

UM AMBIENTE DE ENSINO PARA COMPUTAÇÃO MÓVEL INTEGRANDO J2ME E COMPONENTES DE SOFTWARE

Ivan João Foschini¹ e Sérgio Donizetti Zorzo²

Resumo — Este artigo descreve a implementação de um aplicativo que possibilita a criação de conteúdo didático para ambientes de Computação Móvel, oferecendo mobilidade aos estudantes no processo educacional. Embora parte deste aplicativo tenha sido desenvolvida para ser utilizada em computadores de mesa convencionais, o conteúdo didático gerado pela aplicação é direcionado para ser visualizado em dispositivos que utilizam a Comunicação Sem Fio para acessar a Rede Mundial de Computadores, notadamente os Palmtops. Com isso, deseja-se que os usuários destes dispositivos possam obter material didático em qualquer lugar onde estejam, usufruindo das vantagens inerentes às Redes de Comunicação Sem Fio, tais como independência de uma conexão cabeada convencional e mobilidade. O ambiente descrito foi implementado utilizando-se as técnicas do Desenvolvimento de Software Baseado em Componentes, explorando este método de desenvolvimento em conjunto com a Java 2 Micro Edition, versão da tecnologia Java voltada especificamente para a criação de aplicações para dispositivos móveis.

Palavras-Chave — Componentes de Software, Computação Móvel, Educação a Distância, Java 2 Micro Edition, Mobilidade dos Estudantes

INTRODUÇÃO

A Educação a Distância (EAD) é uma modalidade de ensino consagrada e consolidada em diversas partes do mundo [1]. Há tempos, educadores e outros profissionais perceberam as vantagens de se oferecer cursos a distância, e sentiram a necessidade de utilizar novas tecnologias no processo educacional. Quanto às vantagens dos cursos a distância, podem-se destacar: os aspectos econômicos, já que os estudantes não precisam ter custos com viagens e hospedagem, a possibilidade de o aluno poder estudar nos momentos que melhor lhe convierem e o fato de cada um poder seguir seu próprio ritmo de aprendizado, entre outros fatores.

As novas tecnologias, principalmente os computadores e as redes de computadores, ampliaram ainda mais as possibilidades e as vantagens da Educação a Distância. As redes de computadores, notadamente a World Wide Web, tornam possível que qualquer pessoa, em qualquer parte do mundo onde seja possível acessar a rede mundial de

computadores, tenha fácil acesso a uma grande quantidade de conteúdo didático.

A utilização das tecnologias de comunicação sem fio e de computação móvel ampliam ainda mais os horizontes das aplicações de Educação a Distância, uma vez que, ao se utilizar meios de comunicação sem fio, o único limitante para a distribuição de conteúdo didático passa a ser a área de cobertura de uma Rede Sem Fio, e não mais a existência de uma conexão cabeada convencional com uma rede de computadores.

O objetivo principal deste artigo é apresentar um ambiente, baseado em componentes de software, que possibilita a criação de conteúdo educacional de forma rápida, eficiente e intuitiva. Este material didático, no entanto, não é direcionado para ser utilizado em microcomputadores de mesa ou quaisquer outros dispositivos semelhantes. A preocupação é criar-se conteúdo educacional que possa ser visualizado em pequenos dispositivos dedicados, com tela pequena e características arquiteturais próprias, especificamente os Palmtops.

Dispositivos como este estão capacitados, atualmente, a receber dados da World Wide Web através de um acesso sem fio, o que possibilita a seus usuários receber o material didático onde quer que se encontrem, independentemente de possuírem a sua disposição uma rede de computadores convencional ou não.

O ambiente proposto neste artigo conta com dois módulos: o Módulo do Professor e o Módulo do Aluno. O Módulo do Professor foi desenvolvido para ser utilizado em *desktops*, e permite que os professores criem listas de exercícios com questões de verdadeiro ou falso e de múltipla escolha, e as disponibilizem em Servidores Web. O Módulo do Professor também permite a geração de relatórios de desempenho individuais e coletivos dos alunos na resolução das listas de exercícios geradas, o que oferece subsídios para que o professor possa aprimorar o processo educacional.

O Módulo do Aluno, por sua vez, permite que os alunos acessem as listas de exercícios geradas e as resolvam em seus Palmtops. O ambiente se encarrega de fazer uma correção automática das resoluções dos alunos, bem como de gerar um relatório a respeito do desempenho dos estudantes na resolução destas listas.

Este artigo descreverá brevemente a Java 2 Micro Edition, os Componentes de Software e apresentará uma Descrição Informal, a Arquitetura e a Interface do ambiente

¹ Ivan João Foschini, Universidade Federal de São Carlos, Rodovia Washington Luís (SP-310) – Km 235, 13.565-905, São Carlos, SP, Brasil, ivan@dc.ufscar.br

² Sérgio Donizetti Zorzo, Universidade Federal de São Carlos, Rodovia Washington Luís (SP-310) – Km 235, 13.565-905, São Carlos, SP, Brasil, zorzo@dc.ufscar.br

proposto, expondo ainda um exemplo da criação de uma lista de exercícios e da sua resolução por parte dos alunos, bem como da geração dos relatórios de desempenho.

JAVA 2 MICRO EDITION

O ambiente descrito conta com dois módulos: o Módulo do Professor e o Módulo do Aluno. O Módulo do Professor permite ao educador criar listas de exercícios para avaliação dos alunos e, posteriormente, levantar dados estatísticos a respeito do desempenho dos estudantes. Este módulo é executado em computadores de mesa tradicionais (*desktops*). Já o Módulo do Aluno, desenvolvido com o auxílio da tecnologia Java 2 Micro Edition (J2ME), é executado em Palmtops. Esta seção apresenta esta tecnologia, mostrando as suas principais características. A J2ME foi criada para o desenvolvimento de aplicações a serem exibidas em dispositivos pequenos, tais como Palmtops e telefones celulares, que possuem pequenos computadores dedicados e que freqüentemente utilizam baterias como fonte de energia.

A primeira característica que se destaca nesta tecnologia é que as suas aplicações provêm interfaces gráficas de qualidade, com gráficos coloridos e diversos componentes de interface, o que facilita a interação dos usuários com as aplicações [2]. O fato de a J2ME oferecer interfaces gráficas de qualidade pode vir a resolver um dos maiores problemas práticos encontrados por desenvolvedores de aplicações para dispositivos de tela pequena: criar projetos que realmente atraiam a atenção dos usuários, acostumados a aplicativos multimídia em seus *desktops*.

Outra característica importante da J2ME é que suas aplicações podem comunicar-se com *Servlets*, o que permite que estes aplicativos acessem Servidores Web e bancos de dados remotos através da World Wide Web [3]. *Servlets* são classes baseadas em tecnologia Java que são executadas e instanciadas em associação com Servidores Web, atendendo a requisições realizadas por meio do protocolo HyperText Transfer Protocol (HTTP). Com isso, as aplicações móveis passam a ter muito mais alcance e tornam-se tão poderosas quanto aquelas desenvolvidas para computadores convencionais.

COMPONENTES DE SOFTWARE

O desenvolvimento de componentes de software, ou, simplesmente, componentes vem ganhando notoriedade nos estudos de Engenharia de Software, por oferecer, principalmente, grande possibilidade de reuso de código e redução nos custos de implementação. Os componentes são, basicamente, trechos de código previamente testados, que podem ser instalados e executados em diversas aplicações distintas [4]. Esta propriedade, aliada ao fato de que um componente deve ser tão genérico quanto possível, faz com que o desenvolvedor poupe tempo e recursos de implementação. Atualmente, existem desenvolvedores que

se dedicam exclusivamente à criação de componentes, que posteriormente podem ser comprados e utilizados pelos mais diversos usuários, de acordo com as necessidades individuais de cada um. Um componente oferece serviços aos usuários através da sua interface. A interface de um componente é composta por um conjunto de operações que determinam quais são os serviços que um determinado componente pode oferecer.

Independentemente da metodologia adotada para o desenvolvimento de componentes de software, os desenvolvedores devem ter em mente alguns princípios gerais que tornam o uso de componentes realmente atrativos:

- um componente deve ser simples de ser compreendido, para que possa ser facilmente implementado;
- um componente deve ser genérico, para que possa ser reutilizado em um grande número de aplicações com propósitos distintos;
- quando se desenvolve um conjunto de componentes independentes, estes devem ser facilmente combináveis, para facilitar a montagem da aplicação;
- embora devam ser facilmente combináveis, diferentes componentes devem ter suas dependências minimizadas ou, se possível, evitadas;
- deve haver uma preocupação em se criar componentes que possam ser facilmente estendidos, para que novos serviços possam ser adicionados a eles, em caso de necessidades futuras;
- os componentes devem ser portáteis para diversas plataformas distintas, o que aumenta ainda mais as possibilidades de reuso.

O ambiente descrito neste artigo foi implementado utilizando componentes de software tanto no desenvolvimento do Módulo do Professor quanto no Módulo do Aluno. Com isso, exploram-se as vantagens deste método de desenvolvimento de aplicações, expostas nesta seção.

DESCRIÇÃO INFORMAL DO AMBIENTE

Os dois atores que interagem com o ambiente descrito neste artigo são o Professor e o Aluno. O Professor é identificado pelo seu nome, pela disciplina que ministra, pelo código desta disciplina e pela turma pela qual é responsável. O ambiente proporciona ao Professor a criação de uma ou mais Listas de Exercícios, que possuem os seguintes atributos: um código para identificação da lista, o seu título, a sua data de criação e a sua data de entrega. As Listas são compostas por uma ou mais Questões.

Cada Questão possui um número único para identificá-la e é de um determinado tipo (Questão de Verdadeiro ou Falso ou Questão de Múltipla Escolha). As Questões de Verdadeiro ou Falso têm como atributos o texto da questão, uma justificativa que explica qual é a razão do texto da questão ser verdadeiro ou falso, e qual é a resposta correta à questão (verdadeiro ou falso). Já as Questões de Múltipla

Escolha, por sua vez, têm como atributos o texto da questão, uma justificativa que explica porque determinada alternativa é considerada correta, os textos das cinco alternativas a serem apresentadas ao Aluno e a resposta correta da Questão (alternativa A, B, C, D ou E).

Os Alunos são identificados pelo seu código de aluno e pelo seu nome. O Aluno resolve uma Lista de Exercícios, gerando, com isso, uma Resolução, que contém o seu índice de acertos e um relatório a respeito do seu desempenho, indicando, por exemplo, quais foram as questões respondidas corretamente, quais foram respondidas incorretamente e uma justificativa para o fato de a resposta dada à questão ter sido considerada incorreta.

O ambiente também fornece ao professor estatísticas relativas ao desempenho individual e coletivo dos alunos na resolução das Listas de Exercícios, o que oferece subsídios que auxiliam o professor na tarefa de aperfeiçoar o processo educacional.

A ARQUITETURA DO AMBIENTE

Baseado na Descrição Informal da aplicação, definiu-se, para o ambiente descrito neste artigo, a arquitetura exibida na Figura 1:

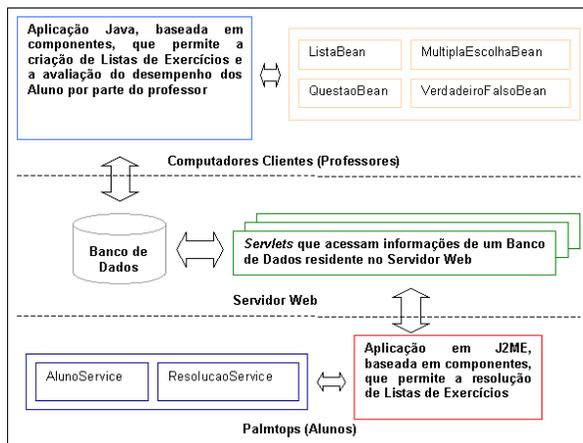


FIGURA. 1

ARQUITETURA DO AMBIENTE DESCRITO, UTILIZANDO COMPONENTES DE SOFTWARE.

O Módulo do Professor previamente mencionado baseia-se em *JavaBeans*, e permite que professores criem listas de exercícios, contendo questões de verdadeiro ou falso e de múltipla escolha. A tecnologia *JavaBeans* foi responsável pela introdução do conceito de componentes em Java. Utilizando-se *JavaBeans*, pode-se criar componentes de software reutilizáveis e independentes de plataforma [5]. Esta aplicação conta com os seguintes *JavaBeans*: *ListaBean*, *QuestaoBean*, *MultiplaEscolhaBean* e *VerdadeiroFalsoBean*. Cada um destes *JavaBeans* possui alguns atributos específicos e métodos para inserir, alterar, recuperar e excluir informações de um banco de dados que armazena as listas de exercícios criadas pelos professores.

Estas listas de exercícios ficam disponíveis em um banco de dados localizado em um Servidor Web remoto, no qual também residem *Servlets* que podem recuperar, alterar e armazenar informações neste banco de dados.

A arquitetura da Figura 1 exhibe ainda o Módulo do Aluno, desenvolvido em J2ME, que faz uso de componentes de software especialmente desenvolvidos para esta tecnologia, o *AlunoService* e o *ResolucaoService*. Na tecnologia J2ME, os componentes de software são chamados de *Services* [6]. O Módulo do Aluno permite que os alunos resolvam as listas de exercícios previamente criadas pelos professores. A aplicação J2ME envia os dados da resolução aos *Servlets* residentes no Servidor Web. Os *Servlets*, por sua vez, recebem estas informações e as armazenam no banco de dados do Servidor Web, deixando-as à disposição do professor. Com isso, o professor pode fazer uma avaliação do desempenho individual e coletivo dos alunos na resolução de uma determinada lista de exercícios, utilizando outra funcionalidade do Módulo do Professor.

A INTERFACE DO AMBIENTE

Esta seção apresenta a interface do ambiente descrito neste artigo, exibindo segmentos tanto do Módulo do Professor quanto do Módulo do Aluno. Quando o professor acessa o ambiente, é apresentada a ele a tela exposta na Figura 2:

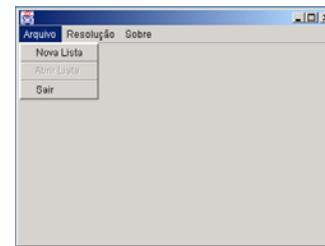


FIGURA. 2

TELA PRINCIPAL DO AMBIENTE.

Conforme pode ser visto na Figura 2, algumas opções são apresentadas ao professor, como, por exemplo, criar uma nova lista (opção “Nova Lista” no Menu “Arquivo” da Figura 2). Quando optar por criar uma nova lista de exercícios, o professor é conduzido à tela apresentada na Figura 3:



FIGURA. 3

TELA PARA A CRIAÇÃO DE UMA NOVA LISTA DE EXERCÍCIOS.

Nesta tela o professor fornece os dados relevantes para a criação de uma nova lista de exercícios. Clicando no botão “Criar Lista” uma nova lista de exercícios é criada. A partir daí, o professor pode começar a trabalhar na elaboração das questões que irão compor a lista recém-criada. Se optar por incluir na lista uma questão de verdadeiro ou falso, a tela da Figura 4 será exibida.

O professor deverá escrever o texto da questão e uma justificativa sobre a razão do texto da questão ser verdadeiro ou falso. Esta justificativa será apresentada ao aluno quando este responder de forma incorreta a uma questão. O professor deverá assinalar ainda qual é a resposta correta (verdadeiro ou falso). Clicando no botão “Incluir Questão”, esta questão será imediatamente incluída na lista de exercícios. Caso o professor deseje criar uma questão de múltipla escolha, a tela a ser exibida será a da Figura 5, e o procedimento para a criação de uma questão desse tipo será praticamente análogo ao adotado para a criação de uma questão de verdadeiro ou falso.

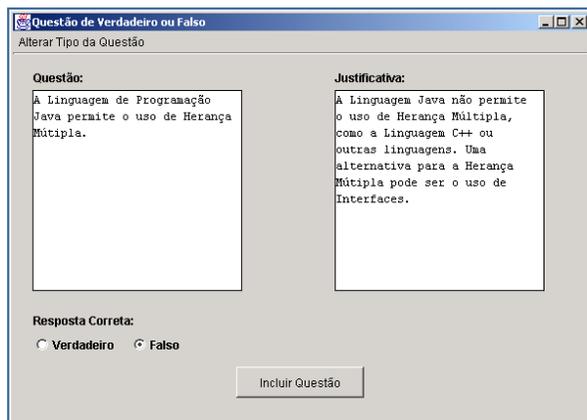


FIGURA. 4

TELA PARA A CRIAÇÃO DE QUESTÕES DE VERDADEIRO OU FALSO.

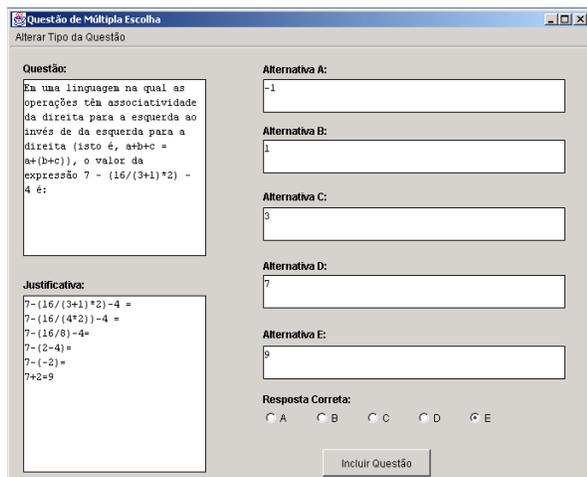


FIGURA. 5

TELA PARA A CRIAÇÃO DE QUESTÕES DE MÚLTIPLA ESCOLHA.

As listas de exercícios criadas com o auxílio deste ambiente ficam armazenadas em um banco de dados residente em um Servidor Web remoto, acessível para os professores que utilizam o ambiente. Todo o processo de armazenamento, consulta, remoção e edição de listas de exercícios é feita com o auxílio do ambiente, que, para isso, utiliza-se de componentes *JavaBeans*. No Servidor Web onde ficam armazenadas as listas de exercícios também residem *Servlets*.

Conforme foi dito anteriormente, aplicações J2ME podem se comunicar com *Servlets*, o que torna possível que essas aplicações acessem bancos de dados remotos na World Wide Web. No caso do ambiente descrito neste artigo, a aplicação J2ME correspondente ao Módulo do Aluno se comunica com *Servlets* que, por sua vez, fazem acesso aos dados das listas de exercícios geradas pelos professores. Estes *Servlets* também são responsáveis por corrigir e por armazenar os dados relativos à resolução dos alunos no Servidor Web, de tal maneira que os professores possam acessar estas resoluções em seus *desktops*, o que lhes permite avaliar o desempenho dos alunos. Dessa forma, o ambiente também permite que os professores possam fazer um acompanhamento e uma avaliação do desempenho dos alunos no processo de resolução dos exercícios. A Figura 6 exhibe exemplos de telas dos Palmtops dos alunos acessando uma lista de exercícios:



FIGURA. 6

EXEMPLOS DE TELAS DOS ALUNOS ACESSANDO OS EXERCÍCIOS.

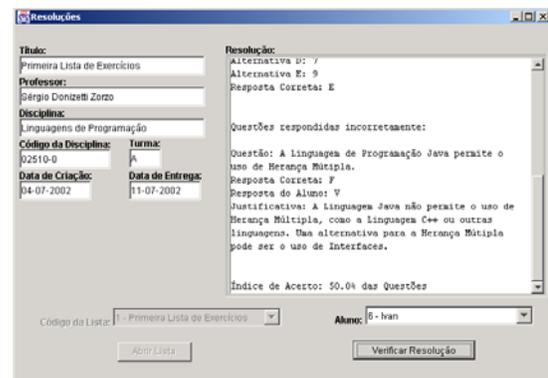


FIGURA. 7

RELATÓRIO DE DESEMPENHO INDIVIDUAL EXIBIDO A UM PROFESSOR.

A Figura 7, por sua vez, mostra a tela que permite ao professor analisar o desempenho individual de um aluno na resolução de uma lista de exercícios, enquanto que a tela exibida na Figura 8 corresponde à interface que possibilita ao professor fazer uma análise coletiva de desempenho.



FIGURA. 8

RELATÓRIO DE DESEMPENHO COLETIVO EXIBIDO A UM PROFESSOR.

Conforme mencionado anteriormente, os *Servlets* desenvolvidos para este ambiente também se encarregam de fazer a correção automática das listas de exercícios e de gerar um relatório de desempenho que será visualizado pelos alunos em seus Palmtops. Este relatório traz o índice de acerto de um aluno na resolução de uma determinada lista de exercícios e também todas as questões respondidas corretamente e incorretamente pelo aluno, bem como as respostas corretas de cada questão e uma justificativa que explica porque uma determinada resposta é considerada correta. A Figura 9 exibe um exemplo de relatório visualizado por um aluno em seu Palmtop.



FIGURA. 9

RELATÓRIO DE DESEMPENHO DE UM ALUNO EXIBIDO EM UM PALMTOPO.

Com isso, o ambiente auxilia o aluno no processo de aprendizado, uma vez que os estudantes não apenas respondem às questões, mas também podem verificar quais questões erraram e quais são as razões pelas quais a sua resposta foi considerada incorreta.

CONCLUSÕES

O ambiente descrito neste artigo tem como principal objetivo possibilitar a criação simples e intuitiva de listas de

exercícios e é composto por dois módulos distintos: o Módulo do Professor e o Módulo do Aluno. Enquanto o Módulo do Professor é executado em computadores de mesa convencionais, o Módulo do Aluno é direcionado para ser utilizado em Palmtops, e foi desenvolvido com a tecnologia Java 2 Micro Edition (J2ME), também exposta neste artigo. Com isso, expande-se ainda mais a ubiquidade oferecida pela Educação a Distância (EAD), que tem como uma das suas principais características levar conhecimento para qualquer pessoa, independentemente do lugar onde esta se encontra.

Utilizando-se Palmtops, dispositivos pequenos e capazes de acessar dados da World Wide Web em ambientes sem fio, pessoas interessadas em acessar conteúdo didático poderão fazê-lo facilmente, sem a necessidade de estarem conectadas a redes de computadores convencionais. Com a utilização da tecnologia J2ME, aplicações criadas para dispositivos dedicados adquirem qualidade comparável a aplicativos desenvolvidos para *desktops*.

O processo de criação do ambiente é apoiado pelas técnicas do Desenvolvimento de Software Baseado em Componentes, tanto na aplicação desenvolvida para *desktops* quanto na aplicação para Palmtops. Detalhes relativos a esta técnica também foram expostos neste artigo. Tanto os componentes desenvolvidos para *desktops*, baseados na tecnologia *JavaBeans*, como aqueles desenvolvidos para Palmtops, os chamados *Services*, são facilmente extensíveis e podem ser reutilizados por outras aplicações de EAD.

Foi formulada, ainda, uma descrição informal do ambiente descrito, visando especificá-lo claramente antes de se elaborar uma arquitetura para este ambiente. Esta arquitetura também foi apresentada neste artigo. Por fim, expôs-se a interface dos Módulos do Aluno e do Professor, mostrando-se um exemplo do processo de criação de uma lista de exercícios e da resolução desta lista por parte do aluno. Foram mostradas também as interfaces do ambiente que permitem ao professor avaliar o desempenho individual e coletivo dos alunos na resolução das listas de exercícios criadas.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] KEEGAN, D, "Foundations of Distance Education", *Livro*, 1996, 240 p.
- [2] MUCHOW, J, "Core J2ME Technology and MIDP", *Livro*, 2002, 710 p.
- [3] GUIGERE, E, "J2ME Tech Tips – Client-Server Communication Over HTTP Using MIDP and Servlets", URL: www.microjava.com/developer/resource_articles. Consultado em 13/11/2002.
- [4] PRADO, A, F, "Desenvolvimento de Software Baseado em Componentes (DBC)", URL: <http://www.recope.dc.ufscar.br/engenhariadesoftware/principal>. Consultado em 13/11/2002.
- [5] QUINN, A, "Trail: JavaBeans", URL: <http://java.sun.com/docs/books/tutorial/javabeans/>. Consultado em 13/11/2002.
- [6] SUN MICROSYSTEMS, "How To Write Your First JES Service", URL: <http://java.sun.com/j2me/docs/pdf/tutorialsol.pdf>. Consultado em 10/04/2002.