

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE ESPECIALIZAÇÃO EM WEB E SISTEMAS DE INFORMAÇÃO

FELIPE AZZOLIN BASTOS DA SILVA

**Proposta de um Agente Gestor de Contexto
Hospitalar**

Trabalho de Conclusão apresentado como
requisito parcial para a obtenção do grau de
Especialista

Prof. Dr. Cláudio Fernando Resin Geyer
Orientador

Prof. Msc. Luciano Cavalheiro da Silva
Co-orientador

Prof. Dr. Carlos Alberto Heuser
Coordenador do Curso

Porto Alegre, dezembro de 2007.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. José Carlos Ferraz Hennemann

Vice-Reitor: Prof. Pedro Cezar Dutra Fonseca

Pró-Reitora de Pós-Graduação: Profa. Valquiria Linck Bassani

Diretor do Instituto de Informática: Prof. Flávio Rech Wagner

Coordenador do WEBSIS: Prof. Carlos Alberto Heuser

Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

AGRADECIMENTOS

Agradeço a meus familiares e amigos próximos pela compreensão e companheirismo. Dedico um agradecimento especial ao Doutorando Luciano Cavalheiro da Silva pela atenção despendida ao desenvolvimento deste trabalho. Finalizo agradecendo aos professores do curso de Especialização da UFRGS.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS	5
LISTA DE FIGURAS	6
RESUMO.....	7
ABSTRACT.....	8
1 INTRODUÇÃO	9
1.1 Objetivo	10
2 TECNOLOGIAS DE APOIO	11
2.1 Computação Pervasiva.....	11
2.2 Projetos Relacionados	12
2.2.1 Japelas.....	12
2.2.2 GlobalEdu.....	13
2.2.3 CHIS – Context-aware Hospital Information System	14
2.2.4 WARD-IN-HAND.....	15
3 AGENTE GESTOR CONTEXTO HOSPITALAR (AGCH).....	17
3.1 HCPA – Atendimento a beira do leito	17
3.2 Modelagem	19
4 ADAPTAÇÃO DA APLICAÇÃO.....	25
4.1 Adaptação da Aplicação.....	25
4.2 Identificação do dispositivo do usuário	27
4.3 Adaptação da aplicação quanto o Hardware e Software – Exemplos	28
5 CONTRIBUIÇÕES DE CONSIDERAÇÕES FINAIS.....	35
REFERÊNCIAS.....	36

LISTA DE ABREVIATURAS E SIGLAS

AGCH	Agente Gestor de Contexto Hospitalar
AJAX	Asynchronous Javascript And XML
AP	Agente Pedagógico
BD	Banco de Dados
CHIS	Context-aware Hospital Information System
CSS	Cascading Style Sheets
DOM	Document Object Model
GPS	Global Positioning System
HCPA	Hospital de Clinicas de Porto Alegre
HIS	Hospital Information Systems
HTML	HiperText Markup Language
HTTP	HiperText Transfer Protocol
IP	Internet Protocol
J2EE	Java 2 Plataform, Enterprise Edition
J2ME	Java 2 Plataform, Micro Edition
J2SE	Java 2 Plataform, Standard Edition
PDA	Personal Digital Assistant
PEP	Prontuário Eletrônico do Paciente
RFID	Radio Frequency Identification
SAMIS	Serviço de Arquivo Médico e Informações de Saúde
SSH	Secure Shell
UFRGS	Universidade Federal do Rio Grande do Sul
UML	Unified Modeling Language
XML	Extended Markup Language

LISTA DE FIGURAS

Figura 2.1: Consolidação do cenário da Computação Pervasiva.....	12
Figura 2.2. Arquitetura GlobalEdu	14
Figura 2.3: Context-aware client indicando a presença de usuários como (a) lista ou (b) mapa.....	15
Figura 3.1: Prontuário do Paciente em papel e eletrônico (PEP)	18
Figura 3.2: Atendimento a Beira do Leito do Paciente	19
Figura 3.3: Funcionalidades do AGCH.	20
Figura 3.4: Encaminha Tarefa.	21
Figura 3.5: Solicita Tarefa.	22
Figura 3.6: Gerência informações do paciente.	23
Figura 3.7: Agente Gestor de Contexto Hospitalar.	24
Figura 4.1: Modelo de aplicação do Ajax.....	27
Figura 4.2: Parte perfil Motorola V360.	33
Figura 4.3. Distribuição de medicamentos de forma adaptada ao dispositivo.	33

RESUMO

O presente trabalho propõe a criação de um modelo que visa representar conhecimento de contexto em um domínio específico. O domínio em questão apresenta um cenário de um complexo hospitalar com seus elementos de contexto, tais como pessoas, eventos e recursos. Além disso, considerando a elevada heterogeneidade de dispositivos, que podem vir a ser utilizados pelos usuários presentes dentro deste cenário, propõe-se um estudo sobre a adaptação da aplicação encaminhada ao equipamento empregado. Essa adaptação deve ocorrer de forma ajustada ao perfil do usuário, agregando o maior número de recursos aceitáveis pelo aparelho, permitindo o acesso à aplicação independente do dispositivo utilizado. Por fim, identificar a precisa localização do usuário torna-se fato relevante na apresentação do contexto que o abrange, para tanto, deve ser prevista na modelagem a localização das pessoas que fazem parte do cotidiano do cenário.

Palavras-chave: localização, gerência de contexto, adaptação da aplicação.

Proposal of an Agent Manager of Hospital Context

ABSTRACT

This work proposes a model that aim to represent knowledge of the context in a specific domain. The domain in question presents a scene of a hospital complex with its elements of context, such as people, events and resources. Furthermore, considering the high heterogeneity of devices, which can be used by users in this present scenario, it is proposed a study on the adaptation of the application forwarded to the employed equipment. This adaptation must be adjusted to the user's profile, aggregating the largest number of acceptable resources by the unit to access the application independent of the device used. Finally, identify the precise location of the user becomes truly relevant in the context of the presentation that covers this will be provided in modeling the location the people who are part of the daily scene. Finally, identify the precise location of the user becomes truly relevant in the context of the presentation that encloses it, for both, must be provided in modeling the location of the people who are part of the daily scene.

Keywords: location, context management, adaptation of the application.

1 INTRODUÇÃO

O desenvolvimento tecnológico e a diversificação de recursos computacionais vêm gerando o crescimento da heterogeneidade de dispositivos disponíveis. Estes, além de variados, estão se tornando mais populares e acessíveis, dentre eles, pode-se citar *desktops*, *notebooks*, equipamentos de rede, PDA's (*Personal Digital Assistants*), celulares, sensores, entre outros. Essa diversidade de recursos faz com que o usuário tenha acesso a uma determinada aplicação utilizando tanto dispositivos mais simples, até tecnologias com maior poder de processamento. Dessa forma, surge, em alguns casos, certa dificuldade na interoperabilidade das aplicações utilizadas pelo usuário. Com isso, atender a essa necessidade de forma adaptada ao contexto, vem como premissa para novas aplicações que visam à integração do ambiente computacional do indivíduo.

Dada à dinâmica aplicada a esses cenários, o desafio torna-se projetar aplicações que prezam pela conectividade, portabilidade e mobilidade. Sendo que, todos esses requisitos buscam atingir o conceito de adaptação, pois mobilidade exige adaptabilidade (SATYANARAYANAN, 1996). Outro fato relevante, inerente à computação e que deve ser considerado, encontra-se na dificuldade de acesso a dados que não estejam contidos em dispositivos acessíveis ao usuário em um dado momento. Essa dificuldade ocorre devido à inexistência de comunicação física, em determinadas situações, entre os dispositivos ao alcance do usuário e as informações desejadas, ou devido à impossibilidade lógica (de software) dessa comunicação. Ou seja, uma estrutura que permita a comunicação entre dispositivos com diferentes arquiteturas e seus componentes de aplicações e serviços (YAMIN, 2004).

Embora a diversidade de equipamentos crie certas dificuldades de acesso a dados distintos, não se pode deixar de ressaltar os benefícios gerados pela mesma. Dessa forma, para solucionar os problemas mencionados anteriormente, e extrair ao máximo os benefícios gerados pela variação de dispositivos computacionais, surge um novo paradigma, a Computação Pervasiva (SATYANARAYANAN, 2001). Esse paradigma visa oferecer distribuição, heterogeneidade, mobilidade e a adaptação aos usuários na execução de suas tarefas, através de um acesso uniforme e imediato às informações. A Computação Pervasiva pode ser tratada como a visão da computação do futuro, na qual, os recursos computacionais (ambientes, dados, aplicações, etc) estarão disponíveis em qualquer local envolvido pelo cenário pervasivo (AUGUSTIN, 2004). Esse paradigma fundamenta-se em cinco elementos principais, os quais devem ser supridos para que a mesma constitua-se como tal: Rede, Elevada Heterogeneidade, Mobilidade, Disponibilidade de Serviços e Dados e Adaptação (YAMIN, 2004).

Tendo em vista as possibilidades criadas pela computação pervasiva, considera-se que conhecer o contexto no qual o indivíduo está inserido, é fundamental tanto na

adaptação da aplicação ao dispositivo, quanto para relacionar e disponibilizar informações deste contexto ao usuário. Assim, além de maior flexibilidade nas aplicações, surge a possibilidade de criar-se um modelo com informações que fazem parte do cenário no qual encontra-se o usuário. Alguns trabalhos acadêmicos, tais como, Japelas (OGATA, 2004) e GlobalEdu (BARBOSA, 2007), buscam por uma proposta na qual os recursos são tratados de forma contextualizada.

Existem técnicas que permitem a análise de requisitos e representação de conhecimento para a construção de modelos de domínios específicos, como diagramas UML (*Unified Modeling Language*). A utilização dessa linguagem permite que a base de conhecimento, a ser inferida no modelo previsto, possa ser convertida com determinada facilidade, em diferentes modelos que pretendam representar cenários semelhantes ao proposto. Outro sim, possibilita a geração de aplicações orientadas a objetos com base em sua estrutura.

1.1 Objetivo

Considerando os aspectos relacionados ao desenvolvimento tecnológico e científico mencionados, este trabalho tem por objetivo geral, criar um modelo de contexto para um domínio específico, neste caso, o atendimento efetuado a beira do leito do paciente no Hospital de Clínicas de Porto Alegre (HCPA). Nesse caso, propõe-se a modelagem de um Agente que deve auxiliar os sistemas de gestão hospitalar já existentes, tendo como atributos informações de contexto relacionadas a pessoas, recursos, eventos e suas localizações específicas. Além disso, pretende-se desenvolver uma forma na qual a aplicação possa ser adaptada aos recursos do dispositivo utilizado. Assim, diferentes formatos da mesma aplicação estarão disponíveis, mas apenas um será apresentado, de acordo com a capacidade proporcionada pelo equipamento em questão.

2 TECNOLOGIAS DE APOIO

Acompanhando o desenvolvimento tecnológico crescente, nota-se o aumento constante da demanda por sistemas que visam disponibilizar informações de contexto ao usuário. Observa-se que novas técnicas permitem a evolução de aplicações que buscam atender tal demanda, oferecendo informações de forma pervasiva e principalmente de forma contextualizada a esse universo. Surgem assim, novas aplicações com suporte da computação pervasiva, adaptativas por natureza, uma vez que necessitam alterar sua forma de execução à medida que seu contexto se altera (YAMIN, 2004).

2.1 Computação Pervasiva

O termo Computação Pervasiva (*Pervasive Computing*), foi sugerido inicialmente, pela IBM em 1999, através de uma publicação em um de seus periódicos.

Este paradigma agrega recursos oferecidos pela computação em grade e computação móvel. Trata-se de uma proposta na qual, o usuário pode ter acesso ao seu ambiente computacional independente de sua localização ou dispositivo (SAHA, 2003). Uma de suas premissas é tratar a computação não apenas do ponto de vista de dispositivos, e sim, como ambiente computacional que envolve o usuário. A Computação Pervasiva visa atingir um cenário no qual os recursos computacionais (ambientes, dados, aplicações, etc) estarão disponíveis em qualquer local envolvido pelo ambiente pervasivo (AUGUSTIN, 2004).

A computação pervasiva, segundo Yamin (2004), fundamenta-se em cinco elementos básicos para sua constituição, representados na figura 2.1, sendo eles:

Elementos de Rede: faz um link físico e lógico entre o usuário e dispositivos ou dados desejados pelo mesmo, sejam recursos próprios ou compartilhados. Uma rede pervasiva compreende arquiteturas de redes estruturadas (com cabos) interligadas a outras arquiteturas sem fio.

Elevada Heterogeneidade: representa a diversidade de dispositivos eletro-eletrônicos existentes, tais como: *notebooks*, PDA's, *smartphones*, *desktops*, dispositivos de rede, sensores, dentre outros, os quais possuem as mais variadas arquiteturas de hardware (YAMIN, 2004). A Computação Pervasiva possibilita que os dados do usuário sejam centralizados dentro do ambiente computacional pervasivo. Dessa forma, os mesmos estarão disponíveis em qualquer dispositivo acessado.

Mobilidade: refere-se ao deslocamento do usuário. Esta é dividida em duas categorias, mobilidade física e lógica (LUNARDI, 2001), sendo a física relacionada ao deslocamento do usuário pelo perímetro da rede. Já a mobilidade lógica, consiste na premissa de que o usuário troque seus componentes de software, independente de sua

posição geográfica, dentro da rede. Quando se fala em mobilidade lógica, pensa-se em mobilidade de software, aplicações e seu contexto (FUGGETA, 1998).

Disponibilidade de Dados e Serviços: o ambiente computacional do usuário deve estar sempre disponível dentro do ambiente pervasivo. De forma que, toda a necessidade de dados, serviços ou recursos solicitados pelo usuário, indiferente do dispositivo utilizado, deve ser atendida. Isso deve se dar independente da localização do usuário dentro da infra-estrutura da rede.

Adaptação: consiste em que o sistema deva prover adaptação, tanto no que se refere à aplicação, quanto ao sistema. Na primeira delas, a aplicação que estiver em execução no ambiente pervasivo, deve moldar-se às necessidades do usuário, independente do tipo de dispositivo computacional que o mesmo esteja utilizando. Ou ainda, caso o usuário queira uma forma diferente de comunicação com a aplicação. Já a segunda, refere-se ao fato do dispositivo se adaptar ao contexto, por exemplo, diminuindo o processamento para economizar bateria, ou ainda, suspender determinado serviço, quando o usuário não utilizá-lo efetivamente, colocando a aplicação em espera.

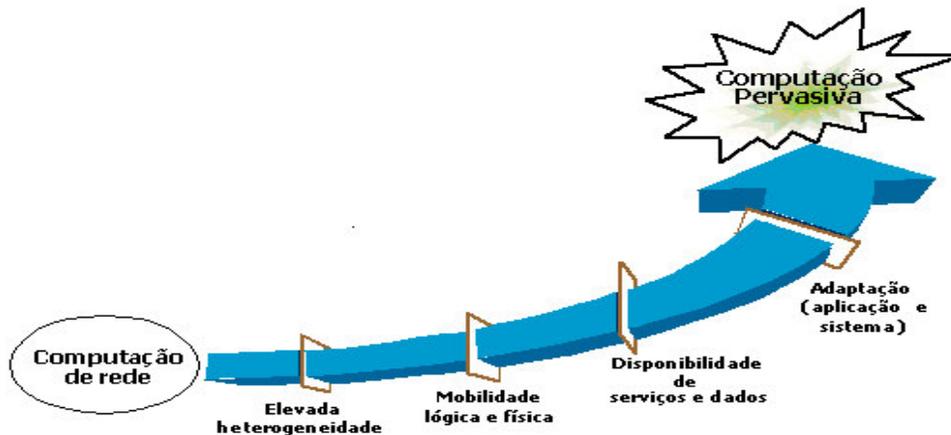


Figura 2.1: Consolidação do cenário da Computação Pervasiva (YAMIN,2004).

2.2 Projetos Relacionados

Alguns projetos, de âmbito mundial, seguem em direção a uma proposta de cenários que disponibilizam informações de forma contextualizada ao usuário, utilizando para isto, recursos da web e/ou pervasivos. Projetos como, Japelas (OGATA, 2004) e GlobalEdu (BARBOSA, 2007), de linhas acadêmicas, implementam o paradigma em um contexto educacional. Já CHIS (*Context-aware hospital information system*) (MUÑOZ, RODRIGUEZ, et al, 2003) e Ward-In-Hand (ANCONA, DODERO, ET, 2000) seguem o foco na área médica hospitalar.

2.2.1 Japelas

Japelas é um sistema consciente do contexto, em um ambiente pervasivo, utilizado para dar suporte ao ensino de expressões de tratamento na língua japonesa. Estudantes estrangeiros, em universidades japonesas, portando PDA's são auxiliados a identificar qual a expressão mais adequada a utilizar, de acordo com o contexto em que se encontram. Portando PDAs, os alunos são assistidos pelo sistema a identificar qual

expressão deve ser utilizada ao se dirigirem a uma determinada pessoa, considerando o contexto em que se encontram.

Sua arquitetura foi desenvolvida para dispositivos móveis, sendo composta dos seguintes módulos (OGATA, 2004):

Modelo do aluno: gerencia o perfil do aluno com dados fornecidos pelo aprendiz. Guarda ainda, as interações durante o uso do sistema, utilizando-as posteriormente para a aprendizagem individual fora do contexto real de interação.

Modelo ambiental: gerencia dados do local onde o aluno está (contexto), o ambiente é detectado pelo módulo de Gerência de Posição, usando o sistema de RFID (*Radio Frequency Identification*) ou GPS (*Global Positioning System*) dos dispositivos. A posição é usada para determinar a formalidade da expressão;

Modelo educacional: controla a base de conhecimento de expressões a serem utilizadas. As expressões básicas são inseridas pelo professor. Alunos e professores podem adicionar ou modificar expressões durante o uso.

Gerente de posição: responsável pela identificação da posição do aluno em um determinado local. O sistema de RFID é usado para identificar o aluno dentro de prédios, enquanto que o GPS é usado para localizá-lo ao ar livre. No experimento realizado pelos pesquisadores do projeto, sistemas de RFID foram acoplados nas portas de entrada das salas de estudo, de forma que estas pudessem ser identificadas;

Recomendação da expressão: este módulo é responsável por fornecer a expressão apropriada a uma dada situação.

2.2.2 GlobalEdu

O projeto GlobalEdu tem como premissa, fornecer subsídios para que o aprendiz possa estabelecer o processo de aprendizagem de forma adaptada ao seu contexto, em qualquer lugar, a qualquer tempo. Considera que elementos característicos do ambiente, no qual o aprendiz está inserido, devem ser integrados ao processo educacional. Tornando assim, a aprendizagem mais significativa, de forma que, as habilidades e competências são desenvolvidas a partir dos seus objetivos (BARBOSA, 2007).

As informações de contexto dentro do GlobalEdu, dizem respeito tanto a aspectos sociais (contexto social), quanto a aspectos físicos (contexto físico). Sendo que as de contexto social estão relacionadas a pessoas, recursos e eventos presentes em um determinado local. As informações de contexto físico referem-se a aspectos como rede, banda, localização, dispositivo utilizado, ou seja, informam o que é de interesse da execução. Esses elementos são fornecidos pelo ambiente pervasivo. Então, de acordo com a movimentação do aprendiz, novas informações de contextos sociais vão ser acessadas. Esses serão repassados ao usuário caso façam parte de seus objetivos.

Sua arquitetura é estabelecida através de três camadas (BARBOSA, 2007), sua estrutura é ilustrada na figura 2.2:

Camada de Aplicação: representada por um Agente Pedagógico (AP) que auxilia a aprendiz em suas interações.

Camada de Sistema: conjunto de módulos necessários para prover o processo educacional, manipulando informações do contexto do aprendiz, seu perfil e seus conteúdos. Além disso, auxilia na execução do AP e dos recursos que o manipulam. Esta camada se divide em dois conjuntos de módulos, Módulos Educacionais,

responsáveis pelo armazenamento e gerenciamento das informações educacionais e Módulos de Suporte, responsáveis pelos elementos de execução do AP e demais módulos da arquitetura. Ressalva para o Módulo de Suporte Comunicação, que faz a comunicação entre o GlobalEdu e o *middleware* de execução, atenuando a complexidade da comunicação entre a Camada de Sistema e a Terceira Camada, de Execução.

Camada de Execução: faz o suporte aos elementos necessários para atender as funcionalidades básicas do GlobalEdu, como percepção do contexto físico, suporte a mobilidade do aprendiz, suporte a comunicação e acesso ao sistema em escala global (o AP acessa recursos da rede independente de sua localização ou tipo de deslocamento).

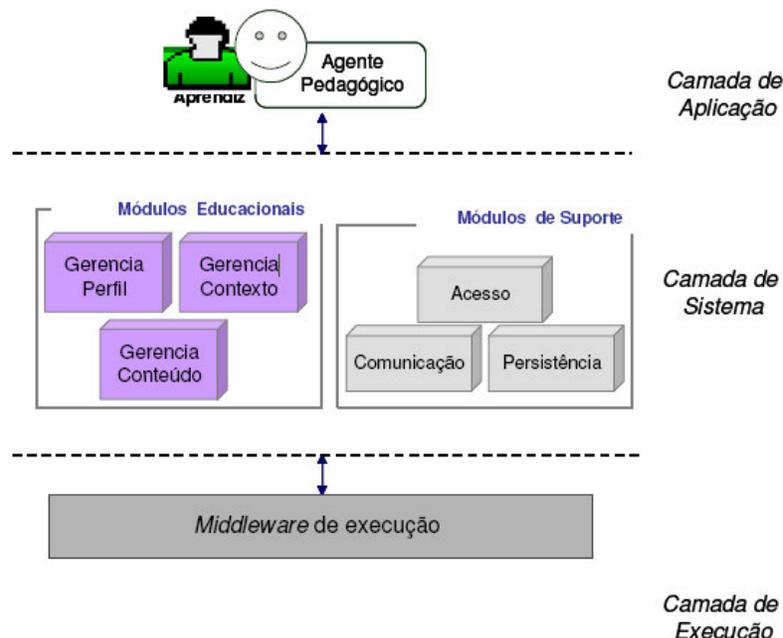


Figura 2.2: Arquitetura GlobalEdu (Barbosa, 2007).

2.2.3 CHIS – Context-aware Hospital Information System

Avaliando a dinâmica existente nas rotinas de sistemas de informação em complexos hospitalares, foi observada pelos autores, a constante necessidade de troca de informações entre diferentes setores e profissionais deste ambiente. Além disso, foi considerada a necessidade de padronizar-se as informações inseridas no sistema, de forma que usuários de áreas distintas pudessem ter a mesma compreensão das informações.

Considerando estes aspectos, o CHIS propõe uma maneira de integrar informações (resultados de laboratório, registros de pacientes, etc), recursos (equipamentos, medicamentos, etc), pessoas (profissionais da área e pacientes), e disponibilizá-las de forma contextualizada aos interesses de cada usuário. Essa forma de comunicação é prevista para um ambiente focado em mobilidade, no qual usuários portando *handhelds* têm acesso a dados do seu contexto, de acordo com sua localização específica. Citando como exemplo dessa aplicação, a seguinte situação: um médico ao aproximar-se de um paciente de sua alçada, recebe automaticamente todas as informações referentes ao registro deste paciente, sejam informações básicas, resultados de exames, dados inseridos por outros profissionais que tratam o mesmo paciente.

A estrutura do CHIS foi desenvolvida, assim como Japelas, seção 2.2.1, ou seja, direcionada a dispositivos móveis, sendo composta dos seguintes elementos (MUÑOZ, RODRIGUEZ, ET, 2003):

Context-Aware Client: estima a localização de médicos e enfermeiros, através de seus dispositivos móveis, permitindo-lhes efetuar solicitações ao sistema, bem como, comunicar-se com outros membros da equipe, os quais são apresentados também com sua localização específica. A definição do posicionamento dos outros usuários pode ser visualizada de duas formas, através de uma lista ou de um mapa, conforme figura 2.3.

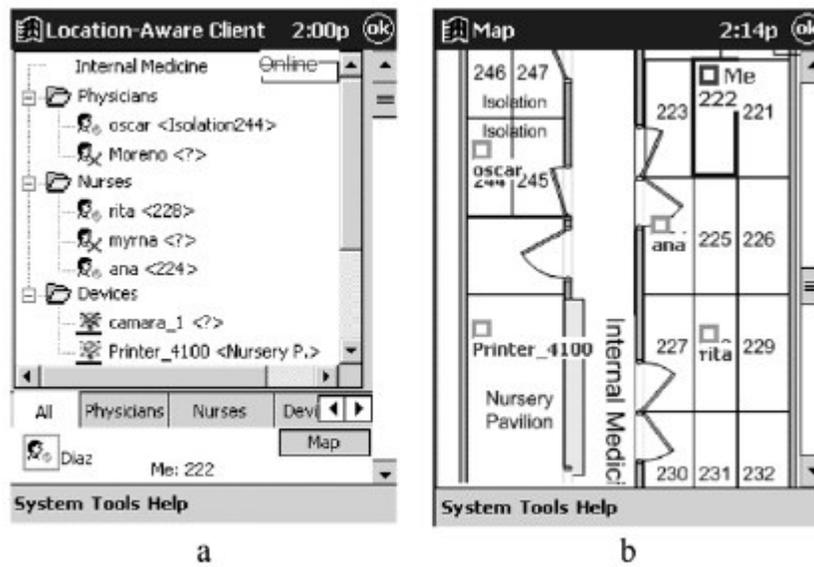


Figura 2.3: Context-aware client indicando a presença de usuários como (a) lista ou (b) mapa (MUÑOZ, RODRIGUEZ, et al, 2003).

HIS (Hospital Information Systems): sistema que gerencia e armazena os registros clínicos dos pacientes, assim como dados relevantes ao ambiente (exemplo, qual o índice de ocupação de cada área do hospital).

HIS Agent: proporciona acesso às informações contidas no HIS. Insere regras para tratar as informações de forma que as mesmas sejam disponibilizadas de acordo com os interesses do usuário, considerando sua localização e função específica.

Location-Estimation Agent: agente que avalia a potência do sinal de diversos pontos de acesso, de forma que possa determinar a posição do usuário.

Agent Broker: permite a comunicação entre agentes, os quais representam os usuários, serviços e dispositivos, utilizando linguagem XML (*Extended Markup Language*).

Context-Aware Agent: toda informação contextualizada é repassada a este agente, o qual monitora o ambiente com a finalidade de avaliar em quais condições encontra-se este cenário, com objetivo de validar as requisições efetuadas.

2.2.4 WARD-IN-HAND

O projeto WARD-IN-HAND (ANCONA, DODERO, et al, 2000), consiste em um estudo desenvolvido na Itália, em conjunto entre setor privado, universidade e alguns

hospitais selecionados. Visa explorar redes sem fio e computadores móveis dentro de diferentes departamentos hospitalares.

Este sistema prevê que informações sejam levadas, através de *handhelds*, até os pontos de atendimento, de forma que possam disponibilizar as mais avançadas aplicações no processamento de registros clínicos. Possui como principais características a capacidade de tolerância à falhas, utilizando ferramentas baseadas em canetas e reconhecimento de voz, para facilitar a liberdade de movimentos dos profissionais. Outra propriedade do sistema é utilizar a heterogeneidade de recursos disponíveis para reduzir custos e torna-lo mais compatível com outros sistemas já existentes. Para tanto, utiliza XML na gerência dos dados. Finalizando sua estrutura principal, agregam-se mecanismos de segurança e privacidade, de forma que apenas usuários autorizados tenham acesso a dados distintos do sistema, de acordo com suas funções e responsabilidades. Abaixo melhor elucidadas as situações acima citadas:

Forma de Interação baseada em Voz: tornar a voz uma forma comunicação entre o usuário e o computador trás maior naturalidade as interações. Neste caso, a segurança do sistema deve ser prevista em dois aspectos, o correto reconhecimento das palavras e a confirmação se a mensagem foi dita por uma pessoa autorizada. Nestes aspectos, a voz deve ser transformada em comandos para ativar funções específicas de determinadas aplicações clínicas. Profissionais devem falar ao invés de escrever tratamentos ou exames. Dependendo do contexto de uso, mecanismos podem excluir vozes que estão no ambiente, por exemplo, vozes de enfermeiros quando os médicos estão prescrevendo um tratamento.

Forma de interação baseada em Caneta digital: neste caso, os usuários interagirão com o sistema usando canetas específicas em sistemas *touch-screen*. Isto será empregado de forma alternativa às interações baseadas em voz.

Segurança nas interações: considera-se importante dentro do projeto, prever redundância de hardware dentro da rede. A identificação do usuário deve ser realizada através de um *login* inicial, por reconhecimento de voz ou através de um *smart card*. Quando o usuário estiver conectado, somente operações de suas área de atuação serão permitidas (por exemplo, médicos podem adicionar e remover prescrições, enfermeiros não).

Registro de informações: os registros efetuados, eletronicamente, são transmitidos para uma base de dados, não podendo, posteriormente, serem removidos ou modificados. Assim, dados sobre pacientes podem ser utilizados como forma de responsabilizar tanto profissionais, quanto o hospital em casos de negligência.

Confidencialidade das informações: naturalmente informações médicas são tratadas como extremamente confidenciais, assim, algoritmos de criptografias devem ser utilizados como forma de manter os dados, que circulam dentro da rede, protegidos.

No próximo capítulo será apresentado um projeto para construção de um Agente que visa trazer algumas das funcionalidades previstas pelos projetos citados, para junto dos sistemas de Gestão Hospitalar do HCPA.

3 AGENTE GESTOR DE CONTEXTO HOSPITALAR (AGCH)

O agente AGCH tem como principal objetivo fornecer informações de contexto relacionadas ao cotidiano de um complexo hospitalar, como citado na seção 1.1. No caso específico do presente trabalho, este agente está focado no modelo utilizado pelo Hospital de Clínicas de Porto Alegre (HCPA). Dentro da cultura de informática utilizada pelo HCPA encontram-se sistemas direcionados a inúmeras funcionalidades. São aplicativos que auxiliam na gestão e funcionamento diário do hospital, fazendo com que procedimentos se tornem mais dinâmicos e eficientes. Dentre as funções realizadas pelo sistema, são previstas rotinas para arquivar digitalmente informações gerais, como por exemplo, prontuários de pacientes e registros de todos os atendimentos efetuados. Além de prever integridade, autenticidade, disponibilidade, privacidade e auditabilidade a todas as informações adicionadas ao sistema (CAYE, 2005). Além das funções tradicionais previstas pelo sistema, o HCPA inova para a área de gestão hospitalar no Brasil, implementando o atendimento a beira do leito, através do Prontuário Eletrônico do Paciente (PEP), acessado por meio de dispositivos móveis. Na próxima seção serão apresentados detalhes do sistema de atendimento a beira do leito.

3.1 HCPA – Atendimento a beira do leito

Inicialmente todas as informações do prontuário do paciente eram gerenciadas de forma não digital, em papel, tornando-se um agravante na administração das mesmas. Nessa forma de manipulação de informações, era necessário o desarquivamento, pelo Serviço de Arquivo Médico e Informações de Saúde (SAMIS), de aproximadamente 2.300 prontuários diariamente. Devendo ainda, após a sua utilização, todos os arquivos serem armazenados novamente. Assim os procedimentos além de trabalhosos, se tornavam mais lentos e suscetíveis a erros.

Posteriormente, com intuito de dinamizar as relações médico-paciente, em meados de 2002, as informações passam a ser gerenciadas de forma digital, podendo ser administradas por acesso local em *desktops*, através do PEP. Este sistema foi desenvolvido em conjunto por profissionais da área da saúde, administradores e analistas do sistema (KLUCK, GUIMARÃES, et al, 2006). Na figura 3.1 tem-se a ilustração de ambos estados de manipulação dos prontuários dos pacientes, arquivos em papel e digitais.

O PEP tem como objetivo, além de facilitar as interações dos profissionais da área saúde, registrar os atendimentos efetuados de forma completa, legível, trazendo maior qualidade aos registros efetuados. Proporcionando também, a identificação de autoria

das informações armazenadas, bem como, privacidade dos dados dos pacientes. Dessa forma, oferecendo maior segurança, visto que, a disponibilidade permanente das informações junto ao paciente, apóia continuamente a assistência prestada, possibilitando a eliminação de erros médicos por falta de dados, ou ambigüidade nas informações dos prontuários de atendimento.

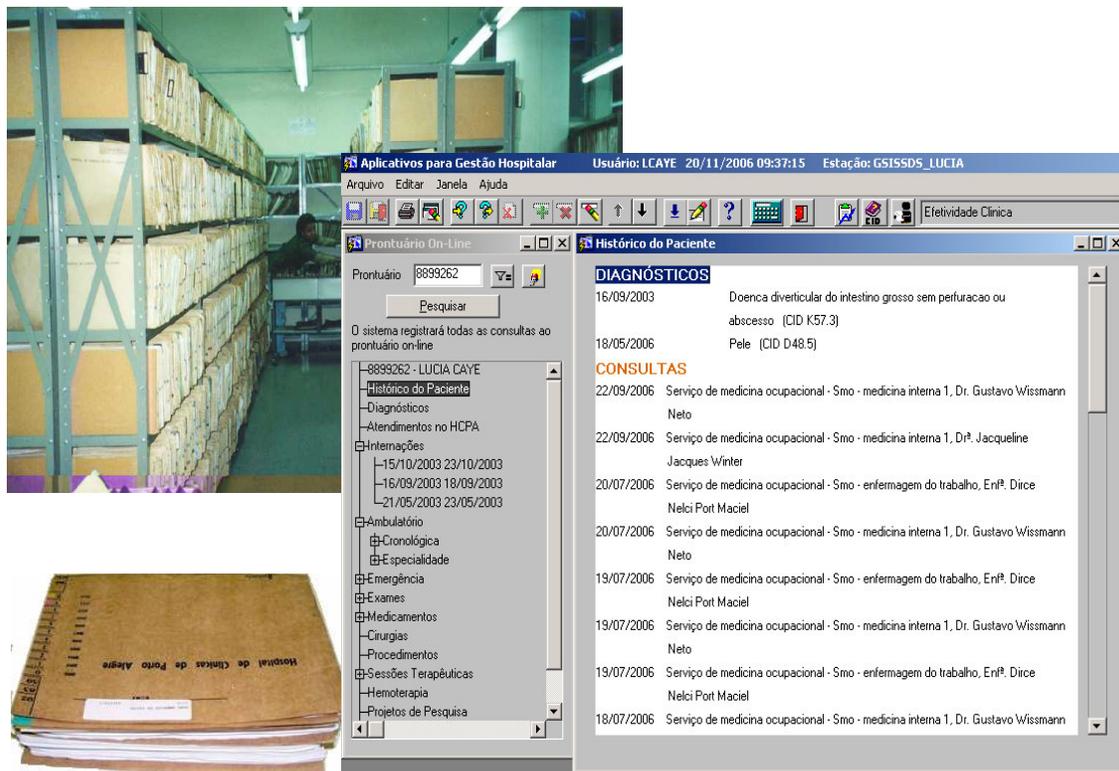


Figura 3.1: Prontuário do Paciente em papel e eletrônico (PEP) (ZIRBES, 2007).

Em um terceiro momento, visando garantir maior segurança ao paciente, as informações inicialmente acessadas de forma local, passam a ser visualizadas em dispositivos móveis como PDA's. Assim, o médico durante seus procedimentos de rotina, substitui as convencionais anotações feitas em papeis, por arquivos digitais já contendo as informações dos pacientes em seus dispositivos móveis, posteriormente inseridas ao prontuário eletrônico do paciente. Possibilitando assim, que consultas ao PEP ou inserções de anotações sejam feitas durante o atendimento, evitando equívocos na transferência de dados, perdas ou trocas de informações dos pacientes

Considerando as limitações apresentadas pelos disponíveis móveis, ressalva-se que as informações gerenciadas por eles, são as mesmas presentes nos *desktops*, no entanto, são apresentadas de maneira diferente, de forma mais limitada, conforme ilustrado pela figura 3.2.



Figura 3.2: Atendimento a Beira do Leito do Paciente (ZIRBES, 2007).

Na situação acima apresentada, nota-se que o Dr. João da Silva deve conectar-se ao sistema, logo após ele recebe a relação de pacientes internados sob seus cuidados, assim, quando desejado, lhe é possível selecionar o paciente e verificar suas informações durante o atendimento. Podendo dessa forma, averiguar medicações, visualizar exames, analisar histórico, fazer anotações de rotina, para futuramente serem repassadas ao sistema.

Dados os aspectos acima mencionados, verifica-se que apesar dos inúmeros benefícios trazidos aos médicos e pacientes através do atendimento a beira do leito, nota-se a possibilidade de torná-lo mais abrangente. Criar um cenário no qual os profissionais da área da saúde possam gerenciar de forma mais dinâmica as informações do PEP, dentro de um cenário pervasivo, gerando uma nova visão aos aplicativos de gestão hospitalar. Assim propõe-se um ambiente no qual situações, tais como, o sincronismo inicial dos dispositivos móveis não seja mais necessário, visto que, o sistema tem a capacidade de identificar a localização do usuário, e direcionar ao mesmo as informações daquela localidade que são de seu interesse, ou seja, os sistemas de gestão hospitalar passam a ser conscientes do contexto e executam com maior interação em relação a seus usuários. Na seção 3.2 será apresentada uma modelagem que visa agregar aos sistemas de gestão do HCPA, funcionalidades que auxiliem a torná-los pervasivos e conscientes de contexto.

3.2 Modelagem

O AGCH integra-se à aplicação de gestão do HCPA com o objetivo de ampliar sua capacidade e auxiliar no melhor funcionamento da instituição de um modo geral. Tem como foco principal o atendimento a beira do leito. O agente, através de seus atributos, deve permitir que funcionários da área da saúde, além de visualizar informações contidas no prontuário do paciente e inserir anotações, tenham a possibilidade de interagir com o sistema, solicitando recursos (um técnico em enfermagem solicita que sejam encaminhadas luvas cirúrgicas para seu setor), serviços (um médico solicita que um técnico em enfermagem se encaminhe a sua localidade, com a finalidade de trocar curativos de um paciente, enquanto ele segue para sua próxima rotina), ou ainda outros usuários (um enfermeiro solicita, pelo sistema, que um técnico em enfermagem se dirija a sua localidade para auxiliá-lo com a remoção de um paciente). Para tanto, não é

necessário que os funcionários informem em qual local encontram-se dentro do cenário, tendo em vista que, o sistema deve localizá-los dinamicamente, conforme sua movimentação dentro da instituição, com o auxílio sensores. Cada pessoa presente no ambiente deve portar consigo algum dispositivo que permita que os sensores identifiquem sua presença, para assim determinar sua precisa localização.

A comunicação entre funcionários e o sistema é realizada através de redes sem fio, as quais permitem que os dispositivos móveis enviem e recebam informações. No entanto, existe a possibilidade de comunicação com dispositivos de menor capacidade de processamento, que não suportam a tecnologia *wireless* (celulares convencionais), por meio de mensagens de texto (o sistema envia uma mensagem de texto comunicando a um funcionário de serviços gerais que ele deve se dirigir ao quarto 403). Destaca-se que na modelagem do AGCH aspectos como a comunicação entre o sistema e o agente serão abstraídos, assim, as interações entre usuários e o AGCH serão demonstradas de forma direta. No entanto, quando implementado, a comunicação deve passar pelos sistemas de gestão hospitalar.

As principais características e funcionalidades previstas pelo AGCH são tratadas no Diagrama de *UseCase*, ilustrado na figura 3.3.

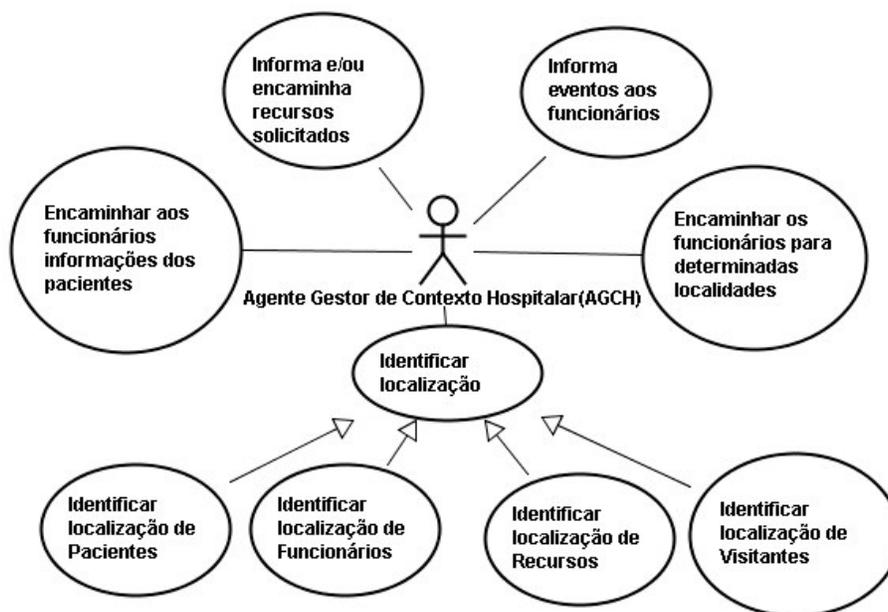


Figura 3.3: Funcionalidades do AGCH.

O *UseCase Identificar localização* tem como função armazenar as informações capturadas pelos sensores, para que cada pessoa presente dentro do cenário seja localizada. Dessa forma, o sistema tem como definir quais informações e/ou tarefas devem ser encaminhadas a cada indivíduo que faz parte do contexto de uma determinada localidade. Além disso, as posições dos recursos também devem estar cadastradas, afim de que, suas localizações sejam identificadas.

O *UseCase Encaminha os funcionários para determinadas localidades* é ativado de duas formas:

- **Primeira:** quando o sistema identifica que existem tarefas ou eventos a serem direcionadas a determinada pessoa, essa situação é ilustrada pelo diagrama de seqüência a seguir, figuras 3.4.

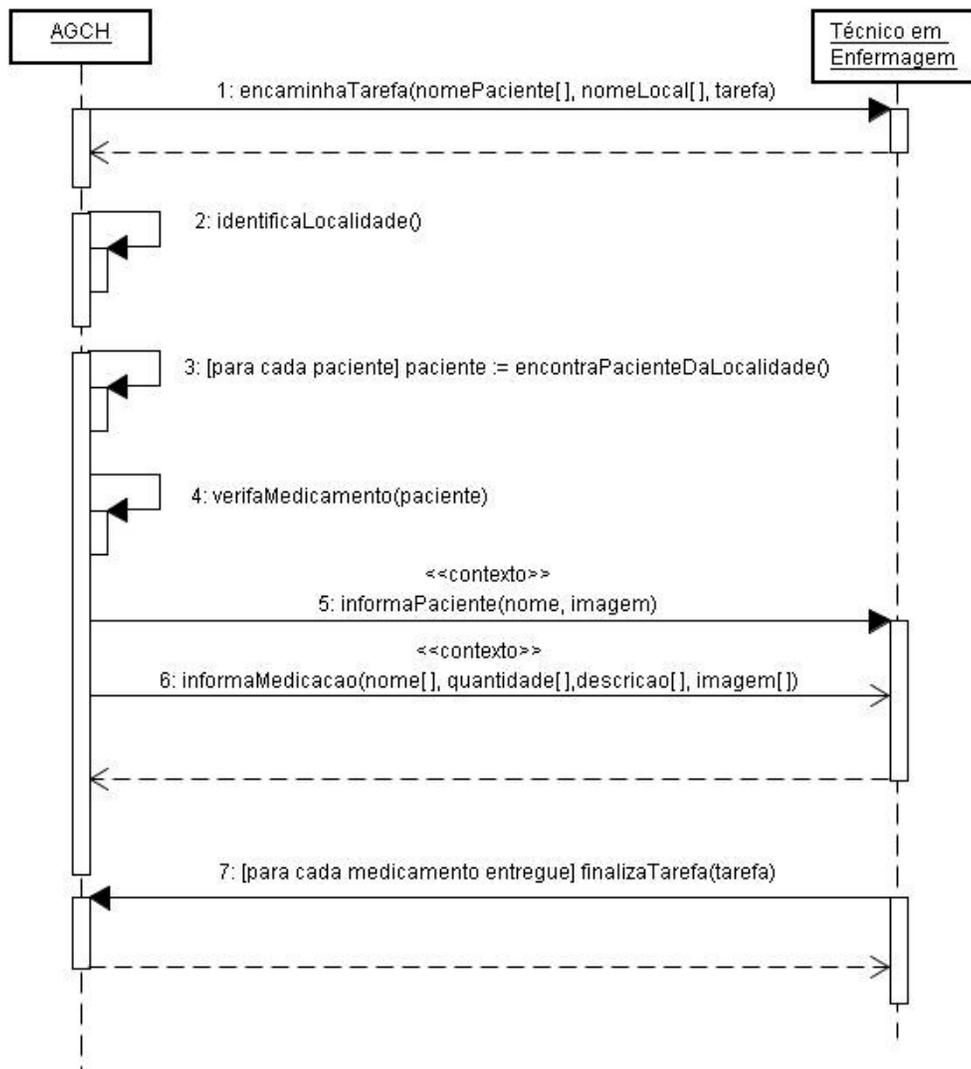


Figura 3.4: Encaminha Tarefa.

No Diagrama de Seqüência Encaminhar Tarefa, figura 7, verifica-se a seqüência de acontecimentos no momento de ministrar medicações aos pacientes. O AGCH de posse das informações dos prontuários com horários nos quais os pacientes devem receber o tratamento, encaminha ao técnico em enfermagem de uma determinada localidade, a tarefa com os pacientes que devem ser medicados naquele momento, bem como suas localidades. Assim que a tarefa encaminhada for aceita pelo técnico em enfermagem, o sistema passa a gerenciar com maior freqüência sua localização exata. Quando ele entra em determinada localização, onde um paciente deve receber medicamentos, é apresentado, em seu dispositivo móvel, um relatório que informa o nome do paciente, sua foto, e a prescrição dos medicamentos relacionados. Com isto, eliminam-se erros em casos de ambigüidade de informações, como a ocorrência de dois pacientes homônimos receberem medicações trocadas, caso outros elementos, como o número do leito, passem despercebidos. Encerrando o procedimento, para cada paciente medicado

o técnico deve sinalizar ao sistema o cumprimento da tarefa, caso contrário, também deve informar que a mesma não pôde ser concluída. Dessa forma, evita-se que o sistema entenda que um paciente já medicado não tenha sido atendido, e encaminhe novamente esta tarefa a outro funcionário.

- **Segunda:** ocorre quando um funcionário requisita a presença de outro em uma determinada localidade, ou então, algum funcionário solicita uma tarefa, esta é repassada ao sistema e o mesmo a delega a pessoa mais indicada, sendo esta situação ilustrada pela figura 3.5.

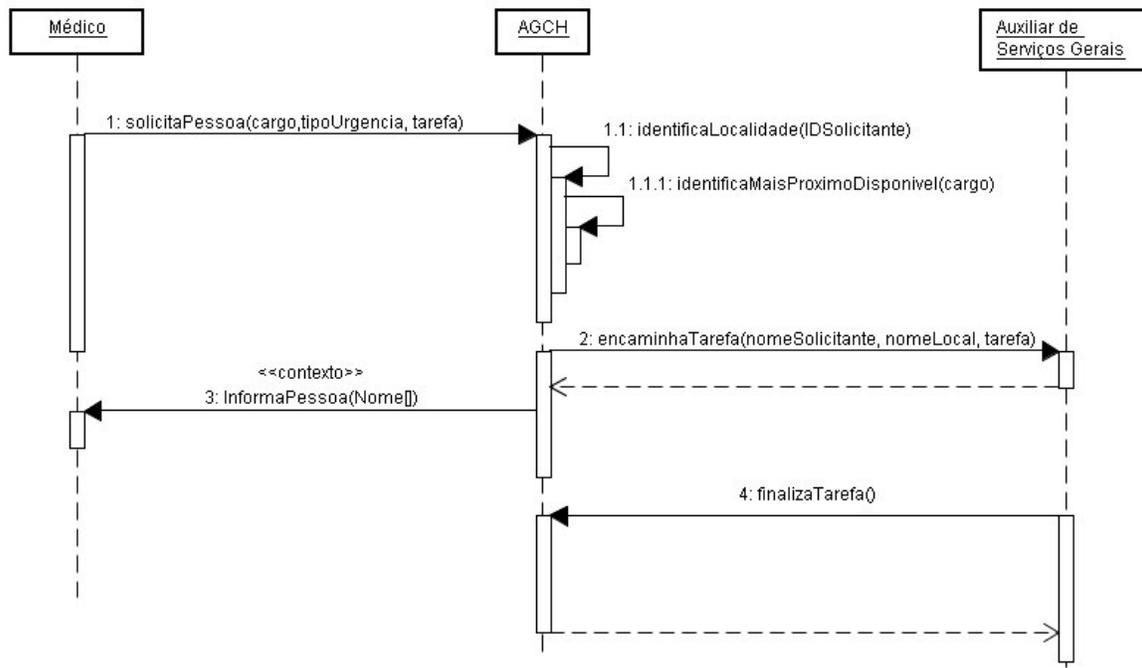


Figura 3.5: Solicita Tarefa.

No Diagrama de Seqüência Solicitação Tarefa, figura 3.5, têm-se as etapas do processo no qual um funcionário solicita a presença de outro para executar uma determinada tarefa. Neste caso, um médico faz a solicitação de uma tarefa ao sistema, passando o perfil da pessoal da qual necessita, através do cargo, qual a urgência e a tarefa que deve ser executada. O sistema identifica a localização do solicitante, para que assim possa definir qual o profissional qualificado e disponível para tarefa, encontra-se mais próximo a ele. Logo após, o sistema encaminha a tarefa ao funcionário selecionado, seqüência representada anteriormente na figura 3.4. Se a tarefa é aceita, será encaminhado ao solicitante uma mensagem informando quem será responsável pela execução da mesma.

No **UseCase Informa evento aos funcionários**, o sistema verifica a agenda de eventos e os comunica aos funcionários relacionados, por exemplo, informa aos médicos residentes que está ocorrendo um estudo de caso no centro cirúrgico. Já o **UseCase Informa e/ou encaminha recursos solicitados**, é ativado quando um funcionário solicita um recurso, então o sistema averigua sua localização e repassa essa informação ao solicitante, ou então, delegada uma função para que algum funcionário se dirija a localidade a fim de executar a solicitação, por exemplo, um técnico em enfermagem solicita seringas descartáveis, caso estejam disponíveis próximas ao local

em que se encontra o técnico, ele recebe esta informação, ao contrário, um funcionário do almoxarifado é acionado.

Fechando as aplicações do AGCH, temos o *UseCase Encaminha aos funcionários informações dos pacientes*, que foca principalmente no atendimento a beira do leito. Tem como objetivo repassar os prontuários dos pacientes aos funcionários da área médica. Cada funcionário deve receber os PEPs dos pacientes que estão relacionados a ele, conforme representado na figura 3.6. Essas informações não são apenas visualizadas, novos dados podem ser inseridos de acordo com o andamento do atendimento, devendo estar disponíveis de forma dinâmica no sistema. Dessa forma, se o resultado de um exame ficar pronto e for inserido no PEP no exato momento em que o médico está atendendo ao paciente, essa informação é atualizada automaticamente em seu dispositivo. Os dados são atualizados no momento em que ocorrem, não mais apenas quando os médicos sincronizam manualmente seus equipamentos.

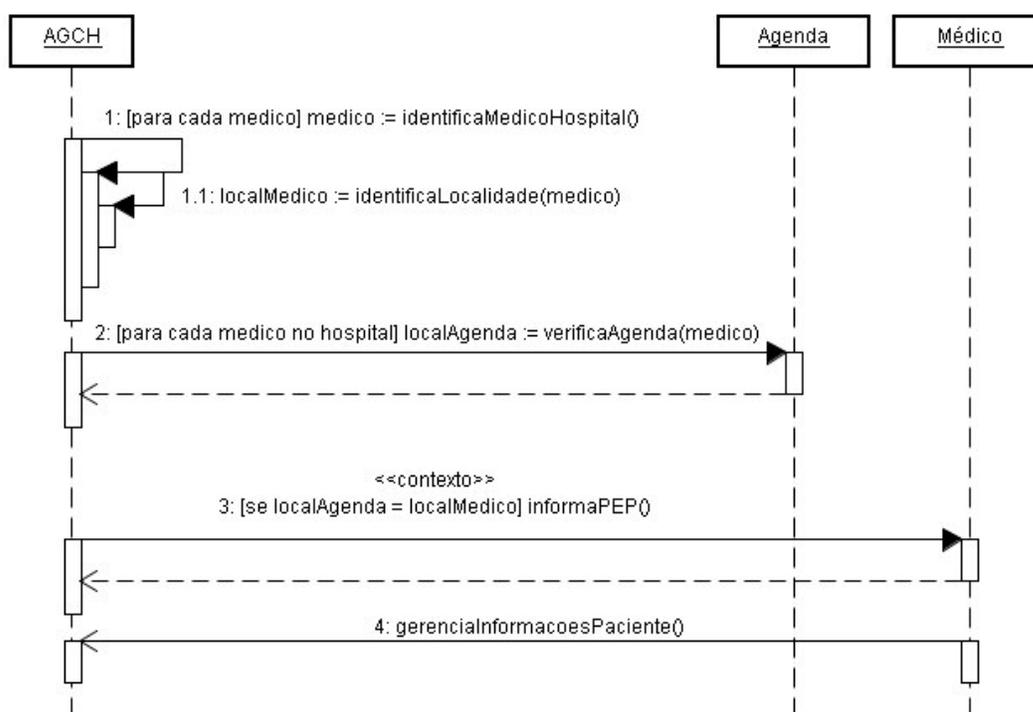


Figura 3.6: Gerência informações do paciente.

Na seqüência representada pela figura 3.6, o AGCH verifica todas as pessoas presentes no hospital, neste caso específico, identifica os médicos que estão na instituição. Posteriormente, verifica a localidade de cada um deles, para então, comparar as informações da agenda com as localidades de cada médico. Caso a localidade de algum item da agenda seja a mesma em que o médico se encontra, informações deste contexto lhe serão encaminhadas, por exemplo, o médico recebe o PEP de cada paciente de sua responsabilidade, em seu dispositivo. Assim, o médico pode gerencia-lo de acordo com seu deslocamento.

Para operacionalizar as funcionalidades descritas na modelagem do AGCH, e representar a interligação estática entre seus módulos, utiliza-se o diagrama de classes Agente Gestor de Contexto Hospitalar, conforme figura 3.7.

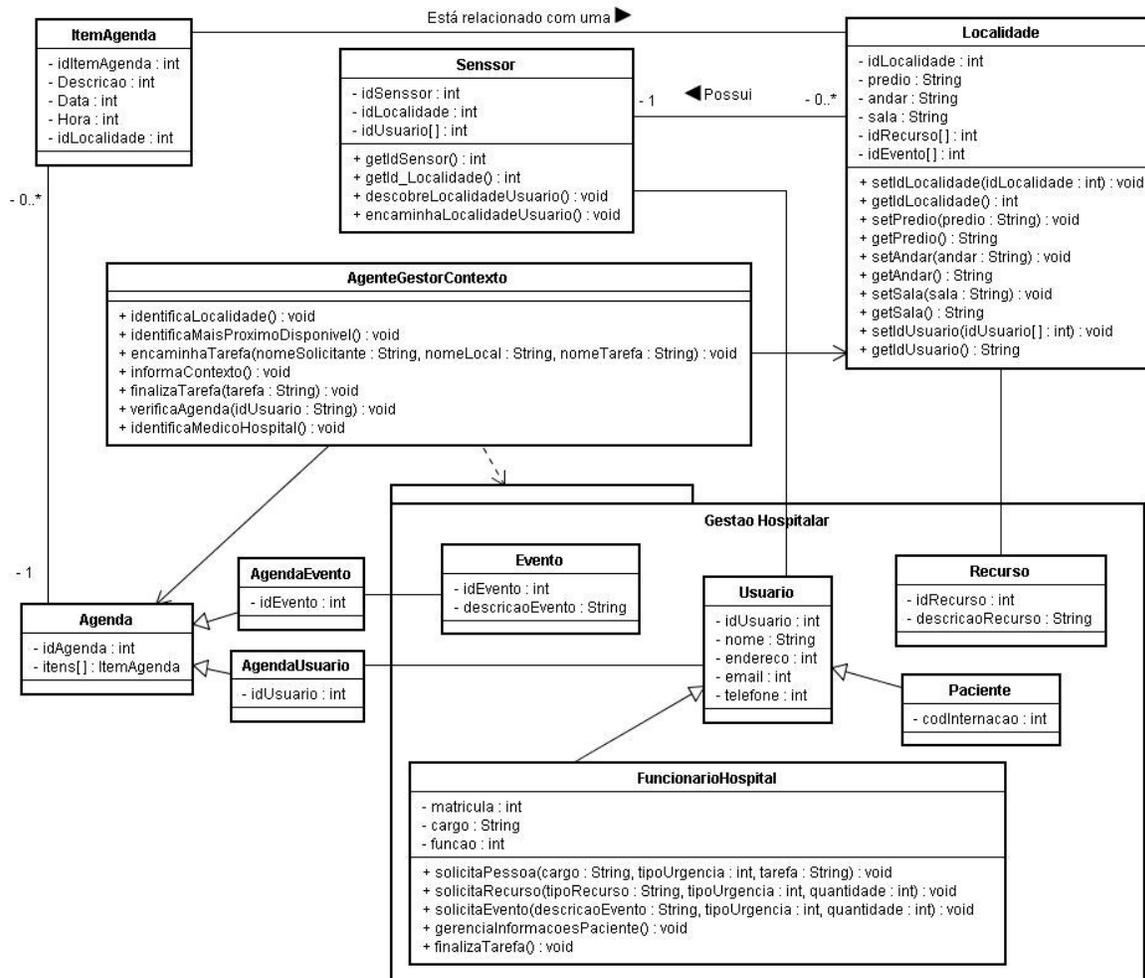


Figura 3.7: Agente Gestor de Contexto Hospitalar.

Considerando as funcionalidades descritas para o AGCH, destaca-se que, é necessário que os módulos e informações já existentes nos sistemas de gestão hospitalar do HCPA operem em conjunto com o AGCH, para que suas aplicações ocorram na maneira prevista. Pressupõe-se que as informações e ligações indispensáveis ao funcionamento do AGCH são os Módulos referentes aos Usuários, Eventos, Pacientes e Funcionários do hospital, considerando que todas as classes referentes a eles, estão agregadas no pacote Gestão Hospitalar. Ressalva-se que a arquitetura, tipos de informação, hierarquia, entre os outros elementos contidos nessas classes não foram explorados pelo presente estudo. Visto que, o acesso às mesmas não foram disponibilizadas integralmente pelo hospital.

No intuito disponibilizar os sistemas de gestão hospitalar ao maior número de usuários possíveis, propõe-se uma forma de adaptação da aplicação, fazendo com que ela atinja uma maior diversidade de dispositivos móveis possíveis, por exemplo, celulares, *smartphones*, PDA's, etc. No capítulo 4 é tratada uma proposta de adaptação da aplicação.

4 ADAPTAÇÃO DA APLICAÇÃO

A adaptação, vista como um dos pré-requisitos da Computação Pervasiva, conforme seção 2.1, deve prover adequação, tanto no que se refere à aplicação, quanto ao sistema. Na primeira delas, a aplicação que estiver em execução no ambiente pervasivo, deve moldar-se às necessidades do usuário, independente do tipo de dispositivo computacional que o mesmo esteja utilizando. Ou ainda, caso o usuário queira uma forma diferente de comunicação com a aplicação. Para elucidar essa adaptação, pode-se imaginar um usuário que está lendo um texto em seu PDA e subitamente alguém o chama para uma reunião em outra cidade, interrompendo sua leitura. Digamos que ele tenha que dirigir seu automóvel até a outra cidade, não podendo mais visualizar o texto contido inicialmente no PDA. Assim, quando o usuário entra em seu veículo, que passa a ser seu novo dispositivo, basta solicitar a aplicação e a mesma se adapta a nova situação, sendo o texto reproduzido nos alto falantes do automóvel.

Já a segunda, adaptação de sistema, refere-se ao fato do dispositivo se adaptar ao contexto, por exemplo, diminuindo o processamento para economizar bateria, ou trocando uma vídeo-conferência por uma conversa por áudio, ocasionada pela diminuição da intensidade de sinais de rede. Ou ainda, suspender determinado serviço, quando o usuário não utilizá-lo efetivamente, colocando-o em espera. Neste trabalho apenas a adaptação da aplicação será tratada.

4.1 Adaptação da Aplicação

Tendo em vista a diversidade de dispositivos existentes, como citado anteriormente, constata-se também, uma grande heterogeneidade de sistemas operacionais presentes nestes equipamentos. Dessa forma, construir aplicações baseadas em uma única plataforma, é extremamente relevante para homogeneização e interoperabilidade entre dispositivos. Além disto, permitir que a aplicação seja disponibilizada com diferentes recursos, de forma adaptada, também gera maior agilidade e flexibilidade nas interações do usuário. Este estudo sugere uma forma de suprir tais necessidades.

Atualmente uma das formas de comunicação mais utilizadas no campo da computação pervasiva é a linguagem Java. Por questões de portabilidade, esta linguagem opera através de uma máquina virtual que se divide em três plataformas, sendo elas: J2SE (*Java 2 Platform, Standard Edition*), utilizada em dispositivos como *desktops* e *notebooks*, J2ME (*Java 2 Platform, Micro Edition*) para dispositivos móveis como PDA's e celulares e J2EE (*Java 2 Platform, Enterprise Edition*) para servidores em geral (SUN,2007). Esta estrutura apesar de facilitar a portabilidade das aplicações, gera certo desconforto para programadores, haja vista que, apesar de sua composição básica ser fundamentalmente a mesma, as três plataformas possuem características

específicas, as quais dificultam, ou até mesmo excluem a possibilidade de reutilização de aplicações desenvolvidas em cada uma delas. Outra dificuldade encontrada nas operações que empregam esta linguagem, consiste no fato de que sua máquina virtual já deve estar previamente instalada no dispositivo do usuário, para que aplicativos possam ser utilizados.

Ponderando sobre estes aspectos, sugere-se que para sanar as dificuldades encontradas, principalmente no aspecto da máquina virtual não encontrar-se instalada no dispositivo, sejam utilizados apenas *browsers* (navegadores web), frequentemente encontrados nos dispositivos utilizados pelos usuários, tendo em vista que, já vêm previamente instalados em equipamentos como PDAs, *smartphones*, *desktops*, celulares, etc.

Os *browsers*, apesar de aceitarem diferentes linguagens de acordo com sua versão, empregam como linguagem básica o HTML (*HiperText Markup Language*). Apesar de ser dotada de regras e sintaxes, ainda assim, é mais simples que outras como Java, Delphi, C++, etc. Ela é utilizada para interligar conteúdos, permitindo a navegação entre informações (SILVA, 2001). Ressalva-se que esta linguagem não permite a execução de aplicações *stand alone* (que não operam sozinhas). Assim, aplicações desenvolvidas em HTML não se adaptam propriamente aos requisitos da computação pervasiva, seção 2.1, a qual requer, entre outros fatores, que o usuário possa interagir com a aplicação mesmo em momentos de desconexão. Dessa forma, surge a necessidade de suprir essa carência, para tanto, propõe-se que em determinados casos, nos quais o HTML não atenda completamente esses critérios, esteja disponível uma segunda opção que supra tais requisitos. A utilização de Ajax (*Asynchronous Javascript And XML*) para o desenvolvimento de aplicações *stand alone* soluciona a deficiência de aplicações criadas apenas em HTML.

O modelo tradicional de aplicação web segue a seguinte seqüência de funcionamento: as interações do usuário são tratadas como solicitações do cliente, o qual dispara requisições HTTP (*HiperText Transfer Protocol*), para o servidor Web e aguarda retorno do mesmo. Após o retorno das informações, as mesmas são convertidas em HTML novamente, e visualizadas pelo usuário através de um *browser*. No entanto, pode-se utilizar aplicações desenvolvidas em Ajax, nesse caso, elas são executadas no próprio navegador (*browser*). Dessa forma, a aplicação libera-se da dependência obrigatória de comunicação com um servidor web, sendo que, para operar com aplicações deste gênero, basta possuir algum dos navegadores modernos, tais como, Mozilla Firefox, Internet Explorer 5+, Opera, Konqueror e Safari.

Ajax não se trata de uma linguagem de programação, mas uma técnica que associa várias tecnologias. Dentre os padrões empregados estão o XML, HTML, DOM (*Document Object Model*), CSS (*Cascading Style Sheets*) e JavaScript, que associados permitem efetuar comunicação assíncrona junto ao servidor. Através do objeto XMLHttpRequest do Javascript, é possível efetuar trocas de informações com o servidor, sem necessariamente ter que atualizar a aplicação, a figura 4.1 ilustra esta situação.

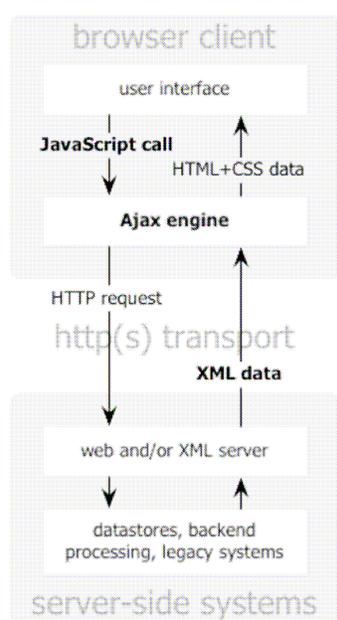


Figura 4.1: Modelo de aplicação do Ajax (GARRET, 2005).

Nesta visão, para adaptação da aplicação, o servidor web deve ter a capacidade de identificar qual dispositivo (PDA, celular, *desktop*, etc) está sendo utilizado pelo usuário, suas características (capacidade de memória, tamanho de tela, cpu, etc), bem como, quais linguagens são suportadas por este equipamento, conforme será tratado na próxima seção. Além disso, o perfil do usuário deve ser identificado, para que as informações sejam ajustadas as suas preferências e de acordo com seu contexto (exemplo, Felipe prefere ler, João ouvir uma informação), mas neste trabalho não serão tratados aspectos relacionados ao modelo de preferência do usuário.

4.2 Identificação do dispositivo do usuário

Nesta seção apresentam-se algumas formas para adaptar-se a aplicação ao dispositivo do usuário. Analisando os critérios citados na seção 4.1, os quais indicam que o servidor web deve ter capacidade de identificar as características do dispositivo, foi desenvolvido um estudo comparativo entre os cabeçalhos HTTP de diferentes equipamentos. Este comparativo tem a finalidade de encontrar as características individuais de cada dispositivo computacional empregado. *Sniffers* foram utilizados como ferramenta na coleta das informações necessárias ao comparativo desenvolvido.

O termo *Sniffer* é utilizado para caracterizar uma ferramenta capaz de analisar os protocolos dos pacotes que passam pela interface de rede de um dispositivo computacional. Todos os pacotes que passam pelo *host* são capturados, e alguns deles são analisados através de determinados padrões, para assim, identificar as informações que estão trafegando pela rede. Dessa forma, pode ser utilizado tanto para fins não lícitos, como capturar senhas e informações bancárias, como para fins lícitos, como identificar as características de dispositivos, sendo essas relevantes a este trabalho.

No processo de análise e seleção das informações, foram empregados dois *sniffers*, Ethereal e CommView. O Ethereal, selecionado pelo fato de ser um dos mais

difundidos, foi utilizado nos primeiros testes de conexão local. Estes experimentos foram desenvolvidos através de um *host* local com a função de servidor web, permitindo assim, apenas conexões locais. Em um segundo momento, foram agregados outros dispositivos, como celulares, PDA's e *SmartPhones*. Para este caso verificou-se a necessidade de ter-se um endereço IP (*Internet Protocol*) fixo válido na internet, permitindo que os equipamentos adicionados pudessem efetuar conexões com o servidor web. Assim, a comunicação entre tais dispositivos foi efetuada com o auxílio de um dos servidores web do Instituto de Informática da UFRGS, o qual possui um IP válido na internet. Para troca de informações entre os servidores foi utilizado o protocolo SSH (Secure Shell), que permitiu o redirecionamento das requisições enviadas ao servidor da UFRGS para o *host* local (servidor web) utilizado de forma cifrada. Surgiram assim, dificuldades pra identificar as informações capturadas pelo Ethereal, tendo em vista que, as mesmas permaneciam criptografadas, não permitindo sua correta compreensão. Este fato fez com que o objetivo inicial, de se obterem as características do dispositivo para definir a melhor adaptação da aplicação, não fosse atingido.

O CommView é aplicado como alternativa de análise das informações não compreendidas corretamente com a aplicação do Ethereal. Este aplicativo permite identificar pacotes maliciosos que trafegam pela rede, endereços desconhecidos, saturação da banda, etc. Possibilita ainda, identificar pacotes que não passam necessariamente pela interface de rede, como conexões locais (*localhost*). Através deste recurso, foi possível descobrir as informações (neste caso não criptografadas) que transitaram entre os servidores.

4.3 Adaptação da aplicação quanto ao Hardware e Software – Exemplos

Visando identificar os tipos de dispositivos utilizados, os mesmos foram divididos em duas classes, dispositivos móveis (celulares, PDA's e *smartphones*) e estáticos (*desktops*, e *notebooks*), ressalva-se que a classificação de *notebooks* como estáticos, tem como objetivo apenas facilitar distinção dos equipamentos analisados no presente estudo. Os exemplos de cabeçalhos HTTP a seguir trazem características distintas dessas subdivisões.

- Cabeçalho HTTP Dispositivos Estáticos:

Exemplo 1:

Tipo de dispositivo: Desktop

Sistema Operacional: Windows XP Professional

Browser: Compatível Mozilla 4.0

Cabeçalho HTTP:

1. GET /CadastroClientes/index.html HTTP/1.1
2. Accept: */*
3. Accept-Language: pt-br
4. UA-CPU: x86
5. Accept-Encoding: gzip, deflate
6. If-Modified-Since: Fri, 05 Oct 2007 23:34:50 GMT
7. If-None-Match: W/"252-1191627290000"
8. User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; Mailinfo [1092156])
9. Host: saloon.inf.ufrgs.br:9090

10. Connection: Keep-Alive
11. Cookie: JSESSIONID=58E71508B5280A4D673B93E7E180F9D5

Exemplo 2:

Tipo de dispositivo: Desktop

Sistema Operacional: Linux Slackware

Browser: Compatível Mozilla 5.0

Cabeçalho HTTP:

1. GET /CadastroClientes/index.html HTTP/1.1
2. User-Agent: Mozilla/5.0 (compatible; Konqueror/3.5; Linux) KHTML/3.5.4 (like Gecko)
3. Accept: text/html, image/jpeg, image/png, text/*, image/*, */*
4. Accept-Encoding: x-gzip, x-deflate, gzip, deflate
5. Accept-Charset: iso-8859-1, utf-8;q=0.5, */q=0.5
6. Accept-Language: en
7. Host: saloon.inf.ufrgs.br:9090
8. Connection: Keep-Alive

- Cabeçalho HTTP Dispositivos Móveis:

Exemplo 3:

Tipo de dispositivo: Smartphone Treo 650

Sistema Operacional: Palm

Browser: Compatível Mozilla 4.0

Cabeçalho HTTP:

1. GET /CadastroClientes/index.html HTTP/1.1
2. Accept: text/html, application/vnd.wap.xhtml+xml, application/xhtml+xml; profile="http://www.wapforum.org/xhtml", image/gif, image/jpeg, image/png, image/bmp, image/vnd.wap.wbmp, application/octet-stream, image/pjpeg, audio/amr, application/MIDI, application/msexcel, application/mspowerpoint, application/msword, application/vnd.ms-excel, application/vnd.ms-powerpoint, application/vnd.palm, application/vnd.rn-realmedia, application/vnd.rn-realsystem-rmj, application/x-pilot, audio/mid, audio/midi, audio/mp, audio/mp3, audio/mpeg, audio/qcelp, audio/vnd.qcelp, audio/wav, audio/x-midi, audio/x-mp3, audio/x-mpeg, audio/x-mpg, image/tiff, image/x-ms-bmp, text/plain, text/x-vCalendar, text/x-vCard, video/3gpp, video/3gpp2, video/avi, video/mpeg, video/x-ms-asf, */*
3. Accept-Language: en, */q=0.8
4. x-wap-profile: "http://downloads.palmone.com/profiles/Blazer400.rdf"
5. Host: saloon.inf.ufrgs.br:9090
6. User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows 98; PalmSource/hspr-H102; Blazer/4.0) 16;320x320
7. Pragma: no-cache
8. Cache-Control: no-cache
9. Connection: Keep-Alive
10. Accept-Encoding: gzip

Exemplo 4:

Tipo de dispositivo: PDA Dell Axim X50

Sistema Operacional: Windows CE

Browser: Compatível Mozilla 4.0

Cabeçalho HTTP:

1. GET /CadastroClientes/index.html HTTP/1.1
2. Accept: */*
3. Accept-Language: pt-br
4. UA-OS: Windows CE (Pocket PC) - Version 4.21
5. UA-color: color16
6. UA-pixels: 320x240
7. UA-CPU: Intel(R) PXA270
8. UA-Voice: FALSE
9. UA-Language: JavaScript
10. Accept-Encoding: gzip, deflate
11. If-Modified-Since: Thu, 04 Oct 2007 01:14:38 GMT
12. If-None-Match: W/"229-1191460478000"
13. User-Agent: Mozilla/4.0 (compatible; MSIE 4.01; Windows CE; PPC; 320x240)
14. Host: saloon.inf.ufrgs.br:9090
15. Connection: Keep-Alive
16. Cookie: JSESSIONID=8BBF6AE53D88719B960F443ED707297C

Exemplo 5:

Tipo de dispositivo: Celular Motorola V360

Sistema Operacional: ?

Browser: MIB 2.2.1

Cabeçalho HTTP:

1. GET /CadastroClientes/index.html HTTP/1.1
2. Host: saloon.inf.ufrgs.br:9090
3. x-wap-profile: "http://motorola.handango.com/phoneconfig/v360/Profile/v360.rdf"
4. accept-language: pt-BR
5. accept: */*, text/x-vcard, text/x-vcalendar, image/gif, image/vnd.wap.wbmp
6. accept-charset: utf-8
7. user-agent: MOT-V360/08.B7.86R MIB/2.2.1 Profile/MIDP-2.0 Configuration/CLDC-1.1
8. accept-application: x-wap-application:wml.ua,x-wap-application:mms.ua
9. Content-length: 0
10. Via: WTP/1.1 nwg2.wgnokia.tim.com.br (Nokia WAP Gateway 4.1/CD15/4.1.93)
11. X-Network-info: GPRS,unsecured
12. X-Nokia-CONNECTION_MODE: TCP
13. X-Nokia-BEARER: GPRS
14. X-Nokia-GATEWAY_ID: NWG/4.1/Build93
15. x-nokia.wia.accept.original:*/*,text/x-vCard,text/xvCalendar,image/gif,image/vnd.wap.wbmp
16. Connection: close

Exemplo 6:

Tipo de dispositivo: Celular Samsung X480

Sistema Operacional: ?

Browser: TSS 1.0

Cabeçalho HTTP:

1. GET /CadastroClientes/index.html HTTP/1.1
2. Host: saloon.inf.ufrgs.br:9090
3. accept: application/vnd.wap.wmlc, application/vnd.wap.wmlscriptc, application/vnd.wap.wbxml, image/vnd.wap.wbmp, image/gif, text/plain, text/html, text/x-vcalendar, text/x-vcard, application/vnd.wap.sic, image/jpeg, image/png, application/vnd.smf, audio/sp-midi, application/x-midi, audio/midi, audio/x-mid, audio/x-midi, audio/mid, text/x-imelody, text/x-imelody, audio/imelody, audio/amr, application/vnd.oma.drm.message, application/xhtml+xml, text/css,

- application/vnd.wap.xhtml+xml, application/vnd.wap.cert-response,
 application/vnd.wap.signed-certificate, application/vnd.wap.hashcd-certificate,
 application/x-wap-prov.browser-settings, text/vnd.sun.j2me.app-descriptor,
 application/java-archive, text/vnd.wap.si, text/vnd.wap.wml, text/vnd.wap.wmlscript
4. accept-charset: ISO-8859-1, US-ASCII, UTF-8; Q=0.8, ISO-10646-UCS-2; Q=0.6
 5. accept-language: pt
 6. user-agent: SAMSUNG-SGH-X480/1.0
 7. x-wap.tod-coded: Thu, 01 Jan 1970 00:00:00 GMT
 8. Via: WTP/1.1 nwg1.wgnokia.tim.com.br (Nokia WAP Gateway 4.1/CD15/4.1.93)
 9. X-Network-info: GPRS,unsecured
 10. X-Nokia-CONNECTION_MODE: CMODE
 11. X-Nokia-BEARER: GPRS
 12. X-Nokia-GATEWAY_ID: NWG/4.1/Build93
 13. x-wap-profile: http://wap.samsungmobile.com/uaprof/SGH-X480.xml
 14. x-nokia.wia.accept.original:
 application/vnd.wap.wmlc,application/vnd.wap.wmlscriptc,application/vnd.wap.wbxml,ima
 ge/vnd.wap.wbmp,image/gif,text/plain,text/html,text/x-vCalendar,text/x-
 vCard,application/vnd.wap.sic,image/jpeg,image/png,application/vnd.smf,audio/sp-
 midi,application/x-midi,audio/midi,audio/x-mid,audio/x-midi,audio/mid,text/x-
 iMelody,text/x-
 imelody,audio/imelody,audio/amr,application/vnd.oma.drm.message,application/xhtml+xml
 ,text/css,application/vnd.wap.xhtml+xml,application/vnd.wap.cert-
 response,application/vnd.wap.signed-certificate,application/vnd.wap.hashcd-
 certificate,application/x-wap-prov.browser-settings,text/vnd.sun.j2me.app-
 descriptor,application/java-archive
 15. Connection: close

Analisando as informações dos cabeçalhos HTTP coletados, verifica-se que através de determinados campos, é possível fazer a distinção entre os dispositivos móveis e estáticos. Nos primeiros exemplos direcionados a dispositivos estáticos, corrobora-se essa definição através das seguintes informações: no exemplo 1 temos na linha 4 a informação referente ao processador utilizado (**UA-CPU:X86**), através da qual pode-se determinar a classificação do dispositivo como estático, visto que, possui arquitetura do processador baseado no intel 8086, o qual não é disponível em dispositivos móveis. Caso esse dado não estivesse disponível, teríamos ainda, uma segunda opção para identificação do dispositivo, na linha 8 (**User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; Mailinfo (1092156))**) verifica-se que o dispositivo em questão utiliza um *browser* compatível com Windows NT 5.1, sistema operacional característico apenas de dispositivos com maior poder de processamento.

No exemplo 2 a informação relevante está na linha 2 (**User-Agent: Mozilla/5.0 (compatible; Konqueror/3.5; Linux) KHTML/3.5.4 (like Gecko)**), na qual tem-se a indicação do sistema operacional Linux. Este sistema pode definir o dispositivo como estático, haja vista que, é utilizado apenas em equipamentos desta classificação.

Partindo para o exemplo 3 começam a surgir as primeiras informações referentes a dispositivos móveis. Neste caso, a linha 6 apresenta a seguinte sentença: (**User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows 98; PalmSource/hspr-H102; Blazer/4.0) 16;320x320**)), dela extrai-se a informação de que a tela deste dispositivo apresenta um resolução de 320x320, específica de dispositivos móveis. Destaca-se que esta informação vem agregada ao *browser* do dispositivo, e que no caso de equipamentos estáticos, a mesma não seria mencionada. Além disto, apesar do *browser* indicar a compatibilidade com o sistema operacional Windows 98, logo após, a sentença apresenta a informação que o dispositivo é compatível com o sistema operacional da Palm, exclusivo de dispositivos móveis.

No exemplo 4 temos uma gama maior de informações que podem ser consideradas. Na linha 4 (**UA-OS: Windows CE (Pocket PC) - Version 4.21**) é informado qual o sistema operacional utilizado no dispositivo em questão. Este equipamento opera com Windows CE aplicado apenas a dispositivos móveis. Na linha 6 (**UA-pixels: 320x240**) a resolução da tela é agregada às informações da sentença, e como citado no exemplo 3, essas informações não seriam mencionadas no caso de dispositivos estáticos, e caso fossem mencionadas, deveriam conter valores em maior escala (exemplo, 800x600). Finalizando, na linha 7 (**UA-CPU: Intel(R) PXA270**) informa o tipo de processador do dispositivo, confirmando que o mesmo encaixa-se no perfil de dispositivos móveis. Na linha 13 encontram-se informações que ratificam as já apresentadas nas sentenças demonstradas.

As informações dos exemplos 5 e 6 podem ser tratadas com os mesmos critérios, visto que, ambos os cabeçalhos foram extraídos de celulares, mesmo sendo de fabricantes distintos, e apresentam perfis em comum. As informações relevantes concentram-se em seu *User-Agent*, linhas 7 e 6 dos exemplos 5 e 6 respectivamente, que indicam o nome dos dispositivos em questão.

De posse dessas informações, já estando apto a reconhecer os dispositivos que estão sendo utilizados pelos usuários, é possível definir algumas adaptações que podem ser realizadas na aplicação. São previstas adaptações tais como: ajustes na aplicação quanto ao tamanho da tela apresentada, ajustes na resolução das imagens encaminhadas (exemplo, reduzir uma imagem de 1200x1200 *pixels* para 120x120 pixels antes de encaminhá-la a um celular), substituição de formatos (exemplo, substituir imagens por texto, ou vídeo por apenas áudio). Sendo que, todas as formas de adaptação devem levar em consideração o perfil do usuário (exemplo, duas pessoas com o mesmo dispositivo tem preferências diferentes, Alice prefere animações e Bob apenas texto).

Além da análise das características de hardware, o servidor web deve verificar também, informações referentes ao tipo de software suportado pelo dispositivo, para que seja determinada com maior precisão, qual forma da aplicação deve ser encaminhada ao usuário. Seguindo os critérios desenvolvidos por este estudo, sugere-se que sejam utilizadas como opções, o envio de aplicações em HTML ou Ajax. Todos os dispositivos, cujos *User-Agent* indicam compatibilidade com Mozilla 4.0 ou superior, podem receber aplicações desenvolvidas em Ajax. Aqueles que não apresentam diretamente esta informação fornecem um link no seu perfil (*x-wap-profile*) com maior detalhamento de suas características. Este link reporta a informações contidas na web a respeito do dispositivo. Através do arquivo disponibilizado por ele, é possível identificar se o equipamento em questão suporta os requisitos básicos para aplicações Ajax. A primeira informação que deve ser localizada neste arquivo apresenta-se na *tag* de descrição, cuja identificação é **BrowserUA..** Nesta descrição é possível verificar se o JavaScript está disponível ou não, descrito pela *tag* **prf:JavaScriptEnabled**. Considerando que este é um elemento fundamental para aplicações Ajax. Na figura 4.2 ilustra-se parte do perfil de um celular Motorola modelo V360, obtido através do link reportado pela linha 3 do exemplo 5, na qual pode-se observar as *tags* mencionadas e definir que o mesmo não suporta aplicações JavaScript. Caso o dispositivo não suporte Ajax, as aplicações serão encaminhadas sempre em HTML.

```

<rdf:Description rdf:ID="BrowserUA">
<rdf:type
rdf:resource="http://www.openmobilealliance.org/tech/profiles/UAPROF/ccppsche
ma-20021212#BrowserUA" />
<prf:BrowserName>MIB</prf:BrowserName>
<prf:BrowserVersion>2.2.1</prf:BrowserVersion>
<prf:FramesCapable>No</prf:FramesCapable>
<prf:TablesCapable>Yes</prf:TablesCapable>
<prf:JavaAppletEnabled>No</prf:JavaAppletEnabled>
<prf:JavaScriptEnabled>No</prf:JavaScriptEnabled>
<prf:DownloadableBrowserApps>
<rdf:Bag>

```

Figura 4.2: Parte perfil Motorola V360.

Avaliando os aspectos mencionados durante o desenvolvimento da modelagem do AGCH, apresenta-se como resultado final um cenário próximo ao ilustrado pela figura 4.3. Nesta situação são mostradas algumas formas possíveis de representação da aplicação no dispositivo do usuário.



Figura 4.3. Distribuição de medicamentos de forma adaptada ao dispositivo.

Analisando a ilustração representada na figura 4.3, identifica-se a seqüência de execução, no dispositivo do usuário, da tarefa de distribuição de medicamentos

exemplificada pelo diagrama de seqüência encontrado na seção 3.2, figura 3.4. No seguimento apresentado tem-se à esquerda a imagem de um celular (Motorola V360). Este é identificado pelo servidor web como não compatível com a tecnologia Ajax, assim, a aplicação lhe é disponibilizada apenas em HTML. Ao centro encontra-se a representação de um PDA (Dell Axin X50), o qual foi identificado pelo servidor web, como habilitado a receber aplicações em Ajax. Finalizando a seqüência, a direita tem-se novamente a representação do mesmo aparelho celular, em uma situação na qual o servidor não teria a capacidade de adaptar a aplicação ao dispositivo, encaminhando a aplicação em Ajax. Destaca-se que, para aplicações conscientes do contexto como sugerido, quando o dispositivo for compatível com Ajax, é sempre mais interessante direcionar a aplicação ao usuário dentro desta tecnologia. Visto que, aplicações desenvolvidas somente em HTML exigem que a mesma seja constantemente atualizada em sua totalidade, a fim de verificar novas informações. Já em Ajax, em intervalos determinados, a aplicação pergunta se existem atualizações a serem feitas, dessa forma, apenas as novas informações serão encaminhadas pelo servidor, economizando processamento e banda de uma forma geral.

5 CONTRIBUIÇÕES DE CONSIDERAÇÕES FINAIS

O propósito final do AGCH é contribuir, junto aos sistemas de gestão hospitalar do HCPA, com o melhor atendimento ao paciente e funcionamento do hospital em aspectos gerais. Proporcionando aos profissionais da área da saúde funcionalidades que permitam um melhor entendimento e integração aos procedimentos da instituição. Automatizando as interações dos usuários através do sistema, como foi proposto, além de melhorar o cotidiano de pacientes e funcionários, o hospital pode fazer uma gerência administrativa sob as rotinas geradas durante as interações. Entre elas, pode-se citar a gestão de pessoal, na qual as tarefas são delegadas de forma setorizada (através da localização do mais próximo disponível) evitando a sobrecarga ou ociosidade dos funcionários. Outra forma de gestão a ser exemplificada é a estatística, através da qual, muitos dados podem ser encontrados.

Destaca-se que o atual sistema de atendimento a beira do leito utiliza em sua estrutura a linguagem SuperWaba, que delimita o uso do PEP, pois não é suportada pela maioria dos aparelhos celulares atuais. Já em aplicações desenvolvidas em HTML ou Ajax, dificilmente algum dispositivo não irá suportá-las.

Para finalizar o presente capítulo, destacam-se algumas possibilidades para o desenvolvimento de trabalhos futuros, tais como, a implementação do AGCH dentro dos sistemas de gestão hospitalar do HCPA, bem como, estender os benefícios aos pacientes e visitantes. Por exemplo, o paciente ao receber um medicamento que causa determinadas reações, recebe informações sobre os tipos de efeitos colaterais gerados pelo mesmo, podendo assim, evitar acionar a equipe médica em alguns casos. Já os visitantes podem receber informações diretas sobre o caminho que devem percorrer para chegar a um determinado local, entre outras possíveis contribuições.

REFERÊNCIAS

ANCONA M. et al. Mobile computing in a hospital: the WARD-IN-HAND project. **Trabalho apresentado no ACM Symposium on Applied computing**, 2000, em Como, Itália.

AUGUSTIN, I. **Abstrações para uma Linguagem de Programação visando Aplicações Móveis Conscientes do Contexto em um Ambiente de Pervasive Computing**. 2004. 194p. Tese (Doutorado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

BARBOSA, D. N. F. **Um Modelo de Educação Ubíqua Orientado à Consciência do Contexto do Aprendiz**. 2007 Tese (Doutorado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

CAYE, L. **Implantação da Tecnologia PDA no HCPA**. 2005. Disponível em: http://www.guors.com.br/documentos_2005/Apresentacao_HCPA_NOV2005.pdf. Acesso em: out. 2007.

FUGETTA, A.; PICCO, G. P.; VIGNA, G. Understanding Code Mobility. **IEEE Transactions on Software Engineering**, New York, v.24, n.5, May 1998.

GARRET, J. J. **Ajax: A New Approach to Web Applications**. 2005. Disponível em : <<http://www.adaptivepath.com/ideas/essays/archives/000385.php>>. Acesso em: set. 2007.

KLUCK, M. M.; GUIMARÃES, J. R.; CAYE, L.; ZIRBES, S. F. Registro Eletrônico do Atendimento Ambulatorial: consolidando o Prontuário Eletrônico do Paciente no Hospital de Clínicas de Porto Alegre. In: CONGRESSO BRASILEIRO DE INFORMÁTICA EM SAÚDE, 10., 2006. Disponível em: <<http://www.sbis.org.br/cbis/arquivos/713.rtf>>. Acesso em: out. 2007.

MUÑOZ, M. et al. Context-aware mobile communication in hospitals. **IEEE Computer**, New York, v. 36, p. 60–67, Sept. 2003.

OGATA, H.; YANO, Y. Knowledge awareness for a computer-assisted language learning using handhelds. **International Journal of Continuous Engineering Education an Lifelong Learning**, [S.l.], v.14, n.4-5, p.435-449, Jan. 2004.

SAHA, D.; MUKHERJEE, A. Pervasive Computing: a Paradigm for th 21st Century. **IEEE Computer**, New York, v.36, n.3, p.25-31, Mar. 2003.

SATYANARAYANAN, M. Fundamental Challenges in Mobile Computing. In: ACM SYMPOSIUM OF PRINCIPLES OF DISTRIBUTED COMPUTING. Philadelphia, Pennsylvania, USA. **Proceedings...** New York: ACM, 1996.

SATYANARAYANAN, M. Pervasive Computing: Vission and Challanges. **IEEE Personal Communications**, New York, v.4, n.8, Aug. 2001.

SILVA, O. J. **DHTML: Estilos e Conteúdo Dinâmico**. São Paulo: Erica, 2001.

SUN MICROSYSTEMS. **Java 2 Software The Platforms**. Disponível em: <<http://java.sun.com/java2/index.html>> Acesso em: set. 2007.

YAMIN, A. C. **Arquitetura para um Ambiente de Grade Computacional direcionado às Aplicações Distribuídas, Móveis e Conscientes do Contexto da Computação Pervasiva**. 2001. Tese (Doutorado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

ZIRBES, S. F. Tecnologias a Serviço do Paciente. In: JORNADA ARGENTINA DE INFORMÁTICA, 39.; SIMPOSIO ARGENTINO DE INFORMÁTICA Y SALUD, 2007. Disponível em: <<http://www.inf.ufrgs.br/~zirbes/Downloads.htm>>. Acesso em: ago. de 2007.