento para dispositivos móveis envolve vários desafios, como ambientes heterogêneos, limitações físicas do aparelho, entre outras.

A reutilização de código é um aspecto fundamental neste contexto. O reuso de interfaces do usuário é complexo devido ao surgimento de novos modelos de aparelhos com características diversas. Para flexibilizar esta reutilização, é necessário criar e utilizar formas de desenvolvimento que sejam menos dependentes das propriedades de um dispositivo específico.

Para que uma página possa ser visualizada tanto em computadores pessoais como em dispositivos móveis, o desenvolvedor pode optar por uma página para o tipo determinado ou fazê-lo uma única vez de forma que a mesma possa ser adaptada e visualizada conforme a necessidade da tecnologia utilizada.

Na primeira abordagem, o esforço exigido por parte do desenvolvedor é grande, pois deve modelar uma página para cada dispositivo. Na segunda, a página é modelada somente uma única vez e adaptada conforme o contexto do aparelho que solicitar uma aplicação. Ao contrário da abordagem anterior, nesta ocorre uma redução no trabalho do *designer*, uma vez que não haverá a necessidade da geração de diversas interfaces. Este trabalho visa apresentar uma ferramenta para a Geração de Interfaces Adaptativas (GIA) que partindo de um modelo genérico tem como objetivo a criação de interfaces que se adaptem às características distintas de dispositivos móveis, retornando dinamicamente para cada cliente.

Utilizar-se-á uma forma de reconhecimento para dispositivos móveis proposta pela *World Wide Web Consortium (W3C)*, denominada *Composite Capability Preference Profile (CC/PP)* que tem a finalidade de proporcionar um mecanismo estruturado e universal para descrever e transmitir informações sobre as características de um dispositivo móvel para um servidor.

O artigo está organizado na seguinte ordem: a seção 2 descreve os objetivos, a seção 3 a revisão bibliográfica, a seção 4 material e métodos, a seção 5 apresenta os resultados e discussões e a seção 7 as conclusões.

2. Objetivos

Este trabalho visa apresentar a proposta de uma arquitetura que tem como objetivo a geração de interfaces de usuário que se adaptam às características de dispositivos móveis, partindo de uma descrição da interface genérica que seja independente de dispositivo e que atenda as solicitações de diversos usuários a uma mesma aplicação. Tem como meta retornar dinamicamente elementos da interface de acordo com as características distintas de cada dispositivo. Através deste processo a interface do usuário poderá se adaptar em tempo de execução, ao tipo de dispositivo móvel que solicitar o serviço, considerando a classe de dispositivos, o tipo de plataforma, aspectos de contexto do usuário entre outros fatores. A validação da arquitetura será realizada através da ferramenta GIA.

3. Revisão Bibliográfica

No campo das aplicações para unidades móveis, muito se tem discutido sobre mecanismos que permitam a elas se adaptarem às alterações do ambiente, dividindo a responsabilidade entre cliente e servidor.

(CONINX, et. al., 2003), apresentou a arquitetura *Dynamically Generating Interfaces for Mobile and Embedded Systems* (DIGYMES) composta por um *framework* que tem como finalidade a criação e design, de um sistema interativo para diversos tipos de dispositivos móveis e embutidos, utilizando o conceito de separação entre as camadas lógica e de apresentação. O autor incorpora várias técnicas de modelo de desenvolvimento de interface do usuário, linguagens como *eXtensible Markup Language* (XML), gerenciamento automático de *layout* e transparência de localização. O *design* começa com um modelo lógico, que uma vez criado, outros poderão estar relacionados ou gerados a partir dele. Uma das limitações do Dygimes é que o mesmo não garante uma apresentação visual agradável da interface perante a migração da mesma para dispositivos móveis, alegando que desta forma o sistema apresenta maior flexibilidade.

(MENKHAUS, 2002) desenvolveu o *Multi User Interfaces Single Application* (MUSA), que tem como objetivo a redução no tempo de desenvolvimento, custos, melhora da manutenção e flexibilidade de utilização, através da separação em camadas da interface, aplicação lógica e comunicação. O sistema MUSA suporta a adaptação envolvendo grande diversidade de dispositivos e usando várias estratégias de adaptação.

Os trabalhos estudados, MUSA descrito por (MENKHAUS, 2002) e DIGYMES por (CONINX, et. al., 2003), utilizaram o conceito de separação por camadas, fazendo uma clara distinção entre o modelo de implementação de cada nível: interface do usuário e modelo lógico. Ambos têm como objetivo a adaptação da interface do usuário no ambiente móvel, mas utilizam técnicas diferentes. Os autores argumentam uma forma de desenvolvimento que seja menos dependente de propriedades de um dispositivo único, justificando desta forma a necessidade do conceito de adaptação da interface do usuário.

As arquiteturas propostas apresentam-se limitadas no sentido de considerar aspectos como modelo lógico, usuários e modelos de plataformas,

deixando sem modelagem os aspectos contextuais como, por exemplo, a identificação em tempo de execução do tipo de dispositivo móvel.

4. Material e Métodos

4.1 Reconhecimento de Dispositivos Móveis

O *Composite Capabilities/Preferences Profile (CC/PP)* é uma especificação do W3C para expressar características dos dispositivos e das preferências de usuários. O perfil CC/PP é uma descrição das potencialidades do dispositivo e das preferências do usuário de acordo com (HANUMANSETTY, 2004). Tem como objetivo proporcionar um mecanismo estruturado e universal para descrever e transmitir informações sobre as capacidades de um cliente *Web* para um servidor, de forma que o conteúdo seja direcionado a essas características. Os perfis podem estar armazenados em um servidor *Web* local ou remoto, segundo (BUTLER, 2002). Cada forma de armazenamento apresenta vantagens e desvantagens, porém a utilização de ambos em uma mesma aplicação pode tornar o reconhecimento de dispositivos móveis mais rápido e eficaz.

De acordo com (SILA E SWICK, 1999), o CC/PP é baseado no *Resource Description Framework (RDF)*, uma linguagem utilizada pela W3C para modelagem de metadados e descrição de documentos XML, permitindo uma maior flexibilidade na criação de novos vocabulários. O perfil CC/PP, é construído em uma hierarquia de dois níveis, o primeiro chamado de componentes de um perfil e o segundo de atributos de um perfil. Um componente é composto de pelo menos um atributo, podendo ser de *hardware*, *software* ou *browser*. Como exemplo de *hardware* pode-se citar tamanho da tela, processador, memória, etc. De *software*, a versão do sistema operacional e de *browser* nome, versão, fabricante, etc.

As características dos dispositivos também poderão ser capturadas através do UAProf (WAP-UAProf), uma especificação definida pelo grupo *Wireless Application Protocol (WAP)*. Tem como objetivo descrever as características e padronizar os vocabulários CC/PP entre os fabricantes de dispositivos móveis. Cada fabricante disponibiliza um documento UAProf para a classe de aparelhos em seu *website*. Para que sejam tratadas as requisições de dispositivos que

possuam perfis CC/PP, existe uma biblioteca denominada *Delivery Context Library* (DELI) que foi desenvolvida pela HP (*Hewlett Packard*). (SILVA, RIBEIRO E ITO, 2005) descrevem um exemplo de uso e aplicação do DELI integrado ao CC/PP.

4.2 Arquitetura para Geração de Interfaces Adaptativas (GIA)

O ambiente da computação móvel cria a necessidade de uma maior flexibilidade do *software*, impondo novos desafios para a modelagem e desenvolvimento. Atualmente diversos estudos estão sendo realizados em busca de interfaces mais dinâmicas, que ajudem o usuário a realizar suas tarefas de maneira mais agradável e eficiente.

A geração de interfaces proposta pela arquitetura GIA utiliza a descrição de uma interface genérica que tem como finalidade principal atender as solicitações de diversos usuários a uma mesma aplicação considerando o tipo de dispositivo móvel para responder devidamente a solicitação em tempo de execução.

Outro fator de destaque é permitir uma implementação que esteja voltada a um ambiente multi-plataforma, ampliando assim, a relação um para um, para uma relação um para n, ou seja, uma aplicação pode gerar várias interfaces.

Como principais vantagens da utilização da arquitetura GIA podem ser ressaltados o processo de adaptação da interface ao tipo de dispositivo acontecerá em tempo de execução, a facilidade para o processo de desenvolvimento de interfaces, pois o desenvolvedor criará uma única interface que será apresentada em múltiplos dispositivos, o designer da interface e o desenvolvedor poderão trabalhar separadamente no projeto de um sistema, o reconhecimento automático do dispositivo através do CC/PP que está acessando a aplicação, a arquitetura será utilizada através de um editor gráfico que fornecerá um ambiente de desenvolvimento com visualização do código em tamanhos de telas escolhidos pelo desenvolvedor, bem como, um gerenciador dos metadados.

A figura 1 ilustra a arquitetura GIA composta por dois níveis: cliente e servidor. No lado cliente, usuários podem fazer solicitações de serviços munidos de diversos tipos de PDA's dentre eles *Palms*, *PocketPCs* ou *Smartphones*. No servidor estão localizados o interceptador, o *framework* DELI, o redirecionador de regiões, o gerador de componentes, o *handler* e o banco de dados.

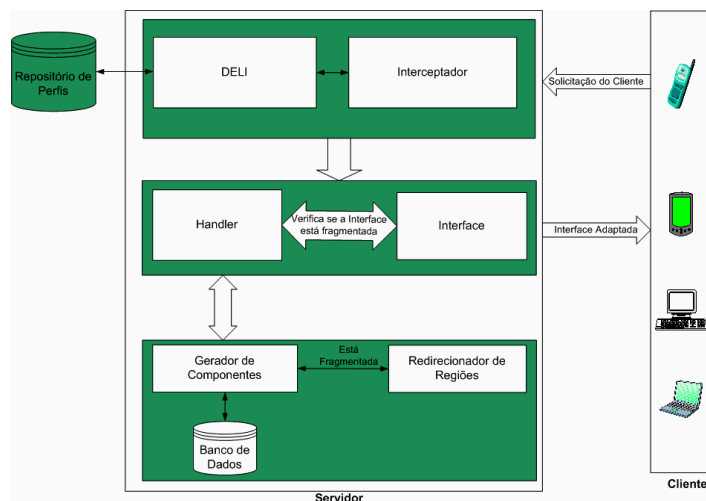


FIGURA 1 - Arquitetura para Geração de Interfaces Adaptativas

Para exemplificar um cenário de utilização da arquitetura GIA, supõe-se que o usuário faz uma solicitação à aplicação descrita em *Java Server Pages* (JSP) através de um dispositivo móvel ou computador pessoal. O interceptador captura-a e faz uma chamada ao DELI, para identificar o dispositivo e localizar no repositório de perfis externo as informações relevantes, tais como, o tamanho de tela, *browser*, sistema operacional, etc.

Após este processo, o interceptador envia os parâmetros para o *handler* que por sua vez ativará o gerador, o qual usará as informações dos componentes armazenados em banco de dados para converter as *tag-libs* declaradas no código em uma linguagem que o *browser* do dispositivo interprete. Caso a interface esteja fragmentada o redirecionador inserirá na página o algoritmo que possibilitará a navegação entre as regiões. Após, a interface será exibida no dispositivo solicitante.

4.3 Metodologia de Desenvolvimento utilizando a Ferramenta GIA

A ferramenta de desenvolvimento GIA tem a finalidade de proporcionar um ambiente para a criação de interfaces adaptativas, conforme a arquitetura proposta. Foi implementada na linguagem *Object Pascal* através do *Borland Delphi 7*. Além do ambiente, possui um gerenciador que possibilita a manutenção dos dados cadastrados em banco.

Ao iniciar a construção da página o desenvolvedor deverá mapear a interface seguindo uma técnica de programação que propõe a divisão do código em regiões, de forma hierárquica, mantendo uma relação lógica entre elas de

acordo com a funcionalidade atribuída, que tem por objetivo a intercalação do código.

Cada região poderá ser composta por sub-regiões e também por componentes visuais como textos, botões, *links*, *combos*, imagens, etc. Foram determinados 9 tamanhos de região, numa escala aritmética de 100 a 500 *pixels* com coeficiente de variação de 50. Para determinar o tamanho mínimo e máximo de uma região analisou-se que o menor tamanho da tela dos dispositivos móveis, incluindo celulares e PDA's, aproxima-se de 100 *pixels* e o maior de 500 *pixels*. Caso o tamanho da tela do dispositivo ultrapasse 500 *pixels* a interface não será fragmentada, sendo mostrada na íntegra. Para viabilizar o processo de adaptação, foi estabelecido que seja mostrada a menor região que mais se aproxima da resolução do aparelho.

A ferramenta GIA permite utilizar durante o processo de desenvolvimento uma forma de reduzir o conteúdo que será visualizado no dispositivo móvel, pois as restrições relacionadas ao tamanho da tela do aparelho tornam necessárias em alguns casos, a classificação das informações. A seleção do conteúdo que será exibido é determinada de acordo com as delimitações das regiões. Caso não seja delimitado o início e o fim de uma região, o conteúdo não será exibido quando a fragmentação for utilizada.

Durante o processo de desenvolvimento, a interface poderá ser visualizada de forma completa ou fragmentada, como ilustra a figura 2.

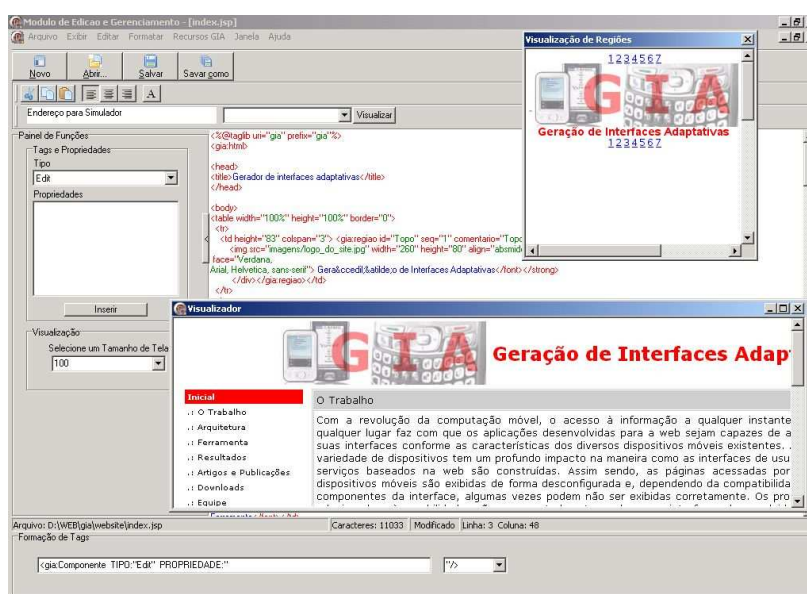


FIGURA 2 – Formas de visualização da interface na Ferramenta GIA

5. Resultados e Discussões

A linguagem de programação que será utilizada para o desenvolvimento das interfaces na ferramenta GIA é a JSP, uma especificação da *Sun Microsystems* que combina Java e HTML com o objetivo de fornecer conteúdo dinâmico para páginas *Web*. Permite a inserção nas páginas JSP de conteúdo HTML, proporcionando agilidade no processo de implementação, já que *designers* e desenvolvedores podem trabalhar separadamente.

Para demonstração do estudo de caso, foi elaborado um *Web site* utilizando a ferramenta proposta. Para tal foi configurado o servidor *Web Apache TomCat 5.0*, o *Internet Explorer 6.0* da máquina local e os simuladores *Openwave Phone Simulator*, de celular e *Palm Os Simulator 6.0.1*, da *Palm*. A figura 3 exibe uma interface no navegador *Internet Explorer 6.0* na resolução de 1024x768 *pixels*.

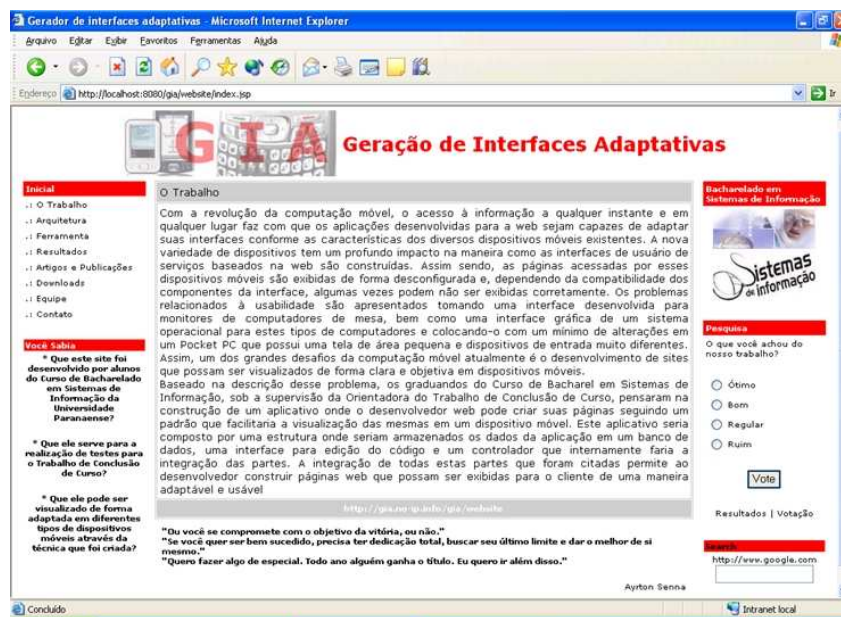


FIGURA 3 – Interface visualizada no browser Internet Explorer 6.0

A partir da captura de telas do *Palm Os Simulator 6.0.1*, observa-se na figura 4 que a interface também se adaptou as características do dispositivo. Devido à tela do *Palm* ser maior em relação a do *Openwave*, o conteúdo visualizado na tela é maior, o que resulta na menor rolagem da barra horizontal. O *handler* inseriu no topo e no final de cada região um menu que possibilita a navegação entre as telas.



FIGURA 4 – Interface visualizada no *Palm Os Simulator 6.0.1*

Acessando esta mesma interface através do simulador *Openwave Phone Simulator*, observa-se através da figura 5 que o *site* foi adaptado às características do dispositivo, pois não gerou barra horizontal e o conteúdo pôde ser visualizado de uma maneira clara e objetiva. Como no *Openwave*, o *handler* inseriu no topo e no final de cada região um menu que possibilita a navegação entre as telas.



FIGURA 5 – Interface visualizada no *Openwave Phone Simulator*

6. Conclusões

Neste trabalho apresentou-se uma Arquitetura para Geração de Interfaces Adaptativas, com o intuito de contribuir para uma maior facilidade de desenvolvimento no ambiente móvel.

Como principais vantagens da utilização da arquitetura GIA destacam-se o processo de adaptação da interface ao tipo de dispositivo móvel em tempo de execução, bem como a facilidade de desenvolvimento, pois um único código poderá ser visualizado em múltiplos dispositivos.

Os resultados obtidos através da implementação de estudos de caso apresentaram-se satisfatórios no contexto da adaptação de interfaces *Web* em um ambiente de desenvolvimento. Para tal, utilizou-se uma técnica de fragmentação do código em regiões que por meio da arquitetura apresentada serão adaptadas de acordo com o tipo de dispositivo que realiza uma requisição *Hiper Text Transfer Protocol* (HTTP).

Referências Bibliográficas

BUTLER, M. H.; DELI: **A Delivery Context Library for CC/PP and UAProf**, 2002. Disponível em: <<http://www.hpl.hp.com/personal/marbut/DeliUserGuideWEB.htm>>. Acesso em: 19 mar. 2006.

CONINX, K.; LUYTEN, K.; VANDERVELPEN, C.; VAN DEN BERGH, J.; and CREEMERS, B.; **Dygimes: Dynamically Generating Interfaces for Mobile Computing Devices and Embedded Systems**. In Chittaro, L., editor, *Mobile HCI*, volume 2795 of *Lecture Notes in Computer Science*, pages 256--270. Springer, 2003.

HANUMANSETTY, G. R.; **Model Based Approach for Context Aware and Adaptive User Interface Generation**. Master's Thesis, Virginia Polytechnic Institute and State University, Falls Church, Virginia, USA, 2004.

MENKHAUS, G.; **Adaptive User Interface Generation in a Mobile Computing Environment**. PhD Thesis, University of Salzburg, Austria, 2002.

SILA, O.; SWICK, R.; **Resource Description Framework (RDF) Model and Syntax Specification**, 1999. Disponível em: <<http://www.w3.org/1999/status/REC-rdf-syntax-19990222/status>>. Acesso em: 13 jun. 2005.

SILVA, A. C. B.; RIBEIRO, N. N. T. J.; ITO, G. C.; **Análise de um Repositório de Perfis para Reconhecimento de Dispositivos Móveis**. Anais, XII Simpep, 2005.