

# SISTEMA MÓVEL PARA ACOMPANHAMENTO DE CURSOS

**José Celso Freire Junior<sup>1</sup>; Pedro Lima Catalani<sup>2</sup>; Agnelo Marotta Cassula<sup>3</sup>**

Universidade Estadual Paulista, Faculdade de Engenharia de Guaratinguetá, Depto. de Engenharia Elétrica

Av. Ariberto Pereira da Cunha, 333

12.516-410 – Guaratinguetá – SP

<sup>1</sup> Jose-Celso.Freire@feg.unesp.br

<sup>2</sup>plcatalani@superig.com.br

<sup>3</sup>agnelo@feg.unesp.br

**Resumo:** *Este trabalho apresenta um sistema que pode ser executado em dispositivos móveis como PDAs (Personal Digital Assistants) para o acompanhamento do desenvolvimento de cursos e registro de presenças. O sistema foi desenvolvido em JME (Java Micro Edition) com a utilização das APIs PP/CDC (Personal Profile/Connected Device Configuration). O sistema desenvolvido fornece uma ferramenta que possibilita que os professores de uma instituição de ensino, de maneira rápida e prática, realizem a gestão de um curso com o registro dos tópicos ministrados em uma disciplina e também que o professor registre a presença (ou ausência) dos alunos à aula. Elimina-se, assim, a necessidade atual de preenchimento manual de tabelas, fazendo com que a disponibilização dos dados na Web para os alunos seja realizada com maior rapidez.*

**Palavras-chave:** *Gestão de Cursos, Dispositivos Móveis, Tecnologia Java, Java Micro Edition.*

## 1. INTRODUÇÃO

Uma grande vantagem do avanço tecnológico da atualidade é facilitar os processos diários em nossas vidas. A tecnologia móvel é um desses avanços caracterizando-se talvez principalmente pela flexibilidade. A tecnologia móvel mais que uma invenção, pode ser considerada uma revolução, pois está sendo capaz de atingir o cotidiano das pessoas e fazer parte de suas vidas, modificando suas rotinas e formas de tomar decisões. A mobilidade que inicialmente surgiu como uma facilidade, atualmente passou a desempenhar papel preponderante na vida das pessoas. A evolução deste segmento foi tão rápida em função deste tipo de tecnologia permitir o acesso a dados e informações em qualquer momento e em qualquer lugar. Isto se tornou um poderoso atrativo.

Procurando aproveitar esta tendência tecnológica, desenvolveu-se um sistema que visa facilitar a vida acadêmica dos professores e agilizar o acesso aos dados pelos estudantes. O sistema desenvolvido se integra ao Sistema WEB de Avaliação e Acompanhamento Acadêmico (SWA3) desenvolvido por FREIRE JUNIOR, *et. al.* (2006). O SW3A tem como objetivo permitir que instituições de ensino possam realizar atividades de gestão de aprendizado, tais como a aplicação de processos de avaliação das atividades de ensino e o acompanhamento pedagógico de seus cursos com a publicação e recuperação de dados relativos aos planos de ensino, avaliações e controle de presença das disciplinas oferecidas.

Estas características deverão auxiliar os Conselhos de Curso<sup>1</sup> no desenvolvimento de atividades de avaliação e planejamento. O sistema foi desenvolvido para ser disponibilizado sob licença pública do tipo GPL e oferece funcionalidades não encontradas normalmente em Sistemas de Gestão de Aprendizagem. O trabalho aqui apresentado pode ser considerado um módulo do SWA3.

O funcionamento deste módulo se inicia quando o professor conecta o PDA à rede da faculdade para que os dados pertinentes aos cursos por ele ministrados bem como os alunos inscritos nesses cursos sejam obtidos. Em seguida, o professor registra o tópico apresentado na aula e utilizando o PDA realiza a chamada dos alunos na sala de aula através de uma interface gráfica amigável. Na seqüência, o professor deve descarregar as informações em um servidor através novamente na rede da faculdade. Isso facilita e agiliza o processo de acompanhamento e gestão dos cursos.

O sistema foi desenvolvido como uma aplicação JME (SUN MICROSYSTEMS INC., 2008) e utilizou-se o conjunto de APIs PP/CDC (*Personal Profile/Connected Device Configuration*), já que este fornece um ambiente de aplicações Java para dispositivos que requerem a utilização da biblioteca gráfica AWT (SUN MICROSYSTEMS INC., 2008a) completa. Esta biblioteca foi utilizada por oferecer um conjunto razoável de componentes gráficos.

O trabalho está organizado do seguinte modo: na Seção 2 são apresentadas e explicadas algumas das tecnologias envolvidas no projeto, possibilitando uma visão geral das alternativas tecnológicas adotadas para sua construção. Na Seção 3 apresenta-se o sistema desenvolvido através da exibição de figuras da interface gráfica, mostrando as diversas possibilidades de utilização. A Seção 4 destaca os pontos principais observados durante o desenvolvimento do trabalho e oferece idéias para possíveis melhorias do sistema.

## 2. TECNOLOGIAS ENVOLVIDAS

Nesta seção é feita uma apresentação resumida de algumas das principais tecnologias utilizadas para o desenvolvimento do sistema. Além das listadas abaixo, deve-se ressaltar que o sistema se integra a um Sistema Gerenciador de Banco de Dados (SGBD).

### 2.1 JME

A plataforma Java Micro Edition (JME) nasceu da necessidade de adaptar a tecnologia Java aos dispositivos móveis com sérias limitações de recursos, quando comparados com os computadores normais. Assim, o JME é um ambiente Java direcionado a uma vasta gama de dispositivos, como celulares, *paggers* e PDAs (*Personal Digital Assistants*).

A tecnologia JME pode ser dividida em dois componentes: *Profiles* e Configurações, conforme se pode observar na Figura 1. Existem dois tipos de Configurações e estas se diferenciam em relação à capacidade de armazenamento, poderes de processamento, JVM<sup>2</sup> (*Java Virtual Machine*) utilizada, dentre outros recursos.

---

<sup>1</sup> Órgãos colegiados da Universidade responsáveis pela gestão pedagógica dos cursos.

<sup>2</sup> Resumidamente pode-se entender uma máquina virtual Java (JVM – *Java Virtual Machine*) como uma sistema que interpreta e executa o código Java em uma plataforma de hardware qualquer.

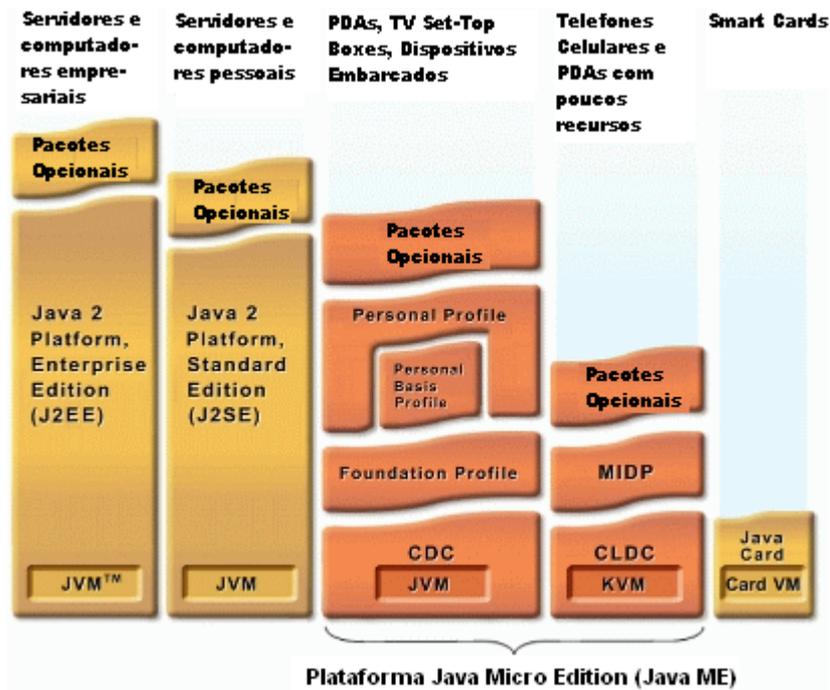


Figura 1 – Plataforma Java – (SUN MICROSYSTEMS INC., 2008).

As configurações existentes são:

- *Connected Limited Device Configuration* – CLDC – (SUN MICROSYSTEMS INC., 2008b): define um ambiente de execução e um conjunto de APIs (*Application Program Interface*) específicas para ambientes extremamente restritos, com capacidade mínima de processamento, fonte de energia limitada, *display* reduzido e conectividade intermitente. As principais características que o dispositivo no qual serão executados os programas deve ter são:
  - Ter pelo menos 128 Kb de memória disponível para a máquina virtual e as bibliotecas CLDC.
  - Ter pelo menos 32 Kb de memória disponível para o *runtime* da máquina virtual.
  - Ter processador de 16 bits ou de 32 bits
  - Apresentar baixo consumo de energia, freqüentemente operando através de bateria.
  - Apresentar conectividade a algum tipo de rede, freqüentemente sem fio, de forma intermitente e com largura de banda limitada.
- *Connected Device Configuration* – CDC – (SUN MICROSYSTEMS INC., 2008c): é a configuração utilizada pelo projeto e especifica um conjunto de tecnologias destinadas à utilização em dispositivos com maiores recursos. Utilizando o CDC podem ser construídas interfaces gráficas mais elaboradas e aplicações mais inteligentes e pesadas em termos computacionais, se comparadas às aplicações feitas em CLDC. Suas principais características são:
  - Necessidade de um processador de 32 bits.
  - O dispositivo ter disponível para o ambiente Java pelo menos 2MB de memória.

- O dispositivo apresentar conectividade a algum tipo de rede, frequentemente sem fio.

Um *Profile* é um conjunto de APIs que reside dentro da JVM do aparelho. Este conjunto de APIs varia de acordo com a configuração utilizada. Cada fabricante pode optar por adicionar APIs próprias ou disponibilizar o conjunto de APIs do J2ME. Os *profiles* existentes são:

- *Mobile Information Device Profile* – MIDP – (SUN MICROSYSTEMS INC., 2008d): O MIDP foi feito para celulares e PDAs. Ele oferece as bases para o desenvolvimento de funcionalidades requeridas pelas aplicações móveis, incluindo interfaces do usuário, conexões com rede, persistência de dados e controle de aplicações. Combinado com o CLDC, o MIDP fornece um ambiente de execução Java completo que alavanca a capacidade de PDAs e minimiza o consumo de memória e energia.
- *Foundation Profile* – FP – (SUN MICROSYSTEMS INC., 2008e): Os perfis CDCs são divididos, podendo ser utilizados em função da funcionalidade e do tipo de dispositivo utilizado. O *Foundation Profile* (FP) é o nível mais básico de perfil CDC. Ele fornece uma implementação de rede do CDC que pode ser usada para construir aplicações sem interface com o usuário. Ele pode também ser combinado com o *Personal Basis Profile* (PBP) e o *Personal Profile* (PP) para dispositivos que necessitem de uma interface gráfica com o usuário.
- *Personal Profile* – PP – (SUN MICROSYSTEMS INC., 2008f): Este é o perfil adotado no sistema. É o perfil CDC utilizado em dispositivos que necessitam de um suporte completo para Interface Gráfica, como PDAs e consoles para jogos. Ele fornece um ambiente de aplicações Java para dispositivos que requerem a utilização da biblioteca AWT completa. A implementação da biblioteca AWT do PP é baseada na combinação da `java.awt` proveniente da JDK 1.1 com alguns pacotes 2D provenientes da Java Standard Edition – JSE. Alguns pacotes e classes são subconjuntos de outros pacotes e classes pertencentes também a JSE.
- *Personal Basis Profile* – PBP – (SUN MICROSYSTEMS INC., 2008g): Este perfil é uma divisão do PP, fornecendo um ambiente para dispositivos conectados que suportem um nível básico de apresentação gráfica ou necessitem do uso de *toolkits* específicos para aplicações. Ambos, PP e PBP, constituem as camadas superiores do CDC.

## 2.2 Perfil *Personal Profile*

Existem diversas possibilidades tecnológicas para a construção de interfaces gráficas para aplicações que executam em dispositivos móveis. Entre elas se pode citar: Light-Weight Visual Component Library – LwVCL – (LWVCL, 2008), Thinlet (BAJZAT, 2008) e a utilização do perfil Personal Profile do JME. Conforme citado acima, esta última alternativa foi a tecnologia escolhida para ser usada no projeto pois permite o desenvolvimento de aplicativos JME que utilizem a biblioteca AWT, o que possibilita o desenvolvimento de interfaces gráficas elaboradas.

## 2.3 XML

É um acrônimo de *eXtensible Markup Language* (W3C, 2008). Trata-se de uma linguagem que pode ser manipulada por software, integrando-se com as demais linguagens. Ela pode ser descrita por diferentes características, entre as quais se pode citar: separação do conteúdo da formatação, simplicidade e legibilidade, tanto para humanos quanto para computadores, possibilidade de criação de marcadores sem limitação, criação de arquivos para validação de estrutura (Chamados DTDs), por constituir uma ferramenta para interligação de bancos de dados distintos e por concentrar-se na estrutura da informação e não em sua aparência.

O XML é considerado um bom formato para a criação de documentos com dados organizados de forma hierárquica, como se vê frequentemente em documentos de texto formatados, imagens vetoriais ou bancos de dados. Ele é usado no sistema para o intercâmbio de dados entre o sistema do PDA e o SW3A.

## 2.4 Web Services

É uma solução utilizada na integração de sistemas e na comunicação entre aplicações diferentes através da Web. Com esta tecnologia é possível que novas aplicações possam interagir com aquelas que já existem de maneira compreensível, reutilizável e padronizada e que sistemas desenvolvidos em plataformas diferentes sejam compatíveis. Os Web Services são componentes que permitem às aplicações enviar e receber dados em formato XML. Cada aplicação pode ter a sua própria "linguagem", que é traduzida para uma linguagem universal, o formato XML.

## 3. ESTRUTURA DO PROJETO

O sistema foi desenvolvido para utilização em dispositivos do tipo PDA, especificamente em um Pocket PC. Deste, os fatores limitantes são a escassez de memória e o poder de processamento. A utilização de um banco de dados interno ao dispositivo foi inicialmente estudada, mas abandonada em função da complexidade da solução e também devido às poucas alternativas viáveis, sejam economicamente, seja tecnicamente. Por esta razão preferiu-se utilizar a tecnologia de Web Services que, como já citado, possibilita a utilização de arquivos em formato XML para armazenamento e troca de informações.

Através de um aplicativo Java desenvolvido como parte integrante do projeto SW3A, utiliza-se um computador comum para acessar o banco de dados central e, em seguida, construir um arquivo XML com os dados que se pretende importar para o PDA. Através da tecnologia de Web Services os dados deste arquivo são enviados ao dispositivo móvel onde será montado um novo arquivo XML. Durante a utilização do sistema este arquivo é consumido e, logo após, seu conteúdo já pode ser reenviado ao *desktop* onde servirá para a atualização do banco de dados. A Figura 2 apresenta um esquema do funcionamento do sistema.



Figura 2 - Estrutura básica do sistema.

### 3.1 Aplicação Auxiliar

O sistema inclui uma aplicação que é executada em um servidor onde o banco de dados está instalado. Esta aplicação carrega das tabelas do banco de dados do servidor, todas as informações necessárias para que a aplicação J2ME possa executar sem problemas no dispositivo móvel.

A aplicação não possui interface gráfica, portanto seu funcionamento não é visível. Ela simplesmente aguarda ser chamada através de uma porta de comunicações. Caso o PDA esteja conectado à máquina cliente basta clicar em um botão apropriado (**RECEBER DADOS**) na tela principal para que esta aplicação carregue as informações do banco de dados no dispositivo móvel ou basta clicar em outro botão (**EXPORTAR DADOS**) para que o banco de dados seja atualizado com as informações de chamada registradas no PDA. Conforme citado, a troca de dados se dá via Web Services.

### 3.1 Aplicação Principal

Esta seção apresenta uma descrição detalhada do funcionamento da aplicação J2ME com ilustrações da interface gráfica desenvolvida.

Inicialmente tem-se a tela principal (c.f. Figura 3a), que une os campos para validação do usuário e as principais funcionalidades do sistema. A data atual é exibida, mas há possibilidade de alterá-la. Para iniciar o uso do sistema, o PDA deve estar conectado diretamente à rede através de uma rede sem fio, ou deve estar conectado a um computador que tenha acesso à rede. Em seguida deve-se clicar em **RECEBER DADOS**. Visualmente esta ação apenas habilitará o botão para validação de usuário (c.f. Figura 3b), mas implicitamente fará com que as informações do Banco de Dados sejam recuperadas pelo dispositivo e sejam inseridas num arquivo XML criado dinamicamente no PDA. Através deste arquivo o sistema manipulará todos os dados, lendo as informações do professor para serem exibidas na tela e armazenando no próprio arquivo as informações do registro da aula e das presenças.

Após esta ação, deve-se completar os campos com o nome e senha do docente e clicar em OK. Se as informações forem validadas, o botão **INICIAR** será habilitado como mostra a Figura 3c. Ao pressionar este botão, o sistema segue para a próxima etapa.



Figura 3 - Tela 1 (a) situação inicial; (b) após recebimento de dados; (c) após validação.

O sistema fornece também mensagens de sucesso ou erro ao usuário, que são exibidas juntamente com a tela que já está aberta. As possíveis mensagens para a primeira tela estão mostradas na Figura 4. Caso seja digitado um usuário que não existe, a mensagem **login não cadastrado** será exibida (c.f. Figura 4a). Se a senha não corresponder ao usuário, será apresentada a mensagem **Senha incorreta** (c.f. Figura 4b). Se, ao clicar em **RECUPERAR DADOS**, por algum motivo o arquivo XML não puder ser escrito, o usuário verá a mensagem **Falha no recebimento de dados**.



Figura 4 – Mensagens (a) de não validação; (b) de erro de login.

A próxima etapa consiste na escolha de disciplina e turma. Cada disciplina é separada com suas possíveis turmas e o professor deve selecionar a opção desejada e clicar em **OK** para que o sistema siga em frente. Caso queira voltar à tela principal sem fazer a escolha, deve-se clicar em **CANCELAR** (c.f. Figura 5).



Figura 5 – Tela 2, Escolha de Disciplina e Turma.

Se o usuário não fizer sua escolha e clicar em **OK**, a mensagem de erro **Selecione uma turma para continuar** será exibida, como mostra a Figura 6.



Figura 6 - Mensagem de Erro para Escolha da Disciplina.

Na etapa seguinte é preciso selecionar o assunto que será ministrado na aula do dia. Cada assunto constitui um tópico do plano de ensino e possui um número total de aulas a serem ministradas. Cada tópico é inicialmente carregado no sistema juntamente com o número total de aulas. O professor, porém, pode desejar não ministrar todas as aulas do tópico no mesmo dia ou mesclar mais de um tópico no mesmo dia. Por isso é possível selecionar mais de um campo e alterar o número que aparece junto aos tópicos, indicando as aulas ministradas. A interface que permite esta manipulação é mostrada na Figura 7.



Figura 7 – Tela3, Escolha do Assunto a ser Ministrado.

Após selecionar os tópicos que vai ministrar e, se preciso, mudar o número de aulas, o docente deve clicar em OK para dar continuidade ao processo de registro de presença ou em **CANCELAR** caso deseje voltar à tela principal e perder os dados já registrados. Se não for selecionado nenhum assunto, a mensagem de alerta **Selecione um tópico** é exibida, como mostra a Figura 8.



Figura 8 - Mensagem de alerta para a escolha do assunto.

Para finalizar, o sistema apresenta a lista dos alunos da turma selecionada e os campos para que seja marcada a presen\u00e7a ou aus\u00eancia do aluno (c.f. Figura 9). A quantidade de campos \u00e9 definida pelo n\u00famero de aulas declaradas na tela anterior.



Figura 9 - Tela4, Lista de alunos para registro de presen\u00e7as.

Ao clicar em **REGISTRAR**, as informa\u00e7\u00f5es sobre o t\u00f3pico que foi ministrado na data escolhida, a turma para qual o t\u00f3pico foi ministrado, quais alunos estiveram presentes e em quantas aulas cada aluno esteve presente, s\u00e3o salvas no arquivo XML presente no PDA. Em seguida o sistema volta \u00e0 tela inicial e exibe a mensagem **Lista registrada com sucesso** (c.f. Figura 10).



Figura 10 - Mensagem Indicativa que o Registro foi Bem Sucedido.

Quando o professor conectar o PDA à rede sem fio ou a seu computador com acesso à rede, ele deve clicar em **EXPORTAR DADOS** para que os dados do arquivo XML sejam enviados para o banco de dados original no servidor. Caso aconteça algum erro no envio dos dados a mensagem **Erro ao exportar dados** será exibida. Caso contrário a confirmação de exportação através da apresentação da mensagem **Dados exportados com sucesso** será efetuada (cf. Figura 11).



Figura 11 – Possíveis mensagens após envio de dados.

É importante destacar que é possível fazer vários registros de presença sem a necessidade de exportar os dados a cada operação, já que é comum o docente ministrar aulas seguidas para turmas diferentes.

#### 4. CONSIDERAÇÕES FINAIS

O trabalho aqui apresentado envolveu o desenvolvimento de uma seqüência composta de estudo, análise e desenvolvimento de um sistema que pode ser utilizado por professores para

facilitar, através do uso de um PDA, o controle do acompanhamento acadêmico dos cursos por eles ministrados, mostrando no PDA dos planos de ensino de suas disciplinas, bem como telas para o registro das presenças dos alunos. O cadastro dessas informações é feito com o objetivo de facilitar também o acompanhamento dos cursos pelos Conselhos de Curso das Instituições de Ensino através de um módulo do SW3A.

Entre as maiores dificuldades encontradas no desenvolvimento do sistema pode-se citar a integração do emulador de PDA no ambiente Windows e os passos necessários para que a aplicação possa funcionar corretamente neste dispositivo. As limitações do Java JME devem ser destacadas em função da pouca memória dos dispositivos aos quais ele se destina. Por este motivo é necessária muita atenção ao utilizar certas funcionalidades e, muitas vezes, só percebe-se que a funcionalidade não é aceita no momento da simulação.

Podem-se destacar algumas possíveis melhorias no sistema. Dentre elas a possibilidade de trabalhar com a API gráfica SWT (THE ECLIPSE FOUNDATION, 2008), que oferece recursos visuais ainda mais elaborados. Também seria útil adicionar uma funcionalidade de bloco de notas para que o docente possa fazer observações sobre as aulas e tópicos que está ministrando.

### ***Agradecimentos***

À Fundação de Amparo a Pesquisa do estado de São Paulo (FAPESP) pelo auxílio através do projeto de pesquisa de nº 2004/08384-3.

### **REFERÊNCIAS BIBLIOGRÁFICAS**

FREIRE JUNIOR, J.C.; SENNE, E.L.F.; SENA, G.J. **Evolução do Desenvolvimento de um Sistema Web de Avaliação e Acompanhamento Acadêmico**. In: XXXIV Congresso Brasileiro de Ensino de Engenharia - COBENGE 2006, 2006, Passo Fundo. Anais. Passo Fundo. CD-ROM

SUN MICROSYSTEMS INC., **Java ME at a Glance**, Disponível em: <<http://java.sun.com/javame/index.jsp>>. Acesso em: 01 jun. 2008.

SUN MICROSYSTEMS INC., **Abstract Window Toolkit (AWT)**, Disponível em: <<http://java.sun.com/javase/6/docs/technotes/guides/awt/>>. Acesso em: 01 jun. 2008a.

SUN MICROSYSTEMS INC., **Connected Limited Device Configuration (CLDC); JSR 30, JSR 139**, Disponível em: <<http://java.sun.com/products/cldc/index.jsp>>. Acesso em: em 01 jun. 2008b.

SUN MICROSYSTEMS INC., **Sun Java Toolkit for CDC Overview**, Disponível em: <<http://java.sun.com/products/cdctoolkit/overview.html>>. Acesso em: 01 jun. 2008c.

SUN MICROSYSTEMS INC., **Mobile Information Device Profile (MIDP); JSR 37, JSR 118**, Disponível em: <<http://java.sun.com/products/midp/index.jsp>>. Acesso em: 01 jun. 2008d.

SUN MICROSYSTEMS INC., **Java ME - Foundation Profile**, Disponível em: <<http://java.sun.com/products/foundation/index.jsp>>. Acesso em: 01 jun. 2008e.

SUN MICROSYSTEMS INC., **Java ME - Personal Profile**, Disponível em: <<http://java.sun.com/products/personalprofile/index.jsp>>. Acesso em: 01 jun. 2008f.

SUN MICROSYSTEMS INC., **Java ME - Personal Basis Profile**, Disponível em: <<http://java.sun.com/products/personalbasis/index.jsp>>. Acesso em: 01 jun. 2008g.

LWVCL, **LightWeight Visual Component Library**, Disponível em: <<http://lwwcl.com/>>. Acesso em: 01 jun. 2008.

BAJZAT, R., **Thinlet**, Disponível em: <<http://www.thinlet.com/index.html>>. Acesso em: 01 jun. 2008.

W3C – World Wide Web Consortium, **Extensible Markup Language (XML)**, Disponível em: <<http://www.w3.org/XML/>>. Acesso em: 01 jun. 2008.

THE ECLIPSE FOUNDATION, **SWT: The Standard Widget Toolkit**, Disponível em: <<http://www.eclipse.org/swt/>>. Acesso em: 01 jun. 2008.

## **COURSES FOLLOW-UP MOBILE SYSTEM**

**Abstract:** *This work presents a system that can be executed in mobile devices such as PDAs (Personal Digital Assistants) whose purpose is the follow-up of courses development and the presence register tracking. The system was developed in JME (Java Micro Edition) using the PP/CDC (Personal Profile/Connected Device Configuration) APIs. The developed system provides a tool that enables the teachers of education institutions not only to accomplish the management of a course with the records of topics given in a subject in a very fast and practical way, but also to record the presence or absence of students to a class. Banishing, this way, with the present need to fill out charts by hand and so making faster the availability of data in the Web for the students.*

**Key-words:** *Courses Management, Mobile Devices, Java Technology, Java Micro Edition.*