

UNIVERSIDADE DO VALE DO RIO DOS SINOS  
CIÊNCIAS EXATAS E TECNOLÓGICAS  
PROGRAMA INTERDISCIPLINAR DE PÓS-GRADUAÇÃO EM  
COMPUTAÇÃO APLICADA

Darci Levis

**PeLeP: Um modelo de perfil  
de aprendiz orientado  
à aprendizagem ubíqua**

São Leopoldo

2007

Darci Levis

**PeLeP: Um modelo de perfil  
de aprendiz orientado  
à aprendizagem ubíqua**

Dissertação apresentada à Universidade  
do Vale do Rio dos Sinos como requisito  
parcial para obtenção do título de Mestre  
em Computação Aplicada.

Orientador: Prof. Dr. Jorge Luís Victória Barbosa

São Leopoldo

2007

## CIP — CATALOGAÇÃO NA PUBLICAÇÃO

Levis, Darci

PeLeP: Um modelo de perfil de aprendiz orientado à aprendizagem ubíqua / por Darci Levis — São Leopoldo: Ciências Exatas e Tecnológicas da Unisinos, 2007.

96 f.: il.

Dissertação (mestrado) — Universidade do Vale do Rio dos Sinos. Ciências Exatas e Tecnológicas Programa Interdisciplinar de Pós-Graduação em Computação Aplicada, São Leopoldo, BR-RS, 2007. Orientador: Barbosa, Jorge Luís Victória.

1. Computação móvel.
  2. Computação ubíqua.
  3. Aprendizagem ubíqua.
  4. Objetos de aprendizagem.
  5. Estilos de aprendizagem.
- I. Barbosa, Jorge Luís Victória.  
II. Título.

UNIVERSIDADE DO VALE DO RIO DOS SINOS

Reitor: Dr. Marcelo Fernandes de Aquino

Diretora da Unidade de Pós-Graduação e Pesquisa: Prof<sup>a</sup>. Dr<sup>a</sup>. Ione Bentz

Coordenador do PIPCA: Prof. Dr. Arthur Tórgo Gómez

*Dedico esta dissertação:  
ao professor Dr. João Goedert;  
à minha namorada, Cleidi;  
aos meus pais, Divaldino e Adelina;  
aos meus irmãos, Reni e Daniel.*

# Agradecimentos

Agradeço primeiramente a Deus, em nome de Jesus Cristo, pela vida e por ter me guiado no desenvolvimento deste trabalho.

Ao meu orientador, Professor Dr. Jorge Luís Victória Barbosa, pelo incentivo, ânimo, paciência, amizade, apoio e orientação, que foram imprescindíveis para a conclusão desta dissertação.

À Professora Ms. Débora Nice Ferrari Barbosa, pelas idéias e conselhos transmitidos, que ajudaram a “alavancar” este trabalho.

Ao professor Dr. Fernando Osório, pelos conhecimentos e indicações à respeito de regras de inferência e aos demais professores do Programa de Pós-Graduação em Computação Aplicada (PIPCA), que de alguma forma tornaram meu trabalho possível.

Aos professores Renata Vieira e Sérgio Crespo Coelho da Silva Pinto por participarem da minha banca. À Universidade do Vale do Rio dos Sinos, pela oportunidade de desenvolver esta pesquisa e pelo apoio financeiro.

À minha namorada Cleidi Padilha, por compreender os momentos de ausência de minha parte, em função do desenvolvimento deste trabalho. À minha família, pelo apoio moral e incentivo aos estudos.

Ao amigo Cícero Rolim (Baboo), pelos conselhos e dicas que ajudaram na implementação das rotinas desenvolvidas para esta dissertação e à todos os colegas de curso, em especial a Gustavo Lermen, que tornaram-se verdadeiros amigos e auxiliaram em vários estudos. Ao bolsista e amigo Rodrigo Hahn (Dormento) do MobiLab, por sua participação no desenvolvimento deste trabalho.

Enfim, a todos que direta ou indiretamente contribuíram para a realização desta pesquisa.

# Resumo

A computação ubíqua traz várias potencialidades para os processos de ensino e aprendizagem. Do ponto de vista educacional, o desafio é a criação de um ambiente de aprendizagem móvel que proporcione recursos apropriados para o aprendiz. No cenário da aprendizagem ubíqua, a criação de um Perfil de Aprendiz, que una suas localizações, interesses e informações referentes a seu estudo torna-se importante. Esta dissertação apresenta o PeLeP – *Pervasive Learning Profile*, que consiste em um modelo de aprendiz voltado para sistemas de suporte à aprendizagem ubíqua. O objetivo do PeLeP é possibilitar que aplicações educacionais possam explorar o ensino e aprendizagem ubíquos, baseando-se em um perfil organizado por categorias que contêm informações da vida cotidiana do aprendiz. O perfil do aprendiz no modelo PeLeP é refinado e enriquecido através de inferências. Estas, são baseadas na mobilidade do aprendiz por contextos, e pelas tarefas que ele executa nos mesmos dentro de um ambiente de computação ubíqua.

**Palavras-chave:** Computação móvel, Computação ubíqua, Aprendizagem ubíqua, Objetos de aprendizagem, Estilos de aprendizagem.

**TITLE:** “PELEP: PERVASIVE LEARNING PROFILE”

## Abstract

The ubiquitous computing brings several potentialities for the processes of teaching and learning. Of the educational point of view, the challenge it is the creation of an environment of mobile learning that provides resources adapted for the learner. In the scenario of the learning ubiquitous, the creation of a Learner Profile, that unites yours locations, interests and referring information its study he/she becomes important. This dissertation presents PeLeP - Pervasive Learning Profile, which consists in a learner profile model oriented to ubiquitous learning systems. PeLeP aims to let applications explore ubiquitous education and learning through user profiles organized by categories containing information about that learner's daily life. Learner profiles in PeLeP are refined and enriched through inferences. These inferences are based on several factors, such as learner's mobility between contexts, and the tasks he executes in each of them while inside a ubiquitous computing environment.

**Keywords:** Ubiquitous education, Ubiquitous computing, Mobile computing, Ubiquitous learning, Learning objects, Learning styles.

# Sumário

<b>Resumo</b>	<b>5</b>
<b>Abstract</b>	<b>6</b>
<b>Lista de Abreviaturas</b>	<b>10</b>
<b>Lista de Figuras</b>	<b>12</b>
<b>Lista de Tabelas</b>	<b>13</b>
<b>1 Introdução</b>	<b>14</b>
1.1 Contextualização . . . . .	14
1.2 Definição do problema . . . . .	16
1.3 Objetivos e metodologia . . . . .	18
1.4 Organização da dissertação . . . . .	19
<b>2 Educação e mobilidade</b>	<b>20</b>
2.1 Computação móvel e ubíqua . . . . .	20
2.1.1 Desafios . . . . .	21
2.1.2 Computação móvel . . . . .	22
2.1.3 Computação consciente de contexto . . . . .	22
2.1.4 Adaptabilidade . . . . .	27
2.2 Evolução das tecnologias da educação . . . . .	28
2.2.1 Educação a distância . . . . .	28
2.2.2 Mobilidade aplicada à educação . . . . .	29
2.3 Objetos de aprendizagem . . . . .	31
2.4 Estilos de aprendizagem . . . . .	32
2.5 Perfil de aprendiz . . . . .	34



2.6	Considerações finais . . . . .	35
<b>3</b>	<b>Trabalhos Relacionados</b>	<b>37</b>
3.1	CC/PP . . . . .	37
3.2	CSCP . . . . .	38
3.3	PAPI . . . . .	39
3.4	LIP . . . . .	40
3.5	SeLeNe . . . . .	41
3.6	<i>Proem</i> . . . . .	42
3.7	JAPELAS . . . . .	42
3.8	GlobalEdu . . . . .	43
3.9	LOCAL . . . . .	44
3.10	Considerações finais . . . . .	46
<b>4</b>	<b>Modelo PeLeP</b>	<b>47</b>
4.1	Princípios do PeLeP . . . . .	47
4.2	Arquitetura . . . . .	48
4.3	Descrição do perfil do aprendiz . . . . .	49
4.4	<i>Tracking</i> do aprendiz . . . . .	52
4.4.1	Localizações . . . . .	53
4.4.2	Recursos acessados . . . . .	54
4.4.3	Dispositivos Utilizados e Aplicativos Executados . . . . .	55
4.5	Normalização dos dados recebidos no <i>tracking</i> . . . . .	55
4.6	Fator de certeza . . . . .	56
4.7	Banco de Resultados . . . . .	57
4.8	Regras . . . . .	58
4.8.1	Categoria Preferências . . . . .	59
4.8.2	Categoria Competências . . . . .	59
4.8.3	Categoria Objetivos . . . . .	60
4.8.4	Categoria Agenda . . . . .	60
4.8.5	Categoria Relacionamentos . . . . .	61
4.9	Regra final . . . . .	62
4.10	Considerações finais . . . . .	63

<b>5 Aspectos de implementação e resultados obtidos</b>	<b>64</b>
5.1 Protótipo . . . . .	64
5.2 Integração LOCAL/PeLeP . . . . .	68
5.3 Resultados obtidos . . . . .	69
5.4 Considerações finais . . . . .	78
<b>6 Considerações Finais</b>	<b>79</b>
6.1 Revisão de tema . . . . .	79
6.2 Conclusões . . . . .	81
6.3 Trabalhos Futuros . . . . .	82
<b>Bibliografia</b>	<b>84</b>
<b>A Exemplo de <i>Tracking</i> representado em XML</b>	<b>93</b>

# Lista de Abreviaturas

<b>ACP</b>	Arquitetura Computacional Pervasiva
<b>ACM</b>	Association for Computing Machinery
<b>AdaptWeb</b>	Ambiente de Ensino-Aprendizagem Adaptativo na Web
<b>AN</b>	Adaptative Navigation
<b>AP</b>	Adaptative Presentation
<b>API</b>	Application Programming Interface
<b>A3P</b>	Agente Pedagógico Pessoal Pervasivo
<b>CC/PP</b>	Composite Capability/Preference Profiles
<b>CPI</b>	Capability and Preference Information
<b>CSCP</b>	Comprehensive Structured Context Profiles
<b>Elena</b>	E-learning Networks access
<b>EA</b>	Estilo de Aprendizagem
<b>FC</b>	Fator de Certeza
<b>GlobalEdu</b>	Global Education
<b>GPS</b>	Global Positioning System
<b>IEEE</b>	Institute of Electrical and Electronics Engineers
<b>IETF</b>	Internet Engineering Task Force
<b>IMS</b>	Instructional Management Systems
<b>ISAM</b>	Infra-estrutura de Suporte às Aplicações Móveis
<b>JAPELAS</b>	Japanese Polite Expressions Learning Assisting System

<b>LIP</b>	Learner Information Package
<b>LOCAL</b>	Location and Context Aware Learning
<b>LMS</b>	Learning Management System
<b>LCMS</b>	Learning Content Management Systems
<b>MobiLab</b>	Mobile Computing Lab
<b>MySQL</b>	My Structured Query Language
<b>OA</b>	Objeto de Aprendizagem
<b>OArp</b>	Objeto de Aprendizagem Recomendado por professor
<b>OAsup</b>	Objeto de Aprendizagem Suplementar
<b>PAPI</b>	Public and Private Information
<b>PDA</b>	Personal Digital Assistant
<b>PeLeP</b>	Pervasive Learning Profile
<b>PHP</b>	Hypertext Preprocessor
<b>Proem</b>	Profiles Exchange Mechanism
<b>P2P</b>	Peer-to-peer
<b>RDF</b>	Resource Description Framework
<b>RFID</b>	Radio-Frequency Identification
<b>SE</b>	Serviços Educacionais
<b>SeLeNe</b>	Self E-Learning Networks
<b>SeLIC</b>	Servidor de Localização e Informação de Contexto
<b>TANGO</b>	Tag Added learNinG Objects
<b>UAProf</b>	User Agent Profile
<b>XML</b>	eXtensible Markup Language
<b>WAP</b>	Wireless Application Protocol

# Lista de Figuras

2.1	Conjunto dos ambientes de <i>e-Learning</i> . . . . .	30
2.2	Comparação de ambientes de aprendizagem [56] . . . . .	31
3.1	Arquitetura do LOCAL . . . . .	44
4.1	Arquitetura do PeLeP . . . . .	48
4.2	Perfil do Aprendiz no PeLeP . . . . .	50
4.3	Funcionamento da Agenda . . . . .	52
4.4	Exemplo de contexto e árvore de contextos . . . . .	53
4.5	Exemplo da estrutura dos contextos adaptado ao MobiLab . . . . .	54
5.1	Rotina parcial do Gerenciador de Perfil . . . . .	65
5.2	Normalização dos dados . . . . .	66
5.3	Cálculo do FC . . . . .	67
5.4	PeLeP inserido na arquitetura do LOCAL . . . . .	68
5.5	Categorias Identificação, Agenda e Relacionamentos . . . . .	70
5.6	Primeira execução para a categoria Competências . . . . .	71
5.7	Primeira execução para as categorias Preferências e Objetivos . . . . .	72
5.8	Terceira execução para a categoria Relacionamentos . . . . .	73
5.9	Terceira execução para a categoria Competências . . . . .	74
5.10	Terceira execução para a categoria Objetivos . . . . .	75
5.11	Terceira execução para a categoria Preferências . . . . .	77
A.1	Exemplo de <i>tracking</i> do aprendiz . . . . .	96

# Lista de Tabelas

2.1	Dimensões de estilos de aprendizagem [26] . . . . .	34
3.1	Avaliação dos principais modelos pesquisados . . . . .	46
6.1	Comparação do PeLeP com os principais modelos pesquisados . . . . .	81

# Capítulo 1

## Introdução

### 1.1 Contextualização

A crescente utilização da computação móvel [51] tem gerado o surgimento de diversos aplicativos específicos para seu uso. Conseqüentemente, modelos de dispositivos estão sendo criados e aperfeiçoados constantemente, oferecendo aos usuários diferentes funcionalidades e qualidade de serviços.

Atualmente, existem PDAs que possuem telefone celular, câmera e localização por GPS acoplados. Isso leva a crer que num futuro próximo estes dispositivos se popularizarão, tal como ocorreu com os aparelhos de telefonia móvel. Possivelmente o mundo se adaptará à utilização de algum tipo de dispositivo portátil com conexões sem fio. Um usuário que possui um dispositivo móvel pode estar em ambientes distintos utilizando diferentes conexões e configurações, de acordo com os recursos de rede disponíveis e suportados pelo seu aparelho.

O uso da computação móvel está cada vez mais presente na educação, visto que o ensino e a aprendizagem são cada vez mais apoiados por ferramentas móveis ou ocorrem em um contexto<sup>1</sup> onde equipamentos móveis estão disponíveis [62]. O desafio que se apresenta é projetar aplicações educacionais em um ambiente onde os níveis de serviço e disponibilidade de recursos são imprevisíveis. Existem propostas [57, 62, 70] que são exemplos do uso de dispositivos móveis (celulares, *handhelds*, etc) no suporte à educação.

---

<sup>1</sup>Contexto pode ser definido como qualquer informação que pode ser usada para caracterizar a situação de uma pessoa ou entidade computacional [20]. E segundo Barbosa [9], contexto dinâmico pressupõe a alteração destas informações em tempo real, em função da mobilidade do usuário.

Outro aspecto tecnológico presente atualmente é a computação consciente do contexto (*context-aware computing*) [74]. Esta percebe os elementos que envolvem o usuário e que são de interesse dele ou de suas aplicações, fazendo com que a computação se adapte ao contexto presente. No cenário educacional, pode-se citar o sistema JAPÉLAS [77, 78] como um sistema consciente do contexto aplicado à educação.

Na computação ubíqua [65] os usuários podem acessar suas informações das mais variadas formas e de lugares distintos, mantendo o acesso ao seu ambiente computacional. Isso aliado as diferentes formas de comunicação gera uma revolução na consulta/transferência de informações. A computação ubíqua propõe, assim, uma computação que envolve o usuário, abstraindo questões como necessidade de adaptação ao dispositivo ou acesso a um determinado recurso.

A computação ubíqua traz também várias potencialidades para os processos de ensino e aprendizagem. As pessoas podem se comunicar e buscar informações, utilizando seus dispositivos móveis para acessar seu ambiente computacional independente de lugar e tempo. Do ponto de vista educacional, o desafio é a criação de um ambiente de aprendizagem móvel que proporcione recursos apropriados para o aprendiz. Isto pode ser alcançado pela adaptação de conteúdos de acordo com as preferências do aprendiz, pelas características da conexão e por um modelo instrutivo adequadamente selecionado.

Uma vez conhecido o contexto, a adaptação a ele é a chave para unir as diversas necessidades dos diferentes aprendizes. A idéia da computação ubíqua no cenário educacional é permitir que os processos educacionais ocorram em qualquer lugar, a qualquer tempo e com qualquer dispositivo. O PeLeP utiliza o modelo de aprendiz do GlobalEdu [7, 8, 9] como estrutura de perfis e é acoplado ao LOCAL [11, 12], que é um sistema para a aprendizagem ubíqua em desenvolvimento no MobiLab [52]. O GlobalEdu é uma infra-estrutura para suporte à aprendizagem ubíqua integrada a um ambiente computacional ubíquo de larga escala. No GlobalEdu, o aprendiz tem a possibilidade de aprender em qualquer tempo e lugar, de forma adaptada ao seu contexto e perfil. O LOCAL usa informações de localização e de contexto no ambiente ubíquo como auxílio ao processo de ensino e aprendizagem. O componente Servidor de Localização e Informação de Contexto (SeLIC) do LOCAL foi utilizado para o fornecimento de informações sobre o histórico do aprendiz para o PeLeP. O



MobiLab é um Laboratório de Pesquisa e Desenvolvimento em Computação Móvel, localizado na Universidade do Vale do Rio dos Sinos.

Para que uma aplicação obtenha proveito dos recursos oferecidos na computação ubíqua, deve adaptar-se a um ambiente de execução dinâmico e distribuído. Isto reflete a própria natureza do uso dos dispositivos portáteis, os quais podem ter seu posicionamento físico continuamente alterado. Neste sentido, o conhecimento da localização geográfica dos dispositivos móveis torna-se de fundamental importância. A associação da localização de um dispositivo a seu sistema identificador viabiliza o surgimento de uma nova gama de aplicações conscientes de contexto. Assim, a eficiência de uma aplicação na computação ubíqua está diretamente relacionada com a localização do usuário dentro do sistema.

No cenário da aprendizagem ubíqua, a criação de um Perfil do Aprendiz, que una suas localizações, interesses e informações referentes à seu estudo e trabalho, torna-se importante. O perfil pode fornecer, além de suas informações pessoais, informações úteis sobre o papel do aprendiz e sua relação com um dispositivo móvel, com outros aprendizes ou com contextos freqüentados por ele. O presente trabalho apresenta o modelo chamado PeLeP – *Pervasive Learning Profile*, que consiste em uma proposta de modelo de aprendiz voltado para sistemas de suporte à aprendizagem ubíqua. A proposta do PeLeP altera os elementos que constituem um perfil do aprendiz, refletindo o seu comportamento no ambiente ubíquo, levando em consideração o seu modelo de mobilidade e de contexto.

## 1.2 Definição do problema

A computação ubíqua ajudará a organizar interações sociais em qualquer lugar e sempre que essas situações possam ocorrer [1, 50]. Esta evolução tem sido recentemente acelerada pelo aperfeiçoamento das potencialidades das telecomunicações das redes sem fio e abertas, do contínuo acréscimo de potência computacional, do aperfeiçoamento de tecnologias das baterias e da emergência de arquiteturas de software flexíveis. Com estas tecnologias, um ambiente de aprendizado individual pode ser integrado à vida real e diária de um aprendiz. O desafio fundamental é projetar sistemas de computação ubíqua que suportem adequadamente processos de ensino e aprendizagem em uma perspectiva também

ubíqua. Desta forma, a “questão de pesquisa” deste trabalho é:

- É possível gerenciar perfis em ambientes ubíquos usando *tracking*<sup>2</sup> para aperfeiçoá-los?

Que deriva os seguintes questionamentos:

- Como fazer para aperfeiçoar as informações do perfil do aprendiz?
- O que considerar para que o aperfeiçoamento seja efetivo?
- Existem aplicações que podem obter proveito do perfil aperfeiçoado?

Um dos pontos característicos de um sistema ubíquo é a capacidade de ser proativo quanto às necessidades do usuário. Neste sentido, um dos problemas consiste em identificar as características de um usuário e atualizar informações contidas em seu perfil automaticamente conforme a sua interação com o sistema. Desta forma, o problema proposto neste trabalho consiste em:

1. Identificar quais são os elementos importantes a serem considerados de forma a representar o perfil de um aprendiz no ambiente de aprendizagem ubíquo;
2. Como estes elementos, em especial os relacionados ao seu contexto e mobilidade, devem ser tratados pelo sistema de forma a modificar o perfil de acordo com as ações e necessidades daquele aprendiz.

O GlobalEdu propõe um modelo de aprendiz, destacando informações que considera importantes de serem utilizadas para representar o aprendiz no ambiente ubíquo. Questões relacionadas a alteração do perfil do aprendiz em função do seu contexto e da sua mobilidade não são abordadas pelo GlobalEdu. Desta forma, o PeLeP complementa o GlobalEdu, propondo a alteração do perfil do aprendiz baseando-se nos diversos contextos que ele frequenta e nas tarefas que ele executa.

Assim, este trabalho propõe incrementar o perfil do aprendiz com informações novas ou alterar informações já existentes através da análise do seu deslocamento e de suas ações nos diferentes contextos em que ele se desloca. Neste sentido, a motivação deste trabalho é o fornecimento de um perfil de aprendiz que contenha informações acerca de suas competências, preferências, atividades a cumprir, seus objetivos educacionais e relacionamentos com os contextos visitados por ele.

---

<sup>2</sup>*Tracking* é o histórico dos contextos visitados e tarefas executadas no ambiente ubíquo.

### 1.3 Objetivos e metodologia

O objetivo deste trabalho é produzir um modelo que refine e enriqueça o perfil do aprendiz através de inferências. Estas inferências serão baseadas no deslocamento do aprendiz por contextos, locais que ele frequenta em sua vida diária, e pelas tarefas que ele executa nos mesmos (recursos de aprendizagem acessados, dispositivos utilizados e programas executados) dentro de um ambiente de computação ubíqua. Para tal, o modelo proposto estrutura os perfis dos aprendizes baseando-se na proposta do GlobalEdu [7, 8, 9].

De forma resumida, o objetivo geral do PeLeP é criar um sistema para aperfeiçoamento de perfis em ambientes ubíquos. Os objetivos específicos são:

- Organizar os perfis por categorias;
- Interpretar as informações de contexto enviadas pelo SeLIC (Servidor de Localização e Informação de Contexto) do sistema LOCAL [11, 12];
- Gerenciar o perfil através de inferências;
- Aperfeiçoar o perfil usando *tracking*;
- Integrar o modelo ao sistema em desenvolvimento no MobiLab [52].

A metodologia utilizada para a concretização dos objetivos foi:

- Revisar os trabalhos relacionados;
- Estudar a arquitetura GlobalEdu e o sistema LOCAL;
- Inserir o PeLeP no sistema LOCAL;
- Definir as informações de *tracking*;
- Realizar as inferências com base no *tracking* do ambiente ubíquo;
- Testar o funcionamento do PeLeP aplicado ao LOCAL.

## 1.4 Organização da dissertação

Este trabalho está organizado da seguinte forma. O capítulo 2 apresenta as tecnologias atuais que se aplicam à Aprendizagem Ubíqua. Em seguida, são apresentados os desafios encontrados com relação à consciência de contexto quando se planeja um sistema que se adapta ao ensino e aprendizagem ubíquos. No mesmo capítulo é apresentada uma comparação entre os ambientes de aprendizagem pesquisados e o que se considera como perfil de um usuário. O capítulo 3 discorre sobre o que foi encontrado na literatura acerca de perfis de aprendizes. Posteriormente, o capítulo 4 descreve o modelo PeLeP. Uma avaliação experimental é apresentada no capítulo 5. Finalmente, o capítulo 6 encerra a dissertação com considerações finais e perspectivas de trabalhos futuros.

## Capítulo 2

# Educação e mobilidade

Este capítulo revisa conceitos e tecnologias importantes nos sistemas de computação para suporte à aprendizagem ubíqua. Além disso, realiza uma breve análise cronológica dos acontecimentos acerca da computação ubíqua desde a origem do termo. Nesta direção são apontadas as necessidades com relação aos dispositivos e sistemas que se propõem a auxiliar o desenvolvimento de novas aplicações para esta área em crescente expansão. Em função do foco do trabalho, aspectos relacionados a consciência de contexto são aprofundados. Posteriormente, são apresentados os desafios encontrados no desenvolvimento de aplicações e dispositivos para ambientes de computação móvel. Em seguida, apresenta-se os ambientes de aprendizagem *Electronic Learning (e-Learning)*, *Mobile Learning (m-Learning)*, *Pervasive Learning (p-Learning)* e *Ubiquitous Learning (u-Learning)*. Paralelamente, discorre-se sobre os Objetos e Estilos de Aprendizagem, e suas relações com o fornecimento e avaliação do conhecimento ao aprendiz. Por fim, são definidos aspectos para criação do Perfil de Aprendiz, cujo gerenciamento é feito pelo PeLeP.

### 2.1 Computação móvel e ubíqua

De forma visionária, Mark Weiser [75] descreveu em 1991 um mundo onde os ambientes cheios de dispositivos computacionais e comunicação, interagem naturalmente com as pessoas, de tal forma que elas passavam a fazer parte desses ambientes. Dentro desta visão surgiu a chamada computação ubíqua [64, 65]. Contudo, a visão de Weiser foi um tanto a frente do seu tempo, e na época não se encontrava disponível a tecnologia necessária para criar o ambiente imaginado. Desde

então as pesquisas na academia e na indústria se focaram na evolução tecnológica que permitiria a criação de ambientes ubíquos.

Na última década, desde que Weiser concebeu sua visão de ubiqüidade, evoluções importantes de hardware podem ser constatadas. Isto permitiu a criação de dispositivos menores, e portanto mais portáteis, bem como sensores e dispositivos de controle. É importante citar ainda as tecnologias para comunicação sem fio, como o *Bluetooth* [14] e o *WiFi* [76], que criam a possibilidade de acesso a informação a qualquer hora, em qualquer lugar, o que não era possível através das redes cabeadas.

### 2.1.1 Desafios

A chegada da computação ubíqua traz consigo novas possibilidades para o desenvolvimento de aplicações. Este aumento no potencial de desenvolvimento se reflete diretamente na complexidade dos problemas relacionados à computação distribuída e móvel, bem como em novos desafios exclusivos desta nova área [15]. Um dos aspectos característicos da computação ubíqua é a mobilidade do usuário e a consciência do contexto que o envolve. Em um cenário como este, as aplicações podem ser executadas a partir de diferentes equipamentos, conectados a diferentes tipos de redes.

Com relação as tecnologias de rede, a diferença na banda disponível para comunicação pode ter variações bastante consideráveis. Isto se torna ainda mais crítico quando a rede em questão é sem fio, onde a variação de sinal reflete diretamente no desempenho da mesma. Para solucionar isso, alguns ambientes se preocupam com modificações em nível de *kernel* do sistema operacional. Outra possibilidade ainda é adaptar a especificação dos dados transmitidos (compactar, modificar resolução, etc), sem modificar o comportamento semântico da aplicação.

No que diz respeito aos equipamentos utilizados para executar aplicações, os casos são ainda mais extremos. Em um ambiente ubíquo, não devem existir limitações quanto ao equipamento usado, podendo este ser um computador de médio/grande porte ou um PDA. Em equipamentos como estes, as diferenças no que diz respeito ao poder de processamento, memória, armazenamento, e mesmo periféricos, são consideráveis. Soluções para estas questões envolvem mapeamento de recursos, balanceamento de carga e seleção dinâmica de código a ser utilizado para o tratamento de entrada e saída na aplicação.

### 2.1.2 Computação móvel

A computação móvel continua sendo uma área de pesquisa bastante ativa, portanto seus conceitos estão em permanente desenvolvimento. As aplicações produzidas até o momento podem ser agrupadas nas áreas de redes móveis, acesso a informação móvel, suporte a aplicações adaptativas, técnicas para economia de energia, sensibilidade e localização [15].

De forma genérica, um cenário que envolve computação móvel é aquele onde alguns dos dispositivos envolvidos na computação, são móveis [6]. Em um ambiente como este, é possível identificar dois perfis para usuários. No primeiro, os usuários são basicamente nômades, ou seja, suas conexões de rede ocorrem em locais e momentos totalmente arbitrários. No segundo grupo se encontram aqueles que estão sempre conectados, utilizando-se para isto redes sem fio. Desta forma é possível identificar algumas propriedades da computação móvel: mobilidade, portabilidade e conectividade [61].

A natureza essencialmente dinâmica de hardware e software, em ambientes de computação móvel, cria a necessidade de existirem soluções para questões como a identificação física de dispositivos quanto à sua localização. Esta questão levanta a necessidade de mecanismos que conheçam a localização de componentes móveis de modo a permitir a interação com os mesmos. Assim, o ambiente que oferecer suporte à computação ubíqua, deve levar em conta estas questões, oferecendo um suporte que permita às aplicações não perderem consistência quando um dispositivo se mover na rede, ou mesmo quando um componente de software migrar para outro dispositivo para continuar sua execução.

### 2.1.3 Computação consciente de contexto

A computação ubíqua apresenta um novo conjunto de desafios de projeto que necessitam novas técnicas. A consciência de contexto é freqüentemente apontada como uma solução para estes desafios, habilitando aplicações de software que exibem os níveis necessários de flexibilidade e autonomia [35]. Aplicações de consciência de contexto exploram informações sobre o contexto em uso, tal como localização, serviços e preferências de usuário. Estas informações podem ser colhidas por sensores ou pelos próprios usuários.

Um ambiente de computação ubíqua conecta usuários em qualquer lugar e em qualquer espaço. O desafio não consiste apenas em produzir informação disponível para as pessoas em qualquer tempo e de qualquer forma. Mas o problema fundamental é como fornecer aos usuários a informação certa, no momento certo e do modo certo [28, 57].

### **Classes da informação de contexto**

Especificamente, três principais fontes de informação de contexto são conhecidas [35]: sensores, usuários humanos e derivações de outros tipos de informações. A informação fornecida pelo usuário é alterada esporadicamente de acordo com as suas necessidades. Este tipo de informação pode ser estática ou dinâmica. Isto é obtido diretamente dos usuários, ou indiretamente de aplicações executadas por eles, como por exemplo a execução de programas que mantêm um histórico de atividades realizadas. Dessa forma, a informação presente no perfil poderá estar desatualizada ou incompleta, e ainda conter erros e imprecisões.

### **Recursos necessários para a representação de contexto**

Informações de contexto são colhidas, armazenadas e interpretadas em diferentes partes de um sistema. Entretanto, a mediação do serviço e do conteúdo dos mecanismos de adaptação necessitam de todas as informações relevantes disponíveis no sistema que controla a mobilidade. Uma representação da informação de contexto deve ser aplicável ao longo de todo o processo de recolhimento, transferência, armazenamento e interpretação da informação de contexto. Conseqüentemente, a junção destas necessidades/recursos constituem um formato de representação. Segundo Held [34], a representação de um perfil no contexto deve ser:

- Estruturada: perfis de contexto podem representar um elevado número de diferentes informações de contexto. Uma representação estruturada fornece meios para filtrar informações relevantes e para uma estruturação natural;
- Permutável: perfis de contexto podem ser permutáveis entre os diferentes componentes do sistema que controla a mobilidade. Desta forma, um perfil não necessitará ser completamente transferido depois da alteração de um simples atributo;



- Uniforme: uma representação uniforme para uma variedade de perfis de contexto (perfil do dispositivo, da rede, do usuário, possivelmente dados adicionais do contexto) facilita a interpretação durante a mediação de processos de serviço e adaptação de conteúdos no sistema;
- Extensível: o conjunto de atributos que podem ser identificados deve ser suficiente para todas as futuras aplicações. Conseqüentemente, um formato de representação de perfis deve servir para extensões futuras;
- Padronizada: perfis de contexto são trocados entre diferentes entidades do sistema, como dispositivos móveis, sistema que controla a mobilidade ou um dispositivo de venda de *web-sites*. Todas as entidades não são tipicamente pertencentes aos mesmos domínios administrativos. Conseqüentemente, se torna necessário uma representação padronizada da informação de contexto.

### **Informações imperfeitas de contexto**

Aplicações conscientes de contexto assumem tipicamente que a informação do contexto é confiável, completa e exata. Entretanto, isso nem sempre ocorre. A informação colhida de um contexto será freqüentemente imprecisa ou indisponível, caso existam ruídos ou falhas nos sensores, quando um usuário forneceu informações que estão sujeitas a problemas, tal como um erro humano. Conseqüentemente, problemas com imperfeitas informações de contexto são algumas vezes observados em aplicações conscientes de contexto.

Henricksen [35] caracteriza quatro tipos de informações imperfeitas de contexto. Ele considera imperfeição com respeito a atributos ou propriedades específicas. Ressalta que estes são aspectos de contexto que podem ser descritos por minúsculas partes da informação, tal como a localização ou atividade de determinado usuário ou as potencialidades de um dispositivo computacional. A propriedade é:

- Desconhecida - quando nenhuma informação sobre a propriedade estiver disponível;
- Duvidosa - quando diversos recursos sobre a propriedade estão disponíveis (por exemplo, quando duas localizações diferentes são lidas para uma determinada pessoa e fornecidas por dispositivos em posições diferentes);

- Imprecisa - quando o estado relatado for correto, ainda que tenha uma aproximação imprecisa do verdadeiro estado;
- Errada - quando houver um desacordo entre o estado atual e o estado relatado da propriedade.

Várias combinações destas quatro classes de imperfeição são possíveis, ou seja, entre um conjunto de informações duvidosas, pelo menos algumas serão erradas ou imprecisas.

### Gerência e uso da informação de contexto

Primeiro, as informações de contexto são processadas por uma aplicação que suporta o ambiente ubíquo. Segundo, o sistema distribuído de gerenciamento de contexto assume um papel de integração, armazenando e gerenciando a informação das fontes, bem como gerando notificações ao usuário quando necessário.

Como exemplo de sistema que considere imperfeições na informação de contexto, pode-se citar o trabalho de Henriksen [35]. Nesse trabalho, o sistema que gerencia o contexto também fornece interfaces para os usuários fazerem atualizações e pesquisarem suas informações. Um conjunto de *plug-ins*<sup>1</sup> recebem e gerenciam a informação do contexto. Um *kit* de ferramentas permite que as aplicações conscientes de contexto explorem os serviços do sistema de gerenciamento usando os modelos de programação e as abstrações descritas por Indulska [36]. O problema das informações imperfeitas de contexto representa um obstáculo significativo para o sucesso das aplicações, embora ainda seja desconsiderado pelas mesmas.

Alguns dos autores pesquisados que abordam infra-estruturas de software, desenvolveram modelos para abordar o problema das informações imperfeitas. Schmidt [66] propôs uma arquitetura de software que possui um modelo de contexto representado por um vetor de pares. Cada par é constituído por um valor e uma probabilidade estimada. O modelo associa a ação do usuário com o contexto freqüentado por ele através de uma probabilidade pré-definida.

Judd [41] descreve uma interface genérica para pesquisar serviços de contexto que permitem aos usuários especificarem suas necessidades, como limites de precisão

---

<sup>1</sup>Um *plugin* ou *plug-in* serve para ativar funções em dispositivos lógicos ou físicos que antes não existiam ou não eram habilitadas.

da informação e tempo de atualização da mesma. Um serviço de contexto acessa as necessidades e responde às consultas indicando o nível de confiança das informações.

### **Informações de contexto em um cenário de aprendizagem ubíqua**

Para ativar a adaptação de um dispositivo à consciência de contexto, informações devem ser colhidas e eventualmente apresentadas à uma aplicação que a executa. Conseqüentemente, um formato comum de representação de informações de contexto se torna necessário. Held [34] propôs um formato para a representação da informação de contexto: CSCP, que é baseado em RDF. Considerando o uso de informações de contexto para apoiar processos de ensino e aprendizagem, CSCP estende mecanismos para expressar preferências de aprendizes.

Se incluem em informações relevantes de contexto as capacidades do dispositivo móvel, as características da rede de conexão e informações específicas do aprendiz. Com respeito aos dispositivos, informações relevantes de contexto incluem características básicas de hardware, tal como potência da CPU e memória, bem como a interface do aprendiz. Esta é caracterizada pelos dispositivos de entrada disponíveis (teclado, reconhecimento de caracteres, *voice*) e de saída (tela, áudio). Com respeito à plataforma de software dos dispositivos, considera-se que seus recursos são suficientes.

Uma das informações específicas de um aprendiz consiste em suas preferências. Estas também podem compreender preferências com respeito a seleção do serviço e aparências de serviço (tamanho de fonte, omissão de conteúdos multimídia e preferência de *frames*). As preferências do aprendiz podem variar dependendo das potencialidades do dispositivo e de condições de contexto. Conseqüentemente, o perfil do aprendiz deve fornecer meios para expressar condições aplicáveis para a preferência de atributos relacionados a dispositivos. Nem sempre todas as preferências de aprendizes serão satisfeitas. Desta forma, o aprendiz deve ser autorizado e orientado para especificar prioridades de preferência. Uma prioridade máxima significa que o serviço não deve ser fornecido se a preferência não é satisfeita. Além disso, prioridades diferenciadas permitem solucionar conflitos entre preferências e potencialidades dos dispositivos [34].

Além de preferências dos aprendizes, informações adicionais específicas são relevantes ao sistema: identificação do aprendiz, autenticação e subscrição de

informações, etc.

#### 2.1.4 Adaptabilidade

A questão da disponibilidade de serviços e dados é bastante importante para qualquer sistema que pretenda dar suporte às aplicações ubíquas. A adaptação de uma aplicação não é ligada apenas a modificações relativas ao contexto, mas também a alterações em sua lógica durante a execução. Uma das formas de modificar a lógica de uma aplicação pode ser através da mobilidade de código<sup>2</sup>. Entretanto, para que uma aplicação mantenha a sua consistência, precisam existir mecanismos que permitam ao código que se moveu, continuar acessando os mesmos dados e serviços que acessava antes de sua mudança de dispositivo.

Outras questões importantes são as estratégias de tolerância a falhas e a adaptabilidade. Ambientes ubíquos são bastante dinâmicos. Desta forma, torna-se necessário oferecer mecanismos para que um sistema seja capaz de suportar este tipo de situação, onde entidades podem deixar de existir, levando consigo dados e serviços que poderiam estar em uso no sistema.

Adaptabilidade é a habilidade de trocar informações com relação ao contexto utilizado [17, 34]. Duas principais categorias de adaptabilidade têm sido identificadas: *Adaptative Navigation* (AN) e *Adaptative Presentation* (AP). AN é a adaptação da estrutura de navegação com os recursos fornecidos à um usuário e AP é a adaptação da apresentação do recurso. AP e AN são técnicas aplicadas para a adaptação do contexto [17, 22]. Com o surgimento da computação ubíqua, ambientes *wireless* e *ad-hoc*<sup>3</sup>, ocorre a necessidade da adaptabilidade do dispositivo/terminal. Como exemplo de aplicação que necessita adaptabilidade, pode-se citar a chamada *e-Learning* via rede *wireless*, que é apresentada na próxima seção deste trabalho.

A idéia de sistemas que se adaptam ao seu ambiente não é nova. Na verdade, mobilidade implica em adaptabilidade [43]. Isto significa que os sistemas devem ter

---

<sup>2</sup>Mobilidade de código pode ser definida informalmente como a reconfiguração de maneira dinâmica da ligação de fragmentos de código com o local onde eles executam.

<sup>3</sup>A principal característica de uma rede Ad-hoc é o fato de não possuir infra-estrutura. Os nós que compõem uma rede sem fio desse tipo são capazes de se comunicar uns com os outros funcionando eles mesmos como roteadores [69, 79].

consciência do contexto aonde estão inseridos e obter proveito desta informação se estruturando de modo distribuído e se reconfigurando dinamicamente. Os desafios criados pela mobilidade ainda são foco de muitas pesquisas. Modelos, arquiteturas e tecnologias ainda se encontram em desenvolvimento [61].

Em resposta a estes desafios, a comunidade científica vem desenvolvendo trabalhos que buscam resolver estas questões [33]. Dessa forma, surgiram soluções que contemplam a gerência de aplicações com suporte à mobilidade lógica e/ou física (software e hardware).

## 2.2 Evolução das tecnologias da educação

### 2.2.1 Educação a distância

Com a rápida disseminação da informação, a defasagem do conhecimento e a necessidade de constante atualização de informações por parte de professores, a aprendizagem passa a ser uma necessidade contínua. Os profissionais de hoje devem aprender continuamente. A educação a distância mediada pelo computador, definida por alguns autores como *e-Learning* (aprendizagem eletrônica), refere-se à utilização das tecnologias da internet para finalidades educacionais [60]. Segundo Rosenberg [63], alguns dos benefícios do *e-Learning* são:

1. Diminuição de custos: é uma maneira econômica de fornecer informação, pois elimina despesas de viagem e reduz significativamente a necessidade de uma infra-estrutura de sala de aula;
2. Conteúdo personalizado: com as tecnologias atuais, é possível personalizar o conteúdo educacional para atender diferentes necessidades;
3. Flexibilidade de tempo: as pessoas podem acessar os recursos como material didático ou uma lista de discussão, a qualquer hora do dia;
4. Criação de comunidades: a internet permite que as pessoas criem comunidades duradouras, em que possam se reunir para compartilhar conhecimentos [25].

Mais do que uma alternativa ao ensino que obriga professores e alunos a se encontrarem no interior das quatro paredes da sala de aula (no mesmo espaço e

no mesmo tempo), o ensino a distância parece se apresentar hoje como resposta às necessidades emergentes de uma sociedade caracterizada por elevados níveis de competitividade em que o “tempo” é um fator crítico no desenvolvimento dos indivíduos e das instituições. O acesso ao conhecimento deve ser possível a qualquer momento e em qualquer lugar e, acima de tudo, quando é considerado necessário e oportuno [63].

### 2.2.2 Mobilidade aplicada à educação

O uso da computação móvel está cada vez mais presente na educação, visto que o ensino e a aprendizagem são cada vez mais apoiados por ferramentas móveis ou ocorrem em um contexto onde equipamentos móveis estão disponíveis [62]. Isso constitui a chamada *m-Learning* (aprendizagem com mobilidade). *M-Learning* pode ser definida como *e-Learning* usando dispositivos móveis, tal como PDAs, telefones móveis, *laptops* e *tablet PCs* [23]. O desafio que se apresenta é projetar aplicações educacionais em um ambiente onde os níveis de serviço e disponibilidade de recursos são imprevisíveis. Algumas propostas analisadas [57, 62, 70] são exemplos do uso de dispositivos móveis (celulares, *handhelds*, etc) no suporte à educação.

*E-Learning* e *m-Learning* começam a emergir de forma integrada como potenciais ambientes de aprendizagem. As tecnologias *mobile* e *wireless* produzem novos ambientes de aprendizagem que podem funcionar em qualquer momento e qualquer lugar.

*Mobile learning* acresce fundamentalmente a capacidade dos aprendizes de se moverem fisicamente em seus ambientes de aprendizagem. Os dispositivos móveis podem conectar a internet através da comunicação *wireless*, e ativar a aprendizagem em qualquer momento e em qualquer lugar. Entretanto, nesta situação, os computadores não estão embutidos no ambiente em que o aprendiz se encontra e não possuem flexibilidade para obter informações do contexto da aprendizagem.

Na aprendizagem *pervasiva* (*p-Learning*), os computadores podem obter informações sobre o contexto do ambiente de aprendizagem de pequenos dispositivos tais como sensores, *pads*, *badges*, etc, que são invisíveis aos aprendizes. Invisível no sentido de que o aprendiz não precisa se dar conta da tecnologia, ela pode estar embutida nos mais diversos dispositivos: computador pessoal, PDA, celular, etc. Os

dispositivos móveis podem executar modelos computacionais que estão embutidos no ambiente de *p-Learning*. Assim, um aprendiz pode perguntar, detectar e explorar o ambiente de acordo com as potencialidades do seu dispositivo. Entretanto, as potencialidades da *p-Learning* dependem altamente da localização.

Por fim, *u-Learning* estende o conceito da *p-Learning* em direção à mobilidade, ou seja, independente da localização, o aprendiz tem acesso aos mesmos recursos computacionais e serviços [57]. A *p-Learning* e a *m-Learning* pertencem a vasta categoria da *u-Learning* e, segundo Ogata [56], todos estes ambientes compõem a *e-Learning*, como representado na Figura 2.1.

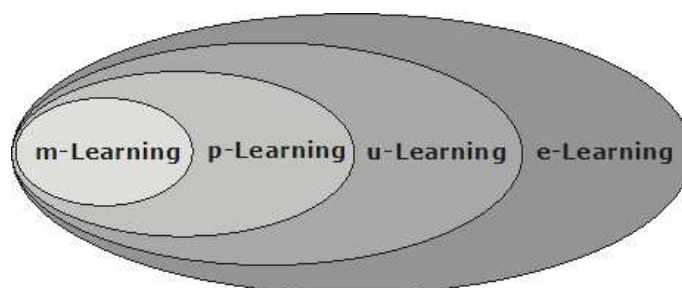


Figura 2.1 – Conjunto dos ambientes de *e-Learning*

A *u-Learning* (aprendizagem ubíqua) é definida como um ambiente de aprendizagem que é suportado por computadores que são invisíveis na vida diária [57]. As principais características de um ambiente *u-Learning* são: permanência, acessibilidade, imediação, interatividade e atividades instrucionais [42, 68].

Permanência significa que os aprendizes nunca perdem os seus trabalhos, a menos que sejam apagados propositalmente. Acessibilidade está relacionada ao fato de os aprendizes terem acesso aos seus documentos em qualquer lugar. Imediação significa que os aprendizes podem adquirir qualquer informação imediatamente, onde quer que estejam. Interatividade indica que os aprendizes podem interagir com peritos e professores das diferentes áreas do conhecimento. Atividades instrucionais significa que se pode fornecer ajuda para os aprendizes notarem as características que são pertinentes a problemas específicos.

A Figura 2.2 mostra a comparação dos quatro ambientes de aprendizagem [56]. A aprendizagem por *desktop computers* está embutida no mundo real, mas os computadores são movidos com dificuldade. Portanto, estes sistemas raramente suportam aprendizagem em qualquer tempo e qualquer lugar. A *m-Learning* é

superior se comparada a aprendizagem em *desktops*, pois fornece aos aprendizes a capacidade de se moverem fisicamente em seus próprios ambientes.

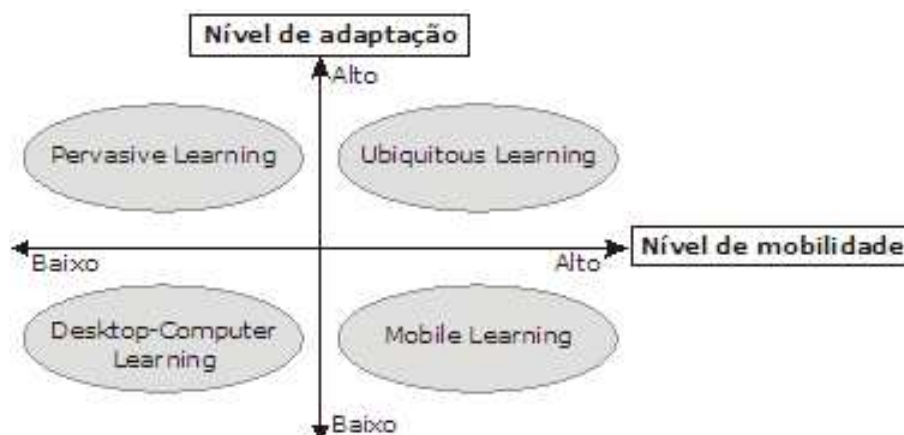


Figura 2.2 – Comparação de ambientes de aprendizagem [56]

## 2.3 Objetos de aprendizagem

Uma das formas de se fornecer conhecimento em um ambiente de aprendizagem é a pré-seleção de conteúdos, feita por professores ou tutores que conhecem um determinado assunto. Estes conteúdos podem ser disponibilizados através de objetos de aprendizagem (OA). Um objeto de aprendizagem [55] é qualquer entidade, digital ou não, que possa ser usada, reusada, ou referenciada durante a aprendizagem suportada por tecnologia [37]. Segundo Jacobsen [39], trata-se de uma coleção de material reutilizável utilizado para apresentar e dar apoio a um único OA ou de um pequeno “componente instrucional” que pode ser utilizado para suportar a aprendizagem em ambientes diferentes.

Pode-se concluir que um OA é um módulo, unidade, capítulo ou lição que se pretende ensinar um conceito, uma idéia, um fato, um procedimento, um processo ou um princípio [24]. Tendo este princípio de componentes, vários ambientes de ensino diferentes podem utilizar o mesmo OA, por meio de sistemas de gerência de aprendizagem (*Learning Management System* - LMS) ou sistemas de gerência de conteúdos de aprendizagem (*Learning Content Management Systems* - LCMS).

Independentemente do tipo de aplicação educacional, os OAs apresentam as seguintes características [29, 47]:



- **Reusabilidade:** Um OA pode ser usado em diversos materiais de aprendizagem diferentes (disciplinas diferentes, por exemplo);
- **Facilidade de pesquisa e atualização:** Com a utilização de padrões de metadados<sup>4</sup> existentes é possível obter informações sobre os OAs, tais como seu conteúdo, utilização, autor, tamanho, formato, etc, tornando-o compreensível para diversas plataformas computacionais;
- **Modularidade:** Um OA pode conter outros OAs, ou estar contido em um ou mais materiais de aprendizagem;
- **Interoperabilidade:** Um OA pode ser utilizado em diversos tipos de hardware, sistemas operacionais, navegadores e ambientes de aprendizagem.

## 2.4 Estilos de aprendizagem

Para muitos autores, aprendizagem é um processo natural na vida de todo ser humano. Para Skinner [67] a aprendizagem seria basicamente uma mudança de comportamento, ou seja, a pessoa demonstra saber algo que não sabia antes. Para David Ausubel [5], psicólogo da aprendizagem, o principal no processo de ensino é que a aprendizagem seja significativa. Pode-se perceber como as pessoas da mesma classe, grau, idade, nacionalidade, raça, cultura ou religião preferem aprender de maneiras diferentes, ou seja, possuem preferências na hora de aprender [21].

Preferência ou estilo de aprendizagem é o modo como cada aprendiz se concentra, processa e retém informação. As características pessoais fazem com que o mesmo método de ensino seja efetivo para alguns e ineficaz para outros aprendizes [13]. Embora alguns aprendizes possam aprender com eficiência sem que suas preferências de aprendizagem sejam consideradas, outros alcançam uma performance significativamente melhor quando suas preferências são capitalizadas.

Felder e Silverman [27] propuseram um modelo de estilos de aprendizagem baseado em cinco dimensões de aprendizagem: percepção, entrada, organização, processamento e compreensão. Segundo o modelo, as dimensões possuem dicotomias: percepção (sensorial/intuitiva), entrada (visual/aural),

---

<sup>4</sup>Metadados são dados estruturados que descrevem as características de um recurso.

organização (indutiva/dedutiva), processamento (ativo/reflexivo) e compreensão (seqüencial/global).

O indivíduo que possui estilo visual de entrada dá preferência à informação em forma de figuras, filmes, símbolos, gráficos, mapas, demonstrações, etc, ao passo que o possuidor do estilo aural consegue tirar maior proveito da palavra falada ou escrita [10]. A percepção sensorial ocorre fundamentalmente através dos sentidos ao passo que a percepção intuitiva através de informação do inconsciente sob a forma de especulação e imaginação.

A organização indutiva é aquela que parte do particular para o geral, onde teorias unificadoras são resultados do estudo de situações individuais. Na organização dedutiva as soluções e aplicações particulares são conseqüências de uma idéia geral. A indução é o estilo natural de aprendizado sendo também a maneira pela qual o conhecimento é, predominantemente, adquirido ao longo do tempo por estudiosos de determinado assunto. Por outro lado a dedução é o estilo natural de ensino de assuntos técnicos e de organização do conhecimento já adquirido.

A informação percebida pelo indivíduo é transformada em conhecimento através de dois modos: a experimentação ativa e a observação reflexiva. A experimentação ativa envolve o uso externo da informação seja através de discussão, explicação ou de teste por aplicação a uma situação concreta; a observação reflexiva consiste em exame e manipulação mental da informação [21].

O perfil do estilo de aprendizagem de um aprendiz fornece uma indicação dos prováveis pontos fortes e possíveis tendências ou hábitos que poderiam conduzir à dificuldades na vida acadêmica. Cabe lembrar que o perfil levantado do aprendiz não reflete a sua adequação ou inadequação para uma determinada disciplina, curso ou profissão, mas ajuda a melhorar a aprendizagem, na medida em que o professor não precisa propor só atividades que vão ao encontro do estilo preferencial de suas turmas, mas pode escolher métodos instrutivos incompatíveis, a fim de poder estimular e fortalecer as habilidades menos desenvolvidas [49].

Após estudos, Felder [26] fez duas importantes mudanças no modelo de estilos de aprendizagem. Foi efetuada a eliminação da dimensão organização e alteração da categoria “visual/aural” para “visual/verbal” [21]. A Tabela 2.1 apresenta as dimensões propostas por Felder.

<b>Dimensões de estilos de aprendizagem</b>
Sensorial intuitiva ⇒ <b>percepção</b>
Visual verbal ⇒ <b>entrada</b>
Ativo reflexivo ⇒ <b>processamento</b>
Seqüencial global ⇒ <b>compreensão</b>

Tabela 2.1 – Dimensões de estilos de aprendizagem [26]

Estas são as dimensões de estilos de aprendizagem que serão utilizadas neste trabalho, sendo que cada uma delas será chamada apenas de “estilo de aprendizagem”, no intuito de possibilitar um texto mais objetivo. Os estilos de aprendizagem serão citados como: **Percepção**, **Entrada**, **Processamento** e **Compreensão**. Apesar de as deduções de Felder [26] já serem antigas, elas foram utilizadas nesta dissertação por ainda estarem muito difundidas no contexto de pesquisa atual. O modelo PeLeP não se restringe à utilização desses estilos de aprendizagem.

## 2.5 Perfil de aprendiz

Um perfil é uma descrição do aprendiz. Ele pode conter preferências, interesses, relacionamentos e opiniões. Como em um ambiente ubíquo há vários contextos, o perfil do aprendiz poderá mudar conforme o contexto em que ele se encontra e conter qualquer informação que seja importante para a sua identificação dentro daquele contexto [71]. Nestas informações estão inclusas as relações do aprendiz com os dispositivos móveis utilizados por ele.

Um típico perfil de usuário consiste em informações que descrevem [19]:

- informações pessoais (por exemplo, informação de contato, informação de emprego, associações profissionais e preferências);
- as relações sociais com outros usuários (por exemplo, quem são seus amigos e com quem ele trabalha);
- o perfil dos dispositivos móveis utilizados e agentes pessoais (por exemplo, o tipo de interfaces de comunicação e as resoluções da tela de exibição de um dispositivo, *ID* de um agente pessoal);

- os encontros diários em horário marcado, associados com preferências nos mesmos (por exemplo, o tempo e localização de uma reunião que o usuário planeja assistir, os *slides* de uma apresentação que ele usa freqüentemente).

Informações típicas que podem constituir um perfil incluem: nome e endereço; telefone; *e-mail*; empresa que a pessoa trabalha; agenda ou lista de tarefas; lista de amigos e relacionamentos; informações de emergência (saúde); lista de *web-sites* favoritos; ID (identificação); títulos de livros e autores preferidos; lista de publicações feitas; palavras-chave que identificam interesses do usuário.

## 2.6 Considerações finais

Este capítulo apresentou as tecnologias necessárias para um ambiente de aprendizagem ubíquo. Foram enfatizados os recursos necessários para a manipulação da informação de contexto, quando um aprendiz se desloca no ambiente ubíquo. Percebe-se que há fatores que podem alterar a precisão das informações de contexto, mas que já existem soluções para tal problema. Mostrou-se que a informação de contexto é significativa em relação à adaptabilidade dos dispositivos móveis. Por fim, discorreu-se sobre as definições mais aceitas para objetos de aprendizagem e estilos de aprendizagem.

Se os dispositivos portáteis com conexão a internet via rede *wireless* se popularizarem, e se as suas potencialidades estiverem ativadas enquanto os aprendizes se moverem, um novo tipo de “educação a distância” pode emergir [62]. Após saírem de um curso, equipes de estudantes serão capazes de coordenar o andamento de um grupo de trabalho de maneira mais próxima. Podem encaixar na coordenação de programas o envio de cada informação atualizada e fazer perguntas e questionamentos entre eles, tudo de lugares diferentes de suas cidades ou bairros. Portanto, a utilização desses novos padrões de comunicação pode crescer em interesse e importância não apenas em aplicações educacionais, mas no mundo empresarial, para reunir indivíduos ou organizações com interesse comuns.

A educação a distância se beneficiará da tecnologia em desenvolvimento para ambientes de aprendizagem ubíquos. O suporte à mobilidade proporcionará que os aprendizes acessem os sistemas em qualquer lugar e qualquer momento, quando a

conexão estiver disponível. Isso causará uma revolução nas tecnologias para suporte à educação e a necessidade de novos ambientes de aprendizagem.

## Capítulo 3

# Trabalhos Relacionados

Conforme as pesquisas realizadas, verificou-se que muitos sistemas utilizam o perfil do aprendiz em aplicações. Porém, o uso desse perfil para auxílio no processo de aprendizagem em um cenário ubíquo é recente. Atualmente, trabalhos como JAPELAS [77, 78] e GlobalEdu [7, 8, 9] focam este cenário. Neste capítulo, aplicações existentes para representação de perfis de aprendizes serão introduzidos e suas limitações discutidas.

### 3.1 CC/PP

CC/PP [18] aborda a estruturação e manutenção de perfis de usuários que utilizam dispositivos conectados via rede *wireless*. Embora não tenha como foco um ambiente de aprendizagem ubíqua, possui aspectos importantes na estruturação dos perfis de usuários e na adaptação destes aos contextos em que se encontram.

CC/PP foi criado para transmitir perfis e preferências de usuários aos dispositivos que se adaptam a alguns *web-services* [3]. O trabalho dos *web-services* é interpretar os perfis e baseando-se neles, instanciar comunicações com os dispositivos. Uma das principais vantagens do CC/PP é que ele não especifica como os recursos de um candidato são descritos.

CC/PP [44] utiliza uma linguagem para a representação de perfis proposta pelo W3C<sup>1</sup>. CC/PP tenta expressar potencialidades de dispositivos e preferências de usuários. As especificações de CC/PP definem uma estrutura básica para perfis.

---

<sup>1</sup>O *World Wide Web Consortium* (<http://www.w3.org>), é um consórcio de empresas de tecnologia que desenvolve padrões para a criação e a interpretação dos conteúdos para *Web*.

Um perfil é basicamente construído com dois níveis hierárquicos restritos: cada perfil possui um número de componentes, e cada componente possui um número de atributos. Os componentes particulares não são definidos pelas especificações CC/PP. A definição de um vocabulário específico atinge outros modelos de padronização. Vocabulários são definidos usando um esquema RDF [16].

CC/PP não considera aperfeiçoamento de perfis de usuários. Uma amostra de vocabulário para descrever perfis em dispositivos (WAP) é definida pelo WAG UAProf [72] no WAP fórum. Entretanto, este vocabulário se adapta às necessidades dos dispositivos WAP, mas não para dispositivos mais complexos como *notebooks* [34]. Além disso, não há meios para expressar as características da conexão de rede corrente. A estrutura do WAG UAProf é bastante desordenada. E ainda, as potencialidades dos dispositivos e preferências do usuário são entrelaçadas, complicando a manutenção dos perfis.

## 3.2 CSCP

CSCP [34] foi analisado pela forma como os contextos são tratados e pelo fato de os autores expressarem a informação de contexto através de sessões de perfis. Estes perfis armazenam inclusive os perfis dos dispositivos móveis dos usuários. Embora não tenha sido desenvolvido para ambientes de aprendizagem ubíqua, CSCP aborda a adaptação dos dispositivos ao contexto em que eles se situam. CSCP não considera aperfeiçoamento dos perfis de seus usuários.

Os princípios usados em CSCP expressam informação de contexto por meio de sessão de perfis. Uma sessão de perfil do usuário é anexada a uma sessão de contexto no portal de mobilidade, que é o ponto central de acesso à informação de serviços para usuários móveis. Pela avaliação da informação de contexto e pelo gerenciamento de perfis de usuários, o portal de mobilidade do CSCP permite a mediação entre a identificação de contexto e adaptação de um dispositivo móvel ao conteúdo do contexto.

Uma sessão de perfil descreve todas as informações relevantes da sessão de contexto. Isto inclui: o perfil do dispositivo, o perfil da rede, perfil do usuário, e possivelmente outras informações de contexto, tal como informações do ambiente (localização, informação de sensores, etc). O perfil da sessão é

inicialmente montado pelo usuário e transferido ao portal durante o estabelecimento da sessão. Entretanto, isso não significa que toda a informação de contexto deve ser colhida do dispositivo móvel do usuário e transferida via *link wireless*. O perfil da sessão pode preferivelmente conter referências a recursos externos, tal como o perfil do usuário armazenado no portal.

Diferentemente de CC/PP, a linguagem CSCP não define qualquer hierarquia fixa. Prefere suportar toda a flexibilidade do formato RDF, que também é constituída para expressar as estruturas naturais da informação do perfil. Nomes de atributos são sensíveis ao contexto e interpretados de acordo com suas posições na estrutura do perfil. Conseqüentemente, atribuir nomes para o perfil inteiro (como necessário com CC/PP) não é necessário.

CSCP pode ser decomposto por meio de padrões e referências externas. Referências externas são utilizadas para extrair sub-perfis que separam documentos CSCP. Por exemplo, o perfil de um usuário pode ser extraído de uma sessão de perfil e armazenado em um documento CSCP separado no portal de mobilidade [34]. Além disso, os mecanismos padrão são utilizados para propagar atualizações de perfil.

CSCP fornece características para acoplar condições e prioridades nos atributos. Isto se estende para expressar preferências do usuário. Desta forma, variações nessas preferências dependem das potencialidades do dispositivo ou de outras condições de contexto, que podem ser especificadas. Além disso, a linguagem CSCP contém um vocabulário para expressar sessões de perfis incluindo o perfil do dispositivo, perfil de rede e perfil de usuário.

### 3.3 PAPI

PAPI [59] é um padrão criado para estruturar a informação pública e privada de aprendizes. Ele contém uma especificação para troca de dados que descrevem a informação para comunicação entre sistemas cooperativos. Cooperação significa que há concordância com os padrões definidos pelo PAPI ou com outras especificações. Os diferentes tipos de informações consideradas constituem a informação do perfil (*Profile Information*) do PAPI.

PAPI não considera aperfeiçoamento de perfis de aprendizes. Seu modelo de aprendiz segue os padrões IEEE 1484.2, e estrutura o perfil com as seguintes



categorias: *Learner Contact Information*, que contém as informações de contato do aprendiz; *Learner Relations Information*, que reúne as relações do aprendiz com outros aprendizes e com seus dispositivos; *Learner Security Information*, que armazena as informações de sua segurança (senhas, chaves de criptografia); *Learner Preference Information*, que contém as suas preferências de interface homem-computador; *Learner Performance Information*, que reúne as suas competências; e *Learner Portfolio Information*, onde são reunidos seus objetos de aprendizagem e informações relacionadas à certificados do aprendiz.

### 3.4 LIP

LIP [46] organiza as informações do perfil do aprendiz através das seguintes categorias: *Identification*, que administra os seus dados de identificação; *Goal*, onde são armazenados os seus objetivos; *Qcl* (*Qualifications, Certifications and Licences*), que administra as suas qualificações, diplomas e certificados; *Activity*, que possui qualquer atividade relacionada ao seu aprendizado concluída ou em andamento, incluindo educação formal e informal, treinamento, etc; *Transcript*, onde é administrado o histórico das atividades acadêmicas realizadas por ele em instituições; *Interest*, que descreve seus *hobbies* e interesses; *Competency*, onde são reunidos os conhecimentos e competências já desenvolvidos; *Affiliation*, que relaciona os membros de organizações profissionais; *Accessibility*, que armazena dados sobre acesso à informações, idiomas e preferências físicas, tecnológicas e cognitivas (estilos de aprendizagem); *Security key*, que administra informações sobre segurança; e *Relationship*, onde estão contidas as relações do aprendiz com as categorias citadas acima.

LIP não considera aperfeiçoamento de perfis de aprendizes e um dos seus maiores problemas, segundo Musa [53], é apresentar categorias que se sobrepõem, causando conflitos. Como por exemplo, as categorias *Activity*, *Qcl*, *Transcript* e *Competency*. Os aprendizes que utilizam o LIP geralmente têm dúvidas sobre que dados pertencem a quais categorias. As categorias do LIP, juntamente com seus elementos, são descritas através de um esquema XML.

### 3.5 SeLeNe

O modelo SeLeNe [45, 58] oferece serviços para descoberta, compartilhamento e criação colaborativa de objetos de aprendizagem, de acordo com o projeto de aprendiz em ambientes *e-Learning*. O sistema é um *Learning Management System* e interliga repositórios de objetos de aprendizagem, distribuídos pela *Web*, através de uma rede P2P (*peer-to-peer*), também denominada SeLeNe. Ele oferece a possibilidade de realizar pesquisa, recuperação e criação colaborativa de objetos de aprendizagem. A arquitetura possui provedores de serviços, que contém os repositórios de objetos educacionais e aprendizes. Uma vez que se constitui de uma rede de aprendizagem, permite aos aprendizes compartilhar serviços, recursos e dados.

SeLeNe utiliza os padrões PAPI e LIP para modelar o aprendiz, sendo que a abordagem híbrida é mais evidente. Ele considera, além dos padrões citados, o perfil cognitivo e de aprendizagem do aprendiz, mas não o aperfeiçoamento do perfil. O modelo apresenta em sua arquitetura a personalização de resultados de consultas de aprendizes através de módulos de perfis: *learner*, *author* e *class profiles*. O modelo de aprendiz, assim como de objetos de aprendizagem, também é gerenciado por metadados descritos em RDF. O modelo do aprendiz é composto de informações explícitas, fornecidas pelo usuário, e implícitas, coletadas de forma transparente pelo sistema. SeLeNe manipula também modelos de grupos de aprendizes (aprendizes com os mesmos objetivos pedagógicos).

SeLeNe considera que o processo de ensino e aprendizagem se dá por adaptação à conteúdo e serviços distribuídos em diversos repositórios, com foco em ensino genérico. O modelo direciona e adapta a procura por recursos. Uma vez encontrado um recurso, o aprendiz o acessa diretamente. SeLeNe oferece também a possibilidade de avisar o aprendiz sobre determinados eventos, quando estes ocorrem. Estes eventos dizem respeito à atualização e inserção de novos objetos educacionais.

SeLeNe possui um serviço de *Trails* que analisa o perfil do aprendiz e o histórico das consultas feitas por ele no ambiente *e-Learning*. SeLeNe possui três classes de *Trails*: autorizadas (*authored*), derivadas (*derived*) e emergentes (*emergent*). Estas podem ser definidas pelo aprendiz (*authored trails*) ou geradas automaticamente pelo sistema (*emergent trails*). Quando geradas automaticamente, as *Trails* são

derivadas de vínculos entre OAs, inferidos de suas descrições (*derived trails*). Através da análise desse histórico, é retornada uma lista de objetos de aprendizagem de acordo com as necessidades do aprendiz. Por não ser um modelo desenvolvido para o ambiente ubíquo, não oferece serviço de consciência do contexto, nem considera a mobilidade do aprendiz.

### 3.6 *Proem*

*Proem* [71] foi analisado por apresentar uma cooperação entre usuários baseada na combinação de seus perfis com a consciência de seus contextos. O modelo possui uma ferramenta denominada *Profile Editor*, onde o próprio usuário define o seu perfil. Este contém uma estrutura de dados que se assemelha à uma árvore, contendo informações sobre o usuário. Os perfis dos usuários não são aperfeiçoados através da análise de seus históricos. Os elementos do perfil são organizados através de pares valor-atributo. Os valores dos atributos são *strings* ou listas de *strings* separadas por vírgula. Por exemplo, o valor do atributo *fullname* é uma *string*, ‘*James Joyce*’, mas o valor do atributo *interests* é uma lista de *strings*, como ‘*computacao, fisica, musica*’.

O modelo inicia o contato entre usuários através da identificação de interesses mútuos entre eles pela análise de seus perfis, que são armazenados em arquivos XML. A seção *friend* do perfil contém três atributos separados: *FirstDegree*, *SecondDegree* e *ThirdDegree*. O valor de cada um desses atributos representa amigos de primeiro, segundo e terceiro graus. Amigos de segundo grau, são amigos dos amigos do usuário. Amigos de terceiro grau, são amigos dos amigos dos amigos do usuário. O usuário só especifica os amigos mais próximos, ou seja, os de primeiro grau. Os valores para os atributos que representam amigos de segundo e terceiro graus são colhidos automaticamente pelo sistema.

### 3.7 JAPELAS

JAPELAS [77, 78] é um sistema consciente de contexto para suporte ao ensino de expressões de tratamento da língua japonesa. Portando PDAs, os aprendizes são auxiliados pelo sistema a identificar a expressão de tratamento adequada à outra

pessoa levando em consideração o contexto em que eles se encontram. O módulo *Learner model* contém o perfil dos aprendizes tal como nome, idade, ocupação, interesses, etc, e o nível compreensível de cada expressão. A entrada dos dados é feita pelos aprendizes. O sistema identifica o aprendiz e acrescenta informações em seu perfil, no ambiente de aprendizagem ubíquo. Mas o perfil do aprendiz não é aperfeiçoado através da análise do seu histórico no ambiente.

Na apresentação do modelo de informações os autores destacam que pessoas em torno do aprendiz influenciam a aprendizagem da língua japonesa, pois pessoas japonesas utilizam diferentes níveis de expressões cultas de acordo com suas idades. Por isso, o sistema não identifica somente o aprendiz, mas também outros aprendizes que o cercam em um mesmo contexto. O sistema fornece a informação depois da interpretação do modelo de aprendiz, que possui inclusive informações sobre recursos dos dispositivos móveis (potência, memória, etc) dos aprendizes.

### 3.8 GlobalEdu

O GlobalEdu [7, 8, 9] é uma arquitetura de aprendizagem que suporta aplicações educacionais considerando um ambiente de computação ubíqua em larga escala. A arquitetura do GlobalEdu possui três camadas: A3P, SE e ACP. O Agente Pedagógico Pessoal *Pervasivo* (A3P) está sempre com o aprendiz, movendo-se para o dispositivo utilizado por ele, auxiliando-o no processo educacional em um ambiente ubíquo. O comportamento do A3P é definido pelo modelo do aprendiz, seu contexto, e pelas informações educacionais acessadas por ele (Objetos de Aprendizagem). Os Serviços Educacionais (SE) fornecem o suporte para a execução do A3P em um ambiente ubíquo de larga escala, pela identificação e adaptação dos recursos conforme o modelo educacional de aprendiz.

A Arquitetura Computacional *Pervasiva* (ACP) atualmente utilizada pelo GlobalEdu é a arquitetura ISAM [4, 38]. Desta forma, o GlobalEdu utiliza serviços do ISAM para obter informações de contexto físico, acessar recursos disponíveis, gravá-los, etc.

O suporte ao processo de ensino e aprendizagem no GlobalEdu é baseado no perfil do aprendiz. O GlobalEdu especifica um modelo de aprendiz, onde define um conjunto de informações que são consideradas quanto ao seu histórico no ambiente

ubíquo (categoria *Trail* do modelo) [8]. Embora isso, a proposta não especifica como o comportamento do aprendiz nos diversos contextos, armazenados em seu histórico, infere alterações em seu modelo.

### 3.9 LOCAL

LOCAL [11, 12] usa informações de localização e de contexto como auxílio ao processo de ensino e de aprendizagem. Um sistema de localização acompanha a mobilidade dos aprendizes e, baseado nas suas posições físicas em um ambiente de aprendizagem ubíquo, explora oportunidades educacionais. LOCAL propõe o uso local (pequena escala) dos conceitos introduzidos pelo modelo global de educação ubíqua GlobalEdu.

A arquitetura LOCAL é formada por seis componentes, que estão representados na Figura 3.1. O primeiro é um sistema de perfis de aprendiz, que armazena dados relevantes ao processo de ensino e de aprendizagem e é baseado no padrão PAPI [59]. Neste componente existem informações que acompanham o aprendiz e ficam armazenadas em seu dispositivo móvel e também informações ligadas aos contextos por onde o aprendiz se desloca. O LOCAL gerencia essas informações para exploração de oportunidades pedagógicas, mas os perfis dos aprendizes não são aperfeiçoados através da análise de seus históricos no ambiente ubíquo.

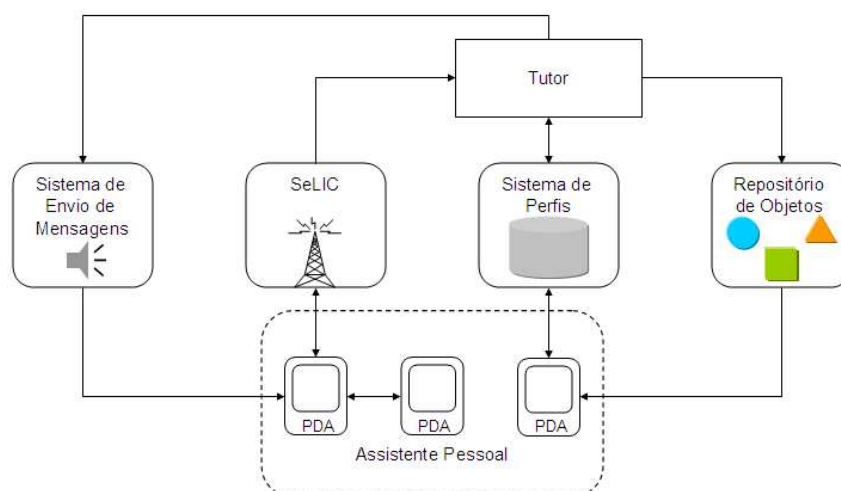


Figura 3.1 – Arquitetura do LOCAL

O segundo componente é um servidor de localização e informação de contexto (SeLIC), que é baseado em uma arquitetura genérica que suporta diferentes técnicas para determinação da posição física de um aprendiz. O sistema vincula informações de localização física com contextos, permitindo o mapeamento em tempo real do deslocamento de um dispositivo móvel. O aprendiz autoriza sua localização e o LOCAL registra todas as suas mudanças de contexto, inclusive com o horário de entrada e saída. Tendo como base essas informações é criado um *tracking* do aprendiz.

O terceiro componente do LOCAL é um Assistente Pessoal (AP) que acompanha o aprendiz, executando em seu dispositivo móvel. O quinto é um sistema de envio de mensagens contextuais e o sexto é um motor de análise (tutor) que realiza inferências usando dados fornecidos pelos sistemas de perfis e de localização.

O quarto componente do LOCAL é um repositório de Objetos de Aprendizagem (OAs), que armazena e indexa o conteúdo relevante ao processo pedagógico. Os OA são cadastrados e indexados através dos seguintes atributos: (1) ID; (2) nome; (3) tamanho do arquivo; (4) duração dos arquivos de mídia, quando o tamanho do arquivo não é descritivo o suficiente; (5) palavras-chave associadas ao objeto, facilitando assim a indexação e a pesquisa; (6) resumo do assunto abordado; (7) tipo de arquivo, para decidir que tipo de software pode abri-lo; (8) área do conhecimento, de acordo com ACM [2]; (9) data de cadastramento; (10) data de expiração; (11) data e hora do último acesso; (12) número de vezes que o OA foi acessado; (13) nível de interatividade (baixo, médio, alto); (14) pré-requisitos, que representam os OA que devem ser acessados antes do atual; (15) objetivos educacionais; (16) nível de dificuldade associado (baixo, médio, alto); (17) comentários sobre o OA; (18) origem (se originário do LOCAL ou importado de outro repositório); (19) uma referência para a manifestação física do objeto (como um endereço *Web* para um *download*); (20) Estilo de Aprendizagem (EA) à que é indicado, segundo Felder [26]; (21) tipo, se é recomendado pelo professor (OArp) ou suplementar (OAsup), sem indicação do professor. Todas essas propriedades são inseridas quando o OA é cadastrado.

O sistema encontra-se em desenvolvimento na Unisinos, onde abrange nove salas (contextos) do prédio onde está localizado o MobiLab [52]. Embora o LOCAL já contenha os módulos básicos para o suporte pedagógico usando consciência de

contexto, possui um sistema de perfis simples, que aborda poucas informações acerca dos aprendizes.

### 3.10 Considerações finais

Um resumo dos tópicos importantes a serem considerados no PeLeP é exibido na Tabela 3.1. Dos modelos analisados para representação do perfil, o GlobalEdu e o LOCAL são os que mais se aproximam dos objetivos do modelo PeLeP por abordarem histórico do aprendiz, representação do seu perfil e por suportarem o ambiente ubíquo. Por isso, o PeLeP utilizará os conceitos de estruturação de perfis do GlobalEdu e os implementará no LOCAL. Nenhum dos sistemas analisados considera aperfeiçoamento de perfis com base no histórico dos aprendizes. Apenas três modelos consideram o histórico do aprendiz: SeLeNe, GlobalEdu e LOCAL. Porém, o fato de o modelo SeLeNe ser produzido para sistemas hipermídia reforça a opção por GlobalEdu e LOCAL, que se adaptam a ambientes ubíquos.

Sistema	Estrutura do perfil	Histórico ( <i>tracking</i> )	Dados	Ambiente	Aperfeiçoamento de perfil
CC/PP	RDF	—	RDF	ubíquo	—
CSCP	RDF	—	RDF	ubíquo	—
SeLeNe	Ontologias	✓	RDF	<i>Web</i>	—
<i>Proem</i>	RDF	—	XML	ubíquo	—
JAPELAS	TANGO	—	RFID	ubíquo	—
GlobalEdu	XML	✓	XML	ubíquo	—
LOCAL	MySQL	✓	MySQL	ubíquo	—

Tabela 3.1 – Avaliação dos principais modelos pesquisados

# Capítulo 4

## Modelo PeLeP

Este capítulo apresenta o modelo PeLeP. São apresentados seus princípios e sua arquitetura. Mostra-se como as informações do perfil do aprendiz foram estruturadas. Apresenta-se as categorias utilizadas para a organização dos perfis. Discorre-se sobre os aspectos importantes para mapear o deslocamento do aprendiz no ambiente ubíquo. Por fim, apresenta-se o que é utilizado para realizar inferências nas categorias do perfil.

### 4.1 Princípios do PeLeP

PeLeP (*Pervasive Learning Profile*) é uma proposta de modelo de aprendiz voltado para sistemas de suporte à aprendizagem ubíqua. O modelo foi desenvolvido para representar, manter e gerenciar perfis de aprendizes em um ambiente de ensino e aprendizagem. Sua função é organizar o perfil de maneira que se possam inferir informações baseando-se no histórico do aprendiz. Percebe-se que ambientes para suporte aos processos de ensino e aprendizagem em um cenário virtual consideram os perfis dos aprendizes [30, 31, 45, 58, 62, 77, 78]. Porém, essas propostas não os aperfeiçoam com base no histórico do aprendiz.

O PeLeP cria um perfil de aprendiz com dados separados por categorias, que incluem questões de privacidade, segurança, preferências, objetivos educacionais, competências, relacionamentos e agenda. As informações de identificação e segurança são manipuladas pelo próprio aprendiz, enquanto que as demais são acrescentadas ou alteradas pelo PeLeP através da inferência baseada no *tracking*<sup>1</sup> daquele aprendiz.

---

<sup>1</sup>*Tracking* é o histórico do aprendiz no ambiente de aprendizagem ubíquo. Ele contém os



## 4.2 Arquitetura

A Figura 4.1 mostra a arquitetura do PeLeP. O modelo proposto não gera informações de localização e de contexto. Por isso, o PeLeP foi padronizado para ser conectado a um Sistema ubíquo que as fornece. As Informações de *tracking* não ficam armazenadas no modelo, apenas são requisitadas pelo PeLeP. Esta requisição é respondida pelo Sistema de Localização, que é o componente do Sistema ubíquo responsável pelo fornecimento dessas informações.

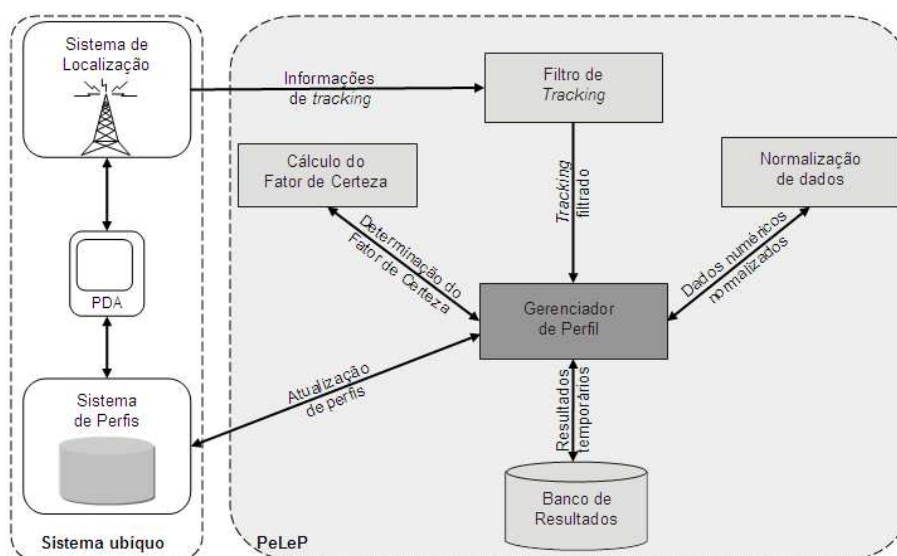


Figura 4.1 – Arquitetura do PeLeP

Para aumentar o grau de confiança dos dados que são gerados para o perfil do aprendiz é necessária a análise de uma grande quantidade de informações. Para o *tracking* reunir esse conjunto de informações acerca das atividades realizadas e contextos visitados pelo aprendiz no ambiente ubíquo, o PeLeP foi programado para requisitá-lo em períodos fixos de tempo, escolhidos de acordo com as necessidades educacionais dos aprendizes. Apenas para a categoria *Agenda* não existe tempo de execução. As informações que o Sistema ubíquo transmite para essa categoria não estão contidas no *tracking*.

A organização das informações recebidas do *tracking* é executada pelo módulo Filtro de *Tracking*. Este componente filtra os dados presentes nos repositórios XML enviados pelo Sistema de Localização, organizando-os para as operações posteriores do PeLeP. A normalização dos dados numéricos é de responsabilidade do módulo contextos visitados e as tarefas executadas dentro dos mesmos.

Normalização de dados. Um fator de certeza (FC) é calculado pelo componente Cálculo do Fator de Certeza para determinar o percentual de certeza dos elementos avaliados. Estes elementos constituem as categorias do perfil do aprendiz. O módulo Gerenciador de Perfil é encarregado de: 1) solicitar os dados filtrados pelo componente Filtro de *Tracking*; 2) transmitir os dados ao módulo Normalização de dados e enviá-los, normalizados, ao módulo Cálculo do Fator de Certeza; 3) armazenar os itens que possuem os maiores FCs no Banco de Resultados, que contém um resumo dos resultados da análise do *tracking*; 4) comparar o resumo com as informações presentes em Sistema de Perfis, que contém a representação e o armazenamento das categorias que constituem o modelo de aprendiz; 5) atualizar o Sistema de Perfis.

A comunicação com os dispositivos móveis é de responsabilidade do Sistema de Localização. Pela sua natureza dinâmica, esses dispositivos possuem conexão intermitente. Por isso, nenhuma informação do perfil fica armazenada nos mesmos. Além disso, o perfil precisa ser analisado. Portanto, para cumprir esta tarefa são utilizados recursos da rede ubíqua mais robustos, com um *desktop* por exemplo.

### 4.3 Descrição do perfil do aprendiz

As informações que constituem o perfil do aprendiz são armazenadas no banco de dados do Sistema ubíquo. O mesmo corresponde ao componente Sistema de Perfis na Figura 4.1. A proposta do modelo de aprendiz do GlobalEdu [7, 8, 9] considera o uso integrado de padrões PAPI [59] e LIP [46], além de informações sobre Estilos de Aprendizagem (*learning styles*) e histórico (*Trail*) do aprendiz. Além disso, o modelo estrutura o seu perfil com as seguintes categorias: *Personal Information*, *Preferences*, *Relations*, *Security*, *Learning Styles*, *Goal*, *Competence* e *Trails*.

Com base no GlobalEdu, o perfil do aprendiz no PeLeP é populado com as seguintes categorias: Identificação, Objetivos, Preferências, Segurança e Competências. Diferentemente do GlobalEdu, são propostas as categorias Relacionamentos e Agenda. Essas categorias estão discriminadas na Figura 4.2.

A categoria Identificação é válida para todos os contextos que o aprendiz visita. Ela contém as informações básicas para a sua identificação no sistema, sendo estas: 1) *IdAprendiz*: identificador único; 2) *Nome*: nome completo; 3) *Endereço*: endereço completo (rua, número, bairro, cidade); 4) *Email*: endereço eletrônico; 5) *Telefone*:

Modelo de Aprendiz do PeLeP			
Identificação	<i>IdAprendiz</i>	Relacionamentos	<i>IdAprendiz</i>
	<i>Nome</i>		<i>ContextoRelacao</i>
	<i>Endereco</i>		<i>Relacao</i>
	<i>Email</i>		<i>Data</i>
	<i>Telefone</i>		
Objetivos	<i>IdAprendiz</i>	Competências	<i>IdAprendiz</i>
	<i>Objetivo</i>		<i>CompetenciaGeral</i>
	<i>Descrição</i>		<i>CompetenciaEspecificica</i>
	<i>Contexto</i>		<i>Contexto</i>
	<i>Data</i>		<i>Data</i>
Agenda	<i>IdProfessor</i>	Preferências	<i>IdAprendiz</i>
	<i>TipoAtividade</i>		<i>EstiloAprendizagem</i>
	<i>Conteudo</i>		<i>FcEA</i>
	<i>DataInicio</i>		<i>ObjetoAprendizagem</i>
	<i>DataFim</i>		<i>TipoOA</i>
	<i>IdEvento</i>		<i>FcOA</i>
	<i>Contexto</i>		<i>Dispositivo</i>
			<i>FcDispositivo</i>
Segurança	<i>IdAprendiz</i>		<i>Aplicativo</i>
	<i>Login</i>		<i>FcAplicativo</i>
	<i>Senha</i>		<i>Contexto</i>
			<i>Data</i>

Figura 4.2 – Perfil do Aprendiz no PeLeP

telefone de contato.

Objetivos contém os objetivos de aprendizagem. A categoria é constituída com os seguintes elementos: 1) *IdAprendiz*: identificador único; 2) *Objetivo*: objetivo educacional; 3) *Descrição*: dado usado para descrever a que se refere o objetivo; 4) *Contexto*: contexto em que o aprendiz possui o objetivo; 5) *Data*: data da última atualização do objetivo. Com a categoria Objetivos são inferidas informações sobre o que o aprendiz quer/gostaria de aprender.

A categoria Preferências se refere às preferências do aprendiz. Essa categoria é populada com os elementos: 1) *IdAprendiz*: identificador único; 2) *EstiloAprendizagem*: estilo de aprendizagem preferido; 3) *FcEA*: fator de certeza da preferência do aprendiz pelo estilo de aprendizagem; 4) *ObjetoAprendizagem*: objeto de aprendizagem (OA) preferido relacionado com cada contexto; 5) *TipoOA*: tipo do OA, que pode ser OArp (OA recomendado pelo professor) ou OAsup (OA suplementar, presente no cadastro de objetos do Sistema ubíquo, mas que não foi recomendado pelo professor); 6) *FcOA*: fator de certeza da preferência pelo OA; 7) *Dispositivo*: dispositivo preferido; 8) *FcDispositivo*: fator de certeza da preferência pelo dispositivo; 9) *Aplicativo*: aplicativo preferido; 10) *FcAplicativo*: fator de certeza da preferência pelo aplicativo; 11) *Contexto*: contexto em que foi

determinada a preferência; 12) *Data*: data da última atualização da preferência.

Segurança contém as credenciais de segurança, que permitem o acesso ao sistema. Esta categoria é populada com os elementos: 1) *IdAprendiz*: identificador único; 2) *Login*: *string* que representa o *login* de acesso. 3) *Senha*: senha de acesso.

Na categoria *Competências*, são listadas as experiências e conhecimentos já adquiridos pelo aprendiz. A mesma é constituída com os elementos: 1) *IdAprendiz*: identificador único; 2) *CompetenciaGeral*: descrição geral do conteúdo da competência (Exemplo: Matemática); 3) *CompetenciaEspecifica*: descreve especificamente à que se refere a competência (Exemplo: Teoria dos conjuntos); 4) *Contexto*: contexto em que o aprendiz adquiriu a competência; 5) *Data*: data da última atualização do competência. Com esta categoria inferem-se informações sobre o que o aprendiz quer/gostaria de ensinar.

A categoria *Relacionamentos* contém os relacionamentos do aprendiz com os contextos visitados por ele. Os relacionamentos considerados são Professor e Aluno. Como existem vários contextos num ambiente ubíquo, um usuário pode ser Professor em um contexto e Aluno em outro. Essa categoria é constituída de quatro elementos: 1) *IdAprendiz*: identificador único; 2) *ContextoRelacao*: contexto que o aprendiz possui algum relacionamento; 3) *Relacao*: relação com o contexto; 4) *Data*: data da última atualização do relacionamento.

A categoria *Agenda* é baseada nas categorias *Activity* e *Qcl* do padrão LIP [46]. Ela foi criada para relacionar informações específicas sobre as atividades diárias do aprendiz, levando em consideração seu contexto. Essa categoria possui os elementos: 1) *IdProfessor*: identificador único do professor que cadastrou a atividade; 2) *TipoAtividade*: tipo de atividade que deve ser cumprida, que pode ser reunião, palestra, aula, congresso, etc; 3) *Conteúdo*: conteúdo da atividade; 4) *DataInicio*: data e hora de início da atividade no formato *datetime* (aaaa-mm-dd hh:mm:ss); 5) *DataFim*: data e hora do fim da atividade no mesmo formato; 6) *IdEvento*: identificador do evento/atividade; 7) *Contexto*: contexto relacionado com a atividade.

A Figura 4.3 mostra o funcionamento da *Agenda*. O Sistema ubíquo transmite as alterações feitas na sua agenda para o Gerenciador de Perfil. Este componente do PeLeP se encarrega de atualizar os elementos da categoria *Agenda*, armazenados no Sistema de Perfis. Estas informações estão relacionadas com os contextos visitados

pelos aprendizes. Sempre que o professor cadastrar um nova atividade em um contexto, esta categoria é atualizada.

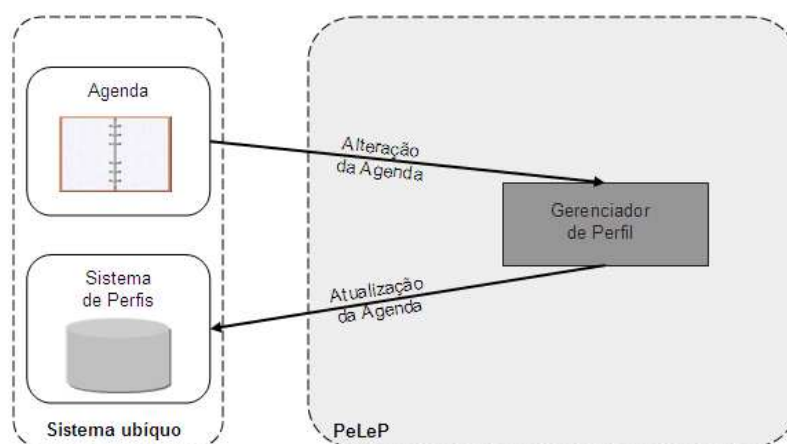


Figura 4.3 – Funcionamento da Agenda

Como exemplo de modificação de elementos da categoria *Agenda* através de inferência, pode-se supor que o aprendiz deve comparecer à uma reunião. Supõe-se ainda que a mesma ocorre dentro do contexto *A* em determinado horário. Se por algum motivo a reunião for adiada ou seu horário modificado, aquela informação será atualizada pelo professor na agenda do Sistema ubíquo, para o contexto *A*. Deste modo, independente de o aprendiz estar ou não no contexto *A*, a *Agenda* do seu perfil estará atualizada. Assim, quando ele consulta a agenda verifica sempre a informação mais atual relativa a mesma dentro daquele contexto.

#### 4.4 *Tracking* do aprendiz

*Tracking* é o histórico do aprendiz dentro de contextos. Ele não faz parte do perfil, pois é um recurso disponibilizado pelo Sistema de Localização. No PeLeP considera-se que o *tracking* deve conter o seguinte conjunto de informações:

1. Localizações;
2. Recursos acessados;
3. Dispositivos utilizados;
4. Aplicativos executados.

Estas informações também são consideradas no GlobalEdu. Porém, os perfis não são aperfeiçoados no seu modelo de aprendiz. Um exemplo do *tracking* utilizado no PeLeP é ilustrado no Apêndice A.

#### 4.4.1 Localizações

Este elemento do *tracking* contém os nomes dos contextos que foram visitados pelo aprendiz. Um contexto pode conter vários sub-contextos, como mostra o exemplo da Figura 4.4(a). Este exemplo pode ser representado através de uma árvore (Figura 4.4(b)). O PeLeP considera que o elemento Localizações do *tracking* possui apenas um nível de interesse devido ao teste do protótipo, que será abordado no próximo capítulo. O modelo PeLeP não é limitado quanto ao número de níveis para a árvore. Um exemplo, adaptado ao MobiLab[52], de estrutura dos contextos utilizados pelo elemento Localizações é exibido na Figura 4.5. Para obter uma alta escalabilidade nesse tipo de abordagem as informações poderiam ter sido armazenadas em RDF e Jena[40]. Estes modelos não foram utilizados porque a estrutura considerada no MobiLab ainda é simples.

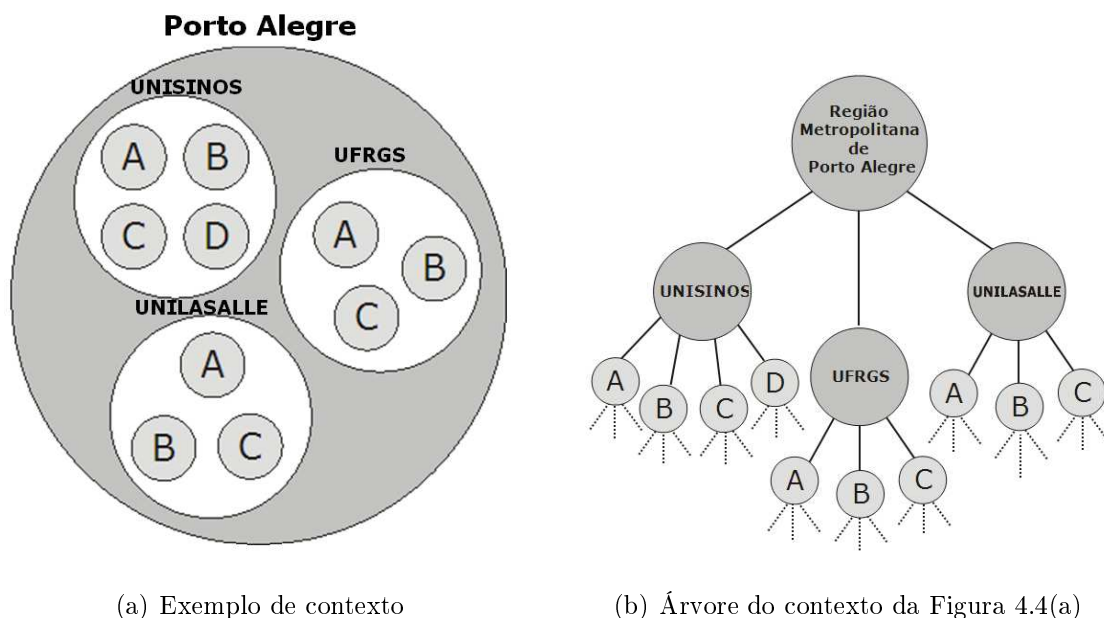


Figura 4.4 – Exemplo de contexto e árvore de contextos

A análise do elemento Localizações é utilizada na alimentação das categorias Agenda, Relacionamentos, Objetivos, Preferências e Competências, descritas na seção 4.3.

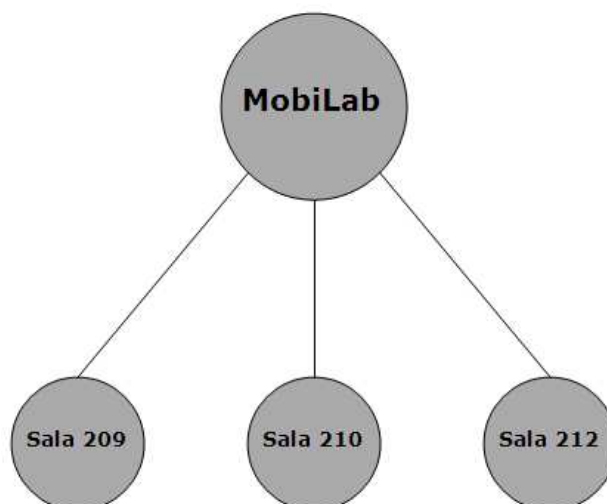


Figura 4.5 – Exemplo da estrutura dos contextos adaptado ao MobiLab

#### 4.4.2 Recursos acessados

Compreende os recursos que foram acessados pelo aprendiz dentro dos contextos visitados por ele. Os recursos considerados são os Objetos de Aprendizagem (OAs) [55] que foram acessados pelo aprendiz no ambiente ubíquo. O Sistema ubíquo deve possuir esses OAs previamente cadastrados. O professor responsável por um contexto seleciona OAs e os disponibiliza para os aprendizes. O seu trabalho consiste em selecionar OAs que abordem o assunto que os aprendizes freqüentadores do contexto precisam aprender. Esses OAs são classificados como OArp (objetos de aprendizagem recomendados pelo professor), enquanto que os demais acessados pelos aprendizes são classificados como OAsup (objetos de aprendizagem suplementares).

Neste trabalho foram considerados quatro Estilos de Aprendizagem (EAs), seguindo a abordagem de Felder [26]: percepção, entrada, processamento e compreensão. O modelo PeLeP não se restringe à utilização desses estilos de aprendizagem. Para cada assunto, o professor deve selecionar/recomendar quatro OAs, um para cada EA considerado, e disponibilizá-los aos aprendizes. Pelo acesso dos aprendizes a esses OArps no período de tempo especificado para o *tracking*, são determinados seus EAs através de inferências. O aprendiz pode acessar também um ou mais OAsups, que não foram recomendados pelo professor. Mas o acesso ao objeto suplementar não é utilizado para determinar o estilo do aprendiz, porque

eventualmente ele pode visualizar objetos com apenas um EA. Se o aprendiz não acessar objetos nos quatro estilos considerados, o estilo inferido se torna duvidoso. Os Recursos acessados são utilizados para alimentar as categorias Preferências, Competências e Objetivos.

#### 4.4.3 Dispositivos Utilizados e Aplicativos Executados

Compreendem-se em Dispositivos Utilizados os dispositivos que foram utilizados pelo aprendiz dentro dos contextos visitados por ele. Os dispositivos considerados são: PDA, *Tablet*, *Notebook*, PC e celular. Este elemento do *tracking* alimenta a categoria Preferências. Com isto infere-se o padrão de dispositivo utilizado pelo aprendiz em cada contexto.

Aplicativos Executados são os software (*Word*, ICQ, *Messenger*, *GTalk*, *Mozilla Firefox*, programas de *e-mails*, jogos, etc) que foram executados pelo aprendiz dentro dos contextos. Este elemento do *tracking* também é utilizado para alimentar a categoria Preferências.

### 4.5 Normalização dos dados recebidos no *tracking*

Como forma de regularizar a importância do valor relativo de cada variável numérica recebida no *tracking* foi necessário operar uma transformação de modo a que se possa garantir que todas as variáveis são equivalentes em termos de magnitude. Este processo é chamado de normalização. As formas mais comuns de normalização, segundo Lopes [48], são a normalização da amplitude, remoção da média (*mean centring*) e a auto-normalização (*auto-scaling*). As duas últimas não serão utilizadas neste trabalho pelo fato de *mean centring* ser um tipo de normalização adequada a métodos que necessitem do cálculo de matrizes de covariância e *auto-scaling* envolver o desvio padrão amostral.

A normalização da amplitude consiste em transformar todas as variáveis de modo que elas tenham uma amplitude idêntica e partilhem o mesmo valor máximo e mínimo. A forma mais comum consiste em aplicar um operador linear a todas as variáveis. Sendo  $m$  o valor mínimo escolhido e  $M$  o valor máximo, então o operador linear aplicado ao elemento  $x_i$  de um vetor é dado pela equação 4.2, onde  $m_x$  e  $M_x$  são os valores mínimo e máximo do vetor. O termo  $z_i$  corresponde à variável



normalizada.

$$z_i = \frac{x_i - m_x}{M_x - m_x} \quad (4.1)$$

$$z_i = (M - m) \frac{x_i - m_x}{M_x - m_x} + m \quad (4.2)$$

Um caso particular é normalizar de forma que todas as variáveis estejam entre os limites 0 e 1 ( $m = 0$  e  $M = 1$ ). Neste caso a equação 4.2 simplifica-se para a Equação 4.1.

Para a aplicação da inferência estatística nos valores das variáveis vindas do *tracking*, as mesmas quando normalizadas não podem assumir o valor zero, pois serão multiplicadas pela relevância percentual de cada fator considerado dentro do contexto específico. Essa multiplicação será apresentada posteriormente. Por isso, optou-se pela normalização da amplitude (Equação 4.2). Os valores escolhidos para o menor valor ( $m$ ) e o maior ( $M$ ), foram 1 e 10, respectivamente. Dificilmente  $M_x$  e  $m_x$  assumirão valores idênticos, pois cada vetor de fatores possui três elementos. Caso isso ocorra, para que não haja divisão por zero na equação 4.2 foi convencionado que  $z_i$  será igual a  $m$ , que vale sempre 1.

Nas linhas 23, 24 e 25 do *tracking* representado no Apêndice A, pode-se verificar que o primeiro objeto de aprendizagem foi visualizado por um tempo total de 145222 segundos e acessado 28 vezes em 3 dias. Utilizando a expressão da normalização da amplitude (Equação 4.2), os valores normalizados seriam, respectivamente: 10, 1.00155 e 1. Estes valores são utilizados para determinar o Fator de certeza (FC), que é calculado para cada elemento do *tracking* para que sejam feitas as inferências.

## 4.6 Fator de certeza

Vários fatores são importantes para a determinação do que vai ser alterado no perfil do aprendiz. Além disso, cada um desses fatores pode ter um grau de importância diferente para a determinação de uma preferência, por exemplo. Por isso, criou-se o Fator de Certeza (FC), que determina o grau de certeza das informações que vão para o perfil considerando o percentual de relevância de cada fator (tempo, número de acessos e número de dias) representado por uma variável numérica no *tracking*.

Como exemplo, analisa-se o *tracking* de um aprendiz, representado no Apêndice A. O mesmo retrata que apenas um contexto foi visitado, “Sala 209” na linha 07. Entre as linhas 116 e 165 se encontram as informações relativas aos aplicativos que foram executados. Os dados relacionados ao aplicativo “Word”, por exemplo, são limitados pelas linhas 118 e 121. Para definir o aplicativo preferido do aprendiz no contexto “Sala 209”, os fatores que determinam o FC dos aplicativos são três: tempo de utilização (tempo), número de acessos ao aplicativo (nAcessos) e número de dias que o mesmo foi utilizado (nDias). Esses fatores representam as variáveis numéricas utilizadas para definir qual aplicativo vai popular a categoria Preferências do perfil. Cada um deles possui um percentual de relevância associado que indica o quanto aquele fator é importante para determinar o aplicativo preferido. Se o FC calculado com os valores normalizados e os percentuais de relevância para o aplicativo “Word” for maior do que o FC do aplicativo que estava armazenado no perfil anteriormente, então “Word” passa a ser o preferido para aquele contexto, substituindo o anterior.

Antes de realizar inferências, definiu-se o percentual de relevância de cada um dos fatores considerados: 10% para o fator tempo, 30% para o número de acessos e 60% para o número de dias. O FC é calculado da seguinte forma:

$$FC = \frac{\sum_{i=0}^n f_i p_i}{\frac{100}{n}} \quad (4.3)$$

onde  $f_i$  representa o valor normalizado para o fator,  $p_i$  seu percentual de relevância e  $n$  o número de fatores considerados. O fator de certeza determina que item (recurso, aplicativo, dispositivo) será enviado para o perfil. Por exemplo, se o FC do item do *tracking* analisado for maior do que o FC do item que já está armazenado no perfil, então o item armazenado é substituído.

## 4.7 Banco de Resultados

Um fator de certeza (FC) que foi armazenado em uma categoria do perfil do aprendiz eventualmente pode assumir um valor bastante alto. Como exemplo, pode-se considerar que o FC de um aplicativo executado tenha sido  $X$  e definido a preferência por aquele aplicativo no perfil. Se por algum motivo o aprendiz parou

de utilizar esse aplicativo, substituindo-o por outro, ou passou a utilizar mais de um aplicativo para executar a mesma tarefa, o FC para os novos aplicativos pode não atingir o valor que está armazenado no perfil. Um exemplo desta situação ocorre se o fator “tempo” for o mais relevante e o aprendiz utilizar os novos aplicativos por uma menor quantidade de tempo. Por isso, um fator de certeza pode eventualmente assumir um valor que dificilmente seja atingido novamente.

Para solucionar este tipo de problema, criou-se um banco de dados temporário, representado na Figura 4.1 como Banco de Resultados. Neste, são armazenados os maiores FCs que foram calculados nas últimas análises de *tracking*, juntamente com suas datas e outras informações consideradas relativas àqueles fatores. Existem quatro tabelas no Banco de Resultados:

1. **Aplicativos temporários:** armazena os aplicativos executados que retornaram os maiores FCs;
2. **Dispositivos temporários:** armazena os dispositivos utilizados que retornaram os maiores FCs;
3. **Objetos temporários:** armazena os objetos de aprendizagem (OAs) que foram mais acessados pelo aprendiz, o que também é determinado através dos FCs;
4. **Estilos temporários:** armazena os estilos de aprendizagem (EAs) dos OAs de maiores FCs.

Voltando ao exemplo citado anteriormente, supõe-se que o último FC calculado para determinar o aplicativo preferido não supera o valor presente no perfil. Nessa situação, uma regra verifica se o aplicativo armazenado no Banco de Resultados teve o maior FC nas últimas três vezes que o PeLeP foi executado. Se a regra retornar *verdadeiro*, então o aplicativo preferido que já estava armazenado no perfil é substituído pelo do Banco de Resultados, mesmo possuindo menor FC.

## 4.8 Regras

O processo de atualização do modelo de aprendiz é baseado em regras e fatores de certeza [32, 54]. Regras permitem obter conclusões a partir de evidências

anteriores. As regras adotadas no PeLeP possuem “condição” e “ação”, e são descritas com o seguinte formato: SE *condição* ENTÃO *ação*.

#### 4.8.1 Categoria Preferências

A substituição e/ou acréscimo de dados na categoria Preferências, que faz parte do modelo de aprendiz, é baseada nas seguintes regras:

1. SE  $item_{br}$  no perfil ENTÃO acrescentar *item* de maior  $FC$
2. SE  $FC_{br} > FC_{perfil}$  ENTÃO substituir *item*
3. SE  $FC_{br} == FC_{perfil}$  ENTÃO acrescentar *item*
4. SE  $FC_{br} < FC_{perfil}$  ENTÃO manter *item*

onde  $FC_{br}$  é o fator de certeza (FC) do *item* armazenado no Banco de Resultados (*br*) e  $FC_{perfil}$  é o FC já armazenado no banco de dados que contém o perfil (Sistema de Perfis na Figura 4.1). Nas regras acima, *item* corresponde a preferência que é analisada, que pode ser estilo de aprendizagem (EA), objeto de aprendizagem (OA), dispositivo ou aplicativo.

No Sistema ubíquo o professor recomenda OAs para assuntos específicos. Cada assunto precisa da recomendação de quatro OAs, um para cada estilo de aprendizagem: percepção, entrada, processamento e compreensão [26]. Todos os OAs recomendados pelo professor (OArps) ficam disponíveis para os aprendizes acessarem. O *tracking* é solicitado a cada período de tempo determinado para que o aprendiz execute um grande número de tarefas no ambiente ubíquo.

À medida que o aprendiz acessa os objetos recomendados para um mesmo assunto, mas de quatro EAs diferentes, certamente um OArp terá um maior número de acessos ou de dias. Ou ainda, poderá ter sido visualizado por maior quantidade de tempo. Isso resultará um FC maior para aquele OArp. Assim, como cada objeto possui apenas um estilo, o EA do aprendiz é o do objeto recomendado que retornou o maior fator de certeza após a análise do *tracking*.

#### 4.8.2 Categoria Competências

O elemento *CompetenciaGeral* da categoria Competências é populado com o *keyword* do OA de maior FC, enquanto que o elemento *CompetenciaEspecificca* é

populado com o *prerequisite* do mesmo OA. A substituição e/ou acréscimo de dados na categoria Competências do perfil do aprendiz é baseada nas seguintes regras:

1. SE  $item_{br}$  no perfil ENTÃO acrescentar  $item_{br}$  de maior  $FC$
2. SE  $FC_{br} == FC_{perfil}$  E  $item_{br} \neq item_{perfil}$  ENTÃO acrescentar  $item$
3. SE  $FC_{br} == FC_{perfil}$  E  $item_{br} == item_{perfil}$  ENTÃO atualizar  $item$

onde  $FC_{br}$  é o FC do *item* armazenado no Banco de Resultados (*br*) e  $FC_{perfil}$  é o FC já armazenado no perfil. Nas regras acima, subentende-se por *item*: competência. A execução dessas regras determina se os dados são acrescentados ou atualizados no perfil. Não existe regra para excluir as competências que foram adquiridas pelo aprendiz.

#### 4.8.3 Categoria Objetivos

O elemento *Objetivo* da categoria Objetivos é populado com o *Educational Goals* do OA com maior FC, enquanto que o elemento *Descricao* é populado com o *subject* do mesmo OA. A substituição e/ou acréscimo de dados na categoria Objetivos do perfil do aprendiz é baseada nas seguintes regras:

1. SE  $item_{br}$  no perfil ENTÃO acrescentar  $item_{br}$  de maior  $FC$
2. SE  $FC_{br} == FC_{perfil}$  E  $item_{br} \neq item_{perfil}$  ENTÃO acrescentar  $item$
3. SE  $item_{br} == item_{perfil}$  ENTÃO atualizar  $item$

onde o significado das variáveis é o mesmo das categorias anteriores. Nas regras exibidas acima, subentende-se por *item*: objetivo. Essas regras determinam se os dados são acrescentados ou atualizados na categoria Objetivos do perfil do aprendiz.

#### 4.8.4 Categoria Agenda

O gerenciamento desta categoria do perfil do aprendiz não ocorre no mesmo intervalo de tempo do *tracking*, como as demais. Ele será ativado automaticamente sempre que uma nova atividade for inserida, ou uma antiga modificada, pelo professor no Sistema ubíquo. Quando o professor cadastra uma atividade na agenda

para um determinado contexto, a informação é acrescida no perfil dos aprendizes. A categoria *Agenda* pode ser visualizada por eles. Assim, se um aprendiz acessar a sua agenda para visualizar as atividades que precisa realizar, as informações já estão atualizadas.

O Sistema ubíquo possui o componente *Agenda* (Figura 4.3), enquanto que o PeLeP possui a categoria *Agenda*, que faz parte do perfil do aprendiz. Todos os elementos dessa categoria são populados com dados presentes no componente *Agenda* do Sistema ubíquo. O módulo Gerenciador de Perfil do PeLeP não armazena esses dados no Banco de Resultados. As informações são armazenadas diretamente na categoria.

Somente professores acrescentam atividades na agenda do Sistema ubíquo. Eles podem acrescentar informações na agenda a qualquer momento. Como as demais regras são executadas no mesmo intervalo de tempo determinado para o recebimento do *tracking*, a atualização das informações da agenda é feita diretamente no Sistema ubíquo. Não depende do tempo do *tracking*. Quando o Sistema ubíquo detecta uma alteração, aciona o gerenciamento da categoria *Agenda* do PeLeP. Portanto, o conteúdo dos elementos *IdProfessor*, *TipoAtividade*, *Data*, *Conteudo*, *HoraInicio*, *HoraTermino* e *Contexto* dependem apenas das regras:

1. SE *atividade* foi modificada no Sistema ubíquo ENTÃO substituir  $item_{perfil\_agenda}$
2. SE *atividade* no perfil ENTÃO acrescentar *atividade*

Percebe-se que estas regras não utilizam nenhum FC. Por isso as informações dessa categoria não são armazenadas no Banco de Resultados. Nas regras acima *atividade* representa a atividade que o professor cadastra no Sistema ubíquo.

#### 4.8.5 Categoria Relacionamentos

As regras que definem os dados que são armazenados na categoria *Relacionamentos* são baseadas na categoria *Agenda*. Na execução das regras, as informações acrescentadas pelos professores são armazenadas na categoria *Agenda* do PeLeP. Pela consulta ao elemento *IdProfessor* da *Agenda* determina-se se um usuário é professor ou aprendiz. Um usuário que é professor num contexto pode ser

aprendiz em outro. Neste sentido, as regras que definem o que é armazenado nessa categoria do perfil são:

1. SE  $IdAprendiz_{Relacionamentos} == IdProfessor_{Agenda}$   
ENTÃO  $Relacao_{Relacionamentos} = professor$
2. SE  $IdAprendiz_{Relacionamentos} \neq IdProfessor_{Agenda}$   
ENTÃO  $Relacao_{Relacionamentos} = aluno$

onde  $IdAprendiz_{Relacionamentos}$  representa o elemento  $IdAprendiz$  da categoria  $Relacionamentos$ , e  $IdProfessor_{Agenda}$  o  $IdProfessor$  da  $Agenda$ . As informações da categoria  $Relacionamentos$  determinam se os usuários do Sistema ubíquo são professores ou aprendizes. As mesmas estão relacionadas com os contextos visitados. A interação entre pessoas não foi abordada devido a existência de sistemas que já a consideram.

#### 4.9 Regra final

A execução das regras apresentadas na seção anterior determina se os dados são acrescentados, substituídos ou mantidos no Sistema de Perfis (Figura 4.1). Por isso, se nenhuma das regras citadas for executada, uma regra final analisa os últimos resultados armazenados no Banco de Resultados. Isso é feito especificamente para cada *item* avaliado nos perfis. Nesse banco de dados são armazenados apenas os itens que retornaram maior FC na análise feita no *tracking* dos aprendizes. Assim, essa regra é aplicada em todas as categorias citadas na seção anterior, exceto a  $Agenda$ , que não contém informações no Banco de Resultados. A regra final implica em:

SE existirem três linhas do Banco de Resultados populadas com aquele *item* nas últimas três análises de *tracking*  
E o *item* for diferente daquele já armazenado no perfil  
ENTÃO substituir *item*.

## 4.10 Considerações finais

Neste capítulo, apresentou-se o PeLeP, seus princípios e sua arquitetura. Foram descritos os processos de obtenção das informações que são manipuladas para popular o perfil do aprendiz. As regras de inferência para cada categoria do perfil foram apresentadas. Foi descrito o processo de análise do *tracking* e apresentada a adaptação do PeLeP à um Sistema ubíquo.

Através do acesso ao perfil do aprendiz gerado e manipulado pelo PeLeP, descobre-se aquilo que o aprendiz quer aprender (**Objetivos**) e o que ele pode ensinar (**Competências**). Além disso, pode-se verificar as suas **Preferências** quanto à estilos e objetos de aprendizagem, dispositivos e aplicativos. Ainda, se seu relacionamento com os contextos é de Professor ou Aluno. Desta forma, com posse dos perfis, o Sistema ubíquo pode indicar aos aprendizes novos objetos de aprendizagem (OAs), por exemplo. Neste sentido, o Sistema ubíquo procura OAs que efetivam a maior quantidade possível de necessidades do aprendiz. A indicação de OAs feita deste modo se torna “clara” para o aprendiz. Assim, o suporte computacional ao ensino e à aprendizagem em ambientes ubíquos pode ser aperfeiçoado pela aplicação do PeLeP à sistemas que os explorem.



## Capítulo 5

# Aspectos de implementação e resultados obtidos

Neste capítulo serão abordadas questões relativas a implementação do modelo, bem como um estudo de caso mostrando possibilidades de utilização dos recursos fornecidos por este trabalho. Serão mostrados ainda, resultados que foram obtidos durante a implementação do modelo aqui apresentado.

### 5.1 Protótipo

Nesta seção abordam-se aspectos de implementação do sistema apresentado no capítulo 4. Para a implementação do protótipo foi utilizada a linguagem de programação PHP 5.1 com banco de dados MySQL 5.0. Será apresentada agora a implementação dos módulos exibidos na Figura 4.1.

O protótipo foi testado no sistema LOCAL [11, 12]. Portanto, o Sistema ubíquo representado na Figura 4.1 é o LOCAL e o Sistema de Localização é seu módulo SeLIC (Servidor de Localização e Informação de Contexto). Este, fornece o *tracking* do aprendiz no ambiente ubíquo para o PeLeP. O *tracking* foi produzido através de arquivos XML com as possíveis saídas. Esta operação foi executada para todos os aprendizes que estavam cadastrados no LOCAL. Implementou-se um “gerador de *tracking*”, que gera de forma aleatória um número de *trackings* igual ao número de usuários cadastrados e acrescenta recursos acessados, dispositivos utilizados e aplicativos executados em todos os contextos existentes no ambiente

ubíquo do LOCAL. Um exemplo de *tracking* gerado para um aprendiz encontra-se no Apêndice A.

```

01 function executa_pelep($arquivo_xml) {
02     $locationsystem = simplexml_load_file($arquivo_xml);
04     if(empty($locationsystem))
05         die("<BR><B>01- Não foi possível carregar o arquivo XML!</B><BR>");
06     $contexto = new Contexto();
07     $id_aprendiz = $locationsystem->userid;//Determina o id do aprendiz.
08     $array_contextos = array();
09     foreach ($locationsystem->localizacoes->sala as $sala)
10     {
11         $contexto->setNome($sala->id);
12         array_push($array_contextos, $contexto->getNome());
13         $MaiorObjeto = new Objeto();
14         $MaiorObjeto->setFCObjeto(0);
15         $i = 0;
16         foreach ($sala->objetos->item as $objetos)
17         {
18             $temp = $contexto->AdicionarObjeto($objetos->id, $objetos->nome,
19                 $objetos->palavrasChave, $objetos->tempo, $objetos->nAcessos, $objetos->nDias,
20                 $objetos->tipo, $objetos->estilo, array(10,30,60), $contexto->getNome(),
21                 "objetos_temporarios", $id_aprendiz, $objetos->prerequisitos,
22                 $objetos->objetivosEducacionais, $objetos->resumoAssunto);
23             if ($temp->getFCObjeto() > $MaiorObjeto->getFCObjeto())
24                 $MaiorObjeto = $temp;
25             $i++;
26         }
27         $MaiorObjeto->SalvarObjetosBanco($contexto->getNome());
28         $contexto->ClearObjetos();
29         $MaiorEstilo = new Estilo();
30         $MaiorEstilo->setFCEstilo(0);
31         $ii = 0;
32         foreach ($sala->objetos->item as $objetos)
33         {
34             $temp = $contexto->AdicionarEstilo($objetos->id, $objetos->nome,
35                 $objetos->palavrasChave, $objetos->tempo, $objetos->nAcessos, $objetos->nDias,
36                 $objetos->tipo, $objetos->estilo, array(10,30,60), $contexto->getNome(),
37                 "estilos_temporarios", $id_aprendiz);
38             if ($contexto->getEstilo($ii)->getTipoObjeto() == "OArp" &&
39                 $temp->getFCEstilo() > $MaiorEstilo->getFCEstilo())
40                 $MaiorEstilo = $temp;
41             $ii++;
42         }
43         $MaiorEstilo->SalvarEstilosBanco($contexto->getNome());
44         $contexto->ClearEstilos();
45     } //fecha foreach
46     for ($i = 0; $i < count($array_contextos); $i++) {
47         require_once("ConectaBanco.php");
48         $classe = new ConectaBanco();
49         $classe->perfilCompetencias($id_aprendiz, $array_contextos[$i]);
50         $classe->perfilObjetivos($id_aprendiz, $array_contextos[$i]);
51         $classe->perfilRelacionamentos($id_aprendiz, $array_contextos[$i]);
52         $classe->perfilPreferencias($id_aprendiz, $array_contextos[$i]);
53     }
54 } //fecha executa_pelep();

```

Figura 5.1 – Rotina parcial do Gerenciador de Perfil

O módulo Filtro de *tracking* faz a leitura do arquivo XML que contém o *tracking*, organizando as informações para as rotinas executadas posteriormente. Objetos de aprendizagem (OAs) e estilos de aprendizagem (EAs) que possuem os maiores fatores de certeza (FCs) são inseridos no Banco de Resultados. A rotina responsável por essa tarefa é exibida na Figura 5.1. Entre as linhas 13 e 28 apresenta-se o código responsável pela inserção dos OAs. O mesmo ocorre entre as linhas 29 e 44 para os EAs. Um EA é adicionado na tabela de estilos temporários se o objeto relacionado a ele for recomendado pelo professor. Isso ocorre quando o tipo do objeto for OArp (objeto de aprendizagem recomendado pelo professor).

As linhas 20 e 36 da Figura 5.1 mostram a representação dos percentuais de relevância dos fatores considerados para os OAs. Os fatores são: tempo, número de acessos (nAcessos) e número de dias (nDias) que o OA foi acessado. Dessa forma, **array(10,30,60)** representa 10% de relevância para o fator tempo, 30% para o fator número de acessos e 60% para o fator número de dias. Ainda na Figura 5.1, o código entre as linhas 49 e 52 representa a execução das regras responsáveis pelo gerenciamento das categorias do perfil do aprendiz. Da linha 18 até a 22 e entre as linhas 34 e 37 mostra-se o trecho responsável pelo retorno dos valores de cada fator já normalizados. Isto ocorre porque além de adicionar objetos e estilos que possuem o maior FC, as funções *AdicionarObjeto* e *AdicionarEstilo* executam de forma implícita a função responsável pela normalização. Esta, está representada na Figura 5.2. A implementação responsável pela inserção dos aplicativos e dispositivos com os maiores FCs foi produzida de forma semelhante.

```

01 class Normaliza{
02     public function normalizaDados($array_a_normalizar) {
03         $m = 1;
04         $M = 10;
05         $mx = min($array_a_normalizar);
06         $Mx = max($array_a_normalizar);
07         for ($i = 0; $i < count($array_a_normalizar); $i++) {
08             if ( ($Mx - $mx) != 0) //Para que não haja divisão por zero.
09                 $array_normalizado[$i] = ($M - $m)*($array_a_normalizar[$i] - $mx)/($Mx - $mx)
10                 + $m;
11             else
12                 $array_normalizado[$i] = $m;
13         }
14         Return $array_normalizado;
15     }
16 }//fecha classe

```

Figura 5.2 – Normalização dos dados

O módulo Normalização de Dados foi implementado para aplicar os valores dos fatores considerados para recursos, aplicativos e dispositivos juntamente com os percentuais de relevância na Equação 4.2. A rotina de implementação desse módulo é mostrada na Figura 5.2. A linha 8 do código certifica que o denominador da equação 4.2 seja diferente de zero.

O módulo Cálculo do Fator de Certeza foi implementado para definir o FC dos OAs, EAs, aplicativos e dispositivos. O FC é definido através da aplicação dos fatores normalizados e seus percentuais de relevância na Equação 4.3. A rotina de implementação desse módulo está representada na Figura 5.3. Entre as linhas 9 e 12 ocorre a aplicação da Equação 4.3. As linhas 4 e 5 garantem que os percentuais de relevância totalizam cem por cento. A utilização deste módulo determina quais aplicativos, dispositivos, objetos e estilos de aprendizagem foram mais utilizados/visualizados/estudados durante o período que constitui um *tracking*.

```

01     class FatorCerteza{
02         public function fc($array_fatores_normalizados, $array_percentuais_relevancia,
03             $array_itens) { //Itens são dispositivos/aplicativos/recursos
04             if (array_sum($array_percentuais_relevancia) != 100)
05                 die("ERRO: A soma dos percentuais de relevância deve resultar 100%!");
06             $num_fatores = count($array_fatores_normalizados);
07             for ($num_item = 0; $num_item < count($array_itens); $num_item++) {
08                 $fc_parcial = array();
09                 for ($i = 0; $i < $num_fatores; $i++) {
10                     array_push($fc_parcial,
11                         $array_fatores_normalizados[$i] * $array_percentuais_relevancia[$i]/100);
12                 }
13                 $fc = array_sum($fc_parcial)/$num_fatores;
14                 $fc_parcial = NULL;
15             }
16             Return $fc;
17         }
18     }

```

Figura 5.3 – Cálculo do FC

O PeLeP permite que o intervalo de tempo entre as solicitações de *tracking* seja escolhido. Para esse protótipo, o tempo escolhido foi de uma semana. Sendo assim, o módulo Gerenciador de Perfil foi implementado para, semanalmente:

- Executar o Filtro de *Tracking*;
- Ativar Normalização de Dados;
- Executar Cálculo do Fator de Certeza;

- Armazenar os maiores FCs no Banco de Resultados;
- Executar as regras de inferência (seção 4.8) comparando as informações existentes no Sistema de Perfis com as do Banco de Resultados;
- Acrescentar/atualizar/substituir as informações do módulo Sistema de Perfis.

Devido a grande quantidade de tarefas que foram implementadas para o funcionamento desse módulo não são exibidos grandes trechos do código. Uma rotina parcial do Gerenciador de Perfil foi mostrada na Figura 5.1.

## 5.2 Integração LOCAL/PeLeP

Um protótipo do LOCAL [11, 12] encontra-se em funcionamento no MobiLab [52] da Unisinos. A Figura 5.4 mostra o PeLeP inserido na arquitetura do LOCAL, que foi exibida na Figura 3.1 e cujos módulos foram explicados na seção 3.9. O módulo SeLIC é o Servidor de Localização e Informação de Contexto, que está em fase de implementação e será o responsável pela produção do *tracking* para o PeLeP. A integração LOCAL/PeLeP foi realizada para que o módulo Sistema de Perfis seja gerenciado pelo PeLeP.

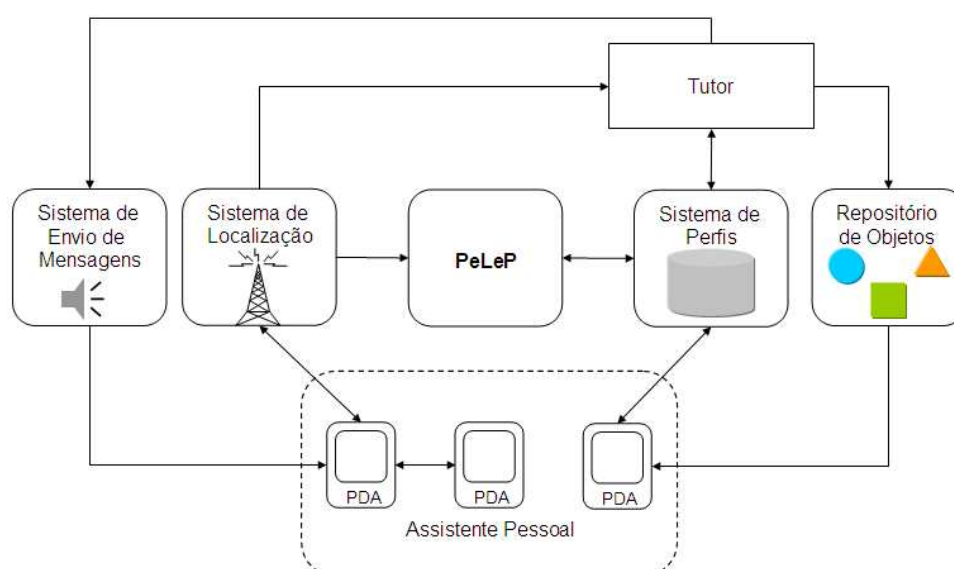


Figura 5.4 – PeLeP inserido na arquitetura do LOCAL

Anteriormente à integração do PeLeP, os perfis considerados pelo LOCAL continham poucas informações estáticas, que eram preenchidas diretamente pelos

aprendizes. A integração do PeLeP possibilitou o armazenamento de um número elevado de informações acerca da aprendizagem do aprendiz. Isso foi possível porque as tarefas do PeLeP se baseiam nos objetos de aprendizagem (OAs) que os aprendizes acessam do próprio LOCAL, em seu módulo Repositório de Objetos. Pelos OAs cadastrados pelo professor, são determinados os estilos de aprendizagem (EAs) dos aprendizes em cada contexto que eles realizam tarefas que envolvem o acesso aos objetos. Na integração LOCAL/PeLeP foi utilizado o modelo de OAs simples do LOCAL. O PeLeP não se reduz a este modelo de objetos de aprendizagem.

A tarefa do PeLeP nessa integração é solicitar o *tracking* ao SeLIC e baseando-se nele aperfeiçoar o perfil do aprendiz através de inferências. As informações sobre EAs, objetivos, competências, relacionamentos, etc, permitem ao Tutor (Figura 5.4) sugerir novos OAs que sirvam para realizar os objetivos educacionais do aprendiz. Essa sugestão, baseada no EA, aplicativo e dispositivo preferidos, pode possibilitar o recebimento de informações relevantes e precisas para a aprendizagem do aprendiz. Dessa forma, a busca de informações, por parte do aprendizes, para o enriquecimento de suas competências pode ficar reduzida.

### 5.3 Resultados obtidos

A validação do PeLeP foi realizada no ambiente de trabalho da equipe que desenvolve o LOCAL, no MobiLab [52]. Os *trackings* foram gerados aleatoriamente para todos os aprendizes e contextos já cadastrados no banco de dados do LOCAL<sup>1</sup>. No Apêndice A é exibido o *tracking* de um aprendiz, produzido para o teste do protótipo. Embora possa representar uma situação real, esse *tracking* é simples, pois mostra um aprendiz, que entrou apenas em um contexto (“Sala 209” na linha 7), utilizou apenas um dispositivo (“Tablet” na linha 110) e acessou poucos OAs.

A rotina implementada para a geração de *trackings* aleatórios produz também arquivos XML extensos, indicando que o aprendiz realizou diversas tarefas em vários contextos do LOCAL. Isso também pode representar uma situação real, de um ambiente de pesquisa, por exemplo. Esses arquivos XML não foram mostrados neste texto devido a suas grandes extensões, mas foram utilizados para a exibição dos resultados obtidos.

---

<sup>1</sup>SeLIC estava sendo desenvolvido em outra dissertação, no PIPCA/Unisinos.

Com posse dos repositórios XML gerados aleatoriamente, o módulo Gerenciador de Perfil foi acionado e todos os demais processos do PeLeP foram executados. Um grande número de aprendizes estão cadastrados no LOCAL. Foram gerados *trackings* para todos eles. O PeLeP mostrou resultados positivos nos testes com todos os aprendizes. Com o intuito de mostrar figuras objetivas, com um menor número de informações, foram selecionados apenas cinco aprendizes para gerar o *tracking* e mostrar os resultados obtidos. Pelo mesmo motivo, o número de OAs cadastrados foi limitado a doze.

ID_APRENDIZ	NOME	ENDERECO	EMAIL	TELEFONE
321	Darci Levis	Unisinos	daricilevis@fisicashow.com	05135911122
322	Rodrigo Dormento	Unisinos	dormento@gmail.com	05134543456
323	Cícero Rolim	Unisinos	ironworld@gmail.com	05135923489
324	Gustavo Lermen	Unisinos	lermen@gmail.com	05138909090
325	Jorge Barbosa	Unisinos	jbarbosa@unisinos.br	05135921435

(a) Categoria Identificação para cinco usuários do LOCAL

CONTEXTO	TIPO_ATIVIDADE	CONTEUDO	DATA_INICIO	DATA_FIM	ID_EVENTO	ID_PROFESSOR
Sala 209	Reunião	Definição do perfil	2006-10-10 14:00:00	2006-10-10 16:00:00	1	325
Sala 212	Palestra	Computação ubíqua	2006-11-10 16:00:00	2006-11-10 19:00:00	2	344

(b) Categoria Agenda com dois eventos

ID_APRENDIZ	CONTEXTO_RELAC	RELACAO	DATA
321	Sala 202	Aluno	2006-11-06 06:18:29
322	Sala 208	Aluno	2006-11-06 06:18:29
322	Sala 202	Aluno	2006-11-06 06:18:29
322	Sala 206	Aluno	2006-11-06 06:18:29
323	Sala 208	Aluno	2006-11-06 06:18:29
323	Sala 206	Aluno	2006-11-06 06:18:29
323	Sala 202	Aluno	2006-11-06 06:18:29
324	Sala 202	Aluno	2006-11-06 06:18:29
324	Sala 209	Aluno	2006-11-06 06:18:29
324	Sala 208	Aluno	2006-11-06 06:18:29
324	Sala 206	Aluno	2006-11-06 06:18:29
325	Sala 202	Aluno	2006-11-06 06:18:29
325	Sala 206	Aluno	2006-11-06 06:18:29
325	Sala 208	Aluno	2006-11-06 06:18:29

(c) Primeira execução para a categoria Relacionamentos

Figura 5.5 – Categorias Identificação, Agenda e Relacionamentos

Os elementos que são considerados em cada categoria foram mostrados na Figura 4.2. A tabela do banco de dados que representa a categoria Identificação para os cinco aprendizes selecionados é mostrada na Figura 5.5(a). A tabela exibida na Figura 5.5(b) representa a categoria Agenda. A categoria Relacionamentos depende

dos eventos cadastrados na **Agenda**. Nesta categoria foram cadastrados apenas dois eventos, para mostrar os resultados obtidos de maneira sucinta. Nota-se que o campo `ID_PROFESSOR`, exibido na Figura 5.5(b), contém os valores 325 e 344 relacionados com os contextos “Sala 209” e “Sala 212”, respectivamente. Esses valores identificam dois professores, pois somente eles acrescentam eventos na **Agenda**.

Os resultados obtidos na primeira execução do PeLeP para a categoria **Relacionamentos** são mostrados na Figura 5.5(c). Percebe-se que esta figura exibe as informações geradas pelo PeLeP na data 06/11/2006 para aquela categoria. As mesmas podem ser visualizadas em qualquer linha do campo `DATA`, representada no formato *datetime*. Analisando-se o campo `ID_APRENDIZ` é verificado que os valores vão de 321 até 325. Isto significa que os cinco usuários cadastrados (Figura 5.5(a)) executaram tarefas em contextos do ambiente ubíquo. O valor 325 representa um professor apenas no contexto “Sala 209”, como mostra a Figura 5.5(b). Conforme ressaltado pela Figura 5.5(c), esse professor não executou tarefas no contexto “Sala 209” na primeira semana que o PeLeP analisou os *trackings*. Por isso, nenhum usuário foi classificado como “Professor” nos contextos em que foram realizadas tarefas na semana considerada.

A Figura 5.6 mostra os resultados da primeira execução do PeLeP para a categoria **Competências**. Pode-se verificar que um aprendiz pode ter uma mesma competência em contextos diferentes. Verifica-se ainda que o aprendiz que possui no campo `ID_APRENDIZ` o número 321 realizou tarefas apenas no contexto “Sala 202”, enquanto que os demais o fizeram em mais de um contexto.

ID_APRENDIZ	COMPETENCIA_GERAL	COMPETENCIA_ESPECIFICA	CONTEXTO	DATA
321	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 202	06/11/2006
322	C++, classes, bibliotecas	Métodos e estruturas da linguagem C++	Sala 208	06/11/2006
322	Java, classes, applet	Métodos e comandos da programação	Sala 202	06/11/2006
322	Java, classes, applet	Métodos e comandos da programação	Sala 206	06/11/2006
323	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 208	06/11/2006
323	Java, classes, applet	Métodos e comandos da programação	Sala 206	06/11/2006
323	Relatividade, Teoria dos gêmeos	Cálculo diferencial e integral	Sala 202	06/11/2006
324	C++, classes, bibliotecas	Métodos e estruturas da linguagem C++	Sala 206	06/11/2006
324	Posicionamento global, territórios, satélites	GPS: Global Positioning System	Sala 209	06/11/2006
324	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 208	06/11/2006
325	Java, classes, applet	Métodos e comandos da programação	Sala 202	06/11/2006
325	Java, classes, applet	Métodos e comandos da programação	Sala 206	06/11/2006
325	Delphi, classes	Introdução a linguagem Delphi	Sala 208	06/11/2006

Figura 5.6 – Primeira execução para a categoria **Competências**



A Figura 5.7 mostra as categorias Preferências e Objetivos, também para a primeira execução. A categoria Preferências teve suas colunas separadas para possibilitar uma melhor visualização. A Figura 5.7(a) retrata que os estilos e objetos de aprendizagem, dispositivos e aplicativos preferidos pelos aprendizes, foram determinados pelo PeLeP. Além disso, essas preferências estão relacionadas com os contextos considerados no LOCAL. A Figura 5.7(b) mostra os objetivos educacionais inclusos na categoria Objetivos.

ID_APRENDIZ	CONTEXTO	ESTILO_APRENDIZAGEM	FC_EA	OBJETO_APRENDIZAGEM	TIPO_OA
321	Sala 202	Compreensão	0.633536	PHP para iniciantes	OArp
322	Sala 208	Percepção	0.635564	Introdução a C++	OArp
322	Sala 202	Entrada	0.634483	Java para iniciantes	OArp
322	Sala 206	Entrada	0.633531	Java para iniciantes	OArp
323	Sala 208	Compreensão	0.633543	PHP para iniciantes	OArp
323	Sala 206	Entrada	0.634601	Java para iniciantes	OArp
323	Sala 202	Compreensão	0.633831	Física relativística	OAs
324	Sala 202	Entrada	0.633545	Java para iniciantes	OArp
324	Sala 209	Percepção	0.63371	GPS - Introdução	OArp
324	Sala 208	Compreensão	0.633493	PHP para iniciantes	OArp
324	Sala 206	Percepção	0.633899	Introdução a C++	OArp
325	Sala 202	Entrada	0.634713	Java para iniciantes	OArp
325	Sala 206	Entrada	0.633333	Java para iniciantes	OArp
325	Sala 208	Processamento	0.633839	Delphi 6.0	OArp

FC_OA	DISPOSITIVO	FC_DISPOSITIVO	APLICATIVO	FC_APLICATIVO	DATA
0.633536	Notebook	1.23592	Word	0.93341	06/11/2006
0.635564	Notebook	1.23372	Word	0.933356	06/11/2006
0.634483	Desktop	1.2341	Internet Explorer	0.933775	06/11/2006
0.633531	Tablet	1.23368	Excel	0.933361	06/11/2006
0.633543	Tablet	1.23374	PowerPoint	0.935505	06/11/2006
0.634601	Notebook	1.23345	Mozilla	0.933527	06/11/2006
0.634233	Desktop	1.23437	Word	0.93469	06/11/2006
0.633545	Tablet	1.23378	Word	0.933376	06/11/2006
0.63371	Notebook	1.23924	Internet Explorer	0.933685	06/11/2006
0.633493	Tablet	1.23385	Internet Explorer	0.933474	06/11/2006
0.633899	Desktop	1.23347	Excel	0.933442	06/11/2006
0.634713	Tablet	1.23337	Word	0.933566	06/11/2006
0.633333	Notebook	1.2339	Word	0.933374	06/11/2006
0.633839	Desktop	1.23342	Word	0.933933	06/11/2006

(a) Categoria Preferências

ID_APRENDIZ	OBJETIVO	DESCRICAÇÃO	CONTEXTO	DATA
321	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 202	06/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 208	06/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 202	06/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 206	06/11/2006
323	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 208	06/11/2006
323	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 206	06/11/2006
323	Compreender o que é e do trata a Teoria da Relativ...	A Teoria da Relatividade é a denominação dada ao c...	Sala 202	06/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 202	06/11/2006
324	Conseguir acoplar a um sistema de localização, o m...	Abordagem detalhada do sistema de posicionamento g...	Sala 209	06/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 208	06/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 206	06/11/2006
325	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 202	06/11/2006
325	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 206	06/11/2006
325	Conseguir programar rotinas simples com a linguagem...	Conhecer a ferramenta Delphi.	Sala 208	06/11/2006

(b) Categoria Objetivos

Figura 5.7 – Primeira execução para as categorias Preferências e Objetivos

Após a primeira execução, os *trackings* aleatórios foram produzidos novamente. Para simular novas semanas, a data do servidor do LOCAL foi alterada para 13/11/2006 e para 20/11/2006, uma e duas semanas após, respectivamente. Todos os passos foram repetidos para esses dois novos testes. As categorias Identificação e Agenda, mostradas nas Figuras 5.5(a) e 5.5(b), respectivamente, permaneceram inalteradas. Os resultados da terceira execução do PeLeP para a categoria Relacionamentos são exibidos na Figura 5.8.

ID_APRENDIZ	CONTEXTO_RELAC	RELACAO	DATA
321	Sala 210	Aluno	2006-11-20 06:35:08
321	Sala 202	Aluno	2006-11-20 06:35:08
321	Sala 206	Aluno	2006-11-20 06:35:08
321	Sala 208	Aluno	2006-11-20 06:35:08
321	Sala 209	Aluno	2006-11-20 06:35:08
322	Sala 209	Aluno	2006-11-13 06:34:36
322	Sala 210	Aluno	2006-11-13 06:34:36
322	Sala 212	Aluno	2006-11-13 06:34:36
322	Sala 208	Aluno	2006-11-20 06:35:08
322	Sala 206	Aluno	2006-11-20 06:35:08
322	Sala 202	Aluno	2006-11-20 06:35:08
323	Sala 209	Aluno	2006-11-20 06:35:08
323	Sala 210	Aluno	2006-11-20 06:35:08
323	Sala 212	Aluno	2006-11-20 06:35:08
323	Sala 208	Aluno	2006-11-20 06:35:08
323	Sala 206	Aluno	2006-11-20 06:35:08
323	Sala 202	Aluno	2006-11-20 06:35:08
324	Sala 202	Aluno	2006-11-20 06:35:08
324	Sala 206	Aluno	2006-11-20 06:35:08
324	Sala 208	Aluno	2006-11-20 06:35:08
324	Sala 210	Aluno	2006-11-20 06:35:08
324	Sala 212	Aluno	2006-11-20 06:35:08
324	Sala 209	Aluno	2006-11-20 06:35:08
325	Sala 212	Aluno	2006-11-20 06:35:08
325	Sala 210	Aluno	2006-11-20 06:35:08
325	Sala 209	Professor	2006-11-20 06:35:08
325	Sala 202	Aluno	2006-11-20 06:35:08
325	Sala 206	Aluno	2006-11-20 06:35:08
325	Sala 208	Aluno	2006-11-20 06:35:08

Figura 5.8 – Terceira execução para a categoria Relacionamentos

Quando existem vários contextos em um ambiente de ensino e aprendizagem, o usuário que é professor em um contexto pode ser aluno em outro. A categoria Relacionamentos do PeLeP foi programada para relacionar essas informações com os contextos que os usuários realizam tarefas no ambiente ubíquo. Percebe-se pela análise da Figura 5.8, que o aprendiz de ID\_APRENDIZ 325 no contexto “Sala 209” teve seu relacionamento classificado como “Professor”. Esta informação foi definida

pela consulta à tabela que contém as atividades da categoria Agenda, mostrada na Figura 5.5(b). Na primeira execução do PeLeP, a relação desse mesmo aprendiz com os contextos não resultou “professor”, como ressaltado na Figura 5.5(c). Isso ocorreu porque até aquela data ele não realizou tarefas, que são mapeadas, no contexto que havia acrescentado atividades na agenda.

ID_APRENDIZ	COMPETENCIA_GERAL	COMPETENCIA_ESPECIFICA	CONTEXTO	DATA
321	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 210	20/11/2006
321	C#, classes	Introdução a C#: conhecimentos básicos	Sala 209	20/11/2006
321	Delphi, classes	Introdução a linguagem Delphi	Sala 206	20/11/2006
321	C#, classes	Introdução a C#: conhecimentos básicos	Sala 202	20/11/2006
321	C++, classes, bibliotecas	Métodos e estruturas da linguagem C++	Sala 206	20/11/2006
321	Java, classes, applet	Métodos e comandos da programação	Sala 208	20/11/2006
322	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 202	20/11/2006
322	Java, classes, applet	Métodos e comandos da programação	Sala 208	20/11/2006
322	C++, classes, bibliotecas	Métodos e estruturas da linguagem C++	Sala 209	13/11/2006
322	Delphi, classes	Introdução a linguagem Delphi	Sala 206	20/11/2006
322	Java, classes, applet	Métodos e comandos da programação	Sala 210	13/11/2006
322	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 212	13/11/2006
322	C#, classes	Introdução a C#: conhecimentos básicos	Sala 208	20/11/2006
322	C++, classes, bibliotecas	Métodos e estruturas da linguagem C++	Sala 202	20/11/2006
322	C++, classes, bibliotecas	Métodos e estruturas da linguagem C++	Sala 206	20/11/2006
323	C#, classes	Introdução a C#: conhecimentos básicos	Sala 208	20/11/2006
323	Java, classes, applet	Métodos e comandos da programação	Sala 212	20/11/2006
323	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 208	20/11/2006
323	C#, classes	Introdução a C#: conhecimentos básicos	Sala 206	20/11/2006
323	C#, classes	Introdução a C#: conhecimentos básicos	Sala 202	20/11/2006
323	Java, classes, applet	Métodos e comandos da programação	Sala 210	20/11/2006
323	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 206	20/11/2006
323	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 209	20/11/2006
324	IEEE 402.11a, captação de frequências	Padrões IEEE	Sala 210	20/11/2006
324	C++, classes, bibliotecas	Métodos e estruturas da linguagem C++	Sala 212	20/11/2006
324	Java, classes, applet	Métodos e comandos da programação	Sala 212	20/11/2006
324	Delphi, classes	Introdução a linguagem Delphi	Sala 209	20/11/2006
324	Java, classes, applet	Métodos e comandos da programação	Sala 208	20/11/2006
324	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 210	20/11/2006
324	Java, classes, applet	Métodos e comandos da programação	Sala 202	20/11/2006
324	Java, classes, applet	Métodos e comandos da programação	Sala 206	20/11/2006
324	IEEE 402.11a, captação de frequências	Padrões IEEE	Sala 208	20/11/2006
324	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 209	20/11/2006
325	Java, classes, applet	Métodos e comandos da programação	Sala 202	20/11/2006
325	C#, classes	Introdução a C#: conhecimentos básicos	Sala 206	20/11/2006
325	Delphi, classes	Introdução a linguagem Delphi	Sala 208	20/11/2006
325	Posicionamento global, territórios, satélites	GPS: Global Positioning System	Sala 212	20/11/2006
325	C++, classes, bibliotecas	Métodos e estruturas da linguagem C++	Sala 210	20/11/2006
325	C#, classes	Introdução a C#: conhecimentos básicos	Sala 209	20/11/2006
325	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 208	20/11/2006
325	PHP, classes, applet	Instalação do Apache e versões do PHP	Sala 206	20/11/2006
325	Triangulação, antenas próximas	Sistemas de localização existentes	Sala 202	20/11/2006

Figura 5.9 – Terceira execução para a categoria Competências

A Figura 5.9 exhibe os resultados obtidos na terceira execução para a categoria Competências. Pela análise da figura percebe-se que o número de contextos que cada aprendiz executou tarefas após três semanas foi considerável. Nota-se que o

aprendiz cujo campo ID\_APRENDIZ vale 324 foi o que mais adquiriu competências no período. Cabe ressaltar também, que uma competência do aprendiz pode ser determinada para mais de um contexto.

ID_APRENDIZ	OBJETIVO	DESCRICAÇÃO	CONTEXTO	DATA
321	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 202	20/11/2006
321	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 206	13/11/2006
321	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 208	20/11/2006
321	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 210	20/11/2006
321	Conseguir programar rotinas simples com a linguagem...	Conhecer a ferramenta Delphi.	Sala 206	20/11/2006
321	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 209	20/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 202	13/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Conhecer a ferramenta Delphi.	Sala 206	13/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 208	13/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 209	13/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 210	13/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 212	13/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 202	20/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 208	20/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 202	06/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 206	20/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 208	06/11/2006
322	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 206	06/11/2006
323	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 206	20/11/2006
323	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 202	20/11/2006
323	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 212	20/11/2006
323	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 208	20/11/2006
323	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 209	20/11/2006
323	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 210	20/11/2006
323	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 206	13/11/2006
323	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 202	13/11/2006
323	Compreender o que é e do trata a Teoria da Relativ...	A Teoria da Relatividade é a denominação dada ao c...	Sala 202	06/11/2006
323	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 206	06/11/2006
323	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 208	06/11/2006
324	Conseguir implementar um sistema de localização co...	Este OA aborda o padrão IEEE 402.11a.	Sala 208	20/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Conhecer a ferramenta Delphi.	Sala 209	20/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 210	20/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 212	20/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 202	20/11/2006
324	Conseguir acoplar a um sistema de localização, o m...	Abordagem detalhada do sistema de posicionamento g...	Sala 209	06/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 208	06/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 206	06/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 206	20/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 206	13/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 208	13/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 209	13/11/2006
324	Conseguir implementar um sistema de localização co...	Este OA aborda o padrão IEEE 402.11a.	Sala 210	13/11/2006
324	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 212	13/11/2006
325	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 206	13/11/2006
325	Conseguir programar rotinas simples com a linguagem...	Conhecer a ferramenta Delphi.	Sala 208	06/11/2006
325	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem C...	Sala 209	20/11/2006
325	Conseguir programar rotinas simples com a linguagem...	Descobrir todo o potencial da linguagem Java, com ...	Sala 206	06/11/2006
325	Conseguir implementar um sistema de localização ut...	Abordagem da triangulação de antenas próximas.	Sala 202	20/11/2006
325	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 206	20/11/2006
325	Conseguir programar rotinas simples com a linguagem...	Descobrir tudo que se pode fazer com a linguagem P...	Sala 208	20/11/2006

Figura 5.10 – Terceira execução para a categoria Objetivos

A Figura 5.10 mostra os resultados da terceira execução para a categoria Objetivos. Na seção 4.8 foram apresentadas as regras de inferência. Para a categoria Objetivos, o cumprimento das regras retorna acréscimo ou atualização de objetivos do aprendiz em seu perfil. A atualização ocorre quando o objetivo que teve o maior fator de certeza (FC) determinado for igual a um que já estava armazenado na categoria Objetivos do perfil. Como o PeLeP foi executado três vezes, nota-se na

Figura 5.10 que ainda existem informações que foram armazenadas na sua primeira execução, nas linhas que o campo DATA possui o valor 06/11/2006. Isso significa que, naqueles contextos, os objetivos que retornaram os maiores FCs durante a execução foram diferentes dos que já estavam armazenados nessa data.

A Figura 5.11 representa a categoria **Preferências** para a terceira execução. Novamente, a tabela mostrada na figura foi dividida para uma melhor visualização. Os campos FC\_EA, FC\_OA, FC\_DISPOSITIVO e FC\_APLICATIVO, representam, respectivamente, os fatores de certeza (FC) dos estilos de aprendizagem (EA), objetos de aprendizagem (OA), dispositivos e aplicativos. TIPO\_OA retrata o tipo do objeto, que pode ser Objeto de Aprendizagem recomendado pelo professor (OArp) ou Objeto de Aprendizagem suplementar (OAs). Este último, foi representado desta forma apenas na implementação, haja visto que no restante do texto utilizou-se o termo “OAsup” para caracterizá-lo.

É importante salientar que para cada assunto são recomendados pelo professor quatro objetos de aprendizagem, um para cada estilo considerado [26]. Por isso, para determinar os EAs dos aprendizes são analisados apenas os objetos do tipo OArp. Os FCs considerados resultam valores diferentes para estilos e objetos se o maior FC ocorre para um OA do tipo “OAs”. Neste caso, o FC que define o estilo será menor. Percebe-se na Figura 5.11, que quando o campo TIPO\_OA vale “OAs”, o valor armazenado em FC\_EA é diferente do FC\_OA, na mesma linha de resultados. Isto pode ser analisado na linha em destaque na figura. Esse fato comprova que os Estilos de Aprendizagem são determinados com base apenas nos objetos de aprendizagem classificados como “OArp”.

As regras que definem o conteúdo da categoria **Preferências** mantêm sempre a preferência atual (seção 4.8). Haverá mais do que uma preferência armazenada para um mesmo contexto, somente se ocorrerem FCs iguais aos já armazenados durante a execução. A probabilidade de isso ocorrer é bem pequena, já que são vários os elementos determinantes (tempo, número de dias e de acessos, e seus percentuais de relevância). Além disso, o número que expressa um FC possui vários dígitos. A Figura 5.11 mostra que para as três primeiras execuções do PeLeP não houve mais do que uma preferência em cada contexto. Basta analisar cada CONTEXTO relacionado com seu ID\_APRENDIZ nas duas primeiras colunas exibidas.

ID_APRENDIZ	CONTEXTO	ESTILO_APRENDIZAGEM	FC_EA	OBJETO_APRENDIZAGEM	TIPO_OA
321	Sala 209	Compreensão	0.634238	C# para iniciantes	OAs
321	Sala 202	Compreensão	0.63378	C# para iniciantes	OAs
321	Sala 206	Processamento	0.633966	Delphi 6.0	OArp
321	Sala 210	Compreensão	0.634569	PHP para iniciantes	OArp
321	Sala 208	Entrada	0.633603	Java para iniciantes	OArp
322	Sala 210	Entrada	0.63413	Java para iniciantes	OArp
322	Sala 209	Percepção	0.635402	Introdução a C++	OArp
322	Sala 208	Percepção	0.635564	Introdução a C++	OArp
322	Sala 212	Compreensão	0.633948	PHP para iniciantes	OArp
322	Sala 206	Percepção	0.635888	Introdução a C++	OArp
322	Sala 202	Entrada	0.634483	Java para iniciantes	OArp
323	Sala 209	Compreensão	0.633629	PHP para iniciantes	OArp
323	Sala 210	Entrada	0.633554	Java para iniciantes	OArp
323	Sala 212	Entrada	0.633829	Java para iniciantes	OArp
323	Sala 202	Compreensão	0.634012	Física relativística	OAs
323	Sala 206	Entrada	0.634601	Java para iniciantes	OArp
323	Sala 208	Entrada	0.633666	C# para iniciantes	OAs
324	Sala 206	Entrada	0.64212	Java para iniciantes	OArp
324	Sala 209	Compreensão	0.634721	PHP para iniciantes	OArp
324	Sala 202	Entrada	0.6337	Java para iniciantes	OArp
324	Sala 208	Entrada	0.634594	Java para iniciantes	OArp
324	Sala 210	Compreensão	0.634344	Padrão IEEE 402.11a	OArp
324	Sala 212	Entrada	0.634146	Java para iniciantes	OArp
325	Sala 206	Compreensão	0.637606	PHP para iniciantes	OArp
325	Sala 212	Percepção	0.634483	GPS - Introdução	OArp
325	Sala 210	Percepção	0.634021	Introdução a C++	OArp
325	Sala 209	Percepção	0.633743	C# para iniciantes	OAs
325	Sala 208	Processamento	0.633839	Delphi 6.0	OArp
325	Sala 202	Percepção	0.635086	Triangulação de antenas	OAs

FC_OA	DISPOSITIVO	FC_DISPOSITIVO	APLICATIVO	FC_APLICATIVO	DATA
0.638989	Desktop	1.23472	Word	0.933435	20/11/2006
0.633895	Notebook	1.23592	Excel	0.933541	20/11/2006
0.633966	Tablet	1.23367	Mozilla	0.933514	20/11/2006
0.634569	Tablet	1.23369	PowerPoint	0.933434	20/11/2006
0.633603	Desktop	1.2337	PowerPoint	0.933544	20/11/2006
0.63413	Tablet	1.23444	Internet Explorer	0.933812	13/11/2006
0.635402	Tablet	1.23348	GTalk	0.933621	13/11/2006
0.635564	Notebook	1.23372	Mozilla	0.934994	20/11/2006
0.633948	Tablet	1.23353	Mozilla	0.933389	13/11/2006
0.635888	Tablet	1.23368	Mozilla	0.933863	20/11/2006
0.634483	Desktop	1.2341	PowerPoint	0.933873	20/11/2006
0.633629	Tablet	1.23347	Word	0.933461	20/11/2006
0.633554	Tablet	1.23334	Excel	0.934129	20/11/2006
0.633829	Tablet	1.23367	Acrobat	0.933602	20/11/2006
0.634233	Desktop	1.23437	Acrobat	0.940146	20/11/2006
0.634601	Tablet	1.23363	Mozilla	0.933527	20/11/2006
0.634977	Tablet	1.23374	PowerPoint	0.935505	20/11/2006
0.64212	Desktop	1.23347	Excel	0.933442	20/11/2006
0.634721	Notebook	1.23924	Excel	0.933762	20/11/2006
0.6337	Notebook	1.23545	Word	0.940904	20/11/2006
0.634594	Notebook	1.23385	Word	0.933517	20/11/2006
0.634344	Tablet	1.23348	Word	0.934125	20/11/2006
0.634146	Tablet	1.23415	Acrobat	0.934825	20/11/2006
0.637606	Notebook	1.23393	Word	0.934306	20/11/2006
0.634483	Desktop	1.23361	Word	0.93448	20/11/2006
0.634021	Notebook	1.23356	Word	0.93365	20/11/2006
0.634414	Tablet	1.23679	Excel	0.933659	20/11/2006
0.633839	Desktop	1.23342	Word	0.933933	20/11/2006
0.635859	Tablet	1.23376	Mozilla	0.937207	20/11/2006

Figura 5.11 – Terceira execução para a categoria Preferências

A categoria Preferências contém o estilo de aprendizagem dos aprendizes de acordo com as atividades que eles realizam, e que são mapeadas, em contextos do ambiente ubíquo. Além disso, a mesma possui a informação dos dispositivos e

aplicativos que eles preferem utilizar naqueles contextos. Desta forma, analisando a linha destacada na Figura 5.11, o Sistema ubíquo conectado ao PeLeP poderia indicar um novo objeto de aprendizagem para o aprendiz de ID\_APRENDIZ 323, que: possua o estilo “Compreensão”; possa ser executado num dispositivo “Desktop”; seja um documento a ser visualizado no aplicativo “Acrobat”. Somado a isso, com a análise das informações da categoria **Objetivos**, o documento indicado abordaria um assunto relacionado aos objetivos do aprendiz no contexto em foco. No LOCAL, o componente responsável por essa indicação é o Tutor, mostrado na Figura 5.4.

## 5.4 Considerações finais

Este capítulo apresentou aspectos de implementação e os resultados que foram obtidos através da aplicação do protótipo no sistema LOCAL [11, 12]. Foram descritos os passos que o PeLeP segue a partir da requisição do *tracking* até o armazenamento das informações no perfil. Apresentou-se rotinas parciais desenvolvidas para cumprir as etapas descritas. Na integração LOCAL/PeLeP foi mostrado que os perfis do LOCAL foram aperfeiçoados e passaram a ser dinâmicos após serem gerenciados pelo PeLeP. Apresentou-se o conteúdo das tabelas geradas no banco de dados para as categorias que constituem os perfis.

A escalabilidade do PeLeP não foi testada porque o SeLIC, que é o responsável pelo fornecimento do *tracking*, estava sendo desenvolvido em outra dissertação, no MobiLab. Pelas três execuções do PeLeP a partir dos *trackings* gerados, mostrou-se que as informações produzidas pelo sistema podem ser aproveitadas para a indicação de novos objetos. Essa indicação torna-se importante nos ambientes ubíquos de ensino e aprendizagem. O tempo destinado pelos aprendizes à procura de materiais acerca dos assuntos que são estudados por eles pode ser reduzido. Isso atesta que o trabalho do PeLeP contribui para a educação.

## Capítulo 6

# Considerações Finais

Este capítulo apresenta as considerações finais da dissertação. Serão revisados os conceitos que motivaram o desenvolvimento deste trabalho, as contribuições que resultaram, conclusões e trabalhos futuros que foram vislumbrados no decorrer desta pesquisa.

### 6.1 Revisão de tema

A utilização da computação móvel está cada vez mais presente na educação, haja visto que o ensino e a aprendizagem são cada vez mais apoiados por ferramentas móveis. Com a computação consciente de contexto, percebe-se os elementos que envolvem os aprendizes e que são de seu interesse, possibilitando uma computação que se adapta ao contexto presente. A computação ubíqua trouxe várias potencialidades para os processos de ensino e aprendizagem. O desafio gerado, do ponto de vista educacional, foi a criação de um ambiente de aprendizagem móvel que proporcionasse recursos apropriados para o aprendiz. Neste sentido, este trabalho propôs o aperfeiçoamento do perfil de aprendiz, refletindo o seu comportamento no ambiente ubíquo.

Esta dissertação apresentou o PeLeP (*Pervasive Learning Profile*), um sistema para aperfeiçoamento de perfis de aprendizes, levando em consideração seus históricos no ambiente ubíquo. Mostrou-se o que é necessário considerar quando se planejam aplicações para ambientes de ensino e aprendizagem ubíquos. Foram apresentadas as tecnologias da educação ubíqua e suas relações com a educação a distância.



Seguindo esta linha, foram estudados sistemas que oferecem suporte à ambientes ubíquos e *Web* que se mostraram mais significativos na abordagem de informações acerca de perfis de aprendizes. Com este estudo foram identificados os principais requisitos atendidos por esses sistemas, o que permitiu adaptar idéias ao sistema LOCAL em desenvolvimento no MobiLab [52]. O LOCAL usa informações de localização e de contexto como auxílio ao processo de ensino e aprendizagem. Os perfis considerados pelo mesmo eram estáticos e possuíam poucas informações a respeito dos aprendizes. A integração LOCAL/PeLeP, desenvolvida nesta dissertação, permitiu o aperfeiçoamento do Sistema de Perfis do LOCAL.

Com posse das informações adquiridas, criou-se um modelo de perfil de aprendiz com dados considerados relevantes para o processo de ensino e aprendizagem ubíquo, separados por categorias. Convencionou-se que *tracking* seria o histórico do aprendiz no ambiente ubíquo e definiu-se as informações que deveriam constituí-lo para conseguir produzir dados através de inferências. Para otimizar o foco na aprendizagem ubíqua, foram acrescentados os Estilos de Aprendizagem na categoria **Preferências**. Estes, foram determinados através do mapeamento dos Objetos de Aprendizagem acessados pelos aprendizes. Definiu-se os módulos de execução das tarefas baseadas no *tracking*, os quais foram implementados na linguagem PHP com banco de dados MySQL. Para que o sistema classificasse informações, os valores numéricos foram normalizados e um Fator de Certeza produzido, juntamente com as regras de utilização das informações geradas.

Embora os resultados tenham sido obtidos através de simulações, os perfis foram gerenciados com êxito. Sistemas educacionais podem aproveitar as informações geradas e armazenadas para oferecer auxílio à aprendizagem dos aprendizes. Esta afirmação é baseada no fato de que nas informações armazenadas no perfil do aprendiz estão inclusos seus estilos de aprendizagem de acordo com Felder [26], objetivos, competências e preferências. Como exemplo de aproveitamento das informações do perfil, apresentou-se o módulo Tutor do LOCAL. Este, pôde utilizá-las para indicar novos Objetos de Aprendizagem que ajudam os aprendizes a atingir os seus Objetivos Educacionais.

## 6.2 Conclusões

A principal contribuição deste trabalho, é o aperfeiçoamento de perfis em ambientes de ensino e aprendizagem ubíquos. Embora o PeLeP tenha sido aplicado ao LOCAL, sua proposta de perfil do aprendiz pode ser utilizada também por outros ambientes de suporte a aprendizagem ubíqua em geral, desde que considerem as suas especificações. A Tabela 6.1 apresenta uma comparação do PeLeP com os principais modelos que foram analisados. Ressalta-se que pelas pesquisas realizadas o PeLeP é a única proposta de aperfeiçoamento de perfis para sistemas ubíquos.

Sistema	Estrutura do perfil	Histórico ( <i>tracking</i> )	Dados	Ambiente	Aperfeiçoamento de perfil
CC/PP	RDF	—	RDF	ubíquo	—
CSCP	RDF	—	RDF	ubíquo	—
SeLeNe	Ontologias	✓	RDF	<i>Web</i>	—
<i>Proem</i>	RDF	—	XML	ubíquo	—
JAPELAS	TANGO	—	RFID	ubíquo	—
GlobalEdu	XML	✓	XML	ubíquo	—
LOCAL	MySQL	✓	MySQL	ubíquo	—
LOCAL + PeLeP	MySQL	✓	MySQL	ubíquo	✓

Tabela 6.1 – Comparação do PeLeP com os principais modelos pesquisados

Nesta dissertação foram identificados os elementos importantes a serem considerados para representar os perfis de aprendizes no ambiente ubíquo. Mostrou-se como os elementos relacionados ao seu contexto e mobilidade foram tratados pelo PeLeP para aperfeiçoar os perfis de acordo com as ações e necessidades dos aprendizes. Neste sentido, as principais conclusões do trabalho foram:

- Existem muitos desafios quando se projetam sistemas para ambiente ubíquos;
- A computação móvel e ubíqua está revolucionando a educação;
- A consciência de contextos é um aspecto tecnológico de fundamental importância na computação ubíqua;

- O perfil dos aprendizes pode reunir informações que os auxiliem no aprendizado;
- Pode-se gerenciar perfis em ambientes ubíquos usando *tracking* para aperfeiçoá-los;
- Aplicações educacionais podem indicar materiais importantes para a aprendizagem baseando-se em perfis.

### 6.3 Trabalhos Futuros

Como primeiro trabalho futuro, pretende-se fazer testes em situações reais na Unisinos<sup>1</sup>. Uma possibilidade é a aplicação da integração LOCAL/PeLeP no âmbito do curso de Engenharia da Computação da universidade. Outra possibilidade é sua aplicação no Programa Interdisciplinar de Pós-Graduação em Computação Aplicada (PIPCA) [73]. Com a integração LOCAL/PeLeP, esses testes podem ser feitos assim que o Servidor de Localização e Informação de Contexto (SeLIC) do LOCAL for concluído. Posteriormente, pensa-se na integração GlobalEdu/PeLeP, que ocorreria de forma parecida com a LOCAL/PeLeP. Pretende-se, portanto, dar ao PeLeP também a tarefa de aperfeiçoar o Modelo de Aprendiz já existente no GlobalEdu.

Não foram encontrados sistemas que controlam perfis de usuários usando as especificações do PeLeP na literatura, inclusive fora do âmbito educacional. Por trabalhar no enriquecimento de perfis na computação ubíqua, pensa-se que num futuro próximo este sistema possa ser utilizado também por empresas que desejarem oferecer serviços personalizados a seus clientes. Como exemplo, o PeLeP instalado em um cinema poderia acrescentar dados em um perfil de usuário e controlá-lo por inferências, fornecendo posteriormente informações sobre novos filmes.

A análise dos sistemas existentes ou em desenvolvimento, revela que já é possível “conversar” com os contextos que um aprendiz visita e atua. Além disso, a rápida proliferação de antenas *wireless* torna previsível uma crescente precisão de localização. Neste sentido, é possível criar novas categorias para o perfil considerado no PeLeP ou acrescentar elementos nas categorias que já foram criadas, de forma

---

<sup>1</sup><http://www.unisinos.br>

a ampliar as informações relevantes para o aprendizado do aprendiz. Dessa forma, poderia ser possível saber que livros o aprendiz retirou, CDs que comprou, vídeos que assistiu, etc.

No meio pedagógico atual consegue-se detectar distúrbios de aprendizagem em alguns adolescentes. Esses distúrbios, quando não detectados na infância prejudicam o aprendizado. Desta forma, pensando num futuro mais distante, pode-se vislumbrar a possibilidade de descobrir à que os aprendizes se adaptam e a forma como eles aprendem ainda na adolescência. Isso poderia ajudar as instituições de ensino a ter aprendizes com objetivos mais claros.

# Bibliografia

- [1] G. D. Abowd and E. D. Mynatt. Charting past, present, and future research in ubiquitous computing. *ACM Transaction on Computer-Human Interaction*, 7 No.1:29–58, 2000.
- [2] ACM. Computing classification system. <https://www.acm.org/class/1998>, Agosto 2006.
- [3] Gustavo Alonso, Harumi Kuno, Fabio Casati, and Vijay Machiraju. *Web Services: Concepts, Architectures and Applications*. Springer, Jan 2004.
- [4] Iara Augustin, Jorge Luis Victória Barbosa, Cláudio Fernando Resin Geyer, and Adenauer Correia Yamin. Isam: a pervasive view in distributed mobile computing. 2002.
- [5] D. Ausubel, editor. *Educational Psychology: A Cognitive View*. New York: Holt, Rinehart and Winston, 2 edition, 1978.
- [6] Aline Baggio. System support for transparency and network-aware adaptation in mobile environments. *SAC'98: Proceedings of the 1998 ACM symposium on Applied Computing.*, New York, NY, USA:405–408, 1998.
- [7] Débora Nice Ferrari Barbosa, Iara Augustin, Adenauer Correia Yamin, Luciano Cavalheiro Silva, Cláudio Fernando Resin Geyer, and Jorge Luis Victória Barbosa. Learning in a large-scale pervasive environment. In *Proceedings of the 2nd PerEl. New York: IEEE Press.*, Pisa, 2006. 2nd IEEE International Workshop on Pervasive Computing.
- [8] Débora Nice Ferrari Barbosa, Cláudio Fernando Resin Geyer, and Jorge Luis Victória Barbosa. Globaledu - an architecture to support learning in

- a pervasive computing environment. In *New Trends and Technologies in Computer-Aided Learning for Computer-Aided Design*. New York: Springer., pages 1–10. IFIP TC10 Working Conference: EduTech 2005, 2005, Perth., 2005.
- [9] Débora Nice Ferrari Barbosa, Cláudio Fernando Resin Geyer, and Jorge Luis Victória Barbosa. Uma proposta de agente pedagógico pessoal pervasivo–consciência da mobilidade e do contexto do aprendiz. In *XVI Simpósio Brasileiro de Informática na Educação*. Porto Alegre : Sociedade Brasileira de Computação, 2005., volume I, pages 63–73. Simpósio Brasileiro de Informática na Educação., 2005.
- [10] F. F. Barbosa. Estilos de ensino e aprendizagem. In *Egatea Digital, Revista da Escola de Engenharia da UFRGS*, volume 85, Porto Alegre, RS, 2001.
- [11] Jorge L. V. Barbosa, Rodrigo M. Hahn, Débora N. F. Barbosa, and Cláudio F. R. Geyer. Mobile and ubiquitous computing in an innovative undergraduate course (to appear, march 2007). In *38th ACM Technical Symposium on Computer Science Education (SIGCSE), 2006, Covington. Proceedings of ACM SIGCSE*, pages 1–5, New York, 2007. ACM Press.
- [12] Jorge L. V. Barbosa, Rodrigo M. Hahn, Solon Rabello, and Débora N. F. Barbosa. Local: Um modelo para suporte à aprendizagem consciente de contexto. In *XVII Simpósio Brasileiro de Informática na Educação (SBIE)*, SBC, Brasília, 2006.
- [13] J. S. Beaudry and Rita Dunn A. Klavas. Survey of research on learning styles. In *Educational Leadership*, volume 46, pages 50–58, USA, Dezembro 1989.
- [14] Bluetooth. How it works. In *The Official Bluetooth Website*. Disponível em: <http://www.bluetooth.com/>. Acessado em Março, 2006.
- [15] Daniel Bonatto. *HNS: Uma solução para Suporte à Execução Distribuída Considerando Aspectos da Pervasividade*. Dissertação de mestrado, Universidade do Vale do Rio dos Sinos – Unisinos, São Leopoldo/RS, Fevereiro 2006.

- [16] D. Brickley and R. Guha. *Resource Description Framework (RDF) Schema Specification 1.0*. <http://www.w3.org/TR/rdf-schema/>, 2000.
- [17] Peter Brusilovsky. Methods and techniques of adaptive hypermedia. *User modeling and user-adapted interaction*, 6(2-3):87-129, July 1996.
- [18] M. Butler. *Implementing Content Negotiation using CC/PP and WAP UAProf HP Laboratories Bristol*. HPL-190, 2001.
- [19] H. Chen, F. Perich, D. Chakraborty, T. Finin, and A. Joshi. Intelligent agents meet semantic web in a smart meeting room. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multi Agent Systems*, New York City, NY, July 19-23 2004. (AAMAS 2004).
- [20] Harry Chen, Tim Finin, and Anupam Joshi. Semantic web in a pervasive context-aware architecture. *artificial inteligente in mobile system*, 2003.
- [21] A. P. Cocco. *Modelo de adaptação de ensino utilizando agentes pedagógicos*. Tese, Instituto de Informática, UFRGS, Porto Alegre. 135f. Tese. (Doutorado em Ciência da Computação.), Porto Alegre, 2004.
- [22] O. Conlan, D. Dagger, and V. Wade. *Towards “anytime, anywhere” Learning: The Role and Realization of Dynamic Terminal Personalization in Adaptive eLearning*. Forbes, 2002.
- [23] Peter Dawabi, Martin Wessner, and Erich Neuhold. Using mobile devices for the classroom of the future. *Learning with mobile devices, research and development.*, pages 55-60, 2004.
- [24] Ricardo de Andrade Kratz. *Fábrica de adequação de conteúdo de ensino para objetos de aprendizagem reutilizáveis (rlos) respeitando a norma scorm*. Dissertação de mestrado, Universidade do Vale do Rio dos Sinos – Unisinos, São Leopoldo, RS, 2006.
- [25] Judith S. Donath. Sociable information spaces. In *Second IEEE International Workshop on Community Networking*, Princeton, USA, 1995. Second IEEE International Workshop on Community Networking.

- [26] R. M. Felder and E. R. Henriques. Learning and teaching styles in foreign and second language education. In *Foreign Language Anais*, volume 28, pages 21–31, 1995.
- [27] R. M. Felder and L. K. Silverman. Learning and teaching styles in engineering education. In *Journal of Engineering Education*, volume 78, pages 674–681, Washington, USA, 1988.
- [28] G. Fischer. User modeling in human-computer interaction. *Journal of User Modeling and User-Adapted Interaction (UMUAI)*, 11 No. 1/2:65–86, 2001.
- [29] N. Friesen. What are educational objects? In *Interactive Learning Environments*, volume 9, Dezembro 2001.
- [30] Isabela Gasparini, Marília A. Amaral, Marcelo S. Pimenta, and José P. M. De Oliveira. Navegação adaptativa no ambiente de ead adaptweb: Uso de xml na adequação de conteúdo e navegação. Congresso Nacional de Ambientes Hipermedia para Aprendizagem, 2004.
- [31] Cláudio F. R. Geyer. Semeai - sistema multiagente de ensino e aprendizagem na internet. Vitória–ES, 2001. XII SBIE 2001 - Simpósio Brasileiro de Informática na Educação.
- [32] J. Giarrato and G. Riley. *Expert Systems - Principles and Programming*. PWS, Boston, 3 edition, 1998.
- [33] R. Grimm, J. Davis, E. Lemar, A. MacBeth, S. Swanson, S. Gribble, T. Person, B. Bershad, G. Borriello, and D. Wetherall. Programming for pervasive computing environments, 2001.
- [34] A. Held, S. Buchholz, and A. Schill. *Modeling of Context Information for Pervasive Computing Applications*. Proc. of the 6th World Multiconference on Systemics, Cybernetics; Informatics (SCI), 2002.
- [35] K. Henricksen and J. Indulska. Modelling and using imperfect context information. *Context Modelling and Reasoning (CoMoRea)*, 2nd IEEE



- Conference on Pervasive Computing and Communications (PerCom)*, 1:33 – 37, 2004.
- [36] K. Henriksen and J. Indulska. A software engineering framework for context-aware pervasive computing. *2nd IEEE Conference on Pervasive Computing and Communications (PerCom)*, 1, 2004.
- [37] IEEE. Learning technology standards committee. Disponível em: <http://ieeeltsc.org/>. Acessado em Outubro, 2006.
- [38] Project ISAM. Project isam-infra-estrutura de suporte às aplicações móveis. Disponível em: <http://www.inf.ufrgs.br/~isam>. Acesso em Março, 2006.
- [39] Peder Jacobsen. Reusable learning objects. Edição de e-Learning Magazine, Novembro 2001.
- [40] Jena. Jena – a semantic web framework for java. Disponível em: <http://jena.sourceforge.net>, Setembro 2006.
- [41] G. Judd and P. Steenkiste. Providing contextual information to pervasive computing applications. *1st IEEE Conference on Pervasive Computing and Communications (PerCom)*, pages 133–142, 2003.
- [42] T. C. Kao, J. P. Sheu, C. Y. Chiang, and Y. S. Chen. A mobile scaffolding-aid-based bird-watching learning system. In *IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE'02)*, IEEE Computer Society Press., pages 15–22, 2002.
- [43] R. H. Katz. Adaptation and mobility in wireless information systems. *IEEE Personal Communications*, 1:6–17, 1994.
- [44] G. Klyne, F. Reynolds, C. Woodrow, and H. Ohto. *Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies*. W3C Working Draft, (URL: <http://www.w3.org/TR/2004/REC-CCPP-struct-vocab-20040115/>), 2004.

- [45] M. Levene, D. Peterson, and K. Keenoy. Personalisation and trails in self e-learning networks. In *SeLeNe: Self e-Learning Networks*. Disponível em: <http://www.dcs.bbk.ac.uk/selene/>, 2004.
- [46] LIP. Learner information package specification 1.0. 2001. Disponível em: <http://www.imsglobal.org/metadata/index.cfm>, Março 2006.
- [47] W. Longmire. A primer on learning objects. American Society for Training and Development. Virginia, USA., 2001.
- [48] João Lopes. Normalização. <http://bsel.ist.utl.pt/2004/PortalQuimiometria/Contents/procdados>, Fevereiro 2006.
- [49] Wilma Maria Guimarães Lopes. IIs-inventário de estilos de aprendizagem de felder-saloman: Investigação de sua validade em estudantes universitários de belo horizonte. Dissertação de mestrado, Universidade Federal de Santa Catarina, Florianópolis, SC, Agosto 2002.
- [50] K. Lyytinen and Y. Yoo. Issues and challenges in ubiquitous computing, communications of acm. *ACM Transaction on Computer-Human Interaction*, 45 No. 12:63–65, 2002.
- [51] Geraldo Robson Mateus and Antonio Alfredo Ferreira Loureiro. *Introdução à Computação Móvel*, volume 1. Décima Primeira Escola de Computação, 2 edition, 1998.
- [52] MobiLab. Mobile computing lab: Research e development. Disponível em: <http://www.inf.Unisinos.br/mobilab/>, Acessado em Maio 2006.
- [53] D. Musa and J. Palazzo. Sharing learner information through a web services based learning architecture. In Latvia Riga, editor, *Web Information Systems Modeling*, pages 122–131, WISM, 2004, 2004. CAiSE Workshops.
- [54] C. Nikolopoulos. *Expert Systems - Introduction to First and Second Generation and Hybrid Knowledge Based Systems*. New York, 1997.

- [55] Learning Technology Standards Committee of the IEEE. Draft standard for learning object metadata. <http://ltsc.ieee.org/wg12/index.html>, Setembro 2004.
- [56] Hiroaki Ogata, K. Matsuura, and Yoneo Yano. Sharlok: Bridging learners through active knowledge awareness. *Computer*, 1(9):601–610, 1996.
- [57] Hiroaki Ogata and Yoneo Yano. How ubiquitous computing can support language learning. *Proceedings of KEST*, pages 1–6, 2003.
- [58] G. Papamarkos, A. Poulouvasilis, M. Levene, D. Peterson, P. T. Wood, G. Loizou, and K. Keenoy. Funcionalidade, user requirements and exploitation scenarios. In *SeLeNe: Self e-Learning Networks*. Disponível em: <http://www.dcs.bbk.ac.uk/selene/>, 2002.
- [59] PAPI. Public and private information for learners. IEEE P1484.2/d7, 2001. Draft standard for learning technology. Disponível em: <http://www.edutool.com/papi/>, Março 2006.
- [60] Otto Peters. A educação a distância em transição. volume 1 of *ISBN 85-7431-0670*, São Leopoldo-RS, 2003.
- [61] G. Roman, A. Murphy, and G. Picco. A software engineering perspective on mobility, 2000.
- [62] J. Roschelle and R. Pea. A walk on the wild side: How wireless handhelds may change cscl, 2002.
- [63] M. Rosenberg. *E-Learning. Strategies for Delivering Knowledge in the Digital Age*. New York, McGraw-Hill, 2001.
- [64] D. Saha and Amitava Mukherjee. Pervasive computing: A paradigm for the 21st century. *IEEE Computer*, 36(3):25–31, 2003.
- [65] M. Satyanarayanan. Pervasive computing: vision and challenges. *IEEE [see also IEEE Wireless Communications]*, 8(4):10–17, 2001.

- [66] A. Schmidt and K. Van Laerhoven. How to build smart appliances. *IEEE Personal Communications, Special Issue on Pervasive Computing*, 8(4):66 – 71, 2001.
- [67] B. F. Skinner. *Sobre o Behaviorismo*. Cultrix, São Paulo, 1982.
- [68] E. Soloway, K. Luchini, W. Bobrowsky, C. Quintana, and M. Curtis. Handheld use in k-12: A descriptive account. In *IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE'02)*, IEEE Computer Society Press., pages 23–30, 2002.
- [69] Yi Soojin, Naldurg Prasad, and Kravets Robin. Security-aware ad-hoc routing for wireless networks. In *The Second ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'01)*, agosto 2001.
- [70] Deborah Tatar, Jeremy Roschelle, Phil Vahey, and William R. Penuel. Handhelds go to school: Lessons learned. *Computer*, 36(9):30–37, 2003.
- [71] T. G. C. Thompson, Z. Segall, and G. Kortuem. Close encounters: Supporting mobile collaboration through interchange of user profiles. In 1999 H/W Gellersen, Springer, editor, *Germany and Lecture notes in computer science*, volume 1707 of ISBN 3-540-66550-1. Proceedings First International Symposium on Handheld and Ubiquitous Computing (HUC99), 1999, Karlsruhe, 1999.
- [72] WAG UAProf. *Wireless Application Protocol Forum Ltd*. Proposed Version 30-May, (URL: <http://www1.wapforum.org/tech/documents/SPEC-UAProf-19991110.pdf>), 2001.
- [73] PIPCA Unisinos. Programa interdisciplinar de pós-graduação em computação aplicada. Disponível em: <http://www.inf.unisinos.br/pipca/>, Acessado em Maio 2006.
- [74] Roy Want, Bill Schilit, and Norman Adams. Context-aware computing applications. In *Workshop on Mobile Computing Systems and Applications*, pages 85–90. IEEE Computer Society, 1994.

- [75] Mark Weiser. The computer for the 21st century. *Scientific American*, Setembro, 1991.
- [76] WiFi-Planet. Maximizing wireless lan performance. Disponível em: <http://www.wi-fiplanet.com/>. Acessado em Março, 2006.
- [77] Chengjiu Yin, Hiroaki Ogata, and Yoneo Yano. Japelas: Supporting japanese polite expressions learning using pda towards ubiquitous learning. *The Journal of Information and Systems in Education*, 3(1):33–39, 2005.
- [78] Chengjiu Yin, Hiroaki Ogata, and Yoneo Yano. Ubiquitous-learning system for the japanese polite expressions. *wmte*, 0:269–273, 2005.
- [79] Zhang Yin and Wenke Lee. Intrusion detection in wireless ad hoc networks. In *MobiCom*, agosto 2000.

## Apêndice A

# Exemplo de *Tracking* representado em XML

```

001    <?xml version=1.0 encoding=latin1?>
002    <locationsystem>
003        <userid>322</userid>
004        <nome>Rodrigo Dormento</nome>
005        <localizacoes>
006            <sala>
007                <id>Sala 209</id>
008                <objetos>
009                    <item>
010                        <id>10</id>
011                        <tipo>OArp</tipo>
012                        <estilo>Entrada</estilo>
013                        <nome>Java para iniciantes</nome>
014                        <palavrasChave>Java, classes, applet</palavrasChave>
015                        <objetivosEducacionais>Conseguir programar rotinas simples
016                        com a linguagem Java</objetivosEducacionais>
017                        <prerequisitos>Métodos e comandos da programação
018                        </prerequisitos>
019                        <resumoAssunto>Descobrir todo o potencial da linguagem Java,
020                        com sua versatilidade, dinâmica e eficiência a serviço do
021                        programador que procura sempre um padrão superior no
022                        desenvolvimento de seus projetos.</resumoAssunto>
023                        <tempo>145222</tempo>
024                        <nAcessos>28</nAcessos>
025                        <nDias>3</nDias>
026                    </item>
027                    <item>
028                        <id>11</id>
029                        <tipo>OArp</tipo>
030                        <estilo>Compreensão</estilo>
031                        <nome>PHP para iniciantes</nome>
032                        <palavrasChave>PHP, classes, applet</palavrasChave>
033                        <objetivosEducacionais>Conseguir programar rotinas simples
034                        com a linguagem PHP</objetivosEducacionais>
035                        <prerequisitos>Instalação do Apache e versões do PHP
036                        </prerequisitos>

```

```

037     <resumoAssunto>Descobrir tudo que se pode fazer com
038     a linguagem PHP.
039     </resumoAssunto>
040     <tempo>62451</tempo>
041     <nAcessos>11</nAcessos>
042     <nDias>3</nDias>
043 </item>
044 <item>
045     <id>12</id>
046     <tipo>OArp</tipo>
047     <estilo>Percepção</estilo>
048     <nome>Introdução a C++</nome>
049     <palavrasChave>C++, classes, bibliotecas</palavrasChave>
050     <objetivosEducacionais>Conseguir programar rotinas simples
051     com a linguagem C++</objetivosEducacionais>
052     <prerequisitos>Métodos e estruturas da linguagem C++
053     </prerequisitos>
054     <resumoAssunto>Descobrir tudo que se pode fazer com a
055     linguagem C++.</resumoAssunto>
056     <tempo>31910</tempo>
057     <nAcessos>3</nAcessos>
058     <nDias>2</nDias>
059 </item>
060 <item>
061     <id>13</id>
062     <tipo>OAs</tipo>
063     <estilo>Processamento</estilo>
064     <nome>C# para iniciantes</nome>
065     <palavrasChave>C#, classes</palavrasChave>
066     <objetivosEducacionais>Conseguir programar rotinas simples
067     com a linguagem C#</objetivosEducacionais>
068     <prerequisitos>Introdução a C#: conhecimentos básicos
069     </prerequisitos>
070     <resumoAssunto>Descobrir tudo que se pode fazer com a
071     linguagem C#.</resumoAssunto>
072     <tempo>70731</tempo>
073     <nAcessos>14</nAcessos>
074     <nDias>7</nDias>
075 </item>
076 <item>
077     <id>14</id>
078     <tipo>OArp</tipo>
079     <estilo>Processamento</estilo>
080     <nome>Delphi 6.0</nome>
081     <palavrasChave>Delphi, classes</palavrasChave>
082     <objetivosEducacionais>Conseguir programar rotinas simples
083     com a linguagem Delphi 6.0</objetivosEducacionais>
084     <prerequisitos>Introdução a linguagem Delphi</prerequisitos>
085     <resumoAssunto>Conhecer a ferramenta Delphi.</resumoAssunto>
086     <tempo>47852</tempo>
087     <nAcessos>17</nAcessos>
088     <nDias>5</nDias>
089 </item>
090 <item>
091     <id>15</id>
092     <tipo>OArp</tipo>
093     <estilo>Compreensão</estilo>
094     <nome>Padrão IEEE 402.11a</nome>

```

```

095      <palavrasChave>IEEE 402.11a, captação de frequências
096      </palavrasChave>
097      <objetivosEducacionais>Conseguir implementar um sistema de
098      localização com o padrão IEEE 402.11a
099      </objetivosEducacionais>
100      <prerequisitos>Padroes IEEE</prerequisitos>
101      <resumoAssunto>Este OA aborda o padrão IEEE 402.11a.
102      </resumoAssunto>
103      <tempo>85762</tempo>
104      <nAcessos>1</nAcessos>
105      <nDias>5</nDias>
106      </item>
107      </objetos>
108      <dispositivos>
109      <item>
110          <nome>Tablet</nome>
111          <tempo>53516</tempo>
112          <nAcessos>41</nAcessos>
113          <nDias>5</nDias>
114      </item>
115      </dispositivos>
116      <aplicativos>
117      <item>
118          <nome>Word</nome>
119          <tempo>114050</tempo>
120          <nAcessos>10</nAcessos>
121          <nDias>7</nDias>
122      </item>
123      <item>
124          <nome>Excel</nome>
125          <tempo>461838</tempo>
126          <nAcessos>22</nAcessos>
127          <nDias>7</nDias>
128      </item>
129      <item>
130          <nome>Mozilla</nome>
131          <tempo>451135</tempo>
132          <nAcessos>14</nAcessos>
133          <nDias>4</nDias>
134      </item>
135      <item>
136          <nome>Internet Explorer</nome>
137          <tempo>104942</tempo>
138          <nAcessos>46</nAcessos>
139          <nDias>2</nDias>
140      </item>
141      <item>
142          <nome>PowerPoint</nome>
143          <tempo>48063</tempo>
144          <nAcessos>18</nAcessos>
145          <nDias>3</nDias>
146      </item>
147      <item>
148          <nome>Acrobat</nome>
149          <tempo>287747</tempo>
150          <nAcessos>12</nAcessos>
151          <nDias>1</nDias>
152      </item>
153      <item>

```



```
154         <nome>GTalk</nome>
155         <tempo>486064</tempo>
156         <nAcessos>35</nAcessos>
157         <nDias>4</nDias>
158     </item>
159     <item>
160         <nome>MSN</nome>
161         <tempo>395617</tempo>
162         <nAcessos>34</nAcessos>
163         <nDias>5</nDias>
164     </item>
165 </aplicativos>
166 </sala>
167 </localizacoes>
168 </locationsystem>
```

Figura A.1 – Exemplo de *tracking* do aprendiz